Mark 2 Manual

Complete Version

Lasergraph DSP Version 2024/12/12

LaserAnimation Sollinger GmbH · Crellestr. 19/20 · D-10827 Berlin
Phone +49 (30) 780 963 00 · Fax +49 (30) 780 963 25
E-mail info@laseranimation.com · Internet www.laseranimation.com

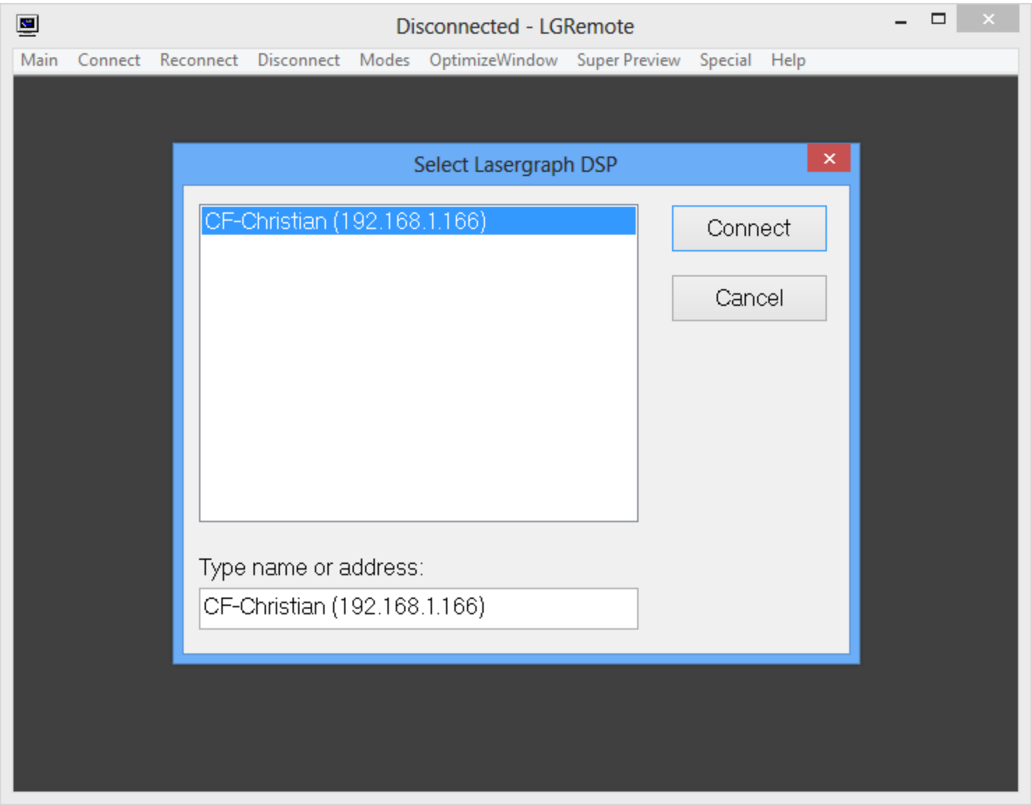
Setting up Remote Control

After connecting the Lasergraph DSP to a power outlet and your network (either an Ethernet switch, or directly to your PC), please install the software LGRemote that came with the Lasergraph DSP (you can also download it from our website www.laseranimation.com) on your PC by double clicking Setup.exe.
The only thing you will be asked for is the auto start of the LGServer which provides Timecode and other I/O information for the Lasergraph DSP. Normally you should choose YES.
After the installation process is finished, please check your network settings.
Call up the Network settings and make sure that the IP-address and subnet mask of the PC fit to the settings of the Lasergraph DSP. Normally the Lasergraph DSP comes with these settings:

IP Address: 192.168.1.xxx
Subnet Mask: 255.255.255.0

If you are not familiar with setting up a network, please contact your system administrator or refer to the chapter → NETWORK Setup.

If all settings are correct you will see the Lasergraph DSP listed after starting LGRemote.

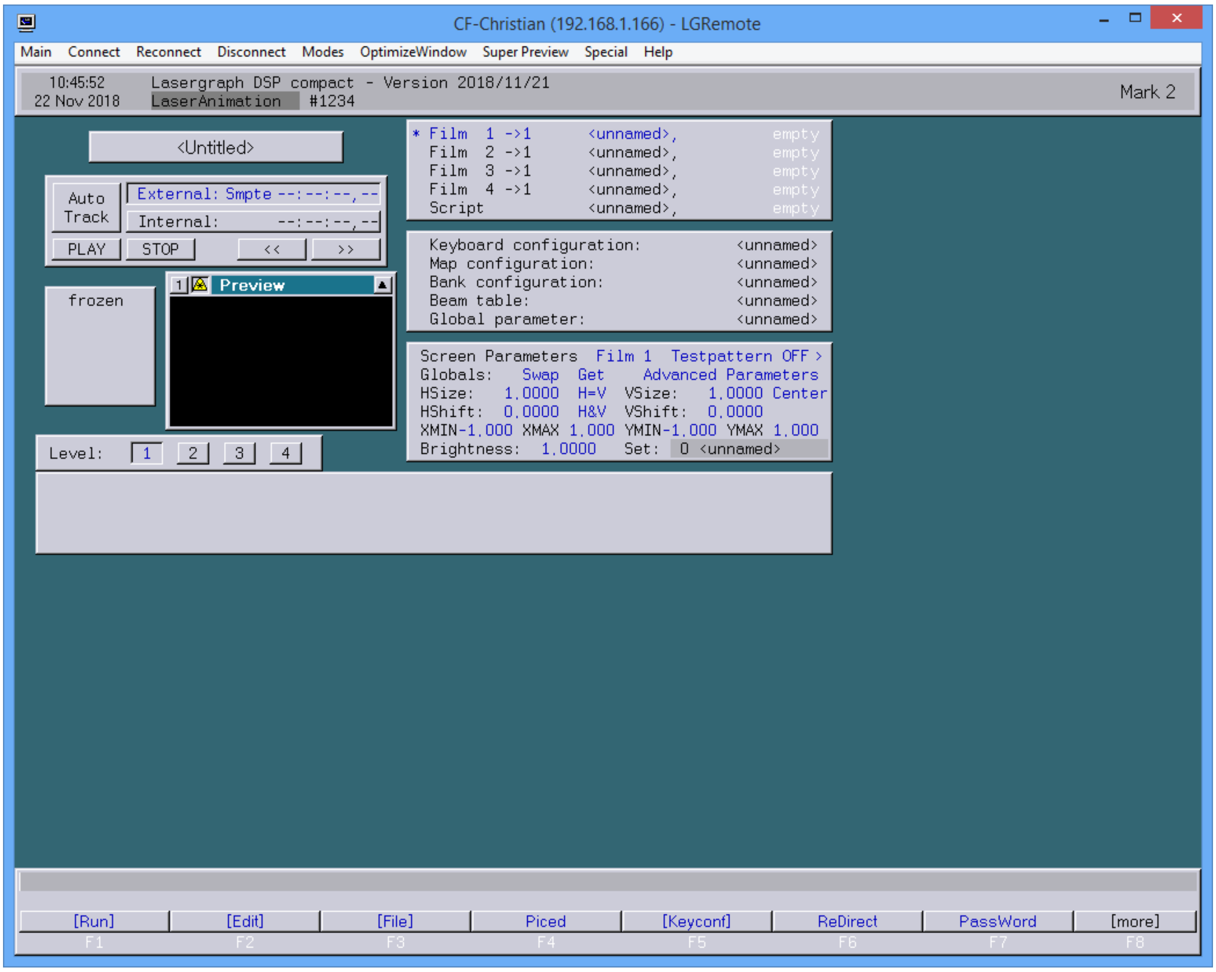


CF-Christian (192.168.1.166)

Type name or address: CF-Christian (192.168.1.166)

Just click onto connect, and you will be connected to the Lasergraph DSP!

Welcome to the Mark 2 !



```
10:45:52      Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018   LaserAnimation   #1234

External: Smpte --:--:--,--
Internal:       --:--:--,--

* Film  1 ->1       <unnamed>,          empty
  Film  2 ->1       <unnamed>,          empty
  Film  3 ->1       <unnamed>,          empty
  Film  4 ->1       <unnamed>,          empty
  Script            <unnamed>,          empty

  Keyboard configuration:            <unnamed>
  Map configuration:                 <unnamed>
  Bank configuration:                <unnamed>
  Beam table:                        <unnamed>
  Global parameter:                  <unnamed>

 Screen Parameters  Film 1  Testpattern OFF >
 Globals:    Swap  Get    Advanced Parameters
 HSize:    1.0000  H=V  VSize:    1.0000 Center
 HShift:   0.0000  H&V  VShift:   0.0000
 XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
 Brightness:  1.0000    Set:  0 <unnamed>
```

Configuring the PC as a Music Server

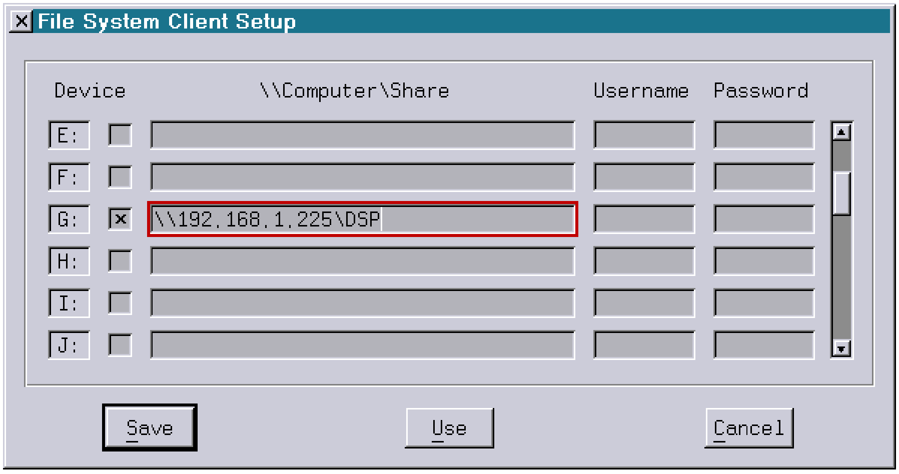
What do you think of watching some of the shows that came with the Lasergraph DSP?
What is a show without music, so, let's copy the music to the PC, so we can listen to it.

If you got the CD that came with the Lasergraph DSP at hand, you can copy the music files onto your hard drive. If not, you have to copy the music files from the internal hard drive of the Lasergraph DSP to your PC.

To do so, we need a directory on the PC that can be accessed by the Lasergraph DSP.
It is recommended to create a new directory. Go to the Windows® Explorer and create a directory, e.g. "DSP" wherever you want, then enable sharing by choosing DSP as the share name.
Then switch back to the LGRemote and go to the Setup of the Lasergraph DSP by typing "SETUP" into the command line or by clicking on the function key.

In the **Setup** you will find a button named "**FS Client"**, which stands for File System Client. Here you have to assign a partition to the shared directory of the PC.

Because there are already 4 partitions, C-D-E-F, choose G: and enable its checkbox.
In the field behind the letter type either the PC-name or its IP-Address (You will find the information in the Network settings) and the share name.
Here is an example of what it should look like:



File System Client Setup

Device		\\Computer\Share	Username	Password
E:	□			
F:	□			
G:	☒	\\192.168.1.225\DSP		
H:	□			
I:	□			
J:	□			

Now choose "**Save**" and exit the **File System Client Setup** by clicking into the left upper corner of the window.

Now you can go into the File system of the Lasergraph DSP by pressing "F3" or typing "File" into the command line.
A window will open showing you all contents of the partition C:.
The red columns you see are directories, the black one is a so called "BOOT" which is a special directory containing all data that are necessary to run a show.

To check if you have access to the shared directory, click into the uppermost part of the File window. Each time you click, the partition changes to the next one from C: to D:, then E:, F: and then to G:. As you can see the directory is empty. Click once again to change back to C:

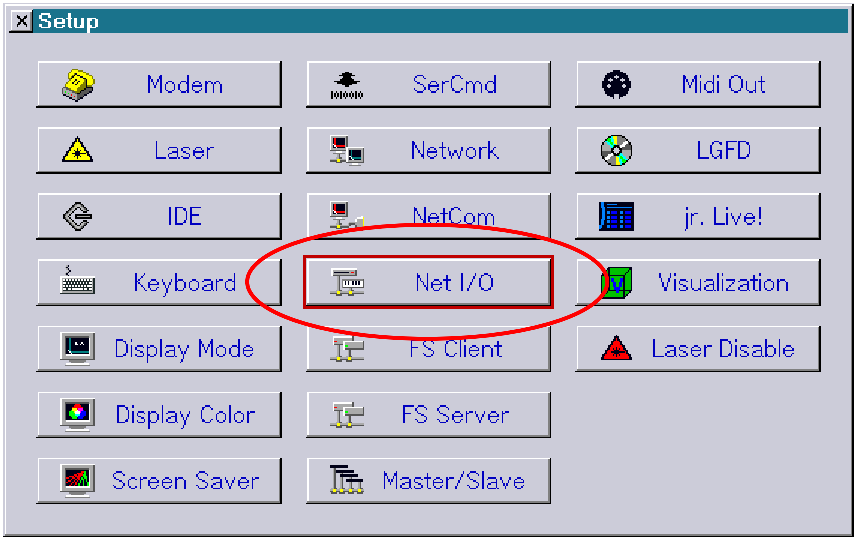
Now click onto the directory "Shows" and then choose the directory "Graphics". To copy the WAV files to the PC type the following or choose the corresponding command in the menu bar.

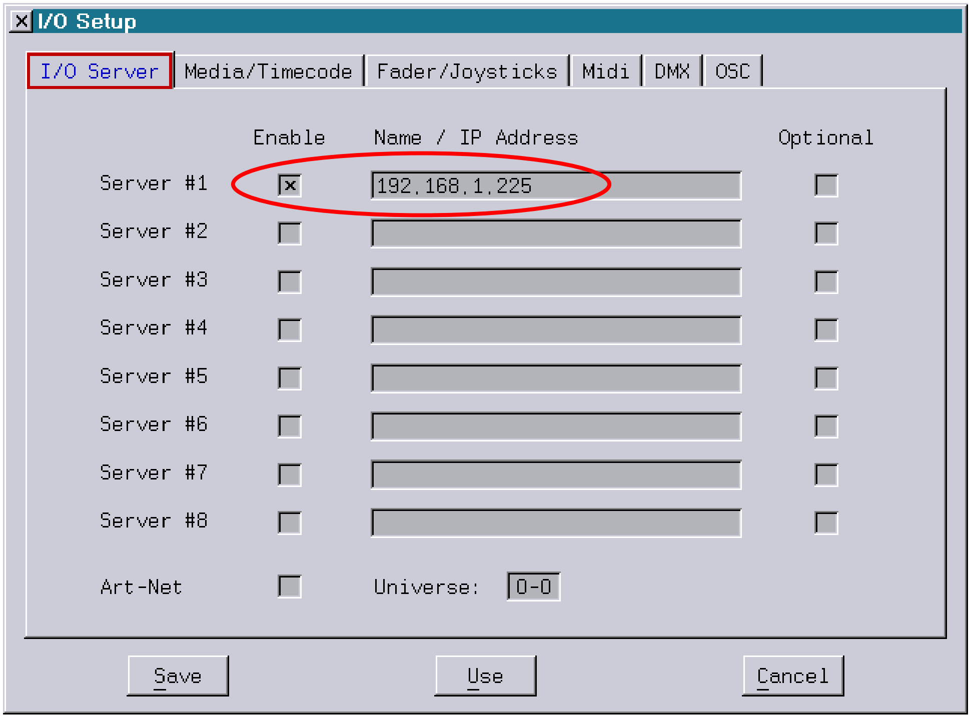
```
Copy *.wav g:
```

This will copy all wave files to the shared directory on the PC, which might take a while over the network.

After this is done, we can prepare the LGServer for "serving" the timecode information to the Lasergraph DSP.
First we have to assign the PC as an I/O server inside the Lasergraph DSP.

Once again go to "**Setup**" and choose "**Net I/O**":



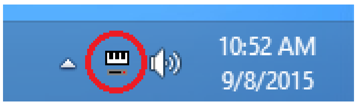


	Enable	Name / IP Address	Optional
Server #1	☒	192.168.1.225	□
Server #2	□		□
Server #3	□		□
Server #4	□		□
Server #5	□		□
Server #6	□		□
Server #7	□		□
Server #8	□		□

Art-Net □ Universe: 0-0

Here you can assign up to 8 I/O servers to the Lasergraph DSP.
For now, we need one Choose the first one, enable the checkbox and type the IP-address or the name of the PC into the field. Now go to the next tab and define what kind of information will be accepted by the Lasergraph DSP: We need the Media, which stands for the Media and all Media Files. Enable the checkbox and leave the **Setup,** saving the settings.

Next is the PC: During installing the LGRemote software you were asked whether to start the LGServer automatically. You will find the program as a small icon (MIDI-keyboard) in the task bar, by clicking on it you open the user interface.



Click "Config" and then enable the check box for WAV/AVI.
Then open the browser, navigate to the directory "DSP" and choose the Wave-file "TheTrickyBook.wav" and close the configuration window with "Save".
Then close the Server window with the button "Hide". Do not choose "Exit" because this would quit the server.

Switching to the LGRemote will show you a new window on the Lasergraph DSP screen that looks familiar, like a media player. By clicking onto the PLAY button you start the playback of the WAV file.
Can you hear the music? Then it's time to load the show into the Lasergraph DSP.

Go to the file menu, you are still in directory "C:\Shows\Graphics".
Choose "TheTrickyBook" by clicking once onto the name. Note that in the command line you will be prompted to accept the loading of the show:

```
Reboot  TheTrickyBook
```

By pressing "RETURN" or by pressing the key "F1" the show is loaded into the memory.
Next time you can confirm the process by double clicking the BOOT.

Leave the File menu by pressing "[Exit]" or by pressing the "ESC" button.
Now, press the key "F1" (Play) and the Lasergraph DSP will start the music on the PC and run the show in sync.

Enjoy the show!

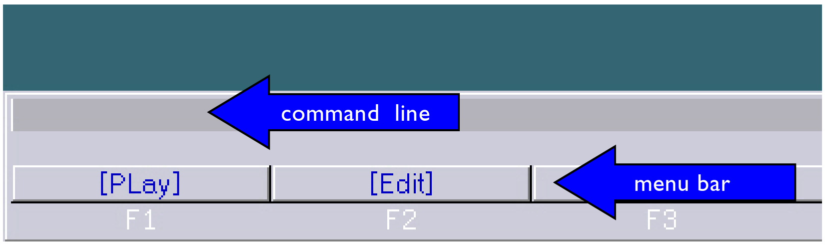
Try some other shows, but only some of the music for the shows came with the Lasergraph DSP (because of copyrights), so be sure that the music is in your PC directory before you load the show, otherwise the show will run without music.

Shows that came with music:

- Burt The Turtle
- Fernweh
- Just One Day
- Just One Fly
- Lase Up
- Showtime
- The Difference
- The Tricky Book

What is next?

Let’s take a closer look at the user interface.
One important part is the command line above the menu bar. We already used it to confirm the loading of the shows.



In the menu bar you will find all in the commands that can be used in the current context. You can execute these commands by clicking onto the buttons, by using the corresponding function key or by typing the commands or their shortcuts into the input line.

The more you get familiar with the commands and shortcuts, the more you will increase the speed of your work by combining mouse and keyboard.

Sections of the user interface:

The **Edit** button will bring you into the section where a trickfilm and the timescript is created and edited.

The **File** button opens up the file system window where you save or load data.

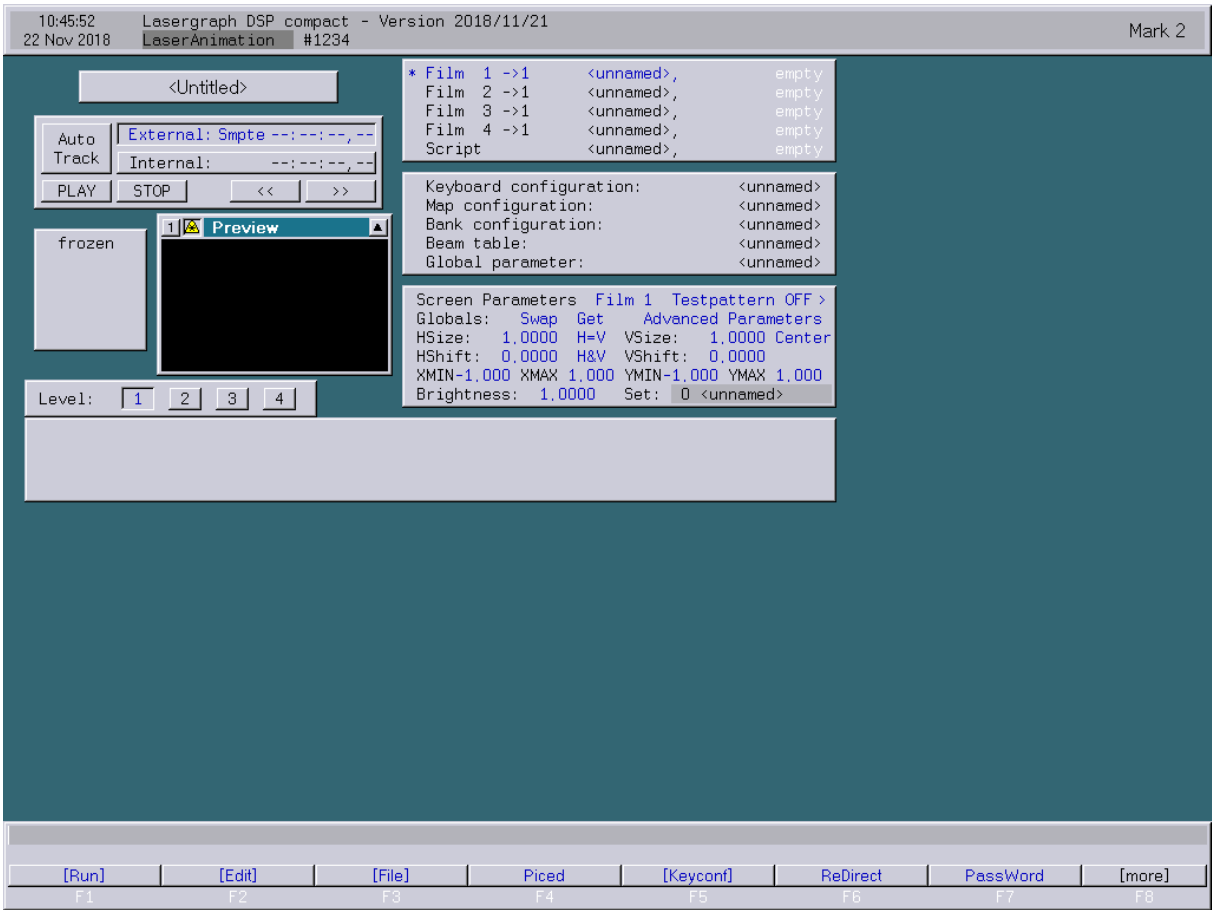
The **Piced** button opens up the "Picture Editor" where you can create and edit pictures/frames.

With the **Keyconfig** button you will open the configuration section for the MIDI-keyboard.

The **[more]** button will "reload" the menu bar with more commands to choose from.

That's it for the quick start! If you need more detailed information, please refer to the specific manual.

Chapter Start Menu



```
10:45:52      Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018   LaserAnimation  #1234                                          Mark 2

* Film  1 ->1       <unnamed>,        empty
  Film  2 ->1       <unnamed>,        empty
  Film  3 ->1       <unnamed>,        empty
  Film  4 ->1       <unnamed>,        empty
  Script            <unnamed>,        empty

  Keyboard configuration:          <unnamed>
  Map configuration:               <unnamed>
  Bank configuration:              <unnamed>
  Beam table:                      <unnamed>
  Global parameter:                <unnamed>

 Screen Parameters  Film 1  Testpattern OFF >
 Globals:    Swap  Get    Advanced Parameters
 HSize:   1,0000  H=V  VSize:    1,0000 Center
 HShift:  0,0000  H&V  VShift:   0,0000
 XMIN-1,000 XMAX 1,000 YMIN-1,000 YMAX 1,000
 Brightness:  1,0000   Set:  0 <unnamed>

External: Smpte --:--:--,--
Internal:       --:--:--,--
```

User's Manual © LaserAnimation Sollinger GmbH, 10827 Berlin, Deutschland.

Information in this document is subject to change without notice and does not represent a commitment on the part of the manufacturer. The software described in this document is furnished under license agreement or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. The purchaser may make one copy of the software for backup purposes. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than for the purchaser's personal use, without written permission.

Table of Contents

General Information

All inputs and outputs are performed using windows and the command line on the *Mark 2*.
Each window has various elements used for specifying and controlling a function.
The command line can be used to type the commands directly with the keyboard.
This combination of the mouse and keyboard makes working with the user interface very efficient.

Every window may be freely shifted around on the interface. To do this, move the cursor to the upper border of the window. Now press the left mouse button and hold it down (grabbing!). Move the mouse cursor in the desired direction. The moving cursor is followed by a thin frame which indicates where the window will reappear when the mouse button is released (dragging!). To resize a window move the cursor to the left, right or bottom border until the cursor changes to a “double-arrow”, then press the left mouse button and move the mouse until you get desired size. By moving the mouse to the lower corners of a window, you can change the horizontal and vertical size simultaneously (indicated by a diagonal “double-arrow”).

The windows may be freely overlapped. When two windows overlap, the "back" window may be made the "forward" window by clicking in the window with the left mouse button.

Notes

- Some windows have a higher priority than others. These include the Menu bar, the File window, and the graphic output of the show. These windows will always be in front of the lower priority windows, but in relationship to other high priority windows they may be brought forward or back with the described methods.

- Some windows have two alternative positions. This can be toggled either by double clicking inside the display area or by clicking the top right button (it depends from the window).

- Windows that are given a fixed size by the system do not respond to this method. The user may only shift but not enlarge or reduce these windows. All windows have a prescribed minimum and maximum size.

- Scrolling allows you to view a document or picture that is larger than the screen or the window.

Input Methods

Command Line Input

The command line is located above the menu bar and the status line. The entire block is usually found usually at the lower edge of the screen. The command line is used with the keyboard for input, e.g., to change values, enter names, or activate commands. The command line consists of only one line but is longer than what is shown on the screen. The entire line may be accessed by scrolling.
All entries to the command line have the following structure:

Command.Subcommand Parameter /Option

Some elements except Command are optional, depending on the command.
The items may also be accessed using the appropriate function keys indicated below the items.

Options without a bracket provide editing help for the command line (located above the Menu bar): the item text is transferred to the command line. If the item is double clicked, it will also be carried out after the transfer. The items in brackets are carried out immediately.

The command line may be edited in either insert or replace mode which may be selected using the <**Ins**> key.
Inputs can be selected, copied, cut and pasted with the usual key combinations, see table. The text input can be entered into other text programs on the PC, e.g. "WordPad", using the Windows clipboard.

Hotkeys for text input:

Key Combination	Effect
<Cursor left>	moves the cursor left one character
<Ctrl+Cursor left>	moves the cursor left one word
<Shift+Cursor left>	selects one character left
<Shift+Ctrl+Cursor left>	selects one word left
<Cursor right>	moves the cursor right one character
<Ctrl+Cursor right>	moves the cursor right one word
<Shift+Cursor right>	selects one character right
<Shift+Ctrl+Cursor left>	selects one word right
<Home>	sets the cursor to the first character in the line
<Shift+Home>	selects from the current cursor position to the beginning of the line
<End>	sets the cursor to the last character in the line
<Shift+End>	selects from the current cursor position to the end of the line
<Ctrl+C> or <Ctrl+Ins>	copies the selection and puts it on the clipboard
<Ctrl+X>	cuts the selection and puts it on the clipboard

<Ctrl+V> or <Shift+Ins>	inserts the clipboard contents
<Ctrl+A>	selects the complete text
<Shift+Del>	deletes all characters from the current cursor position to the end of the text
<Shift+Return>	same as <Shift+Del> followed by Return (immediately executes remaining command)
<Del>	deletes one character at the current cursor position or deletes the selection
<Backspace>	deletes one character in front of the current cursor position or deletes the selection
<Ctrl+Backspace>	deletes all characters in front of the current cursor position to the beginning of the word or deletes the selection
<Ins>	toggles between the insert- and overwrite modes

Input Fields

Input fields change color when touched by the mouse cursor. Here you may

- enter values (value fields),
- select names (File select window),
- switch operation modes (toggle fields) or
- call commands (command fields).
- Change values by moving the mouse while selecting the field (Mouse Poti)

A single click with the left mouse button in a value field causes the current value to be transferred to the command line where it may be edited using the keyboard. The contents of a field which activates a command when clicked on, e.g., Play button, Level buttons etc., is not transferred to the command line. In some cases the value can only be changed by using the Mouse Poti mode (e.g. DGC – Digital Geometric Correction).

Mouse Poti

The Mouse Poti mode is an alternative method to using the keyboard for entering values on the Mark 2. For this you click into a value field and move the mouse back and forth while the mouse button is hold down to change the numerical value. By pressing the middle mouse button an even more precise adjustment of the desired value is possible. If your mouse does not have a middle mouse button, you can emulate the middle mouse button by pressing <Alt> + left mouse button.

Menu Bar

The menu bar is located below the command line and the status line which are all usually displayed at the lower edge of the screen. It is context sensitive, which means it displays the relevant commands and options depending on the mode.
It has eight fields which may be accessed by clicking on them with the left mouse button. Each field may also be activated by pressing the function key (<F1> - <F8>) displayed below it.

[more]

When all menu items cannot be displayed at once, the additional items may be reached by clicking on [more]. This may be necessary more than once.

[Square Brackets]

Menu bar commands that are enclosed in square brackets will be carried out immediately.

<Pointed Brackets>

These fields show a description of the expected parameter which may be selected. If the computer provides a previously set value or is able to generate a reasonable one, it will be transferred to the command line when the field (in pointed brackets) is clicked on.

No Brackets

These commands, parameters, or options will be transferred to the command line. It may be necessary to enter additional parameters before the command is carried out. In most cases, a new menu bar will be provided from which the additional parameters may be chosen.

Abbreviations

All commands may be directly typed into the command line.
The Capital letters can be used as a Shortcut (e.g. *"P" for Piced)*

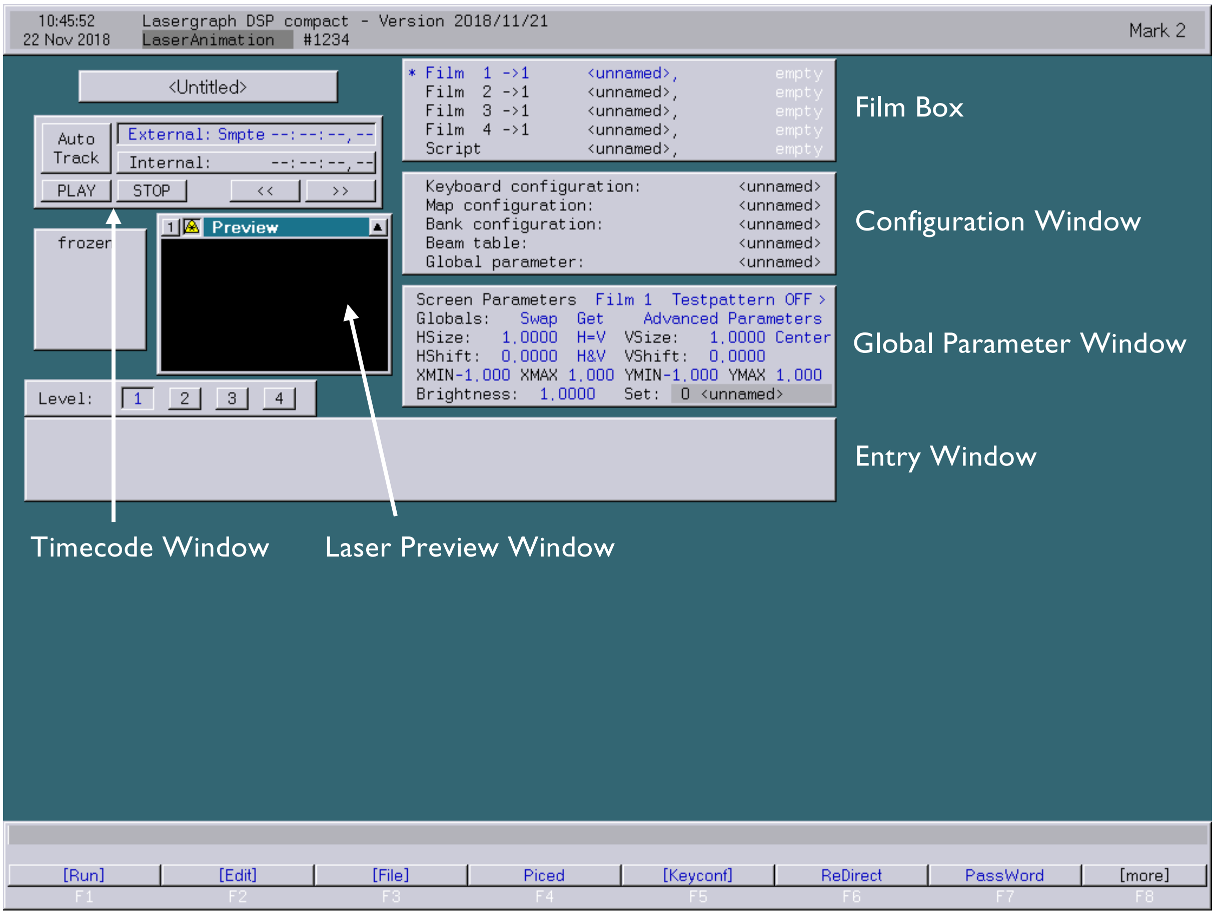
Previous

If a command requires several parameters, the "previous" command may be used to move back one parameter at a time in the input order. The "previous" command may also be activated by simply clicking the right mouse button once.

Double Click with the Right Mouse Button

The next higher command level may be reached by doing two rapid clicks with the right mouse button.

Main Screen



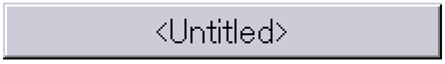
```
10:45:52      Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018   LaserAnimation  #1234                                   Mark 2

* Film  1 ->1      <unnamed>,      empty
  Film  2 ->1      <unnamed>,      empty
  Film  3 ->1      <unnamed>,      empty
  Film  4 ->1      <unnamed>,      empty
  Script           <unnamed>,      empty

  Keyboard configuration:          <unnamed>
  Map configuration:               <unnamed>
  Bank configuration:              <unnamed>
  Beam table:                      <unnamed>
  Global parameter:                <unnamed>

Screen Parameters   Film 1   Testpattern OFF >
Globals:    Swap  Get     Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000    Set:  0 <unnamed>

External: Smpte --:--:--,--
Internal:       --:--:--,--
```

The main screen shown above automatically appears when the *Mark 2* is switched on.
A number of global system parameters that can be directly selected with the mouse are shown in the upper portion of the screen. The items on the menu bar at the bottom of the main screen are selected using the function keys. The option [more] may be selected for additional menu options.
In the windows of the main screen functions and parameters may be directly called and set with the mouse. Parallel to this method the *Mark 2* offers a number of functions that may be accessed from the menu bar in the lower portion of the screen.

The upper portion of the main screen shows some information about the *Mark 2*:

- Date and Time
- Device type (e.g. "Lasergraph DSP compact")
- Computer name
- Serial number
- Software version (e.g. "2018/11/21")
- Software level (e.g. "Mark 2")

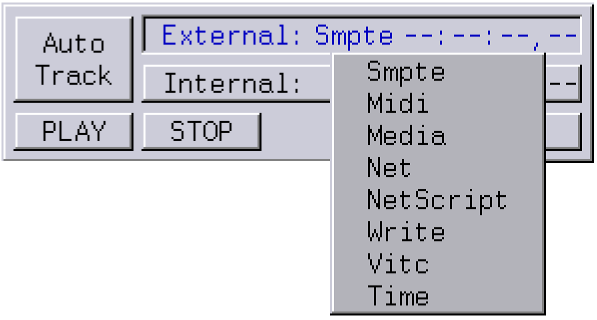
BOOT name



<Untitled>

This window shows the name of the loaded BOOT or <Untitled> if no BOOT is loaded.
A double click on the BOOT name opens the file window to select a BOOT to load.

Timecode Window



The timecode for running a specific show is shown in this window.

Auto Track

If "Autotrack" is active, the external timecode will automatically be switched to internal timecode when the external timecode is missing. As soon the external timecode is available again, the Mark 2 will switch back to the external timecode.

External

The popup menu shows the integrated timecode sources which are supported by the *Mark 2*:

- Smpte timecode (LTC, Longitudinal Timecode)
- Midi timecode (MTC)
- Media timecode (LGServer)
- Net (timecode that is send by another DSP over the network with the option “write”)
- NetScript (timecode from a DSP defined in the Net I/O)
- Write can be used to “stripe” a tape with LTC (SMPTE-Out)
- Vitc (vertical interval Time Code)
- Time (current time of day)

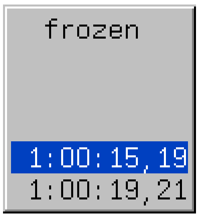
Internal

Displays the internal timecode of the *Mark 2*.

PLAY , STOP

You can start or stop the active script which is marked with '*' in the film box using these buttons.

Frozen


frozen

1:00:15,19

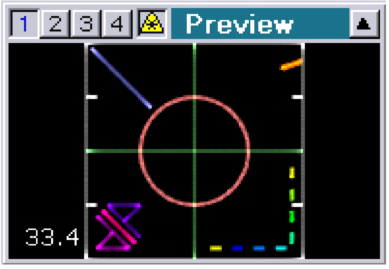
1:00:19,21

This window shows timecode times which may be acquired using the function [GetTime]in the menu bar. These values can be subsequently inserted in the timescript and thus are also available as menu options in the timescript editor ([Edit]).
If one of the displayed times is clicked on, it becomes the start time for the internal timecode.
There are two modes this function can be used in. By clicking onto the word “frozen”/”Auto get”, you can toggle between this functions.
In the mode “frozen” a time is “recorded” only when you click on the function key “GetTime” in the menu.
When “Auto get” is active, the time will be recorded when the timecode stops, too.

Laser Preview Window

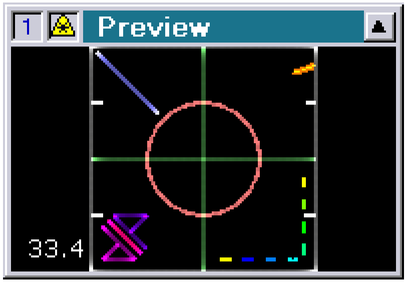
The Laser Preview Window shows a preview of what is displayed by the scanner(s).

workstation Mark 2:



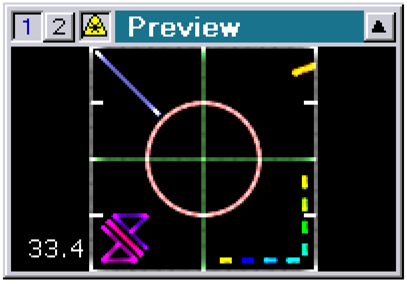
Up to four pairs of scanners can be controlled.
Every installed DSP card can be individually selected for previewing on the w*orkstation* Mark 2.

compact Mark 2:



One pair of scanners can be controlled.

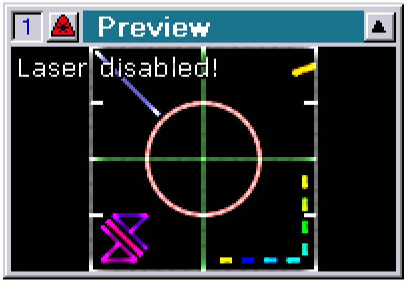
compact dual Mark 2:



Two pairs of scanners can be controlled.

The laser button lets you easily turn the laser output on and off.

When the laser output is deactivated, the button blinks red and yellow and the following message is displayed:

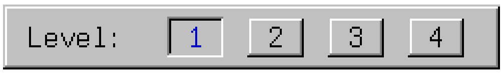


Laser disabled!

Note: When laser output is disabled, the DMX output will be disabled, too.
This function can also be accessed by the following command:
- ENABLELASEROUTPUT ⇨ Yes - turns the laser output on
- ENABLELASEROUTPUT ⇨ No - turns the laser output off

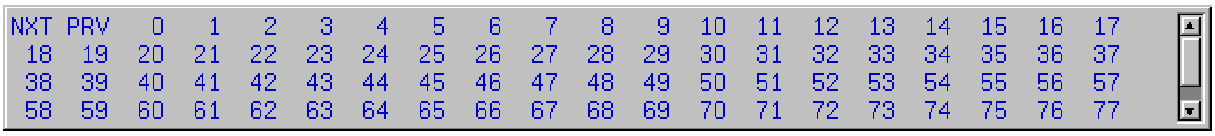
The window size can be toggled by clicking inside the display area. Click the top right hand button for full screen display, to switch back just click the mouse.

Level



An Midi - keyboard may be used for controlling the Mark 2. Each key may be coded for up to four functions, i.e., four levels. The currently active level is shown as an inverse display and may be selected by clicking onto the corresponding button.

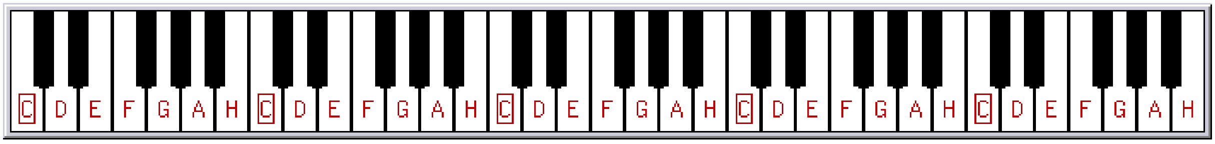
Entry



```
NXT PRV  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 18  19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37
 38  39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57
 58  59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77
```

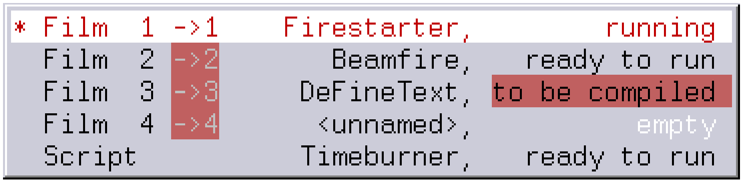
The entry window lies directly below the level window and shows a list of entry numbers for the currently active film (marked with a "*", in the "Film box" window). A film can be started at exactly one of these entry points by clicking on the entry number with the mouse. You may also click on the NXT (next) or PRV (previous) items to jump to the next or previous entry respectively. These functions may be accessed while the film is running.

Keyboard


C D E F G A H C D E F G A H C D E F G A H C D E F G A H C D E F G A H

If the timescript is active (Script is marked with * in the film box), the keyboard will be shown instead of the film entry window. Pressing a key on the real keyboard will cause a color response on the display. If a key configuration is loaded, the corresponding entry or relay commands will be activated.
The keyboard can be used as a a virtual keyboard by clicking onto the keys on the screen. When pressing the “shift”-key on the computer keyboard while pressing or clicking a Midi-key, the key will be activated until it is pressed again.

Film Box



```
* Film  1 ->1   Firestarter,       running
  Film  2 ->2      Beamfire,  ready to run
  Film  3 ->3    DeFineText, to be compiled
  Film  4 ->4     <unnamed>,         empty
  Script         Timeburner,  ready to run
```

This window shows the current status of the four films being run by the Mark 2. The asterisk "*" at the beginning of the line indicates which film is currently active and may be started with [Run] in the menu bar. If a show is to run using a timescript (Script1 has to be activated), it must be started with [Play] in the menu bar.

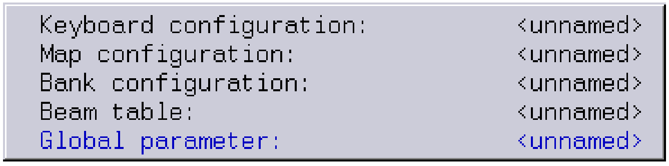
Notes:

- Only a film or a timescript can be active at one time.
- If you click on a film name in the film box, you may load a new film at that position. The computer will warn you if data may be lost by overwriting the previous film and you are given the option to quit or continue the operation.
- The number behind the -> indicates on which projector the film will be shown. Several films may be run on the same projector as long as the films do not run simultaneously, i.e. one after the other.
  If a film is directed to a non existing DSP-Channel, it will be indicated by a red highlighted channel number.

Next in the line is the name of the film or <unnamed> if no name was specified. At the end of the line one of four possible status conditions is shown:

- ready to run means the film has been checked and is ready to run.
- to be compiled means the film has been loaded into the computer but has not been checked. Compiling may take some time (depending on the length of the film).
- running means the film is running.
- empty means the film is empty.

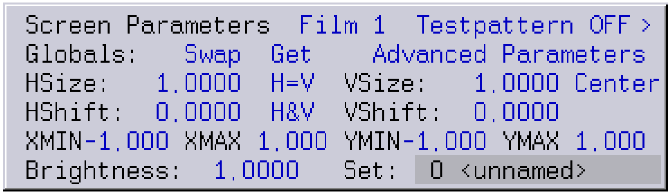
Configuration Window



```
Keyboard configuration:              <unnamed>
Map configuration:                   <unnamed>
Bank configuration:                  <unnamed>
Beam table:                          <unnamed>
Global parameter:                    <unnamed>
```

This window shows the file names of the Keyboard configuration, the Map configuration, the Bank (bench) configuration, the Beam table and the Global parameter.
Clicking on the descriptor (e.g. Beam table) opens the appropriate window.
Clicking on the file name opens the File Window to load e.g. a Map configuration.

Global Parameter Window



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000    Set:  0 <unnamed>
```

In this window you will find options for adjusting the film output to the ambient conditions, e.g. size of the screen. All numerical entries may be adjusted using the Mouse Poti Mode.

The following parameters may be set:

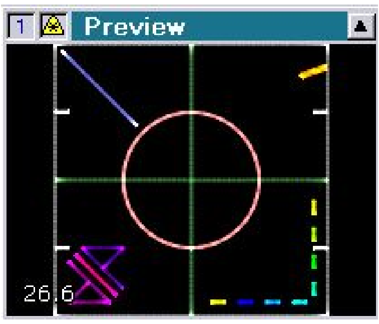
Screen parameter

In the upper line you may select the film for which you would like to set the parameters. Clicking with the left mouse button on the film number brings up the next film, clicking with the middle mouse button returns the previous film.

All the following parameters refer only to the currently selected film. The parameters for the other films remain unchanged.

Testpattern OFF/ON

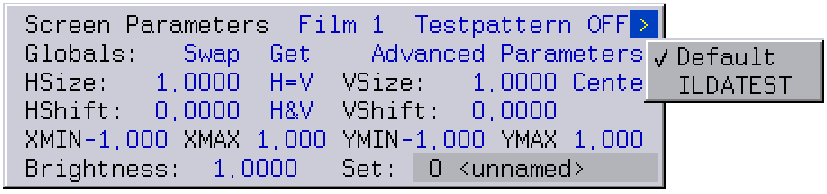
Turns the test pattern of a specific film ON or OFF.



The desired test pattern can be selected from a drop-down menu.

In addition to the default test pattern ("Default") the pictures within the catalog "C:\System\Testpattern.CATL" are available here.

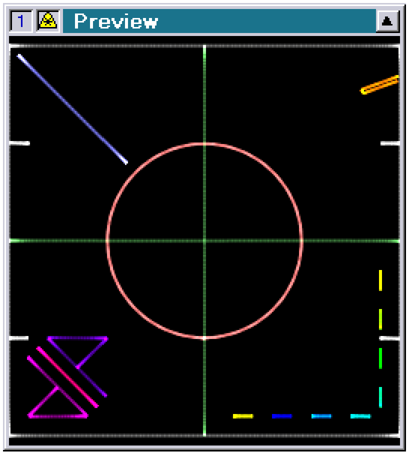
If for instance the picture "ILDATEST.PIC" is located in this catalog the selection list will look like this:



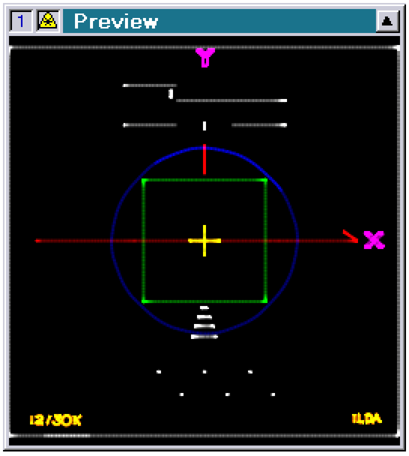
```
Screen Parameters  Film 1  Testpattern OFF >
Globals:   Swap  Get    Advanced Parameters  ✓ Default
HSize:   1.0000  H=V  VSize:   1.0000 Cente    ILDATEST
HShift:  0.0000  H&V  VShift:  0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000   Set:  0 <unnamed>
```

Depending on the selection made the test pattern will change as follows:

Selection "Default":



Selection "ILDATEST":



Notes:

- The list of available test patterns is determined immediately after the Lasergraph DSP is started.
  If a test pattern is added or deleted the Lasergraph DSP has to be restarted for the changes to be visible in the selection list.

- The test pattern can be selected for each trickfilm individually.

- The selection of the test pattern stays active until the next time the Lasergraph DSP is started.

It is also possible to override the default test pattern by loading a certain picture into the picture memory using the following name:

Catalog name: "TestPictures"
Picture name: "Default" -> will be used as default test pattern

To change the DGC test pattern you have to load the following picture:

Catalog name: "TestPictures"
Picture name: "DGC" -> will be used as DGC test pattern

To change the primary color test pattern you have to load the following picture:

Catalog name: "TestPictures"
Picture name: "COL" -> will be used as primary color test pattern

It is also possible to use different test pattern for each film:

Catalog name: "TestPictures"

Picture name: "ForFilm1" -> will be used as default test pattern for Film 1
"ForFilm2" -> will be used as default test pattern for Film 2
"ForFilm3" -> will be used as default test pattern for Film 3
"ForFilm4" -> will be used as default test pattern for Film 4

"ForFilm1DGC" -> will be used as DGC test pattern for Film 1
"ForFilm2DGC" -> will be used as DGC test pattern for Film 2
"ForFilm3DGC" -> will be used as DGC test pattern for Film 3
"ForFilm4DGC" -> will be used as DGC test pattern for Film 4

"ForFilm1COL" -> will be used as primary color test pattern for Film 1
"ForFilm2COL" -> will be used as primary color test pattern for Film 2
"ForFilm3COL" -> will be used as primary color test pattern for Film 3
"ForFilm4COL" -> will be used as primary color test pattern for Film 4

Notes:

- There are some requirements concerning the test patterns like which colors have to be used or which matrix. The easiest way to create your own test pattern is to modify the existing ones.
  You can find the existing ones on drive C: in the directory "Tests".

- To use the default pictures again you can unload the catalog "Testpictures" from the memory.

Globals

Here you may set the horizontal and vertical dimensions of the output.
Swap - swap global parameter with film 1 or film 2 or film 3 or film 4.
Get - get global parameter from film 1 or film 2 or film 3 or film 4.

Output Size

Output size may be adjusted using HSize (horizontal size of film) and VSize (vertical size of film) or by using H=V (horizontal and vertical size).

Picture Position

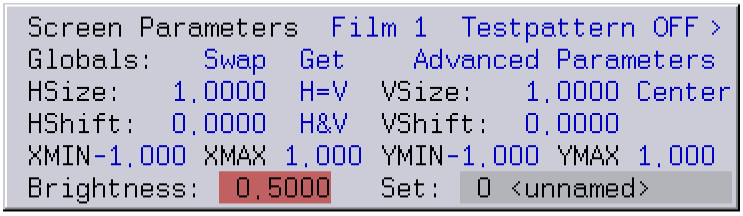
HShift allows horizontal and VShift allows vertical shifting of the output.
Center - use this function to center and maximize the picture within the clipping borders.

Clipping

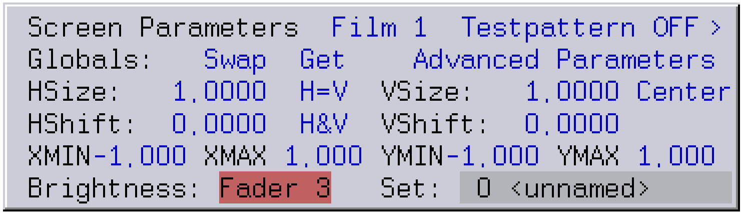
The clipping borders of the picture may be adjusted using the XMIN, XMAX, YMIN, and YMAX parameters.

Brightness

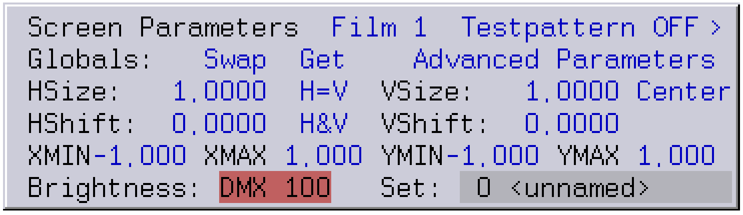
This function allows the adjustment of the global brightness by selecting a fixed value or one of the 64 faders or one of the 512 DMX-In channels, compare the pictures:



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:  0.5000    Set:  0 <unnamed>
```



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness: Fader 3    Set:  0 <unnamed>
```



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness: DMX 100    Set:  0 <unnamed>
```

Notes:

- For safety reasons the adjusted value or the selected fader or DMX channel is not stored! After restarting the DSP the “Brightness” will again be set to 1.000 (100%).
  A "ReBoot" will not change this setting!

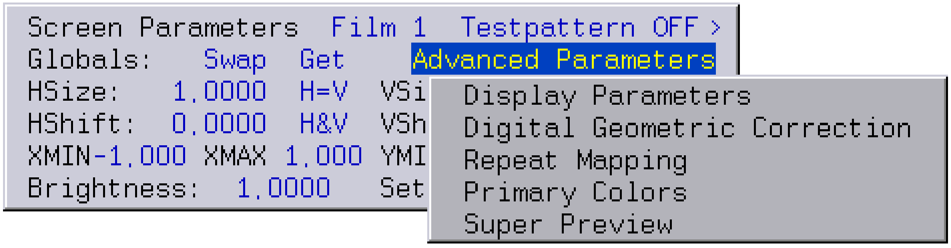
- If DMX is used as source for the global brightness the selected DMX channel number is neither affected by the externally adjusted DMX base address nor by the "DMX-In offset" within the Advanced Display Parameters.

0 <unnamed>

It is possible to create up to 16 different DGC settings for a film for adaptation of the laser output to a projection surface and to save these settings under different names, e.g. Screen 1, Screen 2 etc. For that purpose click on 0 <unnamed>, select the desired set no. (0 to 15) and open the Digital Geometric Correction (DGC) window for this set from the Advanced Parameters.

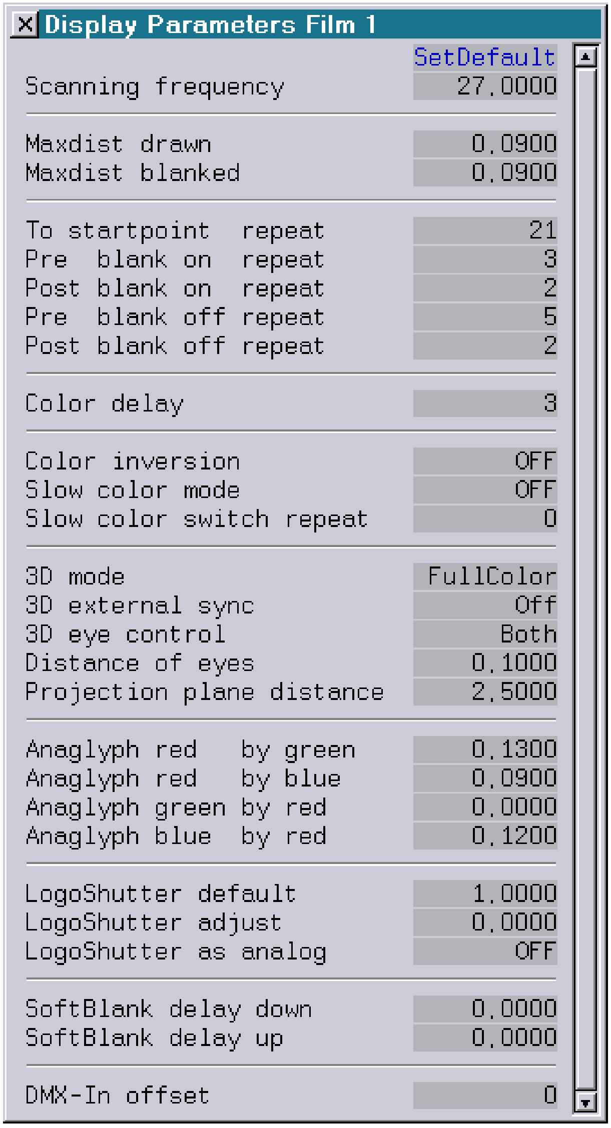
Advanced Parameters

A popup menu will be opened by clicking Advanced Parameters


Screen Parameters Film 1 Testpattern OFF >
Globals: Swap Get Advanced Parameters
HSize: 1.0000 H=V VSi
HShift: 0.0000 H&V VSh
XMIN-1.000 XMAX 1.000 YMI
Brightness: 1.0000 Set
Display Parameters
Digital Geometric Correction
Repeat Mapping
Primary Colors
Super Preview

Clicking on the descriptor (e.g. **Display Parameters**) opens the appropriate window.

Display Parameters



Display Parameters Film 1

	SetDefault
Scanning frequency	27,0000
Maxdist drawn	0,0900
Maxdist blanked	0,0900
To startpoint repeat	21
Pre blank on repeat	3
Post blank on repeat	2
Pre blank off repeat	5
Post blank off repeat	2
Color delay	3
Color inversion	OFF
Slow color mode	OFF
Slow color switch repeat	0
3D mode	FullColor
3D external sync	Off
3D eye control	Both
Distance of eyes	0,1000
Projection plane distance	2,5000
Anaglyph red by green	0,1300
Anaglyph red by blue	0,0900
Anaglyph green by red	0,0000
Anaglyph blue by red	0,1200
LogoShutter default	1,0000
LogoShutter adjust	0,0000
LogoShutter as analog	OFF
SoftBlank delay down	0,0000
SoftBlank delay up	0,0000
DMX-In offset	0

Here different parameters for the whole laser output can be adjusted:

SetDefault

By clicking into this field so called default settings for Crystal, Scanner, Scanner RGB, ISP30k, Blitz, Phaenon30k or Phaenon60k can be selected. The "ScanningParameters" are automatically adapted to the selected setting.

Scanning frequency

Picture quality can be optimized with this. The maximum value is 256 kHz, however this is very demanding for the scanners and the DSP. Usually it is possible to set a perfect frequency by slightly readjusting the Default value. Higher frequencies are advisable especially when working with Abstracts and Bitmaps.

Note:

Be carefully with changing the frequency value. You might damage the scanning system by setting the values too high!

Maxdist drawn, Maxdist blanked

The *Mark 2* automatically inserts points during display of pictures if the distance between two points is too large (automatic interpolation). The value Maxdist drawn can be used to increase or decrease this distance.

In contrast to Maxdist drawn, with Maxdist blanked the blanked lines are interpolated.

Generally the following applies: If the distances are chosen too small the picture will start to flicker, if they are too large the picture quality will suffer.

To startpoint repeat

When the last point is reached during projection, the scanners will jump back to the first point. This value defines how many points will be inserted before the next picture is drawn.

Pre blank on repeat

Defines how many points are inserted before the Blanking is turned on.

Post blank on repeat

Defines how many points are inserted after Blanking has been turned on.

Pre blank off repeat

Defines how many points are inserted before the Blanking is turned off.

Post blank off repeat

Defines how many points are inserted after Blanking has been turned off.

Color delay

Here the delay of the projection before a change of color is set. The parameter is used to correct picture changes due to the inertia of the scanners as compared to the crystal (PCAOM). This is especially important for the bitmap function but it may also be useful to correct general errors; for instance a line that changes color without blanking appears differently in the projection than in the [Piced] as long as the Color delay is not set correctly.

Color inversion

This function is not available anymore, although it is still implemented because of compatibility reasons.

If the inverting of the color output is necessary, you have to open the housing and change the corresponding jumper.

Slow color mode

Should be switched to "On" if color and blanking are not controlled using a crystal but with scanners or color box.

Slow color switch repeat

When very slow color controls are used, it is possible to increase the delay ("Repeats") when a color change is projected.

3D mode

For stereoscopic 3D projections with 3D glasses, either full color or with the anaglyph method. This transforms the original picture into a red/blue or red/green picture (depending on the type of glasses used).

3D external sync

For stereoscopic film- or video projections shutter glasses are often used to achieve the desired 3D effect. For this method pictures are displayed alternately for the right eye and for the left eye.
The shutter glasses cover the eyes alternately, allowing each eye to only see the picture designated for it.
In order to integrate a laser into such a projection, the laser output has to be synchronized with the shutter signal for the glasses.
This can be done by using the "3D external sync":

Off The DSP generates its own synchronization signal.

Low=Left When a low level signal is present at the sync input the picture for the left eye is displayed, when a high level signal is present the picture for the right eye is displayed.

High=Left When a high level signal is present at the sync input the picture for the left eye is displayed, when a low level signal is present the picture for the right eye is displayed.

<u>Notes:</u>

- In order to use the external synchronization, the Mark 2 has to be equipped with a special sync input which is not available on the current hardware. However, this can be added upon request.

- The external synchronization will only work if the output of each individual picture is completed before the sync signal changes its state. If the picture output takes longer the remainder of the of the picture will be visible for the wrong eye. The maximum complexity of the pictures depends on the frequency of the sync signal and on the scanners which are used.

- Since the command "RunningText" will lead to each character being displayed once for the left eye and once for the right eye before display of the next character begins, the external synchronization will lead to a low update rate in this case.

3D eye control

Defines if the film displays both or only the right or left eye.

Distance of eyes

Determines the distance between the right and the left eye of the viewer.

Projection plane distance

Defines the virtual distance of the picture in space.

Anaglyph red by green

Determines the intensity for the left eye (red) - the default value is 0.1300, i.e. 13% red to 100% green.

Anaglyph red by blue
Determines the intensity for the left eye (red) - the default value is 0.0900, i.e. 9% red to 100% blue.

Anaglyph green by red
Determines the intensity for the right eye (green) - the default value is 0.0000.

Anaglyph blue by red
Determines the intensity for the right eye (blue) - the default value is 0.1200, i.e. 12% blue to 100% red.

LogoShutter
There are a few systems where the brightness of the emitted laser pictures can additionally be adjusted through a special PIN output, the so called LogoShutter. Similar to the analog output, constant values as well as fades can be set.

LogoShutter default
Setting for the default LogoShutter position in the value range from 0 to 1

LogoShutter adjust
For fine setting / readjustment (Mouse Poti mode)

LogoShutter as analog
Older DSP cards do not support the LogoShutter, therefore there is a simulation using an analog (25 to 32).

SoftBlank delay down
Determines globally how fast the "SoftBlanking" (makes it possible to let lines fade out smoothly) is turned off.

SoftBlank delay up
Determines globally how fast the "SoftBlanking" is turned on.

DMX-In offset

For setting an offset for all DMX-In channels. Values from 0 to 511 can be entered, where the offset setting corresponds to the following assignments to the DMX channels:

DMX channel	**1**	**2**	**3**	**…**	**510**	**511**	**512**
Offset=0	1	2	3	…	510	511	512
Offset=1	2	3	4	…	511	512	
Offset=2	3	4	5	..	512		
…	…	…	…	…	…	…	…

e.g. if the "DMX-In offset" is set to 10, the internal DMX-In Channel 1 will be controlled by external DMX channel 11.

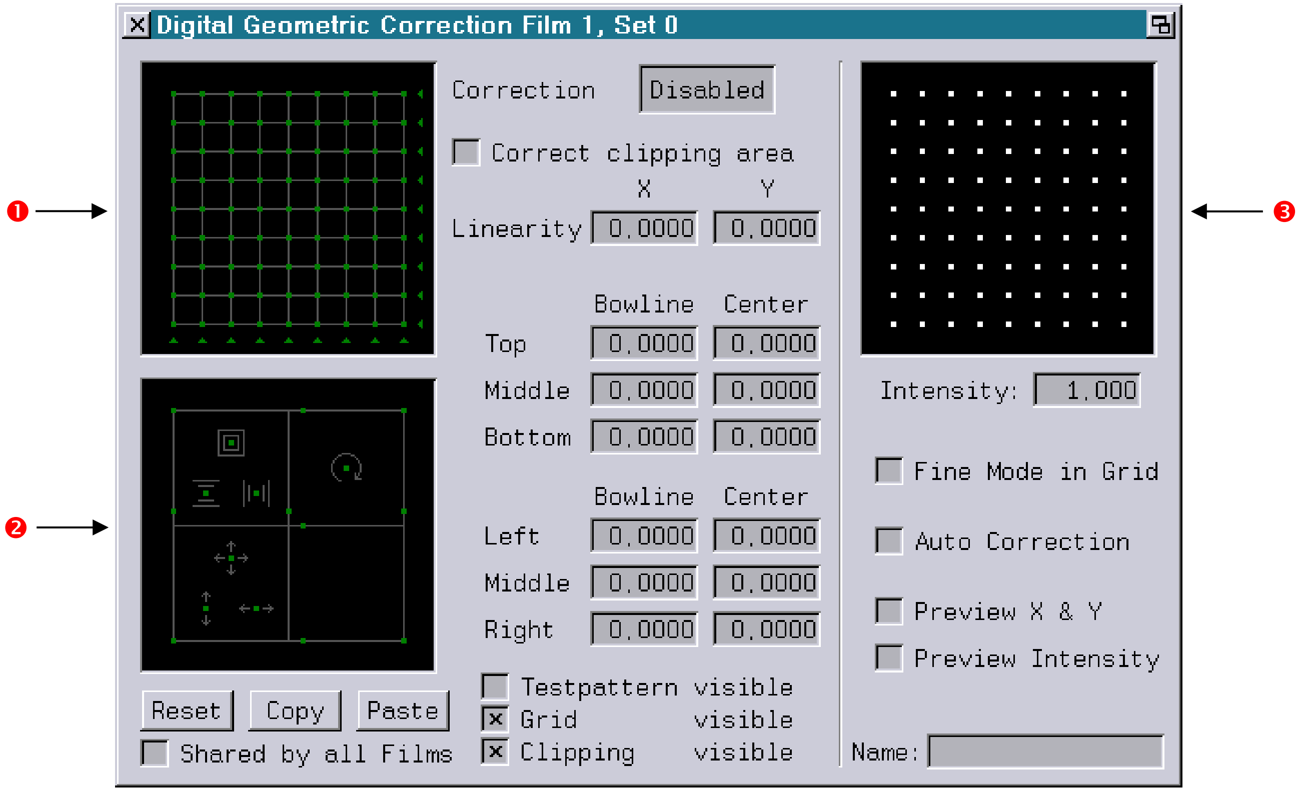
<u>Note:</u>

In some special laser devices like “Blitz Stage” or “Phaenon” where a *Mark 2* controller is implemented, you will find an external DMX-switch which can be used to set the basic DMX-address in addition to the software offset.
The software offset will be added to the “external” switch offset.

The command DMXBASE can be used to display the currently selected DMX base address

Digital Geometric Correction (DGC)

The DGC makes it possible to adjust the laser output to irregular or distorted projection screens as well as to fit the output of several scanners together for multi scanner shows. Clicking onto "Digital Geometric Correction" opens the following window:



Digital Geometric Correction Film 1, Set 0

Correction Disabled

☐ Correct clipping area

	X	Y
Linearity	0.0000	0.0000

	Bowline	Center
Top	0.0000	0.0000
Middle	0.0000	0.0000
Bottom	0.0000	0.0000

	Bowline	Center
Left	0.0000	0.0000
Middle	0.0000	0.0000
Right	0.0000	0.0000

Intensity: 1.000

☐ Fine Mode in Grid

☐ Auto Correction

☐ Preview X & Y

☐ Preview Intensity

☐ Shared by all Films

☐ Testpattern visible
☒ Grid visible
☒ Clipping visible

Name:

The DGC is activated resp. deactivated using the check box Correction Enabled/Disabled.

If the DGC is deactivated after it is individually adjusted, the previously specified values will be preserved.

The upper correction window ❶ makes it possible to place each spot of a 9 by 9 grid individually by dragging the spot to the desired position. Once a spot is placed, indicated by its red color, it is no longer influenced by corner or bowline positions.

A spot can be released by clicking on it.

By using the triangle beside the grid you can move all spots in the corresponding line together.

The direction of movement while shifting individual spots or complete lines in the DGC can be restricted to one direction using the <SHIFT> key.

The window ❷ allows adjusting the shape of the whole laser output. The tools simplify the adjustment. They are activated by clicking on the green points in the center or on the lines:

Enlarge or downsize the output in total

Enlarge or downsize the output vertically

Enlarge or downsize the output horizontally

Shift the output to the left, right, up or down

Shift the output vertically only

Shift the output horizontally only

Rotate the output

These values can only be modified by Mouse Poti.

Two fields are available for Linearity. They shift the center bowlines entirely left or right respectively up and down, stretching and upsetting the corresponding lines towards the edges.

Twelve fields give you access to the bowline correction:
Six for the horizontal (Top, Middle, Bottom) and six for the vertical (Left, Middle, Right)

A checkbox enables or disables the "clipping area".
When this option is unchecked, the corners of the Digital Correction Area correspond to the unclipped test picture. If this option is enabled, the corners correspond to the "clipped" corners of the test picture.

Three checkboxes determine what is displayed on the laser: The Grid for the grid spots to be set individually, the Test picture or the Clipping rectangle. They can be combined in any way, but at least one of them has to be switched on.

In the window for intensity ❸ one or more points or a whole area can be selected for which the Intensity - value can be adjusted by moving the mouse with pressed left mouse button or by entering a value - value 1.000 = full brightness.

As long as "Fine Mode in Grid" is activated all settings in the top left 9x9 grid ❶ are performed in fine mode. When "Fine Mode in Grid" is deactivated the fine mode can still be used by using the middle mouse button resp. the left mouse button in combination with either the <CTRL> or the <ALT> key.

With the option "Preview X&Y" and "Preview Intensity" you will see the DGC on the preview output, too.

Make the desired adjustments in the DGC window and name this setting, e.g. Screen 1.
When the checkbox "Shared by all Films" is activated, the settings of film 1 will be used by all four films.

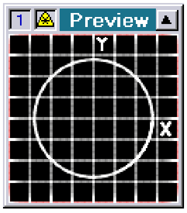
All values may be reset by double clicking the Reset button.

A single click on the Reset button writes the following command into the command line:

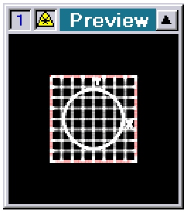
```
DGC.Reset [<hsize> [vsize]]
```

`<hsize>` horizontal size in the range from -1.0 to +1.0
`<vsize>` vertical size in the range from -1.0 to +1.0

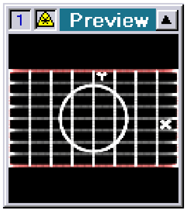
Now there are three possibilities:



1) Confirm without parameter
The DGC is reset to the maximum size.
This corresponds to a double click on the Reset button.



2) Specifying one parameter
The DGC is reset and reduced to the specified size.
The parameter is used for both horizontal and vertical size.



3) Specifying two parameters
The DGC is reset and reduced to the specified sizes.
Horizontal and vertical size may be different.

Using the Copy button copies the DGC adjustments e.g. for Screen 1 to the clipboard.
Paste inserts them e.g. in another setting, e.g. Screen 2.

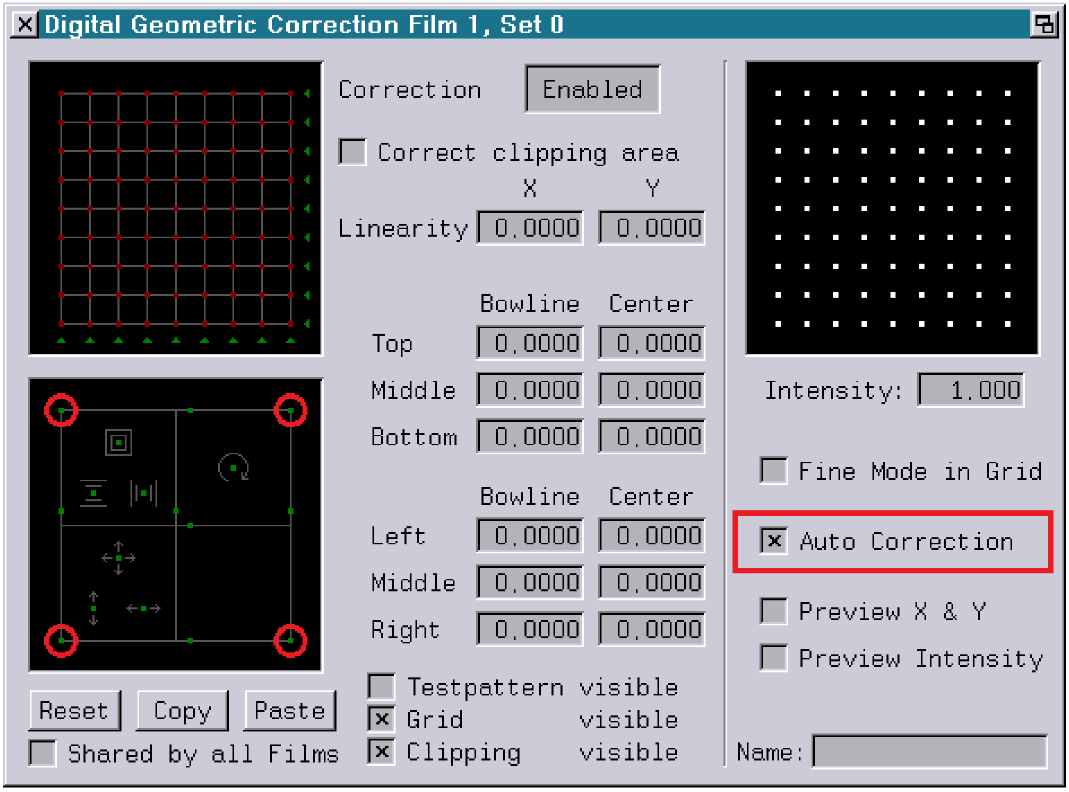
16 different settings can be set up for each film.

Automatic correction of the DGC

Depending on the size of the projection and the position of the projector relative to the projection surface, substantial distortions may occur within the projection due to physical reasons.

The Lasergraph DSP provides an automatic correction of these distortions which enables you to perfectly adjust the DGC in no time.

When "Auto Correction" is activated, the DGC will be automatically calculated based on the four corner points.



Digital Geometric Correction Film 1, Set 0

Correction Enabled

□ Correct clipping area

	X	Y
Linearity	0.0000	0.0000

	Bowline	Center
Top	0.0000	0.0000
Middle	0.0000	0.0000
Bottom	0.0000	0.0000

	Bowline	Center
Left	0.0000	0.0000
Middle	0.0000	0.0000
Right	0.0000	0.0000

Intensity: 1.000

□ Fine Mode in Grid

☒ Auto Correction

□ Preview X & Y

□ Preview Intensity

□ Testpattern visible

☒ Grid visible

☒ Clipping visible

□ Shared by all Films

Name:

If you need to make further changes after the automatic adjustment of the DGC, for example, to compensate minor irregularities on the projection surface, you can still do this via the points in the 9x9 field.

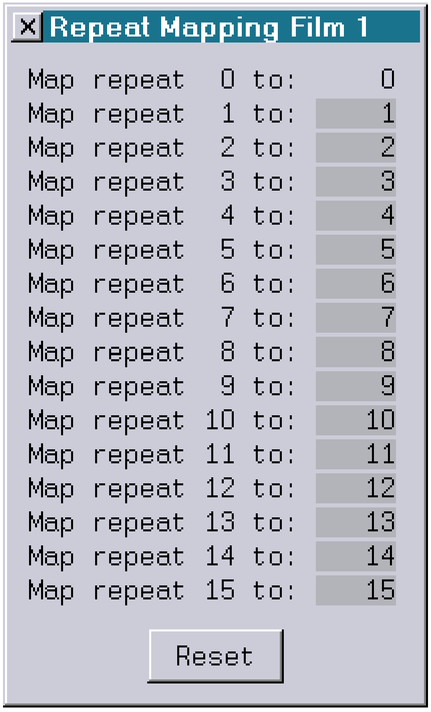
Annotations:

- The automatic calculation of the DGC is performed under the condition that the scanners provide an optical angle of +/-40° at maximum ILDA output level.
  For projectors not having an optical angle of +/-40° the maximum scan angle can be set temporarily using the following command:
  ```
  DGC.AutoCorrection.MaxScanAngle <h> <v>
  ```

- The automatic calculation can be performed only as long as the four corner points define a convex surface. If a concave surface arises during the adjustment, a linear interpolation will be used instead.

- At the Workstation the adjustment of the DGC is very slow if "Auto Correction" is activated. It is therefore recommended to first disable "Auto Correction", position the four corners, and only then enable the automatic correction again.

Repeat Mapping

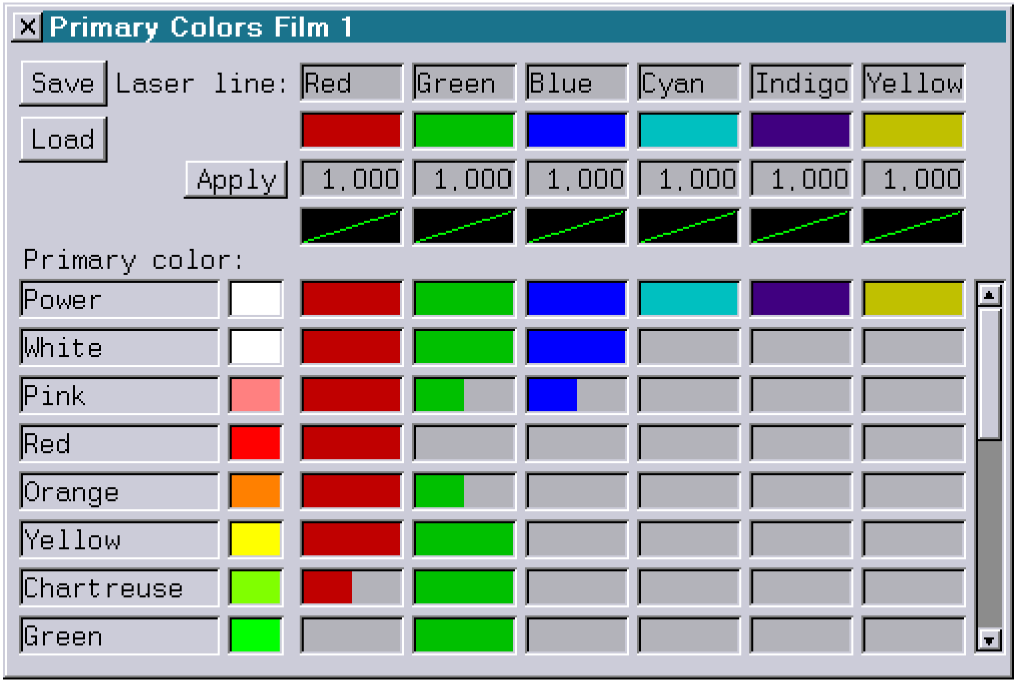

Repeat Mapping Film 1

Map repeat 0 to:	0
Map repeat 1 to:	1
Map repeat 2 to:	2
Map repeat 3 to:	3
Map repeat 4 to:	4
Map repeat 5 to:	5
Map repeat 6 to:	6
Map repeat 7 to:	7
Map repeat 8 to:	8
Map repeat 9 to:	9
Map repeat 10 to:	10
Map repeat 11 to:	11
Map repeat 12 to:	12
Map repeat 13 to:	13
Map repeat 14 to:	14
Map repeat 15 to:	15

A repeat map is a table with 15 entries, one for each "virtual" repeat, which is the one defined in pictures themselves. Each entry determines the "real" repeat of a point which will be used when put to the scanner. The range for "real" repeats is [0; 50]. It is not possible to map a "virtual" repeat of 0, and repeat mapping is not applied to points of blank change.

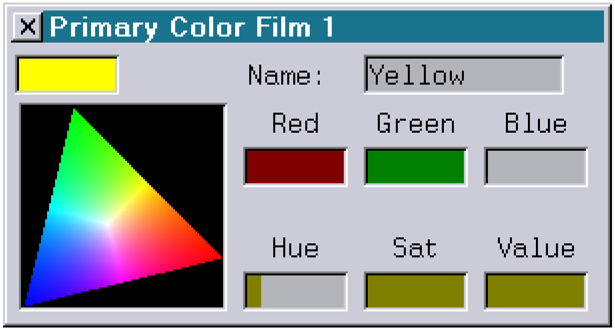
Primary Colors

The color table for the current trickfilm will be opened by clicking onto Primary Colors in the popup menu of the Advanced Parameters:



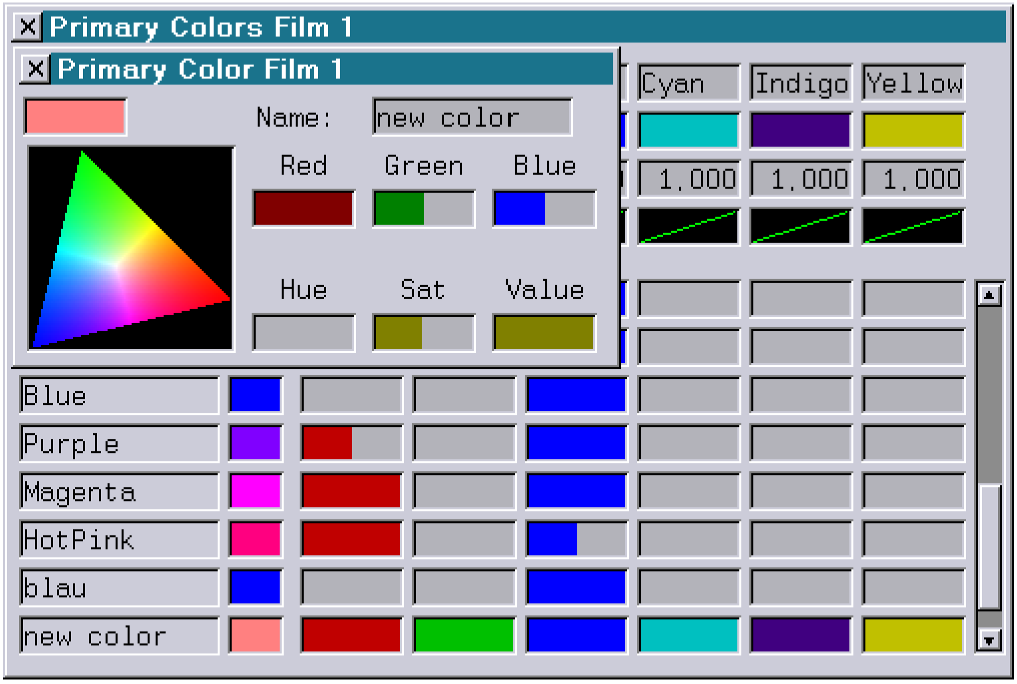
Here you may load a color table of a particular film.
Laser colors vary for different lasers from different manufacturers so a global color correction is necessary. These parameter settings are valid for all color commands of this film.
The six "Laser line" colors Red, Green, Blue, Cyan, Indigo, Yellow represent the color lines of the laser. They are used for the adaptation of the laser colors to the colors of the monitor. It is possible to change or to rename the 16 base colors or insert a new color respectively.

To change a color for the Preview, click into the desired color in the table and adjust it in the color-triangle of the new window:



If you want to change only one color portion click into the field and adapt the color using the Mouse Poti mode.

To add the new mixed color to the existing colors click into the last field of the column Primary Color and enter a new name in the input field name:



Name: new color

1,000 1,000 1,000

Blue
Purple
Magenta
HotPink
blau
new color

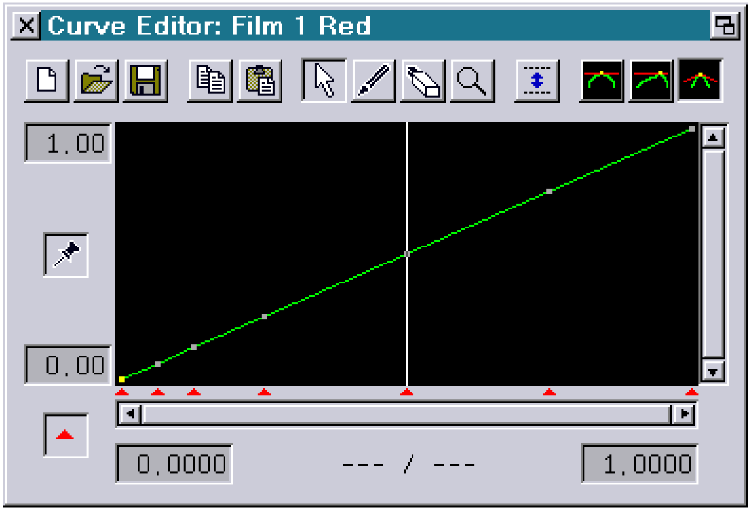
To change the colors on the laser output, click into the corresponding fields behind the color name and change each laser color separately.

To delete an unused primary color simply delete the name of the color.

The individual adjustment of the characteristic line of each individual laser line allows a significant improvement in the color behavior and thereby the whole picture output.

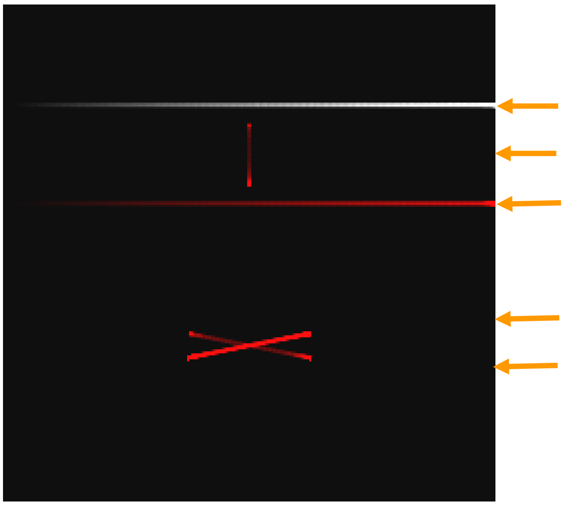
In order to realize this, the Primary Colors window contains curves for each laser line.

Clicking on one of these curves will open the curve editor:



The headline of the window shows the number of the film and also the name of the laser line which is currently being edited, in this case "Red".

When the test picture is activated, the laser will display the following test picture:



all 6 laser lines 0%-100%

editing position

current laser line 0%-100%

current laser intensity

reference intensity

The test picture is divided into several sections:

- The upper part contains a horizontal line which activates all laser 6 lines in the range from 0% (left) to 100% (right).
- A little below that there is a horizontal line which only activates the laser line which is currently being edited, in the range from 0% (left) to 100% (right).
- Between the two horizontal lines a vertical line denotes the current editing position. This line is displayed with the brightness which is set for the editing position.
- Further down there are two intersecting lines.
  One of these lines is the reference line, the other represents the currently set intensity.
  The goal of the adjustment is to achieve equal brightness of these lines for all positions.

Instructions for Adjusting the Characteristic Lines:

1) In case it has not been done: Activate the test picture and open the Primary Colors window.

2) Click on the curve for the first laser line.

3) Grab the point of the curve at 0.500 and hold the mouse button down. Move the mouse up resp. down at the position 0.500 until the two intersecting lines in the lower part of the test picture have the same brightness.

4) After setting the value at 0.500 set the value at 0.250, then at 0.750 and finally at values below 0.250 in the same way.

5) To verify the brightness, click into the black field and hold the mouse button down. Now the brightness of the two intersecting lines should remain equal when the mouse is moved left and right.

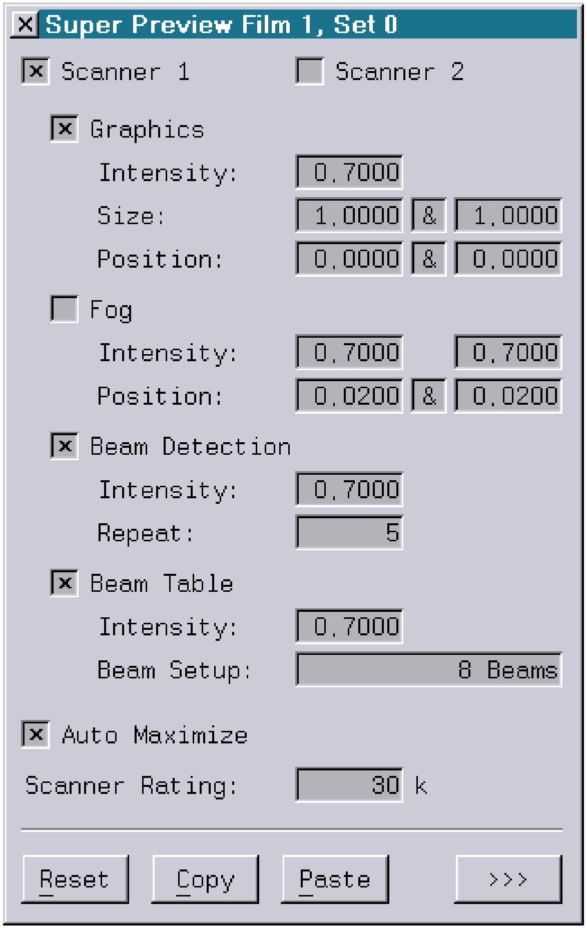
6) Repeat points 3) to 5) for the other laser lines.

Super Preview

This menu item can be used to adjust the Super Preview parameters.

The Super Preview can be found within PC applications like LGRemote (if enabled), LGPreview or Lpv Creator.

Note:
Most Super Preview parameters can be overridden by the corresponding Super Preview Parameters within these applications.



Super Preview Film 1, Set 0

☒ Scanner 1 ☐ Scanner 2

☒ Graphics
Intensity: 0.7000
Size: 1.0000 & 1.0000
Position: 0.0000 & 0.0000

☐ Fog
Intensity: 0.7000 0.7000
Position: 0.0200 & 0.0200

☒ Beam Detection
Intensity: 0.7000
Repeat: 5

☒ Beam Table
Intensity: 0.7000
Beam Setup: 8 Beams

☒ Auto Maximize
Scanner Rating: 30 k

The Headline shows which specific set of parameters the current window applies to.

Examples:

Super Preview Film 1, Set 0

The Super Preview parameters apply to Film 1, DCG Set 0

Super Preview Film 3, Set 11

The Super Preview parameters apply to Film 3, DCG Set 11

Super Preview Set 5 (shared)

The Super Preview parameters apply to DCG Set 5, which is shared by all Films (see "Digital Geometric Correction" option "Shared by all Films").

Scanner 1, Scanner 2

Enables / disables the output of Scanner 1 and Scanner 2.

Usually each projector output is displayed once within the Super Preview.
Under certain circumstances, for example for beam shows where two projectors are fed with the same projector signal, the Super Preview can simulate those two projectors.
To achieve this Scanner 2 has to be enabled.
The output of the second projector is the same as the output of the first projector with the exception that X-Size and X-Pos are mirrored.

Graphics Enables / disables the simulated graphics projection.

Intensity Changes the brightness of the displayed picture.

Size Changes the X-Size and the Y-Size of the displayed picture.
Using the button "&" both Sizes can be adjusted simultaneously while
the aspect ratio is retained.

Position Changes the X-Position and the Y-Position of the displayed picture.
Using the button "&" X-Position and Y-Position can be adjusted simultaneously.

Fog Enables / disables the simulation of fog.

Intensity These two values define the brightness of the output within the simulated fog.
The first value specifies the brightness at the projector aperture.
The second value defines the brightness in space near the observer.

A value of 1.0 will result in having the same brightness on the way from the projector to the observer.

A value of 0.0 means that the beam at the projector aperture has the brightness adjusted by the first intensity value and will fade to black on it's way to the observer.

Typical values should be 0.7 or less.

Position Changes the X-Position and the Y-Position of the laser projector.
Using the button "&" X-Position and Y-Position can be adjusted simultaneously.

Beam Detection

With this option enabled, a point with a certain amount of repeats will be displayed as a "beam", a bright line.
"Enable Beam Detection" works only if fog is enabled.

Intensity

Changes the brightness of the detected lines.

Repeat

Determines the number of repeats needed to detect a point as a "beam".

Beam Table

Enables / disables the display of scan beams (Trickfilm commands "SetBeam" and "SetBeamTimed").
"Enable Beam Tables" works independent of the state of "Enable Graphics" and "Enable Fog". However, the position of the laser projector from the fog section is used as position of the beam source.

Intensity

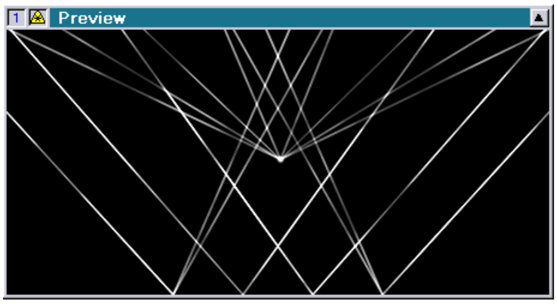
Changes the brightness of the beams.

Beam Setup

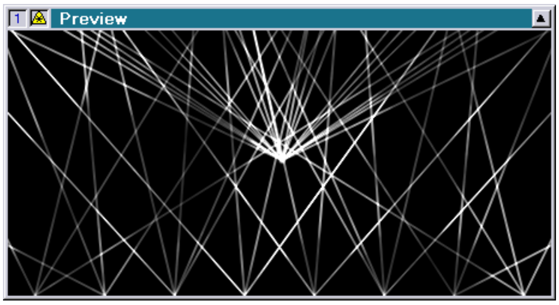
If Trickfilm commands "SetBeamTimed" use entries from the beam table to position the beams, the parameter "Beam Setup" specifies how to handle these beams:

None The X/Y positions from the beam table are used.

8 Beams Up to 8 fixed beam position from the following setup are used:



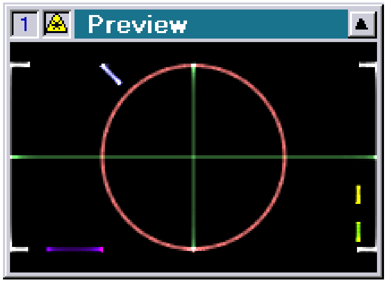
20 Beams Up to 20 fixed beam position from the following setup are used:



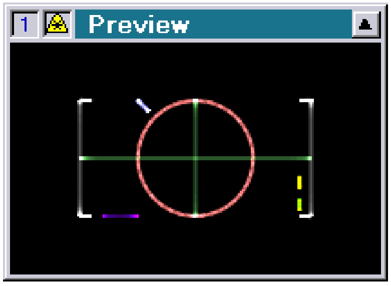
Auto Maximize

With this option enabled the output size in the Super Preview will be increased to optimally use the space available in the window when the laser output is clipped by the Global Parameters.

"Auto Maximize" enabled:



"Auto Maximize" disabled:



Note:
Disabling "Auto Maximize" in the Super Preview setup is not supported by old Lasergraph DSPs and Mark 2 with a System Software Version older than 2015-09-07.

Scanner Rating

Allows simulation of different scanners.

A scanner rating of 30k is equivalent to the previous default setting.

When the PHAENON laser projectors with 60k scanners (CTI 6215H) are used a scanner rating value of 60 should be set for the best representation.
The scanner rating can be set to values in a range from 5 to 120 in order to allow adapting the display to other scanners.

Note:
Scanner Ratings other than 30k are not supported by old Lasergraph DSPs and Mark 2 with a System Software Version older than 2012-01-27.

Reset

Resets all parameters to default.

Note:
If Scanner 2 is enabled the defaults are different than the defaults where only Scanner 1 is enabled.

Copy

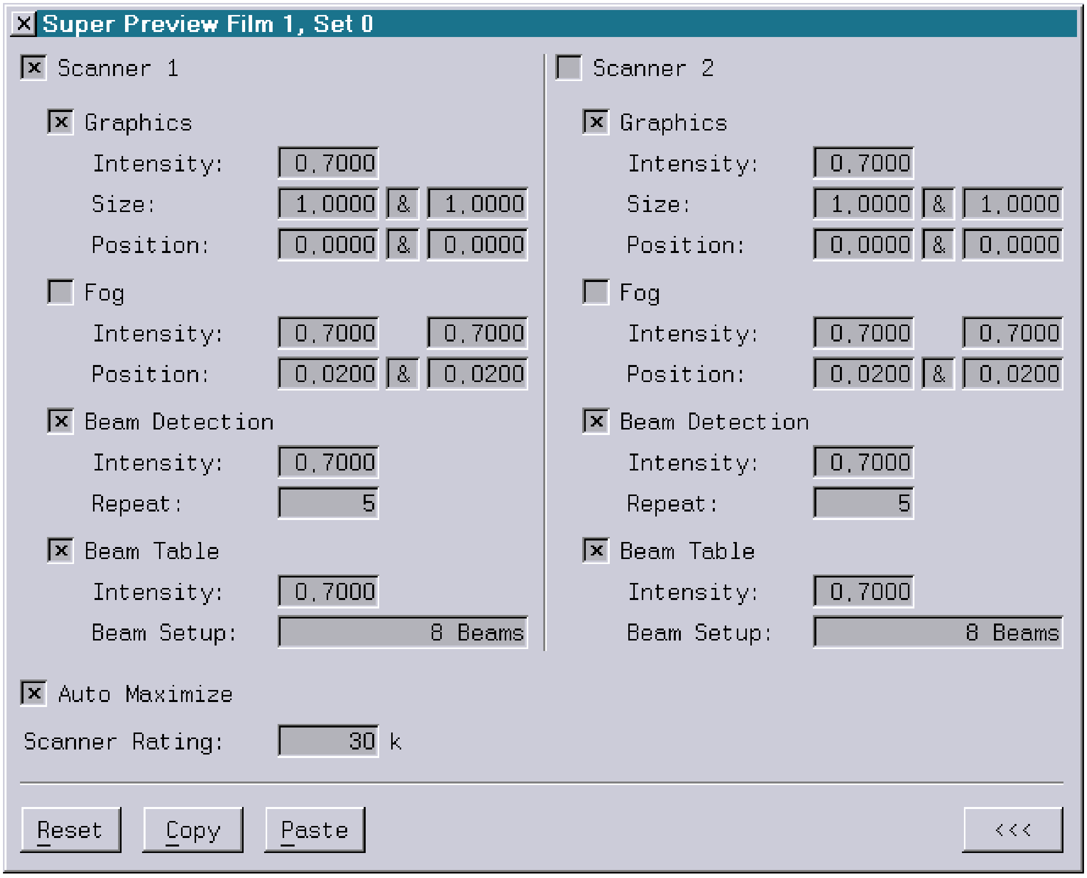
Copies all Super Preview parameters to the clipboard so that they can be transferred to another Film or another DGC Set.

Paste

Overwrites the current Super Preview parameters using the Super Preview parameters from the clipboard.

">>>"

Opens the enhanced Super Preview setup where both scanners can be adjusted individually:



☒ Super Preview Film 1, Set 0

☒ Scanner 1

☒ Graphics

Intensity: 0.7000

Size: 1.0000 & 1.0000

Position: 0.0000 & 0.0000

□ Fog

Intensity: 0.7000 0.7000

Position: 0.0200 & 0.0200

☒ Beam Detection

Intensity: 0.7000

Repeat: 5

☒ Beam Table

Intensity: 0.7000

Beam Setup: 8 Beams

□ Scanner 2

☒ Graphics

Intensity: 0.7000

Size: 1.0000 & 1.0000

Position: 0.0000 & 0.0000

□ Fog

Intensity: 0.7000 0.7000

Position: 0.0200 & 0.0200

☒ Beam Detection

Intensity: 0.7000

Repeat: 5

☒ Beam Table

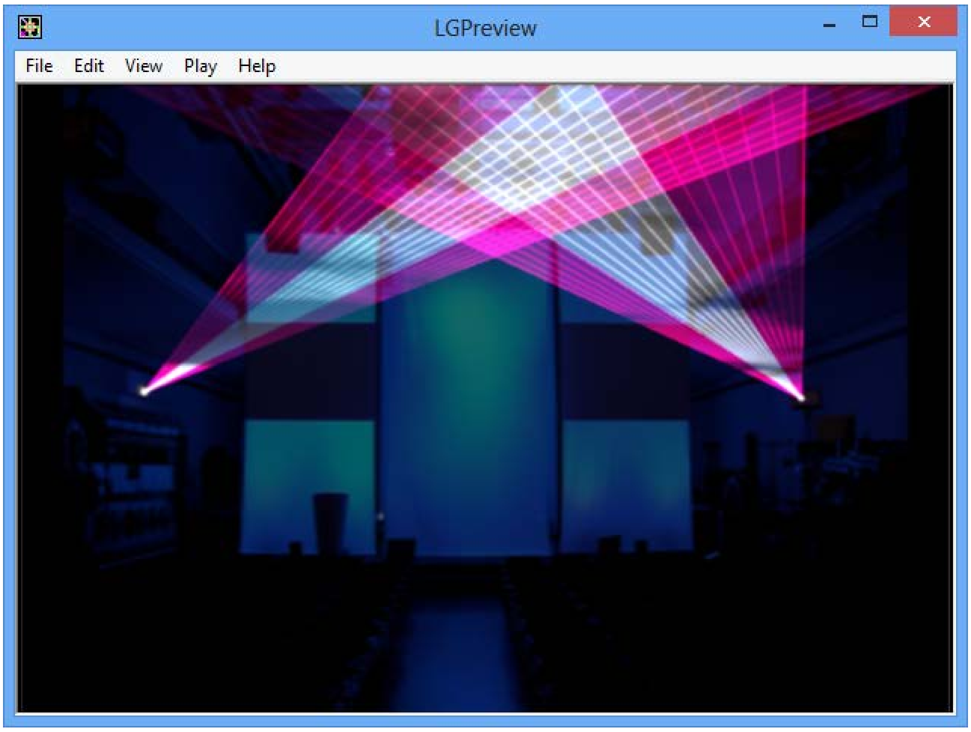
Intensity: 0.7000

Beam Setup: 8 Beams

☒ Auto Maximize

Scanner Rating: 30 k

This way the position and projection direction of the two virtual projectors can be better adapted to a background bitmap.



Menu Bar – Film Selected, Not Running

[Run]

This item starts the currently active film which is marked with a "*" in the "Film box" window. Once the film has been started, several additional functions are available.

[Edit]

Selecting this item opens the trickfilm editor where you can program the currently active film. (You can find a detailed description in the separate chapter "Edit Mode").

[File]

Calls up the Disk Operating System. (You can find a detailed description in the separate chapter "File")

Piced

Calls up the picture editor with which you can produce pictures and graphics for your laser show. (You can find a detailed description in the separate chapter "Picture Editor")

[Keyconf]

This item brings you to the "Key Config" where you can assign various functions to the keys of your Midi Keyboard.

ReDirect

With this item you may set up which film should run on which scanner. Up to four independent scanner sets may be connected to the *Workstation Mark 2*.

PassWord

Enables installing and handling passwords to avoid undesired access to your *Mark 2*.

PassWord.[Enter]

Enter a previously installed password to gain access to the computer.

PassWord.Install

Here you may install a password in your *Mark 2*.

PassWord.Install /Boot
Installs the password for the system software. Without entering the proper password, you do not have access to the *Mark 2*.

PassWord.Install /Harddisks
Installs the password for the hard drive. Without entering the proper password, you do not have access to the hard drive.

PassWord.[DeInstall]

Use this to remove an installed password.

SetTime

Defines a time for the Internal Timecode.

ReBoot

Used to reboot a directory containing the film, pictures, text, and various other parameters.

TestPattern

Turns the test picture of a specific film ON or OFF.

SetBeam

This item is used to set beams.
These beams are "Scanned Beams", which means they will be created with the scanner projection unit, not with an optical bench.
These beams can focused on mirrors or gratings.

SetBeam.[Window]

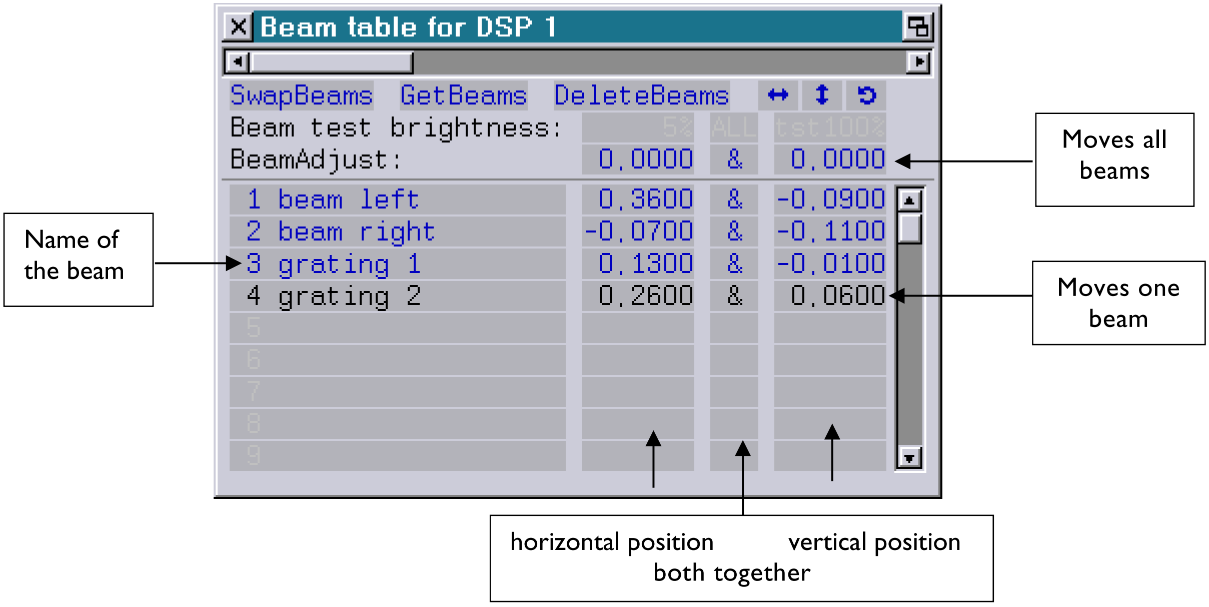
Opens the "Beam table" for DSP 1 to DSP 4. A Beam table contains names and positions of beams (1 to 64) which can be used in the current film. There is a Beam table for every DSP to adjust beams easily and quickly.
Use the horizontal scroll bar in the window to select the corresponding beam table.

The beam settings for different DSP cards can be swapped (SwapBeams) and replaced (GetBeams) with each other or deleted (DeleteBeams).

Enter a name for a beam and adjust the horizontal and/or vertical position of the beam.
In addition it is possible to adjust the brightness and the position of all beams simultaneously.
With the Beam test brightness you can adjust the brightness of the beams while adjusting them. This value will be ignored when scanned beams are used in a film.

Use the “tst100%” field to set the output to maximum for testing. The field “ALL” can be used to display all beams while adjusting them.

All values of the Beam table can be changed by Mouse Poti Mode immediately.



Beam table for DSP 1

Beam test brightness:	5%	ALL	tst100%
BeamAdjust:	0,0000	&	0,0000
1 beam left	0,3600	&	-0,0900
2 beam right	-0,0700	&	-0,1100
3 grating 1	0,1300	&	-0,0100
4 grating 2	0,2600	&	0,0600
5			
6			
7			
8			
9			

Moves all beams

Name of the beam

Moves one beam

horizontal position vertical position
both together

You can also use the corresponding commands in the command line to set a Beam table, for instance:

SetBeam.SwapBeams <with dsp>

Swap beams with another DSP (1 to 4).

SetBeam.GetBeams <from dsp>

Get beams from another DSP (1 to 4).

SetBeam.DeleteBeams /Names

Delete all beam positions.
If "/Names" is specified the name of the beams will also be deleted.

SetBeam.MirrorBeams.X

Mirrors all beam positions horizontally.

SetBeam.MirrorBeams.Y

Mirrors all beam positions vertically.

SetBeam.RotateBeams <angle> /Limit

Rotates all beam positions by a freely selectable angle.

Depending on the position of the beams and the specified angle, it may happen that after the rotation one or more beam positions would no longer be within the permitted value range of -1.000 to +1.000. In this case, the rotation is not performed and the command is aborted with a corresponding error message.

To perform the rotation anyway, the option "/Limit" can be specified. Beams whose positions after the rotation are no longer in the allowed value range are then moved into the allowed value range.

SetBeam.Name <number> <name>

Defines the name of a beam.
<number> must be in range from 1 to 64.

SetBeam.Testlevel <brightness>

Defines the brightness of the test level in percent (0 to 100).

SetBeam.AdjustX <value>

Defines the horizontal beam adjust value. It must be in range from -1.000 to +1.000.
The beam adjust affects all beam positions.

SetBeam.AdjustY <value>

Defines the vertical beam adjust value. It must be in range from -1.000 to +1.000.
The beam adjust affects all beam positions.

SetBeam.X <number> <value>

Defines the horizontal beam position of the specified beam.
<number> must be in range from 1 to 64.
<value> must be in range from -1.000 to +1.000.

SetBeam.Y <number> <value>

Defines the vertical beam position of the specified beam.
<number> must be in range from 1 to 64.
<value> must be in range from -1.000 to +1.000.

DarkScreen

A so-called screensaver is provided by the *Mark 2* which blackens your screen after a set length of time during which no input was made to the computer. You may set this time and the degree of blackening. As soon as a key is pressed or the mouse is moved, the monitor screen reappears in its original form.

[Mapconf]

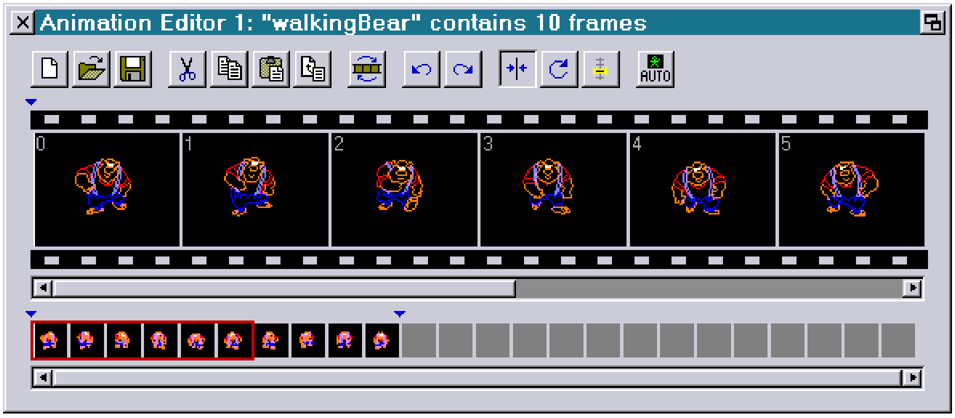
Allows access to a menu where control parameters may be set for the laser benches.

[Bankconf]

Allows access to a menu where your mechanical laser setups can be labeled in the computer. This way you are able to assign program functions to laser benches during map configuration.

AnimEd

Opens the Animation Editor. This is a tool which was designed for comfortably creating new animations and editing existing ones, see separate chapter "Animation Editor".



Animation Editor 1: "walkingBear" contains 10 frames

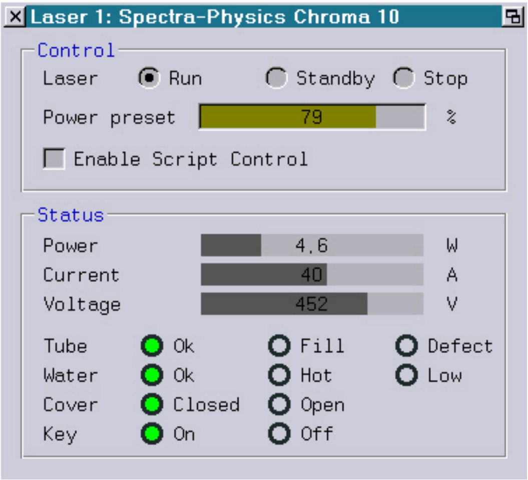
Note:
At first the "Memory" (Picture Memory) will be opened by selecting "AnimEd" to load existing animations or catalogs into the Animation Editor.

LASER

This item is used to handle the laser -> compare separate chapter "Setup".

LASER.Open <Laser>

Opens the window for the remote controlled laser(s) 1 to 4, e. g. for "Laser 1", laser type "Chroma 10":



Laser 1: Spectra-Physics Chroma 10

Control

Laser ● Run ○ Standby ○ Stop

Power preset 79 %

□ Enable Script Control

Status

Power 4.6 W

Current 40 A

Voltage 452 V

Tube ● Ok ○ Fill ○ Defect

Water ● Ok ○ Hot ○ Low

Cover ● Closed ○ Open

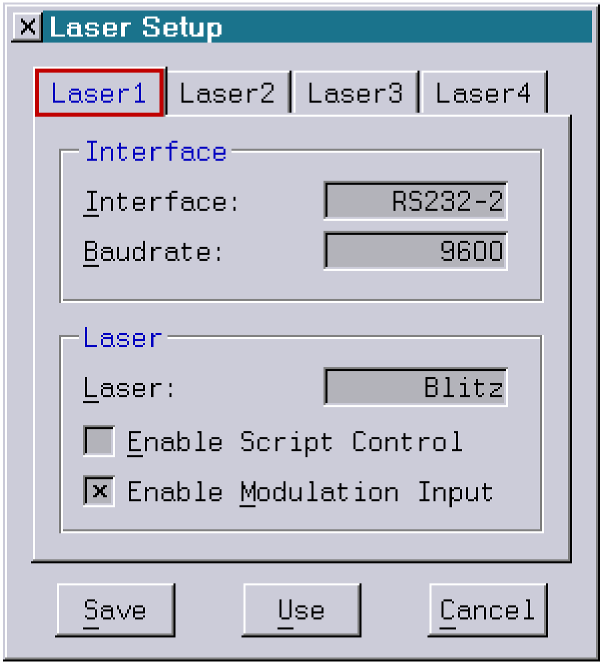
Key ● On ○ Off

LASER.Close <Laser>

Closes the laser window 1 to 4.

LASER.Setup

Opens the window for the Laser Setup:



Interface

Interface: RS232-2

Baudrate: 9600

Laser

Laser: Blitz

☐ Enable Script Control

☒ Enable Modulation Input

LASER.Run <Laser>

The remote controlled laser will be started with this command (1 to 4).

LASER.StandBy <Laser>

Turn on the laser to minimum power.

LASER.STop <Laser>

Turn off the laser.

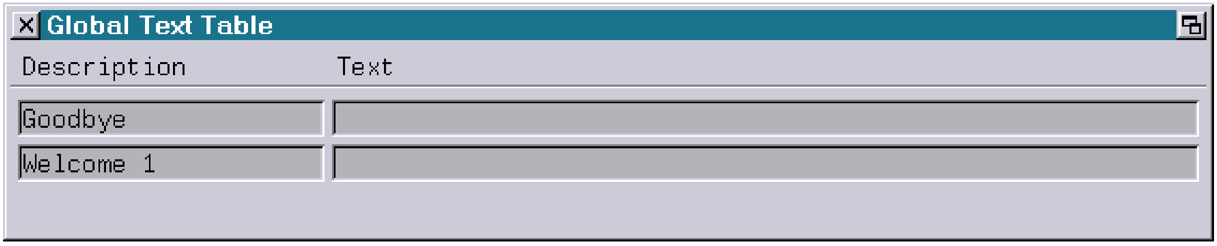
LASER.Preset preset <Laser>

Presets the laser power in % (in a range from 0 – 1, 0=0%. 1=100%).

[GlobalTEXT]

This item opens the Global Text Table. The table contains all texts defined by the Trickfilm command DeFineText.

Note: If there is no DeFineText command in the film, the Global Text Table will be empty.

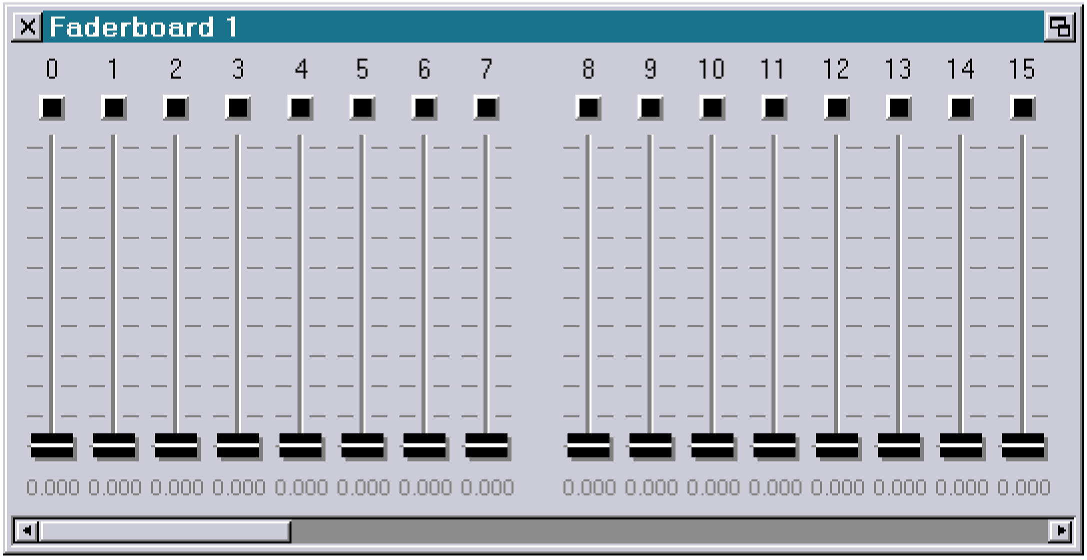


Description	Text
Goodbye	
Welcome 1	

[FADER]

By clicking this item a software emulation of the faderboard (virtual faderboard) will be opened.

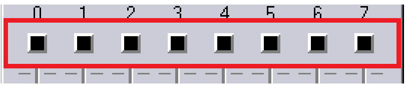
The command FADER with the optional parameter <Faderboard> (1, 2, 3 or 4) can be used to open up to four virtual faderboards.



The virtual fader board serves two different functions: It shows the current state of the fader values and the fader buttons as defined by a physical fader board or an I/O-Server and it can be used to modify the fader values and the fader button states itself.

It is divided into three different parts:

1. Buttons



Each fader button can be pressed ■ or released ■.

A fader button is pressed by clicking on it with the left or middle mouse button.
It stays pressed until the mouse button is released.

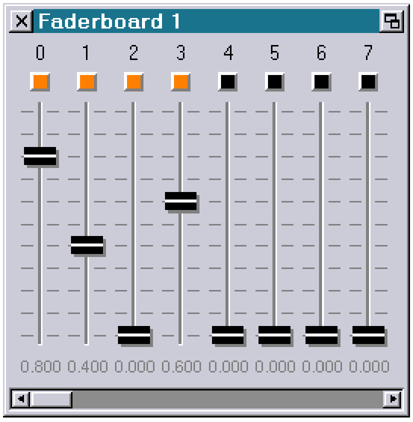
If the mouse button is clicked while <SHIFT> is pressed, the fader button stays pressed even after the mouse button is released.

The state of the buttons can be queried within the Trickfilm using the function "FaderB()".

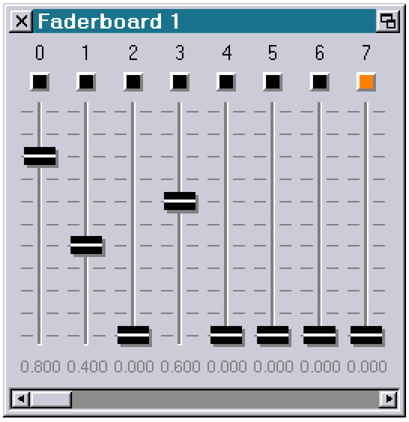
Fader buttons can have two colors: black ■ and orange 🟧.

The color is used to symbolize special functions:

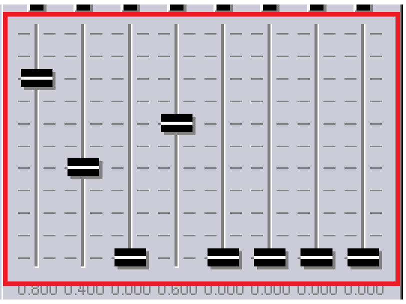
Within the Piced with faders enabled ("Faders default" or "Faders user") the fader buttons above the faders used are shown in orange:

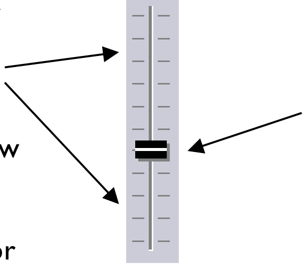


In Edit Mode the fader button above fader 7 can be used to enable "LoopCnt by Fader 7" where the LoopCnt can be controlled using fader 7.



2. Fader





A single click with the left or middle mouse button above or below the fader knob sets the fader to the desired position.

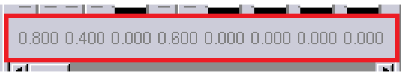
A click and hold above or below the fader knob starts to move the fader in this direction until the mouse button is released or the end of the fader is reached.

The fader knob can be grabbed to modify the fader value with the mouse.

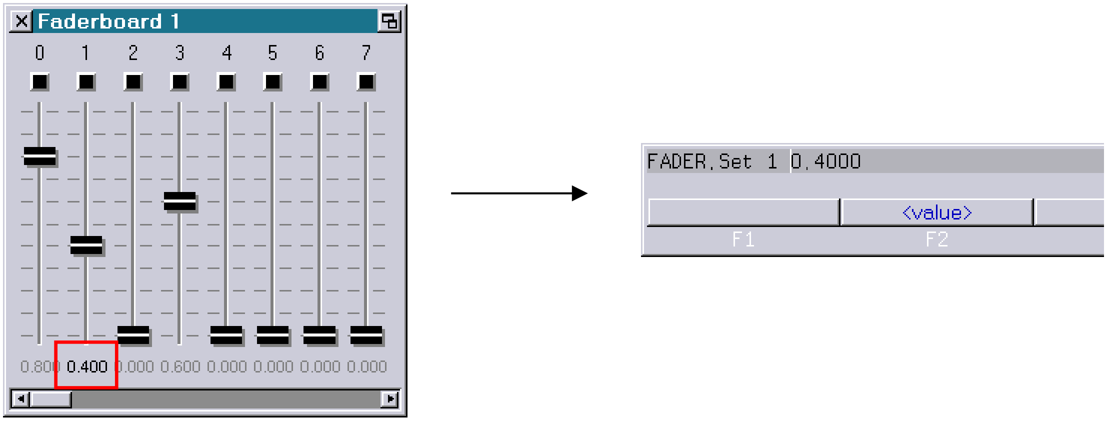
If the middle mouse button is used instead of the left one, the value change is performed in fine mode.

The value of a fader can be queried within the Trickfilm using the function "Fader()".

3. Values



A single click on one of the values puts the command to enter a value into the command line:



```
FADER.Set 1 0.4000
```

If the mouse button stays pressed, the mouse can be used to directly modify the value (mouse poti mode).

FADERboard.Open <Faderboard>

Opens the specified virtual faderboard. If no faderboard is specified the next free faderboard will be opened.

FADERboard.Close <Faderboard>

Closes the specified virtual faderboard.

FADERboard.Set <Fader> <Value1> [<Value2> ... <Value16>]

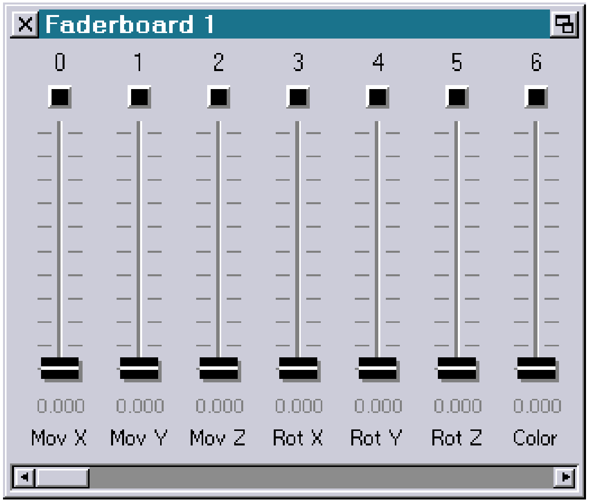
Defines the value of up to 16 of the 64 faders.
<Fader> must be in range from 0 to 63.
<Value> must be in range from 0.000 to 1.000.

FADERboard.Name <Fader> <Name1> [<Name2> ... <Name8>]

Labels individual faders within the virtual faderboard.
The parameter <Fader> determines the first fader to be labeled.
Up to 8 faders can be labeled in one go.
If a label is to contain spaces, the name has to be put in quotation marks.

Example:

```
FADERboard.Name 0 MovX MovY MovZ
FADERboard.Name 0 "Mov X" "Mov Y" "Mov Z" "Rot X" "Rot Y" "Rot Z" Color
```



FADERboard.Name.Clear <Fader>

Removes the label from the specified fader on the virtual faderboard.
<Fader> must be in range from 0 to 63.

FADERboard.Name.ClearAll

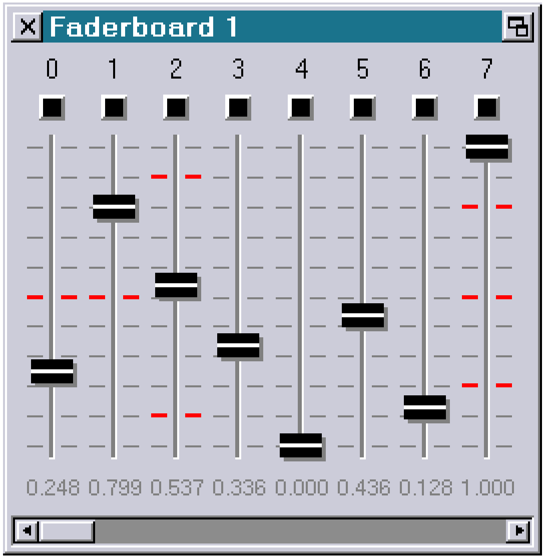
Removes all labels from the virtual faderboard simultaneously.

FADERboard.Marker <Fader> <Value1> [<Value2> ... <Value5>]

Places up to 5 markers on the specified fader within the virtual faderboard.
<Fader> must be in range from 0 to 63.
<Value> must be in range from 0.000 to 1.000.

Example:

```
FADERboard.Marker 0 0.5
FADERboard.Marker 1 .5
FADERboard.Marker 2 .1 .9
FADERboard.Marker 7 .2 .5 .8
```



FADERboard.Marker.Clear <Fader>

Removes all markers from the specified fader on the virtual faderboard.
<Fader> must be in range from 0 to 63.

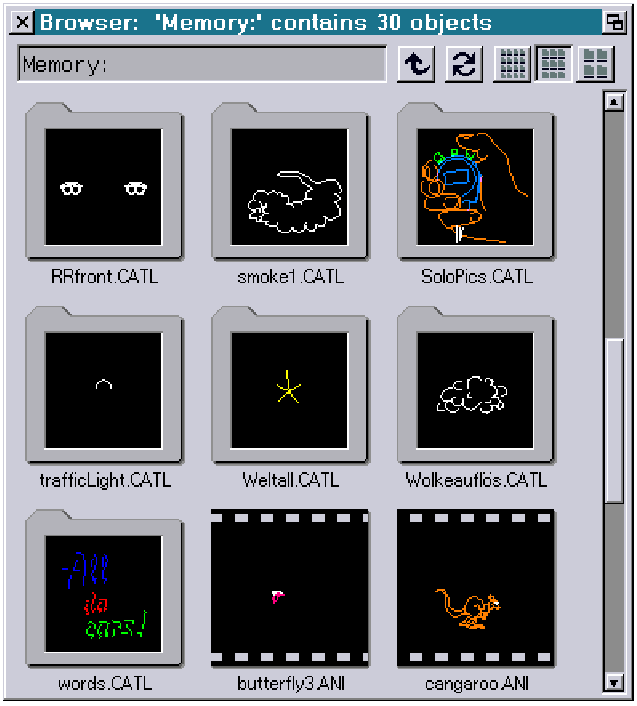
FADERboard.Marker.ClearAll

Removes all markers from all faders on the virtual faderboard.

[BROWSER]

This command opens the picture browser, which displays all pictures of catalogs and animations and so allows simple and fast selection of pictures.

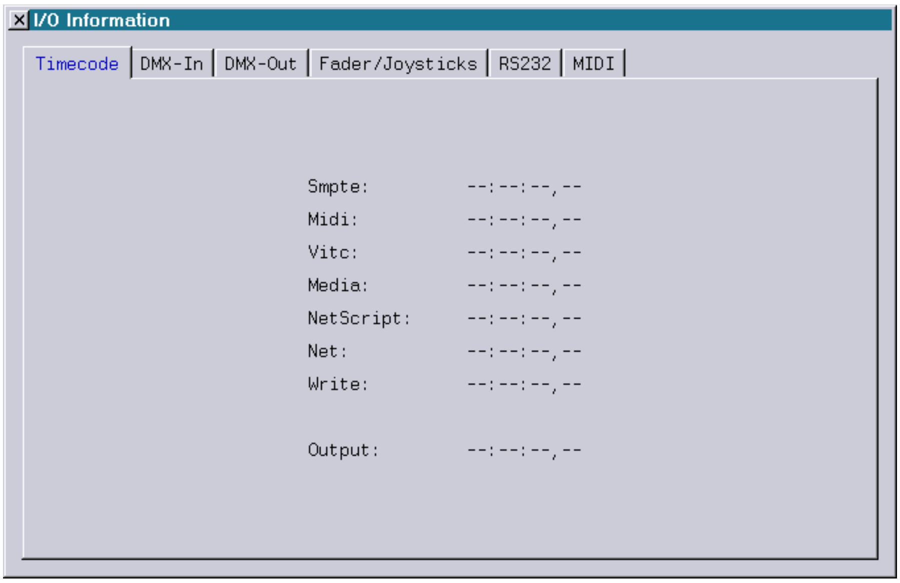
Note:
See separate chapter "File" for details.



[IOINFO]

Opens a window to see the I/O (Inputs/Outputs) of the *Mark 2*,
e.g. if You want to double-check the Timecode coming into the *Mark 2* or the DMX signals being received or being sent.

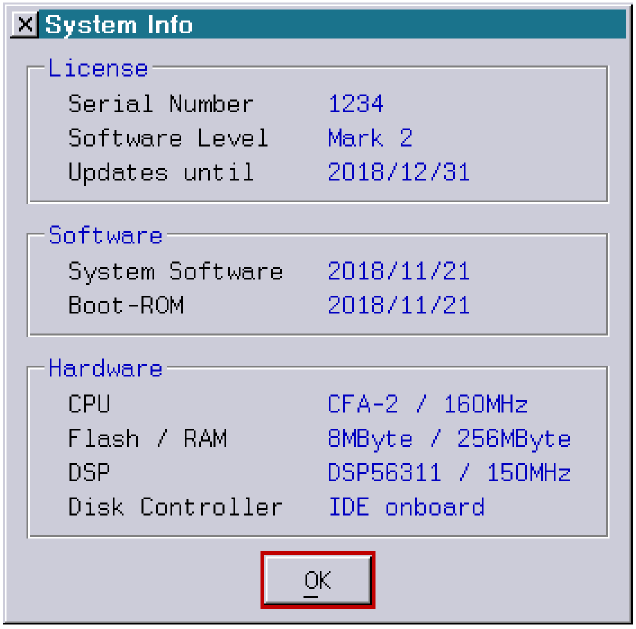
Note:
The separate chapter “I/O Information” contains a detailed description.



```
Smpte:       --:--:--,--
Midi:        --:--:--,--
Vitc:        --:--:--,--
Media:       --:--:--,--
NetScript:   --:--:--,--
Net:         --:--:--,--
Write:       --:--:--,--

Output:      --:--:--,--
```

[INFO]

You will get more general information such as serial number, installed software version etc. by entering the command INFO into the command line:



```
License
  Serial Number       1234
  Software Level      Mark 2
  Updates until       2018/12/31

Software
  System Software     2018/11/21
  Boot-ROM            2018/11/21

Hardware
  CPU                 CFA-2 / 160MHz
  Flash / RAM         8MByte / 256MByte
  DSP                 DSP56311 / 150MHz
  Disk Controller     IDE onboard
```

WriteSMPTE

WriteSMPTE.On <timecode> [<data>]

Starts the output of SMPTE (LTC) timecode at the specified time e.g. for recording on tape.

While SMPTE timecode is written it is possible to send additional user data.
The user data can be one of the following:

#<bootname> All receiving Lasergraph-DSP will automatically load the specified BOOT.
The BOOT has to be located in the directory "C:\AutoLoad".

<string> The specified string will be displayed in the status line of the receiving Lasergraph-DSP.

WriteSMPTE.OFF

Stops the output of SMPTE (LTC) timecode.

WriteSMPTE.Data <data>

While SMPTE timecode is written this command can be used to send new user data.

SERCMD

Gives you access to the serial commands feature.
For more details see chapter "SerCmd".

SERCMD.[Console]

Opens the SerCmd Console window.
The SerCmd Console window is used to display incoming "<wprint" and "<wprintln" commands.

SERCMD.SENDstring <string>

Sends the specified string to control an external device.

SERCMD.[CLS]

Clears the content of the SerCmd Console window.

SERCMD.[Setup]

Opens the SerCmd Setup.

MACro

The Lasergraph DSP Mark 2 supports freely definable keyboard shortcuts.
When the desired keyboard shortcut is entered, a sequence of previously recorded commands is executed. This command sequence is called macro.

MACro.Record

This command is used to record a macro and assign it to a keyboard shortcut.

There are two different kinds of keyboard shortcuts:

[bykey] The macro is executed if the corresponding keyboard shortcut is pressed, independent of the operating mode the Mark 2 is in.

[bykeymode] The macro is only executed if the Mark 2 is in the same operating mode as it was when the macro was recorded.

Different operating modes are for example:

Start screen
File
Piced
Trickfilm Edit
Timescript Edit

To record a macro, perform the following steps:

1) Enter "MACro.Record"
2) Select "[bykey]" or "[bykeymode]"
3) Press the key or key combination you want to assign the macro to
4) Select "[ok]"
5) Now enter all commands you want to record
6) When finished press <CTRL>-<C> to quit recording

Note:

When recording Macros, the commands have to be entered using the keyboard.
If menu items are selected using the mouse these actions are not recorded!

Options:

/Timing If is selected, the macro will be executed at the same speed as when it was recorded.

/Delayed If selected, the recording of the macro does not start immediately.
To start recording press <CTRL>-<S>.

MACro.SetPath

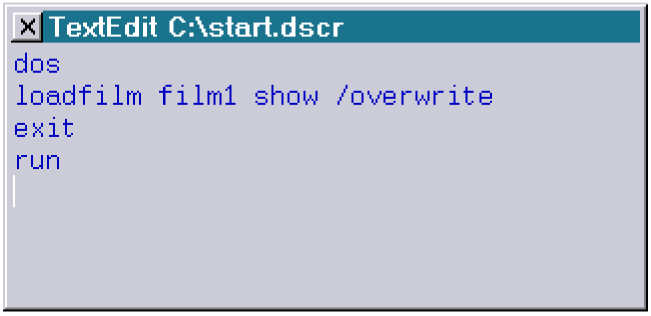
The default path for the macros is "C:\KeyMacros".
This command can be used to select another location where the macros are loaded from.

TextEDIT

Opens a simple text editor. You can load an existing text by typing the path and name (e.g. C:\example.txt) or by creating a new name.

startSCRIPT

Commands which are entered into the input line one after the other and which are then to be carried out in a specific order can be saved in a script (text file) in the format ***.dscr**. With the command startSCRIPT the commands are then automatically executed step by step, e.g. startSCRIPT show.dscr. In principle this works similarly to macros, with the advantage that the recorded scripts *.dscr can be edited in the text editor TextEDIT or in an external Editor (e.g. WordPad).



TextEdit C:\start.dscr

```
dos
loadfilm film1 show /overwrite
exit
run
```

For more information about scripts see separate chapter "Scripting".

ScriptContinueOnError

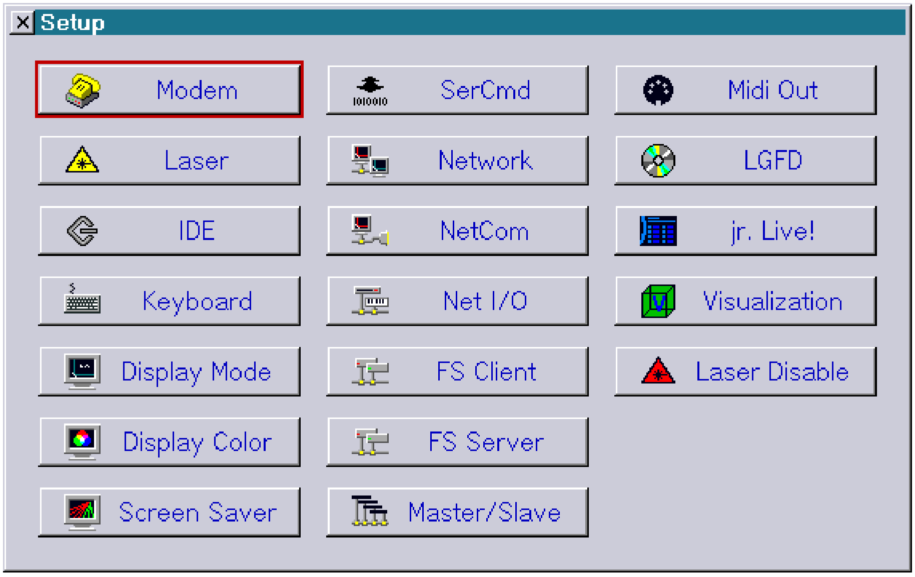
Defines whether a script will be stopped or executed to its end if an error occurs.

EnableClearOutputs

Changes the handling of DMX outputs.
The default state is "enabled" (YES). In this mode, the Mark 2 will set all DMX Channels to "0" (reset) when a film or show is stopped or when switching from one part to another (e.g. from EDIT to Piced).
When the state is disabled (OFF), the Mark 2 will keep the last state of all DMX Channels.

[SETUP]

Opens the Setup window, a detailed description s in the separate chapter "Setup".



Basic settings of the system are adjustable in the Setup. You can configure which external devices, e. g. Modem, Laser, Keyboard, are connected. You can carry out settings for the operation in the network, you can adapt the desktop resolution to the monitor, you can change the colors of your screen surface and so on.

SaveWinPos

Saves the current window positions of your Mark 2 screen.
After changing position and/or size of any window, a restart of the system will put the windows into their default positions and sizes.
When you save the window positions, these positions will be used as default.
Window positions are stored independently for each desktop resolution.

SHUTDOWN

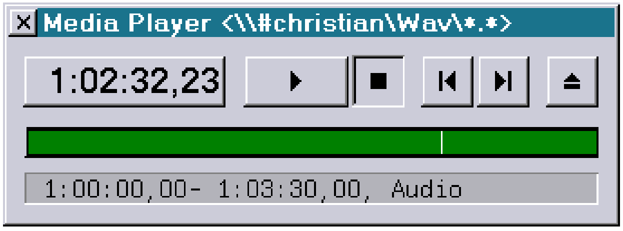
Turns the computer off. It makes sure that all running processes are properly terminated, unsaved films are saved in the directory "C:\Security". (Equivalent to the option 'Shut down' of a PC under Windows® XP/Vista/7/8/8.1/10).

RESTART

It restarts the system, if you do not cancel it within 3 seconds. (Equivalent to the option 'Restart' of a PC under Windows® XP/Vista/7/8/8.1/10).

[MediaPlayer]

Opens the MediaPlayer window:



1:02:32,23

1:00:00,00- 1:03:30,00, Audio

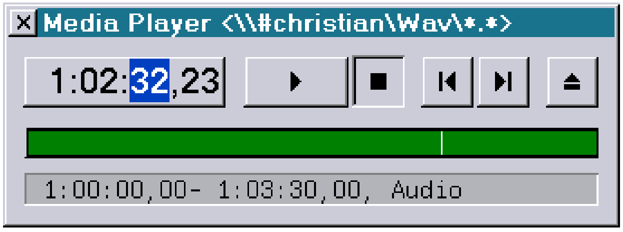
The Media Player is used to play back audio or video on a PC in conjunction with the media server "LGServer". It can be controlled from the Timescript or by using the buttons "Play", "Stop", "Previous Track", "Next Track" and "Eject/Load".

When the LGServer is configured as "CDRom" or "Wav/Avi" server and the PC is entered in the "I/O Setup" of the Lasergraph DSP, the Media Player window will open automatically. If it was closed or does not open automatically you can open it with the command "MediaPlayer".

The name or the IP address of the media server is displayed in the headline of the Media Player. It looks like this:

\\#christian\Wav	Wav/Avi server on a PC with the name "christian"
\\#christian\CDRom.E	CDRom drive E: on a PC with the name "christian"
\\#192.168.1.100\Wav	Wav/Avi server on a PC with the IP address 192.168.1.100

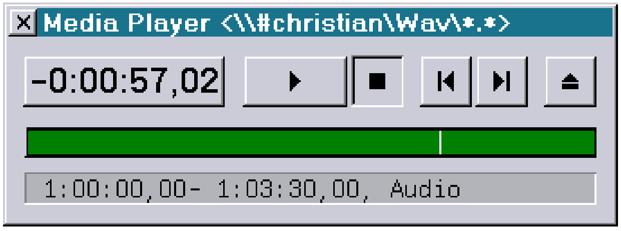
The current time is displayed in the upper left part of the Media Player.
By clicking into the time with the left mouse button the time can be increased, by clicking with the middle mouse button the time can be decreased:



1:02:32,23

1:00:00,00- 1:03:30,00, Audio

A click into the time display with the left mouse button while <CTRL> is pressed toggles the display between elapsed time and time remaining until the end of the current track.

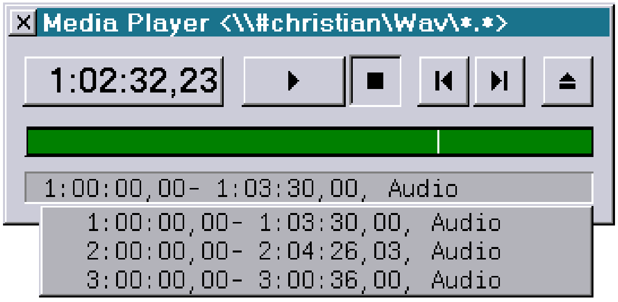


-0:00:57,02

1:00:00,00- 1:03:30,00, Audio

The white line in the green track view represents the relative playback position. By clicking into the track the playback position can be changed.

If more than one track is available you can select the desired track using the track select dropdown list:



1:02:32,23

```
1:00:00,00- 1:03:30,00, Audio
   1:00:00,00- 1:03:30,00, Audio
   2:00:00,00- 2:04:26,03, Audio
   3:00:00,00- 3:00:36,00, Audio
```

[CopyRight]

Note on copyrights.

Menu Bar – Film Selected, Running

[Run]

If you have several DSP cards (1 to 4) in the system, additional films (1 to 4) can be started.

[Break]

Stops all currently running films.

Goto

Allows you to go to a particular entry number from where the film should start. If the film is already running, you may nevertheless choose another entry from where the film should continue.

[GotoNext]

Skips to the next higher entry number for the currently running film.

[GotoPrev]

Skips to the next lower entry number for the currently running film.

GlobalPars

You can also influence the 'Global Parameters' separately for each film using the command line.

GlobalPars.Swap

Exchange parameters between two films

GlobalPars.Get

Copies parameters from one film to another

GlobalPars.Delete

Deletes the global parameters for the current film.

GlobalPars.ScreenPars

GlobalPars.ScreenPars.Size
Output size of the film may be adjusted, e.g. to fit a projection screen. A value of 1.0 corresponds to full size.

GlobalPars.ScreenPars.HSize
Sets horizontal size of film. Negative values cause mirroring in the vertical. A value of 1.0 corresponds to full horizontal size.

GlobalPars.ScreenPars.VSize
Sets vertical size of film. Negative values cause mirroring in the horizontal. A value of 1.0 corresponds to full vertical size.

GlobalPars.ScreenPars.HShift
Allows for horizontal shifting of the film output. A value of 0.0 indicates there is no shifting.

GlobalPars.ScreenPars.VShift
Allows for vertical shifting of the film output. A value of 0.0 indicates there is no shifting.

GlobalPars.ScreenPars.XMIN
Allows to set a minimum horizontal clipping border of picture.

GlobalPars.ScreenPars.XMAX
Allows to set a maximum horizontal clipping border of picture.

GlobalPars.ScreenPars.XMAXMIN
Allows to set maximum and minimum horizontal clipping borders of picture simultaneously.

GlobalPars.ScreenPars.YMIN
Allows to set a minimum vertical clipping border of picture.

GlobalPars.ScreenPars.YMAX
Allows to set a maximum vertical clipping border of picture.

GlobalPars.ScreenPars.YMAXMIN
Allows to set maximum and minimum vertical clipping borders of picture simultaneously.

GlobalPars.Advanced

Allows to set the 'Advanced Display Parameters' as in the "Display Parameters" window opened from the "Globals" via the command line.

GlobalPars.Advanced.Default
Set default for 'Crystal', 'Scanner', 'Scanner RGB', 'Isp30K', 'Blitz', 'Phaenon30k' or ‘Phaenon60k’.

GlobalPars.Advanced.Freq

You can adjust the scanner frequency to optimize the quality of the laser output. However, a value of 256kHz is the maximum but it is very demanding on the scanners. Usually it is enough to reset the frequency a little.

GlobalPars.Advanced.MaxdistD

Allows to set the maximum distance drawn.

GlobalPars.Advanced.MaxdistB

Allows to set the maximum distance blanked.

GlobalPars.Advanced.TSP

(TSP=To Start Point Repeat, see "Advanced display parameters")
When the last point is reached during projection, the scanners will jump back to the first point. This value defines how many points will be inserted before the next picture is drawn.

GlobalPars.Advanced.VBON

(VBON=Pre blank on repeat, see "Advanced display parameters")
Defines how many points are inserted before the Blanking is turned on.

GlobalPars.Advanced.NBON

(NBON=Post blank on repeat, see "Advanced display parameters")
Defines how many points are inserted after Blanking has been turned on.

GlobalPars.Advanced.VBOFF

(VBOFF=Pre blank off repeat, see "Advanced display parameters")
Defines how many points are inserted before the Blanking is turned off.

GlobalPars.Advanced.NBOFF

(NBOFF=Post blank off repeat, see "Advanced display parameters")
Defines how many points are inserted after Blanking has been turned off.

GlobalPars.Advanced.ColorDelay

Here the delay of the projection before a change of color is set. The parameter is used to correct picture changes due to the inertia of the scanners as compared to the crystal (PCAOM). This is especially important for the bitmap function but it may also be useful to correct general errors; for instance a line that changes color without blanking appears differently in the projection than in the "Piced" as long as the "Color delay" is not set.

GlobalPars.Advanced.ColorInv

Is necessary when color controls that work with inverted outputs are used. In this case "ColorInv" has to be activated.

GlobalPars.Advanced.SlowColor

Should be switched to "On" if color and blanking are not controlled using a crystal but with scanners or color box.

GlobalPars.Advanced.CSwitchRepeat

(Slow color switch repeat) When very slow color controls are used, it is possible to increase the delay ("Repeats") when a color change is projected.

GlobalPars.Advanced.3DEyeCtrl
Control for both or the right or left eye of the viewer.

GlobalPars.Advanced.EyeDist
Determines the distance between the right and the left eye of the viewer.

GlobalPars.Advanced.PlaneDist
Defines the virtual distance of the picture in space.

GlobalPars.Advanced.3DMode
For stereoscopic 3D projections with 3D glasses, either full color or with the anaglyph method. This transforms the original picture into a red/blue or red/green picture (depending on the type of glasses used). This method produces the best 3D effect, however with the result of color losses.

GlobalPars.Advanced.RedByGreen
Determines the intensity for the left eye (red) - the default value is 0.1300, i.e. 13% red to 100% green.

GlobalPars.Advanced.RedByBlue
Determines the intensity for the left eye (red) - the default value is 0.0900, i.e. 9% red to 100% blue.

GlobalPars.Advanced.GreenByRed
Determines the intensity for the right eye (green) - the default value is 0.0000.

GlobalPars.Advanced.BlueByRed
Determines the intensity for the right eye (blue) - the default value is 0.1200, i.e. 12% blue to 100% red.

GlobalPars.Advanced.ReductDist
SetPointReduction Distance

GlobalPars.Advanced.SkipDist
SetPointReduction Skip DIstance

GlobalPars.Advanced.ShutterDef
Setting for the LogoShutter in the value range from 0 to 1

GlobalPars.Advanced.ShutterCorr
For fine setting / readjustment (range from –1 to +1)

GlobalPars.Advanced.SoftBlankUp
Determines globally how fast the SoftBlanking is turned on.

GlobalPars.Advanced.SoftBlankDown
Determines globally how fast the SoftBlanking is turned off.

GlobalPars.Advanced.DmxInOffset

For setting an offset for all DMX-In channels. Values from 0 to 511 can be entered, where the offset setting corresponds to the following assignments to the DMX channels

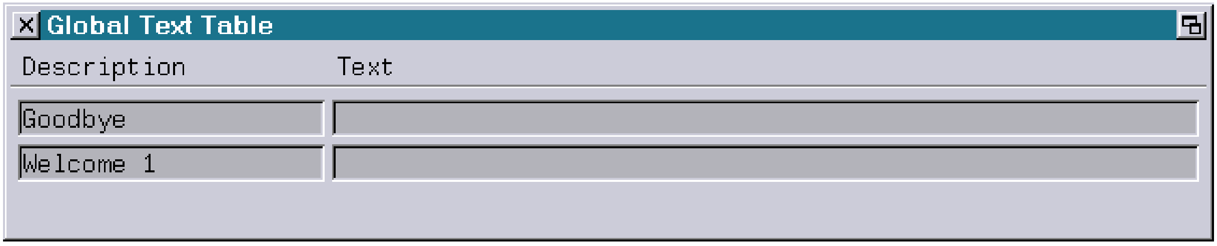
DMX channel	1	2	3	...	510	511	512
Offset=0	1	2	3	...	510	511	512
Offset=1	2	3	4	...	511	512	
Offset=2	3	4	5	...	512		
Offset=3	...	...	...	...	...	...	...

Note:

The "DMX-In offset" only affects the function "DmxIn()" inside the Trickfilm.
It does not affect the display of the DMX input in the "I/O Information" or the global brightness if controlled by DMX.

[GlobalTEXT]

Opens the Global Text Table. This table contains all texts defined by the Trickfilm command "DefineText". It is possible to change texts while the film is running.



Description	Text
Goodbye	
Welcome 1	

Menu Bar – Script Selected, Not Running

[Play]

This item is used to start the timescript which immediately reacts to each timecode that triggers an action defined in the timescript.
This item must be selected in order to run a timecode controlled show.

[Edit]

Selecting this item opens the trickfilm editor where you can program the currently active film. (You can find a detailed description in the separate chapter "Edit Mode").

[File]

Calls up the Disk Operating System. (You can find a detailed description in the separate chapter "File")

Piced

Calls up the picture editor with which you can produce pictures and graphics for your laser show. (You can find a detailed description in the separate chapter "Picture Editor")

[Keyconf]

This item brings you to the "Key Config" where you can assign various functions to the keys of your Midi Keyboard.

Record

Starts recording of Midi press and release actions.

[GetTime]

This button allows you to make a note of the current timecode to the "Auto Track/Frozen" window.

[more]

→ The next commands are the same as the commands in the [RUN] menu.

Menu Bar – Script Selected, Running

[Stop]

Stop the playback of a timescript/show.

SetTime

When a show is running with internal timecode, you can jump to a specific time.
This command will be ignored for external timecode!

[GetTime]

This button allows you to make a note of the current timecode to the "Auto Track/Frozen" window.

[Record On/Off]

Starts and Stops recording respectively.

GlobalPars

You can also influence the 'Global Parameters' separately for each film using the command line.
A detailed description of all sub-commands can be found in “Menu Bar – Film Selected, Running”.

[GlobalTEXT]

Opens the Global Text Table. This table contains all texts defined by the Trickfilm command "DefineText". It is possible to change texts while the film is running.

Chapter Setup

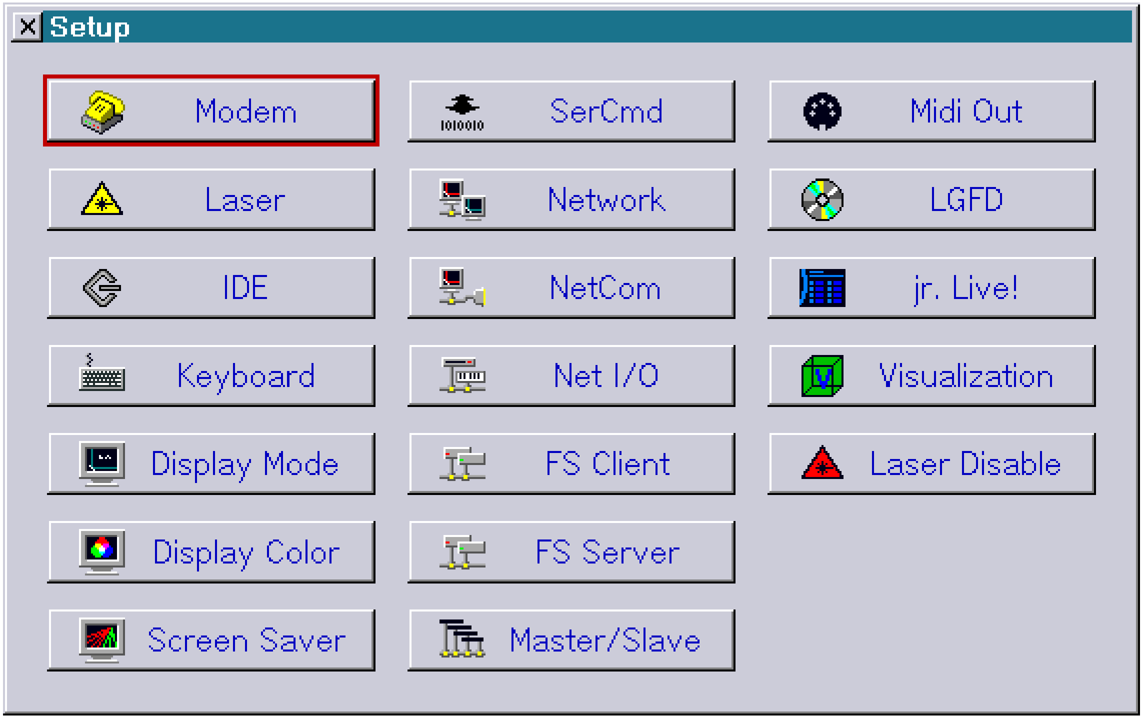
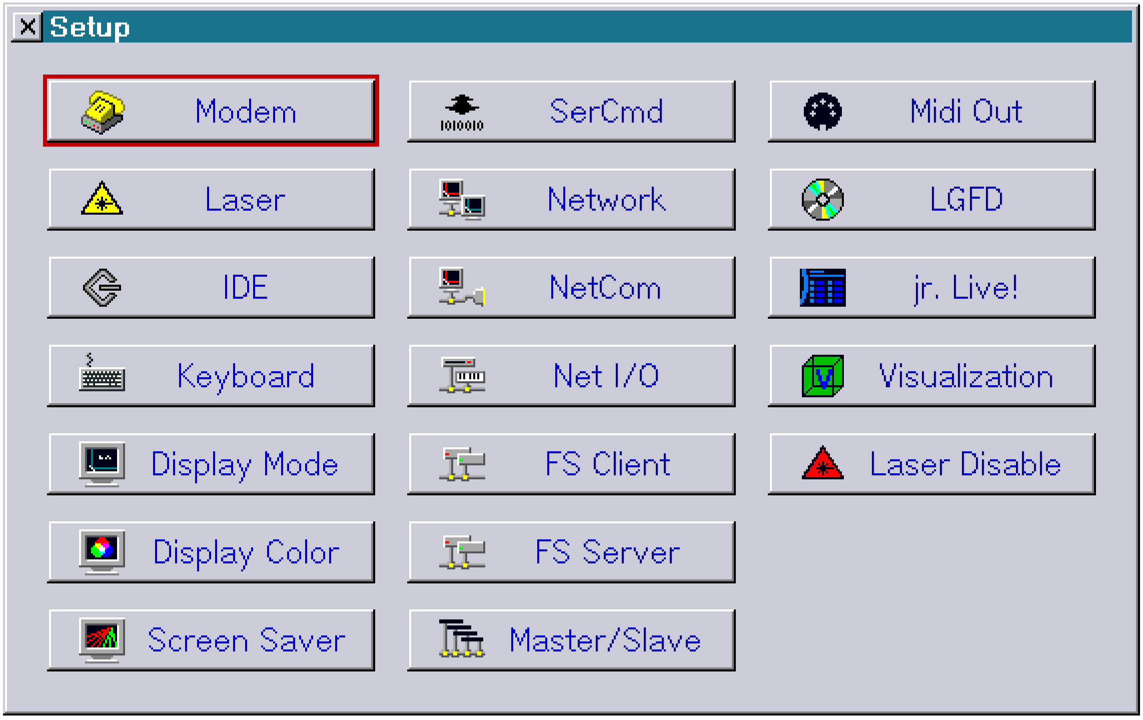


Table of Contents

Several parameters are adjustable in the Setup. Here you can configure which external devices, e. g. Modem, Laser, Keyboard, are connected. You can carry out settings for the operation in the network, you can adapt the desktop resolution to the monitor, you can change the colors of your screen surface and so on.

The Setup window is accessed by entering **SETUP** or by selecting 'SETUP' from the menu. The command SETUP opens the following windows.



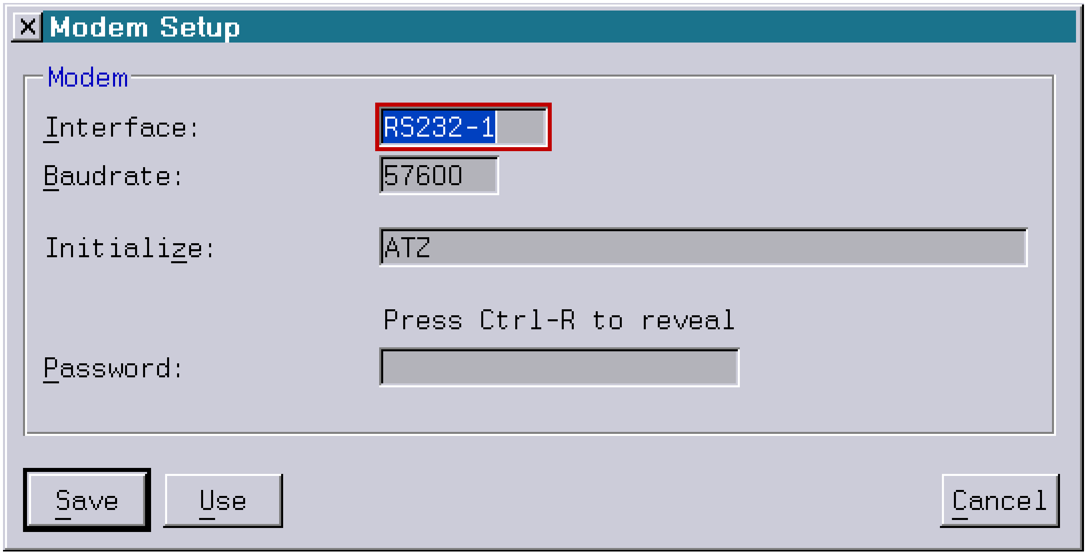
Pressing the <Esc> key or clicking the right mouse button closes the setup window.

For all settings you can choose the following:

Save	Use this setting as the standard setting from now on
Use	Use this setting until the Lasergraph DSP is turned off / restarted
Cancel	Discard all changes and restore the previously saved / used setting

Modem Setup

This settings are only necessary if you have a modem connected to the Lasergraph DSP.



Interface: RS232-1

Baudrate: 57600

Initialize: ATZ

Password:

In this window, the modem configuration can be changed permanently as opposed to the command **MODEM.Config**, where changes are only valid until the system is turned off.

Interface

The port the modem is connected to is selected with the interface popup menu.

Lasergraph DSP *compact*

None	The modem is deactivated.
RS232-1 to RS232-3	The modem is connected to one of the RS232 ports.
NetCom1 to NetCom16	The modem is connected to one of the NetCom ports

Lasergraph DP *workstation*

None	The modem is deactivated.
Modem	The modem is connected to the "Modem" port.
Tablet	The modem is connected to the "Tablet" port. Note that the ring indicator (pin 9) <u>must not</u> be connected because it is used as power supply +12V for the digitizer tablet.
NetCom1 to NetCom16	The modem is connected to one of the NetCom ports

Baudrate

One of the following baud rates can be selected:
9600, 19200, 38400, 57600, 76800, 115200, 153600, 230400 or 250000 baud.
Refer to the user’s manual of your modem to decide which baud rate should be used.

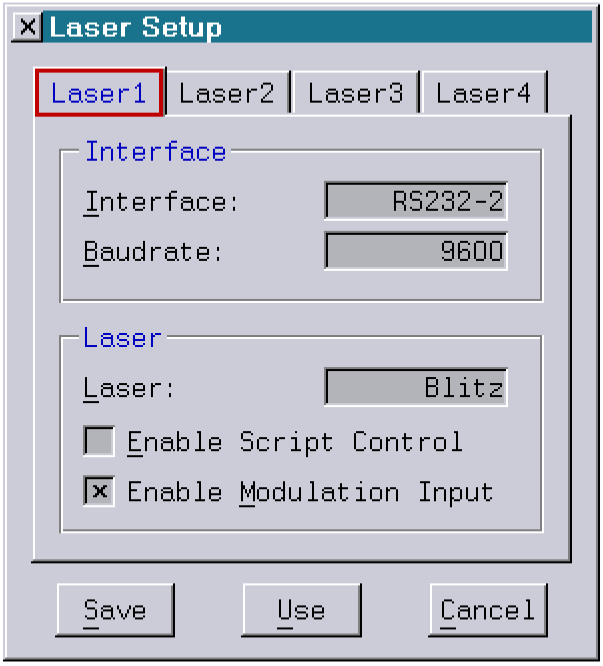
Initialize

May be changed if problems with the modem configuration occur.

Password

A password can be entered here to restrict download and remote control of this Lasergraph DSP. The password function is not case sensitive, it doesn't matter whether you use upper or lower case characters. To protect the password from prying eyes it is camouflaged with asterisks. If you are alone, you can turn off the camouflage by pressing <ctrl>+R.
After specifying a password, it has to be entered before downloading or connecting via remote control.

Laser Setup



Interface: RS232-2

Baudrate: 9600

Laser: Blitz

□ Enable Script Control

☒ Enable Modulation Input

This window can also be accessed by entering the command **LASER.Setup**.
Up to four lasers can be controlled. They are selected by clicking the appropriate tab in the upper part of the window.

Interface

The port the laser is connected to is selected with the interface popup menu.

Lasergraph DSP *compact*

None	Laser control is deactivated.
RS232-1 to RS232-3	The laser is connected to one of the RS232 ports.
NetCom1 to NetCom16	The laser is connected to one of the NetCom ports

Lasergraph DSP *workstation*

None	Laser control is deactivated.
Modem	The laser is connected to the "Modem" port.
Tablet	The laser is connected to the "Tablet" port. Note that the ring indicator (pin 9) <u>must not</u> be connected because it is used as power supply +12V for the digitizer tablet.
NetCom1 to NetCom16	The laser is connected to one of the NetCom ports

Baudrate

One of the following baud rates can be selected:
9600, 19200, 38400, 57600, 76800, 115200, 153600, 230400 or 250000 baud.
Refer to the user's manual of the laser to decide which baud rate should be used.

Laser

With the laser popup menu the laser type is selected. The following laser models are supported:

Chroma 5	(Spectra-Physics Chroma 5)
Chroma 10	(Spectra-Physics Chroma 10)
Verdi V5	(Coherent Verdi 5W)
Viper V8	(Coherent Viper 8W)
Viper V10	(Coherent Viper 10W)
58-GES-XXX	(Melles Griot 58-GES-XXX)
Blitz	(LaserAnimation Blitz)
WhiteLight	(JenLas WhiteLight)
Star 3.5	(Coherent PL Star 3.5W)

By activating the "Enable Script Control" checkbox, certain laser parameters can be controlled from the Timescript with the command:
SetLaser
By enabling the modulation input (only "Blitz" and "MelleGriot")
you can define if the laser will work in cw mode or will be modulated by color or blanking input.

All laser control windows are subdivided in two fields:

Control and ***Status***.

Control is used for settings.

Run Starts the laser.

Standby Turn on the laser to minimum power.

Stop Turn off the laser.

Power preset Setting for power (from minimum to maximum)

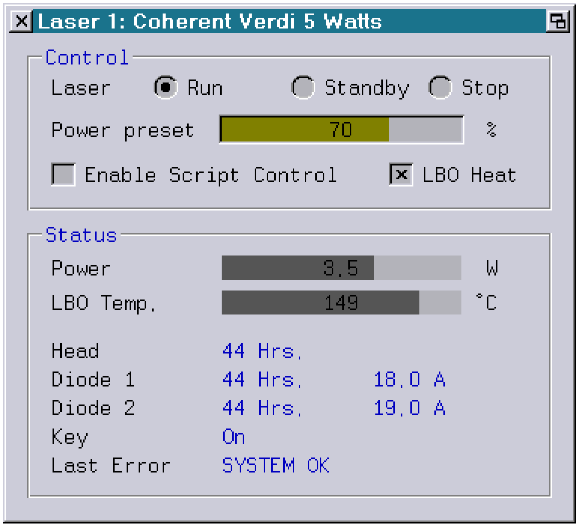
LBO-Heat *(only for Coherent lasers)* Here you can manually start or stop the LBO-heating.

Enable Script Control Enables / Dsables the control of the laser through the Timescript command "SetLaser"

Status includes control displays like *Power*, *Current*, *Voltage*, *LBO-Temp. Shutter*, *Interlock* etc.

Depending on the connected System you might get different information in the status field. See example screenshots below.

Coherent Verdi 5 Watts:



Laser 1: Coherent Verdi 5 Watts

Control

Laser ● Run ○ Standby ○ Stop

Power preset 70 %

□ Enable Script Control ☒ LBO Heat

Status

Power 3.5 W

LBO Temp. 149 °C

Head 44 Hrs.

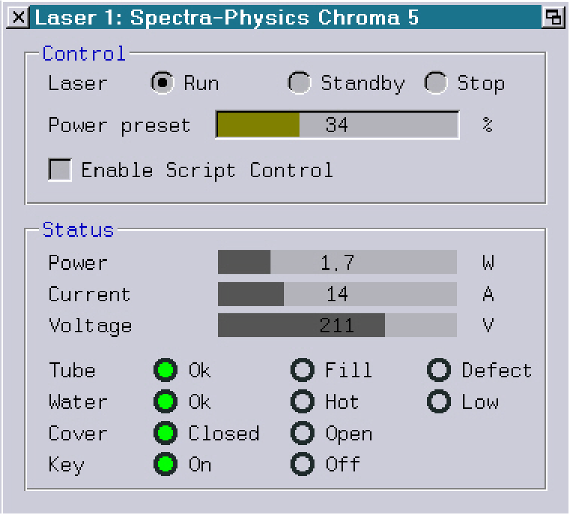
Diode 1 44 Hrs. 18.0 A

Diode 2 44 Hrs. 19.0 A

Key On

Last Error SYSTEM OK

Spectra-Physics Chroma 5:



Laser 1: Spectra-Physics Chroma 5

Control

Laser ● Run ○ Standby ○ Stop

Power preset 34 %

□ Enable Script Control

Status

Power 1.7 W

Current 14 A

Voltage 211 V

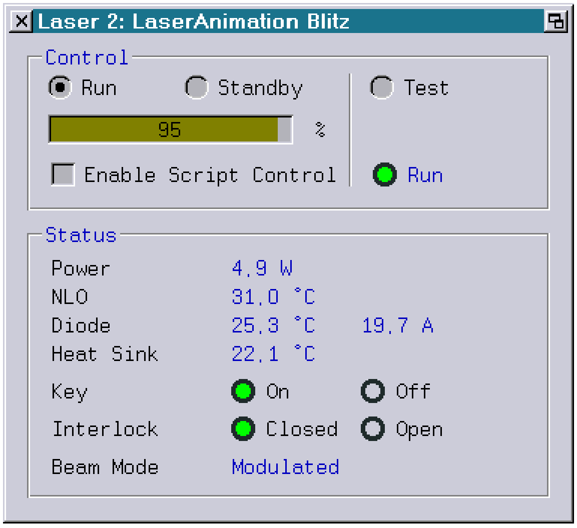
Tube ● Ok ○ Fill ○ Defect

Water ● Ok ○ Hot ○ Low

Cover ● Closed ○ Open

Key ● On ○ Off

LaserAnimation Blitz:



Laser 2: LaserAnimation Blitz

Control

● Run ○ Standby ○ Test

95 %

□ Enable Script Control ● Run

Status

Power 4.9 W

NLO 31.0 °C

Diode 25.3 °C 19.7 A

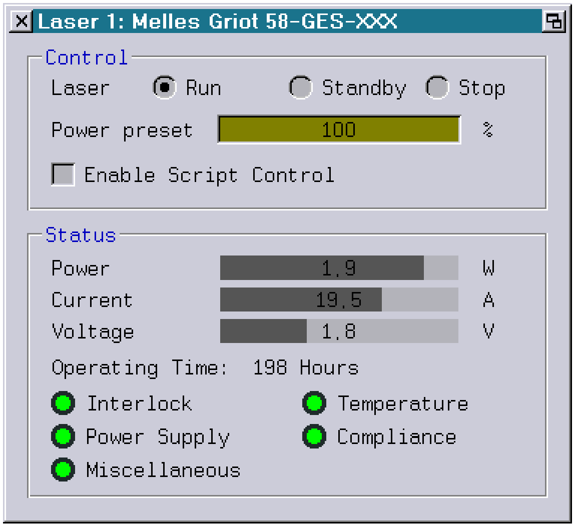
Heat Sink 22.1 °C

Key ● On ○ Off

Interlock ● Closed ○ Open

Beam Mode Modulated

Melles Griot 58-GES-XXX:



Laser 1: Melles Griot 58-GES-XXX

Control

Laser ● Run ○ Standby ○ Stop

Power preset 100 %

□ Enable Script Control

Status

Power 1.9 W

Current 19.5 A

Voltage 1.8 V

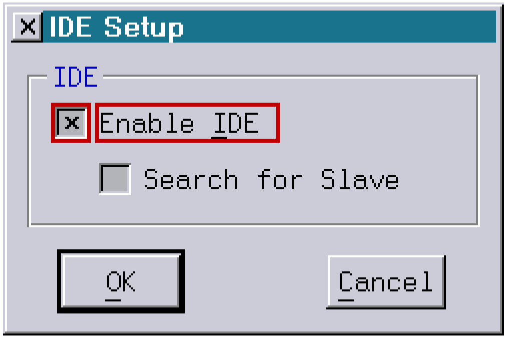
Operating Time: 198 Hours

● Interlock ● Temperature

● Power Supply ● Compliance

● Miscellaneous

IDE Setup (Compact)



IDE Setup

IDE

☒ Enable IDE

☐ Search for Slave

IDE

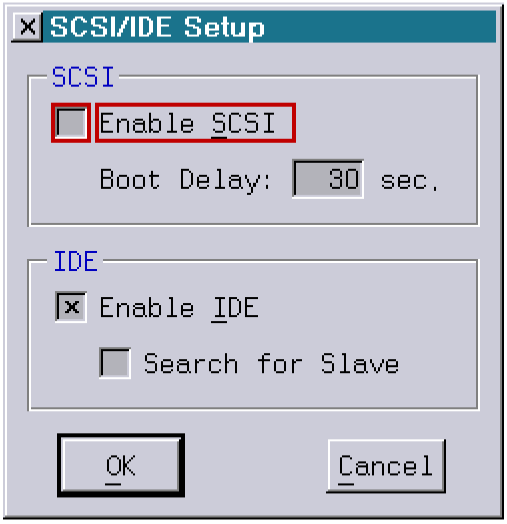
Enable IDE

This Checkbox determines, whether the IDE controller is to be used by the Lasergraph DSP or not. If there is no hard disk or CF-Card connected to the IDE controller, this checkbox can be deactivated.

Search for Slave

Usually only one IDE hard disk or CF-Card is installed in the Lasergraph DSP. It is used as Master. The search for a not installed Slave takes some time. By deactivating this checkbox the IDE controller is told not to wait for an answer from the Slave, which accelerates the booting process.
The setting "Search for Slave" is only effective when the IDE controller is enabled.

SCSI/IDE Setup (Workstation)



SCSI/IDE Setup

SCSI

☐ Enable SCSI

Boot Delay: 30 sec.

IDE

☒ Enable IDE

☐ Search for Slave

SCSI

Note: SCSI is only available on older DSP workstation units.

Enable SCSI

This checkbox determines if the SCSI bus will be used by the Lasergraph DSP or not. If the internal hard disk of the Lasergraph DSP is connected to the IDE controller and if there are no external SCSI devices connected, the SCSI bus can be turned off completely. This will considerably accelerate the booting process of the Lasergraph DSP.

Boot Delay

When the system is started it will wait for the period of time stated at this point before accessing the SCSI bus for the first time. This value has to be determined experimentally; if it is too small there is a danger that slower devices (such as CD-ROMs) will not be detected. If, on the other hand, the value is too high the booting process will take an unnecessarily long time. Possible values are between one and 60 seconds. If the SCSI bus is completely turned off (Checkbox "Enable SCSI" deactivated), the Boot Delay will be ignored.

IDE

Enable IDE

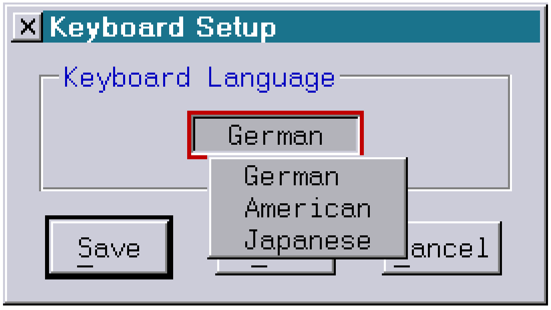
This Checkbox determines, whether the IDE controller is to be used by the Lasergraph DSP or not. If there is no hard disk or CF-Card connected to the IDE controller, this checkbox can be deactivated.

Search for Slave

Usually only one IDE hard disk or CF-Card is installed in the Lasergraph DSP. It is used as Master. The search for a not installed Slave takes some time. By deactivating this checkbox the IDE controller is told not to wait for an answer from the Slave, which accelerates the booting process.
The setting "Search for Slave" is only effective when the IDE controller is enabled.

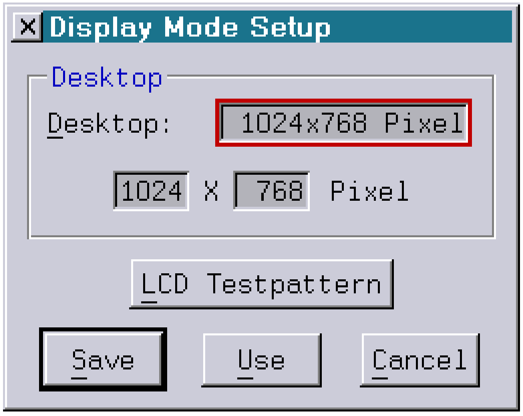
Depending on the existing hardware it may be possible that either the SCSI-, the IDE- or both controllers do not exist.

Keyboard Setup



In the popup menu, select the layout of the keyboard connected to your Lasergraph DSP.
This setting has shows no effect if the Lasergraph DSP is controlled by LGRemote.

Display Mode Setup



Desktop: 1024x768 Pixel

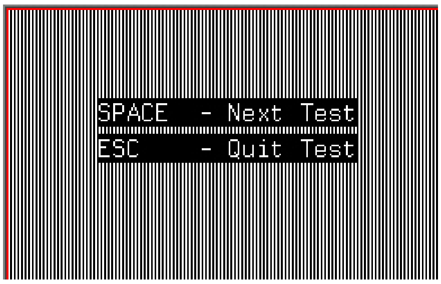
1024 X 768 Pixel

Change the size of the Desktop by choosing one of the default Desktop settings, or define a resolution by setting a specific value (or use the MousePoti Mode).

“Use” will keep the setting until the next restart.
“Save” will keep the setting as the default.

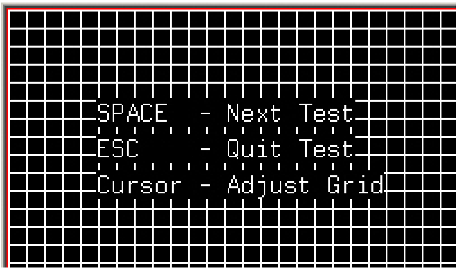
LCD Testpattern:

The button LCD Testpattern displays test pictures for the fine tuning of LCD monitors. There are two screens that can be toggled with the space bar. Both have a thin red border for the adjustment
of width and height. The first test pattern looks like this:



```
SPACE  - Next Test
ESC    - Quit Test
```

Change the width of the picture using the controls on your LCD display until the picture becomes clear. Then hit the space bar to get the second pattern:



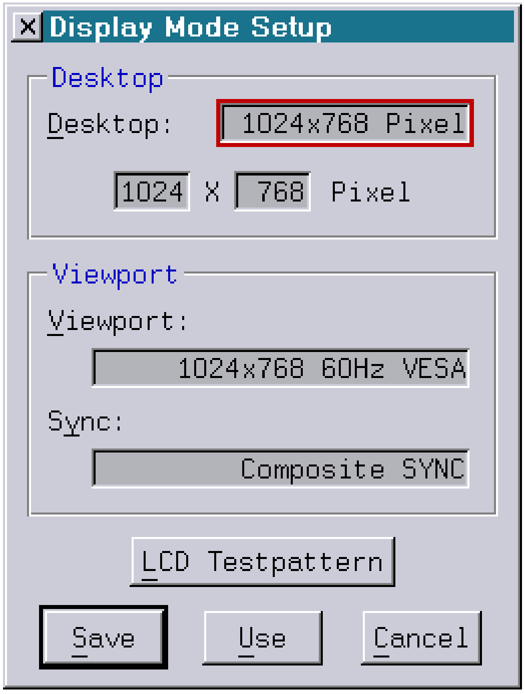
```
SPACE  - Next Test
ESC    - Quit Test
Cursor - Adjust Grid
```

Fine tune your LCD display until the picture is clean black and white.

Lasergraph DSP Workstation:

Lasergraph DSP workstation controllers are no longer equipped with graphic boards.

For older Lasergraph DSP workstation with graphic board support the following adjustments can be made in the Display Mode Setup:



Desktop: 1024x768 Pixel

1024 X 768 Pixel

Viewport: 1024x768 60Hz VESA

Sync: Composite SYNC

The selection list for the "Viewport" contains entries for the different resolutions with the corresponding refresh rates.
If you select the same value in both fields, your screen just changes size. If you want to, you can set the desktop size bigger than the resolution. That way, you have the lower resolution (which is better readable under some conditions) plus the bigger desktop available. Your desktop then has the selected size, and the screen works like a window: move the cursor to the sides of the screen and the window moves.

The size of the desktop can also be set to a custom value (width between 640 and 2048, height between 480 and 2048 pixels).

A note on network operation: Depending on the display mode settings of the different machines there can be effects sometimes which look strange but are harmless: If the desktop of the local machine is set bigger than the desktop of the remote machine, the remote desktop gets displayed in the upper left part of the screen. The area which is usually not visible on the other machine (because it is beyond the desktop edges) looks sort of strange, cluttered with graphic elements, because it is used internally to cache screen elements for fast operation. This is normal and causes no problems, just looks weird. If the local desktop is set smaller than the desktop of the remote machine, only the visible area can be accessed. None of these oddities appear as long as local and remote machines are set to the same desktop values.

When a different resolution is selected, an automatic adjustment of the frequencies is only possible with new video cards. However, the automatic frequency selection only works if the rotary switch "FREQ" on the video card is set to '8'. All other legal settings ('6', '7' and '9') can still be selected on new video cards. They are especially useful/necessary if a certain frequency is supposed to be forced.

With the video cards used up to now a message is displayed asking the user to manually adjust the rotary switch "FREQ" on the video card. The following are the complete data for resolution, refresh rates and the corresponding switch settings for older video cards:

Resolution Refresh Rate Setting "FREQ"

720 x 512 60 Hz ->7
720 x 512 75 Hz ->9
800 x 600 60 Hz ->9
800 x 600 100 Hz ->7
800 x 600 120 Hz ->9
1024 x 768 60 Hz ->7
1024 x 768 75 Hz ->9
640 x 480 VGA 60 Hz ->6
800 x 600 VESA 60 Hz ->9
800 x 600 VESA 72 Hz ->6
1024 x 768 VESA 60 Hz ->7

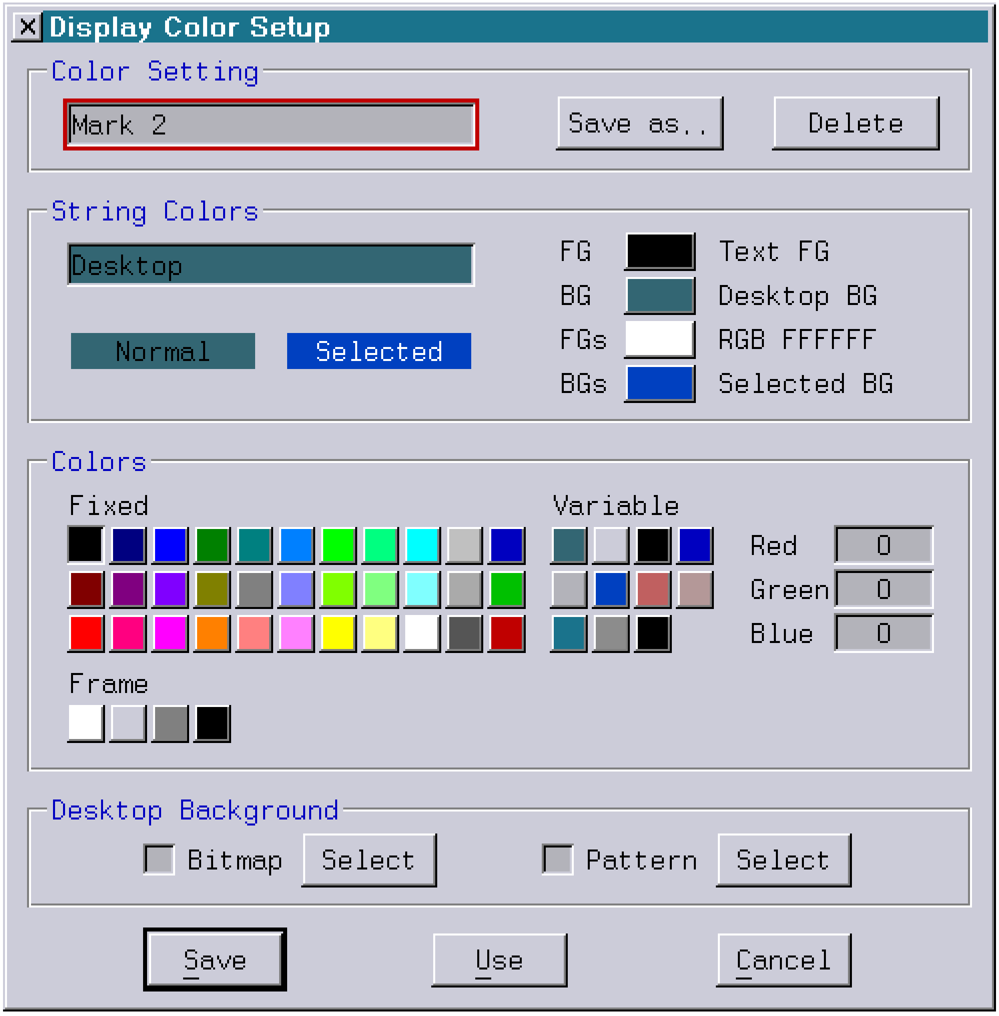
After changing the resolution, you have to confirm the setting within 10 seconds, or the previous mode will be restored.

If you confirmed the setting by hitting return, but your monitor stays black - for this case the following key combinations are available:

‹control›-‹numlock›-7 switch on 720 x 512
‹control›-‹numlock›-8 switch on 800 x 600
‹control›-‹numlock›-9 switch on 1024 x 768 VESA

Note: The key combinations only work after the Lasergraph DSP has completed its booting process.
The key combinations change the resolution only temporarily, which means that after changing the resolution it is still necessary to the resolution by clicking "Save". Otherwise the same problem will occur after the next start.

Display Color Setup



Color Setting: Mark 2

String Colors: Desktop

FG	■	Text FG
BG	■	Desktop BG
FGs	□	RGB FFFFFF
BGs	■	Selected BG

Red	0
Green	0
Blue	0

The window is accessed by entering **ColorSetup** <return> or by clicking the "Display Color" button in the Setup Window.

Color Setting

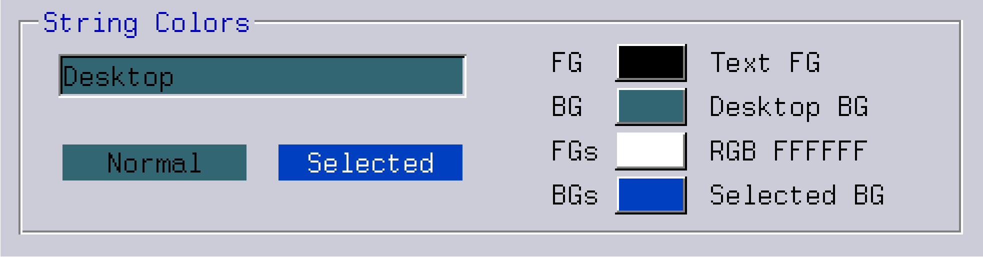
On top, the name of the setting is shown. If you click on it, a popup-menu opens, showing all saved settings plus the default settings "Carbon", "Ocean", "Mark 2" and "Classic".

With the first change of anything in the "Color Setup" window, the name disappears. To save your changes, click the *Save as...*-button. Name your new setting, click *OK*, and the setting is saved to disk. The name of the color setting has to be a legal file name.

To delete a color setting, select it in the popup-menu, then click the *Delete*-button.

Your saved settings are stored in the directory "C:\System\Configuration\ColorSettings" with the extension *.DCfg. You can copy them to disk and load them to another Lasergraph DSP to work with your preferred setting on different machines.

String Colors



String Colors

Desktop

Normal Selected

FG Text FG
BG Desktop BG
FGs RGB FFFFFF
BGs Selected BG

In this part of the setup window the coloring for different elements of the screen can be adjusted. Click onto the name of the currently selected element to get a popup-menu to select another element. In the menu, the elements are listed showing their current settings. Below the name the appearance of the element in normal and selected state is shown.
On the right, there are 4 buttons. They have an abbreviation on their left and a color-code on their right side.

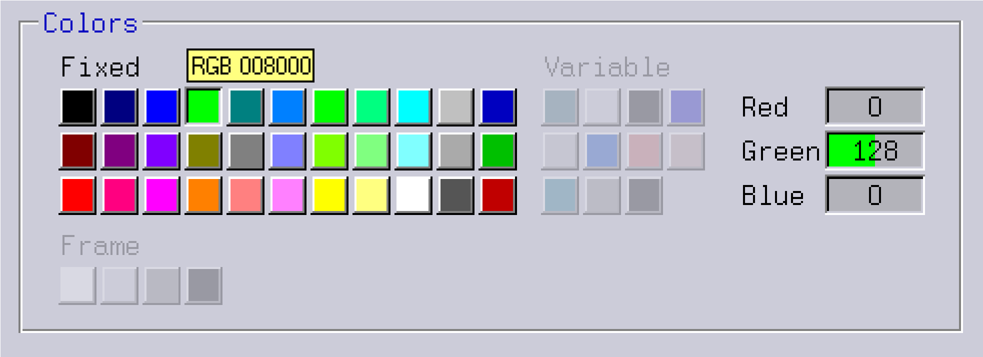
The abbreviations stand for

FG	Foreground (typically the text color of the unselected string)
BG	Background
FGs	Foreground selected (typically the text color of the selected string)
BGs	Background selected

The color-codes show either a number (i.e. "RGB FFFFFF" for white) or a name (i.e. "Desktop BG").

Colors

In the lower part of the window there is the color picker.
It shows 33 fixed colors, 11 variable colors and 4 frame colors.



Colors

Fixed RGB 008000

Variable

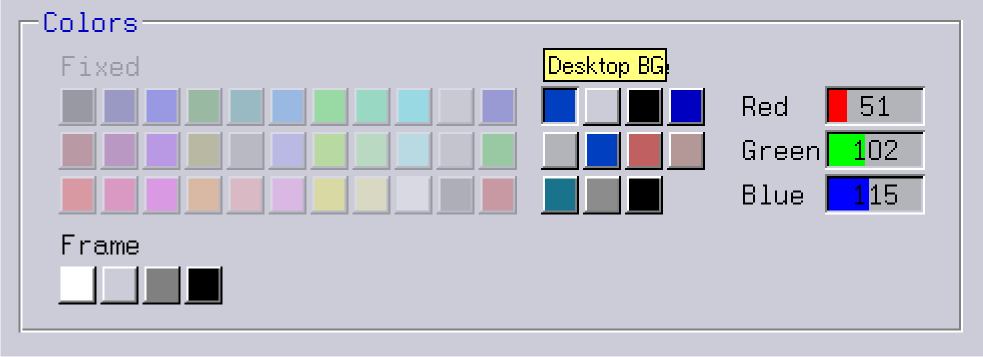
Red 0
Green 128
Blue 0

Frame

By moving the mouse over one of the 33 fixed colors on the left its RGB numbers appears, by clicking on one of them its value for Red, Green and Blue appears on the right side.

The values cannot be changed.

When you move the mouse over one of the 11 variable colors or the 4 frame colors the corresponding object name is shown.



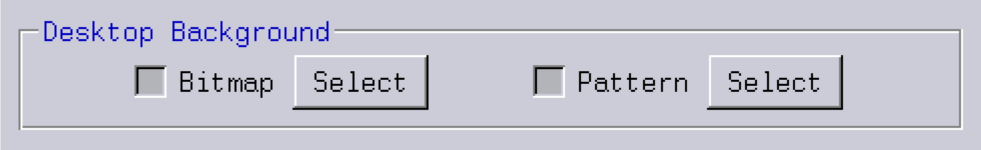
Red 51
Green 102
Blue 115

The selected color can be changed by clicking one of the three RGB-values and keeping the mouse clicked and moving it around. You can also enter a value with the keyboard.

To change the color of a screen element, select the element in the popup-menu, click the button of the part you want to change (FG, BG, FGs or BGs), and then (the cursor changed to the eyedropper symbol) click the color you want, either one of the fixed or one of the variables.

If the default color of the button is a variable (on the right there is a variable name, not a RGB number), then you should consider changing the variable's RGB-values rather than using a new color. Groups of similar elements have the same variable defined by default. If, for example, you want to change the highlighting color (BGs) from blue to another color, you have the choice of changing the variable color "Selected BG", and thereby changing all blue backgrounds at once, or clicking your way through all 34 elements of the popup-menu, finding the 20 elements with the blue background, and changing every single one by hand.

Desktop Background



You can provide your Lasergraph DSP desktop with a suitable background picture.
Two possibilities are available:

- ***Bitmap*** centers a background picture on the desktop
- ***Pattern*** repeats a background picture as often as it fits on the desktop

You can select a background picture for you desktop using *Select* .

Screen Saver Setup

Four different screen savers can be selected in the Screen Saver Setup window:

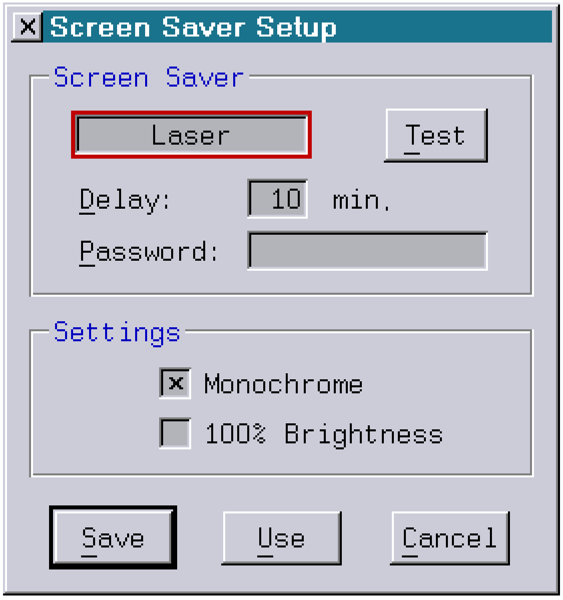
<None>
DarkScreen
Laser
Bezier

If no keystroke or mouse movement was registered for the chosen delay time, the selected screen saver is activated; for a preview, click the *Test* button.

The **DarkScreen** command can be entered at any time to dim the screen right away regardless of the screen saver selected. In this case no password is required even if password protection is enabled.

The screen saver selected in the setup window can be activated at any time by entering the command **StartScreenSaver**.

To restrict access to the Lasergraph DSP while you're away, password protection can be activated.



Screen Saver Setup

Screen Saver

Laser

Delay: 10 min.

Password:

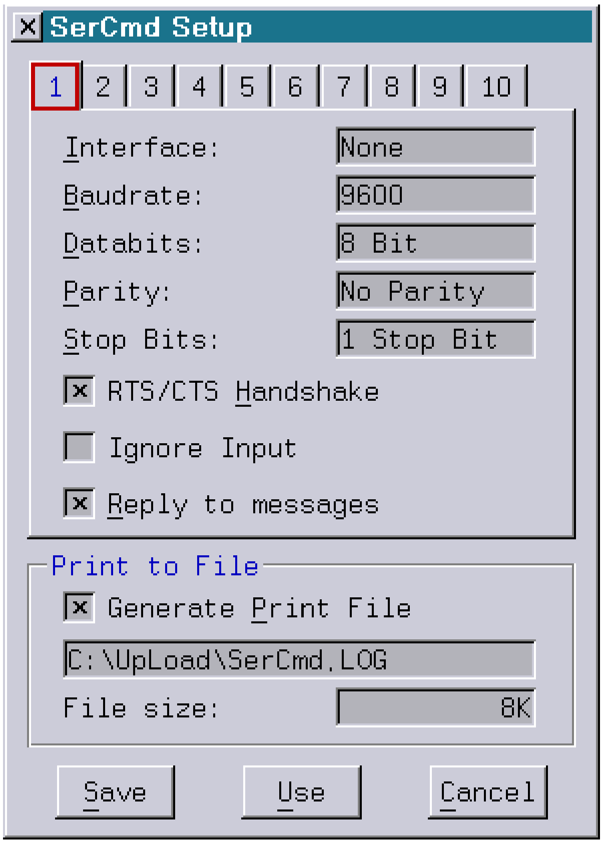
Settings

☒ Monochrome

□ 100% Brightness

A password can be set here to protect your Lasergraph DSP. Once the screen saver is activated, it can not be terminated until the correct password is entered. Clicking the *Test* button in the setup window or entering the **DarkScreen** command with the keyboard does not activate password protection. If the Lasergraph DSP is turned off and on again, the password protection will not be activated until the screen saver is started. For complete protection also define a boot password using the **PassWord.Install** command. Password protection is disabled if no password is set.

SerCmd Setup



Interface: None

Baudrate: 9600

Databits: 8 Bit

Parity: No Parity

Stop Bits: 1 Stop Bit

☒ RTS/CTS Handshake

□ Ignore Input

☒ Reply to messages

Print to File

☒ Generate Print File

C:\UpLoad\SerCmd.LOG

File size: 8K

With the SerCmd (Serial Commands) the Lasergraph DSP offers the possibility to control up to 10 externally connected devices via RS232 interface or to be controlled by external devices.

For more details see "Chapter SerCmd".

Settings for each of the 10 SerCmd interfaces:

Interface

The interface to which the device is connected is selected with the popup "Interface":

Lasergraph DSP *compact*

None	SerCmd is deactivated.
RS232-1 to RS232-3	The device is connected to one of the three local RS232 ports.
NetCom1 to NetCom16	The device is connected to one of the 16 NetCom ports.

Lasergraph DP *workstation*

None	SerCmd is deactivated.
Modem	The device is connected to the "Modem" port.
Tablet	The device is connected to the "Tablet" port. Note that the ring indicator (pin 9) must not be connected because it is used as power supply +12V for the digitizer tablet.
NetCom1 to NetCom16	The device is connected to one of the 16 NetCom ports.

Baudrate
One of the following baud rates can be selected:
9600, 19200, 38400, 57600, 76800, 115200, 153600, 230400 or 250000 baud.

Databits
7 or 8 Bits can be selected.

Parity
No Parity, Even Parity or Odd Parity can be selected.

Stop Bits
1 or 2 Stop Bits can be selected.

RTS/CTS Handshake
Select "RTS/CTS Handshake" if hardware handshake shall be used.

Make sure that all settings (Baudrate, Databits, Parity, Stop Bits and RTS/CTS Handshake) match the settings of the connected device.

Ignore Input
If "Ignore Input" is checked the Lasergraph DSP will ignore all incoming commands from this interface.

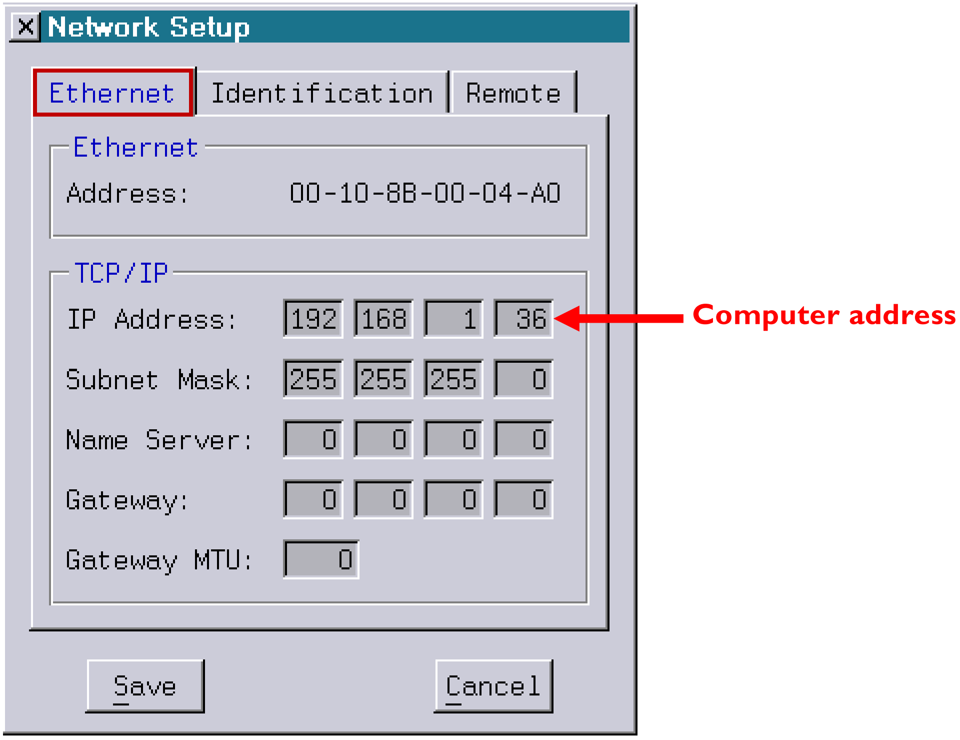
Reply to messages
As long as "Reply to messages" is checked the Lasergraph DSP will respond to incoming commands and queries.

Settings which apply to all SerCmd interfaces:

Generate Print File
If "Generate Print File" is checked all text received with the command "<print" will be appended to the specified text file.

File size
If desired the file size of the Print File can be limited.
One of the following can be selected: 8K, 16K or unlimited.

Network Setup



Ethernet

Address: 00-10-8B-00-04-A0

TCP/IP

IP Address: 192 168 1 36

Subnet Mask: 255 255 255 0

Name Server: 0 0 0 0

Gateway: 0 0 0 0

Gateway MTU: 0

Computer address

Ethernet

The Ethernet Address (MAC) is unique for every Lasergraph DSP. No other device worldwide has the same number. It can not be changed.

TCP/IP

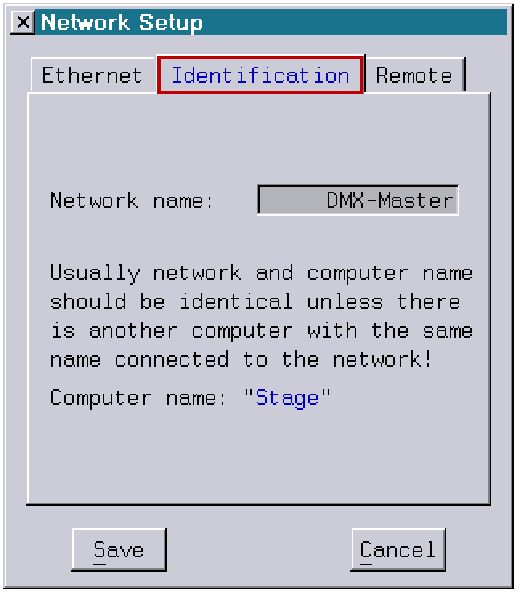
The TCP/IP settings are needed to identify the machine in a network. If you want to connect your Lasergraph DSP to an existing network or to the internet, please ask your network administrator or your internet service provider for the values needed.
If you want to start your own local network by, let's say, connecting several Lasergraph DSPs, you can use the settings as shown in the picture above. These IP addresses are available for use in local networks:

10.1.1.1	to	10.254.254.254
172.16.1.1	to	172.31.254.254
192.168.1.1	to	192.168.254.254

The last value of the IP address is the computer number (in the example above the computer number is 36). Every computer must have a different number between 1 and 255.

Identification

Enter a name for this Lasergraph DSP. All machines in a network should have their own names so that they can be safely distinguished. This name can be freely chosen.



Network name: DMX-Master

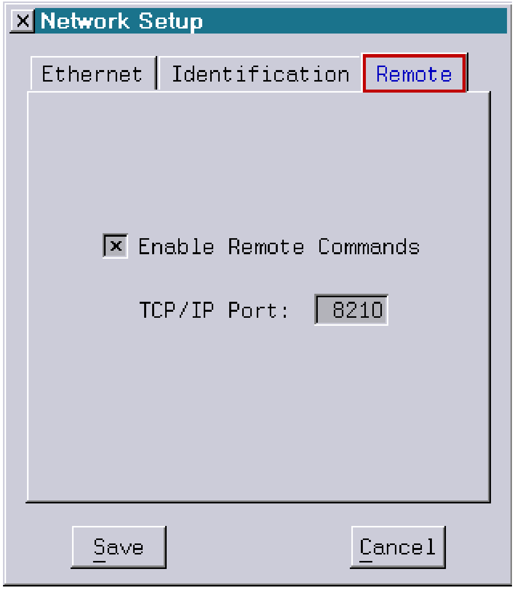
Usually network and computer name should be identical unless there is another computer with the same name connected to the network!

Computer name: "Stage"

Remote

It is possible to send commands to the DSP over a TCP/IP port which can be freely chosen. The commands have to by concluded with “CR/LF”. They will be put into the command line of the DSP and will be executed as if they had been entered locally.

In order to enable this functionality select the tab "Remote":



☒ Enable Remote Commands

TCP/IP Port: 8210

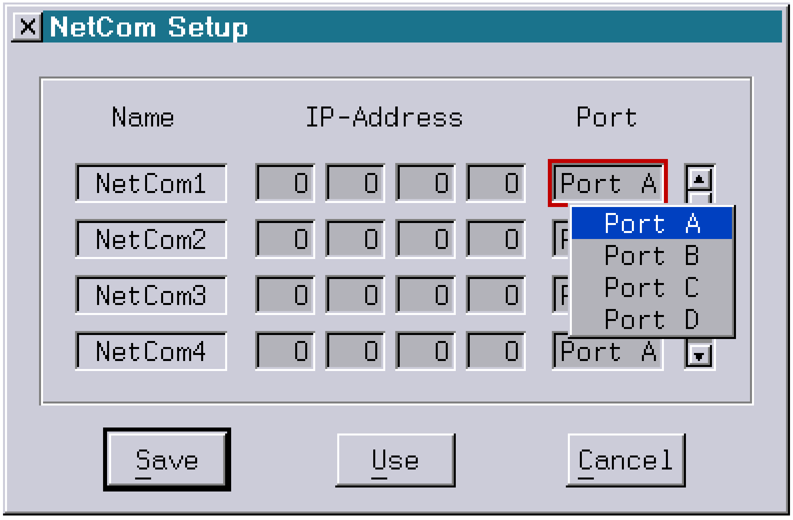
Examples for the use of this feature are control of the DSP through another DSP (using `SerCmd` and `SENDstring`), a media server or even over mobile phones.

NetCom Setup

NetCom Boxes are separate peripheral devices which have up to four RS232 interfaces each. NetCom Boxes are connected to the Ethernet. They allow a Lasergraph DSP the use of further devices connected to their RS232 interfaces.

Up to 16 NetCom Box interfaces can be assigned. They are used like the built-in RS232 interfaces and can be used to connect a modem or for controlling laser or other devices.

For assignment choose "NetCom" from the Setup window.



NetCom Setup

Name	IP-Address				Port
NetCom1	0	0	0	0	Port A
NetCom2	0	0	0	0	P
NetCom3	0	0	0	0	P
NetCom4	0	0	0	0	Port A

To assign a NetCom Box interface to one of the logical Lasergraph DSP NetCom interfaces enter the IP address of the device and select the desired serial interface of the device (port A to D) from the popup menu.

To deactivate a NetCom interface enter the IP address 0.0.0.0.

After altering the NetCom Setup you have three options:

Save	This setting is standard from now on.
Use	This setting is used until the Lasergraph DSP is turned off or restarted.
Cancel	All changes are discarded and the previous settings are restored.

I/O Setup

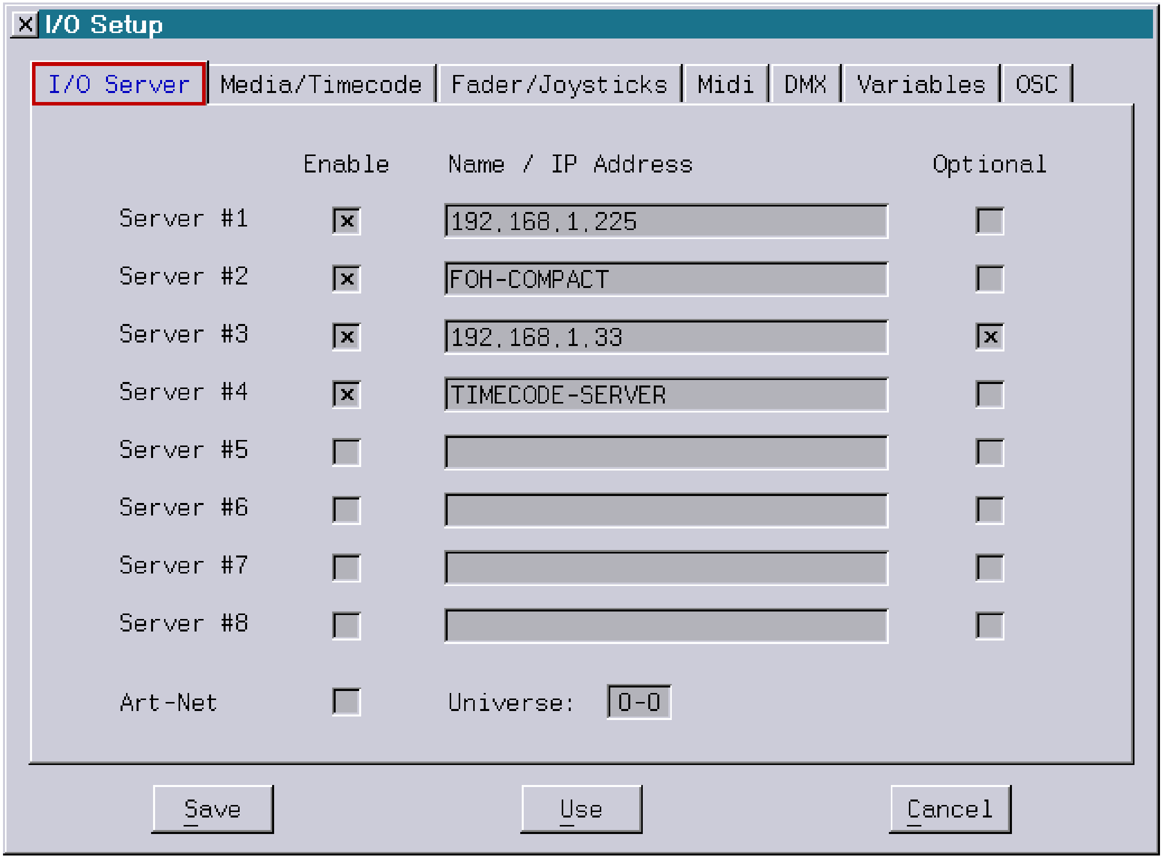
This network option allows the use of different resources, e. g. Smpte, Midi, Faderboard on several Lasergraph DSP and Media from the LGServer.
That means several computers react to the Midi keyboard which is connected to one computer. The computer with the connected Midi keyboard is called **Server** in the network. A server makes its resources available for other computers in the network.
You can also use certain resources of a PC running Windows XP/Vista/7/8/8.1/10, but there are some restrictions, because a PC does not have all the inputs a Lasergraph DSP have. E. g. the Midi-Input is only available if a MIDI-Interface is installed in the PC.

Other computers using the resources of a server are called **Clients**. Here it is necessary to define which resources from which servers shall be used. You can configure this in the I/O Setup.

A Lasergraph DSP can be a server and also a client at the same time. E. g. it is possible to use the Smpte-Timecode of a Lasergraph DSP for other computers in the network. At the same time this Lasergraph DSP can use the Joysticks connected to another computer.

I/O Server

Server #1 to #8

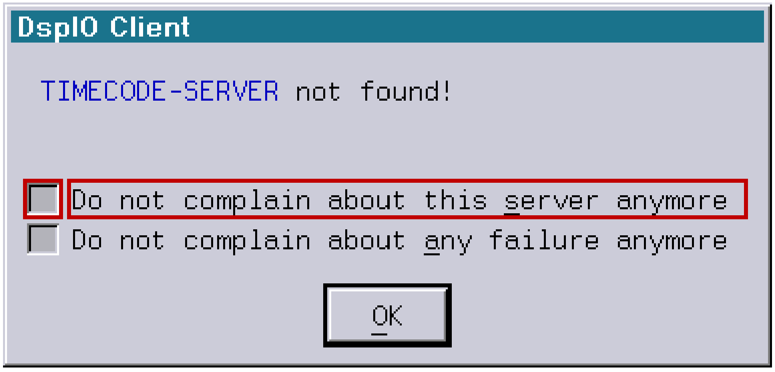


	Enable	Name / IP Address	Optional
Server #1	☒	192.168.1.225	□
Server #2	☒	FOH-COMPACT	□
Server #3	☒	192.168.1.33	☒
Server #4	☒	TIMECODE-SERVER	□
Server #5	□		□
Server #6	□		□
Server #7	□		□
Server #8	□		□
Art-Net	□	Universe: 0-0	

Here you can define the I/O server (Lasergraph DSPs or PCs), which have resources this Lasergraph DSP want to access. Up to eight servers can be entered.

First activate the checkbox "Enable" and fill in the name or the IP address of the corresponding computer into field "Name / IP Address".

If Optional is not active and a server does not answer than you get the following warning:



TIMECODE-SERVER not found!

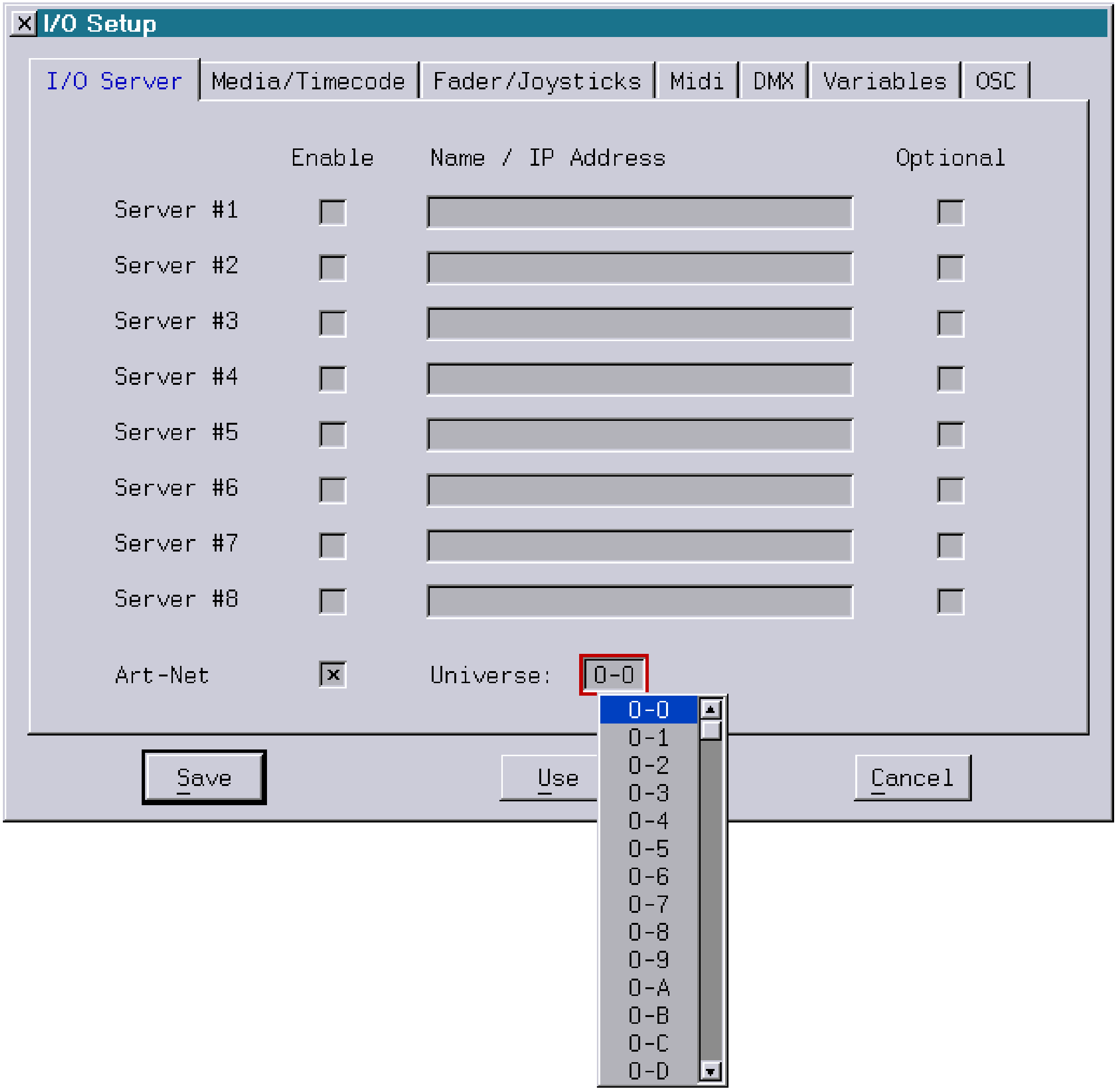
□ Do not complain about this server anymore
□ Do not complain about any failure anymore

You can decide if additional failures of this server or of all servers should be ignored. By activating the checkbox “Optional” this warning never appears. This is useful if one or more server are not always accessable / available.

Art-Net

“Art-Net” is an Ethernet protocol based on TCP/IP for transmitting DMX512.
It was developed by Artistic Licence (UK) Ltd. and it can be used exactly like any other DMX512 source to control the Mark 2.

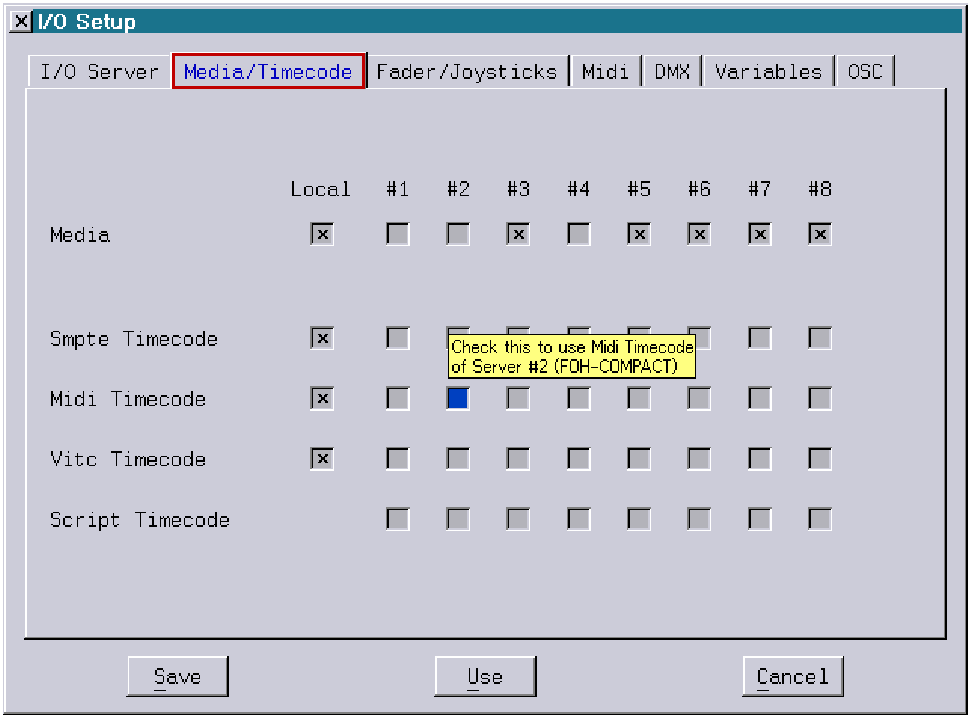
In order to activate the Art-Net client on the Mark2 the checkbox "Enable" has to be activated under "I/O Server" in the I/O Setup and the desired Art-Net universe has to be selected.



	Enable	Name / IP Address	Optional
Server #1	□		□
Server #2	□		□
Server #3	□		□
Server #4	□		□
Server #5	□		□
Server #6	□		□
Server #7	□		□
Server #8	□		□
Art-Net	☒	Universe: 0-0	

Now you can define the I/O-devices and the corresponding servers.
Assign the device or several devices to a corresponding server with the checkboxes.
“Local” always means the use of the device connected to the local Lasergraph DSP.
In addition there are eight device sources “#1” to “#8”, correspond to the 8 servers, available.

Media/Timecode



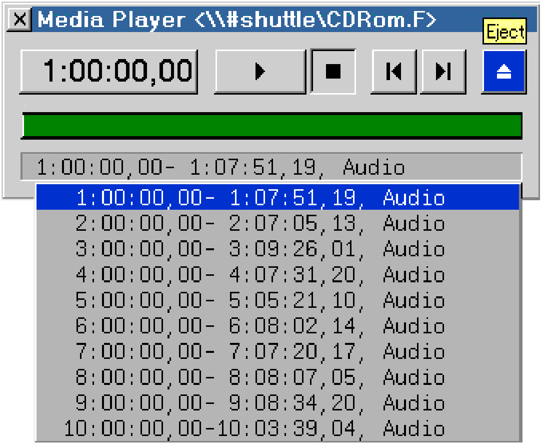
	Local	#1	#2	#3	#4	#5	#6	#7	#8
Media	☒	□	□	☒	□	☒	☒	☒	☒
Smpte Timecode	☒	□	□	□	□	□	□	□	□
Midi Timecode	☒	□	■	□	□	□	□	□	□
Vitc Timecode	☒	□	□	□	□	□	□	□	□
Script Timecode		□	□	□	□	□	□	□	□

Check this to use Midi Timecode of Server #2 (FOH-COMPACT)

Media

Local CD-ROM at local Lasergraph DSP
#1 to #8 CD-ROM of another Lasergraph DSP or PC running LGServer (Wav/Avi)

If several Media servers are assigned, the Media servers will be used one after another.
If you insert an Audio-CD in a CD-ROM drive of a PC or a Lasergraph DSP the Media Player appears with the network path:



```
 1:00:00,00- 1:07:51,19, Audio
 1:00:00,00- 1:07:51,19, Audio
 2:00:00,00- 2:07:05,13, Audio
 3:00:00,00- 3:09:26,01, Audio
 4:00:00,00- 4:07:31,20, Audio
 5:00:00,00- 5:05:21,10, Audio
 6:00:00,00- 6:08:02,14, Audio
 7:00:00,00- 7:07:20,17, Audio
 8:00:00,00- 8:08:07,05, Audio
 9:00:00,00- 9:08:34,20, Audio
10:00:00,00-10:03:39,04, Audio
```

That means, the CD-ROM drive (CDRom.F) of computer "shuttle" is active.

There is a difference between the access to a CD-ROM drive and the access to other drives in a network. That’s why the sign '#' was inserted into the path.
If a CD contains data tracks you can access them in the File mode by entering the path:

```
\\#shuttle\CDRom.F
```

In the File System Client the CD-ROM drive can be assigned to a drive letter, e.g. G:, compare the description in the File System Server Setup.

Smpte Timecode

Local	Smpte-input of local Lasergraph DSP
#1 to #8	Smpte-inputs of other Lasergraph DSP

Midi Timecode

Local	Midi In connector of local Lasergraph DSP
#1 to #8	Midi In connectors of other Lasergraph DSP or PC running LGServer

Vitc Timecode

Local	Video input of the local DSP
#1 to #8	Video input of other Lasergraph DSP

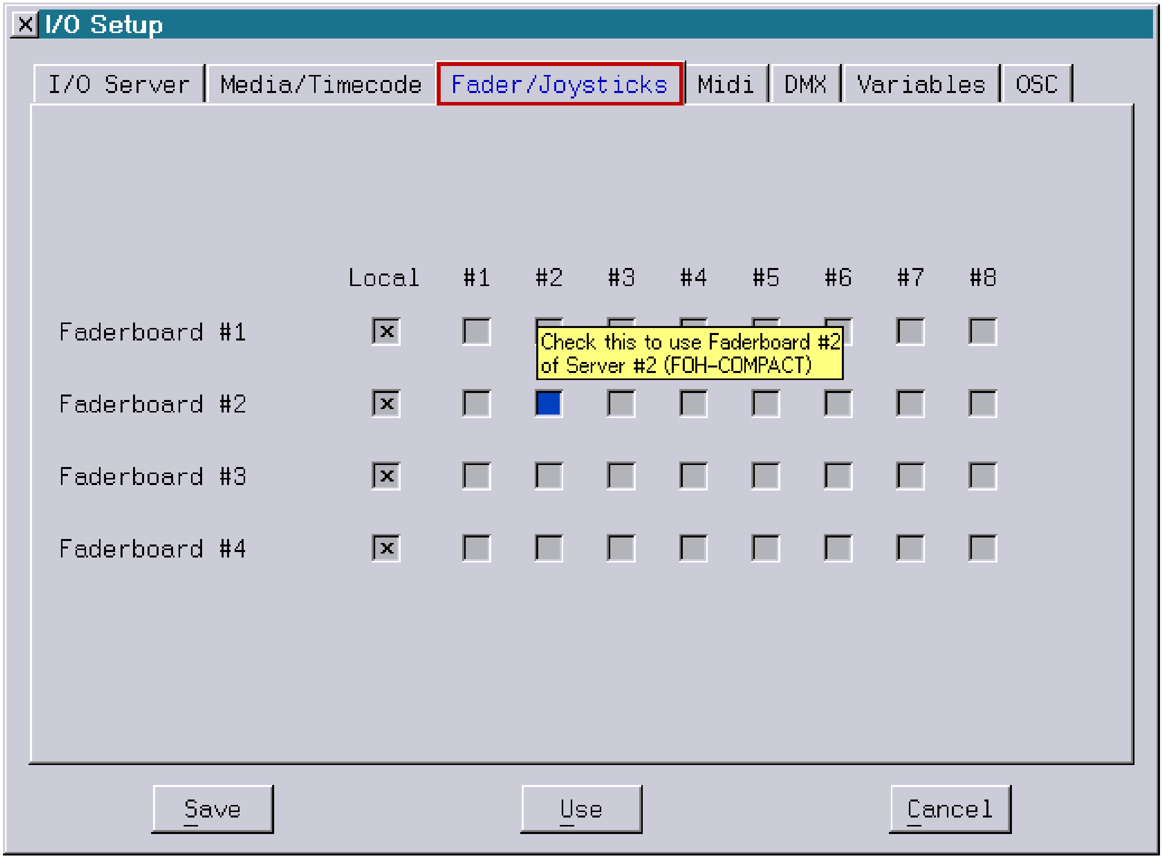
Script Timecode

Only the servers #1 to #8 can be used here.
If there are several Smpte-, Midi- and Script Timecode connected, the first recognized Timecode received will be accepted.

Hotkeys:

<l>:	Selects / deselects all local Media and Timecode sources
<a> / <r>:	Selects / deselects all Media and Timecode sources supplied by server #1 to #8.

Fader/Joysticks



	Local	#1	#2	#3	#4	#5	#6	#7	#8
Faderboard #1	☒	☐	☐	☐	☐	☐	☐	☐	☐
Faderboard #2	☒	☐	■	☐	☐	☐	☐	☐	☐
Faderboard #3	☒	☐	☐	☐	☐	☐	☐	☐	☐
Faderboard #4	☒	☐	☐	☐	☐	☐	☐	☐	☐

Check this to use Faderboard #2 of Server #2 (FOH-COMPACT)

Faderboard units can be assigned as follows:

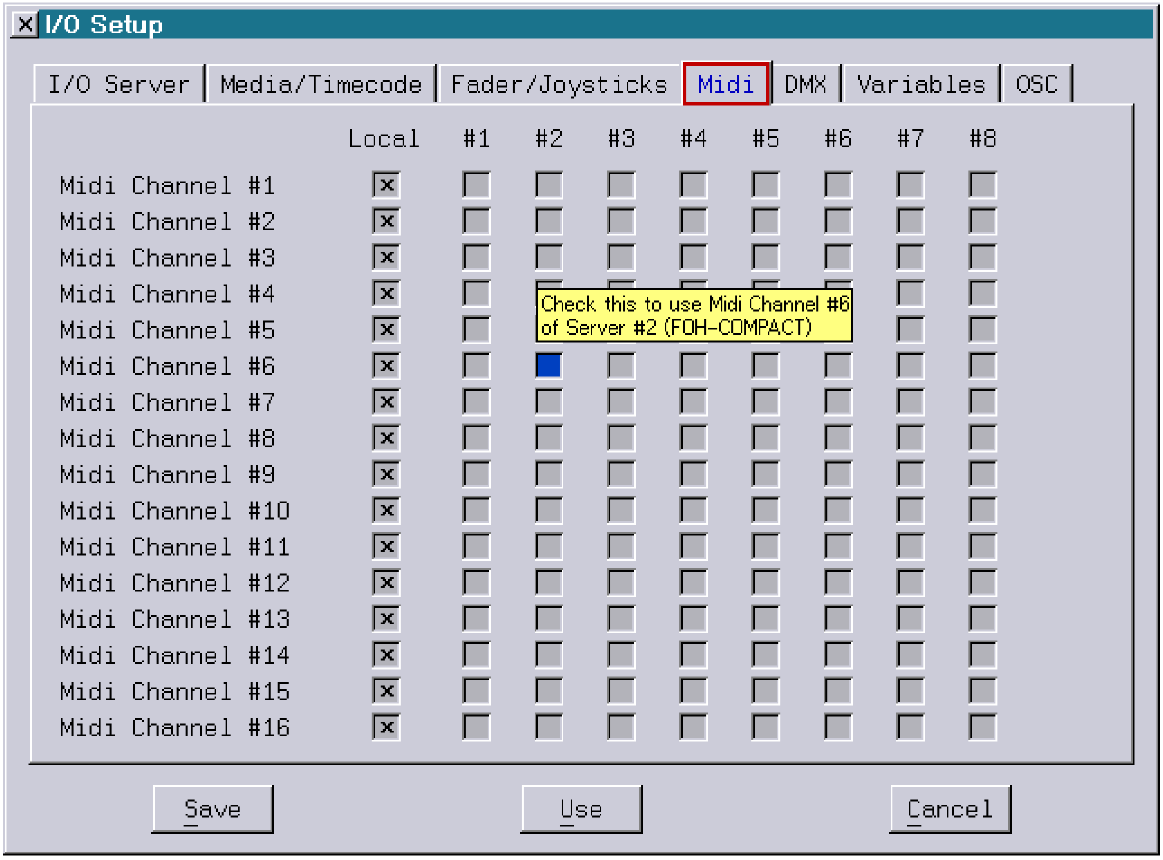
Local Faderboard(s) at local DSP will be used.

#1 to #8 there are three possibilities:

1. Faderboard(s) at other Lasergraph DSP will be used.
2. Faderboard(s) at PC will be used. You need a special adapter for the connection of a faderboard to a PC. You can get this adapter from your distributor. **Do not connect a faderboard to a PC without this adapter! This could break the PC as well as the faderboard.**
3. Joysticks connected directly to a PC (without Faderboard) can be used.

Hotkeys:
<l>: Selects / deselects all locally connected Faderboards
<a> / <r>: Selects / deselects all Faderboards connected to server #1 to #8.

MIDI



	Local	#1	#2	#3	#4	#5	#6	#7	#8
Midi Channel #1	☒	□	□	□	□	□	□	□	□
Midi Channel #2	☒	□	□	□	□	□	□	□	□
Midi Channel #3	☒	□	□	□	□	□	□	□	□
Midi Channel #4	☒	□	□	□	□	□	□	□	□
Midi Channel #5	☒	□						□	□
Midi Channel #6	☒	□	■	□	□	□	□	□	□
Midi Channel #7	☒	□	□	□	□	□	□	□	□
Midi Channel #8	☒	□	□	□	□	□	□	□	□
Midi Channel #9	☒	□	□	□	□	□	□	□	□
Midi Channel #10	☒	□	□	□	□	□	□	□	□
Midi Channel #11	☒	□	□	□	□	□	□	□	□
Midi Channel #12	☒	□	□	□	□	□	□	□	□
Midi Channel #13	☒	□	□	□	□	□	□	□	□
Midi Channel #14	☒	□	□	□	□	□	□	□	□
Midi Channel #15	☒	□	□	□	□	□	□	□	□
Midi Channel #16	☒	□	□	□	□	□	□	□	□

Check this to use Midi Channel #6 of Server #2 (FOH-COMPACT)

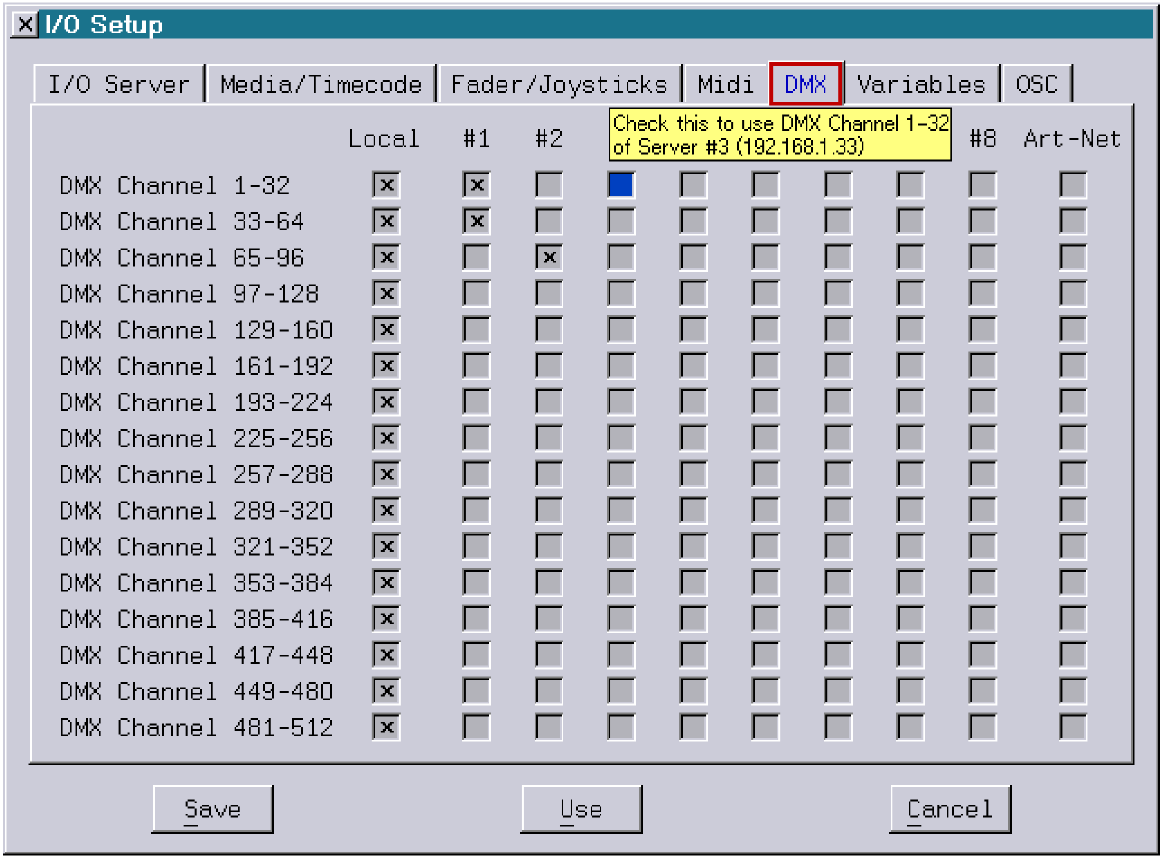
The Midi Channels #1 to #16 can be assigned to the specific servers:

Local	Midi input at local Lasergraph DSP will be used
#1 to #8	Midi input at other Lasergraph DSP or PC running LGServer will be used

Hotkeys:

<l>:	Selects / deselects all Midi channels of the local Midi input.
<a> / <r>:	Selects / deselects all Midi channels of the Midi input of server #1 to #8.

DMX



Check this to use DMX Channel 1-32 of Server #3 (192.168.1.33)

	Local	#1	#2						#8	Art-Net
DMX Channel 1-32	☒	☒	□	■	□	□	□	□	□	□
DMX Channel 33-64	☒	☒	□	□	□	□	□	□	□	□
DMX Channel 65-96	☒	□	☒	□	□	□	□	□	□	□
DMX Channel 97-128	☒	□	□	□	□	□	□	□	□	□
DMX Channel 129-160	☒	□	□	□	□	□	□	□	□	□
DMX Channel 161-192	☒	□	□	□	□	□	□	□	□	□
DMX Channel 193-224	☒	□	□	□	□	□	□	□	□	□
DMX Channel 225-256	☒	□	□	□	□	□	□	□	□	□
DMX Channel 257-288	☒	□	□	□	□	□	□	□	□	□
DMX Channel 289-320	☒	□	□	□	□	□	□	□	□	□
DMX Channel 321-352	☒	□	□	□	□	□	□	□	□	□
DMX Channel 353-384	☒	□	□	□	□	□	□	□	□	□
DMX Channel 385-416	☒	□	□	□	□	□	□	□	□	□
DMX Channel 417-448	☒	□	□	□	□	□	□	□	□	□
DMX Channel 449-480	☒	□	□	□	□	□	□	□	□	□
DMX Channel 481-512	☒	□	□	□	□	□	□	□	□	□

The single DMX channels (1 to 512) can be assigned to the specific servers.

Local	DMX input at local Lasergraph DSP will be used
#1 bis #8	DMX input at other Lasergraph DSP will be used
Art-Net	DMX channels which use Art-Net can be selected

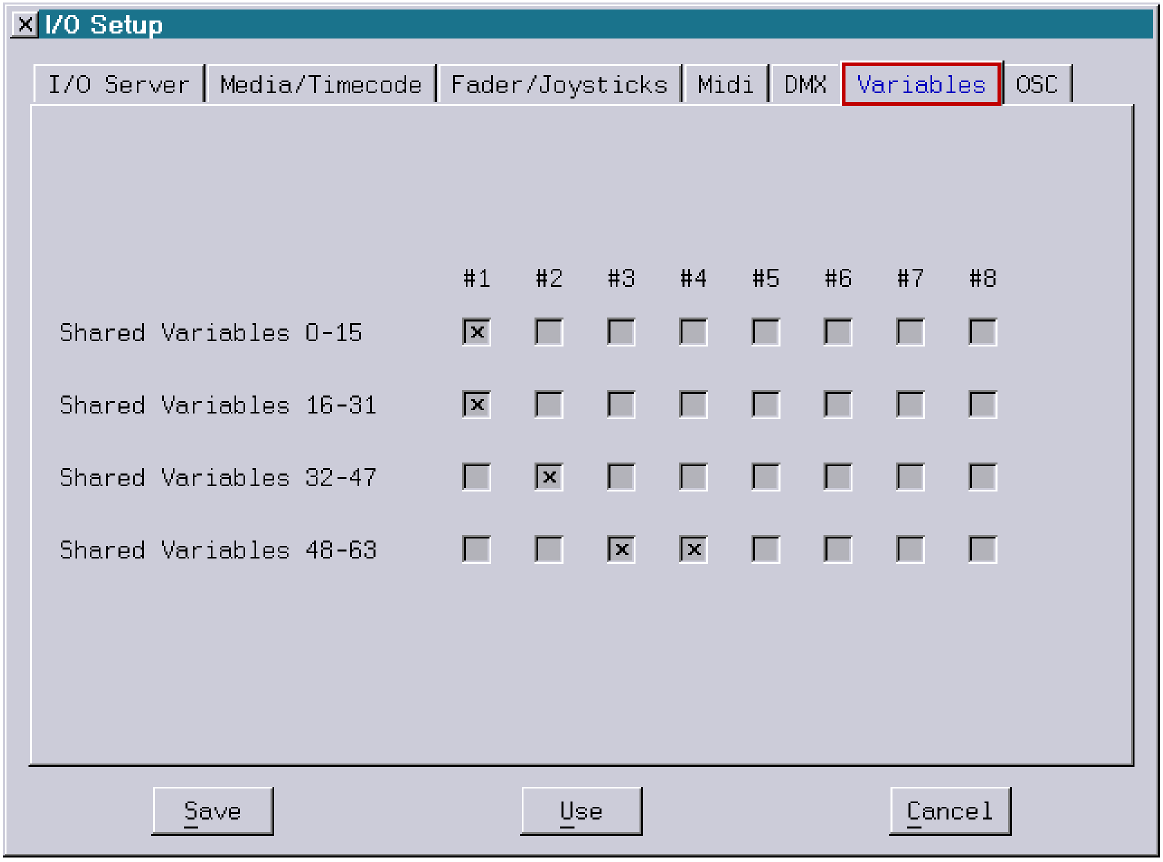
Hotkeys:

<l>:	Selects / deselects all DMX channels of the local DMX input.
<a> / <r>:	Selects / deselects all DMX channels of the DMX input of server #1 to #8 and Art-Net.

Variables

Shared Variables are special variables which can be used for data exchange between different DSPs within a Lasergraph DSP workstation or a Lasergraph DSP compact dual.

They can also be used to exchange data between different Lasergraph DSPs:



	#1	#2	#3	#4	#5	#6	#7	#8
Shared Variables 0-15	☒	□	□	□	□	□	□	□
Shared Variables 16-31	☒	□	□	□	□	□	□	□
Shared Variables 32-47	□	☒	□	□	□	□	□	□
Shared Variables 48-63	□	□	☒	☒	□	□	□	□

The 64 available Shared Variables are divided into 4 groups with 16 Shared Variables each.

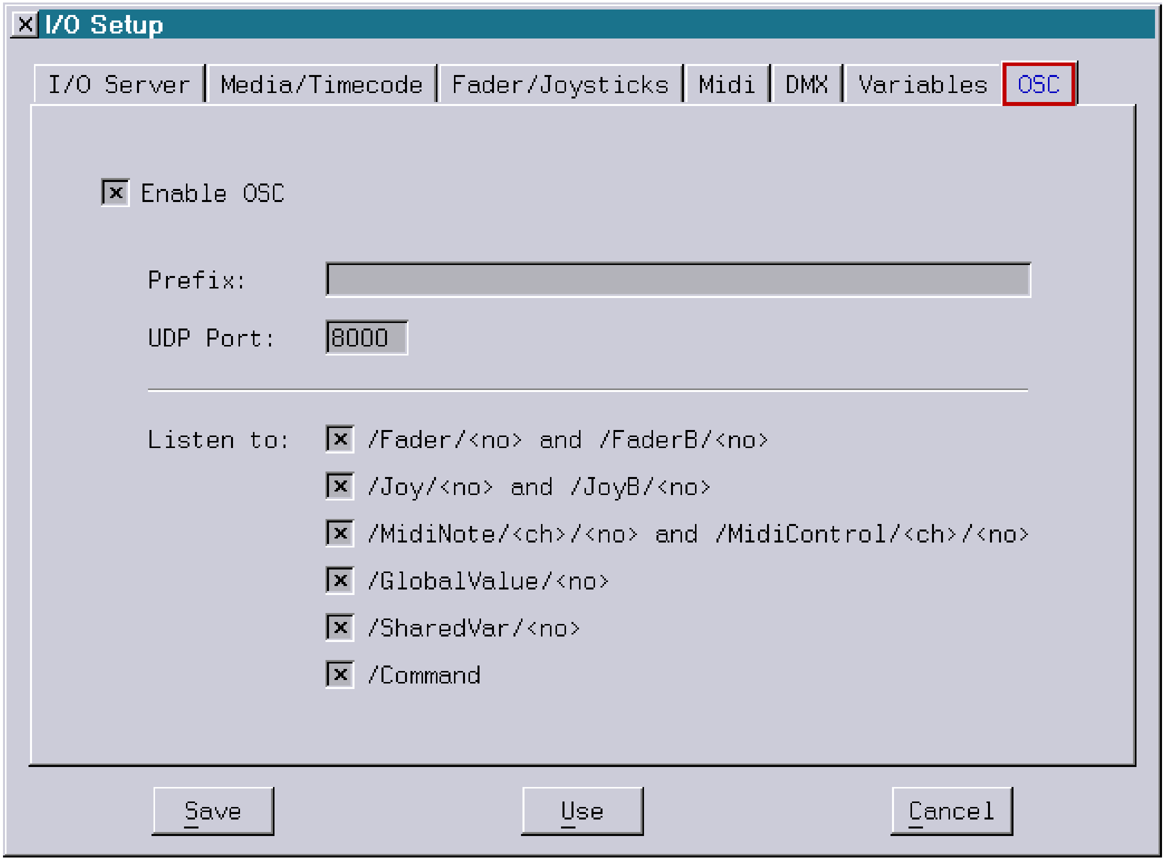
For each of these 4 groups it can be defined independently from which servers they should receive value changes.

Note:

On a Lasergraph DSP workstation changes to the Shared Variables can only be received. Sending, on the other hand, is not possible because the CPU card is too slow for this.

OSC

Open Sound Control is an open, transport-independent, message-based protocol developed for communication among computers, sound synthesizers, and other multimedia devices.



☒ Enable OSC

Prefix:

UDP Port: 8000

Listen to:
☒ /Fader/<no> and /FaderB/<no>
☒ /Joy/<no> and /JoyB/<no>
☒ /MidiNote/<ch>/<no> and /MidiControl/<ch>/<no>
☒ /GlobalValue/<no>
☒ /SharedVar/<no>
☒ /Command

As soon as "Enable OSC" is activated the Mark 2 will receive incoming OSC messages.

In addition to setting the port number used for receiving the UDP packets, here it is also possible to specify the prefix of the address which has to be sent to address this Mark 2.

The complete address is comprised of this freely definable prefix and the key words implemented in the Mark 2.

The different key words can be activated or deactivated in blocks.

The following key words are supported:

/Fader/<no>

sets one or more faders to the specified values.

<no> specifies the number of the first fader which is to be set (0-63). In case more than one value is sent the following values will be used to set the following faders.

value range:	0.0 to 1.0 (float32) resp. 0 to 65535 (int32)

/FaderB/<no>

turns one or more fader buttons on resp. off.

<no> specifies the number of the first fader button which is to be set (0-63). In case more than one value is sent the following values will be used to set the following fader buttons.

value range on:	not equal 0.0 (float32) resp. not equal 0 (int32)
value range off:	0.0 (float32) resp. 0 (int32)

/Joy/<no>

sets the X/Y position of the joystick <no> to the specified value (<no>: 0-7).

value range:	-1.0 to 1.0 (float32) resp. 0 to 255 (int32)

/JoyB/<no>

turns the two buttons of joystick <no> on resp. off (<no>: 0-7).

both buttons off:	0.0 (float32) resp. 0 (int32)
button 1 on, button 2 off:	1.0 (float32) resp. 1 (int32)
button 1 off, button 2 on:	2.0 (float32) resp. 2 (int32)
both buttons on:	3.0 (float32) resp. 3 (int32)

/MidiNote/<ch>/<no>

simulates the reception of a Midi note on resp. note off command.

<ch> is the Midi channel (1-16) and <no> is the number of the Midi key (0-127).

value range:	-1.0 to 1.0 (float32) resp. -127 to 127 (int32)

/MidiControl/<ch>/<no>

simulates the reception of a Midi control change command.

<ch> is the Midi channel (1-16) and <no> is the number of the controller (0-31).

value range:	0.0 to 1.0 (float32) resp. 0 to 127 (int32)

/GlobalValue/<no>

sets one or more Global Value to the specified values.

<no> specifies the number of the first Global Value which is to be set (0-1000000).
In case more than one value is sent the following values will be used
to set the following Global Values.

values:	float32 or int32

/SharedVar/<no>

sets one or more Shared Variables to the specified values.

<no> specifies the number of the first Shared Variable which is to be set (0-63).
In case more than one value is sent the following values will be used
to set the following Shared Variable.

values:	float32 or int32

/Command

Enters the command sent as OSC string into the command line and executes it.

Since not all devices support OSC strings the command can alternatively be sent as part of the address:

/Command/<command>

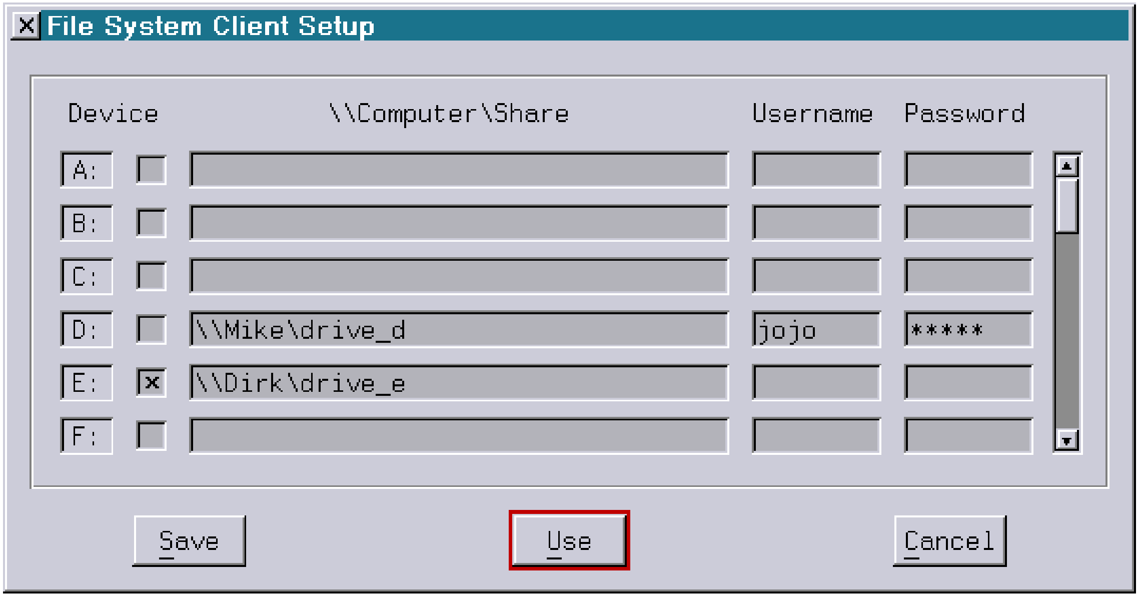
In this case the parameter float32 resp. int32 determines whether the command is to be executed or not. In case it is not equal 0.0 resp. 0 it will be executed, otherwise it will not.

Example:

To set the first fader on the Mark 2 with the defined prefix "/DSP1" to 0.5 the following OSC message has to be sent:

address:	/DSP1/Fader/0
parameter:	0.5 (float32) or 32768 (int32)

File System Client Setup



Device		\\Computer\Share	Username	Password
A:	□			
B:	□			
C:	□			
D:	□	\\Mike\drive_d	jojo	*****
E:	☒	\\Dirk\drive_e		
F:	□			

To make the files of a remote computer (PC running Windows XP/Vista/7/8/8.1/10 or other Lasergraph DSP) available they are assigned as logical devices on the Lasergraph DSP in the File System Client Setup.
The settings for the logical devices A: to Z: can be reached using the scrollbar on the right.

In the example above, the logical drive D: on the Lasergraph DSP is assigned to drive D: (drive_d) in Mike´s PC (named Mike) and the logical drive E: on the Lasergraph DSP is assigned to the drive E: (drive_e) in Dirk´s PC.
In the example, drive D: is not connected to the network device \\Mike\drive_d because the checkbox is unchecked.
Now, if you change to File mode on the Lasergraph DSP and select E:, the contents of drive E: named drive_e on the remote computer named Dirk are displayed as if it was mounted inside the Lasergraph DSP. If a local drive D: exists, it is available in File mode, otherwise no drive D: can be accessed.

Remote volumes can be drives or folders made available for network access on the remote computer. To access these volumes, you may need a username and a password. If you enter this information in the appropriate fields of the FS Client window, you can connect by checking the enable checkbox. The same information you have to enter in the appropriate fields of the FS Server Setup.
If the fields are left empty or username and password do not correspond a window shows up, where you have to type in username and password.

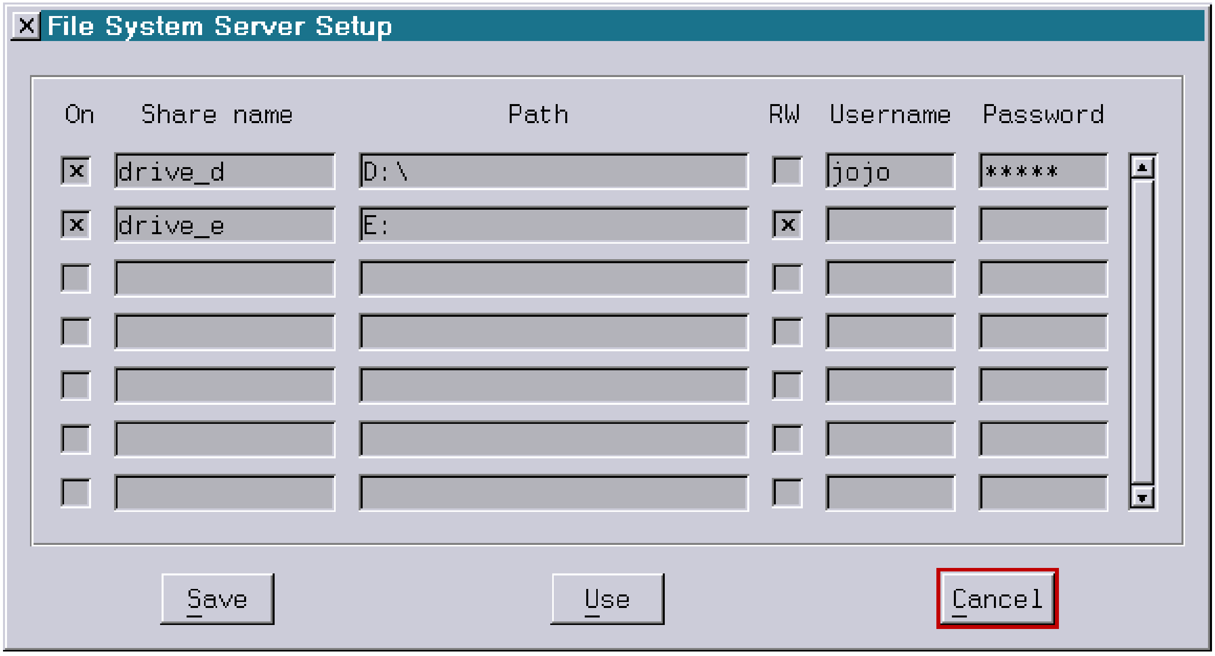
If the checkbox 'Store in FS Client Setup' is active, the information will be entered in the Setup and wont be asked next time.
An unauthorized person do not get an access if you finish a connection saved with a password like this:

- Change to a local drive in the File mode
- Open the FS Client Setup and check that the password is not entered
- Activate 'Use'

All connections will be interrupted and the password will be asked again.

To connect a volume temporarily you can also change to File mode and use the **subst** command. The syntax is **subst drive: \\computer\share**. The connection in the example above could also be established by entering **subst E: \\dirk\drive_d**. This setting would then be active until the Lasergraph DSP is turned off. To cancel the connection, just enter **subst drive:** without a network path; in our example, this would be **subst E:**.

File System Server Setup



File System Server Setup

On	Share name	Path	RW	Username	Password
☒	drive_d	D:\	□	jojo	*****
☒	drive_e	E:	☒		
□			□		
□			□		
□			□		
□			□		
□			□		

You can allow a network access of the Lasergraph DSP's local hard disk drives, directories or CD-ROM drives by another Lasergraph DSP in the File System Server Setup:

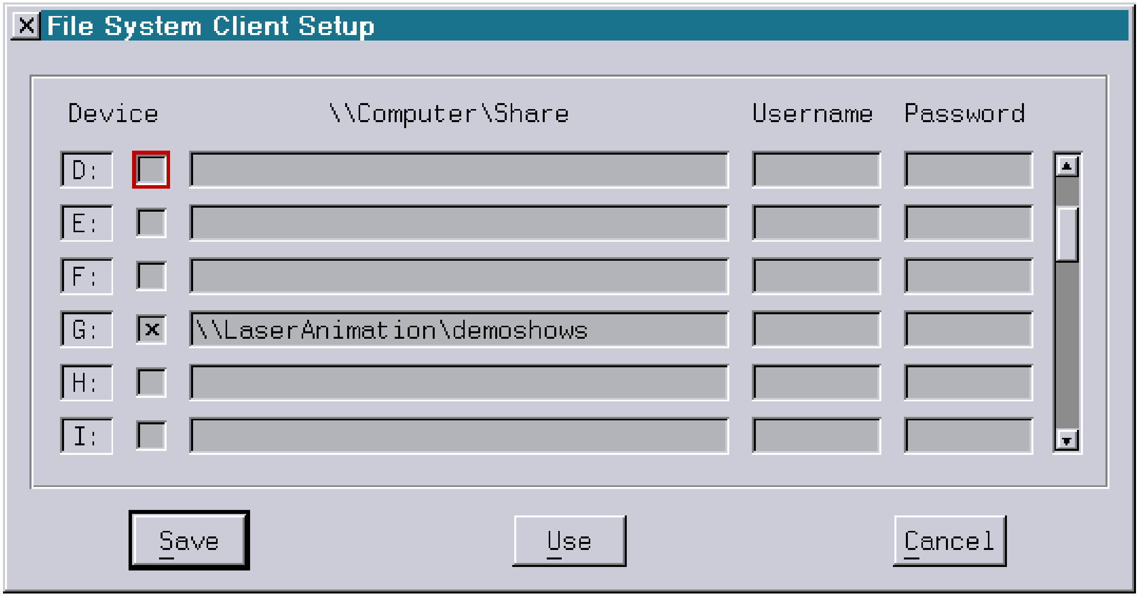
- Click to the left checkbox.
- Declare the **Share name** .
- Enter the drive or the directory in **Path**
- The checkbox **RW** (Read/Write) defines the access rights:
  active = read and write is possible, inactive = only read is possible.
- If you need a password fill in the fields **Username** and **Password**.

The screenshot above illustrates that the access to hard disk drive D:\ allows reading and to the hard disk drive E:\ reading and writing.

Examples

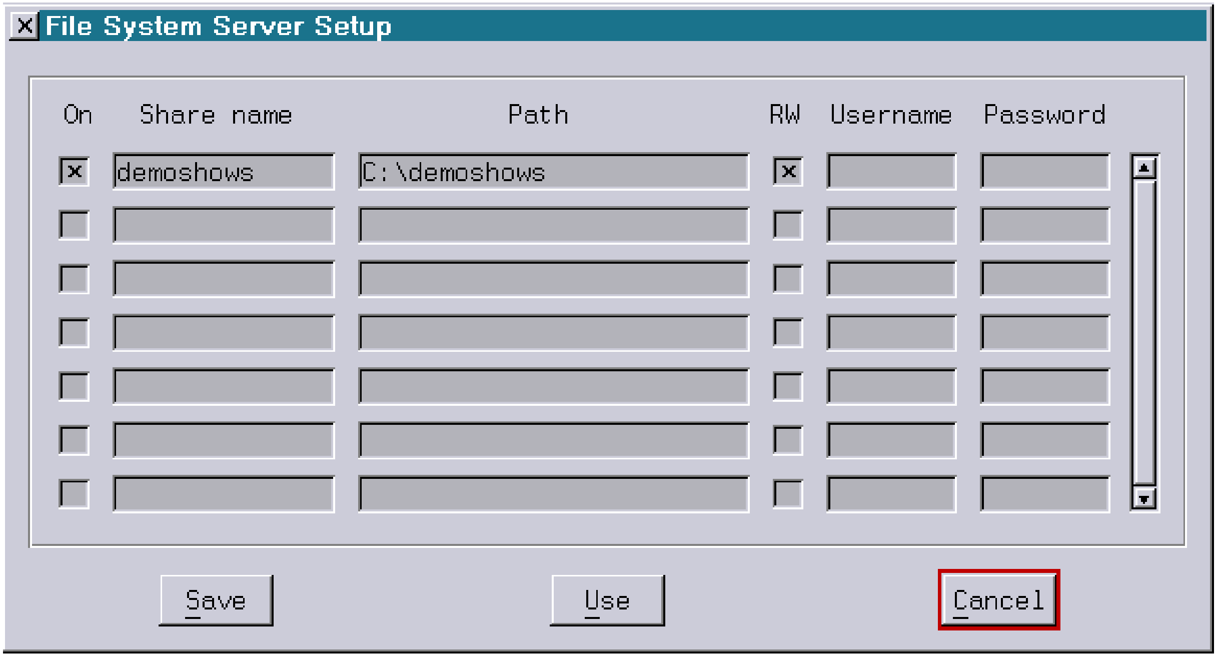
Hard disk drives or directories of several Lasergraph DSPs

Define the drive(s) or the directory(ies) of corresponding computer(s) which should be used as server(s) and assign them to a letter in the FS Client Setup of the computer which should be used as client. You may enter an 'Username' and a 'Password'. E. g. the Lasergraph DSP x (client) should access to the directory 'demoshows' of another Lasergraph DSP named 'LaserAnimation' (server). This directory will be assigned to hard disk drive G: of the Lasergraph DSP x in the File System Client Setup:



Device		\\Computer\Share	Username	Password
D:	□			
E:	□			
F:	□			
G:	☒	\\LaserAnimation\demoshows		
H:	□			
I:	□			

Now the FS Server Setup has to be selected at the Lasergraph DSP 'LaserAnimation' to declare this computer as server:

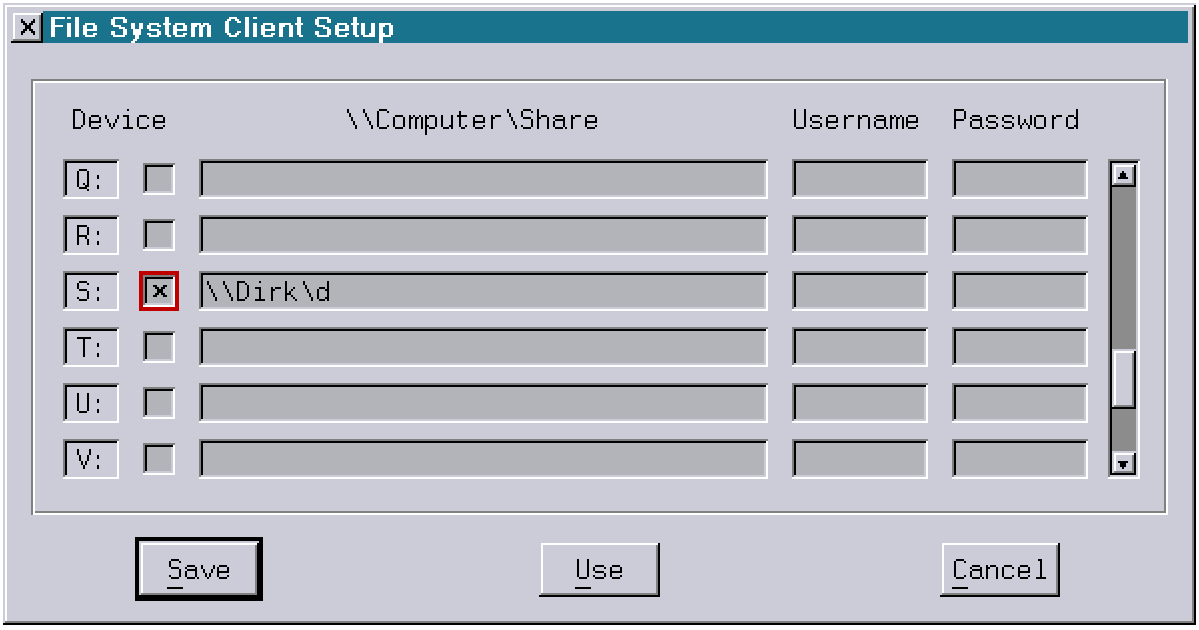


On	Share name	Path	RW	Username	Password
☒	demoshows	C:\demoshows	☒		
□			□		
□			□		
□			□		
□			□		
□			□		
□			□		

The directory 'demoshows' on the hard disk drive C:\ is available. Reading and writing are possible. If you now change to the drive G: on the Lasergraph DSP x the contents of the directory 'demoshows' will be displayed as if they actually existed on the computer.

Hard disk drives or directories between Lasergraph DSP and PC

A PC running Windows cannot use resources of Lasergraph DSPs, the Windows Explorer does not list the Lasergraph DSP drives. But it is possible to make hard disk drives and directories of a PC running Windows XP/Vista/7/8/8.1/10 available for a Lasergraph DSP. So the PC is the server and the Lasergraph DSP is the client. E. g. the Lasergraph DSP x should access the drive D:\ of a PC named 'Dirk'. For that purpose the FS Client Setup on the Lasergraph DSP x has to be selected and the drive D:\ of the PC has to be assigned to a logical drive of this DSP, e. g. 'S:', see picture:



Device		\\Computer\Share	Username	Password
Q:	☐			
R:	☐			
S:	☒	\\Dirk\d		
T:	☐			
U:	☐			
V:	☐			

Master/Slave Setup

Using the operating mode "Master/Slave" several Lasergraph DSPs can be comfortably combined to a show playback system. One Lasergraph DSP acts as Master, all others as Slaves.
The whole system is controlled using the Master, i.e. user actions such as loading, starting and stopping a show are executed on the Master.
The Slaves automatically load the correct show and synchronize their output with the Master.

Application Examples:

- A show which was created on a Lasergraph DSP workstation with several installed DSP cards can be played back on several Lasergraph DSP compact units without changes.
- A show requires more than four output channels.
- The distance between the individual projectors is so long that connecting the Lasergraph DSP and the projectors is impossible or would lead to a significant deterioration of the signal quality. In this case a Lasergraph DSP can be positioned directly with each projector and the Lasergraph DSPs can be connected via Ethernet.

Generally all shows can run in Master/Slave mode. In order to avoid undesirable side effects we recommend beginning the Timescript with the following four commands:

1) `Comment` with the name of the show (required by LGFD and jr.Live!)
2) `DefineMedia` to define the media file (in case LGServer is used)
3) `SetTimeMode` to select the external timecode source
4) `PlayMedia` to start the playback (if using the Media Player)

For accessing more than four films special "Sub-BOOTs" were introduced.
These are independent BOOTs which are embedded in the main BOOT.
In order to ensure that Sub-BOOTs are loaded correctly irrespective of the BOOT name the Sub-BOOTs must have the following names:

```
A.BOOT    B.BOOT    C.BOOT    ...    G.BOOT
```

In order to save a Sub-BOOT you have to enter the directory of the main BOOT in the File mode using the command

```
ChangeDir Name.BOOT
```

where `Name` is the name of the BOOT.

Here the command

```
CreateBoot A.BOOT
```

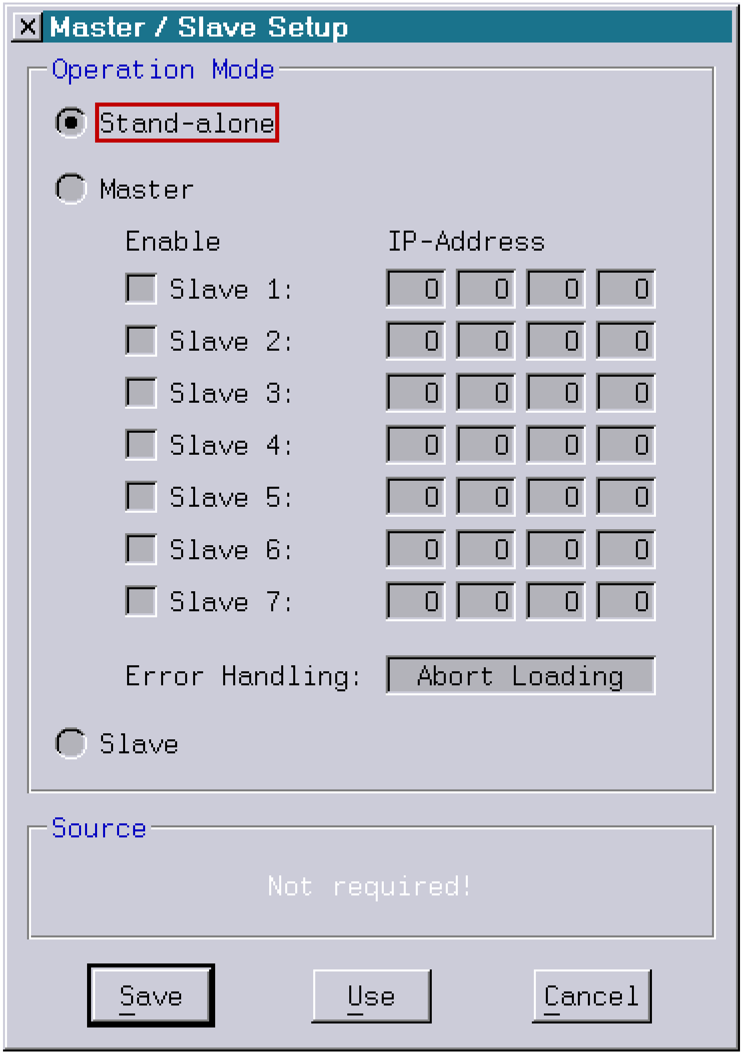
can for instance be used to save the Sub-BOOT "A".

Afterwards you have to leave the main BOOT again with

```
ChangeDir ..
```

Stand-alone

As long as "Stand-alone" is selected the Lasergraph DSP will act as before, i.e. it is neither Master nor Slave but works as an individual unit.



Master / Slave Setup

Operation Mode

● Stand-alone

○ Master

Enable	IP-Address
☐ Slave 1:	0 0 0 0
☐ Slave 2:	0 0 0 0
☐ Slave 3:	0 0 0 0
☐ Slave 4:	0 0 0 0
☐ Slave 5:	0 0 0 0
☐ Slave 6:	0 0 0 0
☐ Slave 7:	0 0 0 0

Error Handling: Abort Loading

○ Slave

Source

Not required!

Master

For setting up a Lasergraph DSP as "Master" the following steps are necessary:

1) Activate the checkbox "Master"

2) Select the connected Slaves (Enable + IP-Address)

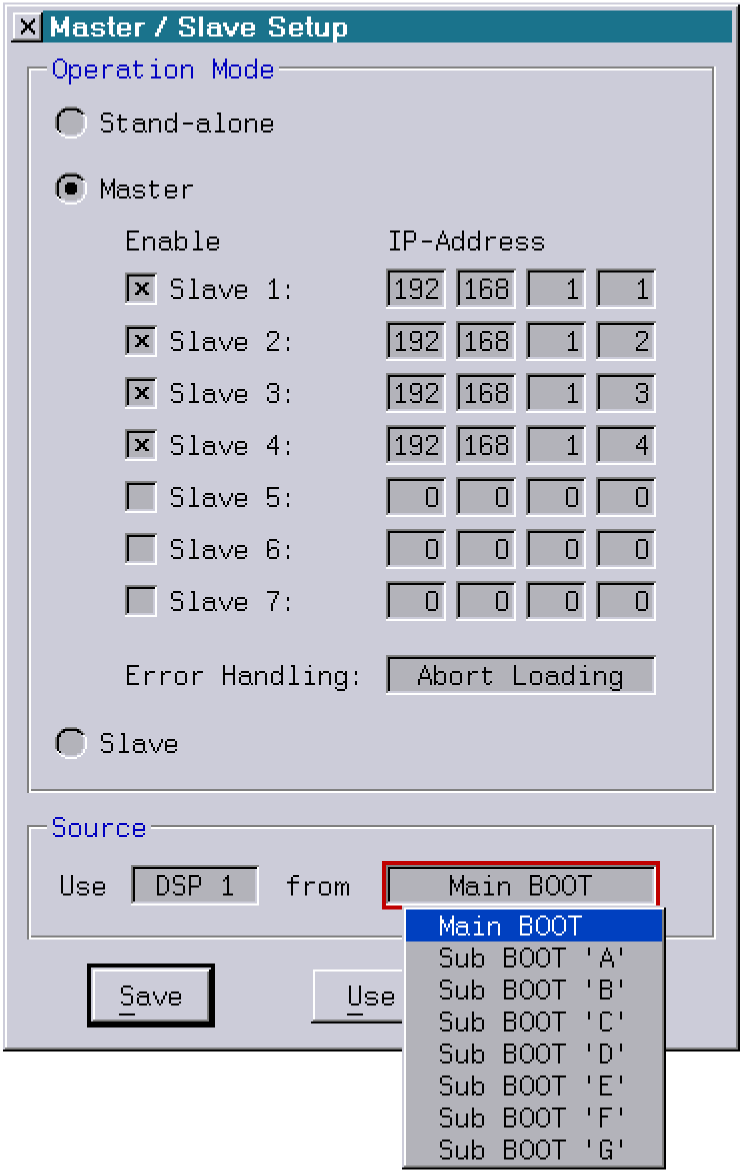
3) Select the behavior of the Master in case loading a show on one of the Slaves fails:

`Abort Loading`	In case loading a show fails on one of the Slaves the Master aborts the loading process and generates an error message.

`Report Errors` In case loading a show fails on one of the Slaves an error message is displayed in the "Message Display" window. The loading process is not aborted by this.

`Ignore Errors` The Master ignores all errors which may occur on the Slaves.

4) Select the desired BOOT under "Source"



Master / Slave Setup

Operation Mode

○ Stand-alone

● Master

Enable	IP-Address			
☒ Slave 1:	192	168	1	1
☒ Slave 2:	192	168	1	2
☒ Slave 3:	192	168	1	3
☒ Slave 4:	192	168	1	4
□ Slave 5:	0	0	0	0
□ Slave 6:	0	0	0	0
□ Slave 7:	0	0	0	0

Error Handling: Abort Loading

○ Slave

Source

Use DSP 1 from Main BOOT

Options are "Main BOOT" or "Sub BOOT 'A'" to "Sub BOOT 'G'".
On a Lasergraph DSP *compact (dual)* you can also select the DSP(s) to be used.

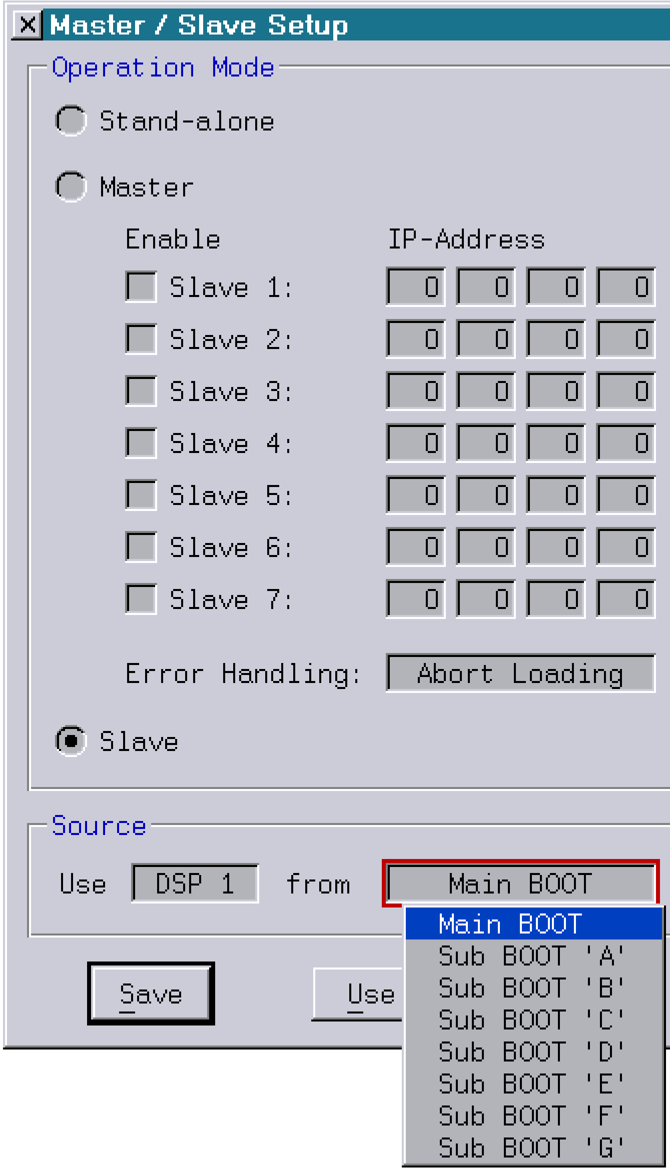
5) Make the changes permanent with "Save".

Slave

For setting up a Lasergraph DSP as "Slave" the following steps are necessary:

1) Activate the checkbox "Slave"

2) Select the desired BOOT under "Source"



Master / Slave Setup

Operation Mode

○ Stand-alone

○ Master

Enable	IP-Address			
□ Slave 1:	0	0	0	0
□ Slave 2:	0	0	0	0
□ Slave 3:	0	0	0	0
□ Slave 4:	0	0	0	0
□ Slave 5:	0	0	0	0
□ Slave 6:	0	0	0	0
□ Slave 7:	0	0	0	0

Error Handling: Abort Loading

● Slave

Source

Use DSP 1 from Main BOOT

Options are "Main BOOT" or "Sub BOOT 'A'" to "Sub BOOT 'G'".
On a Lasergraph DSP *compact (dual)* you can also select the DSP(s) to be used.

3) Make the changes permanent with "Save".

Depending on the timecode source which is used the "I/O Setup" has to be set.
Here we differentiate the following two cases:

1) A PC running LGServer serves as media source.
   This PC has to be entered as I/O server on every Lasergraph DSP (Master and Slaves).

2) An external timecode source is used on the Master (SMPTE, MIDI or VITC).
   In this case the Master has to be entered as I/O server on all Slaves.

When loading a show the file name including the path used on the Slaves is identical to the one on the Master. Here there are the three following possibilities:

1) The show is located on a network drive. It is irrelevant whether the absolute path is specified (\\computer\share\...) or whether the share is mapped to the same drive letter in the "FS Client Setup" on all Lasergraph DSPs.

2) If the show is located on the internal hard disk drive of one of the Lasergraph DSPs this hard disk drive has to be shared in the "FS Server Setup" and all other Lasergraph DSPs have to be connected to this share through the "FS Client Setup".

3) In rare cases it may make sense to have a copy of the show on the local hard disk drive of each individual Lasergraph DSP.

Tip and Tricks:

- All Lasergraph DSPs involved in the Master/Slave operation must have different IP addresses and have to belong to the same subnet, e.g.:
  - Master: IP Address: 192.168.1.1 / Subnet Mask: 255.255.255.0
  - Slave 1: IP Address: 192.168.1.2 / Subnet Mask: 255.255.255.0
  - Slave 2: IP Address: 192.168.1.3 / Subnet Mask: 255.255.255.0

- In case the source selected on the Master is a Sub-BOOT which is not contained in the selected show, the entire show will not be loaded, neither on the Master nor on the Slaves.

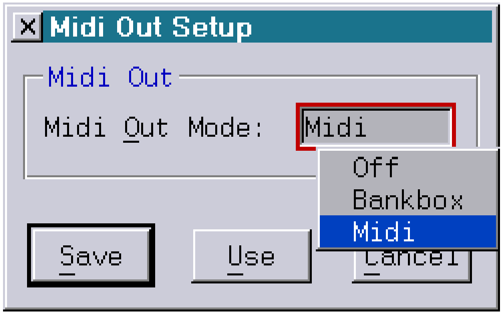
- In case the source selected on the Slave is a Sub-BOOT which is not contained in the selected show, no show will be loaded on this Slave.
  The Master and all the other Slaves will not be affected.

- In order to copy a show with Sub-BOOTs do not load it with `ReBoot` and save it with `CreateBoot` because the Sub-BOOTs would not be copied.
  Use the command `Copy` (with the option `/Subdir`) or `Backup` instead.

- The operating mode Master/Slave can be combined with the playback system LGFD.
  For this LGFD has to be set up on the Master.

- If a Slave is not configured as "Slave" but as "Stand-alone" load operations etc. will still work the same way as long as the Slave is correctly entered in the Master/Slave Setup of the Master. If the Slave is set up in this way it is possible to do things which are usually not possible for Slaves (for example to use "AutoTrack").

MIDI Out Setup



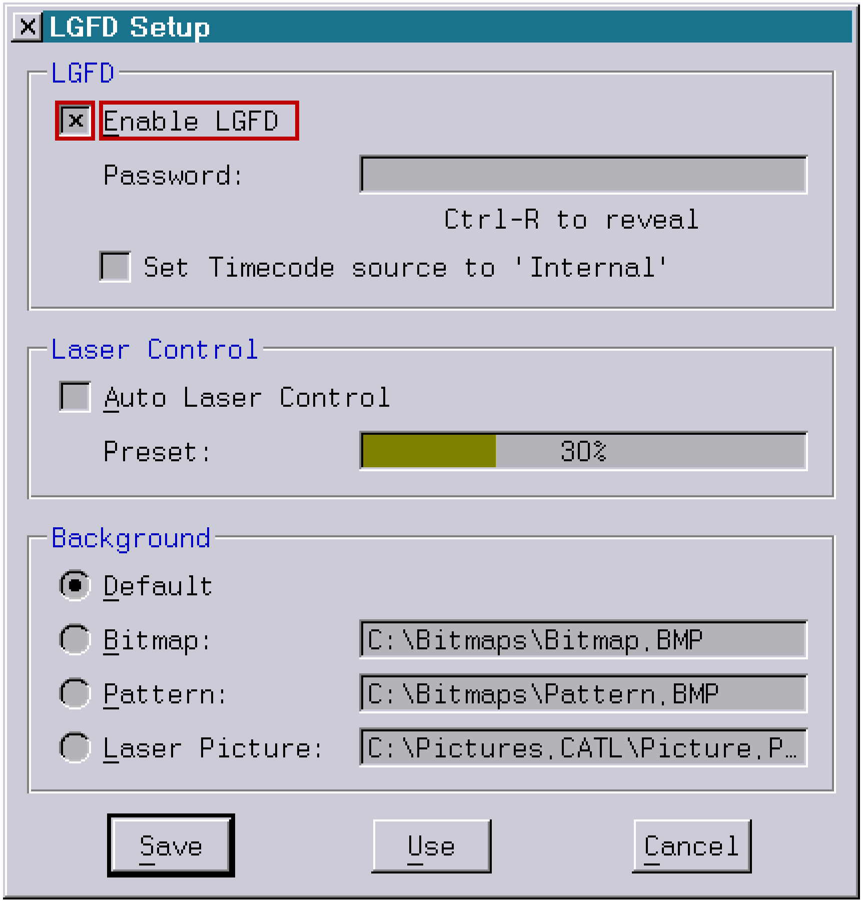
Select “Midi” for the output of MIDI data.

“Bankbox” has to be selected if special peripheral cards to control effect devices etc. on the optical bench are connected to the interface “Midi Out”.

If “Off” is selected the Midi output is disabled.

LGFD Setup

There is a special setup for the LGFD (Lasergraph DSP for discotheques) which can also be accessed using the command "LGFD.Setup":



LGFD Setup

LGFD

☒ Enable LGFD

Password:

Ctrl-R to reveal

□ Set Timecode source to 'Internal'

Laser Control

□ Auto Laser Control

Preset: 30%

Background

● Default

○ Bitmap: C:\Bitmaps\Bitmap.BMP

○ Pattern: C:\Bitmaps\Pattern.BMP

○ Laser Picture: C:\Pictures.CATL\Picture.P…

LGFD

The LGFD mode of the Lasergraph DSP can be activated by "Enable LGFD". Normally the LGFD mode can be exited at any time using the key combination "CTRL-F12". If this is not desired this function can be protected with a password.

If "Set Timecode source to 'Internal'" is checked the timecode source will automatically switched to "Internal" after a show has been loaded.

Laser Control

If the laser should be controlled by the LGFD, the option "Auto Laser Control" can be activated. In this case the laser is automatically started in standby (with minimum power). As soon as a show is started, the laser output is increased to the value specified in the preset. About 30 seconds after the end of the show the laser is automatically switched back to standby.

The laser control in the LGFD always applies to laser no. 1. If the laser control is active, the current laser state is displayed in the upper left corner:
"Laser On" if the laser is running.
"Laser Off" if the laser is not running.
"No Laser" if no laser has been detected.
By clicking on the laser button with the mouse and keeping the mouse key pressed a window is opened which displays additional information abort the laser state.

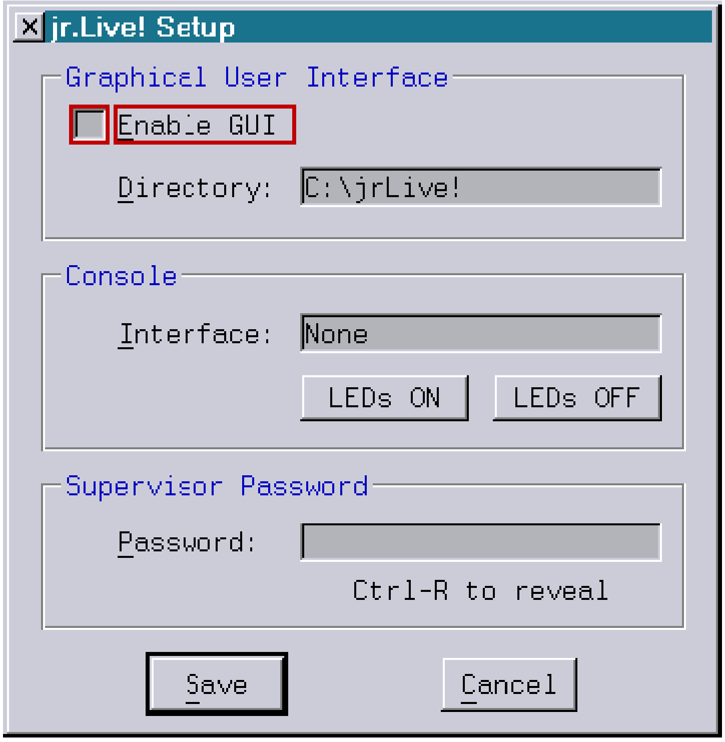
Background

Finally the background of the LGFD display can be specified in the setup:

1. "Default"	the boot picture of the Lasergraph DSP
2. "Bitmap"	a bitmap which is displayed once in the center
3. "Pattern"	a bitmap which is displayed a often as it fits on the screen
4. "Laser Picture"	a user specified laser picture.

For more details on the LGFD interface see “Chapter LGFD”.

jr.Live! Setup



Graphical User Interface

□ Enable GUI

Directory: C:\jrLive!

Console

Interface: None

Supervisor Password

Password:

Enable GUI

In order to get to the "jr. Live!", "Enable GUI" has to be activated.
The "jr. Live!" interface display requires several bitmaps, e.g. for the background or for the numerous buttons.
Under "Directory" you can specify the location of these bitmaps.

Once the GUI is enabled, the DSP will startup in Jr.Live mode.
To exit the jr.Live use the shortcut:

<ctrl> <alt> <backspace>   or   <ctrl> <F12>

Console

For future versions of the Live.BOOT we are thinking of implementing control using a console.
As long as this is not available "None" should be selected as interface.

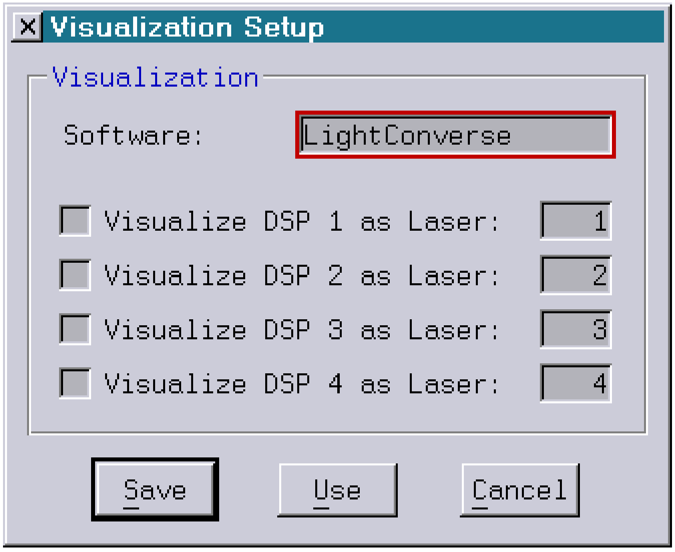
Supervisor Password

Exiting the "jr. Live!" interface can be restricted to authorized persons by setting a password, especially for fixed installations.
Attention! If you set a password here make sure to remember it. Otherwise you will not be able to get back to the normal DSP interface either!

For details on the jr.Live GUI, see the jr.Live manual.

Visualization Setup

workstation Mark 2:



Visualization Setup

Visualization

Software: LightConverse

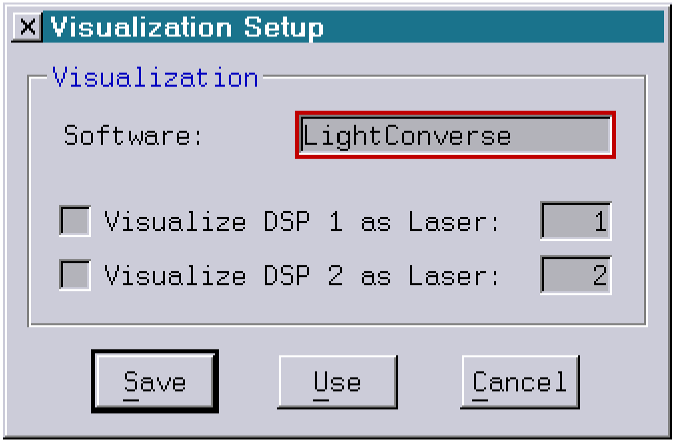
☐ Visualize DSP 1 as Laser: 1

☐ Visualize DSP 2 as Laser: 2

☐ Visualize DSP 3 as Laser: 3

☐ Visualize DSP 4 as Laser: 4

compact dual Mark 2:



Visualization Setup

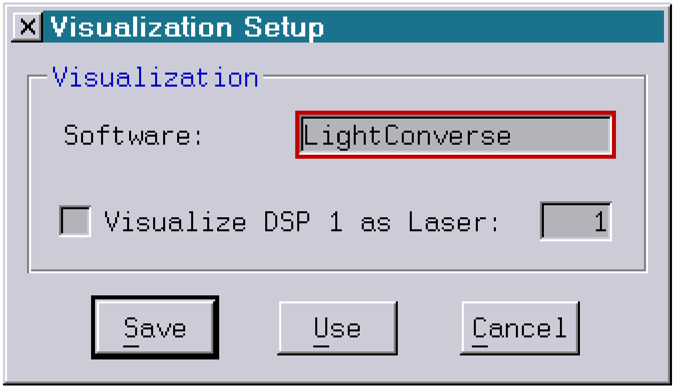
Visualization

Software: LightConverse

☐ Visualize DSP 1 as Laser: 1

☐ Visualize DSP 2 as Laser: 2

compact Mark 2:



Visualization Setup

Visualization

Software: LightConverse

☐ Visualize DSP 1 as Laser: 1

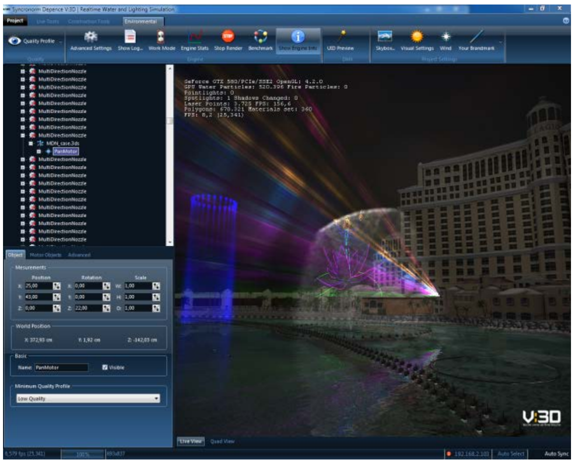
The Setup consists of two separate sections:

- The desired visualization software can be selected in the upper section.

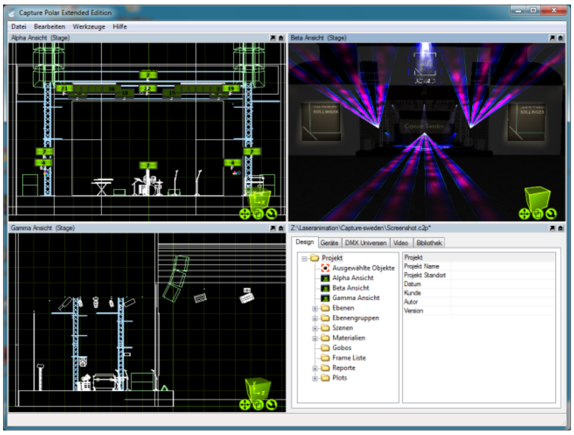
  The following products are supported by the Lasergraph DSP:

LightConverse

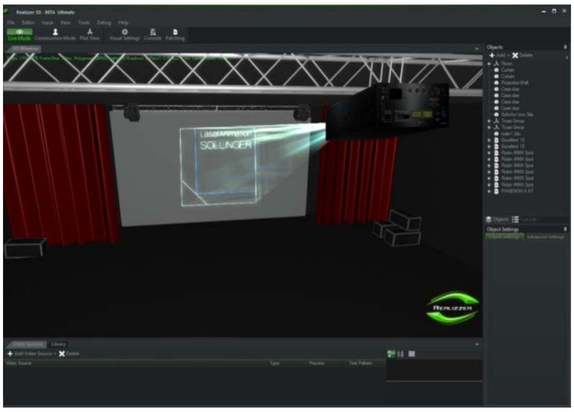
Depence V:3D



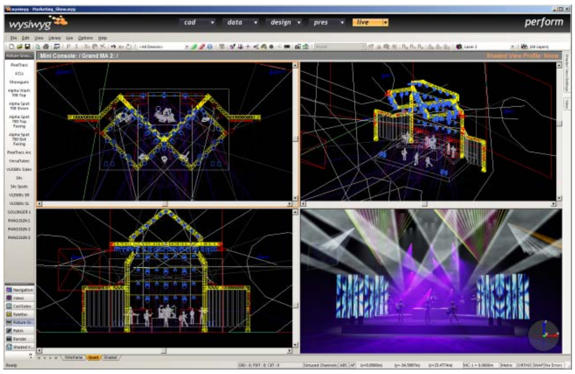
Capture



Realizzer 3D



wysiwyg



- Visualization can be activated or deactivated for the output of each DSP individually in the lower section. In case visualization is activated the virtual laser number is also assigned here.
  The virtual laser number for the laser projector is entered into the visualization software. It serves to identify individual channels.

  The virtual laser number has to be in a range from 1 to 256.

Note:

As with the preview output, visualization requires large amounts of data to be transmitted over the network.

This can lead to a reduction of the laser output quality, especially with simultaneous visualization of several DSPs on workstations with a 68B-4 processor.

Therefore the following points have to be taken into account:

1. Visualization should only be activated when it is actually used.
2. It is advisable to turn off the output in the preview window during visualization.

Laser Disable Setup

The mechanism "Laser Disable" offers a simple way to activate resp. deactivate the laser output on one or more Lasergraph DSP Mark 2 from one or more positions.

In addition the Lasergraph DSP Mark 2 can be combined into groups. If necessary these groups can be turned on or off independent of each other (e.g. during installation or programming).

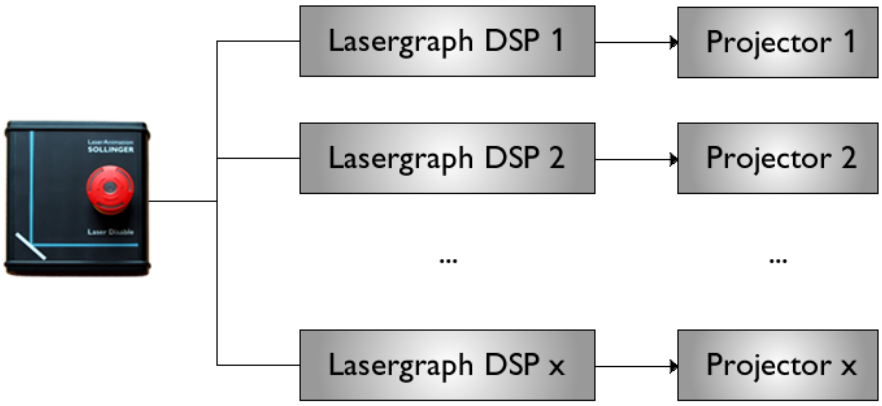
For operation of this mechanism there is the software "Laser Disable" which runs under Windows® or macOS® as well as a special network enabled emergency-off button called "Laser Disable Button".

Since all communication is done over Ethernet no additional cables have to be installed and the individual buttons can be placed in different locations (e.g. FOH and backstage).

Whether the "Laser Disable" mechanism can replace the emergency-off button which is required by law or if it can only be used in addition depends on the respective laws and has to be determined in each individual case.

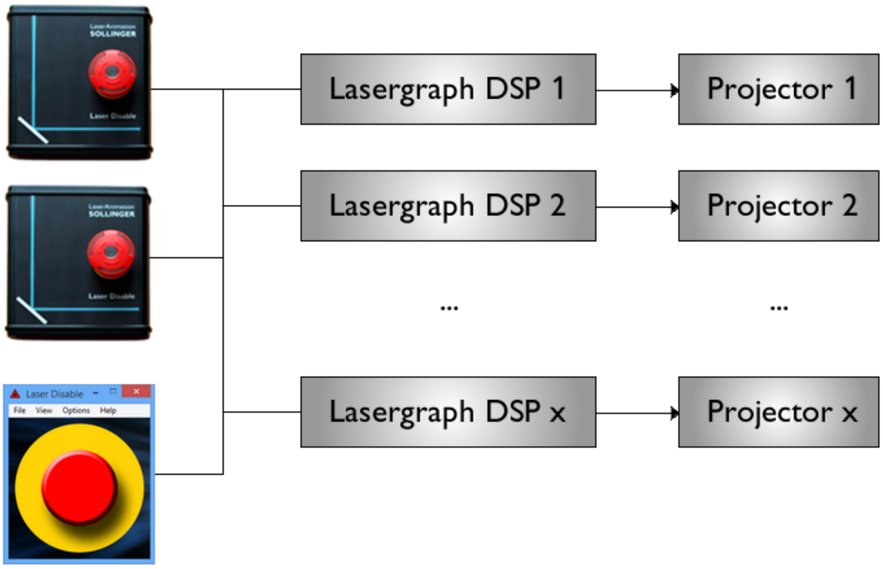
Example 1:

One Laser Disable Button controls one or more laser projectors.



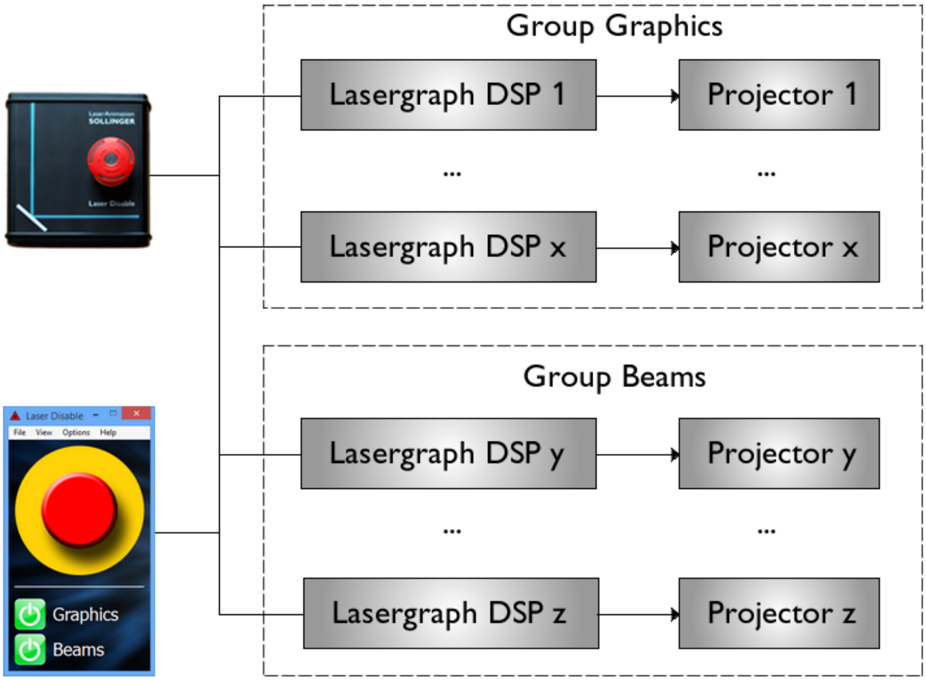
Example 2:

Two Laser Disable Buttons and the software "Laser Disable" control one or more laser projectors.



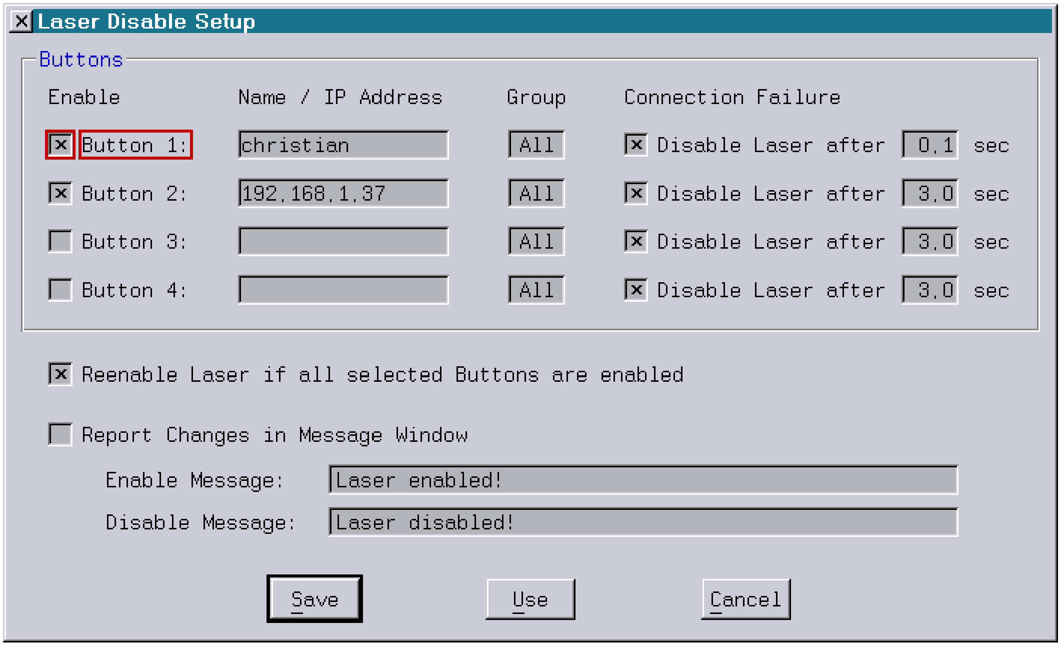
Example 3:

One Laser Disable Button and the software "Laser Disable" control several laser projectors divided into two groups.
The Laser Disable Button disables the laser output on all projectors while the software "Laser Disable" can also be used to control each group individually.



Setting up "Laser Disable" is quite simple:

On each Lasergraph DSP the buttons which this Lasergraph DSP is to react to are entered in the setup:



Laser Disable Setup

Buttons

Enable	Name / IP Address	Group	Connection Failure		
☒ Button 1:	christian	All	☒ Disable Laser after	0.1	sec
☒ Button 2:	192.168.1.37	All	☒ Disable Laser after	3.0	sec
□ Button 3:		All	☒ Disable Laser after	3.0	sec
□ Button 4:		All	☒ Disable Laser after	3.0	sec

☒ Reenable Laser if all selected Buttons are enabled

□ Report Changes in Message Window

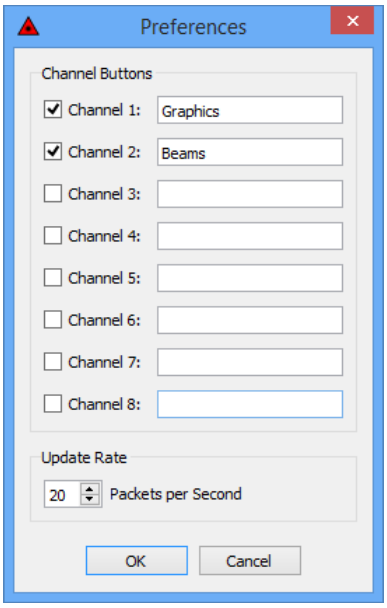
Enable Message: Laser enabled!

Disable Message: Laser disabled!

Each Lasergraph DSP can evaluate the states of up to four buttons simultaneously.

A button (Laser Disable Button or PC/Mac running the software "Laser Disable") is identified alternatively by its network name or its IP address.

For each button it is possible to independently set, which of the 8 possible groups the Lasergraph DSP is to react to. For this the number of the group has to be identical to the channel number used in the software "Laser Disable":



Preferences

Channel Buttons

☑ Channel 1: Graphics

☑ Channel 2: Beams

□ Channel 3:

□ Channel 4:

□ Channel 5:

□ Channel 6:

□ Channel 7:

□ Channel 8:

Update Rate

20 Packets per Second

The "Connection Failure" setting determines whether the Lasergraph DSP is to deactivate its laser output in case the connection to the button is lost.
The timeout range from 0.1s to 60s set here defines the time span during which a valid data packet has to be received from the button.
If no data packet is received during this time the laser output will be deactivated.

In the software "Laser Disable" the number of packets to be sent per second can be set.
To ensure that a lost packet does not lead to an unwanted deactivation of the laser output the update rate should be set so that at least three packets are sent within the timeout.

Example:

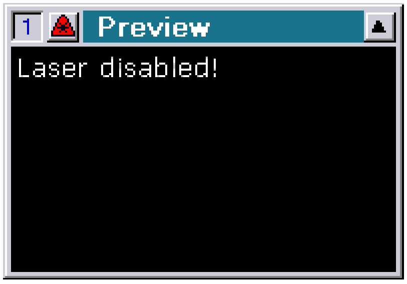
update rate = 1 packet per second	timeout: min. 3 seconds
update rate = 30 packets per second (33ms)	timeout: min. 0.1 seconds (100ms)

Notes:

- The timeout set in the setup is only relevant in case the connection between button and Lasergraph DSP was interrupted.
  As long as the connection exists, pressing the button will immediately deactivate the laser output.

- "Update Rate" and "Timeout" should be set depending on the quality of the connection. For instance, in case of a bad Wi-Fi connection it may be advisable to select a very long timeout or to completely turn off the automatic deactivation so the show is not interrupted unintentionally.

In case "Reenable Laser if all Buttons are enabled" is turned off and one of the specified buttons has turned the laser output off the laser output has to be turned on manually using the button in the preview window.



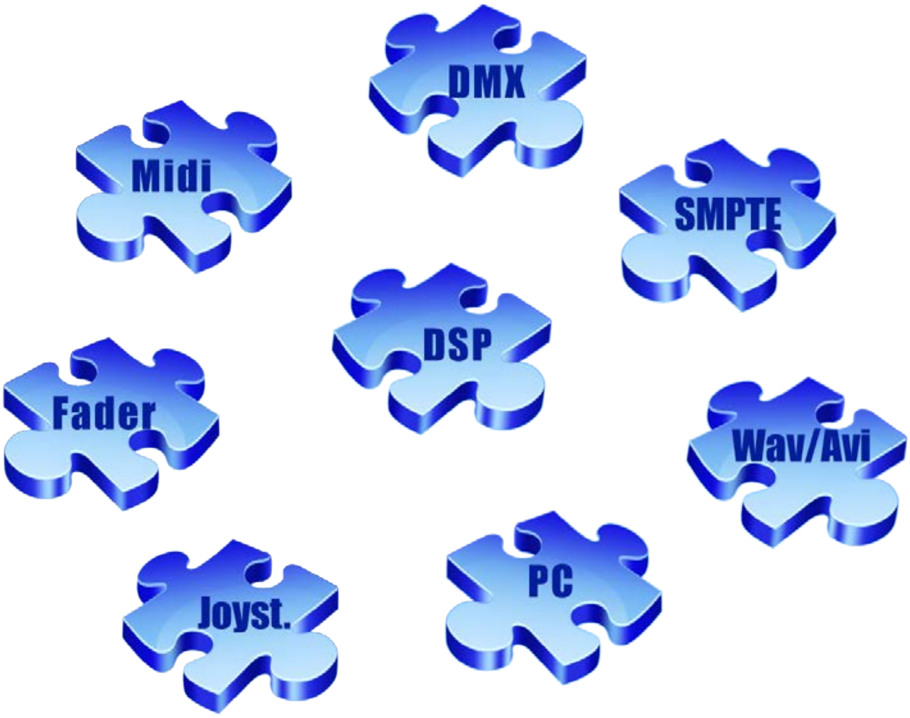
Laser disabled!

If desired a message can be displayed in the "Massage Display" window if the laser output is activated resp. deactivated by the "Laser Disable" mechanism. The messages which are displayed can be chosen freely under "Enable Message" resp. "Disable Message".

The respective message will only be displayed if "Report Changes in Message Window" is activated and if the corresponding text is not empty.

Mark 2 Manual

Chapter Network Setup



LaserAnimation Sollinger GmbH · Crellestr. 19/20 · D-10827 Berlin
Phone +49 (30) 780 963 00 · Fax +49 (30) 780 963 25
E-mail info@laseranimation.com · Internet www.laseranimation.com

Introduction

Setting up a laser system with several lasers, scanners, color controls and control computers as well as the necessary periphery for laser shows requires complicated cable connections, especially when the positions of the components are far apart. This has considerable disadvantages, such as a deterioration of signal quality due to loss of signals as well as the immense amount of work necessary for setting up the system.

If, however, several Lasergraph DSPs are connected via an Ethernet network, the number of necessary cables is greatly reduced. One big advantage is the fact that the Lasergraph DSPs can be positioned in direct proximity to scanners, color controls and optical benches so that no signal losses due to long cables will occur.

For remote controlling the Lasergraph DSPs a PC with the program LGRemote can be used.

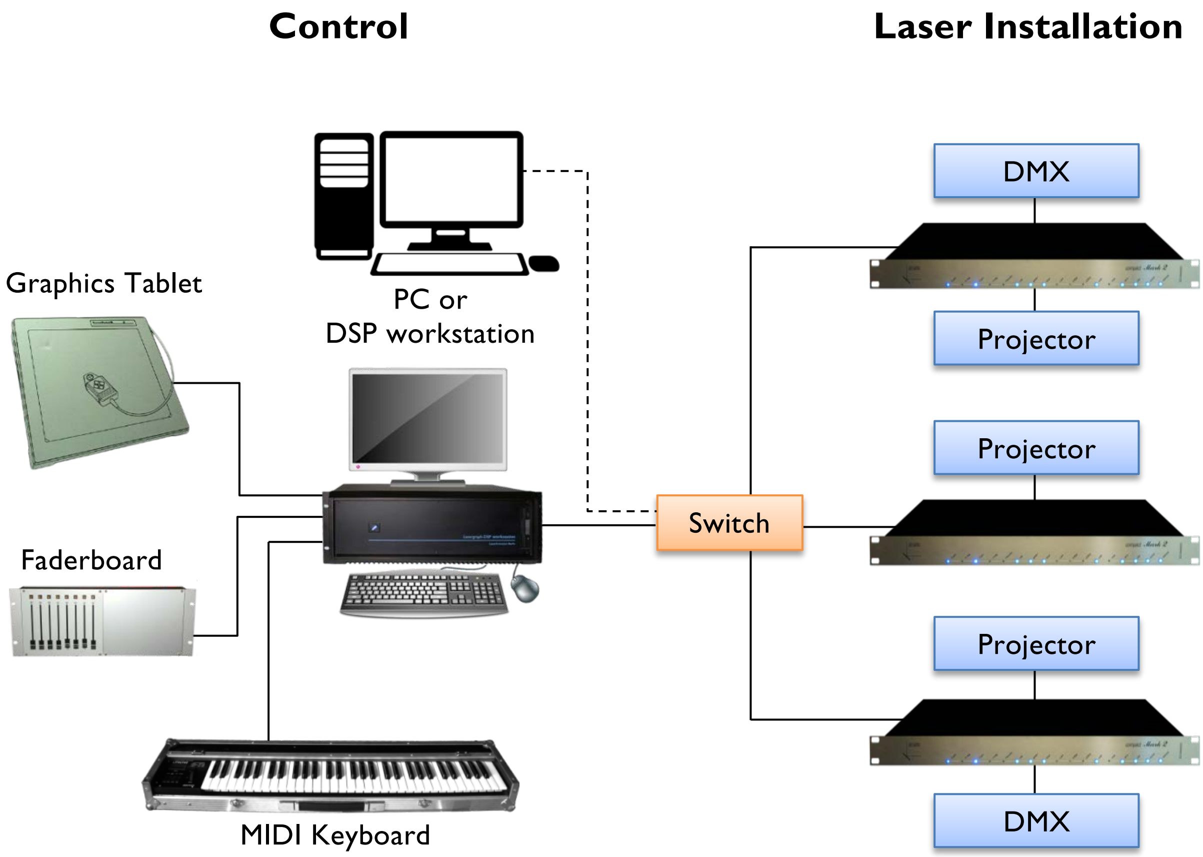
When Lasergraph DSPs and PCs are connected via a network the data transfer between both can easily be handled.
In addition peripheral devices (e.g. CD-ROM, MIDI, Faderboard etc.) that are connected only to one Lasergraph DSP or to the PC are available to other Lasergraph DSPs in the network.

Lasergraph DSPs and PC are connected to an Ethernet hub or switch using standard CAT-5 Patch network cables (maximum length of 100 meters) with RJ-45 connectors.

If only 2 computers are to be connected, e.g. a Lasergraph DSP and a PC, no hub or switch is necessary.

When using wireless Ethernet, no cables between Lasergraph DSP and remote control are necessary at all.

The following picture shows an example for remote controlling a laser installation with several Lasergraph DSP *compact* via network from a PC or a Lasergraph DSP *workstation* to which a graphics tablet, a Faderboard and a MIDI-keyboard are connected.

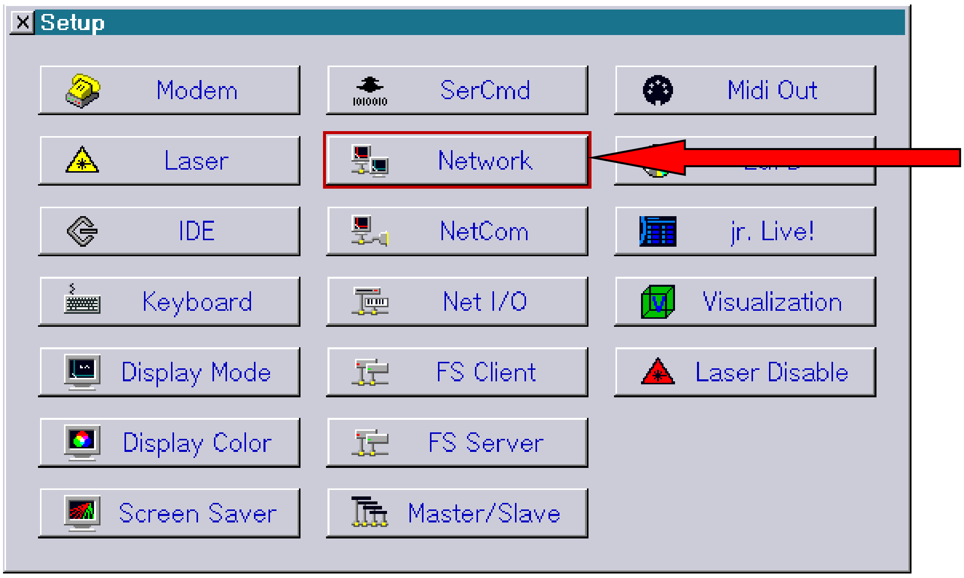


Setting Up the Software

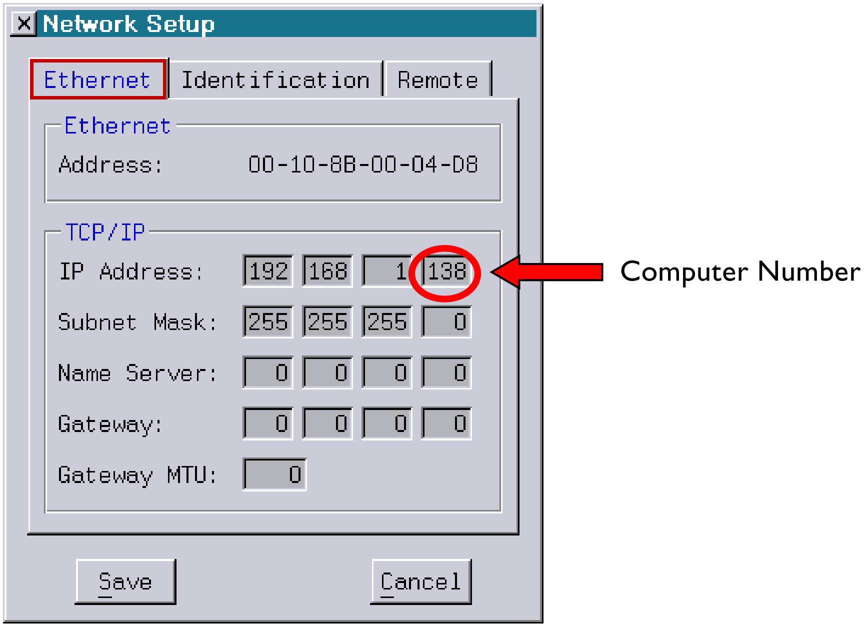
Before a number of computers (PCs, Lasergraph DSPs) can be used via a network, the software has to be configured for network operation.

Setting Up the Lasergraph DSP

Network configuration for the Lasergraph DSP is done in the Setup.
Call up the "Setup" on the Lasergraph DSP and select the button Network.



Ethernet



Ethernet
Address: 00-10-8B-00-04-D8

TCP/IP
IP Address: 192 168 1 138
Subnet Mask: 255 255 255 0
Name Server: 0 0 0 0
Gateway: 0 0 0 0
Gateway MTU: 0

Computer Number

The Ethernet address (MAC) is only valid for this Lasergraph DSP. No other unit worldwide has the same address. It cannot be changed.

The TCP/IP settings are used to identify the computer in the network.
If you want to create a local network (LAN), for instance to connect several Lasergraph DSPs, you can select and enter the values for the IP address yourself. The IP address identifies the computer uniquely in the network. Therefore every computer in a network has to have its own IP address. One IP address may not be assigned more than once.

The following ranges of IP addresses are available for local networks, others may not be used:

`10.1.1.1`	to	`10.254.254.254`
`172.16.1.1`	to	`172.31.254.254`
`192.168.1.1`	to	`192.168.254.254`

A local network is defined by the *Subnet Mask*. In the Subnet Mask the parts of the IP address which have to be identical are defined.
When a computer is to be integrated into an existing network, the Subnet Mask of the existing network has to be used.
If you build up a new network, simply use the Subnet Mask 255.255.255.0. With this you can connect 254 computers to the local network.

The IP address entered in each computer has to be identical in all places in which the Subnet Mask is not equal 0.

For instance:

Subnet Mask	255.	255.	255.	0
IP Address	192.	168.	1.	1
	192.	168.	1.	2
	"	"	"	.
	"	"	"	.
	192.	168.	1.	254

This way, 254 Computers can be connected to a local network and thereby belong to the same network.

Computers do not belong to the same network when the IP configuration for instance looks like this:

Subnet Mask	255.	255.	255.	0
IP Address	192.	168.	1.	100
IP Address	192.	168.	2.	101

No direct connection is possible between computers belonging to different subnets.

If the *Subnet Mask* 255.255.255.0 is selected, the last value of the IP address is the computer number. Every computer in the network has to have a different number between 1 and 254.

The following is an example for possible settings for a local network:

First Lasergraph DSP:	`192.168.1.121`
Second Lasergraph DSP:	`192.168.1.122`
Third Lasergraph DSP:	`192.168.1.123`

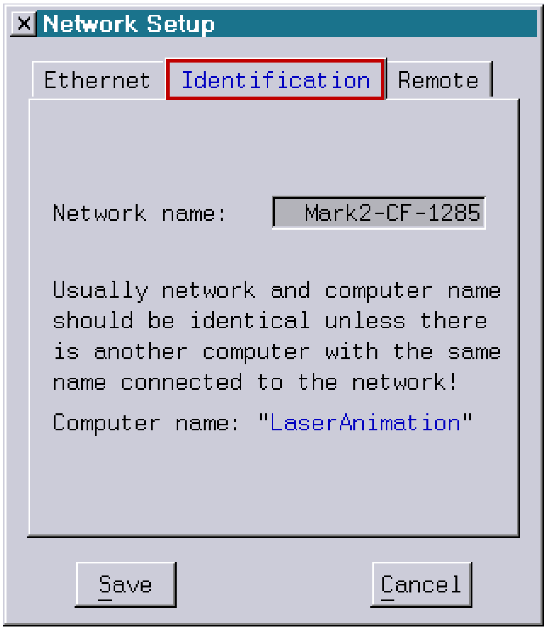
The entry fields for *Name Server* and *Gateway* are only relevant for network access outside the local network, for instance over the Internet.

A name server is a program or computer server that implements a name-service protocol. It maps the name of a host to its computer-recognizable identifier (e.g. IP address 88.33.22.44).

A Gateway is a device that "routes" data from the local network to non local networks (e.g. the internet).

Normally you can keep these values at zero.
If you need to connect to another than your local network or you want to know more about these settings, you will find an example at the end of this document.

Identification



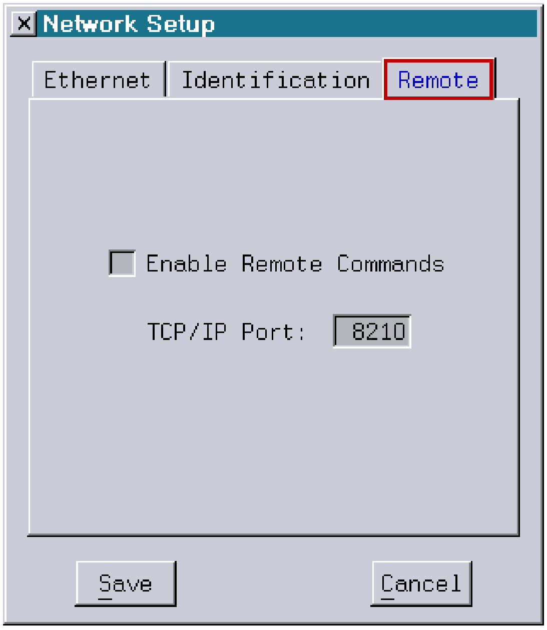
Network name: Mark2-CF-1285

Usually network and computer name should be identical unless there is another computer with the same name connected to the network!

Computer name: "LaserAnimation"

Enter a name for this Lasergraph DSP for identification.
Each computer in a network should have its own name so the computers can be distinguished.

Remote



☐ Enable Remote Commands

TCP/IP Port: 8210

It is possible to send commands to the Lasergraph DSP over a TCP/IP port which can be freely chosen. These commands will be put into the command line of the Lasergraph DSP and will be executed as if they had been entered locally.

In order to use this functionality activate the checkbox "Enable Remote Commands".

For more information see chapter "SerCmd".

After working on the network setup you have two possibilities:

Save This setting is standard from now on.

Cancel All changes are discarded and the previous setting is restored.

Integrating a PC into an Ethernet-Network

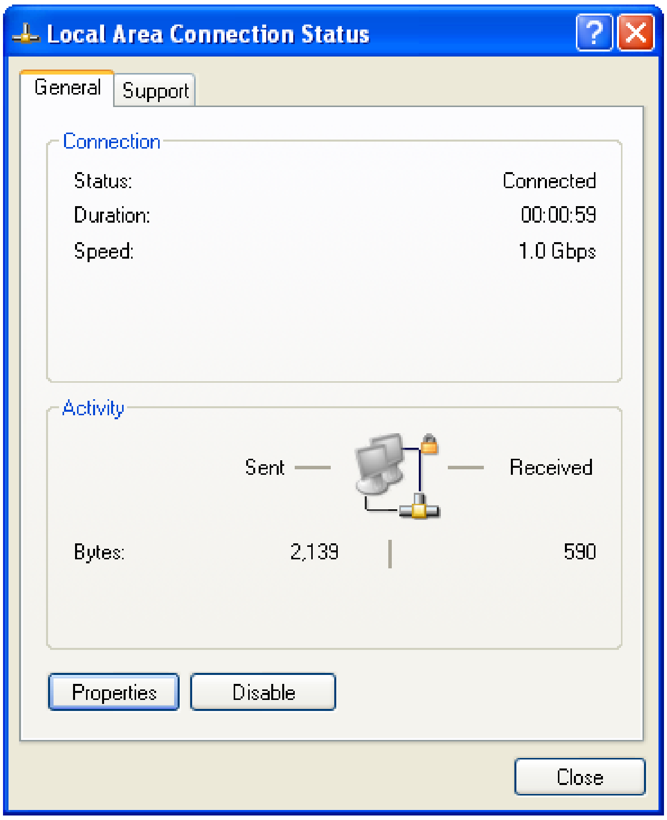
A PC can be connected to an Ethernet network together with other Lasergraph DSPs and can be used with these in the network. Two steps are necessary for this:

1. Integration of the PC into the Network.
2. Installation of LGRemote for remote controlling a Lasergraph DSP from the PC.

Integrating a PC under Windows® XP

Open the *Control Panel* and select *Network and Internet Connections, Network Connections* and then *Local Area Connection*.

You will reach the following window:



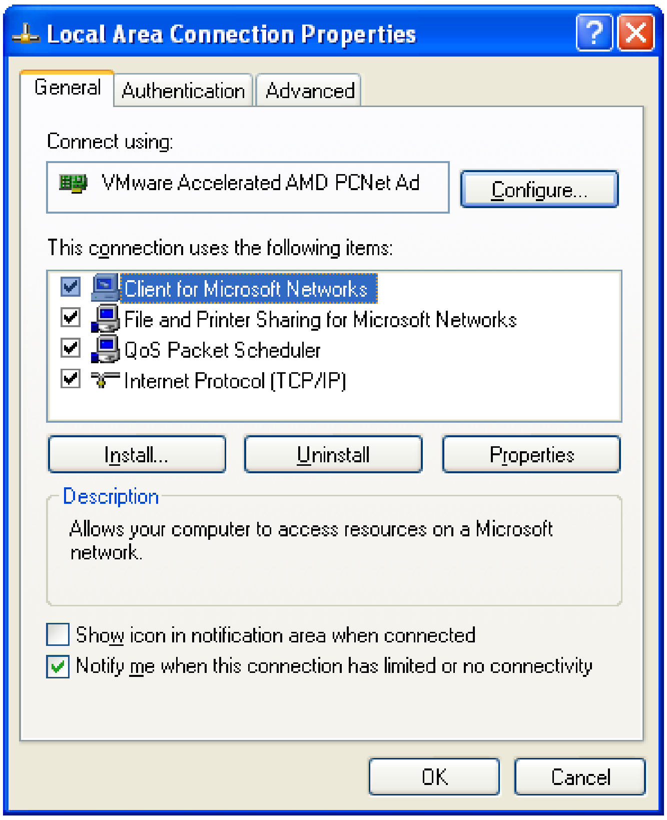
Status: Connected

Duration: 00:00:59

Speed: 1.0 Gbps

Bytes: 2,139 | 590

Click on *Properties* to open the following window:



Connect using:

VMware Accelerated AMD PCNet Ad

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ Internet Protocol (TCP/IP)

Description

Allows your computer to access resources on a Microsoft network.

☐ Show icon in notification area when connected

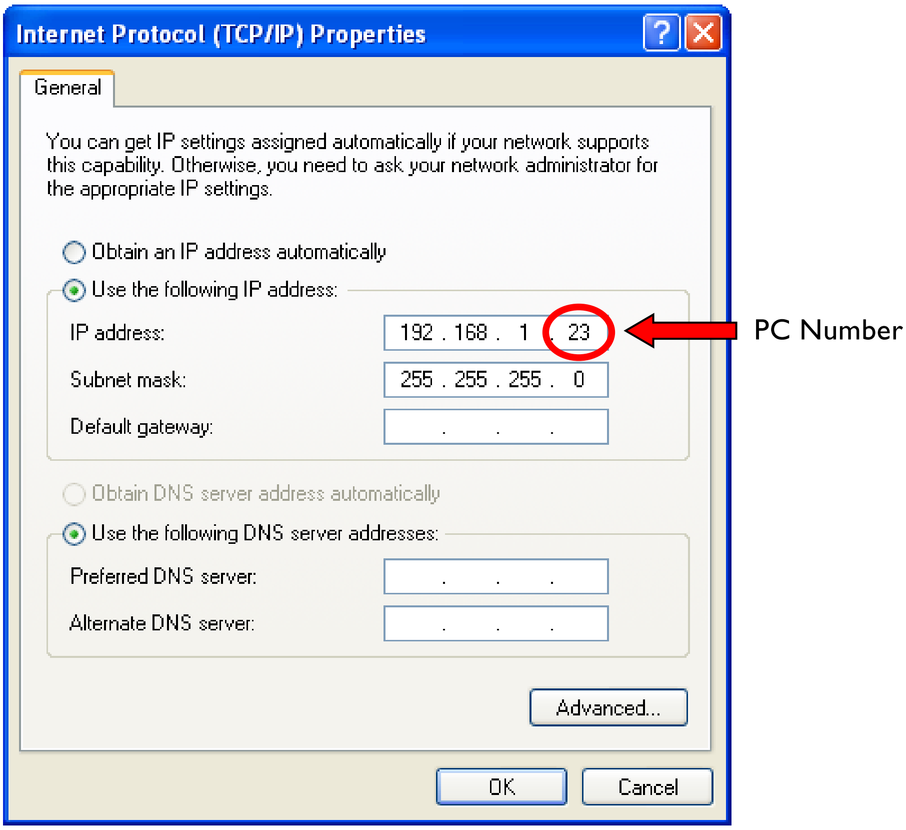
☑ Notify me when this connection has limited or no connectivity

Activate the checkboxes for

- ✓ Client for Microsoft Networks,
- ✓ File and Printer Sharing for Microsoft Networks and
- ✓ Internet Protocol (TCP/IP).

To specify the IP address select Internet Protocol (TCP/IP) and click on Properties in the window.

You will reach the following window:



Internet Protocol (TCP/IP) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address:	192 . 168 . 1 . 23
Subnet mask:	255 . 255 . 255 . 0
Default gateway:	. . .

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server:	. . .
Alternate DNS server:	. . .

PC Number

Here you can enter the *IP address* and the *Subnet mask*.
If you are adding your PC to an existing network, enter the first three values so they are identical to those on the other computers in the network.

If no IP address was defined, here again is the list of the IP addresses reserved for local networks:

`10.1.1.1`	to	`10.254.254.254`
`172.16.1.1`	to	`172.31.254.254`
`192.168.1.1`	to	`192.168.254.254`

Enter a number between 1 and 254 for the PC as the fourth value. Please note that each computer in the network has to have its own number. The computer in the example has the number 23 and no other computer in the network may have this number.

An example for possible settings for a local network is the following:

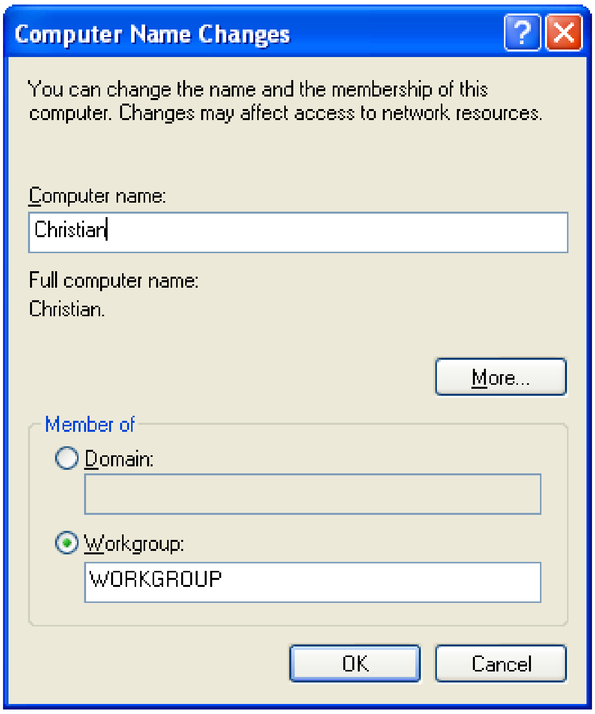
First PC:	`192.168.1.21`
Second PC:	`192.168.1.22`
First Lasergraph DSP:	`192.168.1.121`
Second Lasergraph DSP:	`192.168.1.122`
Third Lasergraph DSP:	`192.168.1.123`

Enter the *Subnet Mask* for the local network, e.g. 255.255.255.0. Leave all other settings as they are and click *OK*.

(The entry fields for *Use the following DNS Server addresses* are relevant for network access outside the local network, for instance to the internet or intranet - DNS = Domain Name System.)

For the identification of the PC in the network change back to your *Control Panel* and select *Performance and Maintenance* and then *System*.

Click on the tab *Computer Name* and select *Change*:



Computer Name Changes

You can change the name and the membership of this computer. Changes may affect access to network resources.

Computer name: Christian

Full computer name:
Christian.

Member of

○ Domain:

● Workgroup: WORKGROUP

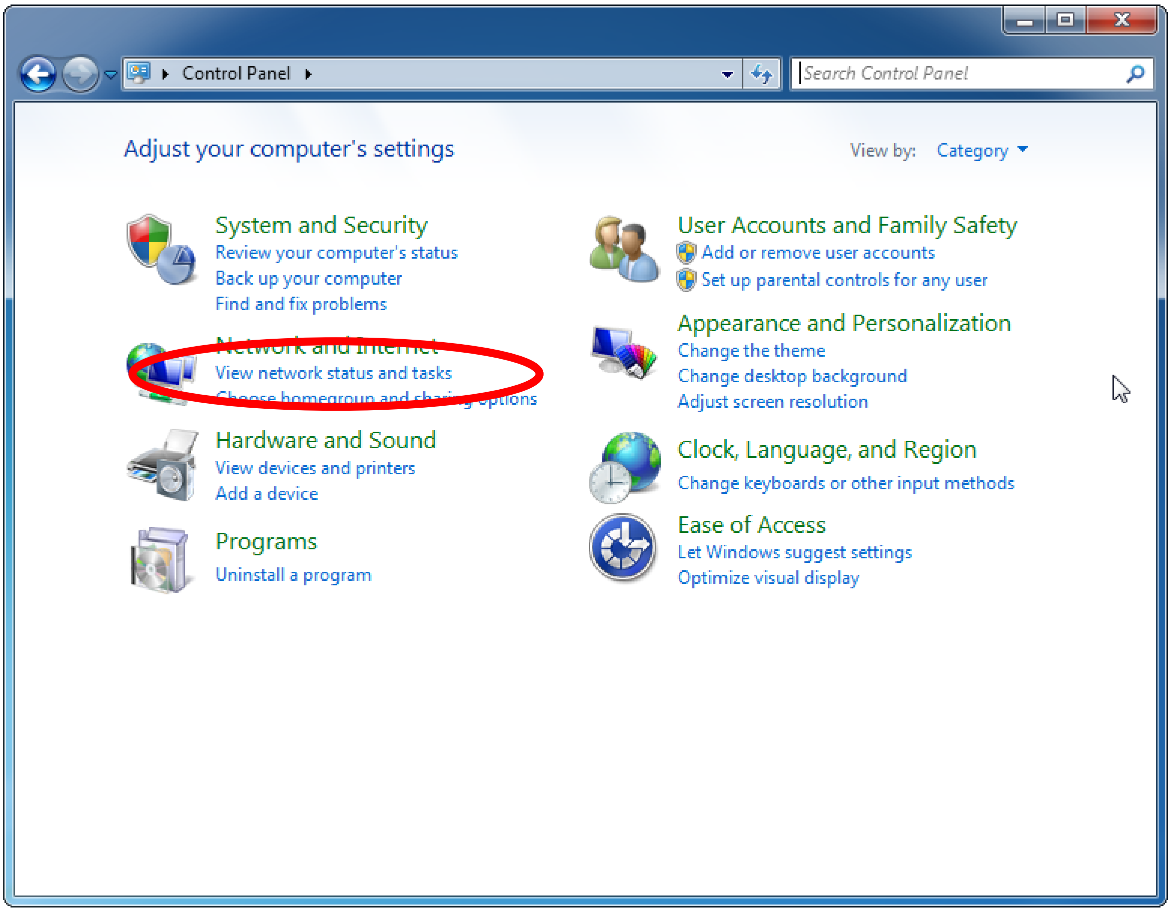
Enter a name of the computer to identify the PC in the network.

The settings will be effective after restarting the computer.

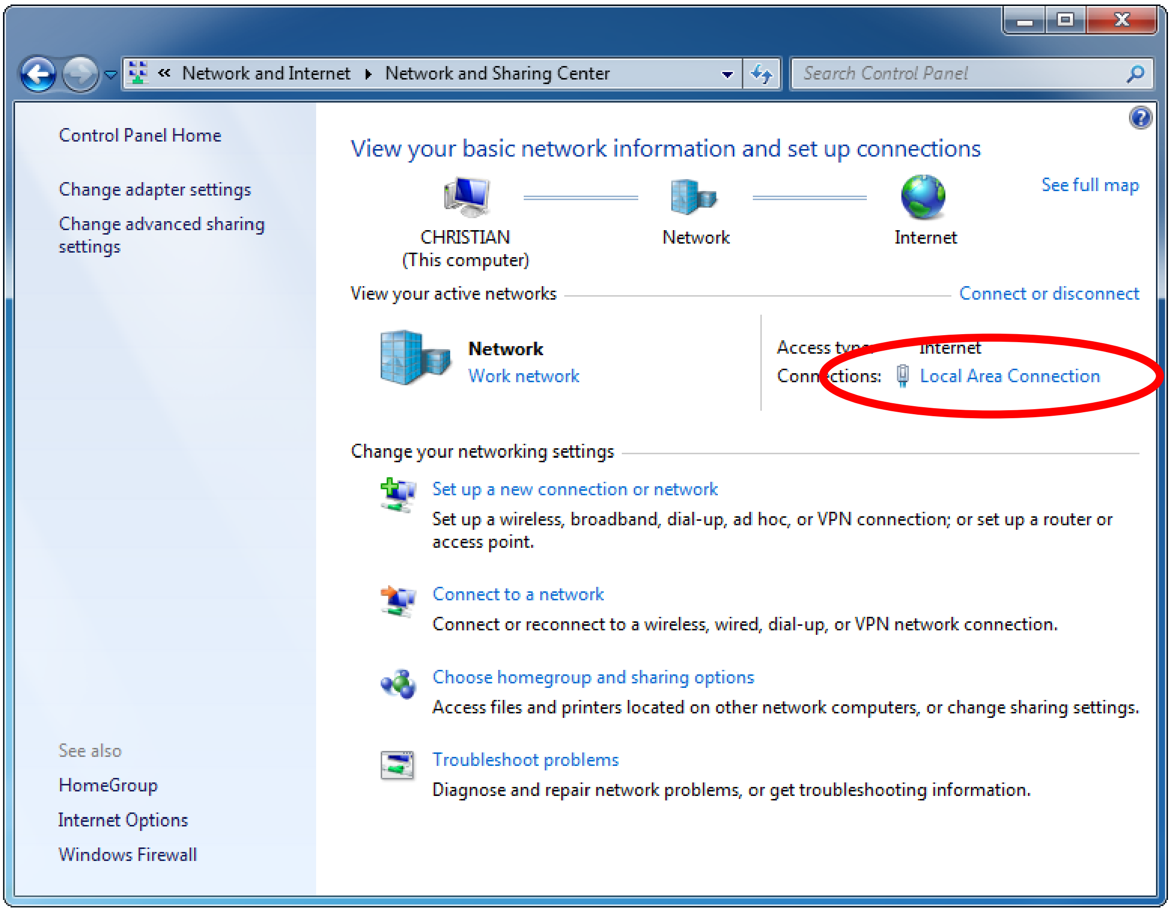
Integrating a PC under Windows® Vista/7

The following description refers to setting up a network under Windows® 7.
For Windows® Vista the setup is similar.

Open the *Control Panel* and select *View network status and tasks.*


Control Panel
Search Control Panel
Adjust your computer's settings
View by: Category
System and Security
Review your computer's status
Back up your computer
Find and fix problems
Network and Internet
View network status and tasks
Choose homegroup and sharing options
Hardware and Sound
View devices and printers
Add a device
Programs
Uninstall a program
User Accounts and Family Safety
Add or remove user accounts
Set up parental controls for any user
Appearance and Personalization
Change the theme
Change desktop background
Adjust screen resolution
Clock, Language, and Region
Change keyboards or other input methods
Ease of Access
Let Windows suggest settings
Optimize visual display

Choose *Local Area Connection*



« Network and Internet ▸ Network and Sharing Center

Search Control Panel

Control Panel Home

Change adapter settings

Change advanced sharing settings

See also

HomeGroup

Internet Options

Windows Firewall

View your basic network information and set up connections

See full map

CHRISTIAN (This computer) — Network — Internet

View your active networks — Connect or disconnect

Network
Work network

Access type: Internet
Connections: Local Area Connection

Change your networking settings

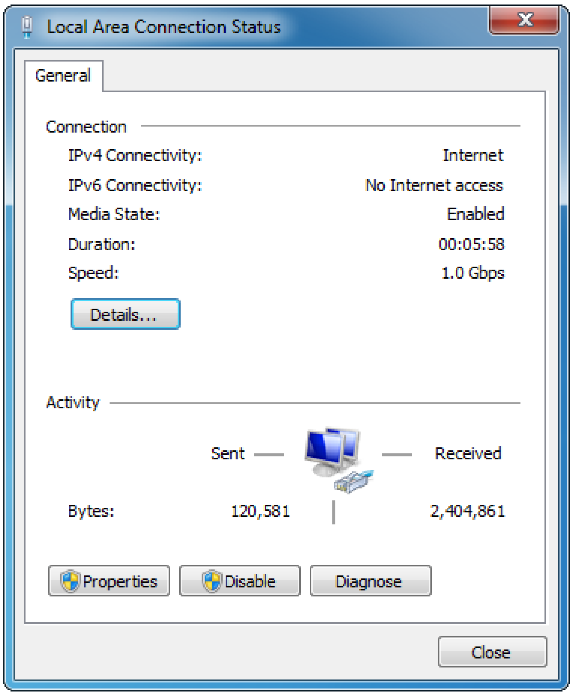
Set up a new connection or network
Set up a wireless, broadband, dial-up, ad hoc, or VPN connection; or set up a router or access point.

Connect to a network
Connect or reconnect to a wireless, wired, dial-up, or VPN network connection.

Choose homegroup and sharing options
Access files and printers located on other network computers, or change sharing settings.

Troubleshoot problems
Diagnose and repair network problems, or get troubleshooting information.

to open the *Local Area Connection Status*



Local Area Connection Status

General

Connection

IPv4 Connectivity:	Internet
IPv6 Connectivity:	No Internet access
Media State:	Enabled
Duration:	00:05:58
Speed:	1.0 Gbps

Details...

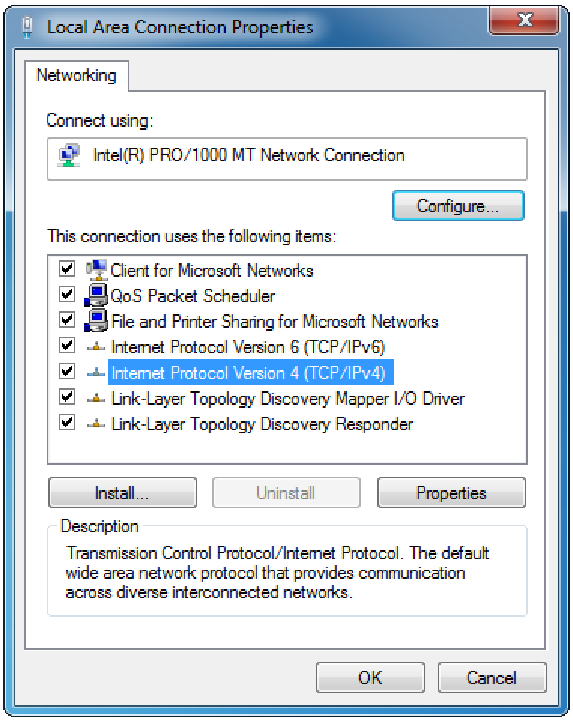
Activity

	Sent	Received
Bytes:	120,581	2,404,861

Properties Disable Diagnose

Close

Choose *Properties* to reach the following window:



Local Area Connection Properties

Networking

Connect using:

Intel(R) PRO/1000 MT Network Connection

This connection uses the following items:

☑ Client for Microsoft Networks
☑ QoS Packet Scheduler
☑ File and Printer Sharing for Microsoft Networks
☑ Internet Protocol Version 6 (TCP/IPv6)
☑ Internet Protocol Version 4 (TCP/IPv4)
☑ Link-Layer Topology Discovery Mapper I/O Driver
☑ Link-Layer Topology Discovery Responder

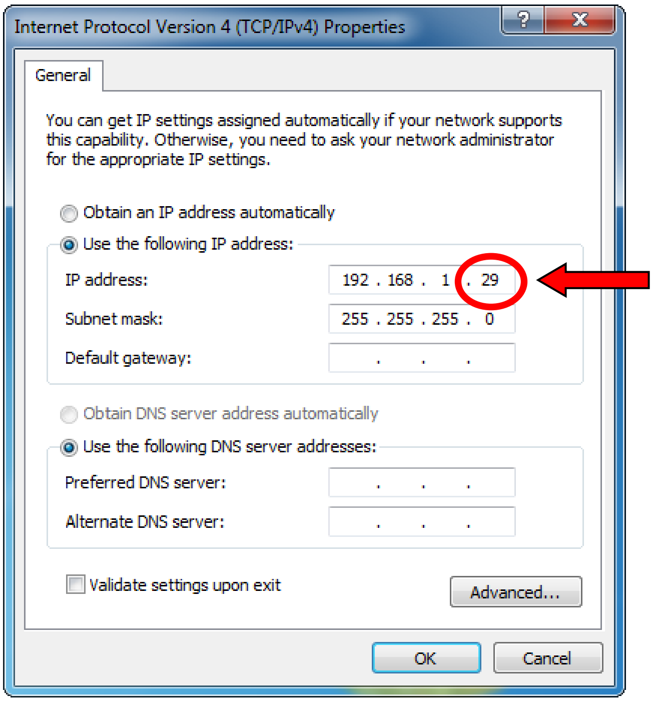
Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Activate the checkboxes for

- ✓ Client for Microsoft Networks,
- ✓ File and Printer Sharing for Microsoft Networks and
- ✓ Internet Protocol Version 4 (TCP/IPv4).

To specify the IP address select *Internet Protocol Version 4 (TCP/IPv4)* and click on *Properties*.



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:

IP address: 192 . 168 . 1 . 29
Subnet mask: 255 . 255 . 255 . 0
Default gateway: . . .

○ Obtain DNS server address automatically
● Use the following DNS server addresses:

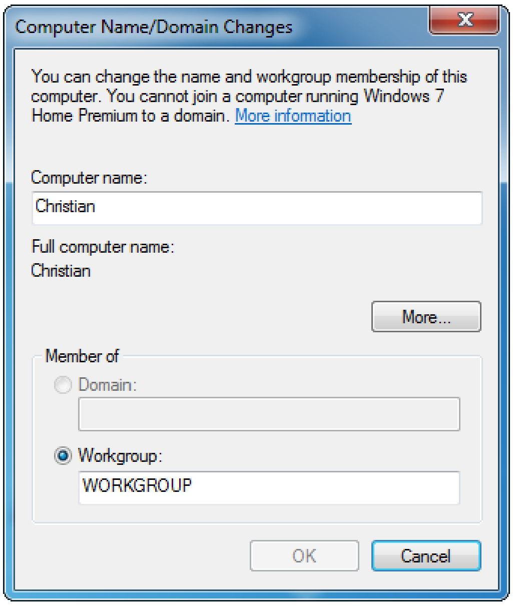
Preferred DNS server: . . .
Alternate DNS server: . . .

□ Validate settings upon exit

PC Number

Here you can enter the *IP address* and the *Subnet mask* => compare explanations p. 11ff.
Leave all other settings as they are and click *OK*.

For the identification of the PC in the network go back to the *Control Panel* select *System and Security* and click on *Set the name of this computer* to open the following window:



Computer Name/Domain Changes

You can change the name and workgroup membership of this computer. You cannot join a computer running Windows 7 Home Premium to a domain. More information

Computer name: Christian

Full computer name: Christian

Member of
○ Domain:
● Workgroup: WORKGROUP

Enter a name of the computer to identify the PC in the network.
The settings will be effective after restarting the computer.

Installing LGRemote on a PC

LGRemote is a program for remote controlling a Lasergraph DSP using a PC running Windows® XP/Vista/7/8/8.1/10. It can be found on the Mark 2 memory stick in the directory "PC".

The latest version can be downloaded from our website under "Customer Area":

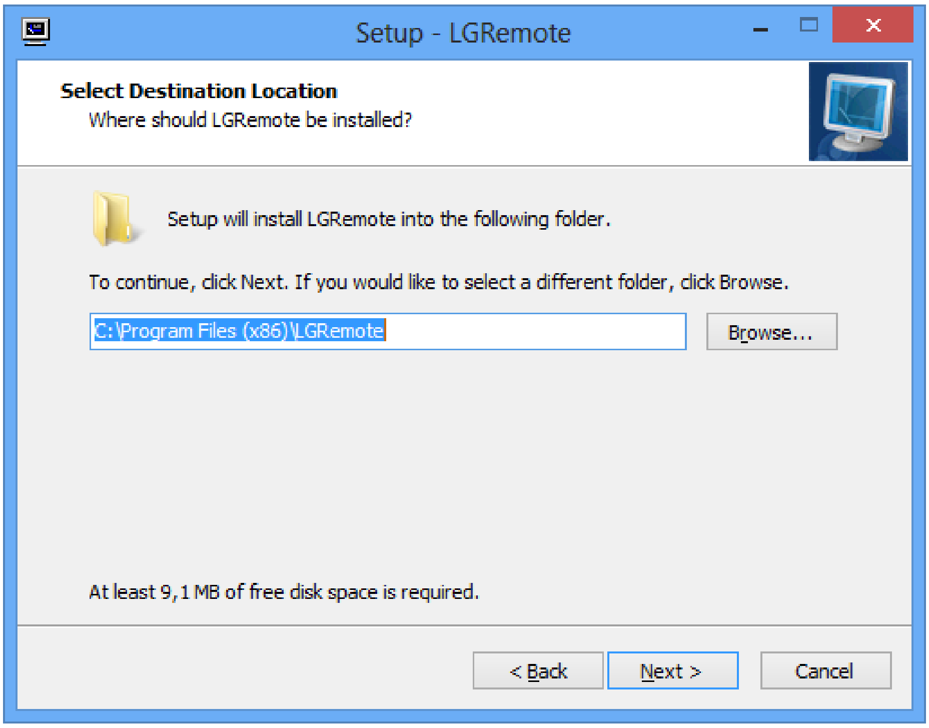
http://www.laseranimation.com

To install LGRemote double-click the installer "LGRemote--yyyy-mm-dd.exe" (yyyy-mm-dd is the release date).

You will be guided through the installation process.

When the greeting screen has appeared click on *Next*.

If you agree with the suggested folder for installation click *Next*, otherwise click *Browse* and select another folder.



Select Destination Location
Where should LGRemote be installed?

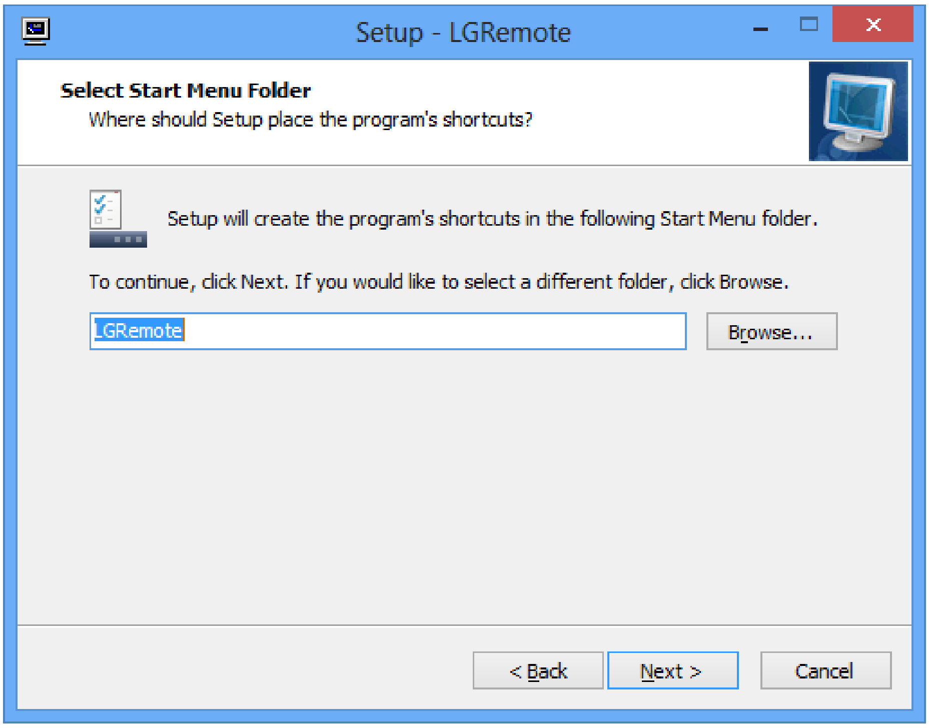
Setup will install LGRemote into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files (x86)\LGRemote

At least 9,1 MB of free disk space is required.

If you agree with the suggested Start Menu folder for the program's shortcuts click *Next*, otherwise select another folder from the list or enter a folder name.



Setup - LGRemote

Select Start Menu Folder

Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

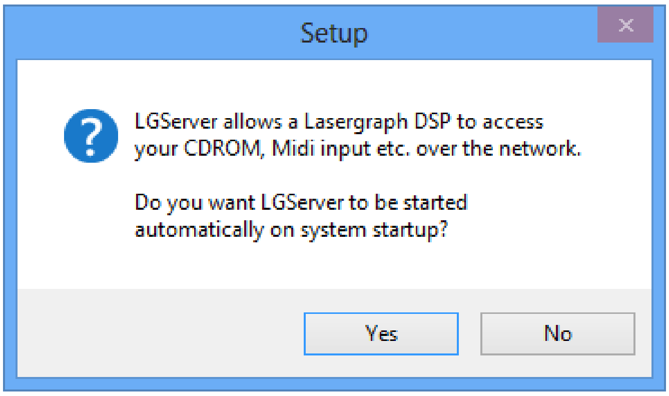
To continue, click Next. If you would like to select a different folder, click Browse.

LGRemote

Now a summary of the selected settings will appear. If everything is in order, click *Install*, otherwise click *Back* and correct the settings.
Afterwards the necessary files will be copied onto the hard drive.

Decide whether the program LGServer is always to be started automatically or not.

The LGServer is used to make peripheral devices connected to your PC, such as CD-ROM, MIDI keyboard, joystick, Faderboard, graphics tablet, available to the Lasergraph DSPs in the network.



Setup

LGServer allows a Lasergraph DSP to access your CDROM, Midi input etc. over the network.

Do you want LGServer to be started automatically on system startup?

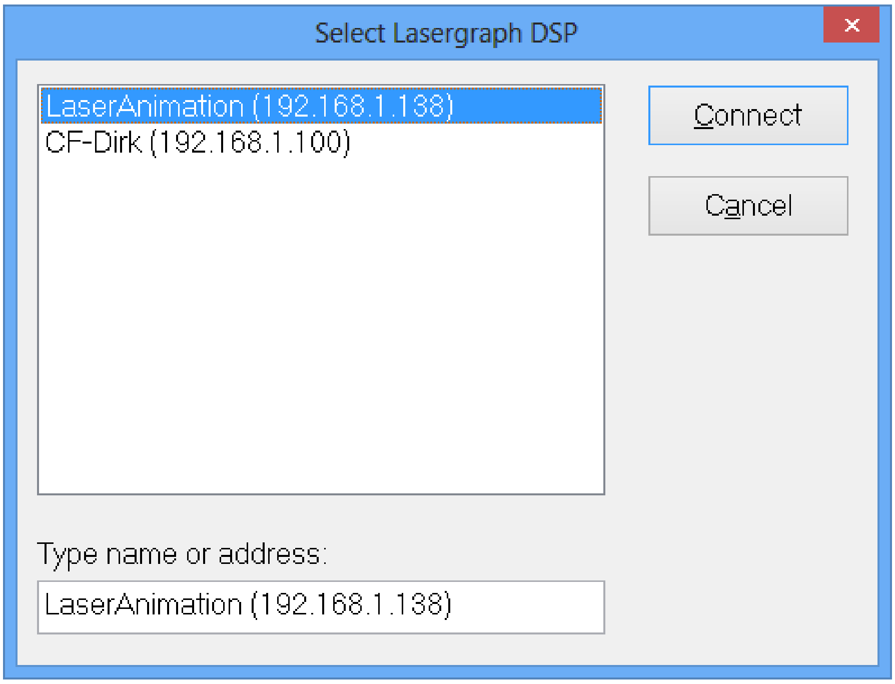
When the installation is completed, click on *Finish* to end the setup program.

Working with the Network

Remote Controlling a Lasergraph DSP by LGRemote

Start the LGRemote program either using the Start menu or by double clicking on the desktop shortcut.

LGRemote automatically searches the network for connected Lasergraph DSPs and lists their network names and IP addresses in a window.

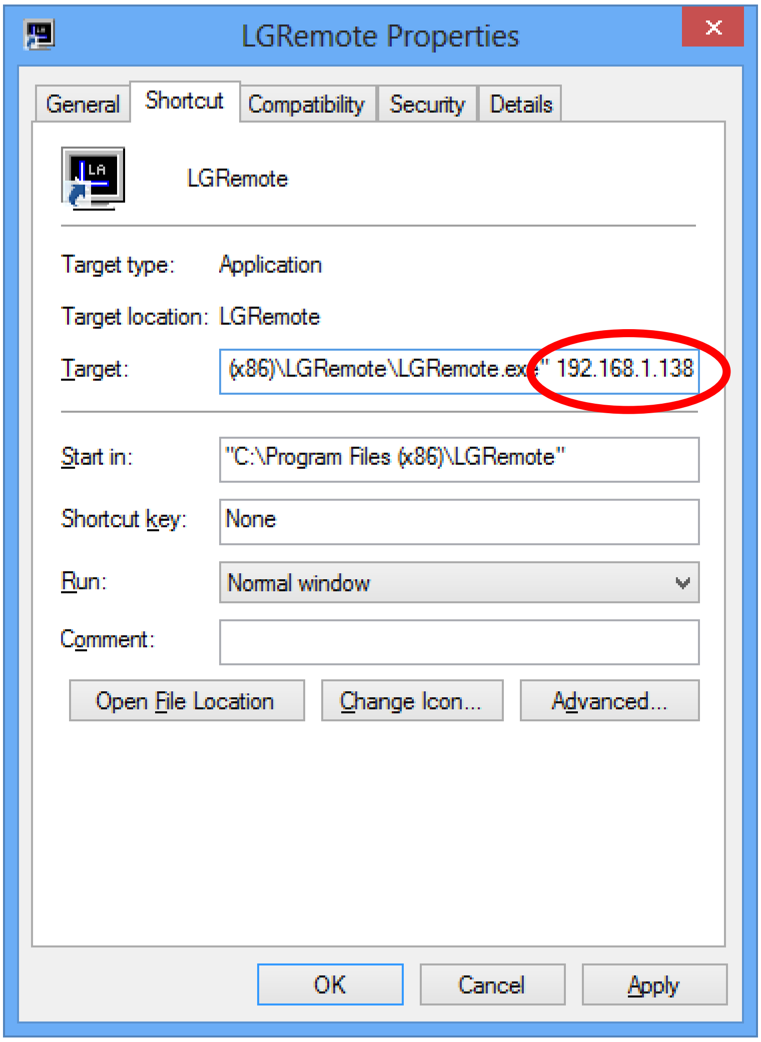


LaserAnimation (192.168.1.138)
CF-Dirk (192.168.1.100)

Type name or address: LaserAnimation (192.168.1.138)

Select the desired Lasergraph DSP and click on *Connect*, or simply double click on the name of the Lasergraph DSP.

It is possible to create a shortcut which links directly to a certain Lasergraph DSP.
To do so, just add the IP address or the network name of the DSP to the Target field of the shortcut. If you work with more than one Lasergraph DSP, this makes accessing the different devices more easy.


LGRemote Properties

LGRemote

Target type: Application

Target location: LGRemote

Target: (x86)\LGRemote\LGRemote.exe" 192.168.1.138

Start in: "C:\Program Files (x86)\LGRemote"

Shortcut key: None

Run: Normal window

Comment:

Notes:

- The key combination <Alt> + <F1> in LGRemote opens the selection window.
  A new connection can be initiated. (The previous one will be terminated.)
  If you want to remote control more than one Lasergraph DSP at the same time, start the program LGRemote several times on your PC.
- By using the key combination <Alt> + <TAB> you can switch back and forth on your PC between the different remote controlled Lasergraph DSPs.
- The key combination <Alt> + <Return> turns the full screen mode on or off.

Working with Files in a Network

The computers acting in a network (Lasergraph DSPs, PCs) are divided into **servers** and **clients**.

A **server** makes its resources, e.g. a hard drive partition D:\, a directory D:\Shows\ or the CD-ROM drive available to other computers in the network.
For this a share has to be defined. For security reasons this sharing has to be allowed explicitly. This has to be done on the computer which makes something available and therefore is the server for a specific drive or a specific file structure.

On a Lasergraph DSP this is done in the *Setup* in the submenu *File System Server Setup (FS Server)*, on a PC it is done under Windows® with the option "Sharing and Security..." which can be found in the context menu when you click on a drive or directory with the right mouse button*.*

A **client** uses outside resources like local resources, e.g. a hard disk drive is used as if it were installed in the client when in fact it is located in another computer.
The use of outside resources has to be defined on the computer which is supposed to use them. On the Lasergraph DSP this is done in the *Setup* in the submenu *File System Client Setup (FS Client).*

PCs under Windows® XP/Vista/7/8/8.1/10 cannot use resources (hard disk or CD-ROM drives) of Lasergraph DSPs, a drive on a Lasergraph DSP can only be accessed by another Lasergraph DSP.

Which computers in a network should be used as server and which as client?

- Master computers in a complex network and computers used to control other computers (e.g. Lasergraph DSP *compact*) should be defined as **servers**.
- Subcomputers or computers using data from servers or devices connected to servers then are **clients**.

Working with Files Between Several Lasergraph DSPs

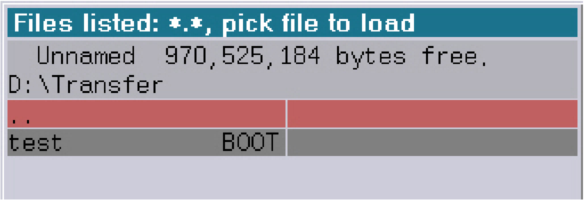
A show named "test" was created on a Lasergraph DSP (e.g. Lasergraph DSP *workstation*) with the network name Alpha. This show is now to be copied onto another Lasergraph DSP (e.g. Lasergraph DSP *compact*) with the network name Beta because this Lasergraph DSP is then supposed to serve as a stand alone device, for instance for playback.

The Lasergraph DSP on which the show was created is defined as server.
You can decide if you want to make the entire drive available, or just the directory in which the show is stored, e.g. D:\Shows.

It is also possible to create a special directory for these transfer purposes, e.g. "Transfer", which is then used as a kind of mailbox.

What is the process?

First the directory, for instance "Transfer", is created on a local drive of the Lasergraph DSP which is to be declared as server (*Alpha*) and the file(s) which is (are) supposed to be available in the network is (are) stored in this directory.

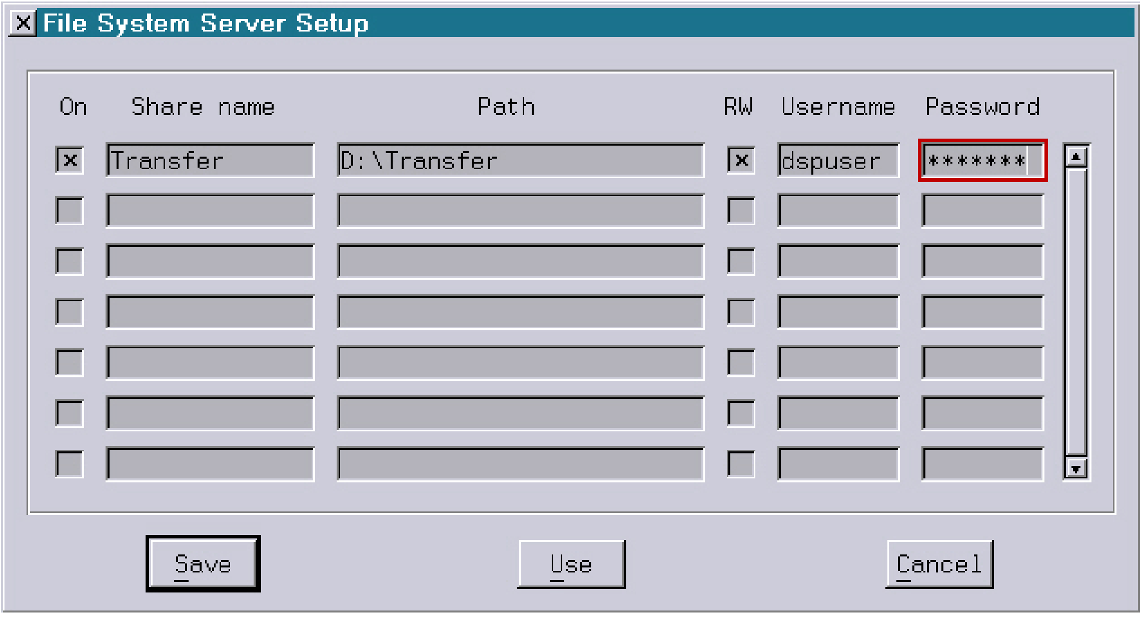


Files listed: *.*, pick file to load

Unnamed 970,525,184 bytes free.
D:\Transfer

```
..
test            BOOT
```

Next, to specify the server settings call up the File System Server Setup ("FS Server") on the Lasergraph DSP named *Alpha*:



File System Server Setup

On	Share name	Path	RW	Username	Password
☒	Transfer	D:\Transfer	☒	dspuser	*******
☐			☐		
☐			☐		
☐			☐		
☐			☐		
☐			☐		
☐			☐		

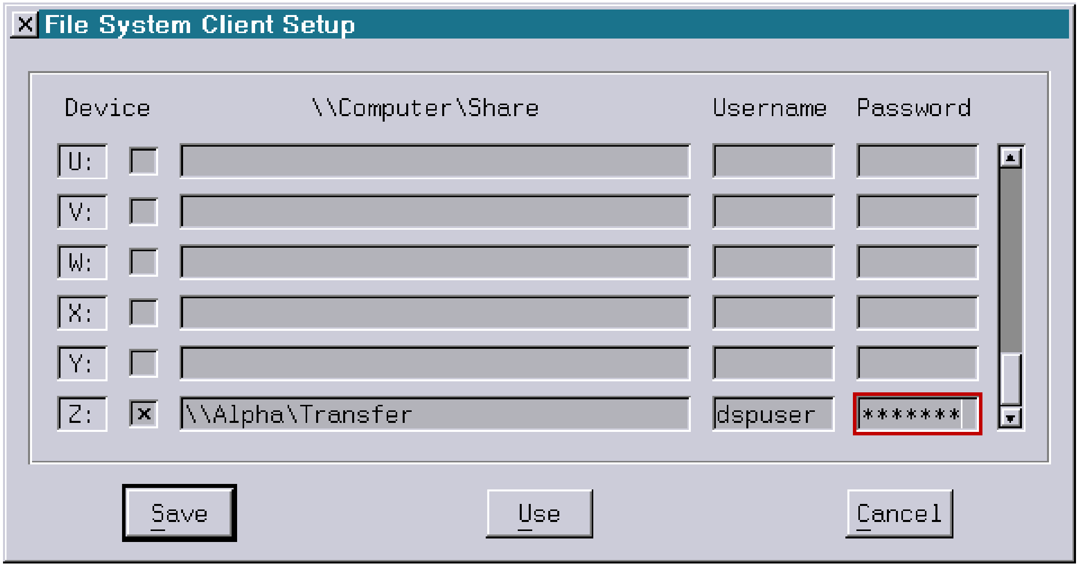
Follow these steps:

1. Activate the checkbox on the left.
2. Enter the name of the share in the field `Share name`, in this example it is called "Transfer".
3. Enter the path for which the share is declared in the column `Path`.
4. Use the checkbox `RW` (Read/Write) to define the permission for the share: checked = permission to read and write, unchecked = permission to read only.
5. Enter user name and password if desired.
6. Save entries - for this there are three options available:

Save	This setting is standard from now on.
Use	This setting is used until the Lasergraph DSP is turned off or restarted.
Cancel	All changes are discarded and the previous settings are restored.

Next the directory "Transfer" on the computer *Alpha* (server) is integrated on the computer *Beta* (client) by assigning it to a logical drive of the computer *Beta*.

For this the File System Client Setup ("FS Client") is called up on the Lasergraph DSP *Beta*:



File System Client Setup

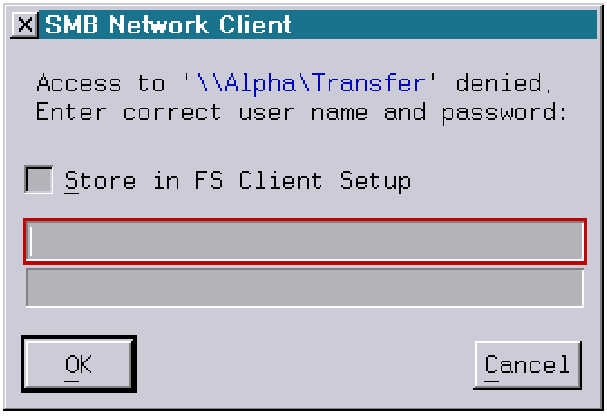
Device		\\Computer\Share	Username	Password
U:	□			
V:	□			
W:	□			
X:	□			
Y:	□			
Z:	☒	\\Alpha\Transfer	dspuser	*******

1. First select the logical drive (A to Z) to which the directory is to be assigned by checking the corresponding checkbox in the column `Device`, in this example it is "Z:".
   When the checkbox is deactivated, the assignment is cancelled.
2. In the column `\\Computer\Share` enter the server with its network name (Alpha) and the name of the share (Transfer).
3. Enter user name and password if necessary.

Entering a password and a user name is optional.

The entries in the FS Client Setup have to be identical to those of the share in the FS Server Setup of the computer containing the drive/file which is to be accessed.

If this is not the case or if the corresponding fields in the FS Client Setup were left empty, the following window will appear:



SMB Network Client

Access to '\\Alpha\Transfer' denied.
Enter correct user name and password:

□ Store in FS Client Setup

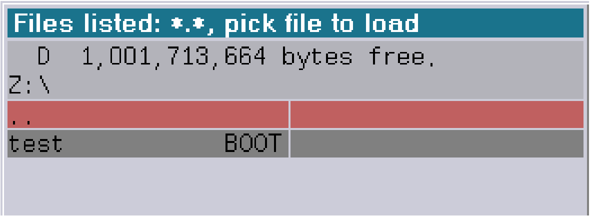
Enter the user name in the upper line and the password in the lower line.

If `Store in FS Client Setup` is activated, these data will automatically be entered in the setup and will not be requested again next time.

Click *OK* to connect or *Cancel* to abort the action.

If you now change to the drive Z:\ on the computer Beta, the contents of the directory "Transfer" of the computer Alpha will be displayed as if they actually existed on the computer Beta.

Now files from this directory can be loaded or copied to local drives of computer Beta.



Files listed: *.*, pick file to load

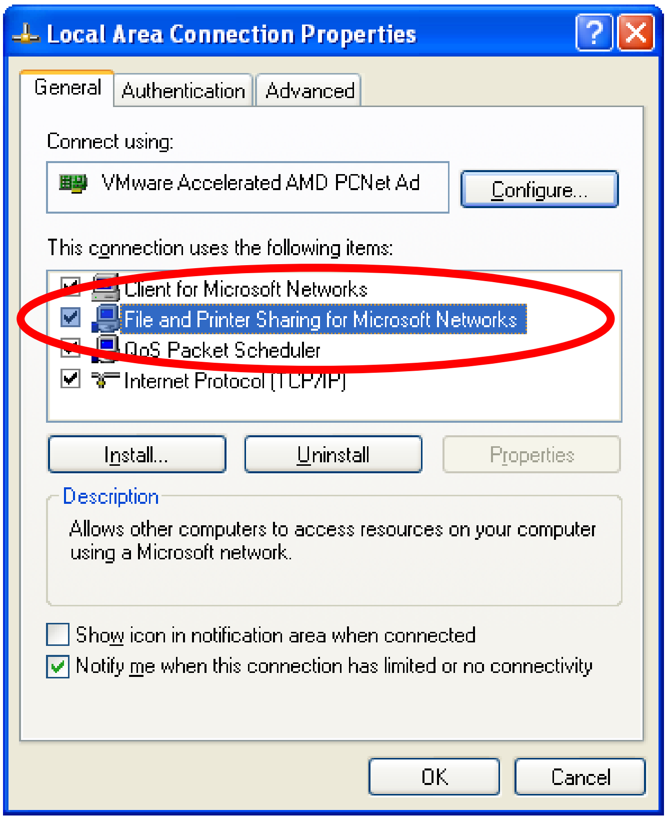
```
  D  1,001,713,664 bytes free.
Z:\
..
test                BOOT
```

Working with Files Between PC and Lasergraph DSP

Accessing Windows® XP Shares

To share files two settings are important:

1) The *File and Printer Sharing for Microsoft Networks* has to be enabled:



Local Area Connection Properties

Connect using:

VMware Accelerated AMD PCNet Ad

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ Internet Protocol (TCP/IP)

Description

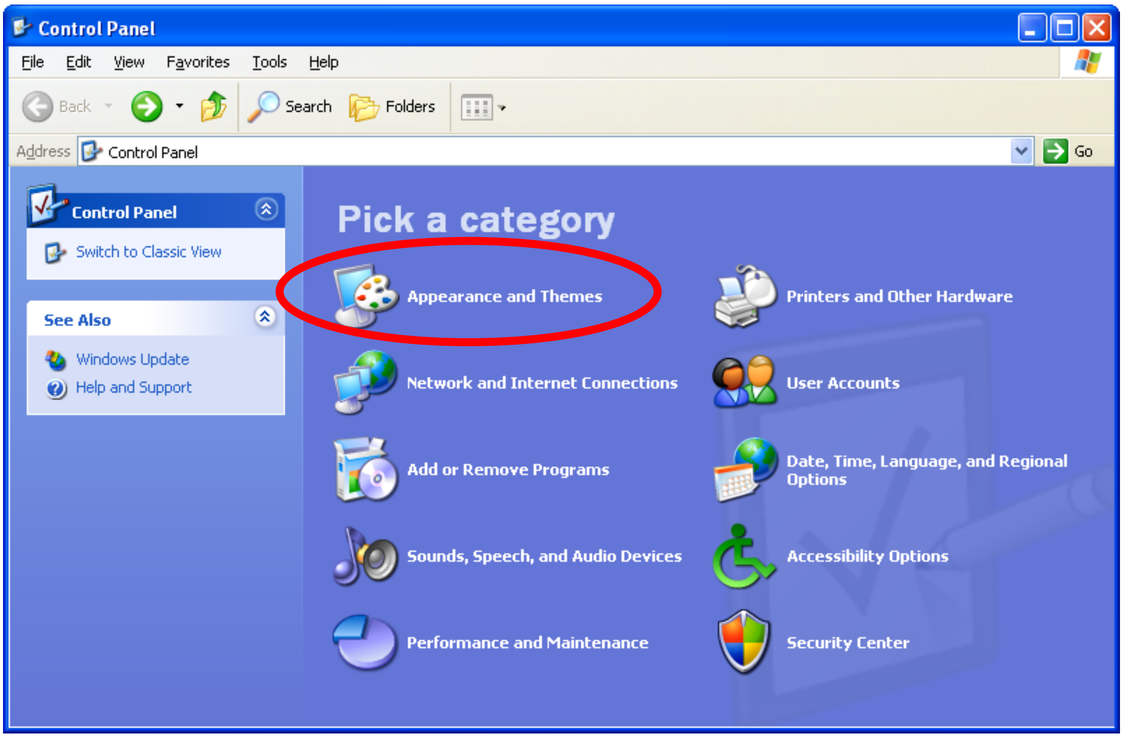
Allows other computers to access resources on your computer using a Microsoft network.

□ Show icon in notification area when connected

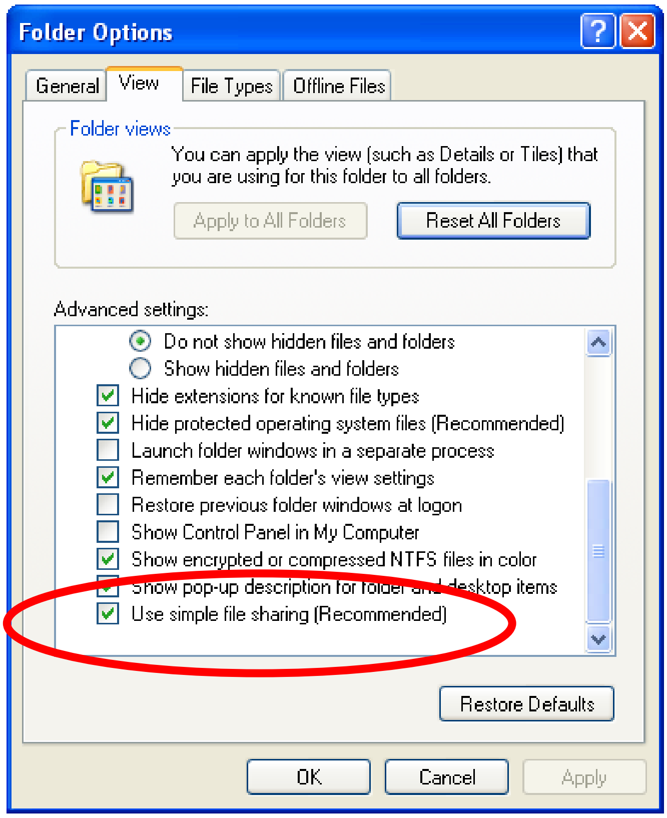
☑ Notify me when this connection has limited or no connectivity

You can access the properties of the "Local Area Connection" through the control panel.

2) If using Windows® XP Professional, simple file sharing should be enabled.



You can access the folder options through the control panel.



You can apply the view (such as Details or Tiles) that you are using for this folder to all folders.

Advanced settings:

- ● Do not show hidden files and folders
- ○ Show hidden files and folders
- ☑ Hide extensions for known file types
- ☑ Hide protected operating system files (Recommended)
- □ Launch folder windows in a separate process
- ☑ Remember each folder's view settings
- □ Restore previous folder windows at logon
- □ Show Control Panel in My Computer
- ☑ Show encrypted or compressed NTFS files in color
- ☑ Show pop-up description for folder and desktop items
- ☑ Use simple file sharing (Recommended)

Then make sure that the "Use simple file sharing" is enabled.

What is the process?

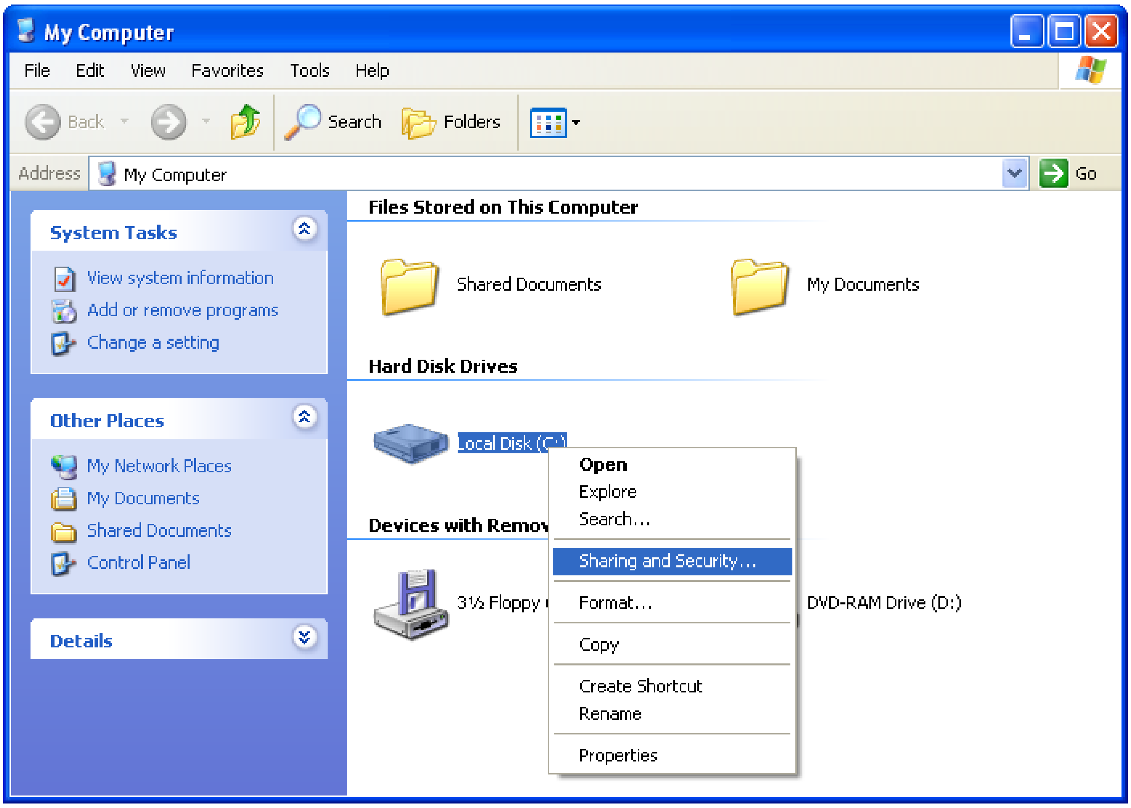
A PC serves as data storage medium for the connected Lasergraph DSPs.

A production company for instance has stored all shows and pictures on a stationary PC which never leaves the office.
The Lasergraph DSP is used for temporary installations. Only the data necessary for the specific job have to be copied to this computer. The valuable company treasure of shows and pictures does not have to be carried along, therefore the risk of loss or theft is low.
Nevertheless, in the office working with the data is as easy as if they were all stored on the Lasergraph DSP.

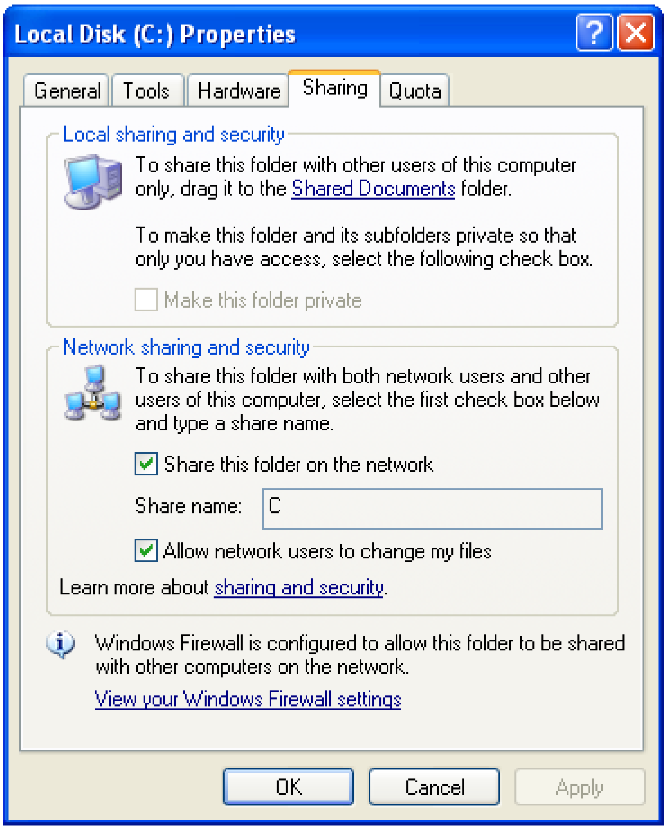
In this case the PC works as the server.
On it the data structures which are to be used by the Lasergraph DSP (client) are created. For instance a folder "LGDSP" with a subdirectory "Shows" containing all shows is set up on the hard disk or a partition of the PC.
This PC directory has to be shared for the network under Windows®. The sharing for drives and directories is done under Windows® as follows:

Click on your icon for *My Computer* and click with the **right mouse button** on the icon for the hard disk, e.g. "C:".

Click on "Sharing and Security..."



and the *Properties* window will open.



Local sharing and security

To share this folder with other users of this computer only, drag it to the Shared Documents folder.

To make this folder and its subfolders private so that only you have access, select the following check box.

□ Make this folder private

Network sharing and security

To share this folder with both network users and other users of this computer, select the first check box below and type a share name.

☑ Share this folder on the network

Share name: C

☑ Allow network users to change my files

Learn more about sharing and security.

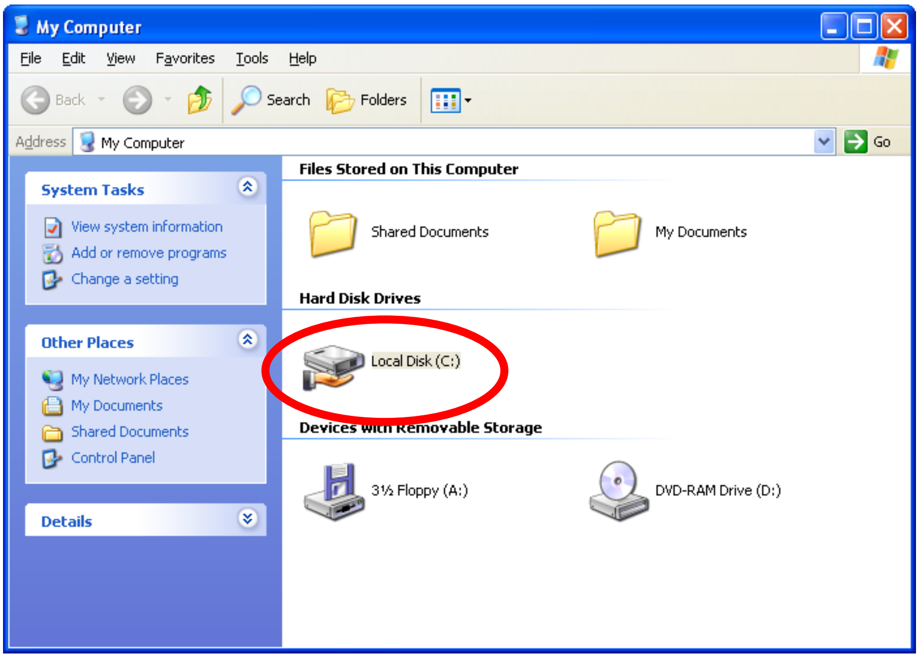
Windows Firewall is configured to allow this folder to be shared with other computers on the network.

View your Windows Firewall settings

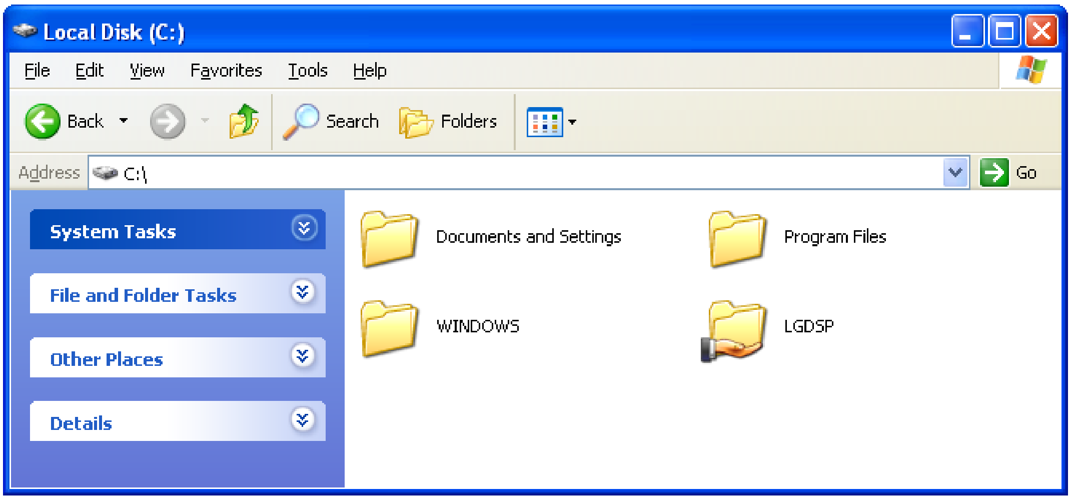
Select *Share this folder on the network* in order to share this drive for network access.

Enter a name for this drive in the field *Share name*.

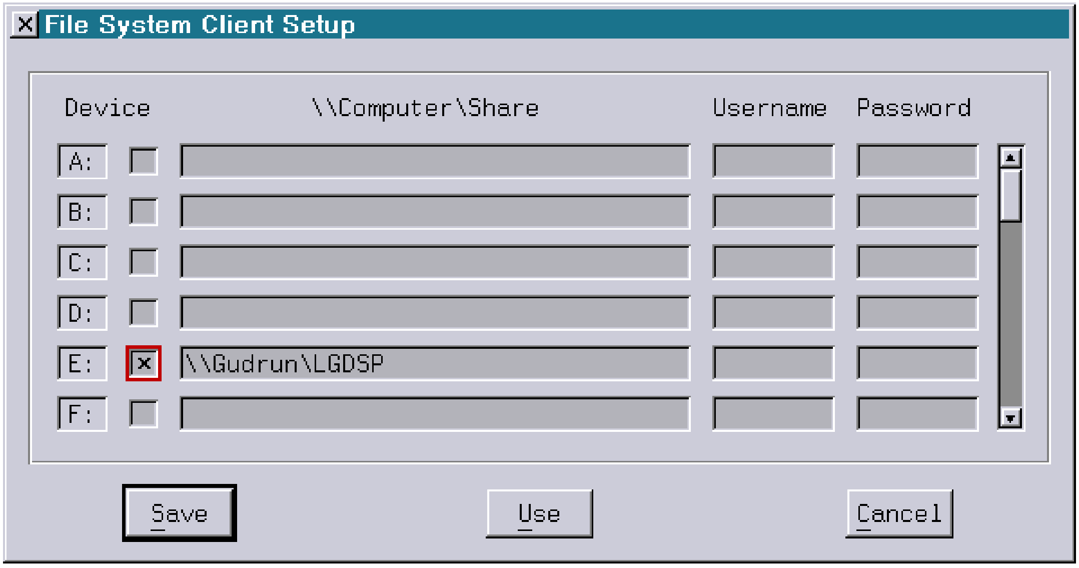
The icon of the drive now looks different to make it obvious that it is a shared drive.



The same procedure applies to directories, for instance to the created directory "LGDSP":



After defining the share on the PC the directory can be integrated on the Lasergraph DSP. For this call up the "File System Client Setup" and assign the PC directory to a logical drive on the Lasergraph DSP.

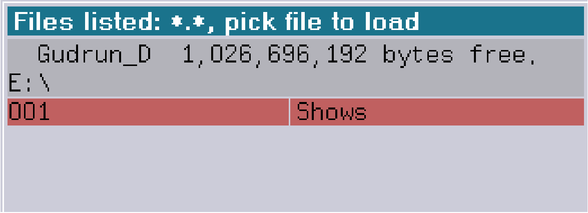


File System Client Setup

Device		\\Computer\Share	Username	Password
A:	□			
B:	□			
C:	□			
D:	□			
E:	☒	\\Gudrun\LGDSP		
F:	□			

In this example the logical drive "E:" is selected, and the network name of the PC "Gudrun" and the defined share "LGDSP" are entered in the column `\\Computer\Share`.

When you change to drive E:\ in the file system on the Lasergraph DSP, the contents of the directory "LGDSP" will be displayed as if it were present on the Lasergraph DSP:



```
Files listed: *.*, pick file to load
 Gudrun_D  1,026,696,192 bytes free.
E:\
001                    Shows
```

In our example the shows could now be loaded and saved as usual or copied to the local hard disk of the Lasergraph DSP if necessary.

Similarly a USB or CD-ROM drive of a PC can be integrated with a Lasergraph DSP.
The Lasergraph DSP can recognize the data stored on an USB memory stick or CD in a format readable by the PC.

Note:

The DSP file system supports directory and file names with 14 characters and 4 character extensions.

Additionally directory and file names must not contain special characters like space, '\', '/', ':' or '.'.

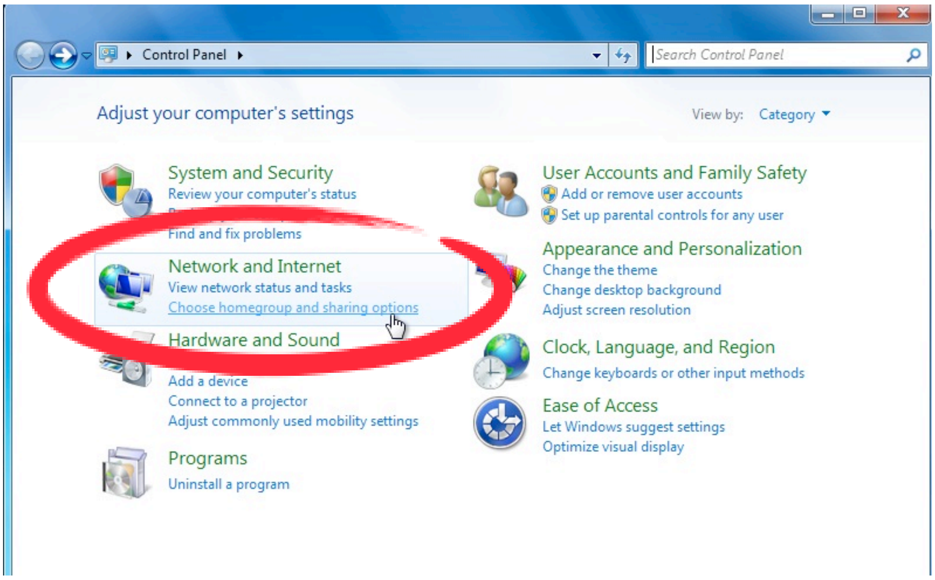
If a filename used by Windows® does not match the DSP name convention the DSP will show the so called 8.3 filename (short filename) which consist of 8 characters and a 3 character extension.

Accessing Windows® Vista/7 Shares

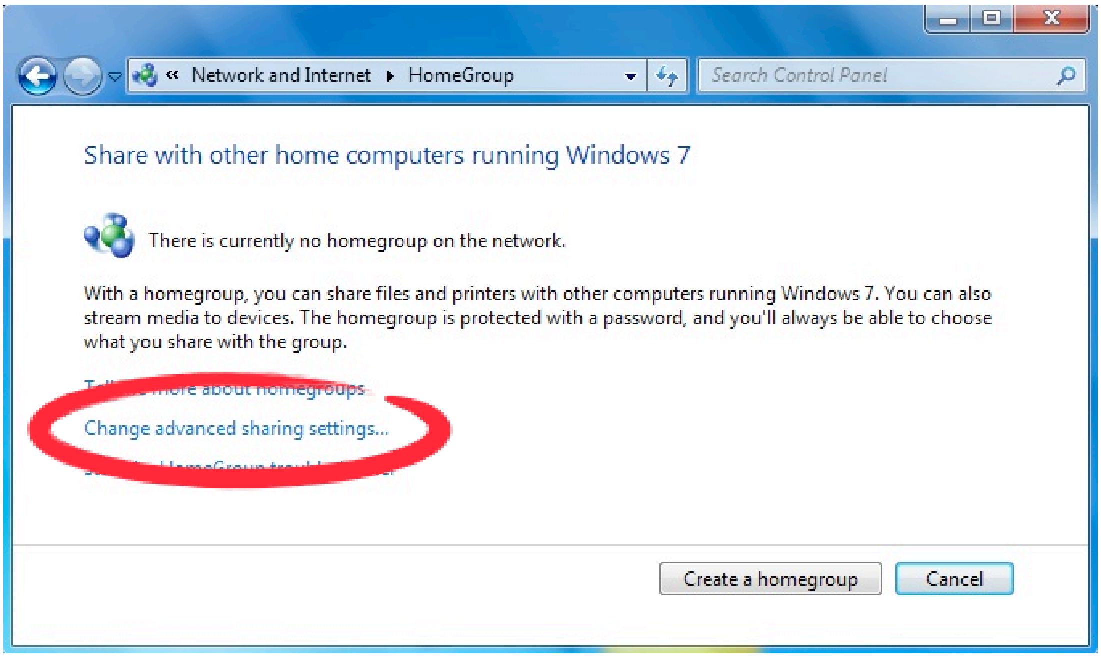
The Lasergraph DSP cannot access password protected shares on PCs running Windows® Vista or later Windows® versions.

If you use Windows® 7 as file server for the Lasergraph DSP it is necessary to adjust some special settings in the Windows® 7 sharing settings.

Provided that you already have a connection established with LGRemote, open the control panel and click on "Choose homegroup and sharing options".


Control Panel
Search Control Panel
Adjust your computer's settings
View by: Category
System and Security
Review your computer's status
Find and fix problems
Network and Internet
View network status and tasks
Choose homegroup and sharing options
Hardware and Sound
Add a device
Connect to a projector
Adjust commonly used mobility settings
Programs
Uninstall a program
User Accounts and Family Safety
Add or remove user accounts
Set up parental controls for any user
Appearance and Personalization
Change the theme
Change desktop background
Adjust screen resolution
Clock, Language, and Region
Change keyboards or other input methods
Ease of Access
Let Windows suggest settings
Optimize visual display

This will open the following window where you have to click on "Change advanced settings...":



Share with other home computers running Windows 7

There is currently no homegroup on the network.

With a homegroup, you can share files and printers with other computers running Windows 7. You can also stream media to devices. The homegroup is protected with a password, and you'll always be able to choose what you share with the group.

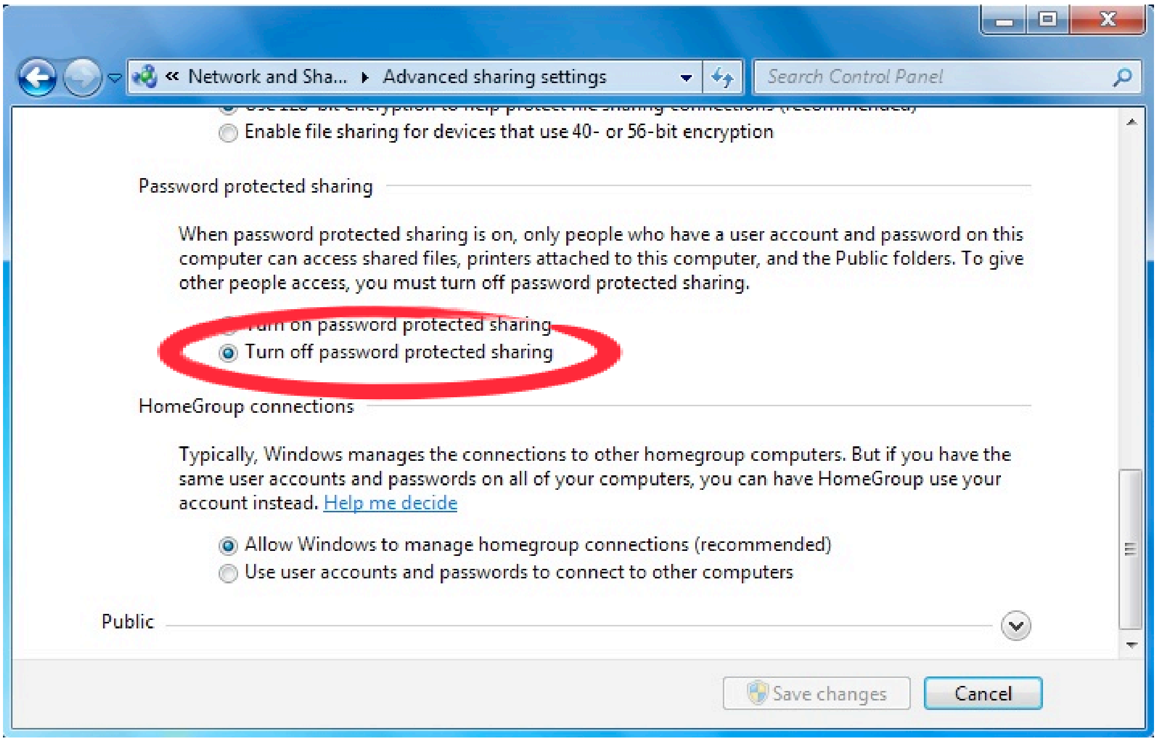
Tell me more about homegroups

Change advanced sharing settings...

Start the HomeGroup troubleshooter

In the next window you can change the sharing options.

Scroll down to "Password protected sharing" and activate "Turn off password protected sharing".



○ Use 128-bit encryption to help protect file sharing connections (recommended)

○ Enable file sharing for devices that use 40- or 56-bit encryption

Password protected sharing

When password protected sharing is on, only people who have a user account and password on this computer can access shared files, printers attached to this computer, and the Public folders. To give other people access, you must turn off password protected sharing.

○ Turn on password protected sharing

● Turn off password protected sharing

HomeGroup connections

Typically, Windows manages the connections to other homegroup computers. But if you have the same user accounts and passwords on all of your computers, you can have HomeGroup use your account instead. Help me decide

● Allow Windows to manage homegroup connections (recommended)

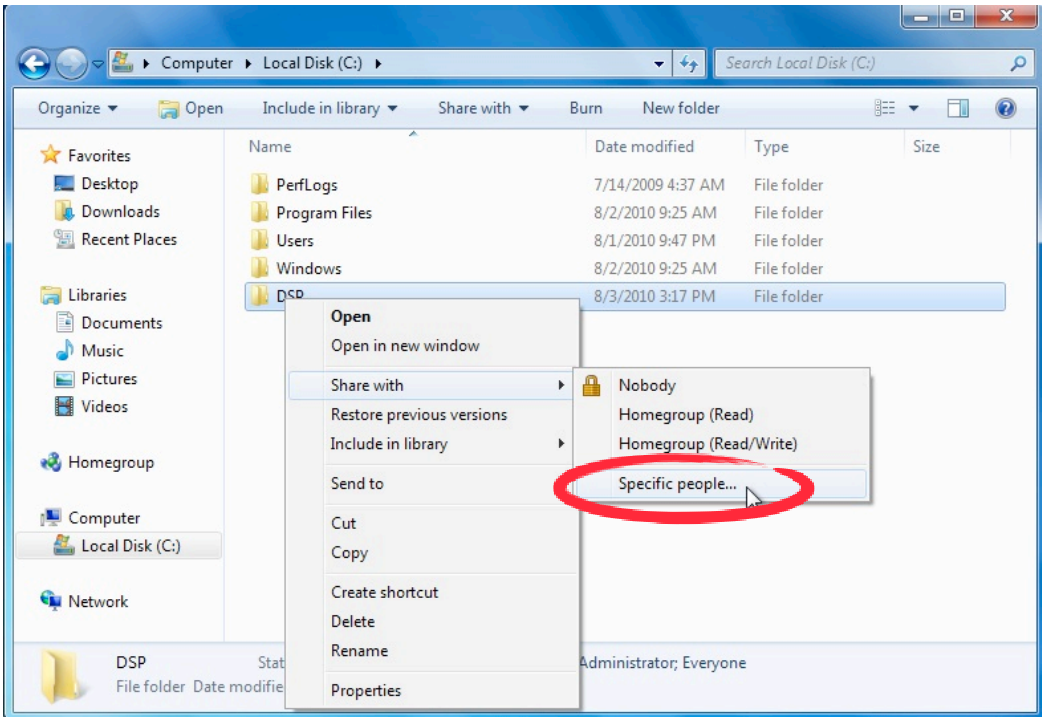
○ Use user accounts and passwords to connect to other computers

Public

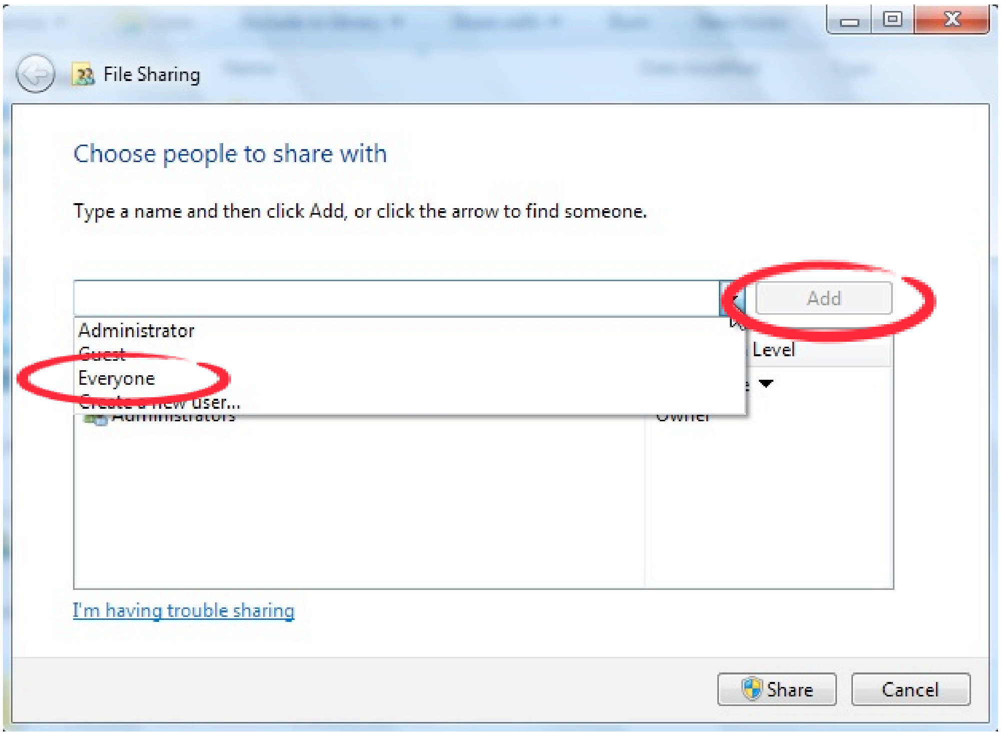
Close the control settings window and choose the directory you want to share.

Click on it with the right mouse button.

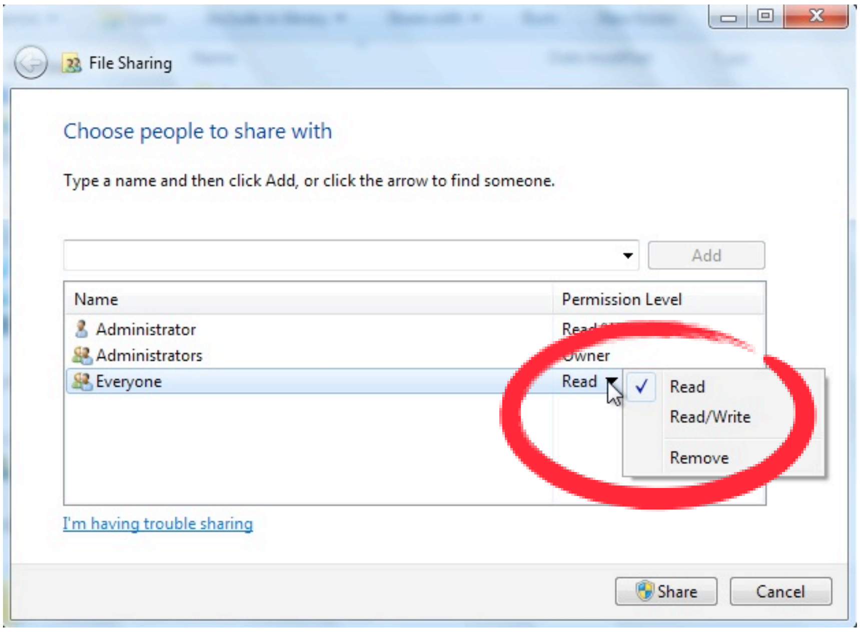
In the context menu which opens click on "Shared with" ⇨ "Specific people...":



Now choose the user "Everyone" next to the "Add"-button and click "Add".


File Sharing
Choose people to share with
Type a name and then click Add, or click the arrow to find someone.
Add
Administrator
Guest
Everyone
Create a new user...
Level
Owner
I'm having trouble sharing
Share
Cancel

Then change the permission of the user "Everyone" to **"Read/Write"**.
Confirm your changes by clicking on the "Share" button.



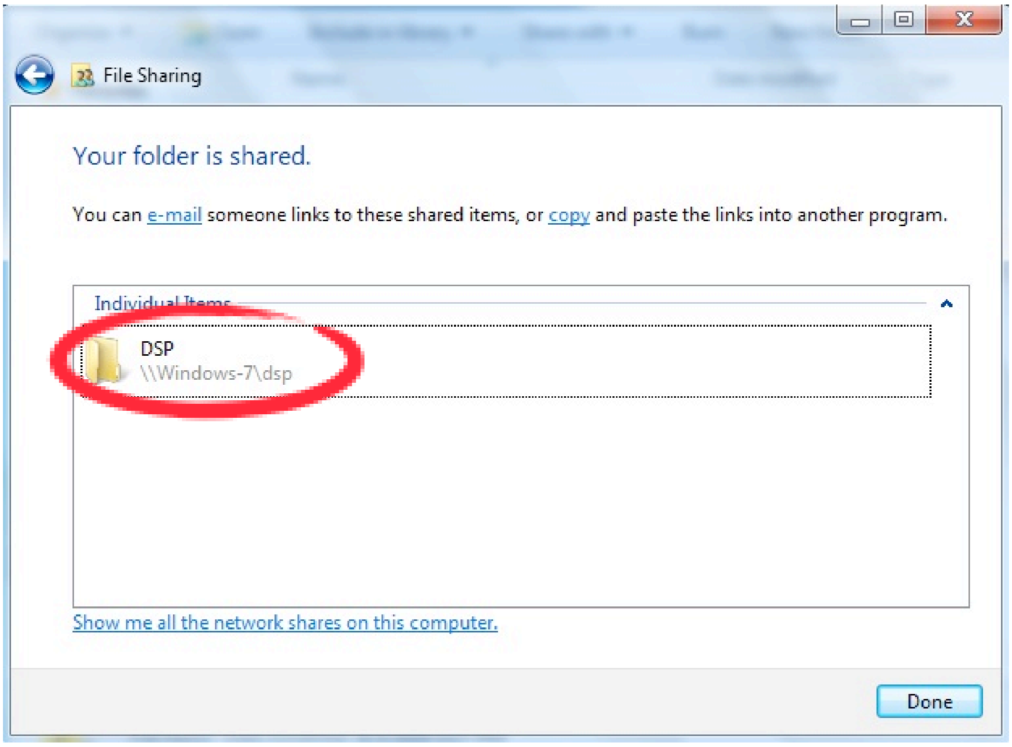
File Sharing

Choose people to share with

Type a name and then click Add, or click the arrow to find someone.

Name	Permission Level
Administrator	Read/Write
Administrators	Owner
Everyone	Read

The folder is now shared.



File Sharing

Your folder is shared.

You can e-mail someone links to these shared items, or copy and paste the links into another program.

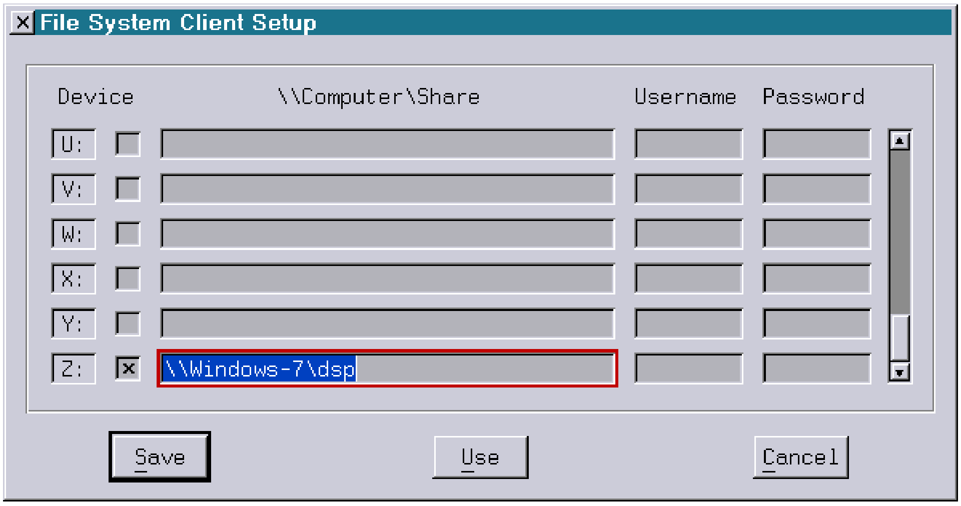
Individual Items

DSP
\\Windows-7\dsp

The folder "DSP" can now be assigned in the "File System Client Setup" of the Lasergraph DSP. Open the "Setup" ⇨ "FS Client" type the address in the column "\\Computer\Share" and assign it to the desired "Device" by enabling the check box.

Important Note: Do not use the IP-address of the Windows® - PC !!!
Only by using the computer name and the share name the file sharing will work!

Save your setting.

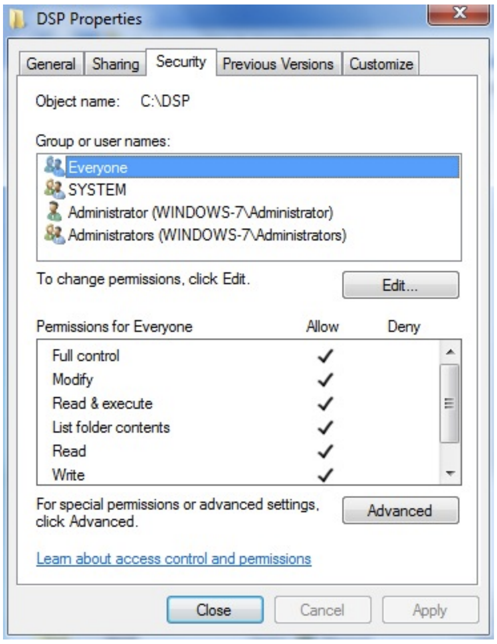


File System Client Setup

Device		\\Computer\Share	Username	Password
U:	☐			
V:	☐			
W:	☐			
X:	☐			
Y:	☐			
Z:	☒	\\Windows-7\dsp		

Now you can access the directory of the PC.

If you have problems to access the directory, check the security settings in the properties of the shared folder. There should be a user called "Everyone" with permission to read and write.
If not, add the user manually.



DSP Properties

Object name: C:\DSP

Group or user names:

- Everyone
- SYSTEM
- Administrator (WINDOWS-7\Administrator)
- Administrators (WINDOWS-7\Administrators)

To change permissions, click Edit.

Permissions for Everyone	Allow	Deny
Full control	✓	
Modify	✓	
Read & execute	✓	
List folder contents	✓	
Read	✓	
Write	✓	

For special permissions or advanced settings, click Advanced.

This might be necessary if you enable the sharing by opening the "advanced sharing options".

Working with Peripheral Devices in the Network

Different peripheral devices, for instance studio devices supporting SMPTE timecode, multimedia devices, MIDI devices, CD-ROM drives, Faderboards or Joysticks, can be used by several Lasergraph DSPs in a network, although they are only connected to one Lasergraph DSP or even a PC.

That means that in the network several computers react to, for instance, pressing a key of a MIDI keyboard, although it is only connected to one computer. The computer to whose MIDI input connector the MIDI keyboard is actually connected works as the server in the network (it provides the resources for the other computers in the network).

Peripheral devices connected to a PC under Windows® XP/Vista/7/8/8.1/10 can be used by Lasergraph DSPs in the network. However there are some limitations because a PC does not have as many input connectors as a Lasergraph DSP.
For instance a PC only has a MIDI input if it has a sound card installed or a MIDI interface is connected. With the program LGServer, which is automatically installed when the program LGRemote is installed, the use of devices connected to a PC is possible from the Lasergraph DSP.

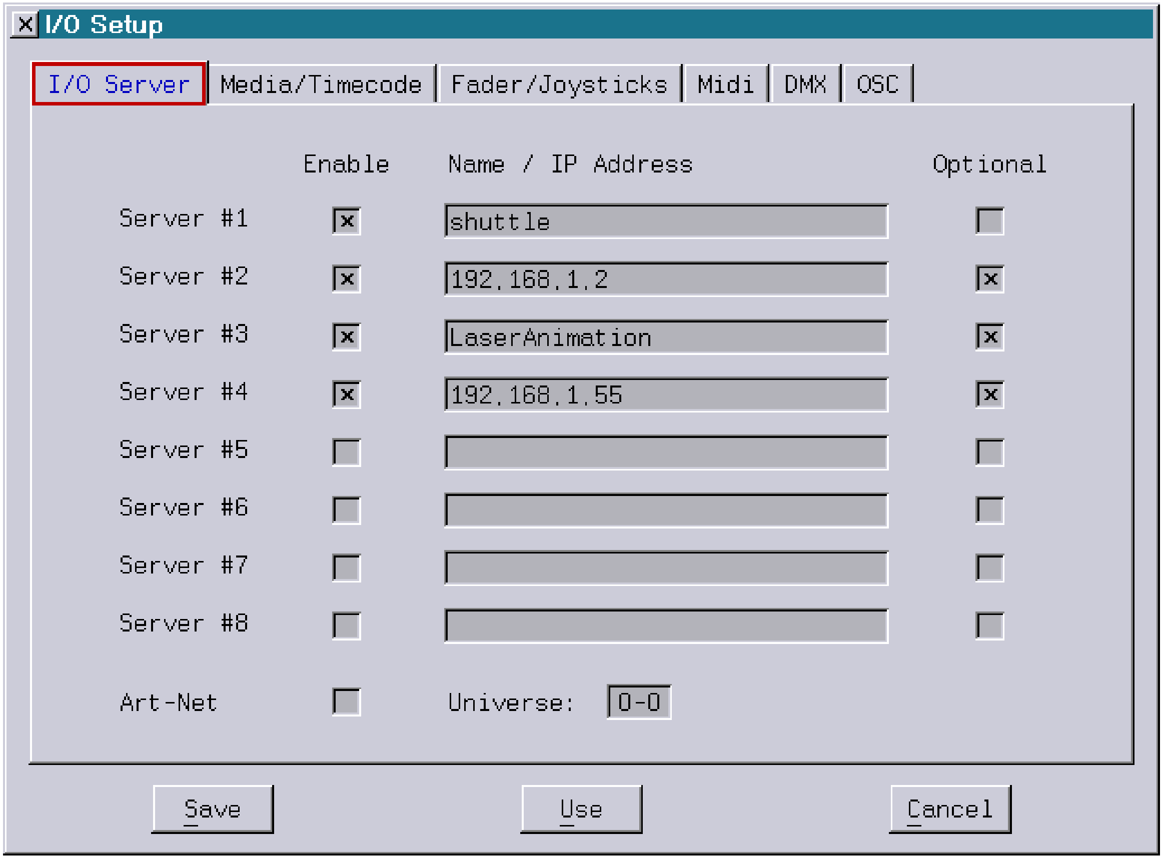
Those Lasergraph DSPs which are to use the resources of other Lasergraph DSPs or PCs in the network are the **clients**.
For each client it is necessary to define exactly which resources of which computer it is supposed to use. This configuration is done in the **I/O Setup** of the Lasergraph DSP. This way you define exactly which actions at which computers will be considered, thereby setting up the whole system optimally.

A Lasergraph DSP can be server and client at the same time. It can for instance provide the timecode present at the SMPTE input to other computers in the network and at the same time follow the movements of a joystick which in turn is connected to another computer.

I/O Setup

Every Lasergraph DSP (client) which is to use devices connected to other Lasergraph DSPs or PCs in the network has to be configured in the I/O Setup.
For this you select the button *Net I/O* in the **Setup of the respective client** and the I/O Setup window appears.

I/O Server

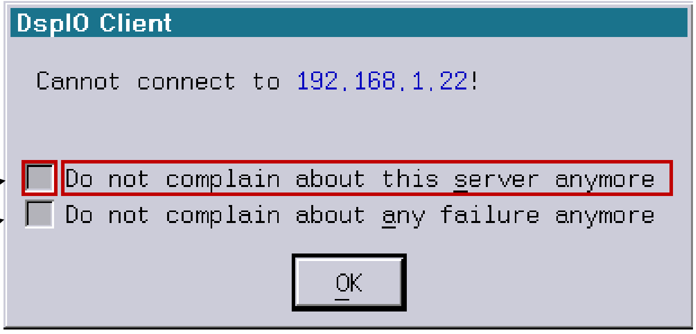


	Enable	Name / IP Address	Optional
Server #1	☒	shuttle	□
Server #2	☒	192.168.1.2	☒
Server #3	☒	LaserAnimation	☒
Server #4	☒	192.168.1.55	☒
Server #5	□		□
Server #6	□		□
Server #7	□		□
Server #8	□		□
Art-Net	□	Universe: 0-0	

Here up to eight servers (Lasergraph DSP or PC) are entered whose connected devices this computer is supposed to use over the network. That means that each Lasergraph DSP can, in addition to its own local devices, use additional devices which can be connected to up to eight other computers.

The corresponding computers (servers) are entered in the column `Name/IP Address` with their network name or IP address and activated with the checkbox `Enable`.
If servers are not always available in the network activate the checkbox `Optional`.
This prevents the warning notice (see following fig.) from appearing.
If `Optional` is not activated and defined servers are not found in the network (for instance because the computer is turned off) an error warning will appear.

Decide if further errors
of this specific server or
of all servers are to be ignored



DspIO Client

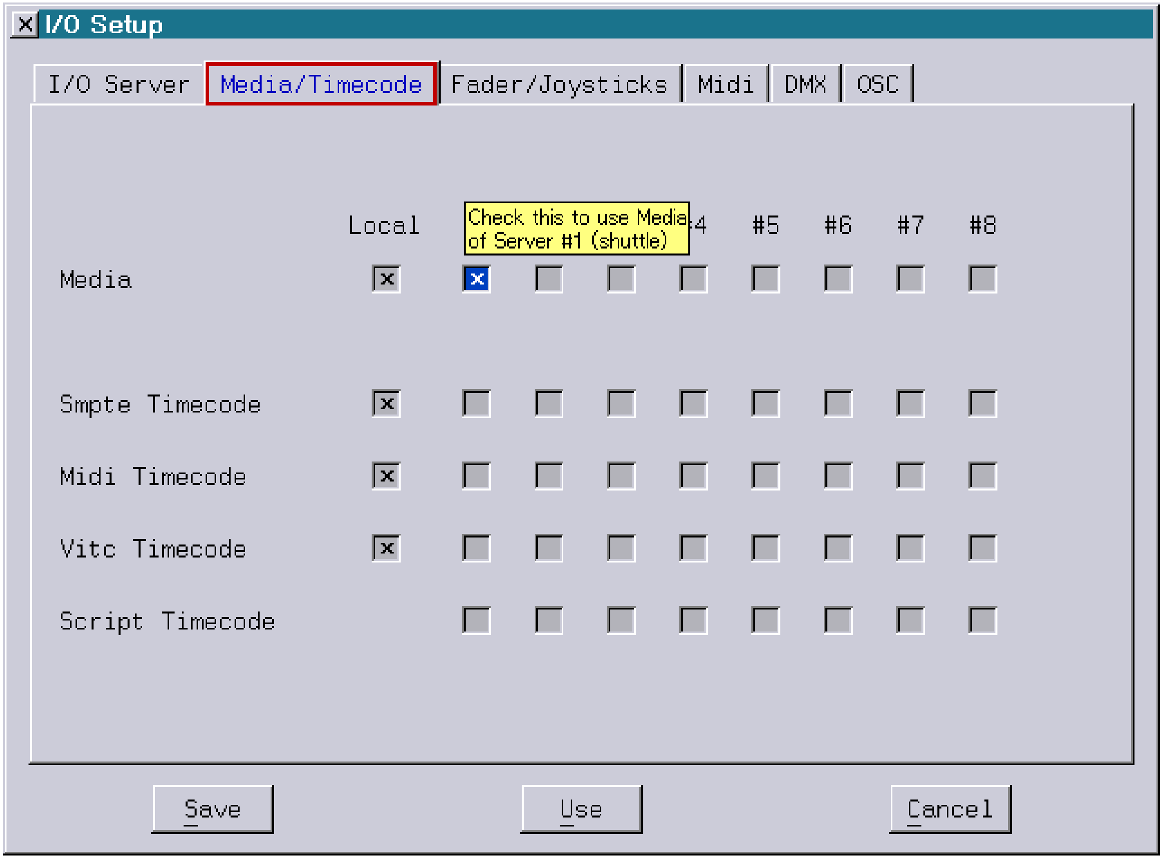
Cannot connect to 192.168.1.22!

□ Do not complain about this server anymore
□ Do not complain about any failure anymore

Now define for this client which devices, "Media/Timecode", "Fader/Joysticks", "MIDI", "DMX", of which servers (#1 to #8) are supposed to be used.

The checkboxes are used for assignment to the corresponding servers.
Local always means use of the device connected to the local Lasergraph DSP. In addition eight device sources #1 to #8, which correspond to the previously defined servers, are available.

Media/Timecode



I/O Setup

Check this to use Media of Server #1 (shuttle)

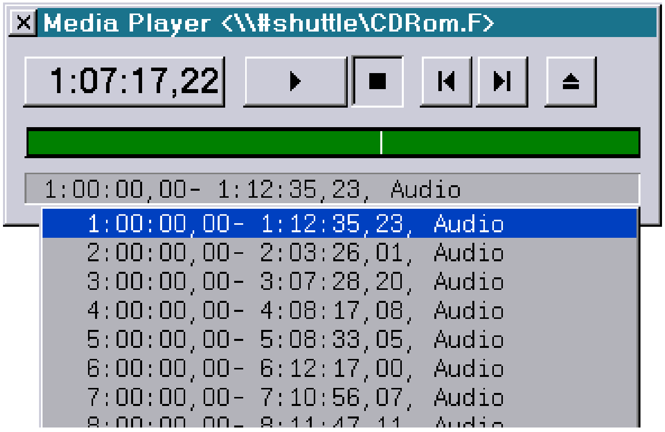
	Local				4	#5	#6	#7	#8
Media	☒	☒	□	□	□	□	□	□	□
Smpte Timecode	☒	□	□	□	□	□	□	□	□
Midi Timecode	☒	□	□	□	□	□	□	□	□
Vitc Timecode	☒	□	□	□	□	□	□	□	□
Script Timecode		□	□	□	□	□	□	□	□

This mask is used to define the servers whose CD-ROM drives (DSP or PC) or whose Media files (only PC) are to be used and which timecode information is to be used.

Example:
In a network consisting of several Lasergraph DSPs a CD-ROM drive is connected to one computer, for instance with the network name "shuttle". This contains a CD with audio tracks which another Lasergraph DSP (client) is supposed to be able to use.

On the Lasergraph DSP (client) the I/O server "shuttle" is entered in the I/O Setup, then "Media/Timecode" is selected and Media is assigned to the corresponding server, see fig. above.

After setting and saving the configuration, the Media Player of the Lasergraph DSP (client) starts and the CD can be used as if a CD-drive were actually connected to this Lasergraph DSP.



Media Player <\\#shuttle\CDRom.F>

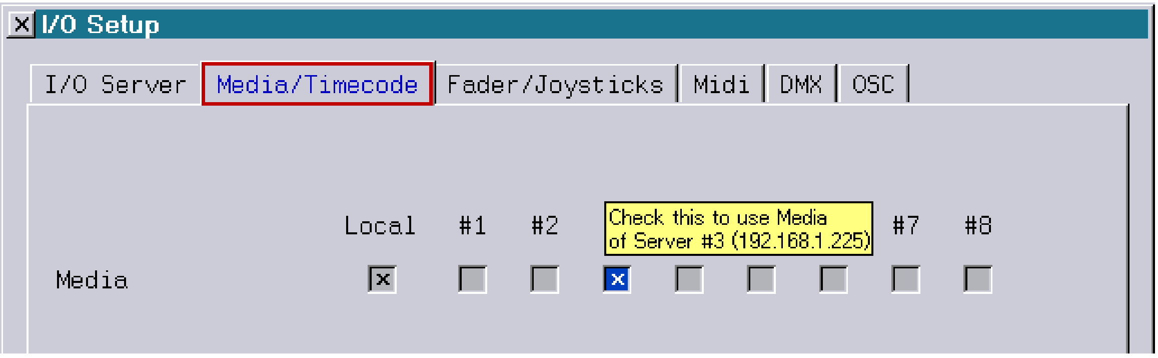
1:07:17,22

```
1:00:00,00- 1:12:35,23, Audio
  1:00:00,00- 1:12:35,23, Audio
  2:00:00,00- 2:03:26,01, Audio
  3:00:00,00- 3:07:28,20, Audio
  4:00:00,00- 4:08:17,08, Audio
  5:00:00,00- 5:08:33,05, Audio
  6:00:00,00- 6:12:17,00, Audio
  7:00:00,00- 7:10:56,07, Audio
  8:00:00,00- 8:11:47,11, Audio
```

The Media Player contains the network path - network name of the server "shuttle" and the CD-ROM drive with its name, for instance "F".
Because the access to a CD-ROM drive is internally different in the network from those to other drives, there is a double cross (‘#’) in front of the computer name.

CD-ROM Drive at the PC

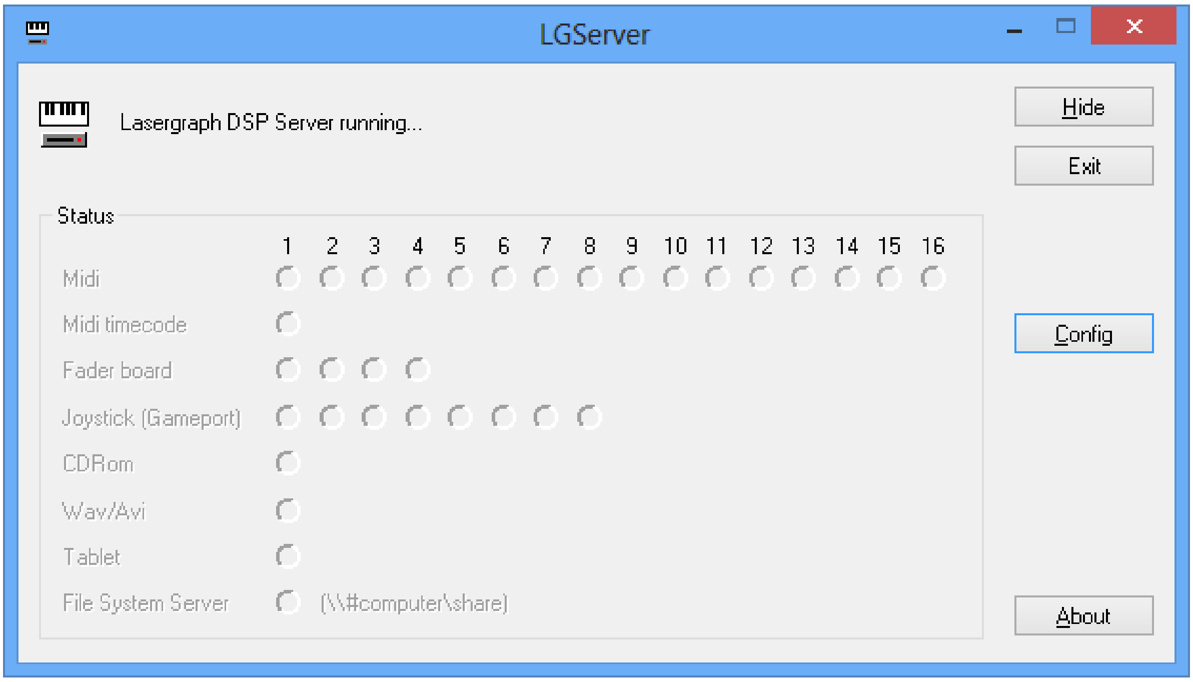
The CD-ROM drive of the PC containing a CD with audio tracks is supposed to be used by the Lasergraph DSP (client) over the network.
The PC (server) has to be entered on the Lasergraph DSP (client) as server, for example with the IP address 192.169.1.225, and "Media" has to be assigned to the corresponding server in the I/O Setup, see fig:



Check this to use Media of Server #3 (192.168.1.225)

	Local	#1	#2					#7	#8
Media	☒	□	□	☒	□	□	□	□	□

Then the LGServer has to be configured on the PC by clicking the icon in the task bar or if it is not running by starting it from the start menu and then by clicking the icon in the task bar.



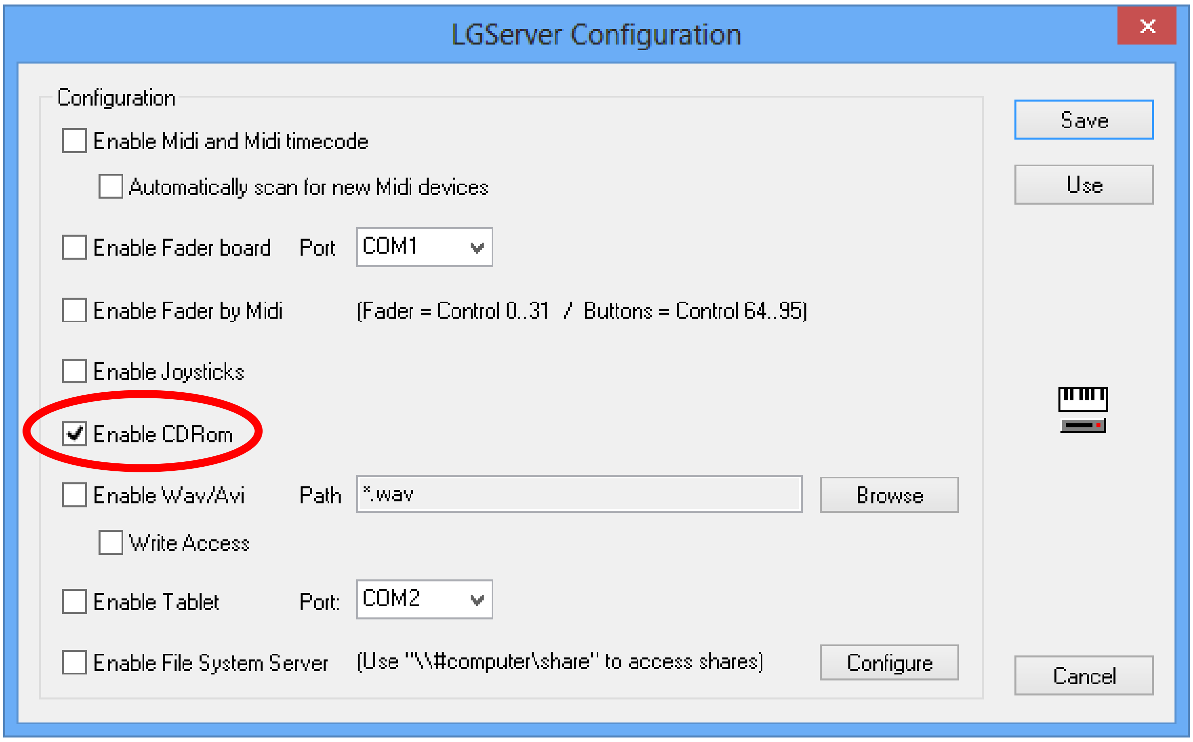
Lasergraph DSP Server running...

Status

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Midi timecode	○															
Fader board	○	○	○	○												
Joystick (Gameport)	○	○	○	○	○	○	○	○								
CDRom	○															
Wav/Avi	○															
Tablet	○															
File System Server	○	(\\#computer\share)														

Status lists the devices which can be connected to a PC.
Hide closes the window of the LGServer, the LGServer keeps running.
Exit quits the LGServer.
Config opens the LGServer configuration window.

After starting the LGServer click the button "Config" to open the configuration window:


LGServer Configuration

Configuration

☐ Enable Midi and Midi timecode

☐ Automatically scan for new Midi devices

☐ Enable Fader board Port COM1

☐ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)

☐ Enable Joysticks

☑ Enable CDRom

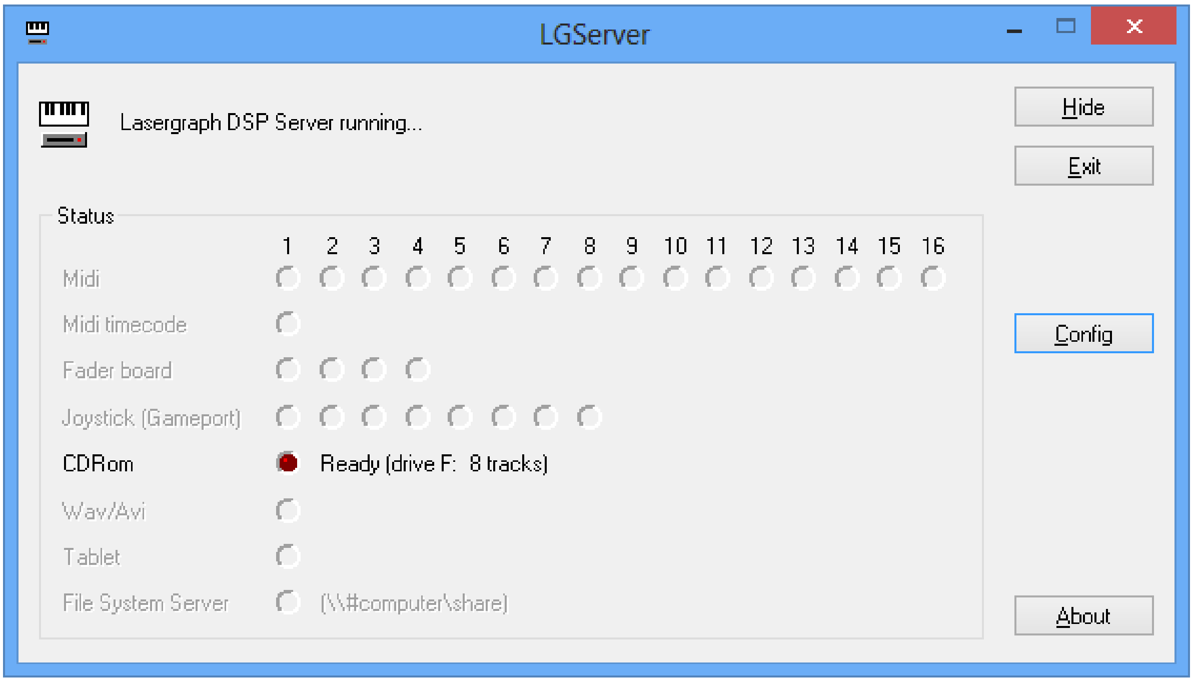
☐ Enable Wav/Avi Path *.wav

☐ Write Access

☐ Enable Tablet Port: COM2

☐ Enable File System Server (Use "\\#computer\share" to access shares)

Activate "Enable CDRom" and save the setting.


LGServer

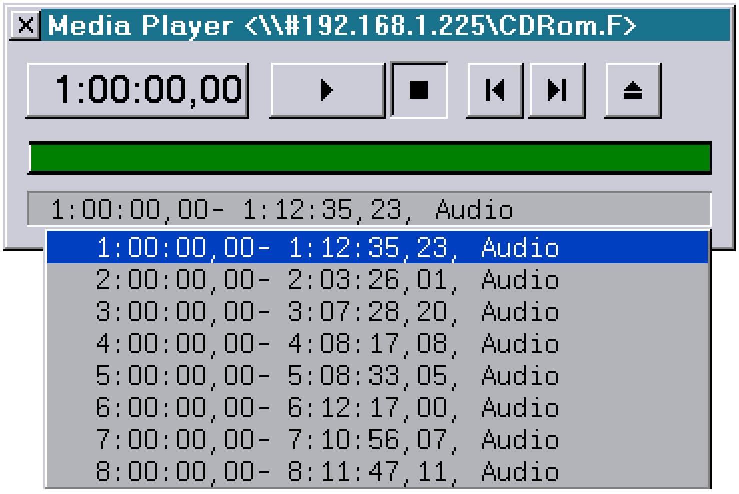
Lasergraph DSP Server running...

Status

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Midi timecode	○															
Fader board	○	○	○	○												
Joystick (Gameport)	○	○	○	○	○	○	○	○								
CDRom	● Ready (drive F: 8 tracks)															
Wav/Avi	○															
Tablet	○															
File System Server	○ (\\#computer\share)															

The red point indicates that the CD-ROM access is enabled. If there is a CD in the drive the number of tracks found on the CD is displayed.
In this example there is a CD with 8 tracks in the CD drive "F:".

After finishing these settings the Media Player of the Lasergraph DSP (client) starts with the corresponding network path - IP number of the PC and CD-ROM drive - and you can access the audio files of the CD although the CD is in the CD-ROM drive of a PC.


Media Player <\\#192.168.1.225\CDRom.F>

1:00:00,00

```
1:00:00,00- 1:12:35,23, Audio
    1:00:00,00- 1:12:35,23, Audio
    2:00:00,00- 2:03:26,01, Audio
    3:00:00,00- 3:07:28,20, Audio
    4:00:00,00- 4:08:17,08, Audio
    5:00:00,00- 5:08:33,05, Audio
    6:00:00,00- 6:12:17,00, Audio
    7:00:00,00- 7:10:56,07, Audio
    8:00:00,00- 8:11:47,11, Audio
```

Note: The LGServer has to stay activated while work is done over the network.

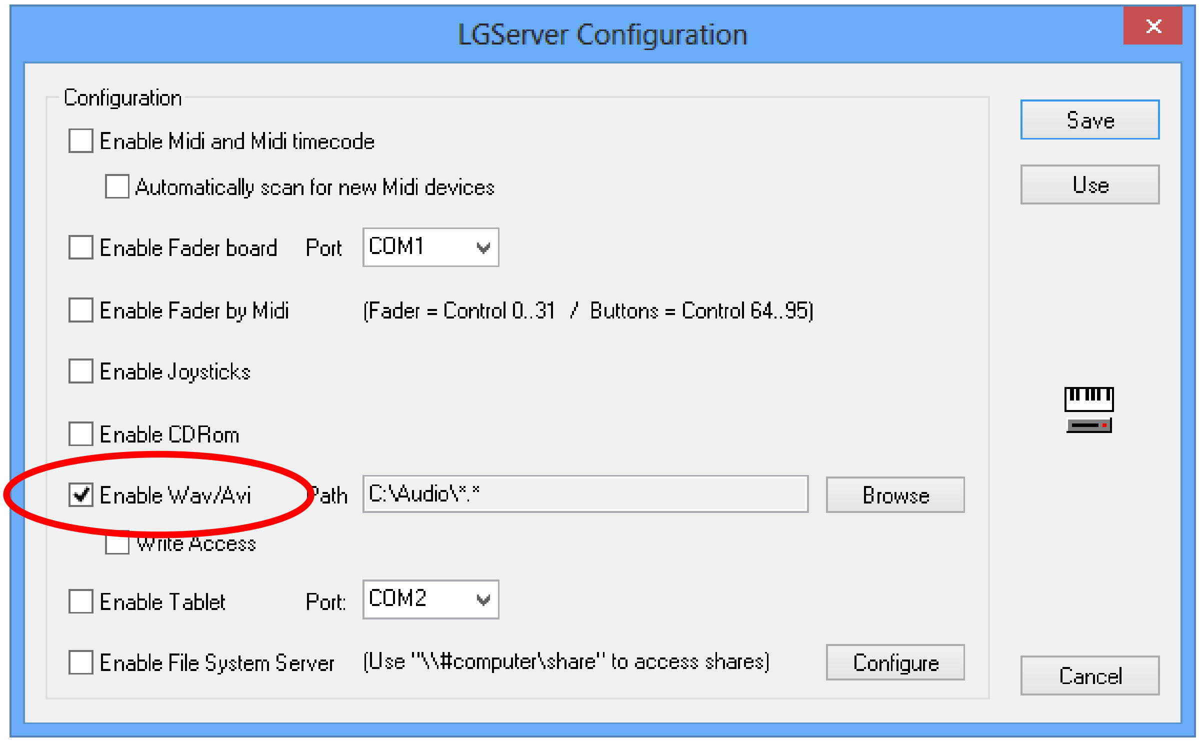
Using Audio Files over the Network from the Lasergraph DSP

The DSP supports the playback of the following media file types:

- ✓ WAV files (*.wav)
- ✓ MP3 files (*.mp3)
- ✓ OggVorbis files (*.ogg)
- ✓ Windows Media Audio files (*.wma)
- ✓ Midi files (*.mid)
- ✓ Audio Interchange Format files (*.aif)
- ✓ Video files (*.avi)

Audio stored on the hard disk of a PC (server) can be used by a Lasergraph DSP (client). An important condition for this is that a **sound card** and the Windows® **Media Player** for replay are installed on the PC.

The PC is entered as server in the I/O Setup of the Lasergraph DSP (client), for instance with the network name "shuttle". On the PC the LGServer has to be configured:



LGServer Configuration

Configuration

□ Enable Midi and Midi timecode

□ Automatically scan for new Midi devices

□ Enable Fader board Port COM1

□ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)

□ Enable Joysticks

□ Enable CDRom

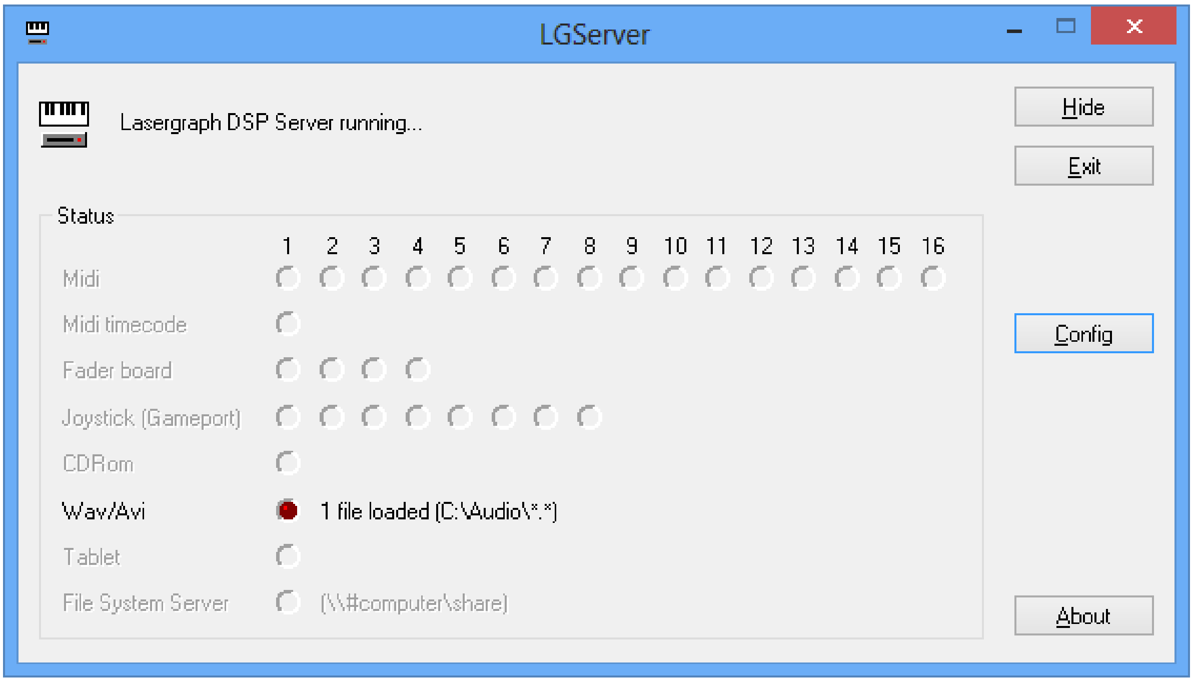
☑ Enable Wav/Avi Path C:\Audio\*.*

□ Write Access

□ Enable Tablet Port: COM2

□ Enable File System Server (Use "\\#computer\share" to access shares)

Activate the checkbox "Enable Wav/Avi" and, using "Browse", select the corresponding file(s) or all *.wav, *.mp3 files from a folder on the hard disk of the PC. Then save the settings.

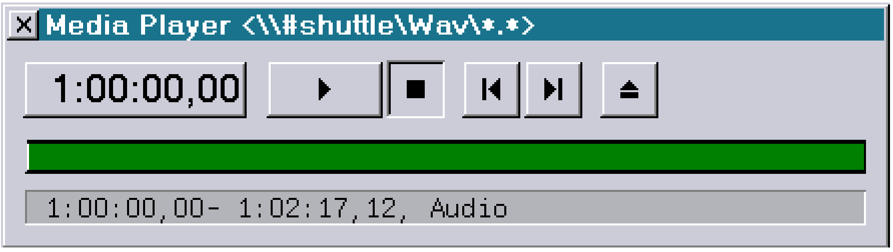


Lasergraph DSP Server running...

Midi ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○
Midi timecode ○
Fader board ○ ○ ○ ○
Joystick (Gameport) ○ ○ ○ ○ ○ ○ ○ ○
CDRom ○
Wav/Avi ● 1 file loaded (C:\Audio\*.*)
Tablet ○
File System Server ○ (\\#computer\share)

If audio files are recognized this is indicated by the red point in the LGServer window and the number of audio files and the path are displayed.

After these settings the Media Player of the Lasergraph DSP starts and the media files of the PC can be accessed:



Media Player <\\#shuttle\Wav\*.*>

1:00:00,00

1:00:00,00- 1:02:17,12, Audio

The headline shows the network path with the network name of the PC, e.g. "shuttle", with the symbol for audio files "Wav" and with the corresponding selection, e.g. "*.*".
The symbol '#' indicates a service provided by the LGServer.

Notes:

- Audio files are sorted alphabetically like all other files in Windows® directories and therefore are listed as audio track 1 to n in the popup menu of the Media Player. If a specific order is desired the audio files can be numbered when they are saved under Windows®, for example:

  `01_Mozart.wav, 02_Beethoven.wav, 03_Wagner.wav,` etc.

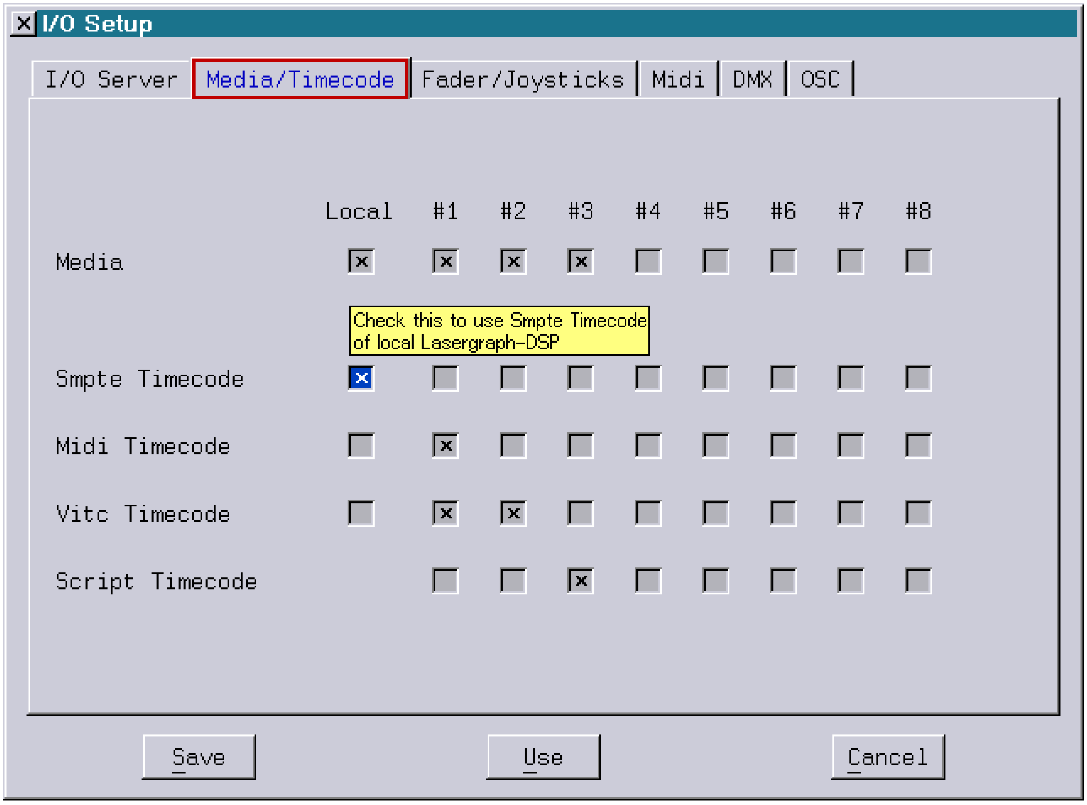
  This way the order in which they appear in the popup menu of the Media Player can be defined.

- Audio files from the hard disk of a PC are used in the time script of the Lasergraph DSP exactly like CD audio tracks.

- Although the DSP supports the use of compressed audio files like MP3 or OggVorbis files, we recommend using uncompressed files like *.wav files only.

Timecode

External timecode sources (e.g. SMPTE, MIDI, VITC, Script) which are connected to a computer (server) in the network can be used by other computers (clients).



	Local	#1	#2	#3	#4	#5	#6	#7	#8
Media	☒	☒	☒	☒	□	□	□	□	□
Smpte Timecode	☒	□	□	□	□	□	□	□	□
Midi Timecode	□	☒	□	□	□	□	□	□	□
Vitc Timecode	□	☒	☒	□	□	□	□	□	□
Script Timecode		□	□	☒	□	□	□	□	□

Check this to use Smpte Timecode of local Lasergraph-DSP

Smpte Timecode
In the example (see above) the SMPTE input at the local Lasergraph DSP is used by the local Lasergraph DSP.

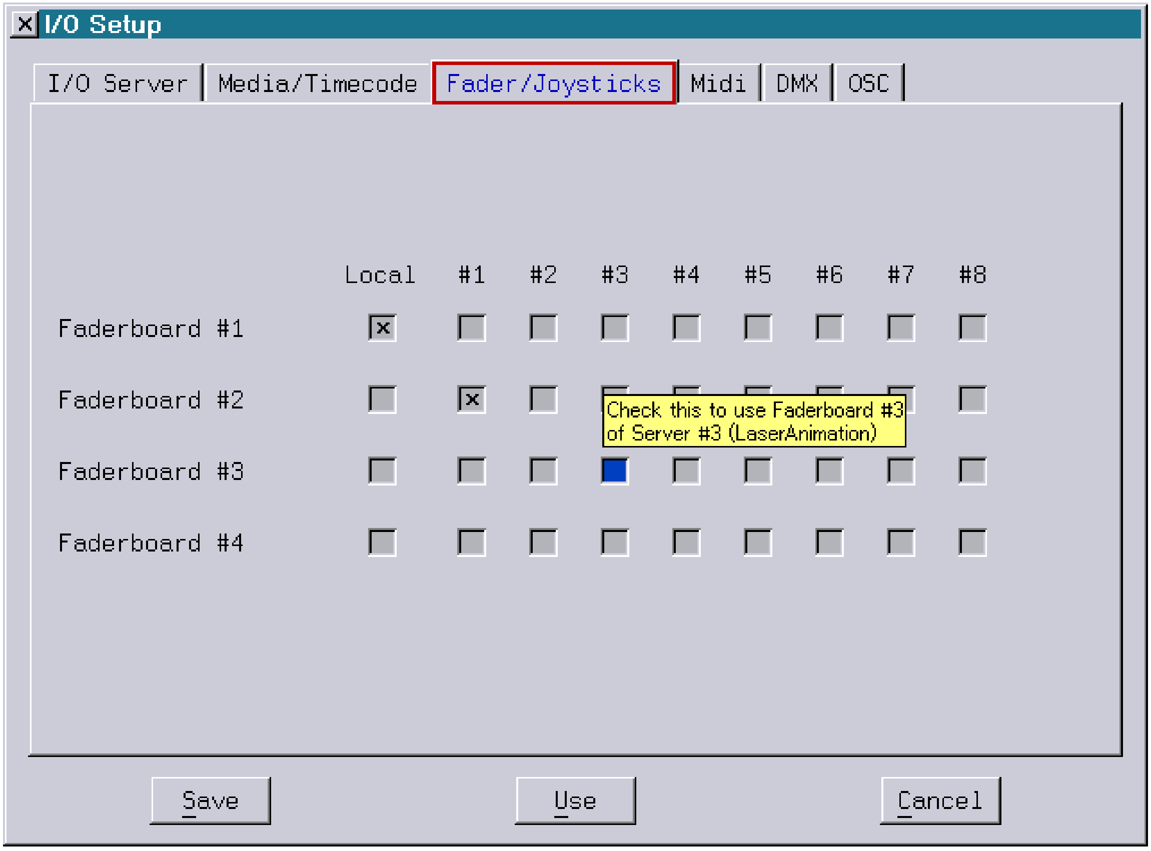
Midi Timecode (MTC)
Here the MIDI input of a Lasergraph DSP or a PC can be used as the timecode source at the local Lasergraph DSP. Select the server for the MTC.

Vitc Timecode (Vertical Interval Time Code)
Select here which server will supply the VITC.
Note: Not all Lasergraph DSPs are equipped with a VITC input!

Script Timecode
Netscript is the current timecode used by the time script of another Lasergraph DSP in the network. Choose the server for the timecode.

Note:
If more than one timecode server for the same source is selected, make sure that only one of them is actually sending timecode.

Fader/Joysticks



	Local	#1	#2	#3	#4	#5	#6	#7	#8
Faderboard #1	☒	□	□	□	□	□	□	□	□
Faderboard #2	□	☒	□	□	□	□	□	□	□
Faderboard #3	□	□	□	🟦	□	□	□	□	□
Faderboard #4	□	□	□	□	□	□	□	□	□

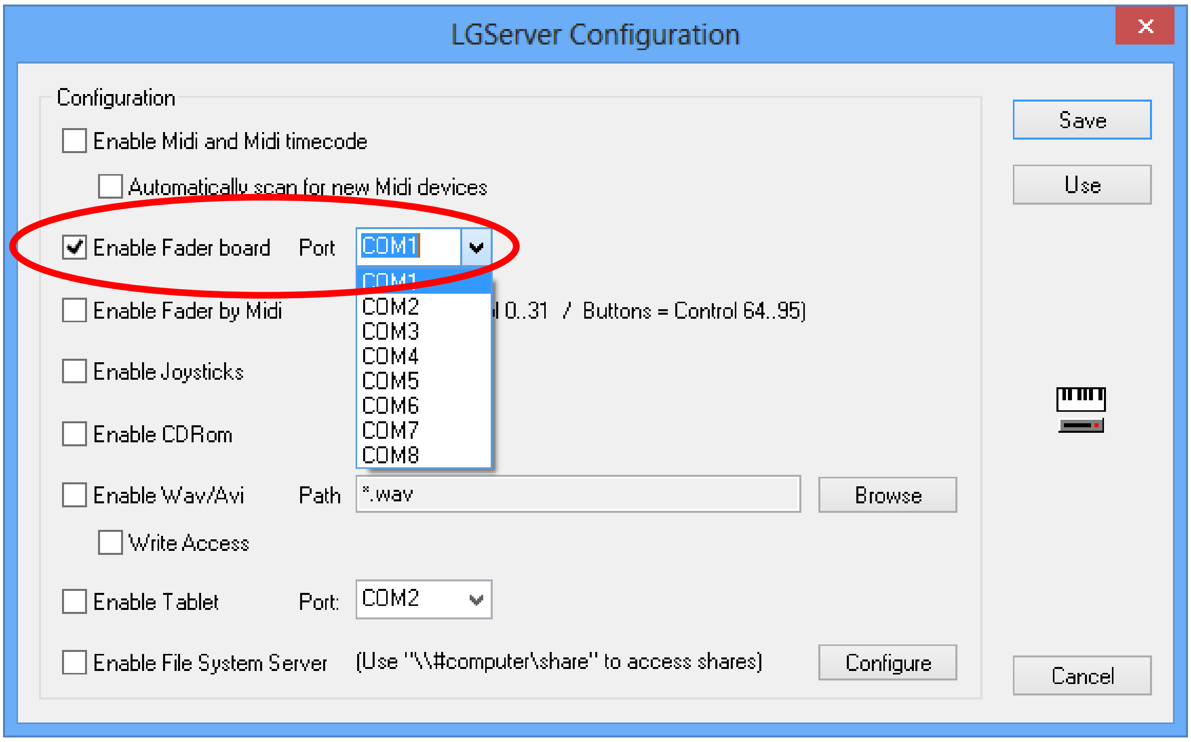
Check this to use Faderboard #3 of Server #3 (LaserAnimation)

Faderboards are connected mainly to Lasergraph DSPs and can be used by other Lasergraph DSPs in the network.

Connection to a PC is also possible, however an adapter is necessary.

If a Faderboard is connected to a PC without the corresponding adapter this can lead to irreparable damages on the PC and the Faderboard.

If a Faderboard is connected to a PC (server) with the necessary adapter and is supposed to be used by a Lasergraph DSP (client) in the network, the following settings have to be made in the LGServer on the PC:



LGServer Configuration

Configuration

□ Enable Midi and Midi timecode

□ Automatically scan for new Midi devices

☑ Enable Fader board Port COM1

□ Enable Fader by Midi ... 0..31 / Buttons = Control 64..95)

□ Enable Joysticks

□ Enable CDRom

□ Enable Wav/Avi Path *.wav

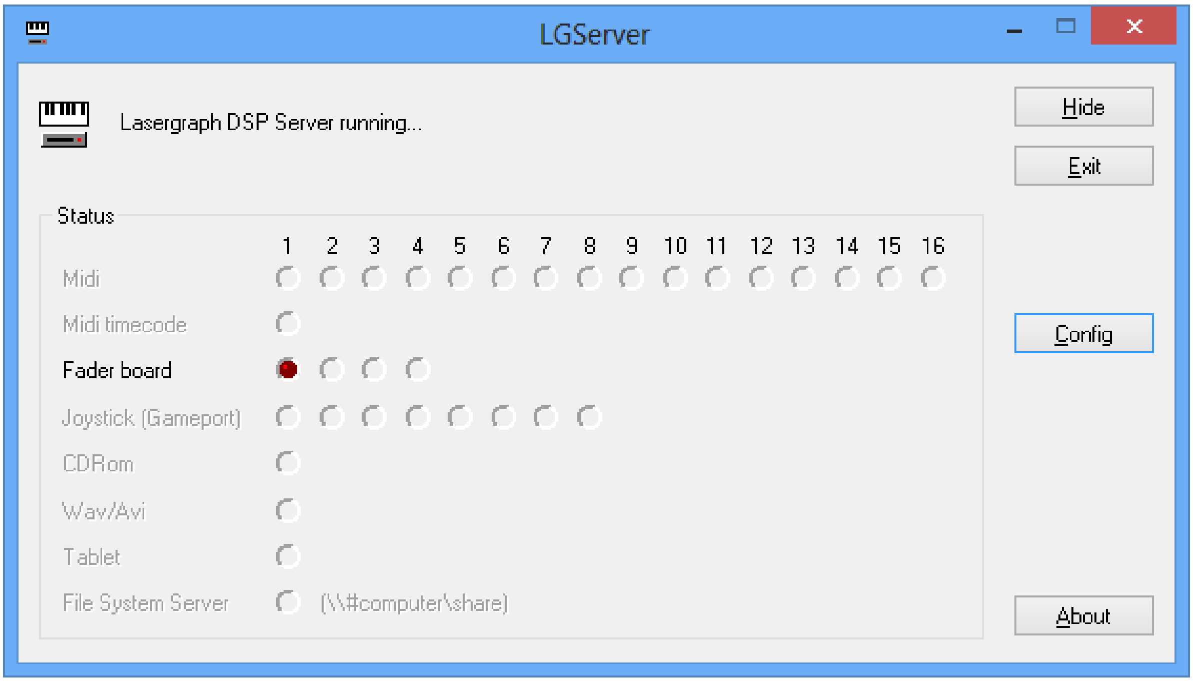
□ Write Access

□ Enable Tablet Port: COM2

□ Enable File System Server (Use "\\#computer\share" to access shares)

Activate "Enable Fader board" with the checkbox and in "Port" select the corresponding interface to which the Faderboard is connected at the PC. Save the settings.

If Faderboard units connected to the PC are detected this is signaled with the red points. They can then be used by the Lasergraph DSP.



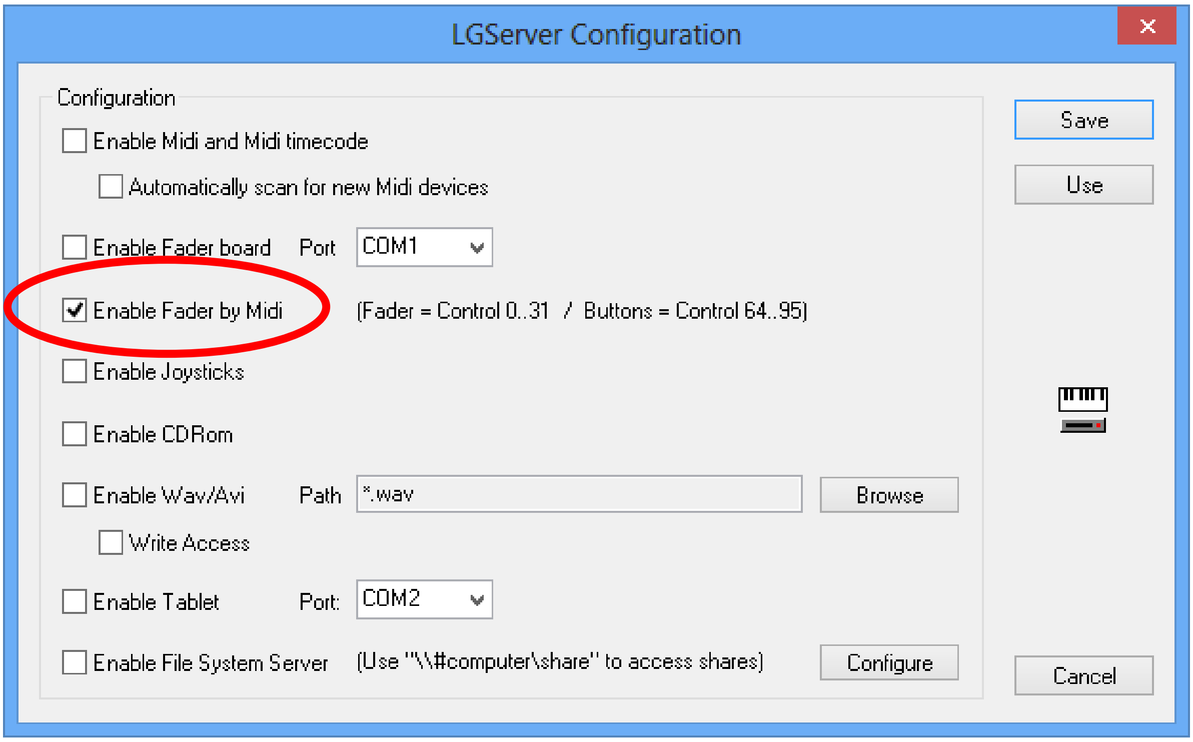
LGServer

Lasergraph DSP Server running...

Status

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Midi timecode	○															
Fader board	●	○	○	○												
Joystick (Gameport)	○	○	○	○	○	○	○	○								
CDRom	○															
Wav/Avi	○															
Tablet	○															
File System Server	○ (\\#computer\share)															

Instead of the Faderboard, a Midi-Controller can be used as Fader unit at a PC.



Configuration

□ Enable Midi and Midi timecode

□ Automatically scan for new Midi devices

□ Enable Fader board Port COM1

☑ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)

□ Enable Joysticks

□ Enable CDRom

□ Enable Wav/Avi Path *.wav

□ Write Access

□ Enable Tablet Port: COM2

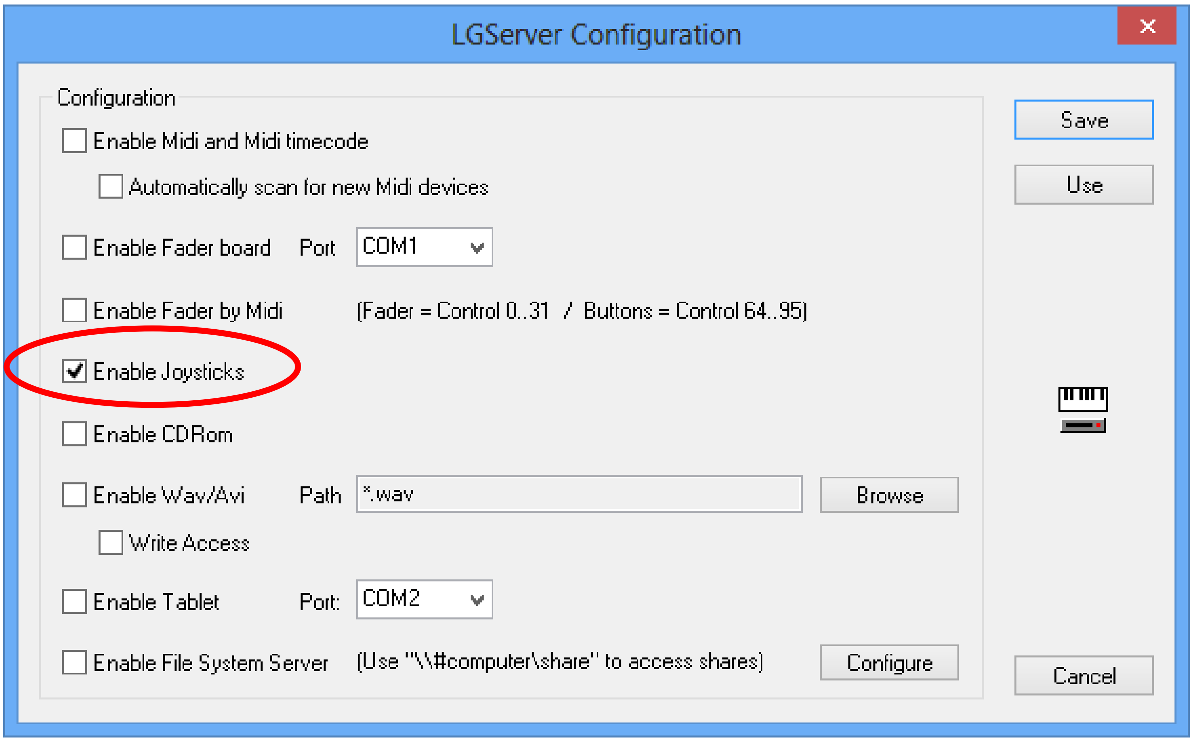
□ Enable File System Server (Use "\\#computer\share" to access shares)

A MIDI-controller can simulate two Faderboards with 16 faders and 16 buttons each. The assignment is as follows:

Fader 0 → Controller 0	Button 0 → Controller 64
Fader 1 → Controller 1	Button 1 → Controller 65
Fader 2 → Controller 2	Button 2 → Controller 66
"" ""	"" ""
"" ""	"" ""
Fader 29 → Controller 29	Button 29 → Controller 93
Fader 30 → Controller 30	Button 30 → Controller 94
Fader 31 → Controller 31	Button 31 → Controller 95

Joysticks connected directly to the game port of the PC or to the USB port can be used over the network by Lasergraph DSPs.

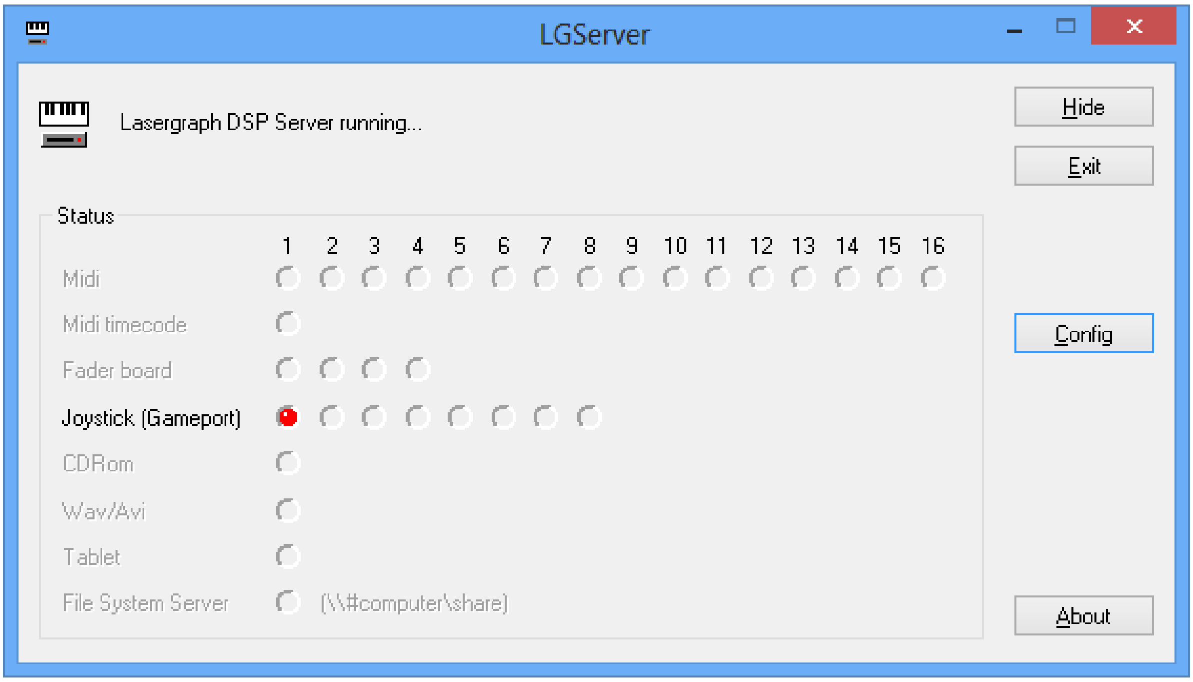
Make the corresponding settings in the LGServer, see the following pictures:



LGServer Configuration

Configuration

- □ Enable Midi and Midi timecode
  - □ Automatically scan for new Midi devices
- □ Enable Fader board Port COM1
- □ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)
- ☑ Enable Joysticks
- □ Enable CDRom
- □ Enable Wav/Avi Path *.wav
  - □ Write Access
- □ Enable Tablet Port: COM2
- □ Enable File System Server (Use "\\#computer\share" to access shares)



LGServer

Lasergraph DSP Server running...

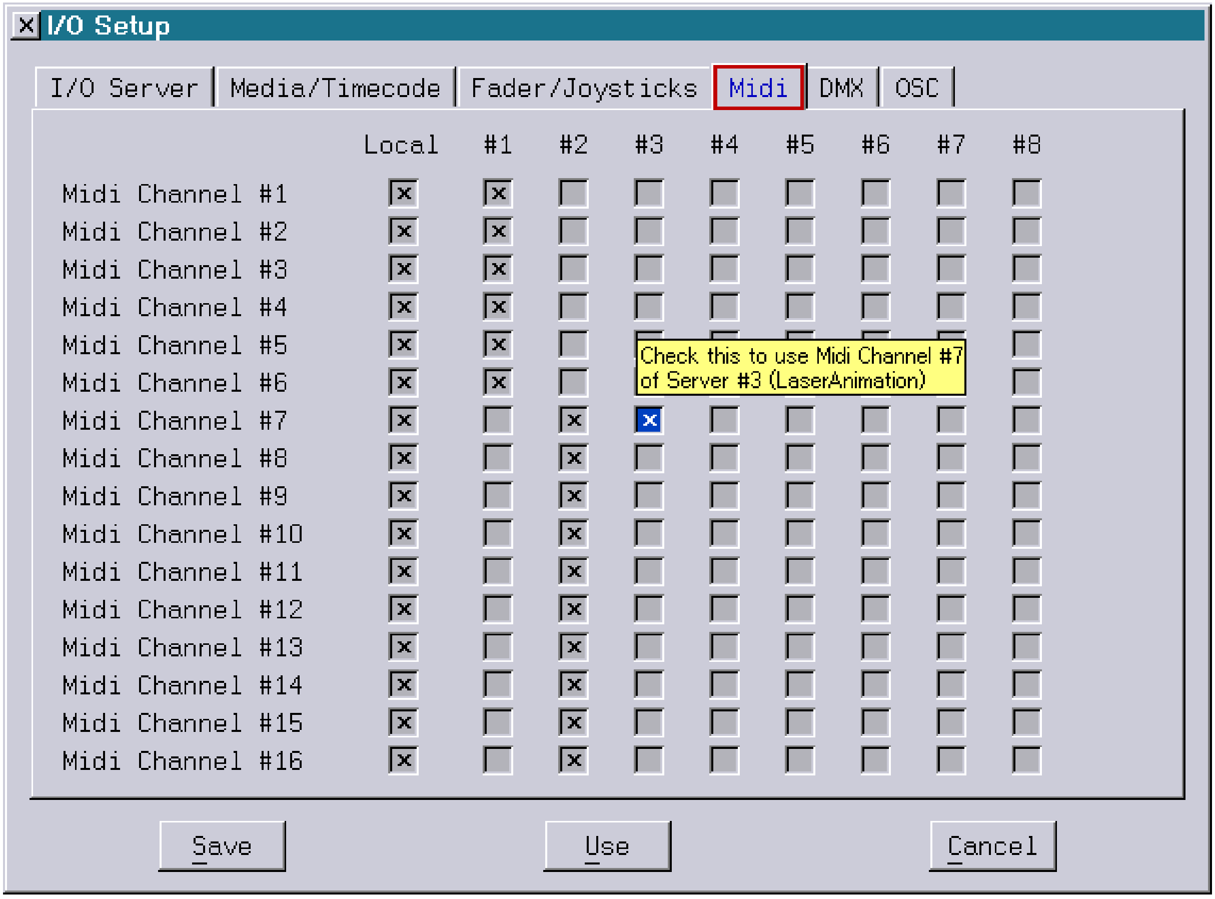
Status

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Midi timecode	○															
Fader board	○	○	○	○												
Joystick (Gameport)	●	○	○	○	○	○	○	○								
CDRom	○															
Wav/Avi	○															
Tablet	○															
File System Server	○	(\\#computer\share)														

If a Joystick is connected through the Faderboard (which is connected to a PC), only the Faderboard has to be activated in the LGServer. That will automatically activate the Joystick, too.

Midi

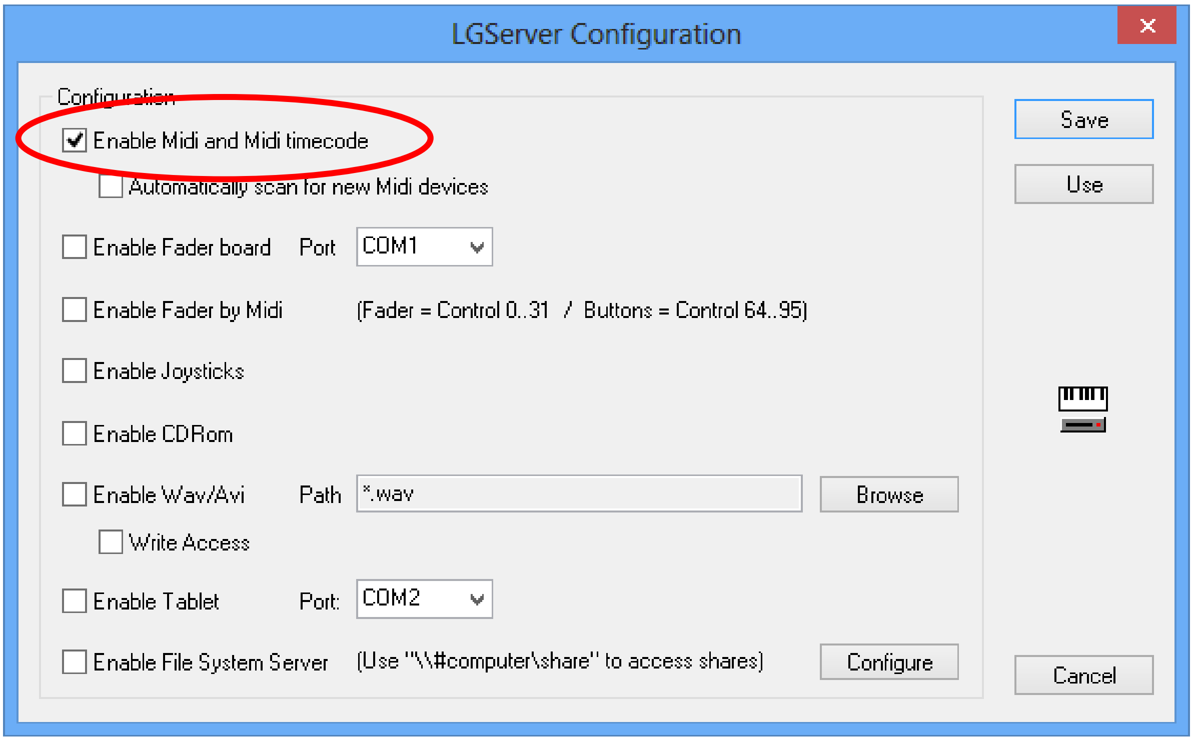
It is possible to define for each MIDI channel which server's MIDI data are used.



	Local	#1	#2	#3	#4	#5	#6	#7	#8
Midi Channel #1	☒	☒	☐	☐	☐	☐	☐	☐	☐
Midi Channel #2	☒	☒	☐	☐	☐	☐	☐	☐	☐
Midi Channel #3	☒	☒	☐	☐	☐	☐	☐	☐	☐
Midi Channel #4	☒	☒	☐	☐	☐	☐	☐	☐	☐
Midi Channel #5	☒	☒	☐						☐
Midi Channel #6	☒	☒	☐						☐
Midi Channel #7	☒	☐	☒	☒	☐	☐	☐	☐	☐
Midi Channel #8	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #9	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #10	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #11	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #12	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #13	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #14	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #15	☒	☐	☒	☐	☐	☐	☐	☐	☐
Midi Channel #16	☒	☐	☒	☐	☐	☐	☐	☐	☐

Check this to use Midi Channel #7 of Server #3 (LaserAnimation)

If a MIDI keyboard is connected to a PC and is supposed to be available in the network, make the corresponding settings in the LGServer on the PC, see pictures:



LGServer Configuration

Configuration

☑ Enable Midi and Midi timecode

□ Automatically scan for new Midi devices

□ Enable Fader board Port COM1

□ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)

□ Enable Joysticks

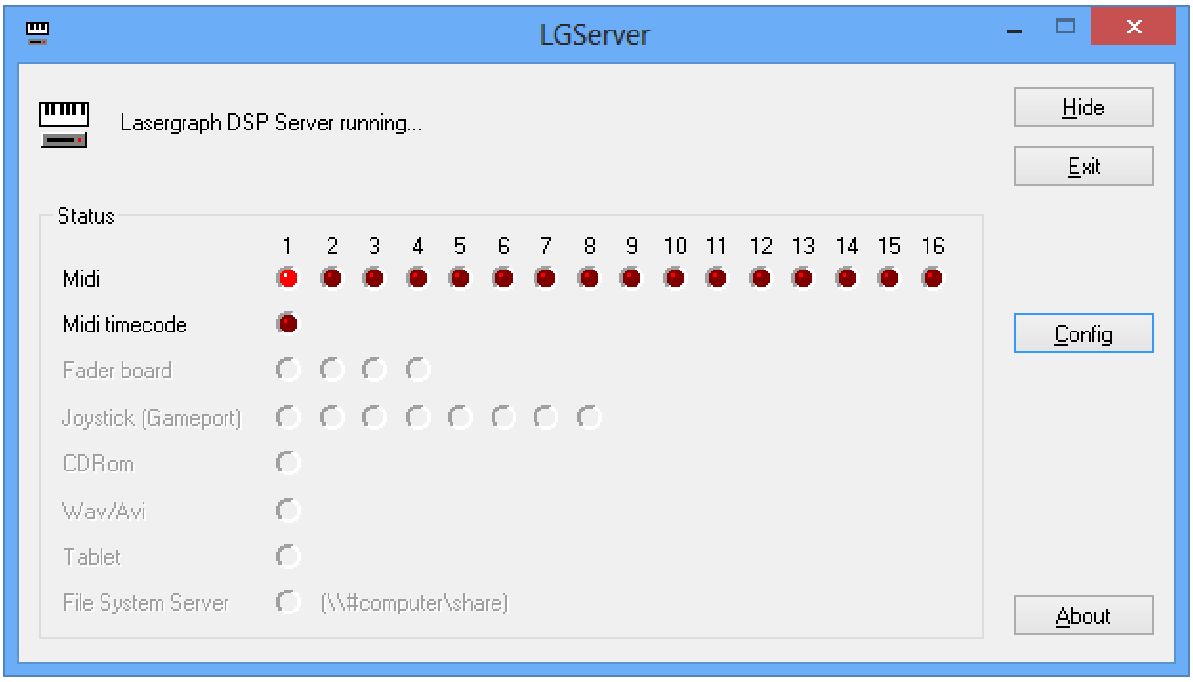
□ Enable CDRom

□ Enable Wav/Avi Path *.wav

□ Write Access

□ Enable Tablet Port: COM2

□ Enable File System Server (Use "\\#computer\share" to access shares)



LGServer

Lasergraph DSP Server running...

Status

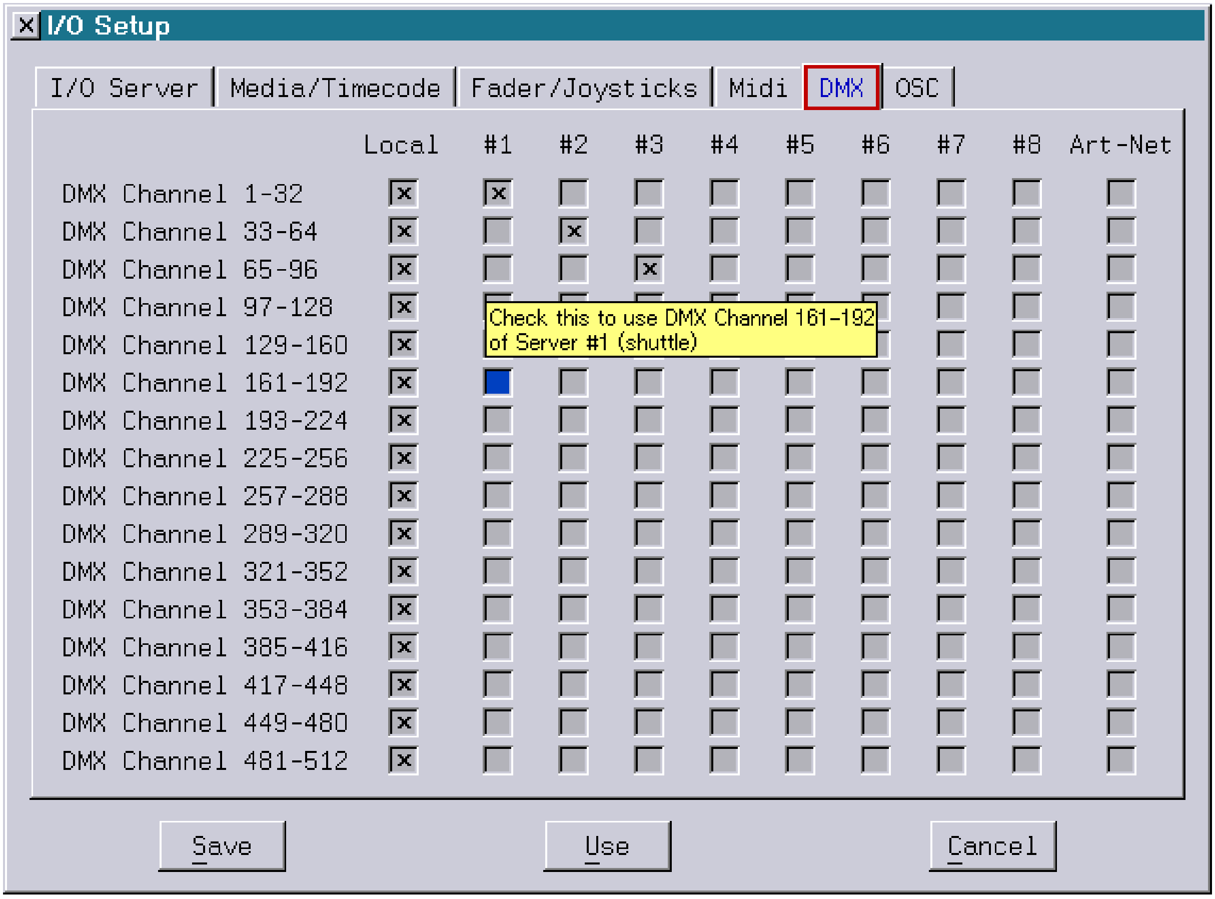
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Midi timecode	●															
Fader board	○	○	○	○												
Joystick (Gameport)	○	○	○	○	○	○	○	○								
CDRom	○															
Wav/Avi	○															
Tablet	○															
File System Server	○	(\\#computer\share)														

Note:
For connection of a MIDI keyboard to a PC a sound card or a Midi interface is required.

DMX

It is possible for a Lasergraph DSP to access the DMX input signal connected to another Lasergraph DSP (server).

For this purpose the Lasergraph DSP at which the DMX input signal is present is entered as server in the I/O Setup of the client. Then activate "DMX" in the I/O Setup and specify the respective DMX channels (1 to 512 in blocks of 32 channels) which are to be used by the client.



	Local	#1	#2	#3	#4	#5	#6	#7	#8	Art-Net
DMX Channel 1-32	☒	☒	□	□	□	□	□	□	□	□
DMX Channel 33-64	☒	□	☒	□	□	□	□	□	□	□
DMX Channel 65-96	☒	□	□	☒	□	□	□	□	□	□
DMX Channel 97-128	☒							□	□	□
DMX Channel 129-160	☒							□	□	□
DMX Channel 161-192	☒	■	□	□	□	□	□	□	□	□
DMX Channel 193-224	☒	□	□	□	□	□	□	□	□	□
DMX Channel 225-256	☒	□	□	□	□	□	□	□	□	□
DMX Channel 257-288	☒	□	□	□	□	□	□	□	□	□
DMX Channel 289-320	☒	□	□	□	□	□	□	□	□	□
DMX Channel 321-352	☒	□	□	□	□	□	□	□	□	□
DMX Channel 353-384	☒	□	□	□	□	□	□	□	□	□
DMX Channel 385-416	☒	□	□	□	□	□	□	□	□	□
DMX Channel 417-448	☒	□	□	□	□	□	□	□	□	□
DMX Channel 449-480	☒	□	□	□	□	□	□	□	□	□
DMX Channel 481-512	☒	□	□	□	□	□	□	□	□	□

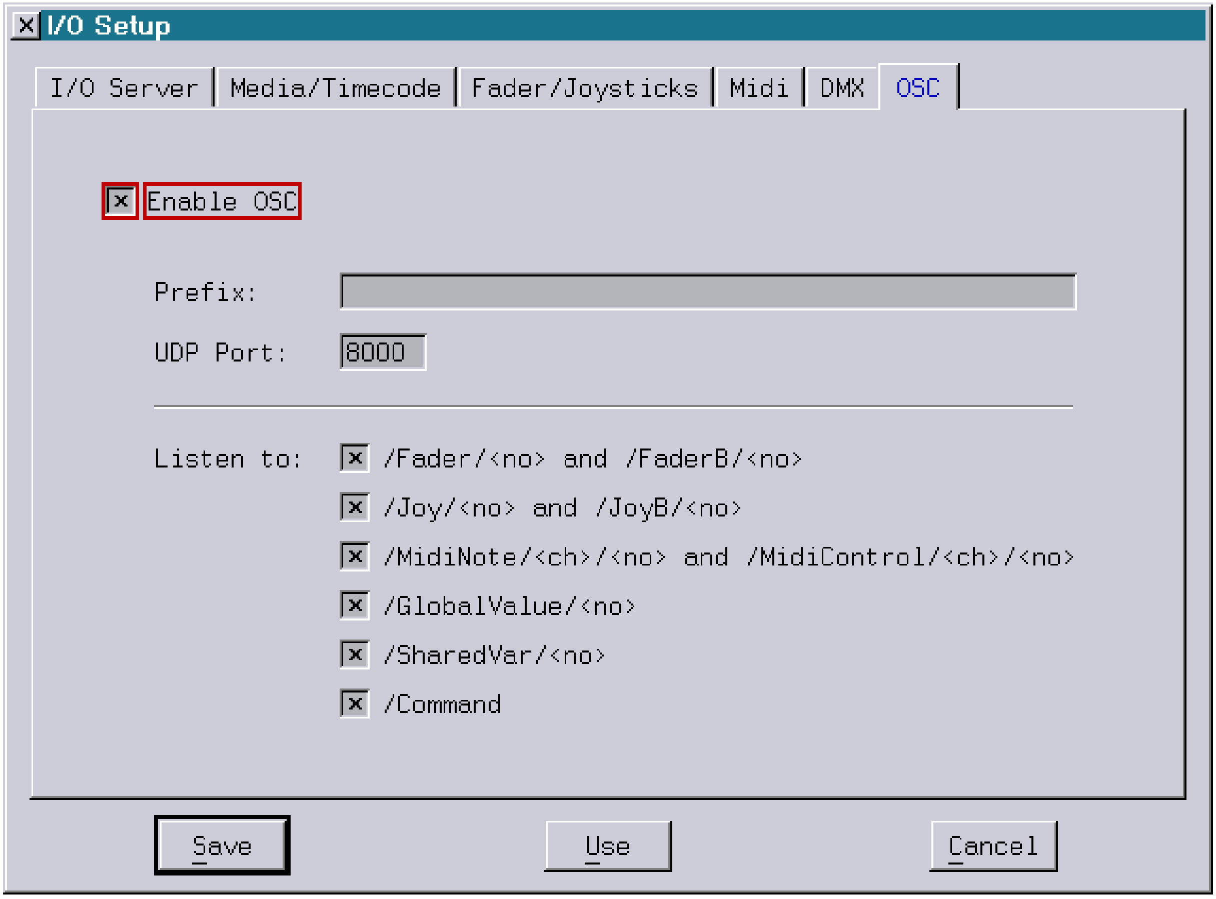
Check this to use DMX Channel 161-192 of Server #1 (shuttle)

You can use the following shortcuts to enable / disable several inputs simultaneously:

<L>	→	enables / disables all local Inputs
<A>	→	enables / disables all server Inputs
Middle mouse button	→	enables / disables all checkboxes of one server.

OSC

Open Sound Control (OSC) is an open, transport-independent, message-based protocol developed for communication among computers, sound synthesizers, and other multimedia devices. It can be configured in the I/O Setup.



☒ Enable OSC

Prefix:

UDP Port: 8000

Listen to:
☒ /Fader/<no> and /FaderB/<no>
☒ /Joy/<no> and /JoyB/<no>
☒ /MidiNote/<ch>/<no> and /MidiControl/<ch>/<no>
☒ /GlobalValue/<no>
☒ /SharedVar/<no>
☒ /Command

As soon as "Enable OSC" is activated the Lasergraph DSP will receive incoming OSC messages.

In addition to setting the port number used for receiving the UDP packets, here it is also possible to specify the prefix of the address which has to be sent to address this Lasergraph DSP.

The complete address is comprised of this freely definable prefix and the key words implemented in the Lasergraph DSP. The different key words can be activated or deactivated in blocks.

The following key words are supported:

/Fader/<no>

Sets the fader <no> to the specified value (<no>: 0-63).

Value range: 0.0 to 1.0 (float32) resp. 0 to 65535 (int32)

/FaderB/<no>

Turns the fader button <no> on resp. off (<no>: 0-63).

Value range on: not equal 0.0 (float32) resp. not equal 0 (int32)

Value range off: 0.0 (float32) resp. 0 (int32)

/Joy/<no>

Sets the X/Y position of the joystick <no> to the specified value (<no>: 0-7).

Value range: -1.0 to 1.0 (float32) resp. 0 to 255 (int32)

/JoyB/<no>

Turns the two buttons of joystick <no> on resp. off (<no>: 0-7).

Both buttons off: 0.0 (float32) resp. 0 (int32)

Button 1 on, button 2 off: 1.0 (float32) resp. 1 (int32)

Button 1 off, button 2 on: 2.0 (float32) resp. 2 (int32)

Both buttons on: 3.0 (float32) resp. 3 (int32)

/MidiNote/<ch>/<no>

Simulates the reception of a Midi note on resp. note off command.

<ch> is the Midi channel (1-16) and <no> is the number of the Midi key (0-127).

Permitted value range: -1.0 to 1.0 (float32) resp. -127 to 127 (int32)

/MidiControl/<ch>/<no>

Simulates the reception of a Midi control change command.

<ch> is the Midi channel (1-16) and <no> is the number of the controller (0-31).

Permitted value range: 0.0 to 1.0 (float32) resp. 0 to 127 (int32)

/GlobalValue/<no>

<no> specifies the number of the first Global Value which is to be set.

Legal value range: [0...1000000]

The value can be queried from the trickfilm using the function "GlobalValue(<no>)".

/SharedVar/<no>

<no> specifies the number of the first Shared Variable which is to be set.

Legal value range: [0...63]

The value can be queried from the trickfilm using the function "Shared(<no>)"

/Command

Enters the command sent as OSC string into the command line and executes it.

Since not all devices support OSC strings the command can alternatively be sent as part of the address: /Command/<command>

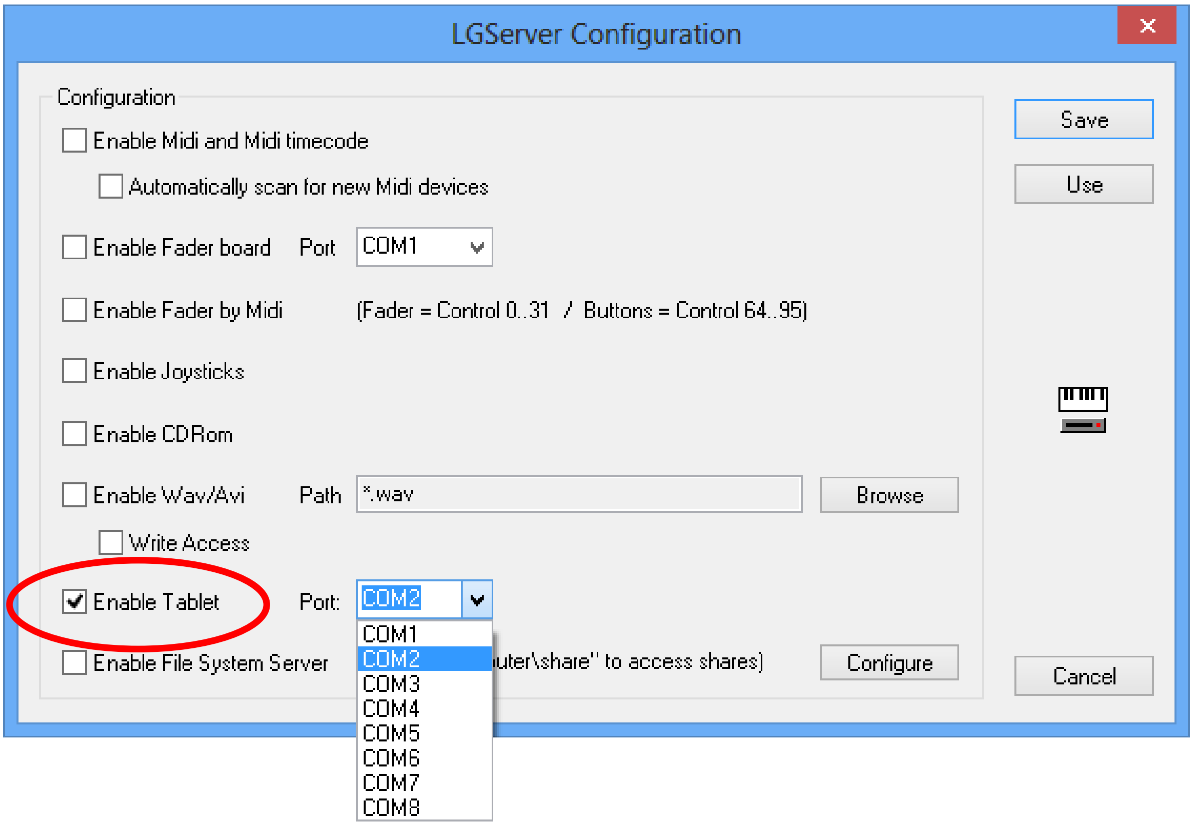
In this case the parameter float32 resp. int32 determines whether the command is to be executed or not. In case it is not equal 0.0 resp. 0 it will be executed, otherwise it will not.

Graphics Tablet Connected to PC

The graphics tablet, type Summasketch III® (Summagraphics) is mainly connected to and used by the Lasergraph DSP. This device can however be connected to a PC and used from a Lasergraph DSP in a network.

In contrast to the Lasergraph DSP interface for the graphics tablet, the corresponding PC interface does not provide the necessary power supply (+12V, 250 mA). If you want to connect a graphics tablet to a PC you need a power supply and a corresponding adapter.

The graphics tablet has to be configured in the program LGServer:



LGServer Configuration

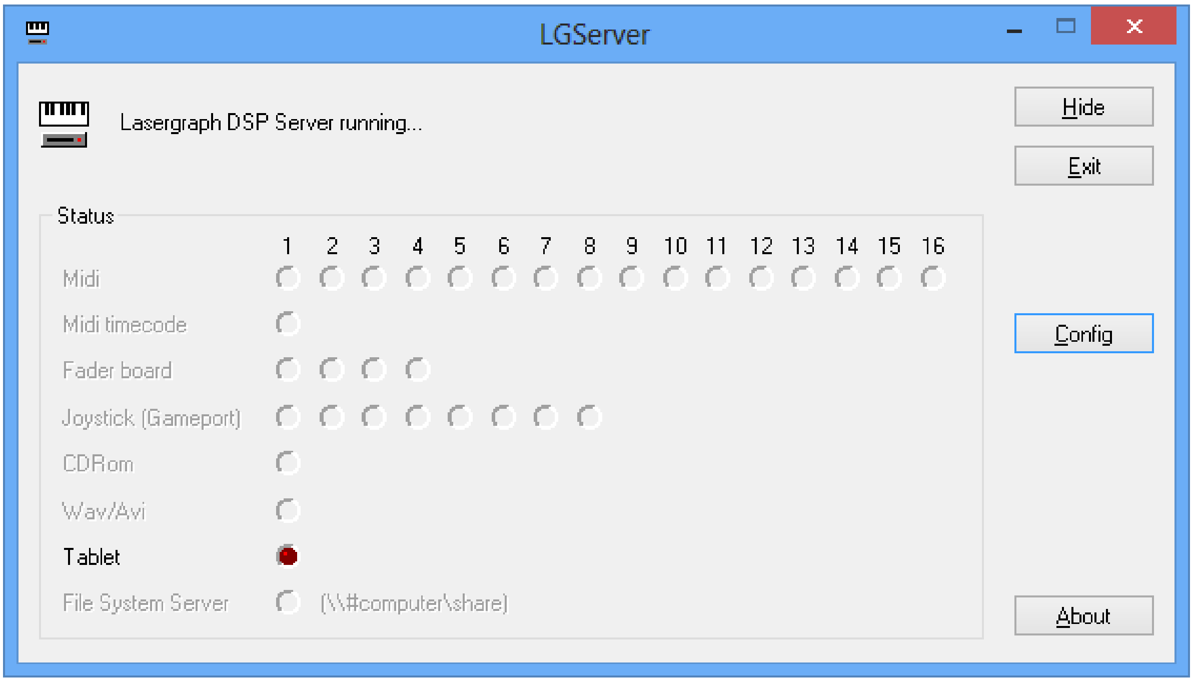
Configuration

- □ Enable Midi and Midi timecode
  - □ Automatically scan for new Midi devices
- □ Enable Fader board Port COM1
- □ Enable Fader by Midi (Fader = Control 0..31 / Buttons = Control 64..95)
- □ Enable Joysticks
- □ Enable CDRom
- □ Enable Wav/Avi Path *.wav
  - □ Write Access
- ☑ Enable Tablet Port: COM2
- □ Enable File System Server ...uter\share" to access shares)

The left checkbox is used to turn the "Tablet driver" on or off.

In "Port" the corresponding interface of the PC to which the graphics tablet is connected has to be selected.

If a graphics tablet is connected and has been detected, this is signaled by the red point:



Lasergraph DSP Server running...

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Midi	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Midi timecode	○															
Fader board	○	○	○	○												
Joystick (Gameport)	○	○	○	○	○	○	○	○								
CDRom	○															
Wav/Avi	○															
Tablet	●															
File System Server	○	(\\#computer\share)														

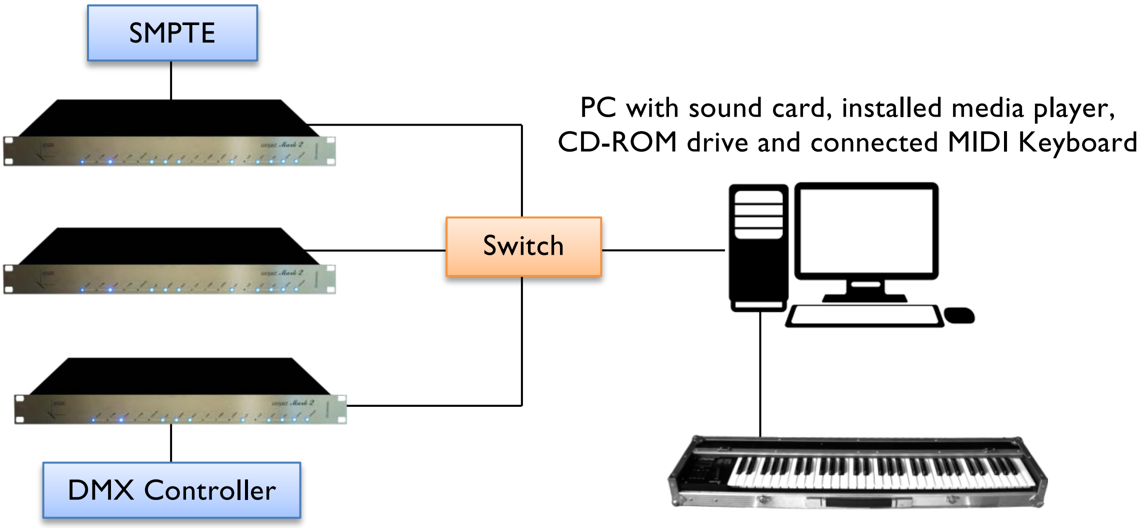
The graphics tablet connected to the PC takes the place of the mouse function and can therefore also be used without starting the LGRemote as long as the LGServer is running.
The installed "Tablet driver" was mainly designed for the use of the graphics tablet from the Lasergraph DSP over the network. Therefore full functionality of the graphics tablet on the PC cannot be guaranteed.

Note:
The driver software provided by Summagraphics for the graphics tablet must not be installed on the PC, otherwise there will be conflicts with the LGServer.

Setup Examples

Several Lasergraph DSPs are Clients - Two Lasergraph DSP are Client and Server at the same time

Three **Lasergraph DSP** ***compacts*** and one PC are connected to an Ethernet network.


SMPTE
PC with sound card, installed media player,
CD-ROM drive and connected MIDI Keyboard
Switch
DMX Controller

The CD-ROM drive of the PC is supposed to be integrated on Lasergraph DSP *compact* 1 for data exchange. In addition Lasergraph DSP *compact* 1 is supposed to make its SMPTE timecode available to *Compact* 2 and 3. It is therefore client and server at the same time.

Lasergraph DSP *compact* 2 is to access audio files from the hard disk of the PC, so it is only client.

The MIDI keyboard connected to the PC is to be used from Lasergraph DSP *compact* 3.
Furthermore *Compact* 1 and 2 are supposed to access the DMX input signal of *Compact* 3.
It is also client and server simultaneously.

Lasergraph DSP *compact* 1 - as client for use of the CD-ROM drive of the PC:

- Share the CD drive of the PC for the network - Windows® share (click drive with right mouse button and enter the share name under Sharing).
- Start LGRemote on the PC and connect to Lasergraph DSP *compact* 1.
- Call up the Setup on *Compact* 1, choose "FS Client", select and assign the logical drive and enter the network name of the PC and the share name and save the settings.
- Change to the assigned drive on *Compact* 1 in the file system and the contents of the CD in the PC will be displayed so that the files can be used on the Lasergraph DSP *compact* 1.

Lasergraph DSP *compact* 1 as server providing its connected SMPTE for *Compacts* 2 and 3:

- Connect to Lasergraph DSP *compact* 2 and 3 over LGRemote.
- Call up Setup on Lasergraph DSP *compact* 2 as well as Lasergraph DSP *compact* 3 and select the button "Net I/O".
- Enter *Compact* 1 with its network name as server under "I/O Server".
- Then select "Media/Timecode" and assign the device "SMPTE Timecode" to the registered server and save the settings.
- Lasergraph DSP *compact* 2 and 3 are now controlled with the SMPTE timecode from Lasergraph DSP *compact* 1.

Lasergraph DSP *compact* 2 as client using the audio files from the hard disk of the PC:

- Call up Setup "Net I/O" on *Compact* 2 and enter the PC with its network name as I/O server.
- Start the LGServer on the PC, click on "Config", activate "WAV" and, using "Browse", select the WAV- or other media files from the hard disk of the PC and save the settings.
- The Media Player on Lasergraph DSP *compact* 2 starts and the selected audio files can be used.

Lasergraph DSP *compact* 3 as client which is to use the MIDI keyboard connected to the PC:

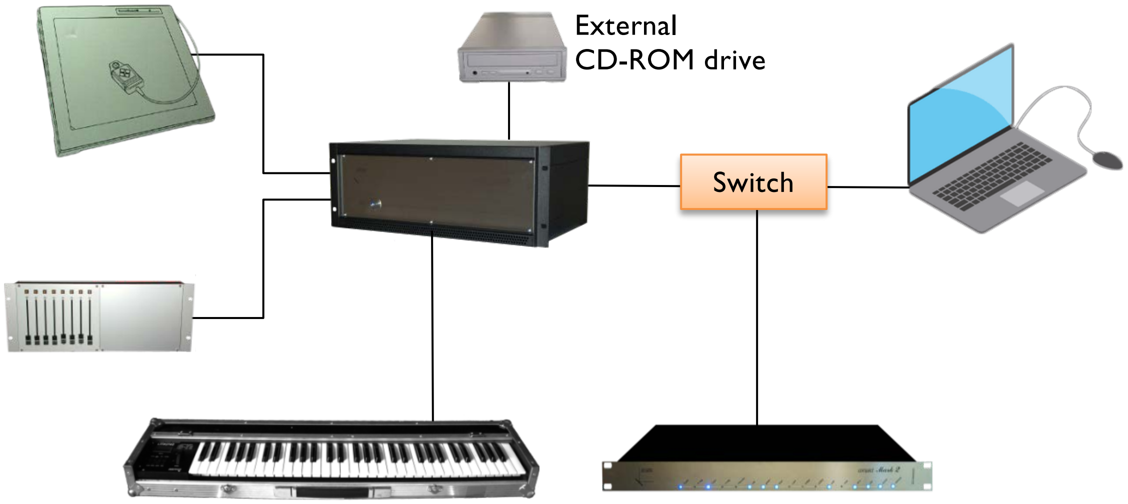
- Call up Setup "Net I/O" on *Compact* 3 and enter the PC with its network name as I/O server.
- Select "MIDI" and assign the respective MIDI channels 1 to 16 to the registered server as desired.
- Start the LGServer on the PC, click on "Config", activate "MIDI" and save the settings.
- The Lasergraph DSP *compact* 3 now reacts to movements of the keys of the MIDI keyboard although it is connected to the PC.

Lasergraph DSP *compact* 3 as server providing its DMX 512 input to Lasergraph DSP *compact* 1 and 2:

- Enter Lasergraph DSP *compact* 3 with its network name as I/O server under "Net I/O" in the Setup of *Compact* 1 as well as *Compact* 2.
- Switch to "DMX 512", assign the respective DMX channels to the registered server and save the settings.
- Now the DMX input signal present at Lasergraph DSP *compact* 3 can be accessed by Lasergraph DSP *compact* 1 and 2.

One Client Uses Several Servers

One Lasergraph DSP *workstation*, one Lasergraph DSP *compact* and a laptop are connected via Ethernet network.



The peripheral devices, Faderboard, MIDI keyboard, external CD-ROM drive, connected to the Lasergraph DSP *workstation* as well as the script timecode are supposed to also be used by the Lasergraph DSP *compact*.

In addition, some files stored on the hard disk of the Lasergraph DSP *Workstation* are supposed to be available on the Lasergraph DSP *Compact*.

For replay purposes mostly only the Lasergraph DSP *compact* and the laptop are used. Therefore the CD-ROM drive of the laptop is also to be available to the *Compact* or it is supposed to have access to certain directories on the hard disk of the laptop. This means the Lasergraph DSP *compact* is client of two servers (Lasergraph DSP *workstation* + PC).

Lasergraph DSP *compact* as client for the devices connected to the *Workstation* and the laptop:

- Call up the Setup, "Net I/O" on the Lasergraph DSP *Compact* and enter the Lasergraph DSP *Workstation* as server #1 and the laptop as server #2 with their network names.
- Select the devices:

"Media/Timecode"	assign "Media" to servers #1 and #2 / "Script Timecode" to server #1.
"Fader/Joysticks"	assign the Faderboard unit to server #1.
"Midi"	assign the respective Midi channels to server #1.

- Save the settings.
- For use of the CD-ROM drive of the laptop share the drive under Windows®, start the LGServer on the laptop, click "Config", activate CD-ROM and save the settings.
- All devices can now be used from the Lasergraph DSP *compact* as if they were directly connected to it.

Note:
If several CD-ROM drives from several servers are connected, the first CD which is placed in one of the drives will be used until the "eject" button is pressed. Then the next available "media" drive will be used.

Lasergraph DSP *compact* as client for data exchange with the *Workstation* and the laptop:

- Define the server sharing:
  Call up the Setup "FS Server" on the Lasergraph DSP *workstation*, enter the share and the corresponding path (drive\directory) for which the share is valid. Define permissions and enter user name and password if desired. Save the settings.
  On the laptop under Windows® share the directory which is to be available.
- On the Lasergraph DSP *compact* call up the Setup, "FS Client", select two logical drives, e.g. W: and L:, assign them, for W: enter the network name and the share of the Lasergraph DSP *workstation* and for L: enter the laptop with its network name and the specified share. Save the settings.
- When you change to drive W: or L: on the Lasergraph DSP *compact* in file system, the files from the directories on the hard disks of the Lasergraph DSP *workstation* and the laptop can be used on the Lasergraph DSP *compact*.

Name Server and Gateway

A Name Server (DNS, Domain Name Server) is a server that resolves names to IP addresses. The DNS can be integrated into a router or be a specific server provided by your ISP (Internet Service Provider).

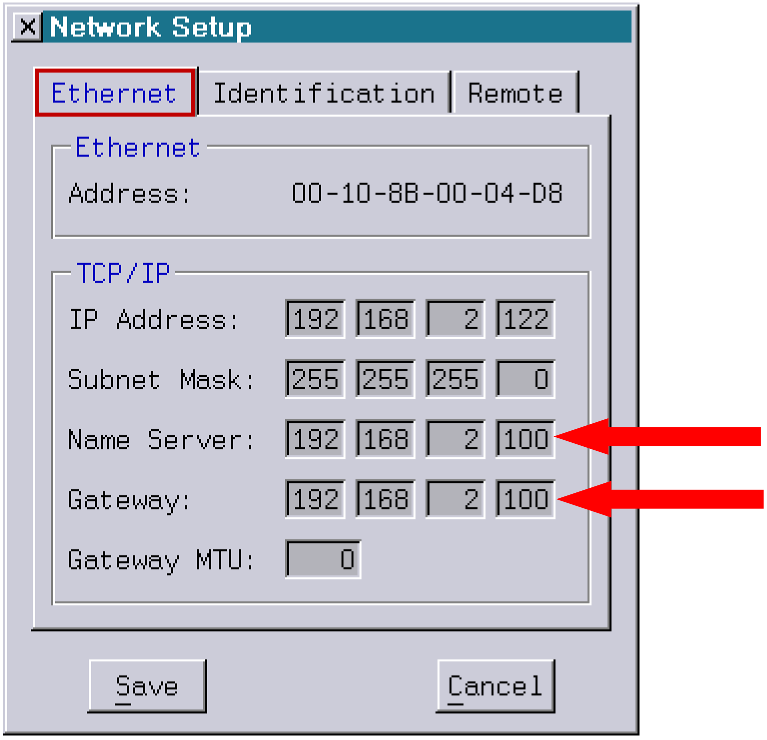
A Gateway is a router that connects different networks. It will forward data beyond the local network.

Name Server:
Type in the IP address of your DNS.

Gateway:
Put the IP address of your local router/gateway into the fields.

Gateway MTU (Maximum Transmission Unit):
The MTU defines the size of the largest data packet that can be sent.
The most common size for an Ethernet network is 1.500 bytes.
In rare cases it can be useful to reduce the packet size.
The smallest MTU size is 576.
The value 0 defines the default value.

Example:
If you want to get access to a Lasergraph DSP from somewhere else than through the local network, the following settings have to be made:



Ethernet
Address: 00-10-8B-00-04-D8

TCP/IP
IP Address: 192 168 2 122
Subnet Mask: 255 255 255 0
Name Server: 192 168 2 100
Gateway: 192 168 2 100
Gateway MTU: 0

In this example the Lasergraph DSP is inside a local network with the following Parameters:

Lasergraph DSP IP address:	192.168.2.122
Subnet Mask:	255.255.255.0
Router IP address:	192.168.2.100

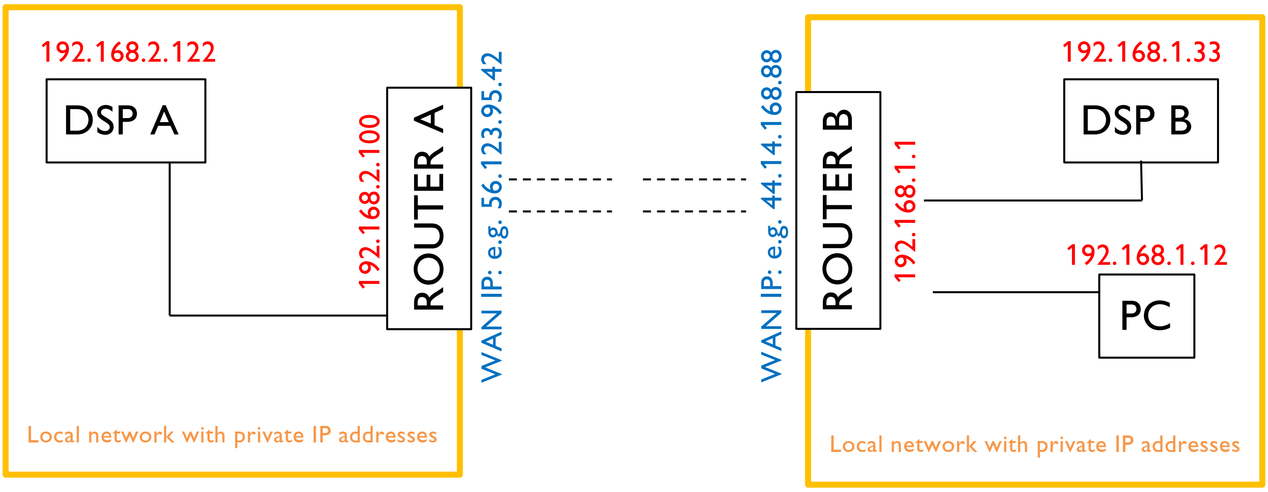
Type the address of the Router into the fields "Name Server" and "Gateway" in the network setup of the Lasergraph DSP.

If the router is configured to hand over the data to the Lasergraph DSP, you should be able to access the Lasergraph DSP by entering the WAN IP address (external IP address of the local network containing the Lasergraph DSP) into the LGRemote on the PC located outside the local network.

If the network is equipped with a firewall, make sure that the Lasergraph DSP belongs to the DMZ (Demilitarized Zone) or that the ports used by the Lasergraph DSP are mapped (port forwarding) to the local Lasergraph DSP.

Important ports are:

8202	File transfer from Lasergraph DSP to Lasergraph DSP.
8200	LGRemote (without SuperPreview)


192.168.2.122
DSP A
192.168.2.100
ROUTER A
WAN IP: e.g. 56.123.95.42
Local network with private IP addresses
WAN IP: e.g. 44.14.168.88
ROUTER B
192.168.1.1
192.168.1.33
DSP B
192.168.1.12
PC
Local network with private IP addresses

Note:
Normally you do not need to assign the Gateway and the Name server.
This is only necessary if you want to connect the Lasergraph DSP to another network than the local net.

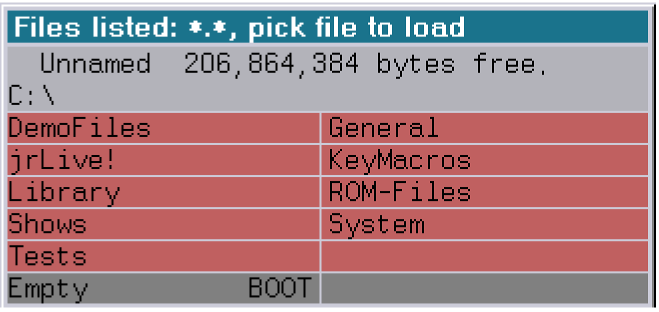
Appendix

Ports used by Lasergraph DSP and Production Suite

To set up a firewall it is necessary to know the ports which are used by the Lasergraph DSP and the Production Suite for communication.

			Port	Protocol
LGRemote	Remote Control		8200	TCP
	Lasergraph DSP select list		8201	UDP
	Super Preview	Connection establishment	8208	TCP
		Afterwards	1024...4095	TCP
	Netboot		8205 + 8206	UDP
	IP configuration		8201	UDP
	Date and Time	Connection establishment	8204	TCP
		Afterwards	1024...4095	TCP
LGServer		Connection establishment	8202 + 8203	TCP
		Afterwards	1024...4095	TCP
LGPreview	Super Preview	Connection establishment	8208	TCP
		Afterwards	1024...4095	TCP
	Lasergraph DSP select list		8201	UDP
LGTimecode	Net Timecode		8201	UDP
Laser Disable			8211	UDP
Remote Commands			8210 (selectable)	TCP
Art-Net			6454	UDP
OSC	Open Sound Control		8000 (selectable)	UDP
Visualization			5568	UDP
LANtoSerial		Beacon	22123	UDP
		Communication	60000	TCP

Chapter File



```
Files listed: *.*, pick file to load
  Unnamed  206,864,384 bytes free.
C:\
DemoFiles               General
jrLive!                 KeyMacros
Library                 ROM-Files
Shows                   System
Tests
Empty             BOOT
```

Table of Contents

Introduction

A laser show consists of a wealth of information that you can generate, set up and edit with the aid of the *Mark 2*. This information is managed in the main memory of the device while in use. Unlike the program data which is loaded from flash memory when a command is executed, all information pertaining to a laser show (user data) which is to be retained must be stored on data media.

Terms

Hard disks and hardware designed to accommodate data storage media (such as floppy disks, CDs or Zip media) are referred to as drives.
The storage capacity of larger drives such as hard disks or Zip media can be organized in up to 4 logical sections. Each of these sections can be used as if it were a separate physical drive. These sections of a drive, which are managed individually by the computer, are known as partitions or logical drives.
During its boot process, a computer checks the installed drives and their partitioning. An internal ID is assigned to each recognized drive to permit its storage capacity to be identified unambiguously. PCs generally use the letters of the alphabet (drive letters) as identifiers.
In addition to local drives, the enabled network directories are handled as partitions, too.

Storage Media

The *Mark 2 workstation* and *compact* are supplied with an internal IDE hard disk drive or an IDE Flashcard (Solid State Disk).
Older DSP workstation units have also a 3.5" HD floppy disk drive.

Internal Data Media Administration

The *Mark 2* assigns consecutive letters to the connected data media in a manner similar to a PC operating system. With the exception of the internal floppy disk drive (A:), four consecutive drive letters are reserved for each drive, beginning with the letters C: to F: for the internal hard disk drive. This permits high-capacity storage media to be organized with up to four partitions which can then be used as if they were four separate physical drives.

A:	internal 3.5" floppy disk drive (older DSP workstation only)
C:	internal hard disc (D:, E:, F: if it is partitioned)
A-Z:	assigned network drives

With older DSP workstations the letters of additional drives connected to the SCSI interface are assigned in ascending order as per their SCSI ID:

G:	(H:, I: , J:)	SCSI device with the lowest ID
K:	(L:, M:, N:)	SCSI device with the next higher ID
etc.		

In this case it is not relevant whether a data medium has been partitioned or not. Unused drive letters are ignored and are thus not available.

This drive identification method permits specific data media to always be addressed via the same drive letter if you take care to observe the sequence of SCSI IDs when connecting the devices.

CDs can contain any number of different sections with different types of information. If a CD contains more than four data tracks, only the first four will be available. The most recent data track will be assigned the lowest of the four available drive letters.

Assignment of local partitions:
Local partitions are either IDE-Hard drives / CF-Cards or SCSI-Drives attached to the SCSI-Bus.

Jumper “PRI” ON (default), set on the disk-controller.

IDE-Master → IDE-Slave → SCSI Drives (workstation only)

A: → Floppy (workstation only)	
C: - F: → first drive	
G: - J: → second drive	
K: - N: → third drive	
etc.	

Jumper “PRI” OFF

SCSI Drives (workstation only) → IDE-Master → IDE-Slave

A: → Floppy (workstation only)	
C: - F: → first drive	
G: - J: → second drive	
K: - N: → third drive	
etc.	

Assignment of local partitions in combination with network drives:
If c: is assigned as a network drive (Setup, FS-Client), the
local drives will start with letter G: - J:.
If any other of the local existing partition is assigned as a network drive,
it won’t be accessible.

The Storage System

Directories and Files

The storage system of the *Mark 2* is similar to the file systems of other computers. The data for a laser show is stored in a variety of files.
Any number of directories can be created to organize the files for multiple laser shows. In addition to the files for a laser show, these directories can also contain subdirectories. This permits all user files to be stored in a clear overview.
We recommend creating a main directory for each show to help maintain your overview of related information. Various versions of a show can be grouped in subdirectories.

Special Directories

In addition to the user-defined directory structure, *Mark 2* has two special directories: **catalogs** and **BOOTs**.

A catalog (*.CATL) is required whenever individual pictures that have been created in the picture editor are to be stored.
Generally, a catalog contains all of the pictures of a show. A show can be put together using pictures from a variety of catalogs.
All catalogs containing pictures for a given show must be saved in the same directory as the show itself, as the program will only seek the catalogs and associated pictures in that directory.

Tip: Create a BOOT for each show to ensure that all information for a given show can be loaded using a single command.

A BOOT that contains all of the information loaded at the time of creation can be created with the FILE command "CreateBoot"
Pictures and animations are stored together in a compressed file named "BootCatalogs.CCAT" (Compressed CATalog).

Loaded animation sequences, the timescript, global parameters, the keyboard and bank configuration, as well as the beam table are stored in this directory. Files previously saved retain their original names. New information that has not been saved yet is named after the BOOT.

A BOOT is useful to load shows quickly or to restore frequently-used combinations of a variety of information (pictures, color settings, keyboard assignments, projection area sizes, etc.) via the single command ⇨ "ReBoot".

Directories in Drive C:

The following directories and subdirectories are set up by default on drive C:\ of the *Mark 2*:

DemoFiles contains sample files for

- the use of all film commands with detailed descriptions in the *Commands* subdirectory,
- specific *Mark 2* programming options in the *Macros* subdirectory,
- Sipro users in the *Sipro* subdirectory,
- spatial depictions that can be viewed with 3D glasses in *3-D*,
- games in the *Games* subdirectory,
- special applications, for example in which an object, a camera or both follow specified paths, in the *Paths* subdirectory,
- *demo shows* by third-party suppliers in *3rdParty.*
- *templates for the use with proTize / ILDA Import.*

Empty (BOOT) can be used to clear the main memory; this directory should not be overwritten.

General contains all included fonts for the creation of texts. Each font has been stored as a catalog (*.CATL) in which all available characters have been stored as pictures.

jr.Live! contains bitmaps for the playback and live interface "jr.Live!

KeyMacros is intended for working with macros in conjunction with the *Mark 2*. Frequently used procedures can be recorded in a macro and associated with a key combination. Macros created in this manner are automatically stored in this directory.

Library (animations) contains over 10,000 first-class frames for all occasions, sorted by topic.

ROM-Files with subdirectories corresponding to the processors used. These subdirectories contain the suitable system software (ROM image file) and generally one or more previous versions as LZip files.

Shows contains high-quality shows.

System is the directory containing the system and configuration files.

Tests contains files for hardware tests, for example for connected scanners, color control, etc.

Naming of Directories and Files

Directories and files receive names when they are created. In some cases, you may specify the names yourself. A name can consist of two parts separated by a dot.

The first part consists of a maximum of 14 characters and, with few exceptions, can be freely chosen; the second part consists of a maximum of 4 characters and is generally assigned automatically. It is used by the *Mark 2* to identify the document or directory type.

Specific file extensions

*.ani	File which references the pictures of a picture catalog as an animation sequence. Animations are created in the *Mark 2* using the "AnimEd" animation editor.
*.ary	File containing the state of the variables array.
*.bcfg	File in which the naming of the DMX-Channels are saved.
*.bin	File containing the complete data of a software version (ROM image file).
*.bmp	Image files created by a PC program and which can be integrated in a show with the animation command "ApplyBitmap".
*.boot	Directory containing the information loaded at the time of its creation. To load the files of a BOOT individually, change to the root directory beforehand with the FILE command "ChangeDir".
*.btab	File containing the definition of beams.
*.catl	Directory that can contain multiple pictures.
*.ccat	File containing pictures and animations in compressed form. "Compressed CATalogs" are created automatically when creating a BOOT. They can be loaded completely by rebooting or using the FILE command "LoadCOmpr".
*.ccyc	File containing information about a color cycle ⇨ "color cycle editor"
*.col	File containing the primary color mixes ⇨ "primary color table"
*.curv	File containing a curve ⇨ "curve editor"
*.dgc	Settings of the Digital Geometric Correction.
*.dgcc	Contains a single setting of the Digital Geometric Correction
*.dscr	Script file containing a sequence of commands (⇨ startSCRIPT).
*.dvs	Video file used by the “ApplyBitmap” command.
*.film	File including picture output commands (⇨ animation).
*.fval	File containing the state of the faders.
*.gpar	File including information about the size of the projection area (⇨ global parameters).
*.kcfg	File containing the key assignments of a MIDI keyboard.

*.lzip	Archive file created with the proprietary DSP archive program LZIP.
*.macr	Files in which frequently used command sequences can be saved as so-called "KeyMacros". These macros can be executed via hotkey combinations assigned to them.
*.mcfg	File which can be used to redirect a bank assignment.
*.pic	Pictures created using the picture editor ⇨ "Piced"
*.sipr	File containing the complete information for a laser show to be presented using a Sipro (Simple Projector).
*.spv	Settings of the SuperPreview.
*.svar	File containing the state of shared variables.
*.tscr	File containing information for the timed output of a show ⇨ "timescript".

Upper/Lower Case

The *Mark 2* is case-sensitive with regard to file names. This is intended to make listings of drive contents easier to read. The *Mark 2* does not distinguish between upper and lower case when loading shows. Files in which the names differ only in case must therefore be stored in different directories to ensure that they are addressed properly.

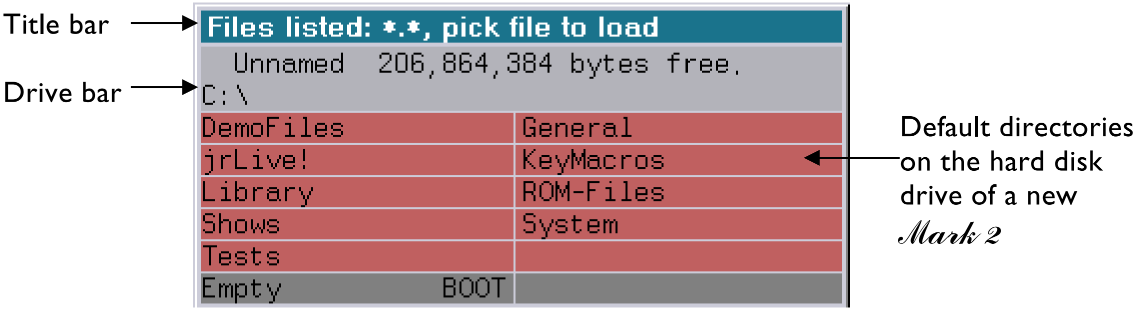
Issues Related to Other Operating Systems

The *Mark 2* also accepts data media that have been formatted with the MS DOS™ and Windows™ operating systems when using the FAT16 or FAT32 file system. This permits data to be exchanged between a variety of laser systems. However, problems may arise when copying files due to the differing naming conventions of the operating systems:

- MS DOS™ only permits file names of 8 + 3 characters; the *Mark 2* can accommodate 14 + 4. When copying files to data media formatted with MS DOS™, long file names are cut off after the eighth character. As a result, files already copied may be overwritten.
- MS DOS™ does not distinguish between upper and lower case.
- MS DOS™ only supports a limited character set for file names. To prevent difficulties with MS DOS™, use only the letters A to Z, the digits 0 to 9, the dash and underline _.
- The Windows™ operating systems support long file names with up to 255 characters. *Mark 2* accepts a maximum of 14 + 4 characters. However, when copying files from the *Mark 2* to data media formatted under Windows™ or vice-versa, the same restrictions apply as for MS DOS™.
- The full 14 + 4 *Mark 2* file names may be used when accessing a shared network drive under Windows™.

FILE Function Block

All commands related to file and data media management have been grouped in the FILE function block.
FILE mode can be invoked by selecting FILE <F3> or through direct keyboard input. FILE mode is also available from a number of different locations throughout the program by clicking the [FILE] button in the command line. The FILE window will then open on the left-hand side of the display:


Title bar
Drive bar
Default directories on the hard disk drive of a new *Mark 2*

```
Files listed: *.*, pick file to load
  Unnamed  206,864,384 bytes free.
C:\
DemoFiles            General
jrLive!              KeyMacros
Library              ROM-Files
Shows                System
Tests
Empty           BOOT
```

The first line of the drive bar indicates the name of the current drive ("Unnamed") and the available storage capacity in bytes (here: "206,864,384").
The second line contains the drive letter ("C:") and the name of the current directory ("\").

All files and direct subdirectories are available below the drive bar. Different background colors are used to distinguish between directories and files.
If the subdirectories and files within a given directory cannot be displayed at once, use the cursor keys <↓>, <↑> to scroll through the list one row at a time, <shift> + <↓>, <↑> to scroll five rows at a time or <PageUp>, <PageDown> to scroll the whole window.

All FILE commands and options are available in the menu bar within the program window.

FILE Mouse Functions

To change to a different drive, click on the drive bar. Use the left mouse button to scroll forwards alphabetically through the available drives, or the middle button to scroll backwards.

To change to a subdirectory, click on its name. To change to the next higher directory level, click on the ".." field.

It is only possible to change into BOOTs (*.BOOT) and picture catalogs (*.CATL) with the "ChangeDir" command.

Clicking on a picture catalog (*.CATL) or "Compressed CATalog" (*.CCAT) will automatically load all of the pictures therein to the picture memory.

Clicking a BOOT directory (*.BOOT) will activate the command "ReBoot". It will appear in the input line together with the name of the selected BOOT. The execution of the command must be confirmed.

Generally, files can be loaded with a left-click.

In the case of films (*.FILM), BOOTs (*.BOOT) and PrimaryColorTables (*.COL), the appropriate FILE command will be copied to the input line to permit additional parameters to be entered.

Double-clicking a BOOT directory (*.BOOT) will execute the "ReBoot" command immediately. Data in the main memory at that time will be overwritten. Unsaved changes will be lost.

Double-clicking a film will overwrite the current contents of the film memory.

FILE Commands

Overview

File Management Commands

The commands to load, save and unload files from the main memory are organized in function groups:

[load]	contains load commands
[unload]	contains commands to remove information from the main memory
[save]	contains save commands
[psave]	contains commands to save files with protection against unauthorized access.

Data Media Management Commands

Management commands for hard disks, floppy disks and other external data media:

Find	searches data media for specific files
[setdevice]	changes between the available drives
Dir	displays the contents of directories and subdirectories
List	displays the contents of the picture memory
MakeDir	creates a new directory
ChangeDir	changes from one directory to another; "ChangeDevDir" can be used to change to a directory on another drive
REName	renames files and directories
Copy	copies files and directories
DELete	deletes files and directories
Protect	write-protects files and directories
UnProtect	removes write-protection
COnvert	converts picture files of other file formats
Backup	makes backup copies of files, including the complete directory structure if required
LZIP	compresses files
UNLZIP	decompresses files
DiskName	assigns a name to the selected drive
DiskPassWd	assigns a password to protect the selected media against unauthorized access

ProtDisk	write-protects complete data media
UnProtDisk	removes write-protection
FORMAT	formats and partitions data media
SUBST	Connects a volume temporarily

The Menu Bar and its Commands

General Information

All FILE commands can only be executed if the FILE function block has been launched.

Directory or file names are optional or mandatory parameters for many commands.

Wildcard characters such as "*" and "?" may be used within a parameter name to specify multiple directories or files.
The asterisk serves as a wildcard for any number of characters within that part of the name (name or extension). The part of the name in which the wildcard is used may not contain any further characters after the wildcard.
The question mark serves as a wildcard for exactly one character. Further characters may follow a question mark within that part of the name.

Many commands can affect directories or files not in the current directory. It is necessary to provide the computer with information about the location of the data: the so-called path. Paths are a part of a parameter and are specified before the desired file or directory name.
A path can be described as the sequence of directory and subdirectory names that you would have to click through to make the directory containing the desired data the current directory. The names of the consecutive directories are separated by backslashes, "\". If the desired data is located on a different drive, the path must begin with the drive letter.

Paths can be specified in absolute or relative terms.

An absolute path describes the path from the root directory of the drive, through the directory structure, to the desired data. The path begins in this case with a backslash (\) that stands for the root directory of the desired drive, e.g.

```
\DemoFiles\Commands\*.film
```

This is the path to all film files (*.film) in the "Commands" subdirectory of the "DemoFiles" directory from the root of the current drive.

```
\\computer\share
```

This is the path to a shared directory in the network.

```
C:\ROM-Files\compact
```

This path points to the subdirectory "compact" of the "ROM-Files" directory in the root directory of drive "C:".

A relative path describes the route through the directory structure of a drive to the desired data from the current directory, e.g.

```
ABC\Shows\*.catl
```

This is the path to all picture catalogs located in the "Shows" subdirectory of the "ABC" directory, which is a subdirectory of the current directory.

```
C:\Shows\*.catl
```

This is the path to all picture catalogs located in the "Shows" subdirectory of the current directory on drive "C:".

Note that every drive has its own “current directory”, which means working in directory “Shows” on drive C: “`C:\Shows`” will set the current directory of drive C: to “Shows”. Changing to another drives, e.g. D:, will still keep the current directory of drive C: “Shows”.

Common Options

/Overwrite

This option discards obsolete data or updates existing data.
For example, when loading picture catalogs with "LoadCatalog", all desired pictures will be loaded, even if they are already present in the picture memory. Unsaved changes to the files already loaded will be lost.
The previous files will be replaced when saving "[save]". If this option has not been specified but a file of the same name already exists, a window will open asking you to confirm the overwrite operation.

/Subdir

Use this option to instruct the *Mark 2* to include all subdirectories in the execution of the command in addition to the current directory.
This option is also required in order to copy a BOOT or a catalog.

Switching Drives and Directories

[setdevice]

Changes to another available drive. The menu bar lists drives A: to G:. For example, "**A:**" changes to the floppy disk drive and "**C:**" to the first partition of the internal hard disk drive.

ChangeDir

Changes directories within a drive. This command also can be used to change to catalogs (*.CATL) or BOOTs (*.BOOT); for example, the command

```
ChangeDir \DemoFiles\Commands
```

changes within the active drive from the root directory via the "DemoFiles" directory to the "Commands" subdirectory.

Being on partition C: it is possible to change the current directory of another partition e.g. D:

```
ChangeDir D:\example
```

With the exception of catalogs (which load immediately) and BOOTs (which activate the ReBoot command), a faster way to navigate between directories is to left-click on the desired directory.

ChangeDevDir

This command changes to a directory on the specified drive. For example,

```
ChangeDevDir C:\DemoFiles\Commands
```

changes to drive "C:" and from the root directory via the "DemoFiles" directory to the "Commands" subdirectory.

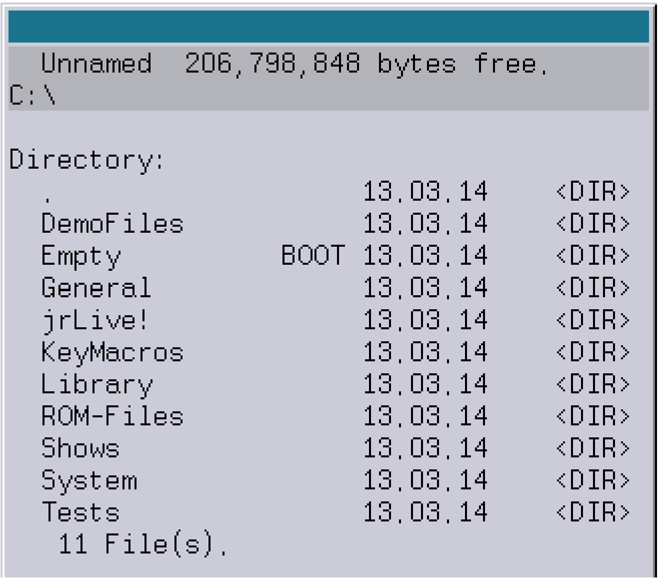
Displaying Directories and Files

The *Mark 2* has two commands that can be used to display the contents of drives, "Dir", or of the picture memory, "LIst". The results are displayed in rows as a list in the FILE window. Both commands are executed immediately when activated by the mouse or associated function key.

[pickmenu]
Switches from the list back to the FILE view.
Use the tab key to toggle between the list and FILE views.

Dir

When this command is selected from the menu bar, the contents of the current directory are listed with the names of the files, their creation date and size:



```
  Unnamed  206,798,848 bytes free.
C:\

Directory:
  .                    13.03.14    <DIR>
  DemoFiles            13.03.14    <DIR>
  Empty           BOOT 13.03.14    <DIR>
  General              13.03.14    <DIR>
  jrLive!              13.03.14    <DIR>
  KeyMacros            13.03.14    <DIR>
  Library              13.03.14    <DIR>
  ROM-Files            13.03.14    <DIR>
  Shows                13.03.14    <DIR>
  System               13.03.14    <DIR>
  Tests                13.03.14    <DIR>
   11 File(s).
```

If the command "Dir" is entered via the keyboard and followed by a space, additional options can be used to restrict the output to a specific directory "<dirspec>" or a specific file.
These options can be combined as required:

`/Subdir`
also lists the contents of the subdirectories

`/Date`
displays the creation date in addition to the name

`/Time`
displays the creation time in addition to the name

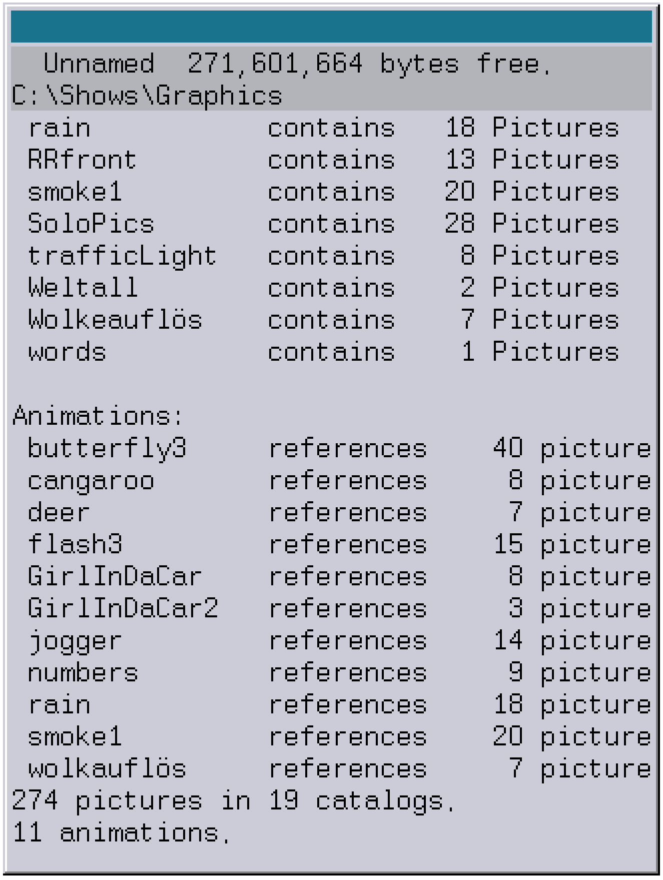
`/Length`
displays the file size in bytes in addition to the name

`/Protection`
displays the file attributes in addition to the file/directory names and provides information on what can be done with the data. The order of the attributes is fixed:

- ***r*** indicates that the file is readable.
- ***w*** indicates that the file is writable. This attribute can be changed using the Protect/UnProtect commands.
- ***d*** indicates that the file is deletable. This attribute can be changed using the Protect/UnProtect commands.
- ***p*** indicates that the files are protected against unauthorized access. This protection is only effective in conjunction with a password (DiskPassWd). If files with this attribute are copied, the protection is lost for the copy.
- ***c*** indicates that the files cannot be copied ([psave], /NoSaveAble option).

LIst

This command displays the contents of the picture memory, i.e. the catalogs (*.CATL) and animations (*.ANI) that are currently loaded, and the number of pictures or references they contain:



```
  Unnamed  271,601,664 bytes free.
C:\Shows\Graphics
 rain            contains   18 Pictures
 RRfront         contains   13 Pictures
 smoke1          contains   20 Pictures
 SoloPics        contains   28 Pictures
 trafficLight    contains    8 Pictures
 Weltall         contains    2 Pictures
 Wolkeauflös     contains    7 Pictures
 words           contains    1 Pictures

Animations:
 butterfly3      references    40 picture
 cangaroo        references     8 picture
 deer            references     7 picture
 flash3          references    15 picture
 GirlInDaCar     references     8 picture
 GirlInDaCar2    references     3 picture
 jogger          references    14 picture
 numbers         references     9 picture
 rain            references    18 picture
 smoke1          references    20 picture
 wolkauflös      references     7 picture
274 pictures in 19 catalogs.
11 animations.
```

Type "List" followed by a space for the FILE window to display all of the picture catalogs loaded in the picture memory for selection. Choose a catalog to list the pictures it contains.

Three commands are available that affect the listing output while the listing is being executed:

`[Halt]`
If the list to be output is too long to be displayed fully in the window, the output will scroll to the end of the list. The scrolling can be interrupted with this command.

`[Continue]`
This command will resume the output of a list previously stopped with [Halt].

`Exit`
This command cancels the output of the list. The display will return to the FILE view.

Finding, Copying, Deleting, Renaming Directories and Files

Find

Parameter: `<filespec>`

This command searches drives for stored data. The name of the desired file or directory must be specified.
Wildcard characters can be used; for example, the command

```
Find t*.ani
```

searches the current drive for animation files (*.ani) whose file names begin with "t".

Copy

Parameters: `<source> <dest>`
Options: `/Quiet /Subdir /Overwrite /NewerOnly`

Use this command to copy directories and files, for example when exchanging them between DSPs. The source must be stated, including the drive letter where applicable. If no destination is stated, the files will be copied into the current directory.

`<source>`
Specifies the files or directories to be copied. Wildcard characters (*,?) can be used in any part of the name to select multiple files and directories.

`<dest>`
Specifies the destination for the copy operation. The copies may be renamed in the process.

`/Quiet`
less information is displayed during copying to speed up the process.

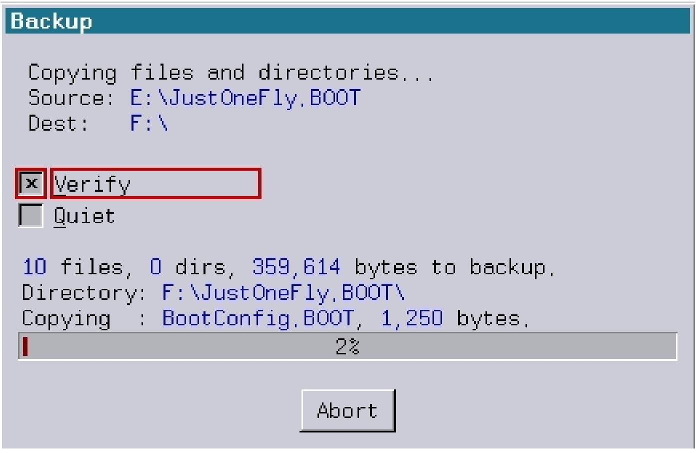
`/Subdir`
This option will also copy the subdirectories of the specified file/directory (necessary for catalogs and boots).

`/Overwrite`
This option overwrites already existing files with the same name.

`/NewerOnly`
This option lets you copy only those files that do not already exist, or those files that are newer than existing files at the destination.

Backup

Parameters: `<source> <dest>`
Option: `/Relpath`



Backup

```
Copying files and directories...
Source: E:\JustOneFly.BOOT
Dest:   F:\

☒ Verify
□ Quiet

10 files, 0 dirs, 359,614 bytes to backup.
Directory: F:\JustOneFly.BOOT\
Copying  : BootConfig.BOOT, 1,250 bytes.
2%
```

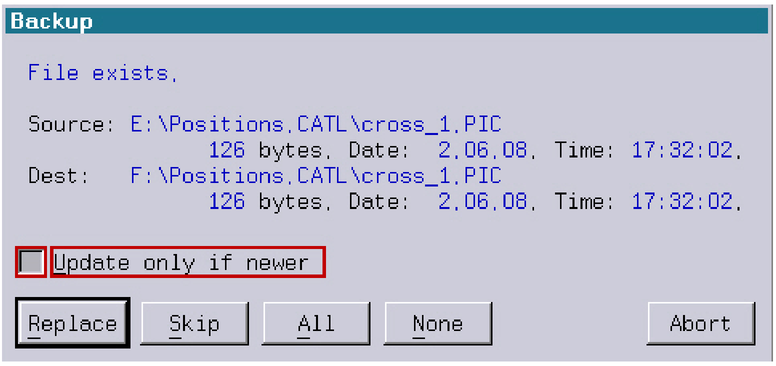
This command copies files and directories together with their subdirectories. The files are automatically verified after the copy operation if not disabled by the user.
Once the backup has started a status window opens up where you can see the progress and disable the verifying of files after copying and enable the option “Quiet” to speed up the process.

`<source>`
Specifies the files and/or directories to be copied.

`<dest>`
Specifies the destination directory for the copy operation. When copying, the complete path of the source file is appended to the desired destination directory.

`/Relpath`
Suppresses the creation of an absolute directory structure (absolute path) in the destination directory. The directory structure created in the destination directory starts with the current directory (relative path).

If one or more files with the same name already exist in the destination directory, you will be asked how these files are being handled:



Backup

```
File exists.

Source: E:\Positions.CATL\cross_1.PIC
            126 bytes. Date:  2.06.08. Time: 17:32:02.
Dest:   F:\Positions.CATL\cross_1.PIC
            126 bytes. Date:  2.06.08. Time: 17:32:02.

□ Update only if newer
```

`Replace`:	the existing file is replaced with the one which is to be copied for backup
`Skip`:	the existing file is not replaced
`All`:	existing files are always replaced during backup if files which are to be copied have the same name
`None`:	existing files are never replaced

LZIP

Parameters: `<filename> [<newname>]`

This command packs one or more files into a single file.
An example of use would be exchanging files between different Lasergraph DSPs.

The directory structure and the names of the packed files are preserved.
This helps avoiding problems with differing naming conventions when storing data on media formatted under MS-DOS™.

The size of the resulting file is reduced by using a proprietary compression algorithm.

To unpack LZip files on a Lasergraph DSP the command "UNLZIP" can be used.
It is also possible to unpack LZip files on a PC running Windows™ using the tool "UnLZip.exe".

`<filename>`
Specifies the files to be packed. Wildcard characters (`*`, `?`) can be used to specify multiple files.

`<newname>`
This parameter determines the name of the resulting file. If no extension is specified the extension "LZip" will be added automatically. A destination path may be specified for the resulting file.

If `<newname>` is not specified, the resulting file will get the same name as the source file or directory with the extension "LZip".

Examples

```
LZIP a*.catl acat.lzip
```

packs all picture catalogs in the current directory whose names start with "a" into a single file called "acat.lzip".

```
LZIP *.* files\backup.lzip
```

packs all files, directories and subdirectories in the current directory into the file "backup.lzip" in the "files" subdirectory.

```
LZIP MyShow.BOOT
```

creates a single file called "MyShow.LZip" containing all files of the directory MyShow.BOOT.

UNLZIP

Parameters: `<filename> [<newname>]`

This command extracts all files and directories from an LZip file.

`<filename>`
Specifies the name of the LZip file containing the files and directories to be extracted.

`<newname>`
Specifies the name of the extracted file.
If the LZip file contains more than one file `<newname>` should not be specified.

DELete

Parameter: `<filespec>`

This command deletes directories and files. Wildcard characters (*,?) can be used in any part of the name to select multiple files and directories.

`/Subdir`
This option also deletes all subdirectories. Use this command with care, as the deletion cannot be undone.

REName

Parameters: `<filename> <newname>`

This command renames directories and files. Do not forget the file extension when specifying the new name. Otherwise, the *Mark 2* will no longer be able to correctly associate the renamed file.
The * wildcard character may be used, for example

```
      REName Images.CATL Pictures.*
```

will rename "Images.CATL" in the current directory to "Pictures.CATL".

Creating Directories

MakeDir

Parameter: `<directory>`
This command creates subdirectories in the current directory. A new subdirectory may in turn contain further subdirectories and files.

`<directory>`
Specifies the name of the new directory.

Naming Partitions

DiskName

Parameter: Drive letter: `<diskname>`
This command assigns names to the available partitions in addition to the internally-assigned drive letters:

```
DiskName E:oscar
```

This command assigns the name "oscar" to the logical drive E:.

Protecting Data

The *Mark 2* offers a number of options to protect files.

Protect

Parameter: `<filespec>`
Option: `/Subdir`

This command protects the content of individual directories and files from changes or deletion.

`<filespec>`
Specifies the files or directories to be protected. Wildcard characters (`*`, ?) can be used in any part of the name to select multiple files and/or directories.

Unprotect

Parameter: `<filespec>`
Option: `/Subdir`
This command unprotects the content of individual directories (if protected).

`<filespec>`
Specifies the files or directories to be unprotected. Wildcard characters (`*`, ?) can be used in any part of the name to select multiple files and/or directories.

ProtDisk

Parameter: `<device>`

This command protects the content of the entire partitions from changes or deletion.

`<device>`
Drive letter of the partition to be protected.

UnProtDisk

Parameter: `<device>`

This command unprotects the content of the entire partitions (if protected).

`<device>`
Drive letter of the partition to be unprotected.

DiskPassWd

This command protects individual local partitions and their contents against unauthorized access. The protection is only effective in conjunction with the locking of files ("DiskPassWd.LockFiles"). When copying this data to MS-DOS™-formatted data media or network drives, the protection is lost.

DiskPassWd.Enter

Parameter: `<device>`
This command provides access to password-protected data.

`<device>`
Drive letter of the partition that you would like to access by entering a password.

DiskPassWd.Install

Parameter: `<device>`
Use this command to assign a password to protect a partition.

`<device>`
Drive letter of the partition to be password-protected.
A password can only be installed if you have access to the drive. It may be necessary to enter the old password first using "DiskPassWd.Enter"!

DiskPassWd.DeInstall

Parameter: `<device>`
This command removes the password protection of a partition after entering the password with "DiskPassWd.Enter".

DiskPassWd.LockFiles

Parameter: `<filespec>`
Option: `/Subdir`

Locks files on partitions protected by a password.

`<filespec>`
Specifies the names of the files to be protected. Wildcard characters (`*,?`) can be used in any part of the name to select multiple files and/or directories.

`/Subdir`
The option “Subdir” locks all subdirectories, too.

DiskPassWd.UnLockFiles

Parameter: `<filespec>`
Option: `/Subdir`

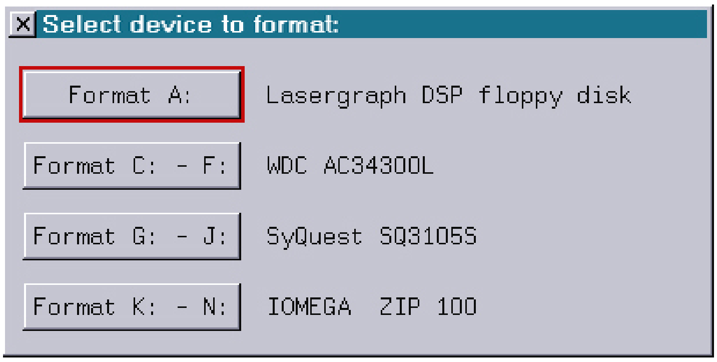
This command unlocks previously-locked files and directories. The correct password must be entered via "DiskPassWd.Enter" before the unlock operation can take place.

`<filespec>`
Specifies the names of the files to be unlocked. Wildcard characters (`*`, `?`) can be used in any part of the name to select multiple files and/or directories.

Formatting Data Media

The formatting of data media such as floppy disks, ZIP disks and hard disk drives is simple and convenient. The operation is based on clear, easy-to-use dialogs.

Enter the command `format` directly or select it from the menu bar, ("FORMAT").
Using a *Mark 2* workstation the following window with a selection menu will appear:



Lasergraph DSP floppy disk
WDC AC34300L
SyQuest SQ3105S
IOMEGA ZIP 100

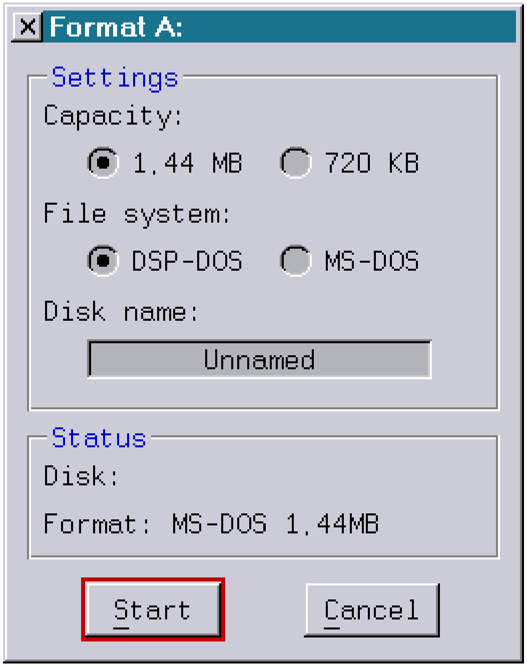
This window displays all of the drives connected to the *Mark 2* workstation. Simply select the required media.

Hint:
This selection window will not be displayed using the *Mark 2* compact.

Formatting a Floppy Disk

(if a floppy drive is installed)

To format a floppy disk, click the "Format A:" button. A window will be displayed which is divided into two areas - "Settings" and "Status". The "Settings" area provides a number of options related to the format operation. The "Status" area shows the current status of the floppy disk:



Settings

Capacity:

● 1.44 MB ○ 720 KB

File system:

● DSP-DOS ○ MS-DOS

Disk name:

Unnamed

Status

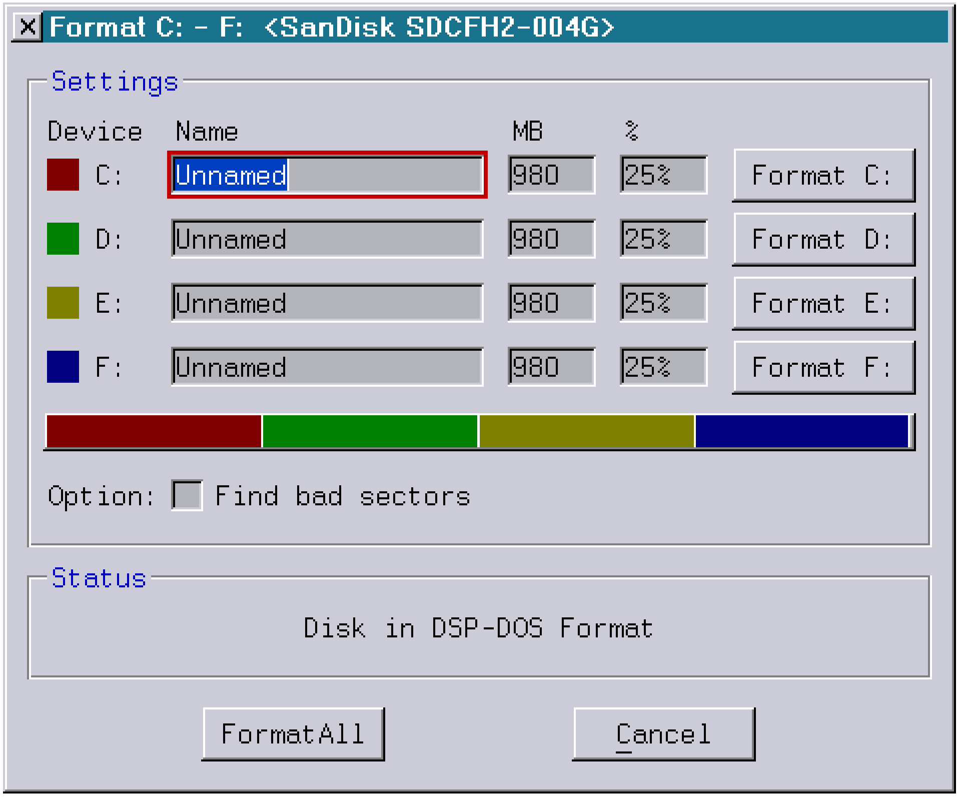
Disk:

Format: MS-DOS 1.44MB

"Capacity"	The system generally will automatically recognize the capacity of the floppy disk to be formatted, i.e. 1.44MB or 720KB. If not, the appropriate capacity can be selected manually.
"File system"	"DSP-DOS" format is selected by default; "MS-DOS" can be selected if required.
"Disk name"	A name for the floppy disk can be entered in this field.
"Start"	Starts the format operation. A confirmation prompt will appear if the floppy disk is already formatted. The progress of the format operation is indicated in percent in the "Status" field. "Cancel" will abort the operation at any time.

Formatting and Partitioning Hard Disk Drives and Removable Media

Click the appropriate button in the selection menu, e.g. "Format C:-F:".



Format C: - F: <SanDisk SDCFH2-004G>

Settings

Device	Name	MB	%
🟥 C:	Unnamed	980	25%
🟩 D:	Unnamed	980	25%
🟨 E:	Unnamed	980	25%
🟦 F:	Unnamed	980	25%

Option: □ Find bad sectors

Status

Disk in DSP-DOS Format

The window shows the current status of the media, i.e. existing partition sizes and their names will be displayed. The bar under the input fields provides a visualization of the existing partitions in various colors: red for the first, green for the second, yellow for the third, blue for the fourth partition, and gray for unused space.

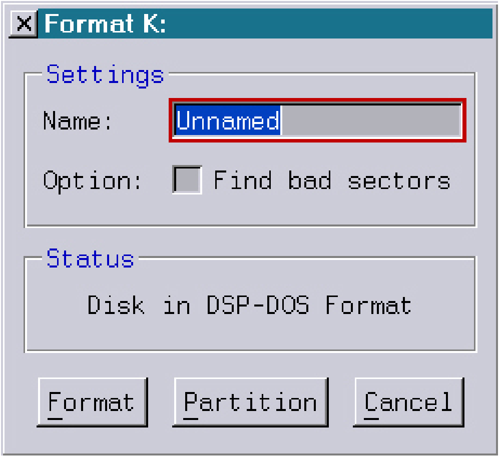
If the medium has not been partitioned or is partitioned in a format other than DSP-DOS such as MS-DOS, suitable settings will be suggested automatically.
Existing partitions may be formatted individually.

The "Settings" window offers a variety of parameters that can be changed by the user. The names and sizes of the partitions can be changed in the appropriate fields. Please observe the following in this regard:

- The "Name" field must not contain special characters and must not exceed 14+4 characters (DSP naming convention).
- As a rule, the capacity of the media must be taken into consideration when determining the partition sizes. The sum of the partition sizes may not exceed the capacity of the media.
- At least one partition with size>0 must be created and an empty partition (size=0) may not be followed by a partition with size>0.

- The bars show a colored representation of the current partition settings. The partition sizes can also be changed here by dragging the lines with the mouse. New partitions can be added with a double-click.
- The medium may also be optionally checked for defective sectors.
- "Format All" starts the formatting operation using the selected settings. A confirmation prompt will appear if the medium is already formatted.
- "Cancel" will abort the operation at any time.

The following window will open if you enter the command on the input line together with a drive letter, such as "FORMAT K:", to format an existing partition or a complete medium:



Format K:

Settings

Name: Unnamed

Option: □ Find bad sectors

Status

Disk in DSP-DOS Format

A name can be entered and the "Find bad sectors" option selected if desired.
"Format" starts the format operation. Select the "Partition" button to open the partitioning window (see "Format C: to F:").

Setting up Network Connections

```
SUBST drive:  \\computer\share
```

This command connects a volume temporarily.

For example: `subst  E:  \\192.168.1.90\data`

This setting is active until the Lasergraph DSP is turned off. To cancel the connection, just enter **`subst drive:`** without a network path.

Loading Data

[load] is the common term for a group of commands with which a variety of data can be loaded into the main memory. After invoking one of these commands, drives and directories can be changed by selecting them with the mouse. In the current directory, only those files with the appropriate extension will be displayed.

LoadAllcat LoadCatalog LoadPicture LoadFilm LoadScript LoadKeyconf LoadCOmpr ReBoot LoadMapconf LoadBankconf LoadBeamTab LoadGlobals LoadColors LoadANI LoadDGC LoadDGCCurrent LoadSuperPreview LoadILDA LoadSharedVars LoadFaderValues LoadArraY MergeScript

LoadAllcat

Options: `/Overwrite /Quiet`

Loads all pictures into the picture memory from all catalogs in the current directory, sorted by catalog, with the exception of pictures already loaded.

`/Overwrite`

Loads all pictures and overwrites existing ones, if in memory.

`/Quiet`

Less information is displayed to speed up the process.

LoadCatalog

Parameters: `<cfilename> <catalog>`
Options: `/Overwrite /Quiet`

Loads all pictures of a catalog to be selected (*.CATL) into the picture memory, with the exception of pictures already loaded. It may be specified whether the pictures

- are loaded with the source catalog,
- are to be added to an already-loaded catalog or
- a new catalog is to be opened in the picture memory.

`<cfilename>`
Name of the catalog to be loaded.

`<catalog>`
This parameter is optional. It specifies the name of the catalog in the picture memory to which the pictures are to be loaded. The FILE window offers all of the catalogs loaded in the picture memory for selection.
If you do not specify this parameter, the catalog will be created in the picture memory if it is not already present; pictures not yet loaded will be added to this catalog in the picture memory.

Hints:

- Left-clicking onto a catalog name in the FILE window without an activated command will also cause this catalog to be loaded.
- If you select a catalog loaded in the picture memory for this parameter, the pictures of the catalog to be loaded will be added to this one if they are not already in it.
- If you specify a catalog name in this parameter that does not exist in the picture memory, the catalog will be created there.

`/Overwrite`
To replace already-loaded pictures of a catalog in the picture memory, please use this option.

`/Quiet`
Less information is displayed to speed up the process.

LoadPicture

Parameters: `<cfilename> <pfilename> <catalog> <picture>`

Loads a selected picture (*.PIC). You must specify whether the picture

- is to be added to an already-loaded catalog or
- a new catalog is to be opened in the picture memory.

`<cfilename>`
Catalog from which an individual picture is to be loaded.

`<pfilename>`
Name of the picture to be loaded.

`<catalog>`
Name of the catalog in the picture memory to which the picture is to be loaded

`<picture>`
Name under which the picture is to be loaded into the picture memory.

LoadFilm

Parameters: `<ffilename> <film>`
Option: `/Overwrite`

Loads a film (*.FILM) into one of the four film memories. At the same time, all of the pictures (*.PIC) used in the film are loaded into the picture memory (if not already existing) - insofar as the appropriate picture catalogs (*.CATL) are located in the same directory as the film to be loaded or can be found in C:\General.
Pictures stored in "Compressed CATalogs" are only loaded along with a film if the name of the "Compressed CATalog" and that of the film are identical. If they are not the same, load the picture catalog first with "LoadCOmpr", then load the film.

`<ffilename>`
Determines the film to be loaded. Select the desired film in the FILE window.

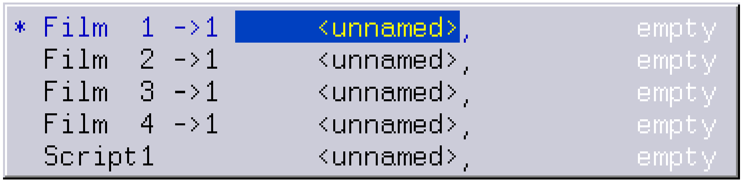
`<film>`
Determines the film memory into which the content of the desired film will be loaded. The buttons Film1 to Film4 are available.

Hints:
Films can also be loaded

- in the FILE window by selecting the file name with the left mouse button, if no command is active. The command will appear on the input line with the current film memory as a parameter, the cursor will be positioned on the first character of this parameter. The film memory can be changed by selecting a different one ("Film1" to "Film4").

- by double-clicking on the file name. The selected film will then be loaded directly to the current film memory (marked by an asterisk in the film box).

- in the start window via the film box by clicking one of the name fields.



```
* Film  1 ->1      <unnamed>,          empty
  Film  2 ->1      <unnamed>,          empty
  Film  3 ->1      <unnamed>,          empty
  Film  4 ->1      <unnamed>,          empty
  Script1          <unnamed>,          empty
```

LoadScript

Parameters: `<sfilename>`
Option: `/Overwrite`

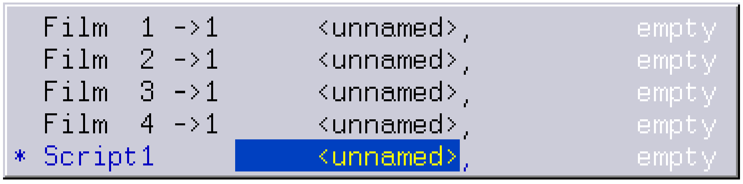
Loads a selected timescript (*.TSCR) into memory.

`<sfilename>`
File name of the timescript to be loaded.

Hints:
Timescripts can also be loaded

- in the FILE window by left-clicking the file name when no command is active.

- in the start window via the film box by selecting the name field.



```
  Film  1 ->1      <unnamed>,          empty
  Film  2 ->1      <unnamed>,          empty
  Film  3 ->1      <unnamed>,          empty
  Film  4 ->1      <unnamed>,          empty
* Script1          <unnamed>,          empty
```

LoadKeyconf

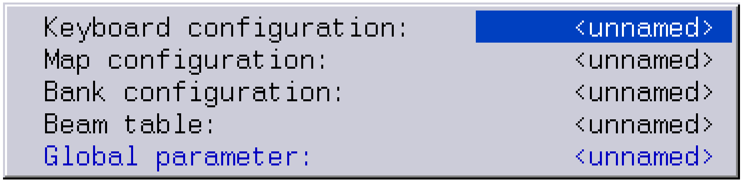
Parameter: `<kfilename>`
Option: `/Overwrite`

Loads a configuration for the key assignments of the MIDI keyboard (*.KCFG).

`<kfilename>`
This parameter states the file name of the keyboard configuration to be loaded.

Hints:
Keyboard settings can also be loaded

- in the FILE window by left-clicking the file name when no command is active.
- in the start window via the "Config" window by clicking on the name field behind "Keyboard configuration.



```
Keyboard configuration:     <unnamed>
Map configuration:          <unnamed>
Bank configuration:         <unnamed>
Beam table:                 <unnamed>
Global parameter:           <unnamed>
```

LoadCOmpr

Parameter: `<comprname> <catalog>  <picture>`
Option: `/Overwrite`

This command loads all catalogs, animations and individual images of a "Compressed CATalog" (*.CCAT) into the picture memory – with the exception of pictures previously loaded.

A "Compressed CATalog" is a file that contains all picture catalogs and animations that were loaded in the picture memory at the time of its creation in compressed form. "Compressed CATalogs" are created automatically when creating a BOOT. They can be created manually using the ⇨ "SaveCOmpr" command.

`<comprname>`
Name of the "Compressed CATalog" to be loaded.

`<catalog> <picture>`
These parameters are necessary when you intend to load only one Catalog or Picture from the compressed Catalog.

When saving pictures of a film manually in a "Compressed CATalog" (*.CCAT) please ensure that the catalog has the same name as the film so that it will automatically be loaded together with the film. If the archived picture catalog and the corresponding film have different names, load the "Compressed CATalog" first, followed by the film.

`/Overwrite`
overwrites all in memory existing pictures with the same name.

Hint
Compressed CATalogs can also be loaded in the FILE window by selecting them with the left mouse button.

ReBoot

Parameter: `<bootname>`
Options: `/NoBankConf /NoBeamTab /NoGlobals /Overwrite`

Loads a BOOT (*.BOOT).
A BOOT is a directory created with the "CreateBoot" command that contains all of the information loaded at the time of creation:
Loaded pictures and animations are stored in an archive file with the name "BootCatalogs.CCAT".
Loaded film sequences, the timescript, global parameters, the keyboard and bank configuration, as well as the beam table are stored in this directory. Files previously saved retain their original names. New information that has not been saved yet is named after the BOOT.

`<bootname>`
Name of the BOOT to be loaded.
Note: `<bootname>` may also contain a path, e.g. "ReBoot C:\Shows\xyz "

`/NoBankConf`
Bank configuration settings that may be contained in the BOOT will not be loaded together with the BOOT.

`/NoBeamTab`
Beam table configuration settings that may be contained in the BOOT will not be loaded together with the BOOT.

`/NoGlobals`
Global parameter settings that may be contained in the BOOT will not be loaded together with the BOOT.

`/Overwrite`
This option removes all previously loaded information, even if it has been edited but not yet saved. Loaded films, the content of the picture memory, advanced parameters, etc. will be replaced. Unsaved changes will be lost.

Hint:
A ReBoot can be invoked by double-clicking on the BOOT name in the FILE window.

LoadMapconf

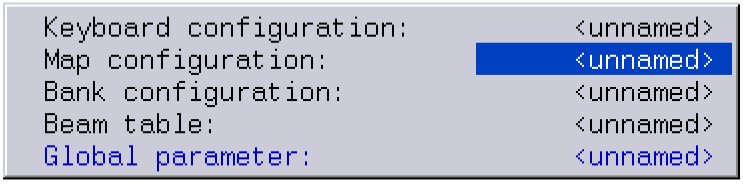
Parameter: `<mfilename>`
Option: `/Overwrite`

Loads a configuration file that regroups the output channels of the bank (*.MCFG).

`<mfilename>`
File name of the map configuration to be loaded.

Hint:
Map configurations can also be loaded

- in the FILE window by left-clicking the file name when no command is active.
- in the start window via the "Config" window by clicking on the name field behind "Map configuration".



```
Keyboard configuration:       <unnamed>
Map configuration:            <unnamed>
Bank configuration:           <unnamed>
Beam table:                   <unnamed>
Global parameter:             <unnamed>
```

LoadBankconf

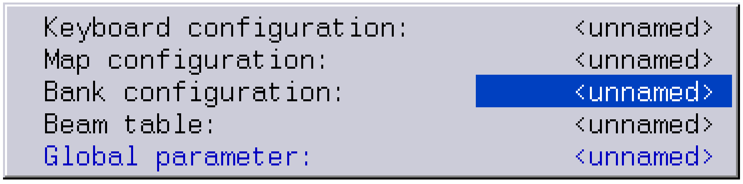
Parameter: `<bfilename>`
Option: `/Overwrite`

Loads a configuration for the output channels of the bank (*.BCFG).

`<bfilename>`
File name of the bank configuration to be loaded.

Hints
Bank configurations can also be loaded

- in the FILE window by left-clicking the file name when no command is active.
- in the start window via the "Config" window by selecting the name field behind "Bank configuration".



```
Keyboard configuration:       <unnamed>
Map configuration:            <unnamed>
Bank configuration:           <unnamed>
Beam table:                   <unnamed>
Global parameter:             <unnamed>
```

LoadBeamTab

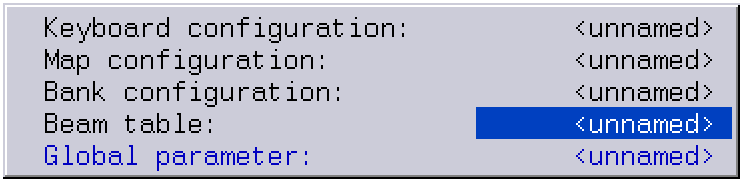
Parameter: `<bt fname>`
Option: `/Overwrite`

Loads a beam table (*.BTab). A beam table contains the names and positions of beams that can be addressed by the film.

`<bt fname>`
File name of the beam table to be loaded.

Hints:
Beam tables can also be loaded

- in the FILE window by left-clicking the file name when no command is active.
- in the start window via the "Config" window by selecting the name field behind "Beam table:".



```
Keyboard configuration:          <unnamed>
Map configuration:               <unnamed>
Bank configuration:              <unnamed>
Beam table:                      <unnamed>
Global parameter:                <unnamed>
```

LoadGlobals

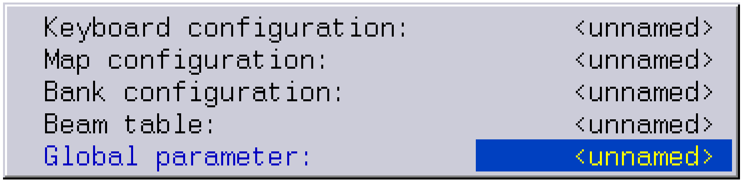
Parameter: `<gp name>`
Option: `/Overwrite`

Loads a configuration for the global parameters (*.GPAR). Global parameters contain the settings that control the output of the 4 films during a show, with the exception of the primary colors, ⇨ screen parameters and advanced parameters.

`<gp name>`
File name of the globals to be loaded.

Hints:
Global parameters can also be loaded

- in the FILE window by left-clicking the file name when no command is active.
- in the start window via the "Config" window by selecting the name field behind "Global parameter:" with the left mouse button.



```
Keyboard configuration:          <unnamed>
Map configuration:               <unnamed>
Bank configuration:              <unnamed>
Beam table:                      <unnamed>
Global parameter:                <unnamed>
```

LoadColors

Parameters: `<ffilename> <film>`
Option: `/Overwrite`

Loads a color table (*.COL). The film for which the color table is to be loaded can optionally be specified as a parameter.

`<ffilename>`
Primary color table to be loaded.

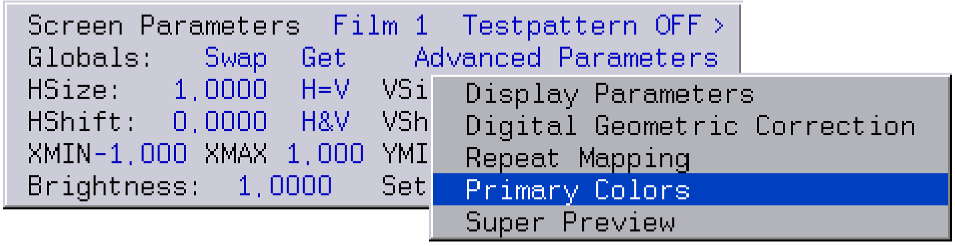
`<film>`
Film memory for which the color table is to be loaded. The buttons Film1 to Film4 are available.

Selection `<film>`
Makes the memory of the current film available. If you do not select a film memory, the color table will be loaded to the current film's memory (marked by an asterisk in the film box).

Hints:
Primary color tables can be loaded faster:

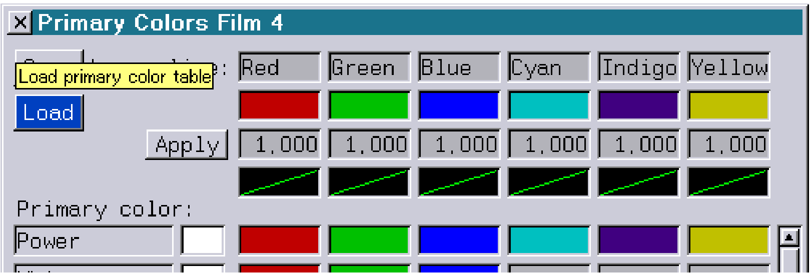
- in the FILE window, if no command is active, by selecting the file name with the left mouse button.
  The complete command will appear on the input line with the current film memory as a parameter; the cursor will be positioned on the first character of this parameter.
  The assignment can be changed by selecting a different film memory ("Film1" to "Film4").
  The selected color table can be loaded directly and assigned to the current film by double-clicking on its file name.

- within the "Primary colors" window via the "Load" button.
  The FILE window will be displayed to allow the user to select the desired color table and the FILE command "LoadColors" will be activated.
  The "Primary colors" window can be accessed in the Start screen via the "Advanced Parameters" ⇨ "Primary colors" popup menu:



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:   Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSi
HShift:   0.0000  H&V  VSh
XMIN-1.000 XMAX 1.000 YMI
Brightness:  1.0000   Set
```

The "PrimaryColorTable" for the current film will be opened automatically:



LoadANI

Parameter: `<afilename> <afilename>`
Option: `/Overwrite`

Loads an animation (*.ANI). An animation is the grouping of multiple pictures to a picture sequence in the animation editor (⇨ "AnimEd"). An animation file does not contain copies of the pictures used; they are only referenced via catalog and picture names. When an animation is loaded, all required pictures are automatically loaded to the picture memory, provided the respective picture catalog is located in the same directory as the animation file or in the C:\General directory.

`<afilename>`
File name of the animation to be loaded. The name is optional. If no name is specified, all animations in the current directory will be loaded.

`<afilename>`
The second optional parameter determines the name under which the animation will be loaded into the main memory.

Hint
Animation files can also be loaded in the FILE window by selecting them with the left mouse button.

LoadDGC

Parameter: `<dgc name>`

Loads all Digital Geometric Correction sets of all films.

Note:
If a file is specified which has been saved with `SaveDGCCurrent` it is loaded into the first Digital Geometric Correction set of the first film (Film1).

LoadDGCCurrent

Parameter: `<dgc name>`

Loads the Digital Geometric Correction set which has been saved with `SaveDGCCurrent` into the currently selected Digital Geometric Correction set of the currently selected film.

Note:
If a file is specified which has been saved with `SaveDGC` the first set of the first film (Film1) is used.

LoadSuperPreview

Parameter: `<spv name>`

Loads the settings for the SuperPreview.

LoadILDA

Parameter: `<ILDA name>`

Loads ILDA frames.
This command has the same parameters and options as the command `COnvert.ILDAtoDSP`.

LoadSharedVars

Parameter: `<filename>`

This command loads the state of the shared variables from a file.

Note:
The command `LoadSharedVars` only changes the state of the shared variables saved in the file. Shared variables outside the saved range are not changed.

LoadFaderValues

Parameter: `<filename>`

This command loads the state of the faders from a file.

Note:
The command `LoadFaderValues` only changes the state of the faders saved in the file. Faders outside the saved range are not changed.

LoadArraY

Parameters: `<filename> [<dsp>]`

This command loads the variables array from a file.

`<filename>`
The filename may contain path specifications. If no path is specified, the array will be loaded from the current directory under the specified filename.

`<dsp>`
It is optional to specify the DSP whose array is to be loaded.
If no DSP is specified, DSP1 will be used.
Valid range: 1…4

Note:
The command `LoadArraY` only changes the elements of the array stored in the file.
Elements located outside the stored area will not be changed.

MergeScript

Parameter: `<sfilename>`

Combines the contents of two timescript files (*.TSCR). The timescript is appended onto the previously loaded one.

`<sfilename>`
File name of the timescript to be appended.

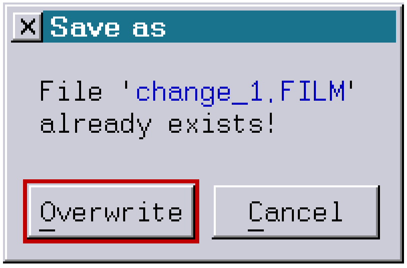
Saving Data

The *Mark 2* provides the two command groups **[save]** and **[psave]** for the saving of information.

Selecting source data
When a command from these groups is invoked, a FILE window with the relevant loaded files will be displayed. Select the files to be saved with a left-click.

Determining the destination
Next, the current directory of the current drive is displayed. Use the mouse to change drives and directories.

Overwrite?
A window will open in which you will have to click "Overwrite" to confirm your action if you select existing directories or files as the destination for changed data.


Save as

File 'change_1.FILM'
already exists!

[save] is the common term for a group of commands to save currently loaded or newly created data.

SaveAllcat SaveCatalog SavePicture SaveFilm SaveScript SaveKeyconf SaveCOmpr CreateBoot SaveMapconf SaveBankconf SaveBeamTab SaveGlobals SaveColors SaveANI SaveDGC SaveDGCCurrent SaveSuperPreview SaveILDA SaveSharedVars SaveFaderValues SaveArraY SaveSWF SaveSIPRO SaveAIFF

SaveAllcat

Options: `/UsedFilmX /NoGenerals /Overwrite`

Saves all loaded pictures of all catalogs in the picture memory under their current name.

`/UsedFilmX`
This option limits the save operation to the pictures used in the specified film. "/UsedFilm1" to "/UsedFilm4" may be selected.

`/NoGenerals`
This option excludes all pictures loaded from the "C:\General" directory, such as used fonts.

SaveCatalog

Parameters: `<catalog> <cfilename>`
Option: `/Overwrite /Quiet`

Saves all pictures of a loaded catalog (*.CATL).
It may be specified whether the pictures are to be:

- saved with the same catalog name,
- added to a saved catalog or
- added to a new catalog.

`<catalog>`
Name of the catalog to be saved which is currently loaded in the picture memory. The FILE window offers the catalogs loaded in the picture memory for selection.

`<cfilename>`
This parameter is optional. It specifies the name under which the pictures are to be saved. The FILE window offers all of the catalogs stored in the current directory for selection.
If this parameter is stated, the pictures will be added to existing catalogs. If no catalog exists with this name, one will be created.

If this parameter is not stated, all pictures of the selected catalog will be stored in the original catalog provided it was not renamed when loading. If the original catalog was renamed, the pictures will be saved in a new catalog that will be created in the current directory with the same name as in the picture memory.

`/Overwrite`
Overwrites the existing catalog.

`/Quiet`

Less information is displayed to speed up the process.

SavePicture

Parameters: `<catalog> <picture> <cfilename> <pfilename>`

Saves a picture (*.PIC) loaded in the picture memory in a new catalog and/or under a new name.

`<catalog>`
Name of the catalog loaded in the picture memory from which a picture is to be saved.

`<picture>`
Picture to be saved which is currently loaded in the picture memory.

`<cfilename>`
This parameter is optional. It specifies the name of the catalog in which the individual picture is to be saved:
If no catalog is specified, the picture will be saved in the original catalog under its old name. The old version of the file will be overwritten.
If a new catalog name is specified, this catalog will automatically be created in the current directory.

`<pfilename>`
This optional parameter can be specified if the destination name was specified earlier (<cfilename>). It specifies the name of the picture to be saved:
If no name is specified, the picture will be saved under its original name.
If a new name is specified, a new picture file with this name will be created in the desired catalog.

SaveFilm

Parameters: `<film> <ffilename>`

Saves a film (*.FILM). The film memory containing the material to be saved and the name under which the film is to be saved must be specified.

`<film>`
Selects the film to be saved. The buttons Film1 to Film4 are available for convenient selection.

Default<film>
To save the current film, copy the name of the film memory to the input line using the <film> button.

`<ffilename>`
File name under which the film is to be saved.

Default <ffilename>
If the film is to be saved under its previous name, copy its current name to the input line using the <ffilename> button.

SaveScript

Parameters: `<sfilename>`

Saves the timescript (*.TSCR).

`<sfilename>`
File name under which the timescript is to be saved.

Default <sfilename>
If the timescript is to be saved under its previous name, copy its current name to the input line using the <sfilename> button.

SaveKeyconf

Parameter: `<kfilename>`

Saves the configuration for the MIDI keyboard (*.KCFG).

`<kfilename>`
File name under which the current configuration of the MIDI keyboard is to be saved.
Default <kfilename>

If the keyboard configuration is to be saved under its previous name, copy its current name to the input line using the <kfilename> button.

SaveCOmpr

Parameter: `<comprname>`
Options: `/UsedFilmX /NoGenerals`

Saves the loaded pictures and animations in "Compressed CATalogs" (*.CCAT). Options can be used to determine whether all of the loaded pictures will be included.

`<comprname>`
File name of the "Compressed CATalog" in which the pictures and animations are to be saved.

`/UsedFilmX`
This option can be used to save only those pictures and animations that were used in one of the specified films. The buttons /UsedFilm1 to /UsedFilm4 are available. Multiple films may also be specified.
When saving pictures of a film in a "Compressed CATalog" (*.CCAT) please ensure that the catalog has the same name as the film so that it will automatically be loaded together with the film.

`/NoGenerals`
This option excludes all pictures loaded from "C:\General" when saving a Compressed CATalog.

CreateBoot

Parameter: `<bootname>`
Options: `/AutoPlay /NoBankConf /NoBeamTab /NoGlobals`
`/NoXGlobals /NoDGC /NoSuperPreview /Overwrite`

This command creates a BOOT (*.BOOT).

`<bootname>`
Name of the BOOT to be created.

Note:
`<bootname>` may also contain a path, e.g. "CreateBoot C:\Shows\MyShow".
In case the underlying path does not exist it will be created automatically.

AutoBoot
To load a BOOT automatically every time the computer is turned on, save it in the root directory of drive "C:" with the name "AutoBoot.BOOT".

`/AutoPlay`
If a timescript was loaded and active in the start screen of the film box (marked by an asterisk) during the creation of a BOOT, this option will invoke the timescript immediately after the BOOT has been loaded.

`/NoBankConf`
This option excludes the bank configuration when creating a BOOT.

`/NoBeamTab`
This option excludes the beam table when creating a BOOT.

`/NoGlobals`
This option excludes the global parameters and DGC when creating a BOOT. These include the film-related settings that affect the output of a film during the course of a show - with the exception of the Primary Colors.

Hint:
If the mirror or projector positions are changed, different projection surfaces used or similar changes are made to the setup of the system, the settings for the picture position and size, scanning parameters and geometric correction of the picture (DGC) must be adapted accordingly to ensure that the laser show will be displayed as intended.

To ensure that the parameters do not need to be individually adapted for each show, it is in principle better to save the "Global parameters", "BankConf" and "BeamTab" files separately on the hard disk drive of the *Mark 2*.

This way they are always available, and in the event of changes to the system the parameters only need to be adapted **once** and not individually for every show. The "/NoBankconf", "/NoBeamTab" and "/NoGlobals" options must be selected for this purpose when creating a BOOT.

Save the relevant *.BCFG, *.BTab and *.GPAR files separately in the "C:\System" directory under the names "SystemBankConf", "SystemBeamTab" and "SystemGlobals". The files saved in this manner are automatically loaded when starting the Mark 2, provided the BOOT does not contain the appropriate files.

`/NoXGlobals`
Settings for the DGC are saved in the BOOT, settings for the global parameters are not saved.

`/NoDGC`
Settings for the DGC are not saved in the BOOT.

`/NoSuperPreview`
Settings for the Super Preview are not saved in the BOOT.

SaveMapconf

Parameter: `<mfilename>`

This command saves the loaded or new settings for the redirection of the output channels of the bank.

`<mfilename>`
File name under which the configuration is to be saved. To replace the old configuration when saving the new version, click this button to copy the file name to the input line.

SaveBankconf

Parameter: `<bfilename>`

Saves the bank configuration (*.BCFG).

`<bfilename>`
File name under which the configuration is to be saved. To replace the old configuration when saving the new version, click this button to copy the file name to the input line.

SystemBankConf
This permits the loaded bank configuration to be declared as the standard configuration. The "SystemBankConf" will be loaded by default when booting or rebooting if no other configuration file is available in the BOOT. It must be saved in the "C:\System" directory for this purpose.

SaveBeamTab

Parameter: `<bt name>`

Saves the loaded or new beam table (*.BTab).

`<bt name>`
File name under which the configuration is to be saved. To replace the old configuration when saving the new version, click this button to copy the file name to the input line.

SystemBeamTab
This permits the loaded beam table to be declared as the standard configuration. The "System-BeamTable" will be loaded by default when booting or rebooting if no other beam table file is available in the BOOT. It must be saved in the "C:\System" directory for this purpose.

SaveGlobals

Parameter: `<gp name>`

Saves the loaded or new global parameters (*.GPAR).

The global parameters include the film-related picture output settings, with the exception of the PrimaryColorTables (⇨ Screen Parameter, Advanced Parameters).

`<gp name>`
File name under which the global parameters are to be saved.
To replace the old configuration when saving the new version, click this button to copy the file name to the input line.

SystemGlobals
This permits the loaded global parameters to be declared as the standard configuration. The "SystemGlobals" will be loaded by default when booting or rebooting if no other global file is available in the BOOT. It must be saved in the "C:\System" directory for this purpose.

SaveColors

Parameters: `<film> <ffilename>`
Option: `/Overwrite`

Saves the primary color table (*.COL, ⇨ "PrimaryColorTable") of a film.

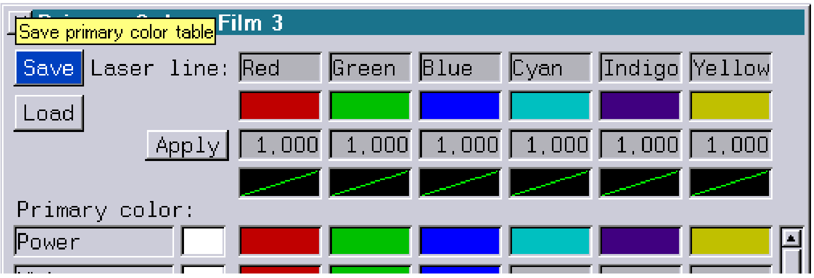
`<film>`
Selects the film for which the "PrimaryColorTable" is to be saved. The buttons Film1 to Film4 are available.
Default `<film>`
If the color table of the current film (marked by an asterisk) is to be saved, copy the film memory to the input line using this button.

`<ffilename>`
Name under which the color table is to be saved.

Default <ffilename>
If the color table is to be saved under its previous name, copy its current name to the input line using this button.

`/Overwrite`
This option forces an existing color table to be overwritten without prompting.

Hint:
PrimaryColorTables can also be saved in the "Primary Colors" (Advanced Parameters) window by using the "Save" button.



Note:
It is possible to store global color settings („Primary Colors“) for film 1 to 4 separately on the hard disk in the directory „C:\System“ as „SystemColors1.COL“ ... „SystemColors4.COL“. They will be loaded automatically for film 1 to 4.

Attention! The colors from the boot will then always be overwritten.

SaveANI

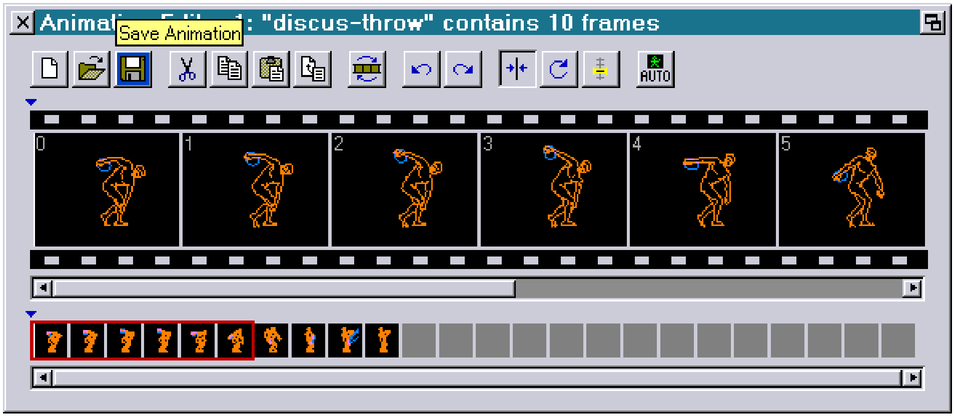
Parameters: `<afilename> <afilename>`
Option: `/Overwrite`

Saves an animation (*.ANI).

`1. <afilename>`
This parameter is optional. It specifies the file name of the animation to be saved. The FILE window offers all of the loaded animations.
If this parameter is not stated, all animations will be saved.
If this parameter is stated, only the selected animation will be saved.

`2. <afilename>`
This parameter is only available if an animation has already been selected. It states the file name under which the animation is to be saved. The FILE window offers all of the animations stored in the current directory for selection.

Hint
An animation can also be saved in the animation editor by using the "Save Animation" button.



SaveDGC

Parameter: `<dgc name>`

Saves all Digital Geometric Correction sets of all films (popup menu ‘Advanced Parameters’ in the ‘Globals’ windows of the *Mark 2* main screen) .

`<dgc name>`
Specifiy the DGC setting.

SaveDGCCurrent

Parameter: `<dgc name>`

Saves the currently selected Digital Geometric Correction set of the currently selected film (popup menu ‘Advanced Parameters’ in the ‘Globals’ windows of the *Mark 2* main screen) .

`<dgc name>`
Specifiy the DGC setting.

SaveILDA

Parameter: `<catalog> [<filename>]`
Options: `/3D /2D /TrueColor /IndexedColor /NoPrecompute`

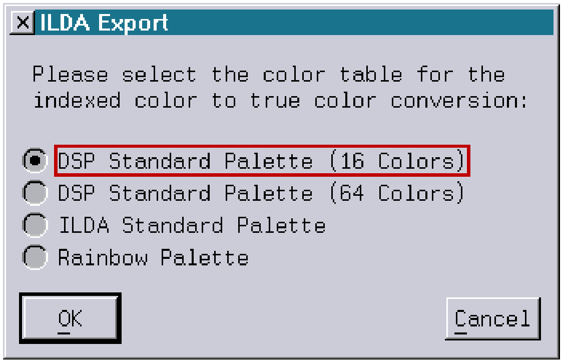
Converts the loaded catalog into ILDA format and saves the result under the specified file name on the hard disk. If no file name is specified, the name of the catalog is used as file name.

`/3D`
The pictures will be saved in the ILDA Format 0 (3D Coordinates with Indexed Color) resp. ILDA Format 4 (3D Coordinates with True Color).
For three-dimensional pictures this option will have no effect.

`/2D`
The pictures will be saved in the ILDA Format 1 (2D Coordinates with Indexed Color) resp. ILDA Format 5 (2D Coordinates with True Color).
For three-dimensional images this option will cause the Z-information to be lost.

If neither /3D nor /2D is specified the most suitable format for each image will be determined automatically.

`/TrueColor`
The pictures will be saved in the ILDA Format 4 (3D Coordinates with True Color) resp. ILDA Format 5 (2D Coordinates with True Color).
If one or more pictures do not contain TrueColor information, the following dialog will open to determine how the indexed colors are to be mapped to the TrueColors:


ILDA Export

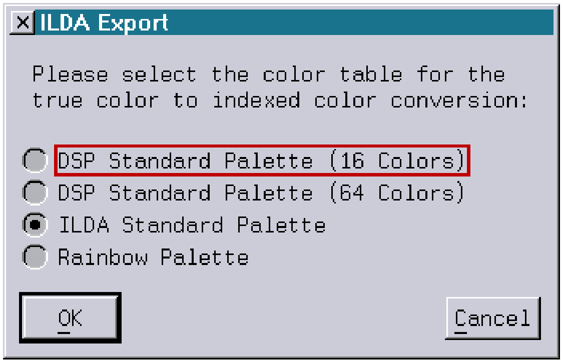
Please select the color table for the indexed color to true color conversion:

● DSP Standard Palette (16 Colors)
○ DSP Standard Palette (64 Colors)
○ ILDA Standard Palette
○ Rainbow Palette

`/IndexedColor`

The pictures will be saved in the ILDA Format 0 (3D Coordinates with Indexed Color) resp. ILDA Format 1 (2D Coordinates with Indexed Color).

If one or more pictures contain TrueColor information, the following dialog will open to determine how the TrueColors are to be mapped to the indexed colors.


ILDA Export

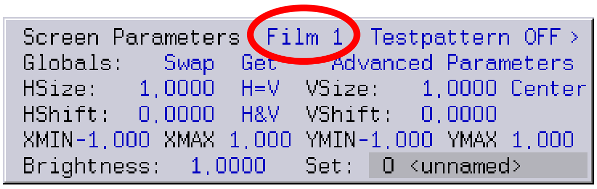
Please select the color table for the
true color to indexed color conversion:

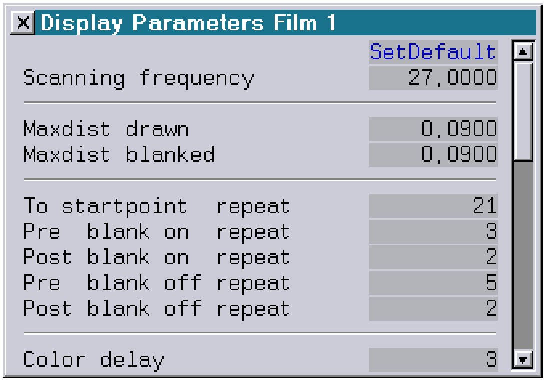
○ DSP Standard Palette (16 Colors)
○ DSP Standard Palette (64 Colors)
● ILDA Standard Palette
○ Rainbow Palette

If neither /TrueColor nor /IndexedColor is specified the most suitable format for each image will be determined automatically.

`/NoPrecompute`

Unless this option is selected the pictures will be precomputed according to the currently active scanning parameters prior to conversion into the ILDA format.


```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:   1.0000  H=V  VSize:    1.0000 Center
HShift:  0.0000  H&V  VShift:  0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:  1.0000   Set:  0 <unnamed>
```


Display Parameters Film 1

```
Scanning frequency          27.0000

Maxdist drawn                0.0900
Maxdist blanked              0.0900

To startpoint  repeat            21
Pre  blank on  repeat             3
Post blank on  repeat             2
Pre  blank off repeat             5
Post blank off repeat             2

Color delay                       3
```

The following scanning parameters will be used for precomputing:

```
Maxdist drawn
Maxdist blanked
--------------------
To startpoint  repeat
Pre  blank on  repeat
Post blank on  repeat
Pre  blank off repeat
Post blank off repeat
--------------------
Color delay
```

SaveSharedVars

Parameters: `<filename> [<first> [<last>]]`

This command saves the state of the shared variables to a file.

`<filename>`
The file name can also contain path information. If no path is specified the shared variables will be saved in the current directory.

`<first>`
Specifying the first shared variable which is to be saved is optional.
Possible value range: 0...63

`<last>`
Specifying the last shared variable which is to be saved is also optional.
Possible value range: 0...63

Note:
By specifying the shared variables which are to be saved using <first> and <last> the shared variables can be divided into several segments which can be administered separately.

SaveFaderValues

Parameters: `<filename> [<first> [<last>]]`

This command saves the state of the faders to a file.

`<filename>`
The file name can also contain path information. If no path is specified the fader values will be saved in the current directory.

`<first>`
Specifying the first fader value which is to be saved is optional.
Possible value range: 0...63

`<last>`
Specifying the last fader value which is to be saved is also optional.
Possible value range: 0...63

Note:
By specifying fader values which are to be stored using <first> and <last> the faders can be divided into several segments which can be administered independently.

SaveArraY

Parameters: `<filename> [<dsp> [<first> [<last>]]]`

This command saves the variables array to a file.

`<filename>`
The filename may contain path specifications. If no path is specified, the array will be saved in the current directory under the specified filename.

`<dsp>`
It is optional to specify the DSP whose array is to be saved.
If no DSP is specified, DSP 1 will be used.
Valid range: 1…4

`<first>`
It is optional to specify the first element to be saved.
Valid range: 0…9999

`<last>`
It is optional to specify the last element to be saved.
Valid range: 0…9999

Note:
Specifying the elements to be saved by <first> and <last>, enables you to save parts of the array as well.

SaveSWF

```
Parameters:  <cfilename>   *
             <filename>  <width>  <height>  <fps>  <line-width>
Options:     /Overwrite /Protect
```

This command saves pictures of a catalog in the Flash™ format and so it is possible to use laser pictures as Macromedia Flash™ preview on the PC.

Important Hint:
Catalogs and pictures must not be protected with the FILE command [psave] ("PSaveAllcat" resp. "PSavePicture").

`<cfilename>`
Specify the catalog that is to be converted.

`*`
All catalogs in the picture "Memory" will be converted.

`<filename>`
Enter a name for the Flash™ file.

`<width>`
Select the width of the SWF file in pixels: 320, 640, 800, 1024.

`<height>`
Select the height of the SWF file in pixels: 320, 640, 800, 1024.

`<fps>`
Select the number of "Frames per Second": 12, 24, 25, 30.

`<line-width>`
Select the line width: 0, 1, 1.5.

`/Protect`
Save the SWF file with protection.

SaveSIPRO

Parameters: `<film> <ffilename> <catalog>`

Creates a file for the SimpleProjector (Sipro) on the basis of a film. The Sipro is a simple laser control computer that can be used as an effect device in standalone mode or as an "intelligent" projector in conjunction with a *Mark 2*. If the film contains commands that cannot be interpreted by the Sipro, an error message will be issued when saving the file.

`<film>`
Selects the film memory with the content to be converted to a Sipro file. The buttons Film1 to Film4 are available.

Default <film>
Copies the number of the current film to the input line (marked by an asterisk in the start screen of the film box).

`<ffilename>`
Name of the Sipro file to be created.

`<catalog>`
This parameter is optional. It permits up to eight font catalogs to be specified for inclusion with the file. These fonts can be used later for running texts under external control.

SaveAIFF

Parameters: `<catalog> [<filename>]`
Options: `/NoMarker`

Converts the loaded catalog into a stream with 5 channels (X, Y, R, G, B) and writes it to an AIFF file. If no file name is specified, the name of the catalog is used as file name.

`/NoMarker`
Normally, "SaveAIFF" creates scene markers in the AIFF file. Scene markers allow the AIFF file to be cut with frame accuracy and provide better output in "LA.preview" or other visualizers.
However, if you want to edit the AIFF file with conventional audio software, for example to change the image size, the colors or the sampling rate, the scene markers might have a negative effect on the laser output.
For this special case you can prevent the output of markers with the option "/NoMarker".

As with the commands "SaveILDA" and "COnvert.DSPtoILDA", "SaveAIFF" also uses the scanning parameters of the currently selected Trickfilm for the conversion.

Note:
Not all systems support freely selectable sampling rates.
For example, if you want to output the AIFF file via a USB2AVB interface over AVB on a PHAENON accurate, the sampling rate has to be 48, 96 or 192 kHz.

[psave]

Parameter: `<comp name>`
Options: `/NoEdit /NoSaveAble`

This is the common term for a group of commands to save data with protection. A name for a computer must be specified upon which the protected file(s) can be loaded.
Corresponding "psave" commands are available for the following "save" commands:

see SaveAllcat	⇨	PSaveAllcat
see SaveCatalog	⇨	PSaveCatalog
see SavePicture	⇨	PSavePicture
see SaveFilm	⇨	PSaveFilm
see SaveScript	⇨	PSaveScript
see SaveCOmpr	⇨	PSaveCOmpr
see CreateBoot	⇨	PCreateBoot

`<comp name>`
This parameter is used to specify the name of the Mark 2 on which the file can be loaded.

Wildcard characters can also be used in the computer name, permitting the protected file to be played on multiple Mark 2 units. The "?" stands for one, the "*" for several characters.

For example, if "Laser*" was specified when saving, the file will be usable on Mark 2 units named *Laser*, e.g. *LaserAnimation, Laser01*, *Laser1234* or even *LaserWizard.*
If "*" is entered by itself as the computer name, the file will be usable on any Mark 2.
Instead of using the name of a Mark 2 for protection, it is possible to use the serial number of a Mark 2 system. The serial number can be found by typing in the command "info".
For this purpose the serial number has to be specified with a preceding "#" as the first parameter. If a show is for instance saved with the following command

```
      PCreateBoot #12345 Showname
```

this show can only be loaded on the Mark 2 with the serial number "12345".

```
      PSaveFilm Laser* Film3
```

With this command, Film 3 can only be used by Mark 2 units that have "Laser" as the beginning of their name.

`/NoEdit`
The data cannot be edited.

`/NoSaveAble`
This option prevents the data from being copied or saved. It may be loaded, output and edited on authorized computers. Changes cannot be saved, however.
Attention: This option is not available for network drives!

Hint:
As a rule, a copy of the unprotected file should be saved separately, as the "psave" protection cannot be removed.

Unloading Data from the Memory

[unload] is the common term for a group of commands with which specific data can be unloaded from the main memory.

UnLoadAllcat UnLoadCatalog UnLoadPicture UnLoadFilm UnLoadScript UnLoadANI UnLoadLib

UnLoadAllcat

Unloads all pictures and animations from the picture memory that are not used by films or animations.

UnLoadCatalog

Parameter: `<catalog>`

Unloads all pictures of the specified catalog (*.CATL) from the picture memory that are not used by films or animations.

`<catalog>`
Picture catalog to be unloaded from the picture memory. The FILE window offers all of the loaded picture catalogs for selection with the left mouse button.

UnLoadPicture

Parameters: `<catalog> <picture>`
Unloads a picture (*.PIC) from the picture memory if it is not being used by a film or animation.

`<catalog>`
Picture catalog containing the picture to be unloaded.

`<picture>`
Picture to be unloaded from the picture memory.

UnLoadFilm

Parameter: `<film>`

Unloads a film (*.FILM) from the main memory.

`<film>`
Selects the film to be unloaded. The buttons Film1 to Film4 are available for selection.
More than one film may be selected.

Default <film>
To remove the current film (marked with *), copy the name of the film memory to the input line using this button.

`/Overwrite`
If the film has been edited and not been saved, this option will although unload the film.

UnLoadScript

Unloads the timescript (*.TSCR) from the memory.

`/Overwrite`
If the script has been edited and not been saved, this option will although unload the script.

UnLoadANI

Parameter: `<afilename>`
Removes an animation (*.ANI) from the main memory if it is not being used by a film.

`<afilename>`
Selects the animation to be removed from the main memory. The FILE window offers all of the loaded animations for selection.

Hint:
If you do not select an animation, all animations that are not used by one of the films will be removed from the main memory.

UnLoadLib

This command removes all pictures of the *Mark 2 library* from the main memory, even if they are being used by a film.

Note: This may be useful if you want to edit the film on an older DSP system without the license to edit pictures from the *library*.

Conversion of Picture Files

The Mark 2 supports the import and export of picture files in a variety of formats.

Graphic Format Types

Graphic files must be available in vector formats. Bitmap formats are not suitable for laser output.
To import a graphic it is best to use the converting tool that is included in the Mark 2 Production Suite “proTize converter”.

Vector Formats

In vector graphics, a picture is described via the corners and edges of multiple polygons. Although properties such as for instance thickness, color, etc. can be stored for the lines, properties such as area fills or color transitions can only be recorded for polygons.

HP-GL / HP-GL2

This vector format is used frequently for graphics output via a plotter. Most plotters support the Hewlett Packard™ Graphics Languages "HP-GL" or "HP-GL2". As nearly all drawing programs support the HP format as a standard export format, Mark 2 also supports this format for the import of picture files.

Laser Graphics Formats

A laser computer projects a graphic by directing a laser beam from one picture point to the next with scanners. This takes place at such a high speed that the viewer does not see individual lines, but a complete image. As this corresponds to the principle of vector graphics, laser computers use vector-oriented formats to store picture data. The differences of the laser picture formats and the processes used for their projection lead to problems with their conversion, however.

Conversion issues

Most laser graphics are structured in such a way that the picture point output is transferred to the scanners at a predetermined frequency. As this frequency is generally not stored in the file to be converted, the conversion program cannot recognize which frequency is required. As a result, the user must set the correct frequency (⇨Display parameters).

Picture points

To ensure that the laser beam can be directed from one picture point to the next at optimal speed, with many laser systems it is necessary to ensure that the picture points are not too widely spaced when creating the graphic. This can be achieved by saving several picture points along a line as required. The Mark 2 does not need these intermediate points, as any additional points that might be required are inserted automatically.
Intermediate points do not adversely affect the conversion of a picture file, but do unnecessarily increase the number of picture points in the Mark 2 format.

Converted pictures that have been saved with intermediate points may flicker during output. This is always the case when the consecutive picture points are so tightly spaced that the scanners cannot reach their optimal speed when tracing the laser beam.

Blank changes

Picture points at which blank changes take place are especially critical. As a rule, "waiting times" are included to achieve a clean blank change:
before the blank change: to ensure that the laser beam has reached the desired position
after the blank change: to ensure that the laser beam is not moved to the next point until the beam has been switched completely on or off.
The required "waiting time" depends on the frequency with which the picture points are transferred to the scanner and the time required to switch the laser completely on or off. Most laser systems store multiple picture points at the switching position to achieve these "waiting times".
As a conversion program cannot know whether the picture points to be converted are required for the correct display of the picture or are intended to ensure the clean blank change, attempting to display a laser picture correctly using a different laser computer can be a problem without prior editing.

Repeat attribute

To achieve sharp corners in their output, most laser formats save the same picture point several times consecutively. This increases the number of picture points unnecessarily. With the *Mark 2* it suffices to save a factor with each picture point that determines the number of times a picture point is to be repeated.

Color attribute

Laser graphics are generally projected in color. Within a laser graphic, the colors to be displayed are either:

- assigned to each picture point as an current color (e.g. red), or
- saved numerically as a color index with each picture point; the current color to be displayed is specified in a color table that may be stored independently of the laser graphic.

Additional attributes

Some laser systems support additional attributes that are assigned to the picture points of the picture file. As not all of the information in a picture file can be taken into account during conversions, these additional attributes may be lost. For example, the *Mark 2* supports the "Matrix" and "Vector" attributes that are not used in other laser systems.

COnvert

All conversion programs can be launched from the COnvert command group of the FILE data management system. Change to the directory containing the data to be converted before invoking a conversion command.

Importing graphics

Filing system

The conversion programs of the Mark 2 convert graphics files from other systems and store them as pictures (*.PIC) in a picture catalog (*.CATL). If the catalog already exists, additional pictures can be added to it during the conversion.
When adding picture files, please take care not to overwrite already converted pictures with pictures that have the same file name.

Picture points

As a rule, all saved picture points of an external graphics format will be saved without changes or interpretation as a corresponding picture point in Mark 2 format.

Merging picture points

Consecutive picture points with the same attributes will be merged to a single picture point that will be given a repeat value corresponding to the number of merged picture points.

Blank changes

Most laser computers switch the beam on and off by assigning the color "black" to a picture point or by supporting a special signal that switches the laser on and off as required. All Mark 2 converters use the second method to recognize a blank change.

Blank change repeats

Saved picture point repetitions within a blank change are not merged to a single picture point.
To ensure that converted laser pictures appear as on the original system when displayed by a Mark 2, the default settings of the advanced parameters must be set to zero ("Pre / Post blank on /off repeats", ⇨ "Display Parameters").

Color attributes

The Mark 2 supports up to 64 colors per picture. These are assigned to the picture points as numerical color indices of a color palette. The current colors are established during the show by the creation of up to four color palettes that can be modified as required while the show is in progress.

Up to 64 colors

Color information from external formats that support fewer than 64 colors are converted to color indices of a future Mark 2 color palette.

More than 64 colors

Additional colors from formats that support more than 64 colors per picture are lost.

Exporting Graphics

Mark 2 pictures (*.PIC) must be prepared before being exported into external formats to ensure that information such as picture point repetitions - which the Mark 2 controls with its global settings - will be integrated in the individual picture file.
First, make copies of the files or catalogs that you would like to export.
The Mark 2 provides two commands in the picture editor ("Piced") as preparation for the export operation, "Make.PreCompute" and "CATalog.PreCompute".

```
        Make.PreCompute
```

This command prepares a picture file for export.

```
        CATalog.PreCompute
```

This command prepares all of the pictures in a catalog for export.

During the preparation for export, the relevant Mark 2 files are expanded:
In the case of longer distances between two consecutive picture points, additional picture points are inserted along the straight-line connections between these points as per the "Maxdist drawn" and "Maxdist blanked" settings.
In the case of blank changes, the picture point coordinates are saved repeatedly as per the "Pre blank on/off repeat" and "Post blank on/off repeat" settings.

Conversion Programs

ILDA

The ILDA format was developed and standardized by the International Laser Display Association. Many laser computer systems support the import and export of this format.
An ILDA file contains a sequence of pictures in which the following information can be stored for the individual pictures:

- two- or three-dimensional description of the graphic
- name of the company that created the picture
- name of the picture and picture sequence
- number of pictures belonging to the sequence
- number of the picture within this sequence
- number of picture points

COnvert.ILDAtoDSP

Parameters: `<filename> <catalog>`
Options: `/KeepNames /IgnorePalette /IgnoreRGB /noRGB /OptBlanks /Overwrite`

Starts the conversion of an ILDA file.

`<filename>`
ILDA file to be converted.

`<catalog>`
This parameter specifies the picture catalog in which the converted pictures are to be saved.

`/KeepNames`
Use this option to retain the internal names for the converted files. This option should only be used if you are certain that all pictures and picture sequences within the ILDA file have been named in a format compatible to the *Mark 2*.

Naming without `/KeepNames`
If this option is not selected, the conversion program will name the files according to the following principle: "00001.PIC", "00002.PIC" etc.

`/IgnorePalette`
Ignores the color palettes contained in the ILDA file

`/IgnoreRgb`
Ignores the color information of true-color pictures

`/noRgb`
Prevents the DSP picture from containing true-color information

`/OptBlanks`
Replaces waiting points on blank change with a blank change with the repeat 1

`/Overwrite`
Overwrites existing pictures with the same name

Conversion
Picture point coordinates
ILDA uses a coordinate system in which the Z-axis goes from back to front. In the *Mark 2*, the Z-axis goes in the opposite direction. The Z values are inverted for this reason.

Blank changes without the option `/OptBlanks`
Blank changes are realized in ILDA files by repeatedly recording the same picture point. These repeated points are converted to normal picture points. The *Mark 2*'s default automatic repetition of picture points should therefore be disabled when projecting these pictures. Set the "Pre blank on / off repeat" and "Post blank on /off repeat" of the display parameters to 0 for this purpose (⇨"Advanced Parameters").

COnvert.DSPtoILDA

Parameters: `<catalog> <filename>`
Options: `/3D /2D /TrueColor /IndexedColor /NoPrecompute`

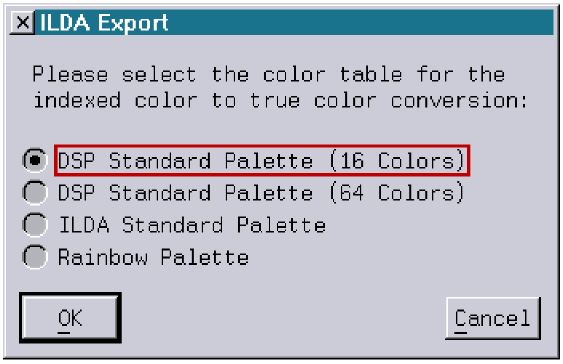
Converts the catalog located on the hard disk into ILDA format and saves the result under the specified file name on the hard disk.

`/3D`
The pictures will be saved in the ILDA Format 0 (3D Coordinates with Indexed Color) resp. ILDA Format 4 (3D Coordinates with True Color).
For three-dimensional pictures this option will have no effect.

`/2D`
The pictures will be saved in the ILDA Format 1 (2D Coordinates with Indexed Color) resp. ILDA Format 5 (2D Coordinates with True Color).
For three-dimensional images this option will cause the Z-information to be lost.

If neither /3D nor /2D is specified the most suitable format for each image will be determined automatically.

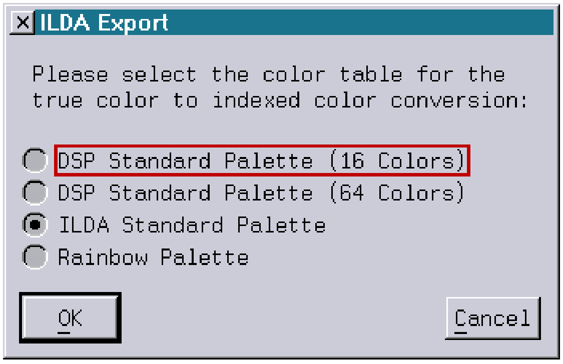
`/TrueColor`
The pictures will be saved in the ILDA Format 4 (3D Coordinates with True Color) resp. ILDA Format 5 (2D Coordinates with True Color).
If one or more pictures do not contain TrueColor information, the following dialog will open to determine how the indexed colors are to be mapped to the TrueColors:


ILDA Export

Please select the color table for the indexed color to true color conversion:

● DSP Standard Palette (16 Colors)
○ DSP Standard Palette (64 Colors)
○ ILDA Standard Palette
○ Rainbow Palette

`/IndexedColor`
The pictures will be saved in the ILDA Format 0 (3D Coordinates with Indexed Color) resp. ILDA Format 1 (2D Coordinates with Indexed Color).
If one or more pictures contain TrueColor information, the following dialog will open to determine how the TrueColors are to be mapped to the indexed colors.


ILDA Export

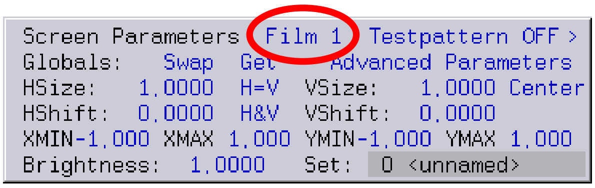
Please select the color table for the true color to indexed color conversion:

○ DSP Standard Palette (16 Colors)
○ DSP Standard Palette (64 Colors)
● ILDA Standard Palette
○ Rainbow Palette

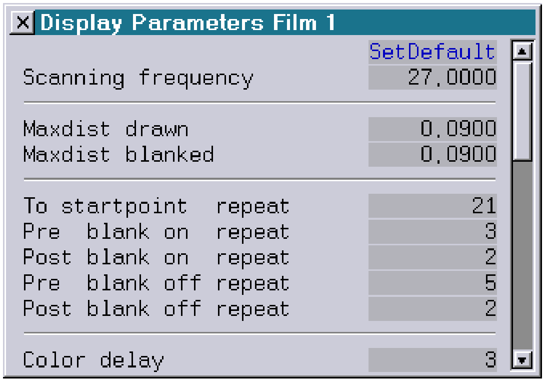
If neither /TrueColor nor /IndexedColor is specified the most suitable format for each image will be determined automatically.

`/NoPrecompute`

Unless this option is selected the pictures will be precomputed according to the currently active scanning parameters prior to conversion into the ILDA format.



```
Screen Parameters  Film 1  Testpattern OFF >
Globals:    Swap  Get   Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000   Set:  0 <unnamed>
```



```
Display Parameters Film 1
                                 SetDefault
Scanning frequency                  27.0000

Maxdist drawn                        0.0900
Maxdist blanked                      0.0900

To startpoint   repeat                   21
Pre  blank on   repeat                    3
Post blank on   repeat                    2
Pre  blank off  repeat                    5
Post blank off  repeat                    2

Color delay                               3
```

The following scanning parameters will be used for precomputing:

```
Maxdist drawn
Maxdist blanked
---------------------
To startpoint  repeat
Pre  blank on  repeat
Post blank on  repeat
Pre  blank off repeat
Post blank off repeat
---------------------
Color delay
```

HP-GL2

HP-GL2 is the abbreviation for HewlettPackard™ Graphics Language Version 2. The conversion program of the *Mark 2* only supports a part of this language's full set of capabilities.

COnvert.HPGLtoDSP

Parameters: `<filename> <catalog>`

This command starts the conversion of an HP-GL2 file.

`<filename>`

Specifies the file to be converted. Wildcard characters (*,?) may be used to specify multiple HP-GL2 files for conversion.

`<catalog>`

Picture catalog in which the converted files are to be stored.

Conversion

Picture point coordinates

As a rule, HP-GL2 files only record picture points in two dimensions. The third dimension, i.e. the Z value, is set to 0 during the conversion.

Blank changes
HP-GL2 files use commands to move the pen from point to point when plotting. The pen may or may not make contact with the paper during the plot operation. This raising and lowering of the pen corresponds to the blank change of laser systems.
To ensure that converted HP-GL2 files are correctly projected, the *Mark 2*'s default automatic repetition of picture points must be enabled. Set the "Pre blank on / off repeat" and "Post blank on /off repeat" of the display parameters to values greater than 0 for this purpose (⇨"Advanced Parameters").

DVS

COnvert.AVItoDVS

Parameters: `<filename> <filename>`

Converts an AVI file into the DVS format.

The following conditions have to be fulfilled for the conversion:

1) The LGServer has to be running and the PC on which the LGServer is running has to be specified as the server in the I/O Setup of the Mark 2.

2) The AVI file has to be located in the directory which is specified under "WAV/AVI" in the LGServer.

The DVS file will be saved into the current directory of the *Mark 2*.

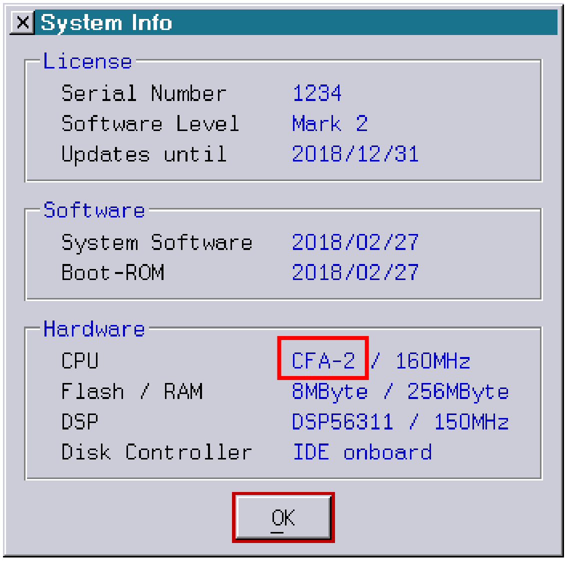
DVS files can be entered into the trickfilm command "ApplyBitmap" without a running LGServer at a later time.

Updating System Software

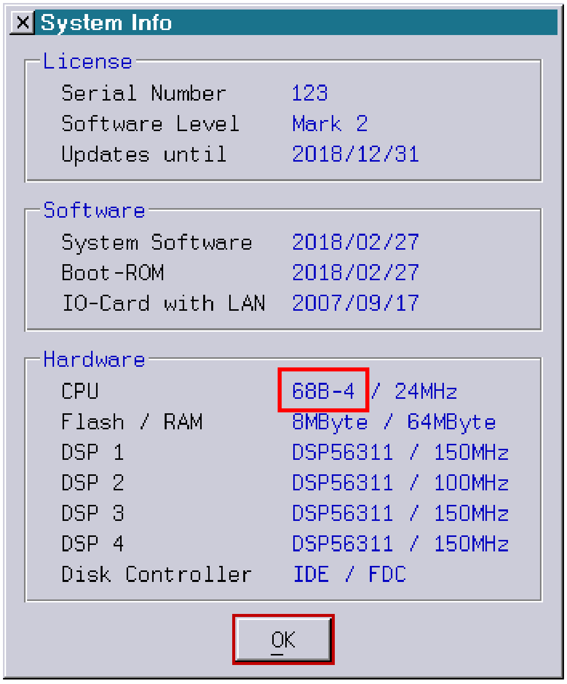
The system software of the Mark 2 is a binary file which is stored in so-called flash ROMs. This has two significant advantages:

1. The Mark 2 is available immediately after it has been switched on; a hard disk drive is not required. The user can update the system software if required.
2. The system software is subject to continuous further development to meet the requirements of our customers. The current version can be obtained from our website.

The CPU type of the controller to be updated must be determined prior to installing the new system software version. This can be done using the command "INFO":



```
System Info
License
  Serial Number      1234
  Software Level     Mark 2
  Updates until      2018/12/31
Software
  System Software    2018/02/27
  Boot-ROM           2018/02/27
Hardware
  CPU                CFA-2 / 160MHz
  Flash / RAM        8MByte / 256MByte
  DSP                DSP56311 / 150MHz
  Disk Controller    IDE onboard
```



```
System Info
License
  Serial Number      123
  Software Level     Mark 2
  Updates until      2018/12/31
Software
  System Software    2018/02/27
  Boot-ROM           2018/02/27
  IO-Card with LAN   2007/09/17
Hardware
  CPU                68B-4 / 24MHz
  Flash / RAM        8MByte / 64MByte
  DSP 1              DSP56311 / 150MHz
  DSP 2              DSP56311 / 100MHz
  DSP 3              DSP56311 / 150MHz
  DSP 4              DSP56311 / 150MHz
  Disk Controller    IDE / FDC
```

CPU types:

- CFA: Lasergraph DSP compact, Lasergraph DSP compact dual, Lasergraph DSP travel
- 68B: Lasergraph DSP workstation

Internet Download

The current version of the system software can be found in the "Customer Area" at www.laseranimation.com.

The following convention applies to the file names:

- CM2-18-02-27.LZip is equivalent to system software dated February 27th 2018 for the Lasergraph DSP compact, Lasergraph DSP compact dual or Lasergraph DSP travel (CPU "CFA")

- WM2-18-02-27.LZip is equivalent to system software dated February 27th 2018 for the Lasergraph DSP workstation (CPU "68B")

Transferring System Software to the *Mark 2*

This can be performed via a network connection (from PC to *Mark 2*) or removable media. We recommend saving the system software on the internal hard disk drive under "C:\ROM-Files" in the subdirectory for the appropriate CPU. This ensures that older versions of the software will also be available in emergencies.

Installing System Software

The system software is provided in compressed form to minimize download times.

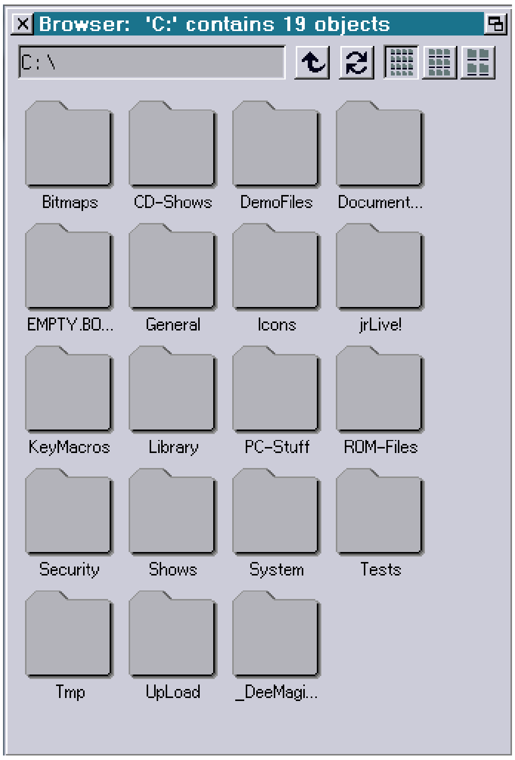
Caution!
Before updating the *Mark 2* system software, complete all tasks and back up your files.

Change to the directory containing the system software and enter the following command:

```
BURNRomImage filename.LZip
```

Then hit <Return> for installing the software.
The *Mark 2* will now begin updating its system software. This may take several minutes.
The *Mark 2* must not be switched off under any circumstances during this time. The *Mark 2* will restart when the update is complete.

The Browser



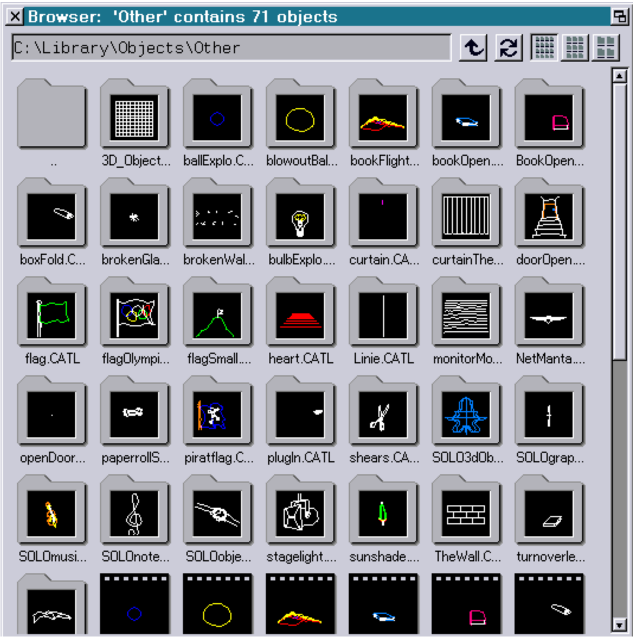
Browser: 'C:' contains 19 objects

C:\

The Browser can be opened by typing "browser" into the command line or by clicking the menu function [BROWSER] from the menu bar.

It can be used to navigate through the file system and to preview pictures and animations.

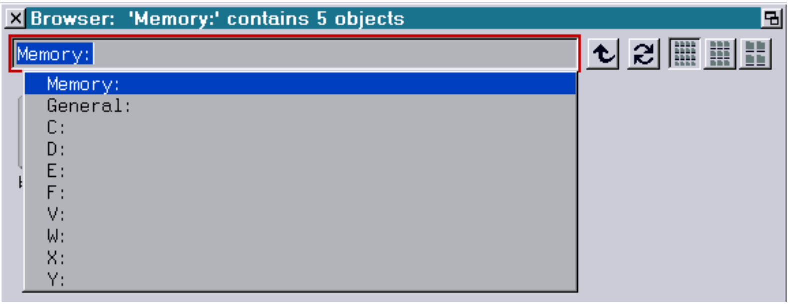
Browser Functions



Browser: 'Other' contains 71 objects

C:\Library\Objects\Other

Path information window



Browser: 'Memory:' contains 5 objects

Memory:

Use the path information window to navigate through the browser by changing the path, e.g. to switch to another partition.
After clicking into the window a popup menu will show you all available partitions. You can also directly enter a new path (e.g. d:\abc\xxxx).

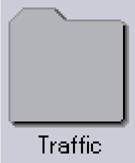
Beside the partitions, two shortcuts to the "Memory" and the "General" directory will always be available. The memory will show you all files that are loaded into the RAM memory and the General catalog shows all available fonts in the General directory in partition C:

Parent directory – clicking this button will bring you to the parent directory

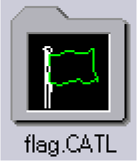
Refresh button – refreshes the content of the browser

Icon size – changes the size of the icons

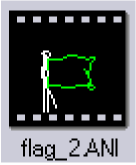
The Icons



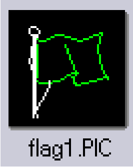
A file directory icon will show no preview content. By clicking onto it, you will open the corresponding directory.



A catalog icon will show one picture (default: first picture).
By moving the mouse cursor onto the icon, the preview will show all pictures in this catalog one after the other.

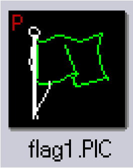


An animation icon will show one picture of the animation.
By moving the mouse cursor onto the icon, the preview will show all pictures in this Animation one after the other.



A picture icon will show the picture itself.

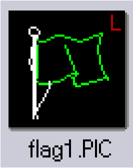
Protected pictures are marked with a 'P':



When a 'P' is displayed this is for one or more of the following reasons:

- The picture is protected for the serial number of the Lasergraph DSP *Mark 2*.
- The picture is protected for the name of the Lasergraph DSP *Mark 2*.
- The picture is not editable.
- The picture cannot be saved.

Pictures from the included library or those that are based on one of the library pictures are marked with an 'L':



The library flag is without consequences for *Mark 2* users because all *Mark 2* systems come with a corresponding library license.

On older Lasergraph DSPs library pictures can only be used when the Lasergraph DSP has a superior license.

By clicking onto a directory, catalog or animation icon, the browser will show you the content of the corresponding directory.
Note that a Boot will be displayed as a directory.

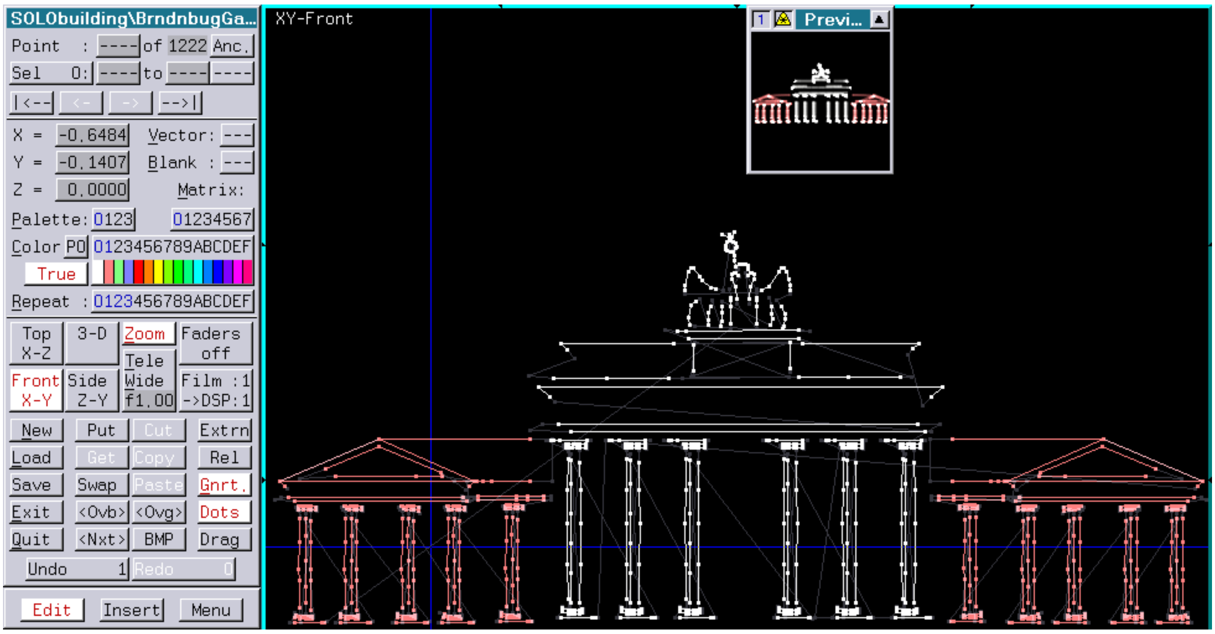
To load a picture, catalog or animation, press the keyboard key <l> while the mouse cursor is on the picture, catalog or animation icon.

Keyboard shortcuts:

Cursor Left	Move one object to the left
Cursor Right	Move one object to the right
Cursor Up	Move one object up
Cursor Down	Move one object down
Page Up	Move one page up
Page Down	Move one page down
POS1	Go to the first object
END	Go to the last object
Backspace	Switch to the parent directory
ENTER	Change into the highlighted directory, select the highlighted object or open the highlighted object in the Piced (depending on the current context)
ESCAPE	Exit the browser
F5	Refresh the browser content
L	Load the highlighted content into the memory
1 / 2 / 3	Change the size of the icons

Mark 2 Manual

Chapter Piced



SOLObuilding\BrndnbugGa...

Point : ---- of 1222 Anc.

Sel 0: ---- to ---- ----

X = -0.6484 Vector: ---

Y = -0.1407 Blank : ---

Z = 0.0000 Matrix:

Palette: 0123 01234567

Color P0 0123456789ABCDEF

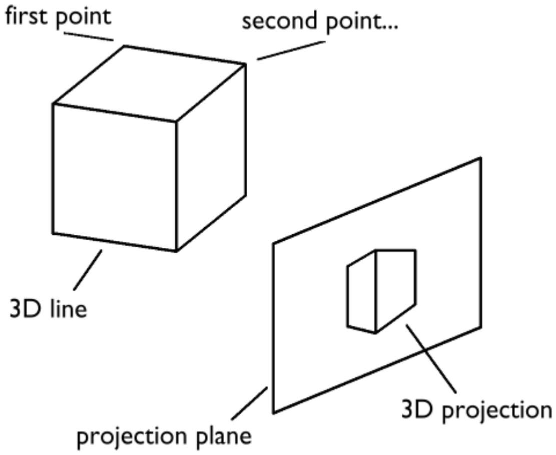
True

Repeat : 0123456789ABCDEF

Undo 1 Redo 0

Introduction

You can create pictures and edit existing or imported pictures with this module. Once the pictures have been saved, they are available in the film editor for further use.
The picture editor can be launched from the desktop with the F4 key or by entering "**Piced**" on the input line. To launch the picture editor directly, double-click the **|Piced|** button in the menu bar. It is possible to enter a name of a catalog plus a name of a picture to load a certain picture immediately.
By clicking the button **|[browse]|** in the menu bar the picture browser will be opened which displays all pictures of catalogs and animations and so allows simple and fast selection of pictures.
The pictures created by Piced are vector graphics, i.e. they are made up of a list of points with X, Y and Z coordinates connected by lines.
Simply storing the coordinates would not be enough, however, as some line sections should not be displayed; these are "blanked", and others are displayed in a variety of colors. For this reason, attributes are assigned to all points to specify these additional properties.



The resulting visible image is a two-dimensional projection of the 3D line drawn by the connected scanner. The laser draws the coordinates sent to it on the projection surface in such quick succession that the illusion of a continuous line results. The process is repeated when the laser reaches the end of the list of points to ensure that the image remains visible. This operation is realized by the deflection mirrors in the scanner which are driven electromagnetically and are thus subject to certain limitations due to their inertia. For example, the transferred coordinates may not always be reached precisely, especially if the distances between the points are very large. The corners displayed in the projection will then appear rounded.
A further effect occurs if the number of points to be drawn is so large that the refresh rate has to be reduced, causing the projected image to flicker.
All pictures should be designed with these effects in mind to ensure that the results correspond to your expectations. If the pictures are to be animated, please note that the viewer of the animation will register fewer details than you see when viewing a static picture in the editor. In addition, a higher degree of abstraction does not necessarily mean a loss of quality.
The processes of creating and editing pictures are generally similar. First, points must be specified using a mouse or digitizer tablet, or an existing picture must be loaded from the hard

drive or out of the picture memory . Some commands generate primitives (such as circles) or text that can also be used to generate points.
These existing points can then be processed further. Commands can be used to select individual or multiple points in order to change their position or attributes.
Perspective views are available to view the image in 3D space.

Saving Pictures

The finished pictures must be saved before they can be made available to the film editor. The picture editor features its own picture memory. The film editor accesses this picture memory when using pictures**. To store pictures permanently, they must also be saved to the hard drive.**
The stored pictures are grouped into catalogs. The catalog name must therefore be specified in addition to the picture name when saving or loading a picture.

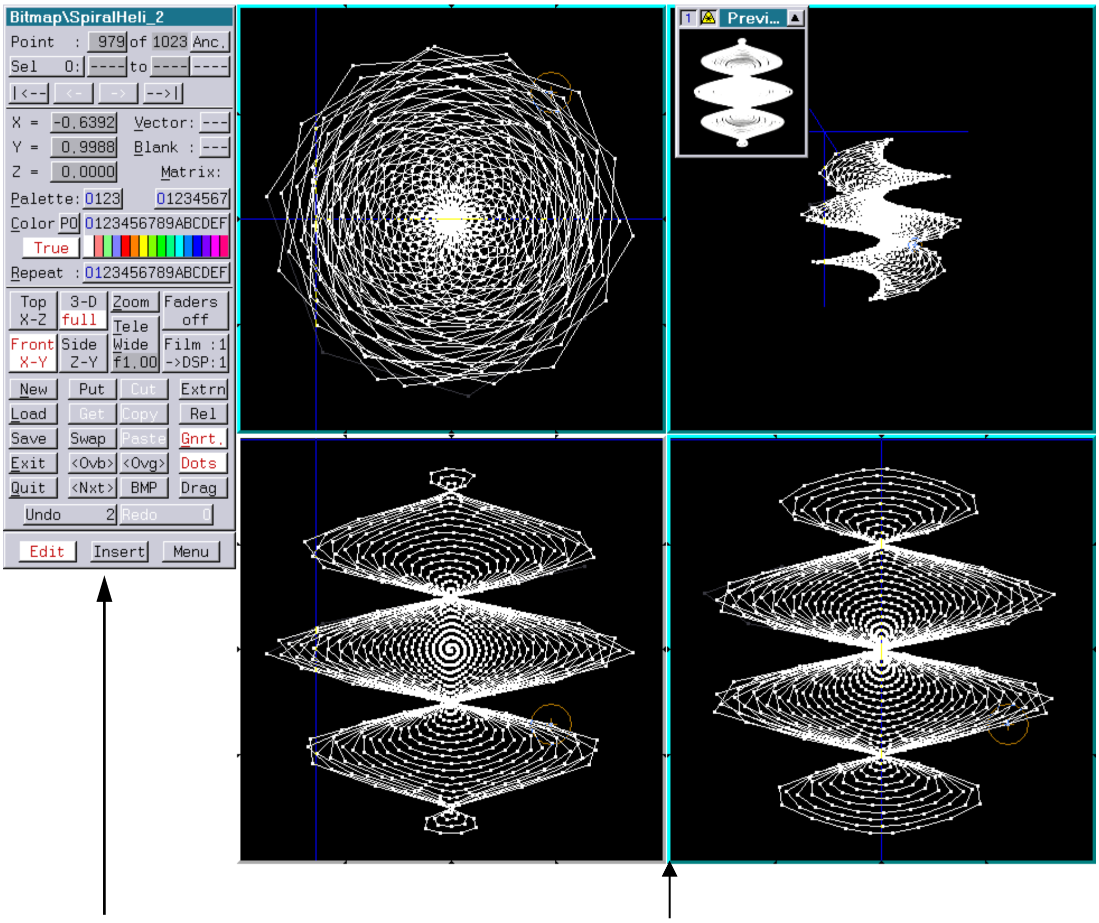
Clipboard

Pictures or parts thereof can be copied to the clipboard. The clipboard can be used to store a single partial or complete picture. It can then be kept handy for quick re-use while editing, for example.

Overlay Picture

One or two overlay pictures and an additional bitmap can be displayed as drawing templates. This function can be very useful when creating an animation, for example.

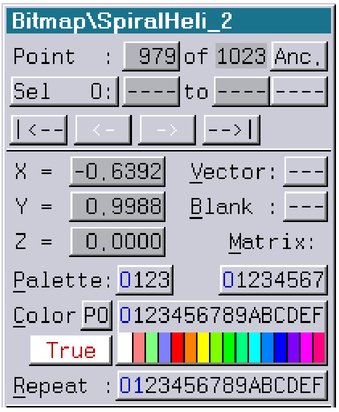
The Graphical User Interface



Editing bar

Workspace

The user interface visible after launching Piced is made up of the workspace on the right and the window with the point list and the editing functions on the left. The graphical editing of the picture takes place in the workspace. The upper half of the window on the left displays the number of points and their attributes. The lower part of the window contains buttons for commonly used functions.
The title bar of the window contains the name of the current picture and its catalog.


Bitmap\SpiralHeli_2

Point : 979 of 1023 Anc.

Sel 0: ---- to ---- ----

X = -0.6392 Vector: ---

Y = 0.9988 Blank : ---

Z = 0.0000 Matrix:

Palette: 0123 01234567

Color P0 0123456789ABCDEF

True

Repeat : 0123456789ABCDEF

Point List

The **Point** line displays the number of the active point and the total number of existing points.

Active Point

Only one point can be active at any given time. To edit multiple points, they must be selected first. The active point is either the last point set while in **Insert** mode, or an existing point when working in **Edit** mode. The **||<--|**, **| <- |**, **| -> |** and **|-->||** buttons can be used to navigate through the point list to select a point. **||<--|** or the "Home" key activates the first point; **|-->||** or the "End" key the last point of the picture. The **| <- |** or **| -> |** buttons activate the previous or next point. The same functions are assigned to the "PgDn" and "PgUp" keys.

Selection

The **|Sel|** line indicates the selected points; these are marked in the picture with a green rhombus. The number in the **|Sel|** button indicates the number of selected points. The last button in this line shows whether the selection is continuous; this is indicated by the abbreviation **|Con.|**. An interrupted selection is indicated by the abbreviation **|Int.|**. If all points are selected **|All|** appears, if no point is selected **|----|** is displayed.

Coordinates

The fields X, Y, Z display the current coordinates of the cross hairs. A mouse click into the appropriate field permits the value of the coordinate to be changed using the keyboard or the mouse poti mode.

Attributes

The values of the attributes of the active point are also displayed in a white field in the appropriate line. The attributes assigned to a point determine additional properties of the preceding line section. The values are saved together with the coordinates in the picture. The value of an attribute for the active point can be set to the desired value with a simple mouse click. Attribute values can also be changed via the keyboard.
The **|True|** and **|Pseudo|** buttons toggle between the color display modes. The **|True|** mode uses the colors defined by the Trickfilm; **|Pseudo|** mode uses standard colors for the screen display.

Attributes

The attributes of a point are used for the further description of the drawn line. Attributes describe the properties of the point and the previous line, the connection to the previous point. Six types of attributes exist: Color, Palette, Matrix, Repeat, Blank and Vector.

Color: 0-15 (0-F)
The point or line section will be given this color of this value from the selected palette.

Palette: 0-3
This value determines the number of the palette to be used for the color of the point and previous line section. Up to 4 palettes with 16 colors each can be used.

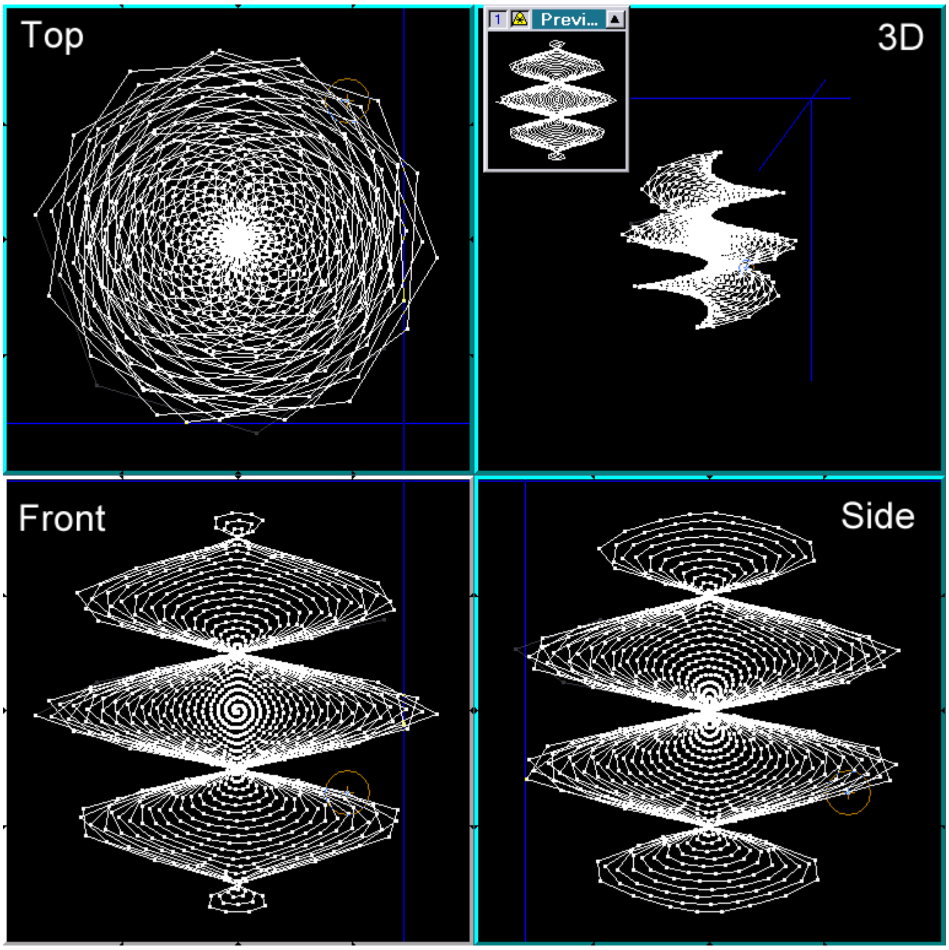
Matrix: 0-7
This attribute permits points with the same matrix number to be saved as a group. This permits part of the picture to be manipulated at a later time by specifying the matrix number. The operation will then affect all points with the same matrix number. For example, they can then be moved independently of the rest of the picture.

Repeat: 0-15 (0-F)
The value of this attribute determines the number of repetitions of the point during the output. If a point is output several times, the scanner will have enough time to depict the point precisely. This counteracts the rounded-corner effect, causing the corner to be rendered precisely in the output. This point will then appear brighter than the rest of the picture.

Blank: On/Off
If the "Blank On" attribute is set for a given point, the line section between the point and the previous point will not be displayed in the output. This is necessary to display two separate objects. The connection between the last point of the first object and the first point of the second object is not visible due to the blanked line section.
As the scanner has to cover this distance nevertheless to reach the next visible part of the picture, this line section will be shown as a gray line in the picture editor. The scanner must also cover the section from the last point of the picture as a whole and the first point, so this section will also be blanked automatically.

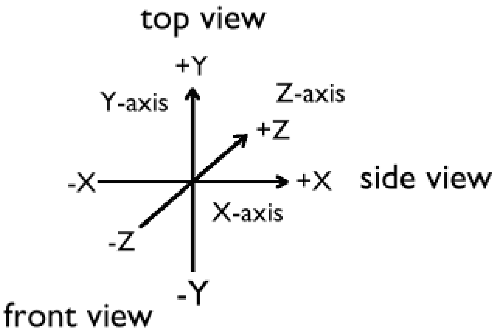
Vector: On/Off
This attribute determines whether the given line section should be interpreted as a normal vector of a surface. This normally shows the orientation of the surface in 3D space. The program can thus distinguish between the front and rear side of a surface. The surface is only visible when the normal is facing in the direction of the viewer. To define a normal vector, the first two points of the line section that describes the surface must point in the right direction with the attribute "Vector On". This line section will not be displayed on the picture rendered by the scanner. These points appear in the point list like all others, the sole difference being the vector attribute which is set to "On".

Views



Coordinate System

Pictures created with Piced consist of lines in three-dimensional space. As a result, it is possible to view the drawn object from a variety of directions. The viewing direction of the selectable views follows one of the spatial axes of the coordinate system. As the picture drawn by the scanner is a 3D view, it is possible to display the picture in a 3D view window. The various views can also be displayed simultaneously in the workspace which is divided into four sections for this purpose.



View Windows

The **Front**, **Top** and **Side** views are referred to as orthographic projections as they show the picture without perspective distortion with the view at right angles to one of the planes of the coordinate system. The **Front** view shows the XY plane; **Top** the XZ plane and **Side** the ZY plane. Mouse inputs relate to these planes, depending on the view activated as the work view.

Coordinate Range

The range of values for the coordinates of the points is limited to the values [-1.0,1.0] for each axis.

The Menu Bar and the Input Line

Menu Bar

The graphical user interface provides many functions that can be executed directly using the mouse. Many functions require a more complex input than a mouse click, however. All available commands are therefore arranged in groups in the menu bar. The menu bar is not always visible to maximize the amount of available workspace. It appears after a brief delay when the mouse cursor is moved to the lower edge of the screen or the **|Menu|** button is pressed in the editing bar. The input line and status bar also appear together with the menu bar. The menu bar also appears when an input is made with the keyboard or a command selected directly with the function keys. Many functions require keyboard input; in this case the menu bar also appears. The menu bar is hidden after a brief delay when the input line is empty and the mouse cursor has been removed from the menu bar area.
The commands in the menu bar are transferred to the input line by a mouse click or the corresponding function key. The commands marked as disabled (default color: black) cannot be executed because certain conditions have not been fulfilled (e.g. no points have been selected).

Parameter Syntax

All commands can also be entered in the input line using the keyboard. The key words for the parameters must be separated from the previous text by at least one space, otherwise an error message will be returned. In the case of parameters with a value, the number must be entered immediately after the key word (e.g. **Color4**). A space between the key word and the number will cause an error message to be returned.

Keyboard Shortcuts

Many commands can be executed via a keystroke or key combination.
See the list of keyboard shortcuts in the attachment.

Creating and Editing Pictures

An empty workspace will appear when Piced is launched without parameter. The picture that was edited last will remain in the picture editor as long as the computer has not been shut down. As no points are present, the **Insert** tool will be enabled. New points can be set with this tool by placing the mouse cursor in the workspace and setting points with the left mouse button. Individual points will be placed in the position marked by the cross hairs. Holding the mouse button will cause points to be set continuously while the mouse is being dragged until the button is released. The cross hairs will remain enabled until the mouse cursor is activated by clicking the right mouse button.

Editing Modes

Two important commands are available when creating new pictures or editing existing ones - the command **Edit** to edit existing points and **Insert** to set new points. These commands can be activated via the **|Edit|** and **|Insert|** buttons located at the lower edge of the command window. A distinction must be made between two input modes, mouse cursor mode and cross hair mode. Commands can be executed via the buttons and the menu bar can be activated when the mouse cursor mode is active. With the cross hairs activated, the picture can be edited and graphic inputs can be executed. It is possible to disable this mode by clicking the right mouse button in order to continue in mouse cursor mode.

Insert

The **|Insert|** button activates the insert mode. In this mode, new points can be positioned with the cross hairs in the workspace within the active view. As soon as the first point has been set, the cross hairs show a connection to the previous point. This "rubber band" represents the line that will be established when a point is set at the cross hair position. As long as no following point has been set, a temporary point will exist at the cross hair location. This temporary point has all of the properties of a real point, except that it has not yet been entered in the point list. As soon as the position of this temporary point has been confirmed, the point is set and a new temporary point is created.

Only one temporary point exists at a time, and this is always the active point. When the active point is changed, the temporary point disappears and a new temporary point will appear. If the active point is deleted, e.g. with "backspace", the temporary point will be removed and the previous point will be activated and will thus be the temporary point. When the active point is changed, a temporary point will appear before this active point.

All mouse inputs are only possible in the plane of the active view. The value for the third coordinate can be changed by -0.2 with the "/" key and by +0.2 with "*".

Insert Parameters

The window for setting the **insert** parameters can be opened by double-clicking with the left mouse button on the **|Insert|** button. The settings refer to continuous input mode with the left or middle mouse buttons being held. The individual functions are explained in detail under **Tools.InsertPars**.

Left Mouse Button

The temporary point is selected or positioned with the left mouse button while in insert mode. A mouse click performs the following actions:

- **Insert Mode**
  If no temporary point exists, a temporary point will be created at the position of the cross hairs or mouse cursor. If an existing temporary point is found, the temporary point will be inserted behind this point. If no point is found, the temporary point will be inserted behind the last available point with a blanked connection. The cross hairs will then remain active to position the temporary point.

  If a temporary point exists and the mouse cursor is active, the temporary point will be set at the position of the mouse click and the cross hairs will be enabled.

  If a temporary point exists and the cross hairs are active, the temporary point will be set at the position of the mouse click and a new temporary point will be created behind this point.

- **Pen Mode**
  Pressing and holding the left mouse button will cause points to be set following the movement of the cross hairs. The points will be set at regular distances or intervals. The settings for this mode are specified in a separate window.

  If a temporary point exists and the mouse pointer is active, the temporary point will be confirmed and a new point will be created behind it with a blanked connection to it at the location at which the mouse button was pressed. The subsequent points will be created with a visible connection line following the mouse movement until the mouse button is released.

  With the cross hairs active, the temporary point will be confirmed and the connection to the newly created points will be visible.

Middle Mouse Button

Clicking the middle mouse button will cause the visible connecting line between the preceding point and the temporary point to be blanked or vice versa. In other words, it toggles the Blank attribute for this point.
Holding the middle mouse button will set a series of points with a blanked connection line corresponding to the set parameters.
If the mouse pointer is active, an existing temporary point will be confirmed and new points set at the position of the mouse pointer.
With the cross hairs active, the temporary point will be confirmed and new points created behind it.
New points will be set with a blanked connecting line until the middle mouse button is released.

Right Mouse Button

A right-click ends the cross hair mode and switches to mouse cursor mode. A temporary point that might be present will be preserved. If the mouse cursor is already active, another right-click will remove the temporary point from the picture.

The Insert command can also be executed by clicking on the **|Insert|** button or using the key combination <Ctrl+ F10>. <Insert> switches between Edit and Insert modes and vice versa.

Edit

The **|Edit|** button in the editing window activates the Edit tool. This command permits existing points to be edited. The point to be edited must be an active point. Only active points can be moved or otherwise modified. Only one point can be active at any given time.

Left Mouse Button

The left mouse button activates a point for editing.
If the mouse cursor is active, a mouse click in the workspace will activate the cross hairs. If a point is located within the capture radius, it will be made to the active point.
If the cross hairs are active, a mouse click will set a point which is already active to the location of the cross hairs and the following point will become active. If no point was active, a mouse click will activate the point within the capture radius. If no point is present within the radius, nothing will happen.
If the mouse button is held, the active point will be set to the position of the mouse click and the following point will be activated. This point can then be moved as long as the left mouse button is held. The point will remain active after the mouse button is released. If the mouse cursor was active when the left button was pressed, it will once again be active after the mouse button is released. While in move mode, the affected point can be released using the right mouse button.

Middle Mouse Button

When the cross hairs are active, the active point can be deactivated with the middle mouse button. If the mouse cursor is active, the middle mouse button will activate the cross hairs.

Right Mouse Button

The right mouse button activates the mouse cursor while the cross hairs are active. An active point will remain active. If the mouse cursor is active, a right-click will deactivate the active point.

The Edit command can also be executed by clicking on the **|Edit|** button or using the key combination <Ctrl+ F9>. <Insert> switches between Edit and Insert modes and vice versa.

Undo

The command Undo.Undo reverses the last action. Repeating the command reverses the action before that. Up to 16 actions can be undone. Only changes to the picture can be undone. The command will not affect changes in the zoom factor, for example, or other commands that do not actually affect the picture.

The command can be executed by clicking on the **|Undo|** button or using the key combinations <Ctrl+z> or <Ctrl+ Backspace>.

Redo

The command Undo.Redo restores the last undone action. Repeating the command restores the action undone before that. Up to 16 actions can be redone. Only changes to the picture can be redone. The command will not affect changes in the zoom factor, for example, or other commands that do not actually affect the picture.

The command can be executed by clicking on the **|Redo|** button or using the key combinations <Ctrl+y> or <Ctrl+Shift+Backspace>.

Digitizer Tablet

The functions of the digitizer tablet mouse correspond to those of a conventional mouse. The left mouse button is the yellow button of the digitizer tablet mouse, the middle button corresponds to the white button and the right mouse button to the green button of the digitizer tablet mouse. The procedures for the digitizer tablet correspond to those for the conventional mouse. The digitizer tablet can also be set to an absolute mode in which the input coordinates are interpreted in absolute terms. The difference between the relative and absolute mode of the digitizer tablet is only noticeable in the position of the cross hair, all other functions remain the same. The red button of the digitizer tablet switches between the relative and absolute entry modes.

The Menu Bar and Its Individual Commands

This command starts the picture editor:

```
Piced [lastedit] [byfilm] <catalog> [browse]
```

- The `[lastedit]` parameter starts Piced and opens the last edited picture.
- `[byfilm]` opens a picture which belongs to a section in a film.
- `<catalog>` can be used to open a picture from a catalog in the picture memory.
- The `[browse]` parameter starts the picture browser which allows comfortable picture selection.

PIcture.

The Picture command group contains all commands needed to load and save pictures. This group also contains the commands used to exit Piced as well as commands to copy entire pictures to the clipboard or load drawing templates.

PIcture.Load <catalog> <picture>

Options: **/Save /Discard /Nodisksave /useProtect**

<catalog>
This command opens a picture with the name **<picture>** from the specified picture catalog **(<catalog>**) in the picture memory. If no picture catalog is loaded, the computer will look in the current folder and in the "C:\General" folder.

/Save
If a picture is currently being edited, it can be saved before the new picture is opened by using this option.

/Discard
The option discards the current picture.

/Nodisksave
The option only saves the picture to the picture memory, not to the hard drive.

/useProtect
If the selected picture is protected, this option must be used to open the picture, as all pictures that will created based on this picture will inherit the protection.

Example: `PIcture.Load testcat picture1`

This command will load the picture named "picture1" from the picture catalog "testcat". This command can also be executed via the **|**Load**|** button or the key combination <l> / <Ctrl+l>.

PIcture.LoadNext

Options: /Save /Discard /Nodisksave /useProtect

This command loads the next picture of the current picture catalog. This option is quite useful for quickly viewing all of the pictures in a catalog.
This command will not be executed if the active picture does not have a name or is the only picture in the catalog. The options are used as for the **PIcture.Load** command.

This command can also be executed by clicking on the right-hand half of the **|**<Nxt>**|** button or by using the ">" key.

PIcture.LoadPrev

Options: /Save /Discard /Nodisksave /useProtect

This command loads the previous picture of the current picture catalog. This command will not be executed if the active picture does not have a name or is the only picture in the catalog. The options are used as for the **PIcture.Load** command.

This command can also be executed by clicking on the left-hand half of the **|**<Nxt>**|** button or by using the "<" key.

PIcture.Save <catalog> <picture>

Options: /Overwrite /Nodisksave

This command saves the current picture in the picture memory and on the hard drive. The picture will be saved under its current name if one has already been assigned. If a new picture name or catalog is specified, Piced will store the picture under that name. If the picture catalog does not exist, it will also be created.

/Overwrite
Use this option to replace an existing picture with the current one.

/Nodisksave
This option only saves the picture to the picture memory, not to the hard drive.

This command can also be executed by clicking the **|**Save**|** button or using the key combination <Ctrl+s>.

PIcture.Exit <catalog> <picture>

Options: /Overwrite /Nodisksave

This command closes the picture editor and saves the current picture. The computer is returned to the state it was in when Piced was called up.

As with the **PIcture.Save** command, a catalog and picture name can be specified. The options **/Overwrite** and **/Nodisksave** have the same effect as with the command **PIcture.Save.** The contents of the clipboard are preserved and will be available when Piced is called up again.

This command can also be executed by clicking the **|Exit|** button or using the <e> / <Ctrl+e> key combination.

PIcture.Quit

This command closes the picture editor without saving the current picture. The computer is returned to the state it was in when Piced was called up. The current picture will not be saved to either the picture memory or the hard drive. The contents of the clipboard will be preserved. The current picture will once again be available when Piced is called up again without additional parameters.

This command can also be executed by clicking the **|Quit|** button or using the <q> / <Ctrl+q> key combination.

PIcture.PUT

This command copies the current picture to the clipboard. Any picture already in the clipboard will be deleted. A picture can thus be saved quickly for subsequent re-use.
If the current picture is protected, it will retain this status on the clipboard.

This command can also be executed by clicking the **|Put|** button.

PIcture.Get

Options: **/Save /Discard /Nodisksave**

This command copies the picture from the clipboard to the current picture. The contents of the clipboard will be preserved and the current picture overwritten.
The options **/Save**, **/Discard, /Nodisksave** must be stated to determine whether the current picture should be saved to the hard disk and picture memory, to the picture memory only, or discarded. If the picture on the clipboard is protected, its status will be transferred to the current picture.

This command can also be executed by clicking the **|Get|** button.

PIcture.SWaP

This command swaps the current picture with the one on the clipboard.

This command can also be executed by clicking the **|Swap|** button.

PIcture.Paste

Options: /SoftBlank /useProtect

This command inserts the picture on the clipboard into the current picture. Its function corresponds to that of the **PIcture.Merge** command, except that the picture to be pasted is taken from the clipboard.

This command can also be executed by clicking the **|Paste|** button.

PIcture.Merge <catalog> <picture>

Options: /SoftBlank /useProtect

This command permits a second picture to be merged into the current one. The picture to be added must be named; it can be loaded from the picture memory or the hard disk. The picture will be inserted at the active point. If this is the last point of the current picture, this point and the first point of the picture to be inserted will be merged.
All points of the inserted picture will automatically be selected and **Drag** Mode will be activated to permit the inserted picture to be positioned correctly.

/SoftBlank

/useProtect
If the picture to be inserted is protected, this option must be used; the status will be transferred to the current picture which will then also be protected.

PIcture.New

Options: /Save /Discard /Nodisksave

This command is used to start a new picture. If a picture is already present, the options **/Save**, **/Discard, /Nodisksave** must be stated to determine whether the current picture should be saved to the hard disk and picture memory, to the picture memory only, or discarded. All points are then deleted and a new picture can be started. The contents of the clipboard will be preserved.

This command can also be executed by clicking the **|New|** button or using the <n> / <Ctrl+n> key combination. A double-click executes the command directly.

PIcture.Overlay.

This command and its subcommands permit one or two pictures and a bitmap to be displayed as templates in addition to the current picture. The first picture is displayed with brown lines, the second in green. These pictures and the current picture can also be exchanged or used as an exact template with **Set.Attraction.ovlBrown/ovlGreen**.

A bitmap can be defined that will be displayed as the background in the current view ("Top X-Z", "Front X-Y" or "Side Z-Y").

PIcture.Overlay.Brown(Green).Toedit

Options: /Save /Discard /Nodisksave

This command makes the current brown or green overlay picture the current picture.
The options **/Save**, **/Discard**, **/Nodisksave** determine whether the current picture should be saved to the hard disk, to the picture memory only, or discarded.

PIcture.Overlay.Brown(Green).Fromedit

Options: /Discard

This command makes the current picture the brown or green overlay picture.
The **/Discard** option causes the current overlay picture to be discarded.

PIcture.Overlay.Brown(Green).Swapedit

This command swaps the current picture with the brown or green overlay picture.

PIcture.Overlay.Brown(Green).Use <catalog> <picture>

Options: /Discard

This command makes the stated picture the overlay picture of the selected color. If no picture name is indicated, the overlay picture with the selected color will be deactivated and will no longer be visible. If it is already deactivated, the previously used overlay picture will be reactivated.

The **/Discard** option causes the current overlay picture to be discarded.

This command can also be executed by clicking on the middle of the **|<Ovb>|** or **|<Ovg>|** button.

PIcture.Overlay.Brown(Green).LoadNext

Options: /Discard

This command opens the following picture in the current catalog as the new overlay picture with the selected color. The **/Discard** option causes the current overlay picture to be discarded.

This command can also be executed by clicking on the right-hand side of the **|<Ovb>|** or **|<Ovg>|** button or by using the <)> or <]> key.

PIcture.Overlay.Brown(Green).LoadPrev

Options: /Discard
This command opens the previous picture in the current catalog as the new overlay picture with the selected color. The **/Discard** option causes the current overlay picture to be discarded.

This command can also be executed by clicking on the left-hand side of the **|<Ovb>|** or **|<Ovg>|** button or by using the <(> or <[> key.

PIcture.Overlay.Brown(Green).Dots [byPicture] [ON] [OFF]

This command highlights the points of the brown or green overlay picture, making the individual points easier to recognize.
[ON] turns the display of dots on
[OFF] turns the display of dots off
[byPicture] turns the display of dots on/off depending on the condition of the picture in the foreground.

PIcture.Overlay.Use [Brown(Green)] <catalog> <picture>

This command makes the stated picture the overlay picture of the selected color. If no picture name is stated, the overlay picture with the selected color will be disabled and will no longer be visible. The statements are optional. The command will refer to both overlay pictures if neither a picture name nor an overlay picture color is stated.

This command can also be executed by clicking on the middle of the **|<Ovb>|** or **|<Ovg>|** button.

PIcture.Overlay.LoadNext

Options: /Discard

This command opens the pictures following both overlay pictures in the current catalog. The **/Discard** option causes the current overlay picture to be discarded.

This command can also be executed with the <}> key.

PIcture.Overlay.LoadPrev

Options: /Discard

This command opens the pictures before both overlay pictures in the current catalog.
The **/Discard** option causes the current overlay picture to be discarded.

This command can also be executed with the <{> key.

PIcture.Overlay.Dots [byPiture] [ON] [OFF]

This command highlights the points of the overlay pictures, making the individual points easier to recognize.

[ON] turns the display of dots on
[OFF] turns the display of dots off
[byPicture] turns the display of dots on/off depending on the condition of the picture in the foreground.

PIcture.Overlay.BMP <filename> <bright>

Option: /Grayscale

This command can be used to define a bitmap that will be displayed as the background in the current view ("Top X-Z", "Front X-Y" or "Side Z-Y"). The parameter <filename> refers to the name of the bitmap and <bright> to the brightness adjustment from '0' to '1' ('0' = dark; '1' = maximum brightness).

The **/Grayscale** option causes the bitmap to be displayed in a grayscale.

The display of the defined bitmap can be toggled with the command **PIcture.Overlay.BMP**. The bitmap can also be toggled in the current view with the "BMP" button on the editing bar. A selection list will open if no bitmap has been defined at this point.

View.

This command and its subcommands permit the current picture to be viewed from all sides, zoom the picture and permit the 3D view to be changed.

View.Top

This command activates the top view as the working view. All graphic inputs with the mouse or digitizer tablet refer to the XZ plane of the coordinate system. The Y value can only be changed via the keyboard.

This command can also be executed by clicking on the **|Top|** button or using the <a> key.

View.Front

This command activates the front view as the working view. All graphic inputs with the mouse or digitizer tablet refer to the XY plane of the coordinate system. The Z value can only be changed via the keyboard.

This command can also be executed by clicking on the **|Front|** button or using the <y> key.

View.Side

This command activates the side view as the working view. All graphic inputs with the mouse or digitizer tablet refer to the ZY plane of the coordinate system. The X value can only be changed via the keyboard.

This command can also be executed by clicking on the **|Side|** button or using the <x> key.

View.3D.

When editing 3D-pictures, the view in 3D helps you to get on overview of what you´re doing in the 3 axis views.

View.3D.On

This command activates the 3D view as the active working view. If zoom mode is activated, i.e. only one view is visible in the workspace, then 3D view is visible. In the normal fourfold view mode, the 3D view is shown together with the three orthographic modes. All changes to the picture are shown in the 3D view.

This command can also be executed by clicking once or twice on the top half of the **|3D|** button or by using the <s> key. This advances a step at a time through the three modes On, Off and Lock.

View.3D.Off

This command deactivates the 3D view. If zoom mode is activated, i.e. only one view is visible in the workspace, then the selected orthographic mode will be visible instead of 3D mode. In the normal fourfold view mode, the 3D view is shown together with the three orthographic modes.
This command can also be executed by clicking once or twice on the top half of the **|3D|** button or by using the <s> key. This advances a step at a time through the three modes On, Off and Lock.

View.3D.Lock

This command deactivates the 3D view. If zoom mode is activated, i.e. only one view is visible in the workspace, then it will not be possible to enable 3D view. In the normal fourfold view mode, an empty window is displayed instead of the 3D view. This reduces the processing time required for the display of the remaining views.

This command can also be executed by clicking once or twice on the top half of the **|3D|** button or by using the <s> key. This advances a step at a time through the three modes On, Off and Lock.

View.3D.Fullsize

This command displays the 3D view without enlargement. In other words, the focal length of the virtual lens is set to 1.0. As a result, the entire picture is visible in the 3D view window.

This command can also be executed by clicking on the lower half of the **|3D|** button.

View.3D.Normal

This command displays the 3D view without rotation if it has been previously rotated around one or more axes. It also activates the 3D view if it was previously disabled or switched off.

This command can also be executed with the <5> key.

View.3D.Rotation <Xangle> <Yangle> <Zangle>

This command rotates the view of the picture in the 3D view window by the stated number of degrees around the selected axis. The number of degrees must be entered immediately after the key word for the axis; a full rotation corresponds to 360°. With no parameters, the picture will be displayed in the same manner as after the command **View.3D[Normal]** without rotation.

Example: `View.3D.Rotation Yangle30.0`

This command rotates the 3D view of the picture by 30.0 degrees around the Y-axis.

The view of the picture can also be rotated in the 3D view window using the mouse. If no other commands are active, the picture can be rotated around all axes by pressing and holding the mouse button. The rotation is indicated by the cross hairs; the new view will not be displayed until the mouse button has been released, however.

This command can also be executed using the numerical keypad.

- The view will be rotated around the selected axis by +5° or -5°.
- Rotation around the X-axis: <8> or <2>
- Rotation around the Y-axis: <0> or <,>
- Rotation around the Z-axis: <4> or <6>

View.Zoom

This command switches between the normal screen layout and zoom mode. The normal layout shows all four view windows; in zoom mode the active view is shown using the entire workspace.

This command can also be executed by clicking the **|Zoom|** button or using the <z> key.

View.ZoomFactor.

The ZoomFactor enables you to zoom into the picture to make precise editing more convenient.

View.ZoomFactor.Set <factor>

This command sets the zoom factor of the editing area. Factor 1.0 shows the entire editing area in the viewing windows. Values in the [0.25, 50.0] range can be entered. In the case of values greater than 1.0, an enlargement of the picture is displayed with a cropped view of the work area. This view can be panned by moving the cross hairs past the edge of the viewing window.
The panning mode must be enabled for this (see **View.Panning)**.

Example: `View.ZoomFactor.Set 2`
This command sets the zoom factor to the value 2, thus showing all views of the picture in 2x enlargement.

The current zoom factor is shown in the editing bar in the **|Tele|** / **|Wide|** button. Clicking these buttons changes the current zoom factor by a factor of 1.41 (Tele) or 0.71 (Wide). Clicking on the zoom factor display (f 1.0) executes the command View.ZoomFactor.Set, permitting a numerical value to be entered.
Other available keyboard shortcuts:

- <Ctrl F1> zoom factor 1.0
- <Ctrl F2> zoom factor 2.0
- <Ctrl F3> zoom factor 4.0
- <Ctrl F4> zoom factor 8.0
- <Ctrl F5> zoom factor 16.0
- <Ctrl F6> zoom factor 32.0
- <t> change zoom factor by 1.41
- <T> change zoom factor by 2.0
- <w> w>change zoom factor by 0.71
- <W> W>change zoom factor by 0.5

Mouse Wheel:

The zoom factor in Piced can be changed with the mouse wheel.

View.ZoomFactor.byDrag

Use this command to graphically select a part of a view for enlargement. After the command has been executed, the menu bar will be hidden and a violet square will appear around the cross hairs. A rectangular area which will be displayed enlarged can be selected by pressing and holding the left or middle mouse button.

This command can also be executed with the <Ctrl + t> key combination.

View.ZoomFactor.BySelection

This command ensures that the visible area contains all selected points and that the view is as large as possible.

View.Generate

When this function is active (default), the pictures are generated as fast as possible.
Note: For slow PCs the AutoGenerate function can be disabled in Piced if the picture generation is too slow.

This command can disabled by clicking the **|Gnrt|** button or using the <g> key.

View.Center

This command centers the view relatively to the cross hairs.

View.Panning [ON] [OFF]

This command toggles the automatic panning mode. When enabled, this mode permits automatic panning of enlarged views when the cross hairs go beyond the limits of the active view window. A white border around the activated view window indicates the mode; the border is violet when the mode is disabled.

View.Dots [ON] [OFF]

This command toggles highlighting of the points of the picture, making the individual points easier to recognize.

Using the **|Dots|** button gives you 3 options:

1) All dots in fixed color, regardless of the selected line color
2) All dots in the selected line color
3) No dots

View.BlankedLines [ON] [OFF]

This subcommand enables/disables blanked lines for the current picture and the Overlay pictures.

Point.

This command and its subcommands permit access to individual points of the picture.

Point.SetNext

This command activates the following point in the picture and sets the active point to a new position, if moved. This command cannot be executed if no point is active, i.e. when the display lists Point "----".

This command can also be executed by using the <Return> key.

Point.SetPrev

This command activates the previous point in the picture and sets the active point to a new position, if moved. This command cannot be executed if no point is active, i.e. when the display lists Point "----".

Point.Insert

This command inserts one point before the active point and the new point can then be positioned. The previously active point remains at its old position. This command cannot be executed if no point is active.

This command can also be executed with the <Ctrl+Insert> key combination.

Point.SetInsert

This command inserts one point **after** the active point. The previously active point takes a new position, if moved. This command cannot be executed if no point is active.

This command can also be executed with the <Shift+Return> key combination.

Point.DELete

This command deletes the active point and activates the following point. This command cannot be executed if no point is active.
This command can also be executed with the <Del> key.

Point.Backspace

This command deletes the active point and activates the previous point. This command cannot be executed if no point is active.
This command can also be executed with the <Backspace> key.

Point.Coordinates <X> <Y> <Z> <DX> <DY> <DZ>

This command permits new coordinates to be entered for the active point. The entries for the X, Y and Z coordinates are absolute values. The values specified with DX, DY and DZ are relative, i.e. these values are added to the previous coordinate values.

Example: `Point.Coordinates X-0.5 Y0.2 Z0.0`

This command sets the coordinate values for the active point to the value -0.5 for the X value, 0.2 for the Y value and 0.0 for the Z value.

It can also be executed by clicking into the **X**, **Y** or **Z** fields. The coordinate values can also be modified with the mouse poti mode.

Point.Attributes <Color> <Palette> <Matrix> <Repeat> <Blank> <Vector>

This command sets the attributes for the active point via the input line. The new value must be entered directly after the key word for the desired attribute. A "+" or "-" before the value will cause it to be interpreted as a relative value to be added to the existing value.
If an attribute has not been specified, the value will remain the same. If the maximum or minimum value of an attribute has been exceeded or undercut, then the maximum or minimum possible value will be set as the new value. 0 and 1 are the only valid values for the attributes **<Blank>** and **<Vector>**; the 0 stands for **Off** and 1 for the **On** state.

Example: `Point.Attributes Repeat2`

This command sets the repeat value of the active point to the value 2.

The attributes of a point can also be changed in the editing bar with a mouse click on the desired numerical value in the row with the desired attribute. The following keys change the attributes by +1 or -1.

Attribut	+1	-1
Color	<c>	<C>
Palette	<p>	<P>
Matrix	<m>	<M>
Repeat	<r>	<R>
Blank	<b>	<B>
Vector	<v>	<V>

The values of the <Color> and <Repeat> attributes can also be set by pressing the key with the number for the new value. This mode applies to the attribute that was last modified. The attribute is indicated as valid in that its field is displayed as a pressed button.

Point.First

This command activates the first point of the picture. The previously active point remains at its old position.
This command can also be executed by clicking the **|I<--|** button or using the <Home> key.

Point.Prev

This command activates the point before the currently active point. The previously active point remains at its old position. This command cannot be executed if no point is active.

This command can also be executed by clicking the **|<-|** button or using the <PgUp> key.

Point.NeXt

This command activates the point after the currently active point. The previously active point remains at its old position. This command cannot be executed if no point is active.

This command can also be executed by clicking the **|->I|** button or using the <PgDn> key.

Point.Last

This command activates the last point of the picture. The previously active point remains at its old position.

This command can also be executed by clicking the **|-->I|** button or using the <End> key.

Point.Number <number>

This command activates the point with the specified number. The previously active point remains at its old position. The menu entries **<first>**, **<current>** and **<last>** enter the values for the first, active and last points into the input line. The selection **[None]** deactivates the active point.

Example: `Point.Number 3`

This command activates point No. 3 of the picture.

This command can also be executed by clicking in the "Point" field of the editing bar or by using the <#> key.

Point.APpend

This command sets a new point at the end of the picture and activates positioning with the mouse. The previously active point will be left at its old position.

Point.FirstSlct

This command activates the first point in the list of selected points. The selected points are displayed in the editing bar in the **Sel** line. The previously active point will be left at its old position.

Point.PrevSlct

This command activates the previous point in the list of selected points, if one is available. The selected points are displayed in the editing bar in the **Sel** line. The previously active point will be left at its old position.

Point.NextSlct

This command activates the next point in the list of selected points, if one is available. The selected points are displayed in the editing bar in the **Sel** line. The previously active point will be left at its old position.

Point.LastSlct

This command activates the last point in the list of selected points. The selected points are displayed in the editing bar in the **Sel** line. The previously active point will be left at its old position.

Point.ALternative

This command activates points located over one another in the same position. A mouse click will always activate the point with the lowest number. With this command, "hidden" points can also be activated.

This command can also be executed with the <Shift+Tab> key combination. If more than two points are located above one another, each subsequent keystroke activates the following point at the location.

Point.Select <True> <False>

This command selects or deselects the active point. **<True>** selects the point, **<False>** deselects it. The selected points are marked in the views with a green rhombus, the deselected ones with a yellow square.

This command can also be executed with the <u> key.

Point.PrevObj

If a picture consists of more than one “object” (parts that are divided by an blanked line), this command can be used to select the first point (after the blanked line) from the previous object.

Point.NextObj

If a picture consists of more than one “object” (parts that are divided by an blanked line), this command can be used to select the first point (after the blanked line) from the next object.

Set.

This group of commands contains many default settings and commands for the selection of points.

Set.Grid <Distance> <Show>

This command sets and activates a drawing grid. When the grid is visible, new points can only be added at the points of the selected grid. The existing picture will not be changed.
The **<Distance>** parameter determines the spacing of the grid points.
<Show> determines the ratio of visible and invisible grid points. If the entered value is ‘n’, one in every ‘n’ points will be displayed.

Specifying these values is optional. If none are specified, the grid will be disabled or the grid with the previous set of values will be restored.

Example: `Set.Grid Distance0.1 Show2`

This command sets a drawing grid with a point spacing of 0.1, of which every second point is visible.

The drawing grid can also be toggled with the <f> key.

Set.Margins <Left> <Right> <Bottom> <Top> <bacK> <Front>

This command limits the usable value range of the coordinates. The space available for drawing is thus limited by the margins specified with the key words. New points can only be set within these margins. The existing picture will not be changed.

- The maximum values are within the [-1.0, 1.0] range, as long as no scaling factor greater than 1 has been set.
- The key words "**Left-0.5 Right0.5**" limit the value range of the X-axis to between [-0.5, 0.5].
- The key words "**Bottom-0.5 Top0.5**" limit the value range of the Y-axis to between [-0.5, 0.5].
- The key words "**bacK-0.5 Front0.5**" limit the value range of the Z-axis to between [-0.5, 0.5].

The first value for an axis must always be smaller than the following one, otherwise a drawing area of less than zero will be defined. The values are optional; if key words are omitted, the corresponding values will remain unchanged.

The command without parameters toggles the margins. With the margins switched off, the margins with the last specified values will be switched back on.

Set.Selection.

This group of commands selects and deselects points from the point list according to a wide variety of criteria. All **Do.** commands can then be applied to the group of selected points.
The selected points are marked in the picture with a green rhombus, the deselected ones with a yellow square. The numbers of the selected points are displayed in the editing bar in the **Sel** line.
All points can be selected or deselected with a mouse click on the graphical user interface.

The last button of the **Sel** row switches the entire selection off if points were selected, or selects all available points if no selection was active previously (can be recognized by "----" in this button).

Set.Selection.List <point list>

This command selects points according to their numbers. The affected points are specified in a list. The numbers of the desired points are listed, with multiple points separated by commas, or continuous series of numbers indicated as from-to with a dash as a separator.

The starting and end points are included in the selection. Blanks are not permitted within the selection list. The **<list>**, **<first>**, **<current>** and **<last>** menu items display points and their numbers in the input line. **<list>** specifies a list of previously selected points in the input line; **<first>** a "1", **<current>** the number of the active point and **<last>** the number of the last point. If nothing is specified, the current selection will be switched off, or if the selection is switched off, the last active selection will be switched back on.

Example: `Set.Selection.List 1-5,7,8,10-12`
The command shows an example for a selection list.
After the command has been executed, the points 1 to 5, 7, 8, and the points 10 to 12 are selected.
An existing selection will be canceld.

Set.Selection.Begin <starting point> <end point>

This command selects a series of points beginning with the specified starting point up to the already selected point or the specified end point. A zero as a starting point indicates the **<OFF>** state; all points are deselected. The **<first>**, **<current>** and **<last>** menu items display the numbers of the relevant points in the input line.

This command can also be executed with the first button in the **Sel** line of the editing bar.

Set.Selection.End <end point>

This command selects a series of points ending with the specified end point up from the already selected point. A zero as an end point indicates the **<OFF>** state; all points are deselected. The **<first>**, **<current>** and **<last>** menu items display the numbers of the relevant points in the input line.

This command can also be executed with a click on the second button in the **Sel** line of the editing bar.

Set.Selection.Invers

This command inverts the current selection. In other words, all points currently selected will be deselected and vice versa.

Set.Selection.ToObject <point number>

Options: /Only /Add /Except

This command does not select individual points, but rather all points of an object. An object consists of a series of points that are connected with a visible line. The end of an object has been reached when a blanked line follows a point; this marks the beginning of another object.

<point number>
This parameter determines which object should be selected. The object containing the specified point number will be selected. The **<first>**, **<current>** and **<last>** menu items display the numbers of the relevant points in the input line.

/Only
This option causes the command to select only the points of the relevant object; all other points will be deselected.

/Add
This option causes the points of the relevant object to be selected in addition to the already selected points.

/Except
This option retains the previous selection and deselects the points of the relevant object.

Set.Selection.byAttribute <Color> <Palette> <Matrix> <Repeat> <Blank> <Vector>

Options: /Only /Add /Except

This command selects points on the basis of their attribute values. All points of a picture with attribute values corresponding to those specified in the input line will be selected. The values are optional; if key words are omitted, the associated values will not be affected. The values for **<Blank>** and **<Vector>** can only be 0 or 1; the value 0 means **Off**, 1 means **On** for the given attribute.

/Only
This option causes the command to select only these points; all other points will be deselected.

/Add
This option causes the points of the relevant object to be selected in addition to the already selected points.

/Except
This option retains the previous selection and deselects the relevant points.
Example: `Set.Selection.byAttribute Repeat4`
This command selects all points of the picture with a repeat value of 4.

Set.Selection.byDrag

This command activates the graphical selection mode. The menu bar is hidden and the cross hairs activated. Selection mode is indicated by a white square around the cross hairs.
A click with the left or middle mouse buttons will select the first point of a rectangular area in the selection window.

Dragging with the mouse button held will determine the selection area within which all points will be selected or deselected.
The third coordinate will be ignored when working in zoom mode; only the selection rectangle is relevant.

In the normal view mode, the points must be contained in the rectangle.
The third coordinate can be changed using the "/" and "*" keys.
An area opened with the left mouse button selects all of the points contained therein.
An area opened with the middle mouse button deselects all of the points contained therein.
The right mouse button ends this mode. The "Esc" key also ends the mode.

This command can also be executed by clicking the **|Sel|** button.

Set.Selection.Objectdrag

This command activates the graphical selection mode for objects (see **Set.Selection.ToObject**). The menu bar is hidden and the cross hairs activated.
A white circle around the cross hairs indicating the size of the selected capture radius indicates object selection mode.

When the cross hairs are positioned over a point so that it is located within the capture radius, a click on the left mouse button will select the object containing that point.
A click on the middle mouse button will deselect the object containing the point.
The right mouse button or "Esc" key ends this mode.

The selection mode for objects can also be activated with the key combination <Ctrl+W>.

Set.Selection.ON (OFF)

This command switches the current selection on or off. If the selection is switched off, no points are selected. When the selection is switched back on, the previous selection is restored.

Set.Attraction.

The Attraction mode “snaps” points to already existing points or to points in an overlayed picture.

Set.Attraction.ON (OFF)

This command enables (disables) the attraction mode. When the attraction mode is enabled, the cross hairs automatically jump to the position of an existing point in their vicinity; the existing point is "captured". The newly set point is then located in the same position as the existing one. The distance from which an existing point is captured is referred to as the capture radius. This function is very useful in cases in which parts of an existing picture must be copied precisely.

The attraction mode can also be enabled or disabled using the <o> key.

Set.Attraction.Picture

This command enables attraction mode for the current picture. **[Picture]** ensures that only the points belonging to the current picture will be affected.

Set.Attraction.ovlBrown (ovlGreen)

This command enables attraction mode for the overlay pictures. With **[ovlBrown]** and **[ovlGreen]**, only the points of the brown or green overlay pictures are affected.

Set.Attraction.Distance <distance>

The parameter <distance> determines the distance from which the cross hairs will jump to the existing point. The value cannot be greater than 0.25, assuming a scaling factor of 1.

Example: `Set.Attraction.Distance 0.1`

This command states that all points within a radius of 0.1 will be captured.

Set.Tabletmode.

These commands affect the behavior of the digitizer tablet. Normally, the movements of the digitizer tablet mouse are interpreted as relative, i.e. the movement of the mouse on the digitizer tablet is realized starting from the current position of the mouse cursor.

It is also possible to associate the drawing area of the digitizer tablet precisely with that of the drawing area of the screen. In other words, the top left corner of the digitizer tablet corresponds to the top left corner of the screen drawing area, with the same applying to the lower right corners of the drawing areas. Any given position on the digitizer tablet thus corresponds to precisely one position within the screen drawing area.

It is thus easy to transfer a picture precisely from the digitizer tablet to the picture editor. This absolute mode is only active when inputs are being made with the cross hairs. The mouse cursor will continue to function in relative mode. Absolute and relative mode can be toggled with the red button of the digitizer tablet mouse.

Set.Tabletmode.Relative

This command switches the digitizer tablet into relative input mode. The coordinates of the digitizer tablet are not linked to the screen coordinates.

The command can also be executed by clicking the red button of the digitizer tablet mouse. This switches between absolute and relative mode.

Set.Tabletmode.Absolute

This command switches the digitizer tablet into absolute input mode. The coordinates of the digitizer tablet are linked to the screen coordinates. The entire workspace of the digitizer tablet is linked to the screen workspace by default.

The command can also be executed by clicking the red button of the digitizer tablet mouse. This switches between absolute and relative mode.

Set.Tabletmode.areabyScan

This command defines the absolute drawing area with the digitizer tablet mouse. After the command has been executed, the user will be prompted by a message in the status line to enter the first point of the workspace on the digitizer tablet.

Drawing a diagonal line with the white button held defines the rectangular workspace on the digitizer tablet. Once the mouse button has been released, the picture editor computes a screen workspace that corresponds to the proportions of the rectangle drawn on the digitizer tablet. The computed workspace will automatically be adapted to the maximum work area of the picture editor.

Once the input has been completed, the command issues the current parameters of the defined workspace in the status line. These can be noted for re-use at a later time.
The defined workspace is retained when switching from absolute to relative input mode. When returning to absolute input mode the defined workspace is once again activated. The

workspace defined on the digitizer tablet may not be smaller than 1 cm², otherwise the previous workspace will be used.

Set.Tabletmode.areabyPara <Left> <loWer> <Right> <Upper> [fullsize] [halfsize]

This command defines a workspace on the digitizer tablet by specifying parameters. The pairs of values specify the lower left and upper right corners on the digitizer tablet that will become the new tablet workspace. The numbers are interpreted with the current scaling factor.

Set.DragGrid <Move> <Size> <Rotate> <SHear>

This command sets and enables grid values for DragMode (see **Do.Drag**). Grid values can be defined for all four drag mode functions. If a function is not specified, the previous values will be retained. A zero behind a key word will switch off the grid for this function. A command without parameters will toggle the DragGrid.

Example: `Set.DragGrid Move0.01`

This command sets a grid with a spacing of 0.01 for moving in drag mode.

The DragGrid can also be toggled with the <F> key.

Set.DefaultAttr <Color> <Palette> <Matrix>

This command sets the default values for the relevant attributes. Each newly set point receives these attributes, which can be changed later.
The default attribute values can also be set by starting a new picture. As long as no points are present, the attribute value settings will be interpreted as the default settings and displayed in black on white.

Example: `Set.DefaultAttr Color2`

This command sets the Color attribute for all newly set points to color number 2 of the active palette.

Set.BlankRepeat [ON] [OFF]

This command toggles the automatic blank repeat. With blank repeat enabled, every point involving a blank change is set to the repeat value 1. This enables the automatic blank switching of the *Mark 2* and results in the output of additional points before and after a blank change; the location of the blank change will appear precise and "hard". To generate "soft" blank changes, use this command to disable the automatic blank repeat.

Set.endRepeat <endrepeat>

This command increases the repeat value of the active temporary end point of the picture by the specified value. This results in the exact depiction of this point by the scanner when drawing the picture. The previous repeat value will remain unchanged; the value specified with this command will automatically be added to the previous repeat value. The setting will also be retained for the following pictures.

Example: `Set.endRepeat 2`

This command increases the repeat value of the last point by 2.

Set.Faders.

The **Set.Faders** commands permit the picture in the editor to be modified using the fader controls (#0 to #15). One of the possible transformations from the InitPosition command can be assigned to each of the controls. The value range for the transformations can also be defined freely. Pictures can thus be modified under realistic conditions. The numbers 0 to 15 have been assigned to the fader controls from left to right.

Set.Faders.Off

This command disables the fader controls.
This command can also be executed by clicking the **|Faders|** button. This switches between the three possible fader control modes, Off - Default - User.

Set.Faders.Default

This command enables the fader controls for use in the picture editor. The following transformations have been assigned to fader controls 0 to 3 by default:

Fader control	Operation	Range
0	Rotate X	0 to 360
1	Rotate Y	0 to 360
2	Rotate Z	0 to 360
3	Move Z	0 to 5

All other fader controls remain unused. The scanner assigned to the picture editor will be activated, all others do not display a picture.

This command can also be executed by clicking the **|Faders|** button. This switches between the three possible fader control modes, Off - Default - User.

Set.Faders.User

This command enables the fader controls for use in the picture editor. In this case the assignments and value ranges set with **Set.Faders.Assign** will be applied.

The following transformations have been assigned to fader controls 0 to 5 by default:

Fader control	Operation	Range
0	Rotate X	0 to 360
1	Rotate Y	0 to 360
2	Rotate Z	0 to 360
3	Move Z	0 to 5
4	Move X	-1.0 – 1.0
5	Move Y	-1.0 – 1.0

All other fader controls remain unused. The scanner assigned to the picture editor will be activated, all others do not display a picture.

This command can also be executed by clicking the **|Faders|** button. This switches between the three possible fader control modes, Off - Default - User.

Set.Faders.Assign <Fader> <Operation> <RngBegin> <RngEnd>

With this command the fader control with the number **<Fader>** is assigned the operation stated in **<Operation>**.
The value range for this operation is determined with the numbers **<RngBegin>** and **<RngEnd>**. Stating the value range is not required as an appropriate range will be set by default.

The following operations can be used:

- [ON/OFF->] ON and OFF enable and disable the fader controls with the stated numbers. The settings for the controls are retained.
  Executing this command automatically activates the user mode for the fader controls.
- [Mov/Rot->] MoveX, MoveY, MoveZ, RotateX, RotateY, RotateZ
- [Size->] Size, SizeX, SizeY, SizeZ
- [shear->] shearXByY, shearXByZ, shearYByX, shearYByZ, shearZByX, shearZByY
- [GlobPar->] Horzoff, Vertoff, FoclenHorz, FoclenVert

Note: The settings will be retained even after power down.

Set.Faders.Clearassign

This command clears all previous settings for the fader controls and uses the default settings described under **Set.Faders.User**. User mode is activated with the fader controls.

Set.Film <film1> <film2> <film3> <film4>

This command assigns the film with the specified number to the picture editor for picture output. The scanner assigned to the film then displays the pictures from the picture editor; all others do not display pictures.
This command can also be executed by clicking the **|Film|** button. The picture editor's output is thus switched to the next film.

Set.DisplayMode [Intern] [External]

[External]
This command activates the output of the picture through a film with **[External]**. The scanner assigned to the film with **Set.Film** will be activated. The picture will be displayed on the scanner with all modifications made in the film. To use this command, call up Piced from the film editor with the parameter **<byFilm>**.

[Intern]
The **[Intern]** parameter sends the picture to the scanner with the settings and view from Piced.

Set.SCaLe <scale>

This command sets a scaling factor for the entire coordinate area of the picture editor. Normally, Piced uses the [-1.0,1.0] range for the X,Y and Z coordinate values. This sometimes leads to inconvenient numerical entries when it becomes necessary to enter pictures in other units.
If a scaling factor has been stated, all coordinate-related numerical input will be interpreted using the factor. The scaling factor is only applied in the picture editor; the film editor will take the picture over within the [-1.0, 1.0] range.

Even if a freely determined scaling factor has been established, all coordinate-related numerical input can be entered with a fixed scaling factor. The following symbols preceding a number will cause Piced to interpret the number with a fixed scaling factor, independently of the freely defined scaling factor.

- $: The value is will be interpreted within the -1.0 to 1.0 range
- % : The value is will be interpreted within the -100.0 to 100.0 range
- # : The value is will be interpreted within the -32767 to 32767 range

Example: `Set.Margins Left%-50 Right%50`

This command corresponds to the **Set.Margins Left-0.5 Right0.5** command as long as no value other than 1 was established with **Set.SCaLe**.

Set.AnchorCheck Automatic <Length> <Angle> <Repeat>

Options: /ClrManual

A picture for the *Mark 2* contains straight lines, curves and sharply displayed corner points. These corner points are known as anchor points. Anchor points are output with a repeat value set by the user so that the scanner has enough time to reach the point's position precisely and then start a motion in another direction. Users often set anchor points and their repeat values manually, but the Lasergraph DSP also features a command for this purpose.

<Length> <Angle>
This command automatically checks the edited and newly set points for the conditions stated in **<Length>** and **<Angle>**. If these conditions are true, i.e. the line before or after the point is longer than the distance specified in **<Length>** and the angle of the change of direction greater than the angle stated in **<Angle>**, that point will be turned into an anchor point. This is under the condition that both lines connecting to the point are visible, not blanked.

<Repeat>
Anchor points automatically receive a new repeat value which is set with the **<Repeat>** parameter. If the anchor point no longer fulfills these conditions, its earlier repeat value will be restored. Anchor points can also be determined individually. Active points can be turned into anchor points with the space bar or by clicking on the **|Anc.|** button with the middle mouse button. An anchor point defined in this manner will be exempted from automatic checking.

/ClrManual
Manually set anchor points can be reverted to "normal" points with the command **Set.AnchorCheck /ClrManual**.

Example: `Set.AnchorCheck Automatic Length0.03 Angle40.0 Repeat4`

This command sets the repeat value of 4 for the points for which one of the two connected lines is longer than 0.03 and the angle between these lines is less than 40.0 degrees.

Automatic anchor point checking can be toggled with a left mouse click on the **|Anc.|** button. The <^> key has the same function.

Set.AnchorCheck Off

This command disables automatic anchor point checking.
Automatic anchor point checking can be toggled with a left mouse click on the **|Anc.|** button. The <^> key has the same function.

Set.CatchMark ON / OFF

This command toggles the capture radius (circle with cross hairs in the middle around the point).

Do.

This command and its subcommands edit all selected points. The correct selection and skilled use of commands can save a considerable amount of manual work.

Do.DELete

This command deletes all selected points. The command must be confirmed to avoid the accidental deletion of points.

This command can also be executed with the <Ctrl+Del> key combination.

Do.CuT

This command copies all selected points to the clipboard and removes them from the picture. The previous content of the clipboard will be overwritten. The points of the picture on the clipboard form a continuous line; changes may occur to the picture if the sequence of the selected points was not continuous.

This command can also be executed by clicking the **|Cut|** button. A double-click on the button executes the command directly.

Do.CoPy

This command copies all selected points to the clipboard. The previous content of the clipboard will be overwritten. The points of the picture on the clipboard form a continuous line; changes may occur to the picture if the sequence of the selected points was not continuous.

This command can also be executed by clicking the **|Copy|** button. A double-click on the button executes the command directly.

Do.Attributes <Color> <Palette> <Matrix> <Repeat> <Blank> <Vector>

This command sets the attribute values of the selected points. The attribute specified in the command is set to the specified value for all selected points.
A + or - preceding the numerical value after the key word will cause the value to be interpreted as relative, i.e. the value will be added or subtracted from the existing attribute value.
The attributes **<Blank>** and **<Vector>** are switched off with the value 0 and on with 1.

Example: `Do.Attributes Repeat2`

This command sets the repeat value of all selected points to 2.

The command for the relevant attribute can also be selected with the following key combinations insofar as at least one point has been selected:

Attribute	Keys
Color	<Ctrl+c>
Palette	<Ctrl+p>
Matrix	<Ctrl+m>
Repeat	<Ctrl+r>
Blank	<Ctrl+b>
Vector	<Ctrl+v>

Do.Transform [again] REVerse <Move> <MoveX> <MoveY> <MoveZ> <Size> <SizeX> <SizeY> <SizeZ> <Rotate> <RotateX> <RotateY> <RotateZ> <shearXByY> <shearXByZ> <shearYByX> <shearYByZ> <shearZByX> <shearZByY>

This command transforms all selected points with the specified operations. Up to 18 operations can be performed simultaneously. The operations will be performed in the order in which they have been stated.
The **[again]** command or the command without parameters performs the previous operations again with the same values.
REVerse repeats the last operations with a reversed operational sign.

<Move>
This parameter moves all selected points on each axis by the specified value.
The **<MoveX>**, **<MoveY>** and **<MoveZ>** parameters move the selected points by the specified value along the corresponding axis.

<Size>
This parameter scales all selected points for each axis by the same value.
The **<SizeX>**, **<SizeY>** and **<SizeZ>** parameters scale the selected points by the specified value along the corresponding axis.

<Rotate>
This parameter rotates all selected points on each axis by the specified value.
The **<RotateX>**, **<RotateY>** and **<RotateZ>** parameters rotate the selected points by the specified value around the corresponding axis.

<shear>
This parameter shears all selected points with the stated value. The shearing is performed in the direction of the second axis of the coordinate system stated in the key word. All possible combinations are available.

The transformations may result in points that lie outside the set range of values.
The coordinates of these points take on the maximum possible value and assume a position at the edge of the value range. The transformations refer to the internal coordinate system, i.e. scaling and shear operations use the origin as the zero point; rotations use the X, Y or Z axes as rotational axes. Mirroring is possible by using negative values.

Example: `Do.Transform MoveY0.1`

This command moves all selected points by 0.1 in the Y direction.

Do.Drag

This command launches the interactive transformation mode, the so-called drag mode.
When the mode is launched, a square appears around the selected points and a new window opens. A button in this window represents all possible transformations.
Clicking on the appropriate button activates the desired transformation. Moving the mouse now changes the parameter values. The valid value is displayed under the name of the activated transformation.
A value can also be entered using the keyboard by clicking on the **Value** field with the mouse to activate keyboard input. The results of the transformation are displayed in a simplified manner in the square.

The **|Show|** button provides an exact preview of the results. The scanner shows the changes resulting from the transformation during the mouse action. If the result is satisfactory, the transformation is executed with the set values by clicking on the **|Ready|** button and drag mode is ended.

The **|Abort|** button cancels drag mode without changing the picture. The **|Undo|** button permits previous transformations to be cancelled one step at a time.

The transformations may result in points that lie outside the set range of values. The coordinates of these points take on the maximum possible value and assume a position at the edge of the value range. The transformations refer to the internal coordinate system, i.e. scaling and shear operations use the origin as the zero point; rotations use the X, Y or Z axes as rotational axes. Mirroring is possible by using negative values.

Drag mode can also be activated by the key combination <Ctrl+d>.

The following keyboard shortcuts are available during drag mode:

<d>	Switches one transformation further
<D>	Switches one transformation back
<G>	Switches on a precise preview (corresponds to the **|Show|** button)
<Return>	Executes the transformation and ends drag mode
<Esc>	Cancels drag mode
<F>	Toggles the grid for the values
<Tab>	Switches to keyboard input in the **value** field. The value entered must be confirmed with <Return>. <Esc> cancels the input and returns to drag mode. Both values for the move transformations can be edited using the <PgUp>and <PgDn> keys in the **value** field. "++" or "--" will cause the input to be interpreted in a relative manner, i.e. added or subtracted from the current values.

Do.Interpolate <Distance> <InsPoints> <PicPoints> <OnePoint>

This command inserts additional points between all successive selected points. These additional points are positioned on the interconnection lines between the point pairs due to the applied linear interpolation. Therefore at least two consecutive points must be selected in order to use this command. The additional points can be inserted according to four different criteria. Of the possible four parameters, only one can be applied at any given time.

<Distance>
This parameter specifies the distance at which the interpolated points should be inserted.

<InsPoints>
This inserts exactly the specified number of points in the complete selected area.

<PicPoints>
This parameter can be used to determine the number of points the complete picture should have after the interpolation.

<OnePoint>
This always inserts the specified number of points between a point pair.

Example: `Do.Interpolate OnePoint3`
This command inserts exactly three points between two successive selected points.

The last used parameters will automatically be applied if the command is executed without parameters.

Do.SPLine <Distance> <InsPoints> <PicPoints> <OnePoint>

This command inserts additional points between all successive selected points. The additional points will be located on a curve determined on the basis of the adjacent points. A spline interpolation is used.
The points will be set more closely at locations with a strong curvature. Existing points will not be changed.
To perform the interpolation, at least two successive points must be selected. Of the possible four parameters, only one can be applied at any given time.

<Distance>
This parameter specifies the distance at which the interpolated points should be inserted.

<InsPoints>
This inserts exactly the specified number of points in the complete selected area.

<PicPoints>
This parameter can be used to determine the number of points the complete picture should have after the interpolation.

<OnePoint>
This always inserts the specified number of points between a point pair.

Example: `Do.SPline OnePoint3`

This command inserts exactly three points between two successive selected points.

The last used parameters will automatically be applied if the command is executed without parameters.

Do.Bspline <Distance> <InsPoints> <PicPoints> <attract>

This command performs a B-Spline interpolation with the successive, selected points.
The result is a curve that follows the course of the selected points and that is appended to the existing picture. The points of the computed curve are located near the starting points. The form of the curve depends on the distance of the starting points and the angle of the interconnection lines to one another.

<attract>
The value of this parameter determines how much of an effect the adjacent control points are to have. This value must always be stated as the second value after the value for the selected parameter (e.g. "PicPoints32 4"). A smaller value will cause the computed curve to follow the previous curvature more closely; the value cannot be smaller than 2, however.
Of the possible three parameters, only one can be applied at any given time.

<Distance>
This parameter specifies the distance at which the interpolated points should be inserted.

<InsPoints>
This inserts exactly the specified number of points in the selected area.

<PicPoints>
This parameter can be used to determine the number of points the complete picture should have after the interpolation.

Example: `Do.Bspline PicPoints38`

This command will append points until the complete picture has a total of 38 points.

The last used parameters will automatically be applied if the command is executed without parameters.

Do.SPHere <Hangle> <Vangle> <NormHRad> <AbsHRad> <NormVRad> <AbsVRad>

Options: /Noradius /Radius

This command transforms all selected points in the form of a sphere. Sometimes it is better to perform an interpolation prior to executing this command to ensure that a sufficient number of evenly distributed points are present in the picture.

<angle>
The **<Hangle>** and **<Vangle>** parameters specify the maximum angle by which the transformed points will be rotated around the Y and X axes. A value of 360° results in a sphere; smaller values will result in a sphere segment.

<AbsRad>
Specifying **<AbsHRad>** and **<AbsVRad>** establishes the radius of the sphere in horizontal and vertical directions.

<NormRad>
The **<NormHRad>** and **<NormVrad>** parameters specify the radius of the sphere in a relative manner. If the angles stated with **<Hangle>** and **<Vangle>** are less than 90°, the result will be adjusted to the full available width and height.
The values of **<NormHRad>** or **<NormVRad>** scale the result of this adaptation.

/Radius
This option results in the computation of a spherical form with a radius resulting from the value of the Z coordinates of each point.

/Noradius
This command computes the spherical form on the basis of the specified values and is automatically activated if nothing else has been specified.

Example: `Do.SPHere Hangle180.0 Vangle90.0 AbsHRad0.8 AbsVRad0.8`
This command generates the form of a quarter sphere with a radius of 0.8.

Only the angle must be specified if the command is to be performed again; the other parameters are optional. If no other parameters have been specified, a spherical form with the last specified values will be assumed.

Do.CYLinder <Angle> <NormRad> <AbsRad> <TopRad>

Options: **/Noradius /Radius**

This command transforms all selected points in the form of a cylinder. Sometimes it is better to perform an interpolation prior to executing this command to ensure that a sufficient number of evenly distributed points are present in the picture.

<Angle>
This parameter specifies the maximum angle by which the transformed points will be rotated around the Y-axis. A value of 360° results in a circle; smaller values will result in a circle segment.

<AbsRad>
Specifying **<AbsRad>** establishes the radius of the imaginary cylinder in the horizontal direction.

<NormRad>
This parameter specifies the radius of the cylinder in a relative manner. If the angle stated with **<Angle>** is less than 90°, the result will be adjusted to the full available width. The value of **<NormRad>** scales the result of this adaptation.

<TopRad>
The value of this parameter specifies the ratio of the upper circle to the lower one. This also permits the use of forms related to the cylinder. A **<TopRad>** value of 0 results in a cone; values between 1 and 0 will result in truncated cones. Negative values can also be used and will result in the shape of an hourglass.

/Radius
This option results in the computation of a cylindrical form with a radius resulting from the value of the Z coordinates of each point.

/Noradius
This command computes the cylindrical form on the basis of the specified values and is automatically activated if nothing else has been specified.

Example: `Do.CYLinder Angle180.0 AbsRad0.8 TopRad0.5`
This command will generate the form of a truncated cone in which the upper radius is half the size of the base radius.

Only the angle must be specified if the command is to be performed again; the other parameters are optional. If no other parameters have been specified, a cylindrical form with the last specified values will be assumed.

Do.TORus <thickness> <Hangle> <Vangle> <NormHRad> <AbsHRad> <NormVRad> <AbsVRad>

Options: /Noradius /Ringradius /Tuberadius

This command transforms all selected points in the form of a ring. Sometimes it is better to perform an interpolation prior to executing this command to ensure that a sufficient number of evenly distributed points are present in the picture.

<thickness>
Once the **<thickness>** parameter which defines the ratio of the inside radius to the outside radius of the ring has been defined, the other parameters may also be entered.

<angle>
The **<Hangle>** and **<Vangle>** parameters specify the maximum angle by which the selected points will be rotated around the Y and X axes. A value of 360° for **<Hangle>** results in a ring; smaller values will result in a ring segment. The value for **<Vangle>** will determine whether a full cross section will result or only a segment.

<AbsRad>
Specifying **<AbsHRad>** and **<AbsVRad>** establishes the radius of the imaginary ring in horizontal and vertical directions.

<NormRad>
The **<NormHRad>** and **<NormVRad>** parameters specify the radius of the ring in a relative manner. If the angles stated with **<Hangle>**and **<Vangle>**are less than 90°, the result will be adjusted to the full available width and height. The values of **<NormHRad>** or **<NormVRad>** scale the result of this adaptation.

/Ringradius
The **/Ringradius** option will cause the command to compute the radius of the ring from the value of the Z coordinate of the point.

/Tuberadius
The **/Tuberadius** option will have the same effect on the radius of the ring cross section.

/Noradius
The **/Noradius** option is automatically activated if no other option is specified.

Example: `Do.TORus 0.5 Hangle180.0 Vangle180.0 AbsHRad1.0 AbsVRad0.5`
This command generates a ring form with an elliptical cross section.

Only the angle must be specified if the command is to be performed again; the other parameters are optional. If no other parameters have been specified, a ring form with the last specified values will be assumed.

Do.Extrude <offsetX> <offsetY> <offsetZ> <Repeat>

Options: /Hiding

This command assigns a spatial depth to the selected points. All selected points will be moved along a vector to realize this. The starting points are preserved and are connected with the newly generated points, creating a three-dimensional body.

<offset>
The **<offset>** parameters determine a direction vector and a value for the distance by stating one or more **<offset>** parameters.

<Repeat>
This parameter determines a repeat value for the newly generated points.

/Hiding
The **/Hiding** option realizes the interconnection lines between the old and new points as polygons. The first point of each polygon is output with the attribute Vector=On (see Attributes, vector section). Next, the normal vectors can be computed for these polygons with the **Make.Hiding** command. The computer can then recognize the polygons as spatial planes.
The orientation of the surfaces can be reversed with the **Do.reOrientate** command.

Example: `Do.Extrude offsetZ0.4 offsetY0.4 Repeat2`
This command extrudes all selected points in the Z and Y directions; the newly generated points are given the repeat value 2.

Do.Rotarybody <Angle> <Steps> <aXis> <repeat>

Options: **/Hiding**

This command generates a rotary body from the selected points. In the process, all selected points are copied and these points are rotated around the specified axis of the coordinate system.

<Angle>
This parameter specifies the angle around which the points are to be rotated.

<Steps>
Determines how many intermediate steps of the rotation are to be made visible.

<aXis>
This parameter determines whether the X,Y or Z-axis of the coordinate system should be used as the rotation axis.

<repeat>
This parameter determines a repeat value for the newly generated points.

/Hiding
This option realizes the interconnection lines between the old and new points as polygons. The first point of each polygon is output with the attribute Vector=On.
Next, the normal vectors can be computed for these polygons with the **Make.Hiding** command. The computer can then recognize the polygons as spatial planes.

The orientation of the surfaces can be reversed with the **Do.reOrientate** command.

Example: `Do.Rotarybody Angle180.0 Steps6 aXisY 3`
This command generates a rotary body with 6 intermediate steps; all selected points are rotated by 180 degrees around the Y-axis and the newly generated points receive a repeat value of 3.

Do.CoNe

This command transforms all selected points in the form of a cone . The cone moves through the coordinate space with its point along the axis perpendicular to the viewing plane from the direction of the active viewing plane. In the process, the selected points of this cone are moved and result in the new form of the picture. The form of this cone fits exactly into the coordinates space with a basic circle radius =1 and height =2.

Do.HeliX <angle>

This command gives the selected points the form of a helix. The points are rotated around the Y-axis.

<angle>
This parameter determines the number of degrees by which the helix should be rotated. Values exceeding 360° are also possible.
The helix effect is the result of the nature of the computation for this command. The point with the smallest Y value is not turned; the one with the greatest Y value is turned the full angle. The points in between are turned by a smaller angle according to their Y coordinates.

Example: `Do.HeliX 360.0`

This command puts all selected points into a helical form in which the point with the highest Y value is turned by 360.0 degrees.

Do.ALign [All] <XLow> <XMid> <XHigh> <YLow> <YMid> <YHigh> <ZLow> <ZMid> <ZHigh>

This command aligns all selected points in one or more directions of the coordinate space. The points can be aligned in the direction of the lowest (**<Low>**) or highest (**<High>**) value for each axis.
Specifying **<Mid>** for an axis centers the points along this axis. If no parameter or the parameter **<All>** is stated, all selected points will be centered in the directions of all three axes.

Example: `Do.ALign XHigh`

This command moves all selected points in the X direction until the point with the highest X value reaches the limit of the permitted coordinate range.

Do.Maximize [All] <MaximizeX> <MaximizeY> <MaximizeZ>

This command scales all selected points up to the maximum possible value. All selected points will be enlarged until one point reaches the limits of the permitted coordinate range. All selected points are scaled with the resulting factor; the proportions of the points are preserved.

<Maximize>
The **<MaximizeX>**, **<MaximizeY>** and **<MaximizeZ>** parameters have the same effect, but only in the direction of the specified axis. Two or three axes can also be specified. However, scaling is performed separately for each axis; i.e. a different scaling factor is applied to each direction, unlike the **[All]** parameter, in which only one scaling factor is applied to all directions.
If nothing is specified, or the **[All]** parameter was stated, all selected points will be scaled to the maximum permitted coordinate values using the same factor.

Example: `Do.Maximize MaximizeY`

This command scales all selected points in the Y direction until the point with the highest or lowest Y value reaches the limit of the permitted coordinate range.

Do.AnchorCheck <Length> <Angle> <Repeat>

This command performs an anchor check for all selected points. All points that fulfill the criteria stated with **<Length>** and **<Angle>** will become anchor points.
The value of the repeat attribute for all anchor points will be set to the value stated in the **<Repeat>** parameter. A point becomes an anchor point when the distance to one of its neighboring points becomes greater than the value of **<Length>** and the angle between the lines connecting to the point is less than the value stated in **<Angle>**. Neither of the interconnecting lines may be blanked.

Example: `Do.AnchorCheck Length0.03 Angle40.0 Repeat4`

This command sets the repeat value of all selected points to the value 4 where one of the two interconnecting distances is longer than 0.03 and the angle between the distances is less than 40.0 degrees.

Do.enGrid

This command moves all selected points to the closest points on the selected drawing grid. The drawing grid must be activated (<f> key) in order to execute this command.

Do.ColorCycle <Colors> <Startcol> <Pixpercol>

Options: /Restart

This command assigns a different color to the selected points. The color for each point is based on the sequence of the colors in the color palettes.

<Colors>
The number of colors used is set with the **<Colors>** parameter. The sequence of the colors corresponds to the colors of the four color palettes, the value thus cannot be higher than 64.

<Startcol>
This value determines the number of the color that will be used as the start. The first point is given the **<Startcol>** color, the following points will be assigned the subsequent colors of the palette.

<Pixpercol>
This parameter determines how many consecutive points will be assigned the same color.

/Restart
This option causes the coloring to restart with the **<startcol>** color after each blanked line.
Executing this command causes the values of the color attributes to be reset for all selected points and a message to be issued about the number of points that were changed.

Example: `Do.ColorCycle Colors8 Startcol2 Pixpercol2`

This command gives all selected points a new color value. The color used first is number 2 from palette 0. When color number A is reached the coloring is restarted with color 2; two consecutive points will receive the same color value.

Do.Shape

This command can be used to edit the section of selected points. The points must be a continuous selection and at least three points must be selected for this command to be performed.
The first and last point of the selected range will be connected by a line when this command is invoked. The mouse is then used to establish a third point to create a curve upon which the selected points are arranged as a parabola. A preview shows the result on screen and on the scanner during the mouse action.

The command is invoked and ended with the left mouse button. A click with the middle mouse button establishes the point and switches to the next point that can be moved. The first selected point, the last selected point and a supporting point on the curve can be edited.
The left mouse button or "Return" ends the input and executes the command. The command can be cancelled without changes with the right mouse button or by pressing “Esc”.

Do.PointSwap

This command reverses the order of the selected points. The last point becomes the first and vice versa. In other words, the line between the first selected point and the preceding point is connected to the last selected point after the command has been executed. The same applies in reverse for the last selected point. The command can only be used on a consecutive selection. If the selection is interrupted by a deselected point, the command will be cancelled and an error message displayed.

Do.reOrientate

This command swaps the visible side of spatial surfaces with the invisible side. This reverses the direction of the normal vector that points toward the visible side. To execute the command, both points of the normal vector, i.e. the line that indicates the alignment of the surface (Vector attribute = On), must be selected. This command only works with already existing normal vectors.

Do.deForm

Options: /Clip

This command deforms all selected points according to the picture on the clipboard.
The X coordinates are used as the parameter for this, i.e. one point at the far left of the X-axis is given the value 0, at the far right the value 1, and in between a mean value in accordance with the X coordinates.
Parameters from 0 to 1 are also assigned to the points on the clipboard according to their order, i.e. the first point on the clipboard has the value 0; the last point is assigned the value 1. Each selected point is then associated with a point on the clipboard in accordance with this parameter.

/Clip
If this option is not stated, a check is initially run to ensure that the deformation will not lead to the exceeding of value ranges. If any ranges are exceeded, the operation will be aborted with a message stating the number of points exceeding their ranges.
If the deformation succeeds immediately or the **/Clip** option has been specified, the deformation will be carried out and the number of points that were not deformed or that were clipped will be stated.
If the deformation results in a position outside the permitted range for one or more points, the **/Clip** option must be stated for the command to be executed. The affected points will then be on the edge of the coordinate range.

Do.RUBbermat

Options: /Clip

This command deforms all selected points. The deformation is determined on the basis of the deformation of a grid. This grid can be distorted in various directions like a rubber mat. When executed, this distortion will be applied to the selected points. The points will be moved in XY direction according to the grid; in the Z direction an offset value will be added to the existing value.
A square grid will appear when the command is invoked. The corners of this grid can be moved, thus determining the degree of the deformation. The next corner point can be activated with the middle mouse button or the space bar. The left mouse button or the "Return" key finishes the input and executes the command. The right mouse button or "Esc" cancels the command.

/Clip
If the deformation results in a position outside the coordinate range for one or more points, the **/Clip** option must be stated for the command to be executed. The affected points will then be on the edge of the coordinate range.

Do.Wobble <Xrange> <Yrange> <Zrange> <Range>

This command moves all selected points in all three directions in a random pattern.

<Range>
This parameter indicates the maximum distance that the selected points may be moved. The selected points will be moved in all three axis directions when **<Range>** is specified.
If the **<Xrange>**, **<Yrange>** or **<Zrange>** parameters are stated, the points will only be moved in the direction of the stated axis. The value of the parameter states the maximum range of the movement.

Example: `Do.Wobble Range0.3`

This command moves all selected points by a maximum of 0.3 in all directions.

Do.SoftBlank <Repeat>

Options: **/OnlyUp /OnlyDown**

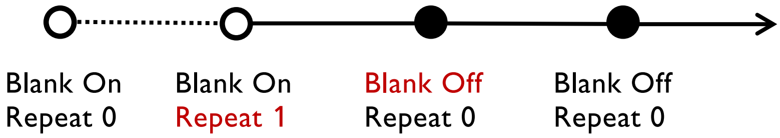
The command **Do.SoftBlank** can be used to convert hard line endings into soft line endings. In addition to the fact that the pictures look more pleasant, the refresh rate is also increased in most cases.

<Repeat>
The parameter **<Repeat>** is optional.
If no repeat is specified, a repeat of 4 will be used.

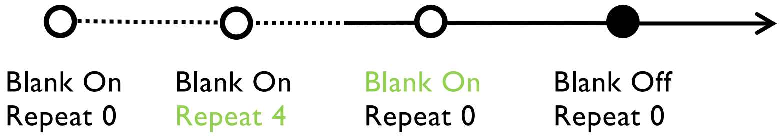
/OnlyUp
By default, all line beginnings and line endings will be modified.
If **/OnlyUp** is specified, only line beginnings will be modified.

/OnlyDown
By default, all line beginnings and line endings will be modified.
If **/OnlyDown** is specified, only line endings will be modified.

Example of a line beginning with hard blank change:



The same line beginning after **Do.SoftBlank** is entered:



Notes:

- The command **Do.SoftBlank** only affects lines with hard blank changes (repeat > 0). Therefore, closed contours, such as circles, are not affected.

- To prevent short lines from disappearing completely, only lines with a minimum length of 3 resp. 2 sections are modified.

Make.

This command and a number of its subcommands generate points with preset forms such as circles and words. The other commands of this group affect all points of a picture, to optimize the picture output for example.

Make.Circle <Points>

Options: /Right /Left /Overlap /Nooverlap

This command interactively generates a circle in the active view. When this command is invoked, the cross hairs are replaced by a circle; the radius of this circle can be increased or decreased with the mouse or cursor keys.
Clicking with the middle mouse button or using the space bar will activate the center point and allow the circle to be positioned.
Another click with the middle mouse button sets the center point and permits the radius to be adjusted once again.
The left mouse button sets the active circle point and the circle is drawn with the specified number of points. The right mouse button or the "Esc" key aborts the command without drawing the circle.
The points of the circle are all generated with a repeat value of 1 to ensure a clean depiction.

If no parameters or options have been specified, the command will be executed with the values used last. Values that were last used with related commands (Make.Ellipse and Make.Arc) will also be used.

All points of the drawn circle are automatically selected after the command has been executed and can be processed further immediately.

<Points>
This parameter determines the number of points used to draw the circle.

/Right /Left
This options determine whether the circle is to be drawn clockwise (**/Right**) or counterclockwise (**/Left**).

/Overlap
This option inserts a number of additional points. The circle does not begin and end with the same point, but begins several points before and ends several points later. A number of line sections are thus output twice. The first and last line sections are blanked. A "soft" blank change ensures that the output of the circle appears precisely round.

/Nooverlap
This option results in a circle being drawn with exactly the specified number of **<Points>**, with the same starting and end point.

Example: `Make.Circle Points32 /Nooverlap`

This command generates a circle with exactly 32 points and no overlap.

Make.Ellipse <Points>

Options: /Right /Left /Overlap /Nooverlap

This command interactively generates an ellipse in the active view.
The parameters and options correspond to those of the command **Make.Circle**. Unlike the circle, not only the radius is specified, but also the ratio of the two radii to one another. This ratio is displayed in the form of a rectangle.

All points of the drawn ellipse are automatically selected after the command has been executed and can be processed further immediately.
If no parameters or options have been specified, the command will be executed with the values used last. Values that were last used with related commands (Make.Circle and Make.Arc) will also be used.

Example: `Make.Ellipse Points32 /Nooverlap`

This command generates an ellipse with exactly 32 points and no overlap.

Make.Arc <Points> <Distance>

This command interactively generates an arc in the active view.
When the command is invoked a point is inserted and activated after the active point that will determine the end point of the arc. This point can be positioned with the left mouse button; next, the length of the arc can be set by determining the center point of the associated full circle with the mouse or the up and down cursor keys.

Once the center point has been determined with the left mouse button, the end point can be repositioned.
Clicking the middle mouse button or using the space bar changes the drawing direction, the arc will appear on the other side of the bowstring.
A double-click with the left mouse button or using the return key will end the command and the arc will be drawn with the selected parameters. The next arc can now be drawn.
The command can be cancelled with the right mouse button or "Esc" key without drawing another arc.
All points of the drawn arc are automatically selected after the command has been executed and can be processed further immediately.

<Points>
This parameter specifies a fixed number of points to be used to draw the arc. The specified value refers to a full circle. The number of points used to draw the arc is thus dependent on the length of the section that the arc occupies on the full circle.

<Distance>
This parameter specifies the distance at which the points are to be set on the arc.
The most recently used parameters will be used again if the command is invoked without parameters.

Example: `Make.Arc Points8`

This command generates an arc with 8 points.

Make.Shape <Points> <Distance>

This command interactively generates a parabolic curve. When the command is invoked a point is inserted and activated after the active point that will determine the end point of the curve. This point can be positioned with the mouse.
A click with the middle mouse button or using the space bar establishes the point and switches to the next point that can be moved.

The starting point, end point and a supporting point on the curve can be edited. A preview shows the result on screen and on the scanner.
The left mouse button or the "Return" key executes the command and the curve is drawn with the specified parameters. The next curve can then be drawn.
The command can be cancelled with the right mouse button or "Esc" key without drawing another curve.
All points of the drawn curve are automatically selected after the command has been executed and can be processed further immediately.

<Points>
This parameter specifies a fixed number of points to be used to draw the curve.

<Distance>
This parameter specifies the distance at which the points are to be set on the curve.
The most recently used parameters will be used again if the command is invoked without parameters.

Example: `Make.Shape Points16`
This command generates a curve with exactly 16 points.

Make.Lissajous <points> <FreqX> <FreqY> <FreqZ> <PhaseX> <PhaseY> <PhaseZ>

Options: **/XSinewave /XTriangle /XsQuare /XsAw**
/YSinewave /YTriangle /YsQuare /YsAw
/ZSinewave /ZTriangle /ZsQuare /ZsAw
/genagain /Overlap /Nooverlap

This command generates a two- or three-dimensional Lissajous figure. The generated patterns are computed on the basis of oscillations.
A different oscillation can be used for each axis. Waveforms and associated frequencies can thus be set for the X, Y and Z axes. The starting value of the oscillation is determined with the Phase parameter.
Not all parameters and options are required to receive a result.

<points>
First, the number of points that the Lissajous figure should contain must be specified with the **<points>** parameter.

Next, at least two frequencies, e.g. for the Y and X axes must be specified; the command can now be executed. A sine wave with a phase shift of 0 can be used as a waveform. The result will fill the entire coordinate range. A picture present previously will be preserved and the points of the figure will be inserted at the active point.

<FreqX/Y/Z>

This parameter is used to set the frequency of the used waveform. A value of 1 means that the oscillation along the relevant axis is executed exactly once in the [-1.0, 1.0] range. Closed figures will result as long as the frequencies are entered as whole numbers.

<PhaseX/Y/Z>

This parameter determines the starting value of the selected oscillation. In other words, the oscillation is moved along the axis. A value of 180 moves the curve by exactly one half the length of one period.

A different waveform can be selected for each axis. Four different waveforms are possible:

- **Sinewave** specifies a sine
- **Triangle** a triangular
- **sQuare** a square
- **sAw** sawtooth waveform

/genagain

The waveforms last used are sent to the input line when the **/genagain** option is used.

/Overlap

This option inserts a number of additional points. The figure does not begin and end with the same point, but begins several points before and ends several points later. A number of line sections are thus output twice. The first and last line sections are blanked.
A "soft" blank change ensures that the figure is output precisely.

/Nooverlap

This option results in a figure being drawn with exactly the specified number of **<points>**, with the same starting and end point.

Example: `Make.Lissajous 32 FreqX2.0 FreqY4.0 /XSinewave /YSinewave`
This command generates a Lissajous figure with 32 points; a sine wave is used for pattern generation in the X and Y directions.

Make.Word <catalog> <string> <distance>

Options: **/Proport /Kerning /sameSize /UseProtect**

This command generates a text as a picture. The text is inserted at the end of an existing picture. The size of the letters is based on the length of the input text. The text will be distributed across the full available width of the horizontal axis of the active view. The spacing between the individual letters can be set with the **<distance>** parameter.

<catalog>

This parameter specifies a picture catalog from the picture memory that contains the letters and characters of the selected font. The catalog must be loaded in the picture memory in advance. The included fonts can be found in the **C:\General** folder.

<string>
The text is specified with the **<string>** parameter. The text must be enclosed in quotation marks (e.g. "My Text") if it contains spaces. Otherwise the remaining text in the input line will be interpreted as text for the picture.

<distance>
This parameter determines the spacing between the characters. The value follows the **<string>** parameter, separated by a space. A value of 1.0 corresponds to the width of one letter. With a value of 0 the letters are set without spacing. Negative values can also be used.

/Proport
This option adjusts the spacing between the letters to suit the width of the individual letters. Without this option, the same width will be used for all characters.

/Kerning
This option results in letters being set in the same manner as with the **/Proport** option; the difference is that the letters can also overlap.

/sameSize
This option generates text of the same size as the last generated text. This permits lines with the same character size to be placed below one another. The best approach is to start with the longest line, then generate the remaining text and finally position the lines. If the input text is too long for the previously used font size, the size can be scaled with a fixed factor (e.g. 0.5 or 0.33, etc.).

/UseProtect
When creating a word from a protected font, you will be notified that the created picture will have a protection. To confirm that, you have to use the option **/useProtect**.

Example: `Make.Word Helvetica "Hello!" 0.1`
This command generates the text "Hello!" in Helvetica with a letter spacing of 0.1.

Make.Hiding

This command automatically computes the normal vectors required to suppress the display of hidden surfaces.
This requires the generation of a picture that shows the three-dimensional object with its exterior surfaces. In this picture, each surface must be defined as a closed polygon by its surrounding edges. In other words, edges at which two surfaces meet are drawn twice, once for each surface.

The polygon that describes the surface must be drawn in the mathematically positive sense of rotation if the normal vector is to subsequently display the direction of the view. A blanked point must be inserted after the last point of each surface, even if the next surface starts at the same point. Confusing lines running across the picture will otherwise appear in the output, as the end of the surface will not be recognized correctly.

The "Vector" attribute must be set to "On" for each point at which a new surface begins. The attribute is the indicator for this command that a new surface is to begin at this point for

which a normal vector must be computed. No point may be active when executing this function.
The function then inserts a vector before each surface which is perpendicular to it. The normal vector is defined by two points for which the "Vector" attribute is set to "On".
When the command has been successfully executed, the number of normal vectors generated is displayed and the picture is then ready for display with hidden surfaces. If an error occurs, the command will abort while activating the point at which the error occurred to permit the error to be eliminated quickly.

This command can be repeated; i.e. the function recognizes normal vectors already computed and computes them again without inserting additional points.
The points belonging to the vector will be inserted for surfaces for which a normal vector has not yet been computed.
The function can thus be called up again after an error indicated by it has been eliminated without the already-performed partial computation of the normal vector affecting the result.

Make.BlankRepeat

Options: /set1to0

This command searches the current picture for points located between a blanked and a visible line section (blank change). The repeat value of these points is set to 1 if no higher value was specified earlier. This results in the output of as many additional points at the location of a blank change as is stated in the advanced parameters.

/set1to0

This option sets the repeat value of all points at a blank change to 1, even if the point had a higher repeat value before. For points at which the blanking does not change, the repeat value is set to 0 if 1 is specified.

Make.Optimize Blanking <Repeat> <Straight>

This command optimizes the point distribution and repeat values in the current picture.

Blanking

This parameter deletes points located on a blanked section. The points of the blank change are retained and all points between the start and end points of the blanked section are deleted.

<Repeat>

This parameter changes the repeat value of all points that have a higher value than the specified one. The repeat value of these points is set to the specified value. The repeat values of the other points are not affected.

<Straight>

This parameter deletes points in the current picture that are located close to one another and in which the connected line sections have a flat angle to one another. The value indicates the distance up to which a point will be interpreted as being on a straight line.

Example: `Make.Optimize Repeat4 Straight0.05`

This command optimizes the distribution of points and limits the repeat value to 4.

Make.ReShuffle <number>

This command changes the sequence of the points in the point list. The current picture will be divided at the location of the point with the specified number. The specified point will become the new first point of the picture; all previous points will be appended to the previously last point of the point list with a blanked line.

<number>
This parameter indicates the number of the point that is to be the new first point of the picture. If no point is specified, the command will use the active point.

Example: `Make.ReShuffle 4`

This command renumbers all points; the previously fourth point of the picture will become the new first point.

Make.SoftBlank <number>

This command blanks a closed object "softly". A closed object is a series of lines that have their start and end points at the same position, which contains no blanked sections and is bordered by blanked lines.
This command inserts additional points along the contour of the object, before the start point and after the end point. The inserted points are blanked and have a repeat value of 0. The scanner can display the object contour more cleanly due to these superimposed lines as the change in direction takes place continuously.

<number>
This parameter indicates the number of a point contained by the target object in order to select the object. If no point is specified, the command will use the active point.

Make.PreCompute <picsize>

Options: **/ExtractRepeat**

This command prepares the picture for export to an external laser computer format. As most other laser computer systems do not work with dynamic interpolation and automatic blank changing, the pictures must be prepared for output to other systems to ensure that they will be displayed correctly.

To prepare a picture for export, it is first edited normally and its advanced parameters are set so as to ensure a good display with the *Mark 2*.
Next, the Make.PreCompute command is executed. This command uses the advanced parameter settings to insert additional interpolated points on the lines and points on the blanked lines to achieve a precise blank change.
The pictures processed in this manner should be saved to a separate picture catalog on the hard drive and are now ready to be saved in an external format using an export filter in the File module.

<picsize>
This parameter can be used to specify the size of the picture to be exported. This value can only be <1; the interpolated points are then inserted farther apart, as the picture will not take up the full display area on the external system.

/ExtractRepeat
This option will extract the Repeat attributes of all points, too.
If selected, a point with e.g. 4 Repeats will be changed to 5 points at that position.
This option should be used when preparing a picture to be exported to a system that doesn´t use Repeat attributes.

Example: `Make.PreCompute 0.8`

This command edits the current picture; the picture to be exported is to be displayed a little smaller than previously.

CATalog.

The commands of the CATalog group can be used to manipulate all of the pictures of a picture catalog loaded in the picture memory. These commands only work with unprotected pictures.. The results are saved in the picture memory and must therefore be written to the hard disk in the File module for permanent storage (e.g. with **SaveCatalog**).

With all commands of the **CATalog** group it is possible to use "wildcards" (so called jokers) for stating the catalog name <catalog>, "*" for any number of characters and "?" for exactly one character.

CATalog.Attributes <catalog> <Color> <Palette> <Matrix> <Repeat> <Blank> <Vector>

This command changes the attribute values of all points of the pictures in the specified picture catalog.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

<Attribute>
One or more attributes may be specified. The attributes will be set to the specified values for all points of the pictures of the specified picture catalog. The attributes not specified will not be changed.
For further information please refer to **Do.Attributes**; this command functions in the same manner as the **CATalog.Attributes** command.

CATalog.Transform <catalog> [again] REVerse [Move->] [Size->] [Rotate->] [shear->]

Options: /cRop /Clip

This command transforms all points of the pictures in the specified picture catalog with the specified operations.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

<Transformation>
All permitted transformations can be specified as parameters. Multiple transformations are also permitted and will be executed in the sequence in which they have been listed.

[again]
This parameter repeats the command with the previously specified parameter values.

For further information on valid transformations please refer to **Do.Transform**; this command functions in the same manner as the **CATalog.Transform** command.

REVerse
This parameter reverses the last transformations by performing them again with a reversed operational sign.

/Clip
This option ensures that all points that would be outside the coordinate range as a result of one of the stated transformations are positioned at the limit of the coordinate range.
The command **Do.Transform** always checks whether points of the picture exceed the limits of the coordinate range as a result of the transformations. If this is the case, the command will be aborted and a message displayed in the status line if the **/Clip** option was not stated.

/cRop
This option deletes all points that would be outside the coordinate range as a result of the transformation.

CATalog.ALign <catalog> [All] <XLow> <XMid> <XHigh> <YLow> <YMid> <YHigh> <ZLow> <ZMid> <ZHigh>

Options: **/Individual**

This command aligns all pictures of a selected catalog in one or more directions of the coordinate range.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

[All]
This option centers all pictures of a catalog along all three axes, X,Y and Z.

[Xalign ->], [Yalign ->], [Zalign ->]
The pictures can also be aligned for each axis in the direction of the highest **<High>** or lowest **<Low>** values.
Specifying **<Mid>** for an axis centers the points along this axis.

Example: `CATalog.ALign 300-Picture XMid`
This command centers all pictures in the "300-Pictures" catalog along the X-axis.

/Individual
This parameter causes each picture of the entire catalog to be aligned individually.

CATalog.Maximize <catalog> [All] <MaximizeX> <MaximizeY> <MaximizeZ>

Options: /Individual

This command enlarges all pictures of a catalog to the limits of the coordinate range.
The enlargement can be performed for each axis individually or for all three axes together.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

[All]
This option uses a single scaling factor for all three axes, X, Y and Z; i.e. all pictures are scaled with the same factor. The original proportions of the individual pictures are thus retained.

MaximizeX, MaximizeY, MaximizeZ
These parameters permit individualized scaling for each axis. Two or three axes can be stated and the scaling will nevertheless be performed individually for each axis, i.e. each with a different factor.

Example: `CATalog.Maximize 300-Pictures MaximizeX`
This command scales all pictures of the "300-Pictures" catalog in the X direction until the picture with the greatest X value reaches the limits of the coordinate range.

/Individual
This parameter causes each picture of the entire catalog to be scaled individually.

CATalog.Interpolate <catalog> <Distance> <InsPoints> <PicPoints> <OnePoint>

Options: /Skip

Like the command **Do.Interpolate**, this command executes a linear interpolation between all points of the pictures in the catalog. The interpolation type and density is stated as in the **Do.Interpolate** command.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

/Skip
This option lets the command skip pictures that exceed the maximum number of points stated in the **<PicPoints>** parameter. Without this option the command would be aborted if a picture exceeds the maximum number of points.
If no parameter is specified, the previous parameter value will be applied.

CATalog.Spline <catalog> <Distance> <InsPoints> <PicPoints> <OnePoint>

Options: /Skip

Like the command **Do.Spline** this command executes a spline interpolation between all points of the pictures in the catalog. The interpolation type and density is stated as in the **Do.Spline** command.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

/Skip
This option lets the command skip pictures that exceed the maximum number of points stated in the **<PicPoints>** parameter. Without this option the command would be aborted if a picture exceeds the maximum number of points.
If no parameter is specified, the previous parameter value will be applied.

CATalog.AnchorCheck <catalog> <Length> <Angle> <Repeat>

This command executes an anchor point check for all of the pictures in the catalog. The command functions in the same way as the **Do.AnchorCheck** command.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

<Length> <Angle>
These parameters specify the criteria for an anchor point. See **Do.AnchorCheck**.

<Repeat>
This parameter states the repeat value that all anchor points should receive.

CATalog.BlankRepeat <catalog>

Options: /set1to0

This command searches all pictures in the catalog for points where blanking is switched on or off. The repeat value of these points is set to 1 if no higher value was specified earlier. Otherwise, the command functions in the same way as **Make.BlankRepeat**.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

/set1to0
This option sets the repeat value of all points at a blank change to 1, even if the point had a higher repeat value before. For points at which the blanking does not change, the repeat value is set to 0 if 1 is specified.

CATalog.Optimize <catalog> Blanking <Repeat> <Straight>

This command optimizes the point distribution and repeat values in all pictures of the specified catalog.

Blanking
The **Blanking** parameter deletes points located on a blanked section. The points of the blank change are retained and all points between the start and end points of the blanked section are deleted.

<Repeat>
The **<Repeat>** parameter changes the repeat value of all points that have a higher value than the specified one. The repeat value of these points is set to the specified value. The repeat values of the other points are not affected.

<Straight>
The **<Straight>** parameter deletes points in the current picture that are located close to one another and in which the connected line sections have a flat angle to one another. The value indicates the distance up to which a point will be interpreted as being on a straight line.

Example: `CATalog.Optimize Test Repeat4`
This command optimizes the distribution of points in all pictures of the catalog Test and limits their repeat value to 4.

CATalog.PreCompute <catalog> <picsize>
Options: /ExtractRepeat

This command prepares all pictures of a catalog for export to an external laser computer format. As most other laser computer systems do not work with dynamic interpolation and automatic blank changing, the pictures must be prepared for output to other systems to ensure that they will display correctly.
This command uses the advanced parameter settings of the currently selected trickfilm to insert additional interpolated points on the lines and points on the blanked lines.
The pictures processed in this manner should be saved to a separate picture catalog on the hard drive and are now ready to be saved in an external format using an export filter in the File module.

<picsize>
The **<picsize>** parameter can be used to specify the size of the picture to be exported. This value can only be <1; the interpolated points will then be inserted with a greater spacing as the picture will not take up the full display area on the external system.

/ExtractRepeat
This option will extract the Repeat information of the picture points into multiple points, e. g. a point with 4 repeats will be extracted to 5 single points at the same position.

Example: `CATalog.PreCompute 0.8`

CATalog.ColorShift <catalog> <shift>

The function "ColorShift" can be used to shift color and blanking by a specified number of points relative to X/Y for all pictures within one or more catalogs.

<catalog>
Name of the catalog. The catalog has to be in the Picture memory. It may contain wildcards ('*' or '?') if more than one catalog is to be modified.

<shift>
Number of points the color and blanking information is to be shifted.

Note:
The function "ColorShift" should only be used on pictures which have no repeats. This can for example be pictures which have been converted from another format (e.g. ILDA format) to the DSP.

CATalog.inQuiry.

The inQuiry command informs you about the requesetd details of pictures.

CATalog.inQuiry.Colors <catalog>

This command returns statistical information about the colors used in the pictures of the specified catalog. Otherwise, the command functions in the same way as **inQuiry.Colors**.
The information appears in the status line for each picture; any key can be used to browse the information. A summary for the entire picture catalog is shown at the end.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

CATalog.inQuiry.Extremes <catalog>

This command returns statistical information about the maximum coordinate values of the pictures of the specified catalog. Otherwise, the command functions in the same way as **inQuiry.Extremes**.

The information appears in the status line for each picture; any key can be used to browse the information. A summary for the entire picture catalog is shown at the end.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

CATalog.inQuiry.Linelength <catalog>

This command returns statistical information about the length of the lines in the pictures of the specified catalog. The command shows the length of the drawn line of each picture in the catalog. The number of points and share of blanked lines is also returned.
The information appears in the status line for each picture; any key can be used to browse the information. A summary for the entire picture catalog is shown at the end.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

CATalog.inQuiry.Points <catalog>

This command returns statistical information about the number of points in the pictures of the specified catalog.
The information appears in the status line for each picture; any key can be used to browse the information. A summary for the entire picture catalog is shown at the end.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

CATalog.inQuiry.PreCompute <catalog> <picsize>

This command returns statistical information about the number of points that the pictures of the specified catalog would contain after an executed PreCompute command. Otherwise, the command functions in the same way as **inQuiry.PreCompute.**
The information appears in the status line for each picture; any key can be used to browse the information. A summary for the entire picture catalog is shown at the end.

<catalog>
This parameter states the name of the picture catalog in the picture memory that contains the pictures to be edited.

<picsize>
This parameter functions in the manner described under **Make.PreCompute**. As the parameter affects the number of points, it must also be stated in the **CATalog.inQuiry.PreCompute** command.

inQuiry.

The **inQuiry** command group returns statistical information about the current picture. The information is displayed in the status line.

inQuiry.Colors

This command shows the colors used in the current picture. The colors are sorted by palette; the number of the colors used is displayed. Colors not used are marked
by a dash "-".

inQuiry.Extremes

This command returns the extreme values of the current picture. The points with the largest and smallest coordinate values are displayed for each axis. XLo is the point with the lowest X value; XHi is the point with the highest X value. The extreme points of the other axes are displayed in the same way.
The point number and coordinate values of the point are displayed as information.

inQuiry.Linelength <number> <number>

This command returns the drawn line length between two points of the current picture.
It also shows which portions are visible, which portions are blanked, and the percentage of the total distance for these portions.

<number>
The **<number>** (Point No.1) and **<number>**(Point No.2) parameters specify the two points between which the drawn line should be measured.

inQuiry.Distance <number> <number>

This command returns the distance between two points of the current picture.

<number>
The **<number>** (Point No.1) and **<number>**(Point No.2) parameters specify the two points between which the distance should be measured.

inQuiry.Angle <number> <number> <number>

This command displays the angle between three points of the current picture. The degree values displayed refer to the 3D angle, the angle as a projection in the XY plane, the angle in the XZ plane and in the ZY plane.

<number>
Three points must be specified for this command. An imaginary line is drawn between Point No. 1 and Point No. 2, as well as between Point No. 2 and Point No. 3. The angle between these lines is then measured.

inQuiry.PreCompute <picsize>

This command returns statistical information about the number of points that the current picture would contain after an executed **PreCompute** command.

<picsize>
This parameter functions in the manner described under **Make.PreCompute**. As the parameter affects the number of points, it must also be stated in the **inQuiry.PreCompute** command.

Tools.

Tools.Edit

See "Edit".

Tools.Insert

See "Insert".

Tools.InsertPars

This command opens the window with the parameters for **insert** mode. The settings refer to continuous input mode with the left or middle mouse button being held. All modes can be combined as required.

Insert by time

When the **Insert by time** input mode is activated using the associated switch, points are set at regular intervals along the path of the mouse movement while the mouse button is being held.

Dots/s

The **Dots/s** field determines the number of points to be set per second.

Dots/s by Fader #6

The **Dots/s by Fader#6** switch in the parameter window or the button over the fader control #6 can be used to set the **Dots/s** value with this control during input.

Minimum Distance

The value in the **Minimum Distance** field determines the minimum distance between two points before further points are set. This parameter prevents an excessive density of points at locations at which the mouse is moved slowly.

Repeat by time

The **Repeat by time** switch activates a mode in which the value of the repeat attribute of the previous point increases in increments up to a set limit as long as the left mouse button is being held and a fixed distance to the point has not been exceeded. The function with which the repeat value of a point is increased by the value 1 by a further mouse click remains available.

Repeat Delay

The **Repeat Delay** value is a delay time in seconds after which the repeat value of the preceding point should be increased incrementally.

Repeats/s

The value in the **Repeats/s** field determines how quickly the repeat value should be increased.

Repeats/s by Fader #5

The **Repeats/s by Fader#5** switch in the parameter window or the button over the fader control #5 can be used to set the **Repeats/s** value with this control during input.

Repeat Distance

The value in the **Repeat Distance** field determines the distance within which the cross hairs must be located to increase the repeat value for the affected point. The stated distance refers to the screen output and is therefore independent of the scaling factor.

Repeat Maximum

The **Repeat Maximum** value determines the maximum to which the repeat value can be increased while the mouse button is held.

Insert by distance

When the **Insert by distance** mode is activated using the associated switch, points are set at a regular distance along the path of the mouse movement while the left or middle mouse button is being held.

Distance

The value in the **Distance** field determines the distance to the previous point after which a new point should be set.

Distance by Fader #4

The **Distance by Fader#4** switch in the parameter window or the button over the fader control #4 can be used to set the **Distance** value with this control during input.

Limitations ⇨ If the number of points to be set per second is not very high due to the specified values, it is possible that this frequency may not be reached. This effect can be observed especially when points are being set between already existing points. To counter this effect, disable the laser window on the screen. This saves processing time and permits points to be set with a higher frequency.

A double-click on the **|<u>Insert</u>|** button also opens the Insert Parameter window.

Undo.

Undo.Undo

Compare "Undo".

Undo.Redo

Compare "Redo".

Undo.INIT

This command clears all saved **Undo** and **Redo** operations.

Hotkeys in Piced

In addition there are hotkeys to make working with the picture editor easier, see the following list.

Note: Most hotkeys only work as long as the menu is deactivated!

Select the Active Point

Pos1	activate the first point for editing
End	activate the last point for editing
PageUp	activate the previous point
PageDown	activate the following point
#	activate a specific point

Edit the Active Point

CursorUp	move point upwards (only as long as mouse is cross-hair)
CursorDown	move point downwards (only as long as mouse is cross-hair)
CursorLeft	move point left (only as long as mouse is cross-hair)
CursorRight	move point right (only as long as mouse is cross-hair)
Delete	delete active point and make following point active
Backspace	delete active point and make previous point active.
Ctrl-Insert	insert point at current position
Return	exchanges the mouse for the cross-hair or, if this is already done, sets down the active point and makes the following point active
Shift-Return	set down point and insert point at the same position
Escape	abort editing or release point (according to context)
v / V	turn vector attribute on or off
b / B	turn blank attribute on or off
m / M	increase / decrease matrix by 1
p / P	increase / decrease palette by 1
c / C	increase / decrease color by 1
r / R	increase / decrease repeat by 1
0 ... 9	sets color resp. repeat to a fixed value between 0 and 9. Whether color or repeat is used depends on which icon is displayed as pressed. In 3D-display the number keys are used for positioning.
o	turns attraction mode on or off. In attraction mode the current point automatically jumps to the position of another point as soon as it is moved close enough to that point.

Space	toggles between normal point and anchor point (with more repeats)
^	turns the automatic detection of corners on or off (Anchorcheck)

Select Points resp. Edit Selected Points

u	select resp. deselect the active point
Ctrl-a	select resp. deselect all points
Ctrl-v	select point by vector resp. change vector selected point
Ctrl-b	select point by blank resp. change blank selected point
Ctrl-m	select point by matrix resp. change matrix selected point
Ctrl-p	select point by palette resp. change palette selected point
Ctrl-c	select point by color resp. change color selected point
Ctrl-r	select point by repeat resp. change repeat selected point
Ctrl-d	transform selected points (Drag)

Change Picture View

a	activate X-Z view ("Top X-Z")
y	activate X-Y view ("Front X-Y")
x	activate Z-Y view ("Side Z-Y")
s	activate 3D-view ("3-D")
z	toggle between single view and all 4 windows ("Zoom")
t	increase picture detail by factor 1.4 ("Tele")
w	decrease picture detail by factor 1.4 ("Wide")
Ctrl-t	define picture detail using the mouse
°	center the picture display relative to the cross-hair
ScrollLock	turn automatic scrolling when leaving the visible area in zoomed state on and off

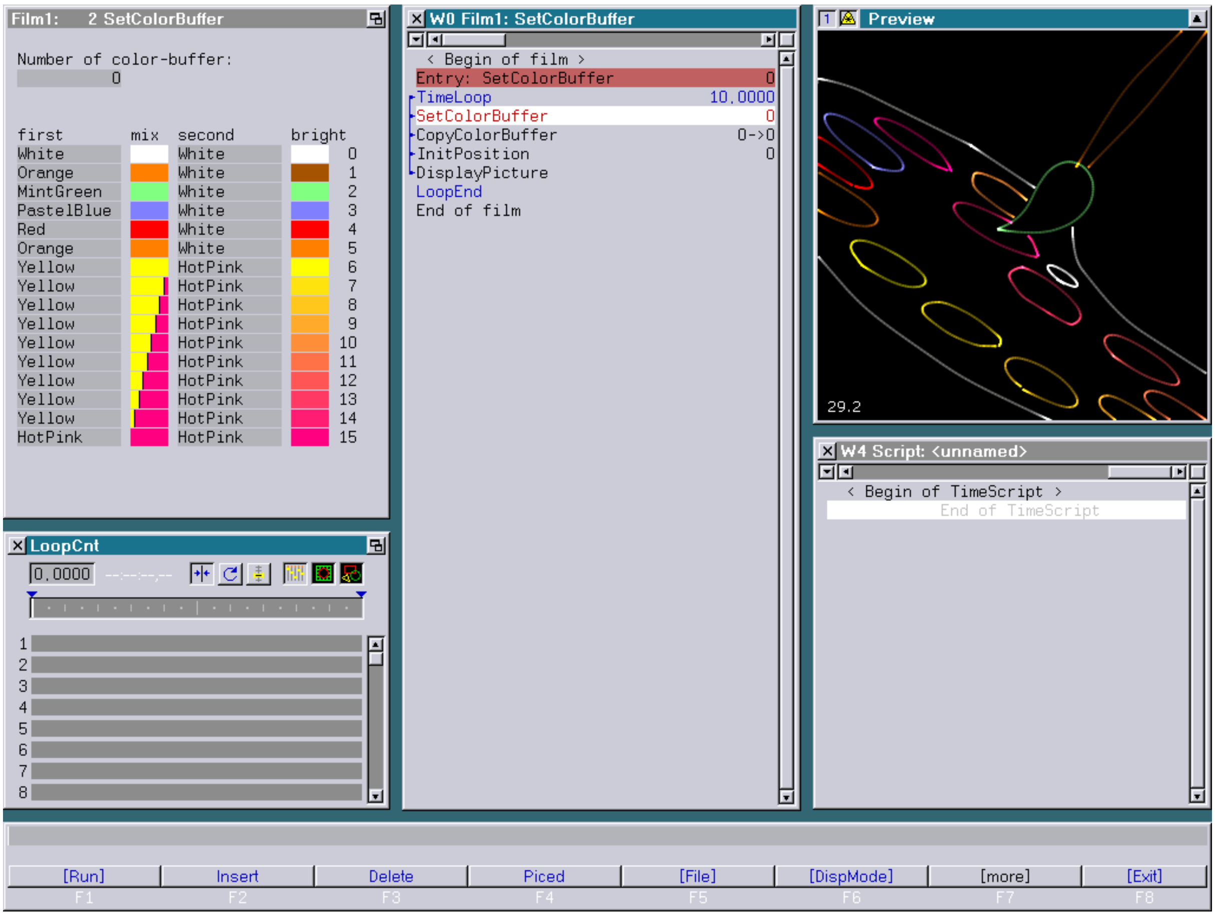
Picture Options

Ctrl-n resp. n	discard current picture and begin editing a new one
Ctrl-l resp. l	load picture
Ctrl-s	save picture
Ctrl-e resp. e	save picture and then leave picture editor
Ctrl-q resp. q	leave picture editor without saving picture
Ctrl-z resp. Ctrl-Backspace	undo one editing step
Ctrl-y resp. Ctrl-Shift-Backspace	redo a step which was previously undone
Insert	toggle between edit- and insert mode
< / >	load previous / following picture from same catalog (in alphabetical order)
(/)	load previous / following brown overlay picture
[/]	load previous / following green overlay picture
f	turn grid on or off
g	refresh picture

Meaning of the Function Keys

F1...F8	calls up the corresponding entry of the main menu (PIcture, View, Point, Set, Do, Make etc.)
F9/F10	allows selecting the commands last entered

The Trickfilm Editor "[Edit]"



Number of color-buffer:
0

first	mix	second	bright	
White	⬜	White	⬜	0
Orange	🟧	White	🟫	1
MintGreen	🟩	White	🟩	2
PastelBlue	🟦	White	🟦	3
Red	🟥	White	🟥	4
Orange	🟧	White	🟧	5
Yellow	🟨	HotPink	🟨	6
Yellow	🟨	HotPink	🟨	7
Yellow	🟨	HotPink	🟨	8
Yellow	🟨	HotPink	🟧	9
Yellow	🟨	HotPink	🟧	10
Yellow	🟨	HotPink	🟧	11
Yellow	🟨	HotPink	🟥	12
Yellow	🟨	HotPink	🟥	13
Yellow	🟨	HotPink	🟥	14
HotPink	🟥	HotPink	🟥	15

```
 < Begin of film >
Entry: SetColorBuffer                 0
TimeLoop                        10,0000
SetColorBuffer                        0
CopyColorBuffer                    0->0
InitPosition                          0
DisplayPicture
LoopEnd
End of film
```

```
 < Begin of TimeScript >
       End of TimeScript
```

Contents

1. Trickfilm Editor

1.0 Short Introduction

The *trickfilm editor* is the central block of functions which you will use to program your laser show. In this part of the program the pictures which you have previously created with the *Picture Editor* will learn to 'walk', so to speak.

The **Lasergraph DSP** can control up to four films simultaneously. Each film controls one *projector* (x/y scanner pair, blanking and RGB color changer resp. x/y scanner pair and crystal). This means you can only run one film per *projector*. It is however possible to show several films consecutively or alternately with one *projector*.

If your **Lasergraph DSP** has more than one DSP-card (Compact Dual or Workstation), you can run a corresponding number of films (up to four films) simultaneously with one *timescript*.

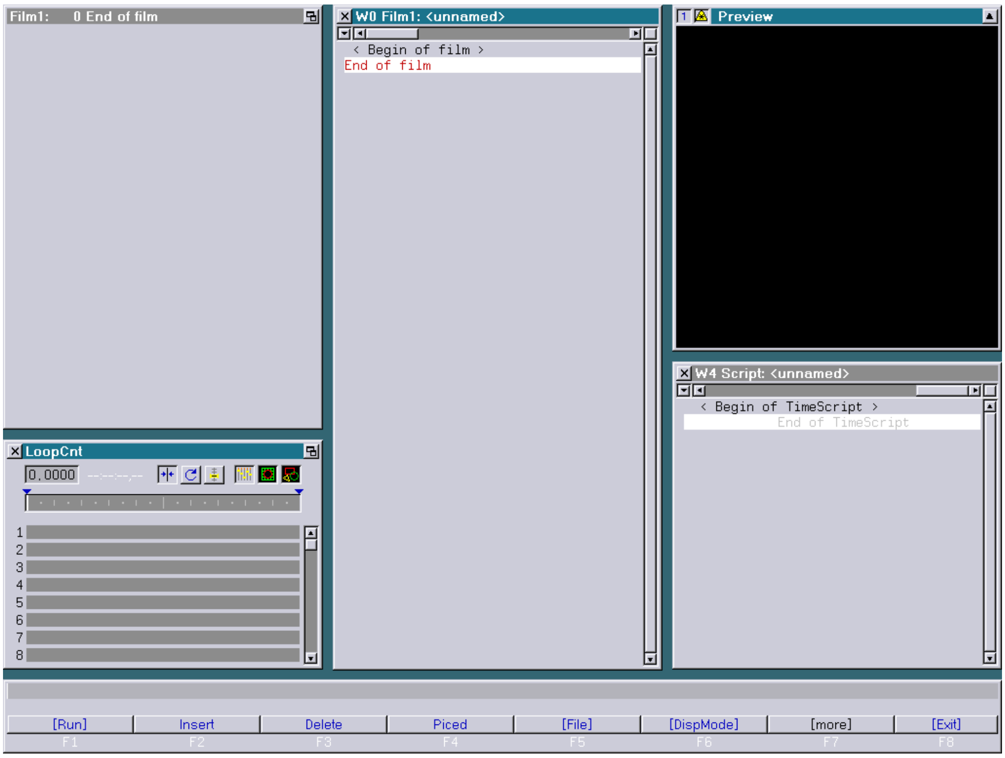
Even if your **Lasergraph DSP** only controls one projector it is very helpful for different reasons that you can load and edit four films simultaneously.

If for instance you are programming your first own **Lasergraph DSP** shows you simply additionally load a demo film as a 'cheat sheet'. This will save you time and errors. Of course you can also copy previously programmed sequences from other films into your current film - also very time saving! (Should not be exaggerated however, it may quickly lead to confusion in the audience.)

1.1 Structure of the Windows in the Trickfilm Editor

[Edit]
F2

You reach the *trickfilm editor* either by clicking on [Edit] in the *menu bar* at the bottom with the left mouse button or by pressing the function key F2. The following menu appears:



The Menu in [Edit]

Of everything you see here, at least the *menu bar* at the bottom and the small black *laser preview window* should be familiar.

Display Window

Let us presume that you have not loaded a film yet. You then see four more or less empty areas. The largest part of the screen display on the left side is the *display window* where the *parameters* of the respective trickfilm line are shown and edited. You can see in the *headline* if you are editing trickfilm 1, 2, 3, 4 or the *timescript* and which line you are in. By clicking with the left mouse button on the little button on the right you can enlarge this area and reduce the size again.

The column to the right of this area shows a section of the program lines from one of the four possible films or the *timescript*.

Program Window
W0

Let us just call them *program windows*. When you activate the program window by clicking on it with the left mouse button the headline is highlighted by its color.

In the headline you read:

`W0` (for Window 0) `Film 1: <unnamed>`

and to the right of that:

`W4` (for Window 4) `Script: <unnamed>`

As soon as you have loaded a film, its name will appear behind the film number.

When you create a new film, the corresponding name will appear after the first saving.

You can read `TimeScript` in the partly visible window at the bottom to the right. If you click on this window with the left or middle mouse button it is activated and brought to the foreground.

In the headline highlighted by color you now read:

`W4 Script: <unnamed>`

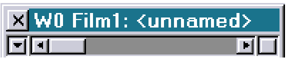
You can have each of the *program windows* display one of the four films or the *timescript*.

The *Scroll Bars* in `[Edit]`

For this use the horizontal *scroll bar* right beneath the headline.

The *scroll bar* looks like an indented rail in which the raised *scroll box* can be slid back and forth.

The Horizontal *Scroll Bar*



W0 Film1: <unnamed>

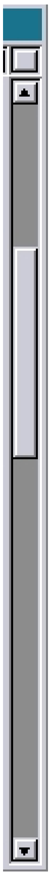
↑ *Scroll Box*

Slide the *scroll box* to the left or right with pressed left mouse button until the desired film number is displayed above. Or click on one of the little buttons with the black triangle to the left and right of the *scroll bar* with the left mouse button. Clicking or sliding to the right counts the films up from 1 - 4 to the timescript, left back to film 1. You now see how the film number not only changes above the *program windows* but also above our *display window*.

Since trickfilms are usually longer than can be displayed in the limited space available in the *program window*, we only see a section of the complete film.

The Vertical *Scroll Bar* of the *Program Windows*

However, we can still look at the film from start to finish with the help of the vertical *scroll bar* which is located on the right side of each program window.



The vertical *scroll bar* generally looks just like the horizontal one. However, the size of the *scroll box* varies according to the percentage of the film which is visible in the window. I.e. the longer the trickfilm the smaller the display of the *scroll box*, providing more free rail space for *scrolling*. If the scroll box fills the whole bar we know that the whole film is displayed in the window.

The *scroll box* also informs you of the current position within the film. When the *box* is at the top of the *scroll bar* you see the beginning of the film, if it is in the middle of the bar you see a section from the middle of the film and so forth.

You can now scroll through the film in different ways.

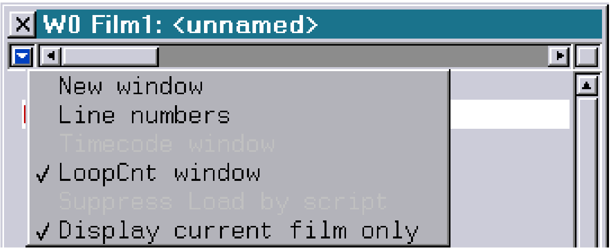
You grab the *scroll box* by clicking on it with the left mouse button and slide it along the *scroll bar* with pressed mouse button until you reach the desired position in the film.

If you click with the left or middle mouse button on the buttons with the triangles which are located at the top and bottom ends of the *scroll bar*, you scroll the film line by line. If you keep the mouse button pressed, the film is scrolled automatically.

You can also scroll page by page, where the section visible in the window is equivalent to one page. For this you click into the empty bar above or below the *scroll box*, depending on whether you want to scroll forwards or backwards. Keeping the mouse button pressed again results in automatic scrolling.

Clicking into the square next to the *scroll box*, the following *menu* opens with different options:

Menu



`New Window`

Click on `New Window` with the left mouse button. A *new program window* will open, namely `W2`. It is positioned above `W0`, but the displayed section is much shorter. If you now click anywhere on `W0`, for instance to also open `W3`, `W2` seems to disappear again. In fact it was only moved behind `W0` since you carried out an action in `W0`, i.e. the window was activated and thereby moved to the foreground.

Usually you click on windows in the background once with the left mouse button to activate them. However, this requires at least a small section of the window to be visible.

Finding Hidden Windows

Even if the window is completely covered there are several options for finding it:
If, as in this case, you know exactly behind which window the window you are looking for is, the simplest method is to click on the *headline* of the topmost window with the middle mouse button. This moves the window to the background, the previously covered window reappears in the foreground.

Switching Back And Forth Between Windows With Key Combinations

Option 2: Changing windows with key combination.
If you keep the *CTRL key* pressed you can switch back and forth between all opened windows with the *TAB key*. The currently activated window gets a bright frame which you can also see if the window is hidden behind another one. As soon as you release the *CTRL key* the window you have selected is shifted to the foreground. The key combination SHIFT CTRL TAB inverts the direction of switching.

Moving Windows

Option 3: Shifting windows in the foreground.
You can also move all windows in the *trickfilm editor* wherever you want to.
Click on the *headline* of `W0` with the left mouse button, keep the mouse button pressed and move the window anywhere. As soon as you release the mouse button the action is carried out and `W2` will appear behind the old position of `W0`.

In this way you can place all windows wherever it makes sense for you.

Changing Window Sizes

You can also increase or decrease the size of the windows. Move the mouse pointer to the edge of the window until the usual pointer changes into a double arrow. Now again keep the left mouse button pressed and move the mouse in one of the arrow directions. The bright rectangle shows you the resulting size. When you release the mouse button the window is longer or shorter, wider or narrower - give it a try.

☞

If you want to work with additional *program windows* it is advisable to place them so they do not cover each other completely. It can be quite irritating if you continuously have to look for covered windows.

Windows With Priority

However, some windows have a higher priority. They can be moved to the back- and foreground among each other in the way described, but they cannot be covered by the *display window* or *program windows*.
Among these are the `File` *window*, the *laser preview window* and the *LoopCount* window (`LoopCnt`).

Menu and *Input Line*

The *menu* and *input line* at the bottom has an absolutely exceptional status. The window can be moved, however it always keeps its size and is always positioned in the foreground.

`LoopCnt` *Window*

The *LoopCount window* is usually open while a film is edited. It always stays in the foreground, just like *the laser preview window*, even if the *display* or *program window* behind it is activated.

⧉

The window size can be toggled by clicking on the *button* on the right side of the *headline*.
You can also assign both window sizes different positions which they will assume again automatically with each switching of sizes.

✕

If you want to turn off the *LoopCount window* completely, click on the left *button* in the headline. When you need it again, either enter `loopcnt` and confirm with *return* or open the *menu* which is also used to open additional *program windows* and click on `LoopCnt window`. You can also hide this window with this menu.

▲

The *laser preview window* cannot be closed, but you can switch sizes and positions by simply clicking into the preview window or by doubleclicking onto the headline.
With a mouse click on the right button in the *headline* it takes up the whole screen. To switch it back simply click again with the left mouse button or hit any key of the keyboard.

If you do not like the default settings of the windows just place everything where you want it.

Save New Window Positions

When you have found a configuration which is most suitable to you and your working method you can save it so that you do not have to start over each time you restart the computer.
For this simply write

`SaveWinPos`

```
SaveWinPos
```

The writing then appears in the *input line* and you confirm it with the *return key*.

As confirmation you then read in the *input line*:

```
Current window positions saved as default.
```

The window positions are stored independently for each desktop resolution in separate files in the following directory:

```
C:\System\Configuration\WinPos\
```

The name of these files results from the desktop resolution:

```
<width>x<height>.DCfg
```

1.2 Structure of a Film

Now you can perfectly handle the windows in the *trickfilm editor*. Next, you should know how a film is structured, why this is so, and what the connection to the *timecode* is.

For this you should load one of the finished films that are delivered with every **Lasergraph DSP**.

Loading a BOOT Show

Press the function key `F5` or click on the button `[File]` in the *menu bar* at the bottom. In `DRIVE_C` you click on `Shows`, choose a directory e.g. `Beams` or `Graphics,` then on any of the shows (extension: BOOT). If the structure is different on your hard disk, please choose the subdirectories until you find some show files.

If you only click once (double clicking will load the show without confirming) you will be asked in the *input line*:

`ReBoot` <name of the selected show>

Confirm with the `F1` key, by clicking on `[ok]` or with the *return* key.

☞ By double clicking on the desired show you can avoid the requester and the show is loaded immediately.

Now you leave the *File level* either by clicking the right mouse button, clicking on `[Exit]` at the bottom to the right with the left mouse button or by pressing the `F8` key.

You have now loaded a so-called *Boot*. All components necessary for playing back a *timecode* controlled show are now present.

They are:
One or more trickfilms, the graphics required for the film, the timing information, the *screen parameters* and - if necessary - *beam table* and the *keyboard-*, *map-* and *bank configurations*.

This definitely has advantages, and you should save each finished film as a *Boot*. How this is done and what has to be taken into account will be discussed later.

If you scroll through the loaded film slowly you will see that it is divided into *blocks*. The structure of these *blocks* is predefined. A *block* always begins with an *Entry* and ends at the next *Entry*.

The order of the film commands is in no way random but follows a strict hierarchy.

The Command Hierarchy:

- When you *initialize* a command (read: enter), the computer will insist on carrying it out until you *reinitialize* it.

Example: InitPosition: if you define a position, this postion will be used as long a new postion is defined.

- All commands will be carried out by the computer **one by one in sequence**.

Therefore it makes sense to put all commands which do not yet lead to a *picture output* but have an influence on the output (e.g. colour/position) **before** the *picture output*.

It is equally logical to determine a position first before defining a movement.

This results in the following obligatory order:

```
TimeLoop
InitPosition
(ChangePosition)
DisplayPicture
LoopEnd
```

- Everything within a *loop* is called up by the computer again and again as long as the *loop* is running.

Therefore all commands which only have to be provided or adapted once are set outside the *Timeloop*. This concerns for instance *entries*, the *GoSub* command, *master positions*, the provision of a graphic in the *buffer* and the setting of *color buffers*.

Let us now look at the loaded film.

Each block begins with the command `Entry`.
The *command line* `Entry` is highlighted by its color, making the film's structure more evident by marking each new sequence.

`Entry`
→ Chapter 3.2.1

The `Entries` are the most important structural element in a film. They are numbered from 0 to max. 999 and allow you, by calling up the corresponding *entry*, to call up and execute any trickfilm block in any order or to interrupt it with another block.

The next command is usually a `GoSub` with a *subroutine* number beginning from 1000.
A subroutine is a special Entry, see Chapter 3.2.3 for details.

`GoSub`
→ Chapter 3.2.6

The next 'obligatory' command is the `TimeLoop` with which you determine how long the programmed sequence is supposed to run.

`TimeLoop`
→ Chapter 3.3.2

At the end of the *block* the `TimeLoop` always has to be *deinitialized* with the command `LoopEnd`.

Within the time loop you will always find an `InitPosition`, possibly a `ChangePosition` and always a command for the *picture output* - in exactly this order.

Before each *entry* you see one or more different-colored lines with various texts.
The corresponding film command is `Comment`. This command has no effect on the progression of the film, it only serves to structure the film more clearly.

`Comment`
→ Chapter 3.1

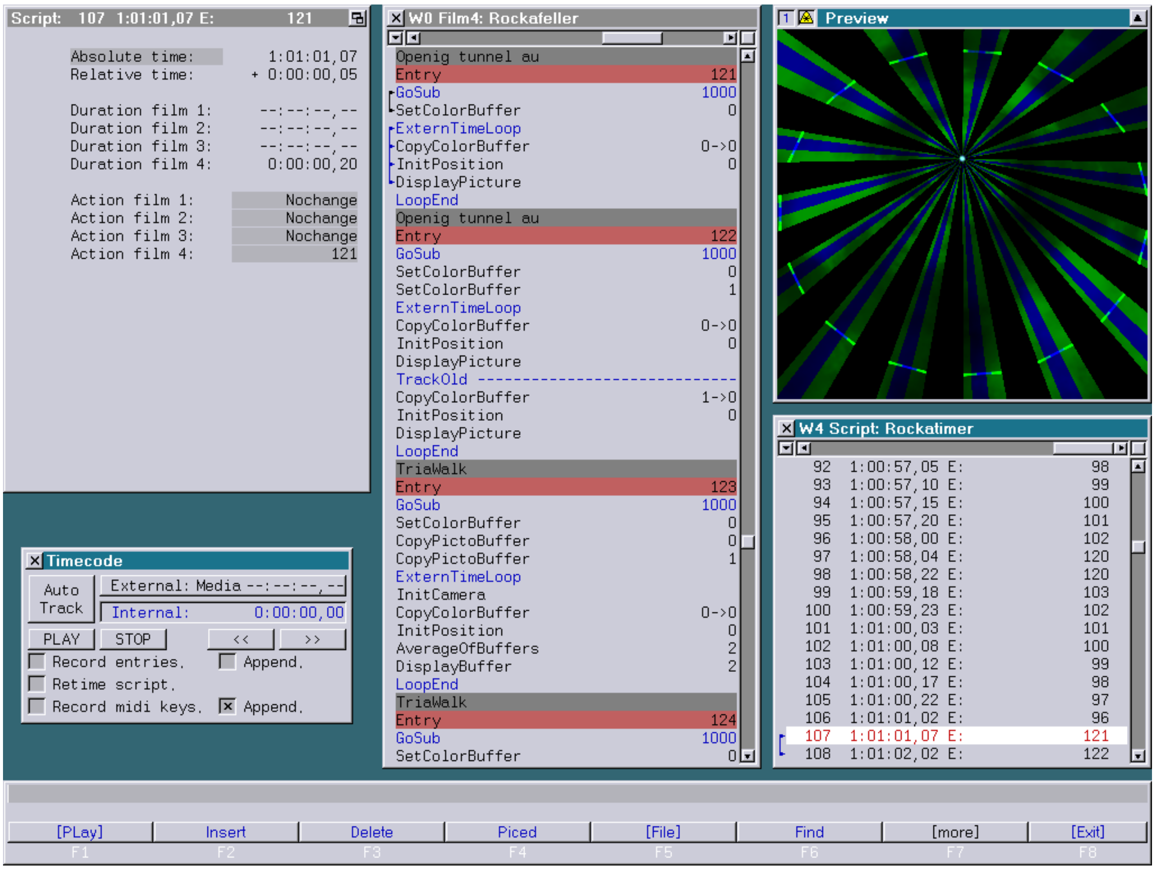
If you search for a specific sequence in a film it is very helpful if there is a short description of the corresponding effect in front of every *block*.

1.3 Connection to the Time Script

In the TimeScript you determine
- which film sequence
- from which film

is to run
- when
- for how long.

In this context we will only look at the connecting elements between trickfilm and Time-Script.



```
Script:  107  1:01:01,07 E:          121

Absolute time:          1:01:01,07
Relative time:        + 0:00:00,05

Duration film 1:      --:--:--,--
Duration film 2:      --:--:--,--
Duration film 3:      --:--:--,--
Duration film 4:        0:00:00,20

Action film 1:            Nochange
Action film 2:            Nochange
Action film 3:            Nochange
Action film 4:                 121
```

```
W0 Film4: Rockafeller
Openig tunnel au
Entry                          121
GoSub                         1000
SetColorBuffer                   0
ExternTimeLoop
CopyColorBuffer               0->0
InitPosition                     0
DisplayPicture
LoopEnd
Openig tunnel au
Entry                          122
GoSub                         1000
SetColorBuffer                   0
SetColorBuffer                   1
ExternTimeLoop
CopyColorBuffer               0->0
InitPosition                     0
DisplayPicture
TrackOld ---------------------------
CopyColorBuffer               1->0
InitPosition                     0
DisplayPicture
LoopEnd
TriaWalk
Entry                          123
GoSub                         1000
SetColorBuffer                   0
CopyPictoBuffer                  0
CopyPictoBuffer                  1
ExternTimeLoop
InitCamera
CopyColorBuffer               0->0
InitPosition                     0
AverageOfBuffers                 2
DisplayBuffer                    2
LoopEnd
TriaWalk
Entry                          124
GoSub                         1000
SetColorBuffer                   0
```

```
W4 Script: Rockatimer
 92  1:00:57,05 E:        98
 93  1:00:57,10 E:        99
 94  1:00:57,15 E:       100
 95  1:00:57,20 E:       101
 96  1:00:58,00 E:       102
 97  1:00:58,04 E:       120
 98  1:00:58,22 E:       120
 99  1:00:59,18 E:       103
100  1:00:59,23 E:       102
101  1:01:00,03 E:       101
102  1:01:00,08 E:       100
103  1:01:00,12 E:        99
104  1:01:00,17 E:        98
105  1:01:00,22 E:        97
106  1:01:01,02 E:        96
107  1:01:01,07 E:       121
108  1:01:02,02 E:       122
```

```
Timecode
Auto Track   External: Media --:--:--,--
             Internal:       0:00:00,00
□ Record entries.     □ Append.
□ Retime script.
□ Record midi keys.   ☒ Append.
```

The connecting link between the Time-Script and the trickfilm are the film- and *entry* numbers.
In the Time-Script you enter the *entry* numbers of the desired film sequence. While you are working in the TimeScript, the trickfilm automatically jumps to the spot with the specified *entry*.

Another connection exists when you work with the command ExternTimeLoop. The sequence will then run in the period of time until the next *entry* is called up in the timescript or a termination is programmed.

2. Mode of Operation

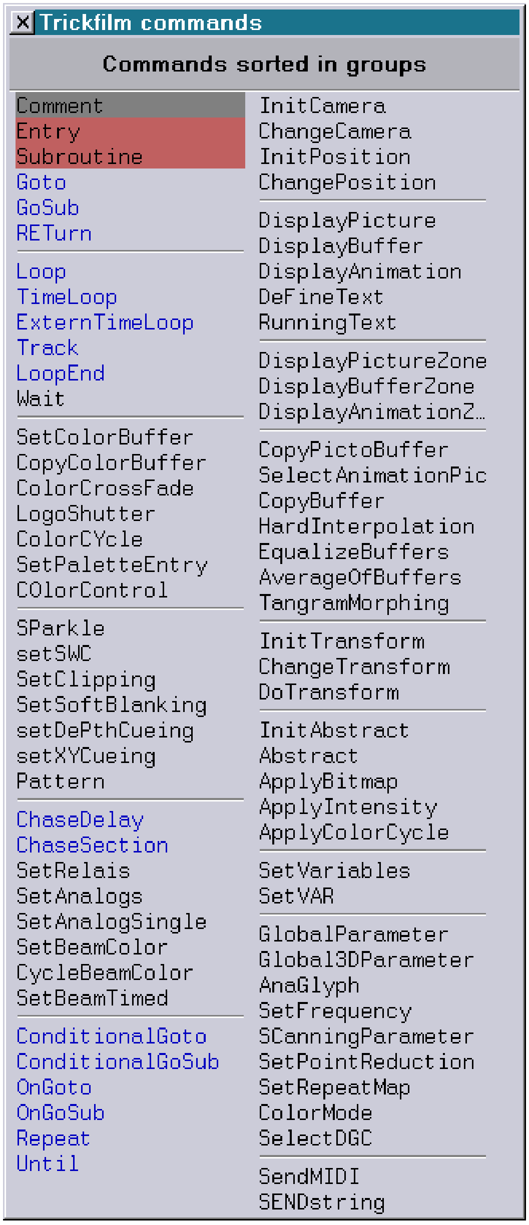
In this chapter you will learn how to set and edit commands and how to copy, paste or delete individual lines or complete blocks.

2.1 Setting Commands

In order to set commands you have to open a new window. Let us call it the *command window*.
For this click on `Insert` in the *menu bar* at the bottom or press the function key F2.

Command Window

The following window will open:



Trickfilm commands

Commands sorted in groups

Comment
Entry
Subroutine
Goto
GoSub
RETurn

Loop
TimeLoop
ExternTimeLoop
Track
LoopEnd
Wait

SetColorBuffer
CopyColorBuffer
ColorCrossFade
LogoShutter
ColorCYcle
SetPaletteEntry
COlorControl

SParkle
setSWC
SetClipping
SetSoftBlanking
setDePthCueing
setXYCueing
Pattern

ChaseDelay
ChaseSection
SetRelais
SetAnalogs
SetAnalogSingle
SetBeamColor
CycleBeamColor
SetBeamTimed

ConditionalGoto
ConditionalGoSub
OnGoto
OnGoSub
Repeat
Until

InitCamera
ChangeCamera
InitPosition
ChangePosition

DisplayPicture
DisplayBuffer
DisplayAnimation
DeFineText
RunningText

DisplayPictureZone
DisplayBufferZone
DisplayAnimationZ…

CopyPictoBuffer
SelectAnimationPic
CopyBuffer
HardInterpolation
EqualizeBuffers
AverageOfBuffers
TangramMorphing

InitTransform
ChangeTransform
DoTransform

InitAbstract
Abstract
ApplyBitmap
ApplyIntensity
ApplyColorCycle

SetVariables
SetVAR

GlobalParameter
Global3DParameter
AnaGlyph
SetFrequency
SCanningParameter
SetPointReduction
SetRepeatMap
ColorMode
SelectDGC

SendMIDI
SENDstring

Absolute Jump →

Time Control →

Color Control →

Display Control →

External Hardware Settings →

Conditional Jump →

← *Viewpoint Definitions*

← *Picture Output*

← *Picture Output at Projection Zones*

← *Buffer Operations*

← *Transformations*

← *Abstracts and Bitmaps*

← *Variables*

← *Global Settings*

← *Send MIDI/String*

If the window isn´t big enough to show all commands, there is a scroll bar on the right which can be used to scroll through all remaining commands.

Commands Sorted Alphabetically

If for some reason you do not like this arrangement according to command groups you can switch to alphabetical order. Just click anywhere in the window with the middle or left mouse button.

→ `Sipro`

When you click again the `Sipro` commands will be displayed, clicking a third time will get you back to the group order.

Setting Commands By Mouse

To set a command in your trickfilm point the mouse at the desired *command line* which will then be highlighted with a different color and click once with the left mouse button.

Setting Commands By Keyboard

You can also set each command with a key combination.
For this enter the capital letters of the respective command, e.g.

`e` for `Entry`
`ip` for `InitPosition`, etc.

If you still have the `Boot` loaded, in `W0` display a film which is still empty and enter a '*standard block*' consisting of:

```
Entry
TimeLoop
InitPosition
DisplayPicture
LoopEnd
```

2.2 Editing Commands

Before you can edit commands you have to leave the `Insert` level. The easiest way is clicking with the right mouse button. You can also press F8 or press the ESC key.
Now the last line of your film `End of film` is highlighted with a different color.

To edit the *command line* click on the desired line or use *the cursor key* ↑ to get to the respective line. The selected *command line* is now marked and the *display window* to the left shows the corresponding specifications and coordinates which you can influence.

Input Fields

Depending on the command you will see a different number of *input fields* in the *display window*. You will recognize *input fields* by the way their color changes when you point the mouse at them. When you click on one of the fields it lights up.

In these fields you can change colors or times, change or add *coordinates*, enter *curves* or *variables* or specify names.

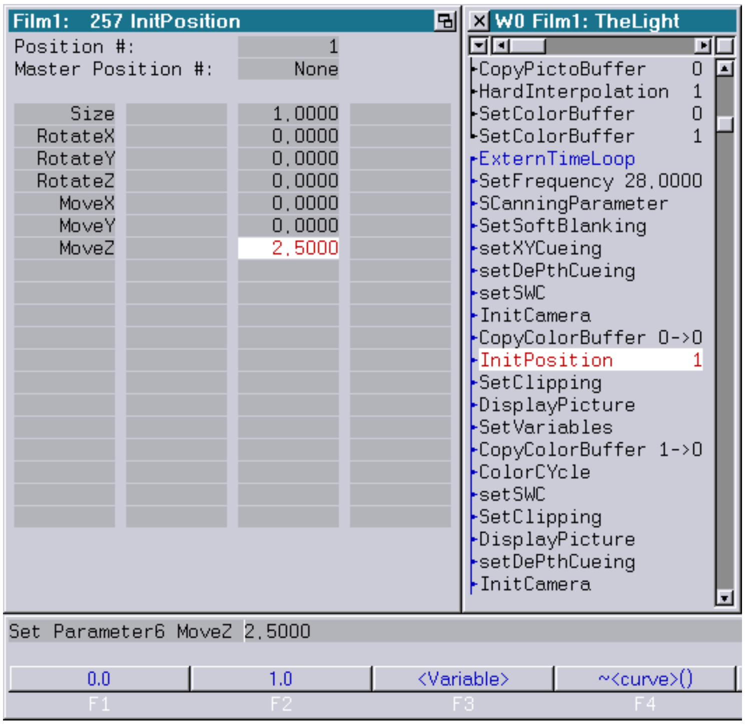
Commands serving the *picture output* for instance have *input fields* which - when clicked on - will open a `File` *selection window* where you can select catalogs and graphics.

The different input methods will be described in the following chapter.

2.3 Input Methods

Input Line

The so called *input line* is located above the *menu bar* at the bottom and the *status line*.
As soon as an *input field* is selected the current value -if it exists - is copied to the *input line*.
Otherwise you can see here what entry is expected of you.
Behind this you see a lighted *cursor*.



```
Film1:   257 InitPosition
Position #:                 1
Master Position #:       None

      Size             1,0000
   RotateX             0,0000
   RotateY             0,0000
   RotateZ             0,0000
     MoveX             0,0000
     MoveY             0,0000
     MoveZ             2,5000
```

```
W0 Film1: TheLight
CopyPictoBuffer      0
HardInterpolation    1
SetColorBuffer       0
SetColorBuffer       1
ExternTimeLoop
SetFrequency 28,0000
SCanningParameter
SetSoftBlanking
setXYCueing
setDePthCueing
setSWC
InitCamera
CopyColorBuffer 0->0
InitPosition         1
SetClipping
DisplayPicture
SetVariables
CopyColorBuffer 1->0
ColorCYcle
setSWC
SetClipping
DisplayPicture
setDePthCueing
InitCamera
```

Input Line → `Set Parameter6 MoveZ 2.5000`

Status Line →

Menu Bar →

0.0	1.0	<Variable>	~<curve>()
F1	F2	F3	F4

Cursor

The *cursor* is a bright vertical line which shows you that your next keyboard input will be displayed here. With every letter you enter this mark moves one character to the right.

Just as we have seen with the *program windows* and the *command window*, only a part of the *input line* is displayed on the screen.

If your entry is longer than the section displayed the text will be moved one step to the left with each new entry.

Scrolling In The *Input Line*

The *input line* does not have a *scroll bar* because it happens very rarely that the displayed length of the *input line* is not sufficient. However, you can scroll through the line with the *cursor*.

Cursor Keys ← →

In order to move through the entry left or right character by character press the *cursor keys* ←/→. If you keep one of the keys pressed the *cursor* will continuously move through the text in the corresponding direction.

You can also position the *cursor* at any point of the *input line* by clicking on the desired position with the left mouse button.

A click with the right mouse button will move the *cursor* to the left by one command / option.

If you simply want to get to the beginning of the line by the fastest way press the Home key.

You reach the end of your entry with the End key.

There are several ways of editing your entries:

Delete

If you realize that you have just typed a wrong character simply press the Backspace key and the character to the left of the *cursor* will be deleted. At the same time the *cursor* moves one position to the left. Then just continue writing.

You can also delete characters anywhere in the *input line*.
Place the *cursor* at the position you want to delete and press the Delete key. The character to the right of the *cursor* will be deleted. If you hold down the Shift key while pressing the Delete key you will delete everything to the right of the *cursor*.

You can select one or more characters by moving the *cursor* over these characters while keeping the left mouse button pressed or by moving the cursor with the *cursor keys* ←/→ while pressing the Shift key. The selected part will then appear with a different background color. If you now press Delete the selected characters will be deleted.

Pressing the Backspace key will delete the character to the left of the *cursor*, even in the middle of the text. The *cursor* will move left for the corresponding number of positions together with the following text.

The most radical method is pressing the Esc key. This will delete the whole line.

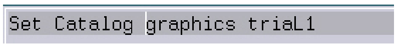
Insert

In order to insert something into the *input line* place the *cursor* to the right of the place where you want to insert. Everything you write now will be inserted to the left of the *cursor*.

Overwrite

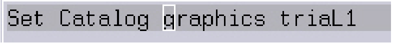
You can switch the entry mode into the overwrite mode. Press the Insert key. The *cursor* which was previously filled with color will now be displayed with its outline only.
If you now place the *cursor* on a character you want to change and enter something else, the previous text will be overwritten. With every new entry you will overwrite your old text from left to right character by character.

You can toggle between these two modes using the `Insert` key.



Set Catalog graphics triaL1

cursor in insert mode



Set Catalog graphics triaL1

cursor in overwrite mode

Shortcuts
Cut/Copy/Paste

The key combination `CTRL-X` cuts the selection and puts it on the clipboard, `CTRL-C` resp. `CTRL-INSERT` copies the selection and puts it on the clipboard and `CTRL-V` resp. `SHIFT-INSERT` inserts the clipboard contents.
The clipboard can be used to copy text from one to another DSP, or from the PC to a DSP (if using LGRemote).

Retireve Previous
Entries

By pressing the function key `F9` repeatedly you can retrieve the last 16 inputs to the *input line*, provided they were actually executed resp. confirmed with *Return* or [`ok`].
With `F10` you can scroll forward again accordingly.

Confirm Input
Return

All keyboard entries have to be confirmed with the *Return* key. The position of the *cursor* is irrelevant for this. When you press *Return* everything entered into the *input line* is taken over.

Shift / Return

An exception to this is the key combination *Shift* and *Return*. If you confirm with *Shift/Return* everything to the right of the *cursor* is truncated and will not be processed.

Values By
Mouse Poti

Instead of entering specific values by keyboard many *input field* values can be set using the so called *mouse poti*.

For this you click into the corresponding field with the left mouse button, keep the button pressed and move the mouse up (displayed value will increase) or down (value will decrease). When the desired value is reached simply release the mouse button.

For more precise adjustments use the middle mouse button instead of the left mouse button or the left mouse button while you press the *CTRL* key. It is not necessary to additionally confirm entries by *mouse poti*.

Values By
Mathematic functions

To enter or change a value, you can use mathematic functions. For example if you want to rotate a picture 7 times around an axis, you enter the value 360 and multiplicate that with 7.

→ 360*7

The DSP will calculate that for you and enter the value.

Entries By Selection
Menu Bar

If you click on an *input field* the *menu bar* at the bottom will automatically change, offering you different selection options. You only have to click on the applicable field or press the corresponding function key.

[...]

Numeric values will be taken over directly when using this input method.

All commands in squared brackets [...] are either immediately executable commands, commands that lead to further commands or aborting commands and will be effective immediately.

< ... >

Everything set in angle brackets < ... > shows which entry is now expected or alternatively possible. When you click on these fields, with some of them an additional window with predefined selection options (e.g. *variables*) will open.

The specific input options will be discussed when the individual commands are explained.

2.4 Deleting/Copying Commands

Delete

Deleting one single *command line* from your trickfilm is very simple:
Click on the respective line and press the function key `F3` or click directly on `Delete` in the *menu bar* at the bottom.

The *input line* now reads `Delete`. However, your line is still there.

The *menu bar* at the bottom has also changed.
You can now choose between `[ok]` and `previous`.
You can now decide whether to really delete the line because the computer awaits your final ok.

Confirm the delete command by pressing `F1` or *Return* or click on the button `[ok]`.

If you changed your mind press ESC and the *menu bar* will show its usual options.

You can avoid having to confirm the delete command by double clicking on the `Delete` button with the left mouse button.
The delete command will then be executed immediately.

However, these double clicks should not become habitual because: **gone is gone!** 💣

Copy

→ 2.5 Copying Blocks

To copy a line click on it with the middle mouse button. It will then be marked in a different color than when you click on it with the left mouse button. The procedure then is the same as for copying a *block*. See the next chapter.

2.5 Deleting/Copying Blocks

Selecting A Block

If you want to delete several connected lines you can of course do it the way described in the previous chapter, deleting line by line. However, if you want to delete a large number of lines there is a more efficient way.

Selecting A Block

You can select all the lines you want to delete as a so called *block*. For this, click on the first line which is to be deleted with the middle mouse button, *scroll* the film to the last line you want to delete and click on this one also with the middle mouse button. You will see that all lines in between are now marked in a different color.

You can also drag the mouse down to the last line of the intended *block* with the middle mouse button pressed. You will then see the colored marking immediately while you drag the mouse over the lines.
If at the end you mark too many lines, keep the mouse button pressed and go back up over the unnecessary lines. You will see that the marking color disappears again. Then release the mouse button and the *block* is selected.

→ *Block* Copy / Insert

Now you can delete all selected lines or copy them and insert them elsewhere.

If you want to add a line to the block you can do that by clicking on the line with the middle mouse button, either at the beginning or the end of the *block*. By the same method you can also deselect lines from the block.

However, do not drag the mouse, otherwise the marking of the whole *block* will be undone and a new selection begins.

Defining A *Block* By *Menu Bar*

You can also define a *block* using the *menu bar* at the bottom and by keyboard entries.

Click on [`more`] with the left mouse button or press `F7` to call up more input options.

`BeginBlock`

Click on `BeginBlock` or press `F1`, Now you have several selection options.

`<line>`

If you click on `<line>` or press `F1` the input line will show the number of the line which you had last clicked on in the film, i.e. this is now the starting line of your *block*.

`Top`

If you click on `Top (F3)` instead, the *block* will begin with the first line of the film.

`Bottom`

If you click on `Bottom (F4)` the *block* will begin with the last line of the film, which, however, does not make a lot of sense.

Having `Bottom` available as a selection option does, however, make sense if you know that you can enter two parameters under `BeginBlock`, namely starting- and ending point of the desired *block*.

The first entry defines the beginning of the *block*, the second one its end.

So if you first click on `Top`, then on `Bottom` and then confirm you will select the complete film.

By clicking on `Top` first and then on `<line>` you define the block from the beginning of the film up to the current line.

Entering Line Numbers

If you know the numbers of the first and last lines of the desired *block* you can also enter the corresponding numbers. Put a space between the start and end numbers and confirm with *Return* and the *block* is selected.

The fastest method to select a complete film is entering:

☞

```
bb t b  (BeginBlock Top Bottom)
```

This also works with line numbers, e.g.:

`bb 105 b`	selects a *block* from line 105 to the end
`bb t 56`	selects a *block* from the beginning to line 56
`bb 48 201`	etc.

Deleting A *Block*

You have selected a *block* and now you want to delete it.
For this press `F7` or click on [more] in the *menu bar* at the bottom.
Then press `F5` or click on `DELeteBlock`.

`[more]`
`DELeteBlock`

For security reasons the program does not execute your command immediately but waits for a final ok.
You can confirm the command with *Return* or `F1` or a mouse click on `[ok]`.

💣

By double clicking on `DELeteBlock` you avoid having to confirm the command.

By entering the keyboard shortcut `delb` or `db` the command is also executed immediately.

But be careful, as mentioned before: **gone is gone!**

Block
Copy / Insert

You can copy a section of the film of any length which you have marked as a *block* and insert it elsewhere in the same film or another film.

For this click on [more] in the *menu bar* at the bottom or press F7.

Then click on InsertBlock or press F4.
In the *menu bar* you can now choose between:
[ok] , <line>, Top, Bottom and /CopyProt.

[ok]

If you have already clicked on the film and the spot where you want to insert the *block* you only have to click on [ok] or confirm with F1 or *Return*.

☞

Provided after selecting the *block* you have already clicked on the line in front of which you want to insert the sequence it is sufficient to double click on InsertBlock and the command is executed.

Top

If you click on Top resp. press F3 the *block* will be inserted at the beginning of the active film.

Bottom

If you click on Bottom (F4) the *block* will be added at the end accordingly.

If you want to copy a sequence from film 1 to film 4 you have to make sure that the 'target film' is activated, not the 'source film'.

In the field <line> you see from the angled brackets that the computer expects you to enter a line number here to specify where to insert the *block*.
When you click on <line> the *input line* will show the number of the line which you had last clicked on in the trickfilm. You can then check once more if you really want to insert the *block* here, if yes confirm.

If the *block* is to be inserted elsewhere and you know the line number, overwrite the wrong line number in the *input line* and confirm with [ok] or F1 or *Return*.

If you do not know the number delete the number, click on the right spot in the film and confirm.

☞

As was the case when selecting a *block* you can save detours and thereby time when inserting by working with keyboard shortcuts.

When you have selected the correct spot for insertion just enter `ib` and confirm with *Return*.

For placing a block at the beginning of a new film activate the respective film and enter:

```
ib t
```

Other examples:

`ib b`	the block is added at the end
`ib 123`	the block is inserted before line 123

You only need the command `/CopyProt` if you want to copy sequences from a protected but still editable film.

`/CopyProt`
→ 2.7 Saving

💣

Be careful when you copy parts from a protected film to another film because you also copy the protection. The film will then not run on any other computer.

2.6 Compiling

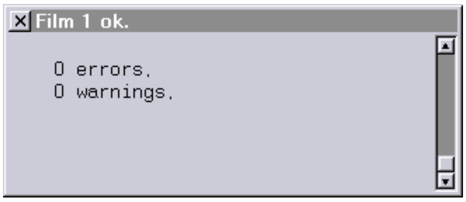
Before you can start a film it has to be *compiled*.
The computer checks your film for consistency.

[Compile]

To start the *compilation* you can either use the *menu bar* at the bottom (find the command [Compile] with [More]) and then click [Compile] at the bottom to the left or press F1.

A faster way is to simply enter c and press *Return*.

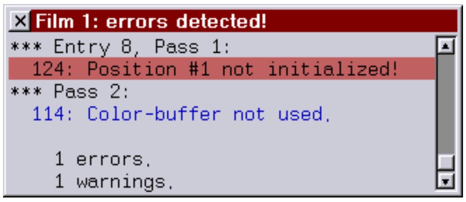
After *compiling* is completed you will see a new window on the left side of your screen which at best will look like this:



Film 1 ok.

```
    0 errors.
    0 warnings.
```

However, if you see something like in the window below your film will not run until you have fixed the error.



Film 1: errors detected!

```
*** Entry 8, Pass 1:
  124: Position #1 not initialized!
*** Pass 2:
  114: Color-buffer not used.

    1 errors.
    1 warnings.
```

As you can see the *compilation* program helps you and lets you know which error occurred and in which line you can find it. By clicking on the errors or warnings the film will show you the corresponding film command.

If only warnings are listed and no errors the film can be started anyway.

Warnings point out command lines that do not make sense, mostly they are redundant. The film will run but it is advisable to correct even these little mistakes.

If you click on [run] at the bottom or press F1 without having *compiled* before, the *compilation* will automatically be executed before the film starts.

The *compilation* will also be run automatically if necessary when you start a film or a *time script* in the *main menu*.

2.7 Saving

The laser show which you have programmed is at the moment in the random access memory of your **Lasergraph DSP**, but only for as long as the computer is supplied with power.

If you want to keep your show you have to transfer it from the random access memory to a floppy disk or the hard drive. Only then it is permanently saved and available for later editing or performance.

Remember: 'save often - save early'.
In other words, better save once too much than once too little. ☞

If for instance you save every 30 minutes you only lose a maximum of 30 minutes of work in case of a power failure.
It is always possible that someone accidentally pulls the plug on the **Lasergraph DSP** when you have been programming for three or four hours and forgot to save...

One more advice:
Give the film a different name or number each time you save. ☞
If a show is named 'Funny' save consecutively as Funny1, Funny2, Funny3...
In case of an error you can then always fall back on the last version which is 30 minutes old at the most.
When the show is finished you can simply delete all the other versions.

Unfortunately it is not sufficient to click somewhere on 'save' and everything is saved.

First open the `File` *window* by clicking on `[File]` in the *menu bar* at the bottom or by pressing `F5`.

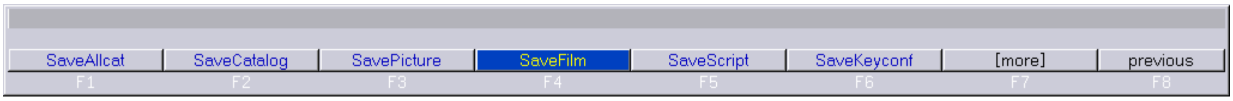
In the `File` *window* make sure you are in the *hard disk sector* and the *directory* where you want to save the show. If necessary switch to the right sector or to the floppy disk drive `A:\`.

→ `File` Commands

In case you want to create a new *directory* for the show do it now.

Then click on [save] at the bottom or press F4.

You now have a lot of possibilities of selecting individual show components, all of which you can save separately.



Let us assume that you want to save the film which you are currently editing.

SaveFilm

Click on SaveFilm or press F4.
Now the computer wants to know which film you want to save, Film1, 2, 3 or 4? Click on the corresponding button or press the *function key* shown underneath.
Now enter a name for the film and confirm with *Return*.

Now you have saved the film - and nothing but the film.

If you have used graphics from different catalogs from other *directories* in the film you should always also copy these catalogs into the current *directory*.
When you load the film again the used catalogs are also loaded automatically, but only if they are located in the same *directory* as the film.

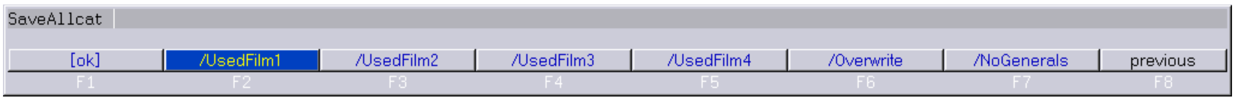
→ File Commands

The only exception are catalogs stored in the *directory* Generals.

All other catalogs cannot by found by the computer and you will get an error message.

SaveAllcat

To avoid this click on SaveAllcat or click on F1. You will see the following:



SaveAllcat: [ok]

If you now simply click on [ok] all catalogs located in the memory at the moment will be saved to the current *directory*, whether you used them in the film or not.

If these catalogs are already saved somewhere else on your hard drive this does not make much sense because you waste storage space this way.

It makes more sense to only save what is actually necessary for the film.

So if you have programmed and saved Film1 click on `/UsedFilm1` or press F2 and confirm with [ok], F1 or the *Return* key.
All catalogs used in the film will then be saved, but from these only the pictures which are actually used.

`SaveAllcat: /UsedFilm1`

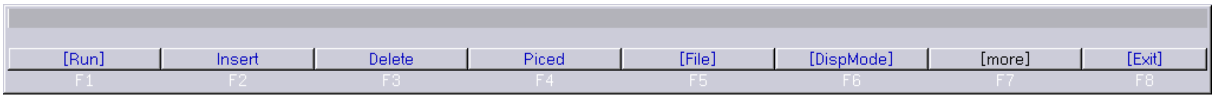
On the one hand this saves storage space, on the other hand it can turn into a nasty trap if you do not have a consistent structure on your hard drive.

→ `File` Commands

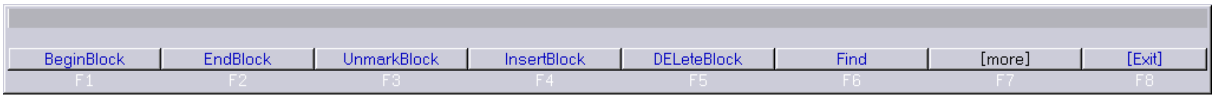
Save your films as a `Boot` after completion at the latest and delete all ballast that has accumulated in the *directory* during the programming phase, i.e. all individually saved film versions, catalogs, *time scripts* etc.

→ `CreateBoot`

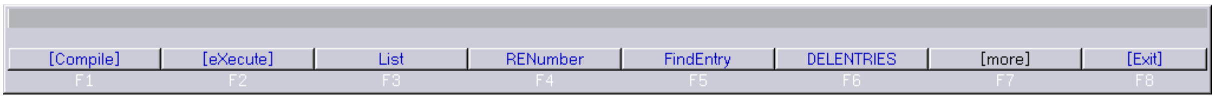
2.8 Commands in the Menu



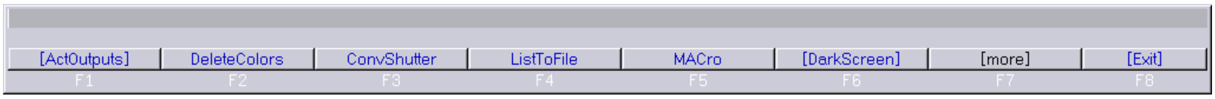
`[Run]`	starts the Film	Function key F1
`Insert`	opens the Trickfilm commands window and allows you to insert commands	Function key F2
`Delete`	deletes the current command in the Film window	Function key F3
`Piced`	changes to the Picture Editor with 4 options:	Function key F4
`[lastedit]`	→ keeps the last picture in the Piced	
`[byfilm]`	→ takes the current picture in the film into the Piced	
`<catalog>`	→ let's you choose a picture	
`[browse]`	→ opens the browser to choose a picture	
`[File]`	opens the File window	Function key F5
`[DispMode]`	toggles the laseroutput to a single frame or to multiple pictures. When in single mode it shows the current frame.	Function key F6
`[more]`	jumps to the next menu level	Function key F7
`[Exit]`	changes into the main menu (Root)	Function key F8



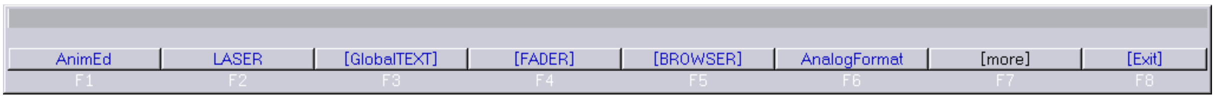
`BeginBlock`	marks the beginning of the selection (block)	`Function key F1`
`EndBlock`	marks the ending of the selection (block)	`Function key F2`
`UnmarkBlock`	unmarks the selection (block)	`Function key F3`
`InsertBlock`	inserts the selection (block) before the current film position	`Function key F4`
`DELeteBlock`	deletes the selection (block) `"DB"` also works as shortcut for this	`Function key F5`
`Find`	search the film for a certain command	`Function key F6`
`[more]`	jumps to the next menu level	`Function key F7`
`[Exit]`	changes into the main menu (Root)	`Function key F8`



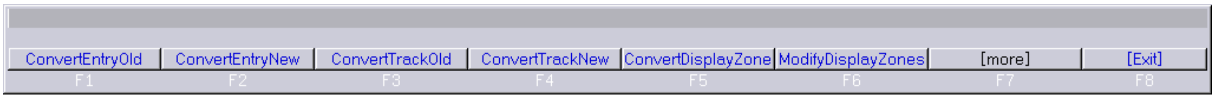
`[compile]`	compiles the film	`Function key F1`
`[eXecute]`	compiles the film anf starts it if possible	`Function key F2`
`List`	jumps to a certain line or to the top/bottom	`Function key F3`
`RENumber`	changes the numbers of the Entries in the film	`Function key F4`
`/BlockOnly`	→ renumbers only Entries that are selected	
`/NoScript`	→ renumbers without changing the corresponding Entries in the Timescript	
`/NoKeyconfig`	→ renumbers without changing the corresponding Entries in the Keyconfig	
`FindEntry`	locates an Entry	`Function key F5`
`DELENTRIES`	Deletes all unused Entries	`Function key F6`
`[more]`	jumps to the next menu level	`Function key F7`
`[Exit]`	changes into the main menu (Root)	`Function key F8`



[ActOutputs]	informs you about the programmed DMX outputs in the film	Function key F1
DeleteColors	deletes all unused primary colors from the current trickfilm after a security query is confirmed	Function key F2
ConvShutter	converts DMX Channels that were defined as Logoshutter in the AdvancedParameters into the command LogoShutter	Function key F3
ListToFile	saves the film command structure to a text file	Function key F4
MACro	recording or playing back a Macro	Function key F5
[DarkScreen]	fades the screen to gray	Function key F6
[more]	jumps to the next menu level	Function key F7
[Exit]	changes into the main menu (Root)	Function key F8



AnimEd	opens the AnimationEditor	Function key F1
LASER	opens the Laser Control window	Function key F2
[GlobalTEXT]	opens the Global Text Table	Function key F3
[FADER]	opens the FADER emulation	Function key F4
[BROWSER]	opens the Browser window	Function key F5
AnalogFormat	switches the numerical format of analog values in the Keyconfig and in the Trickfilm- and Timescript commands SetAnalogs between "Fractional", "Decimal", "Hexadecimal" and "Percent".	Function key F6
[more]	jumps to the next menu level	Function key F7
[Exit]	changes into the main menu (Root)	Function key F8



`ConvertEntryOld` converts all old Entry commands (without description) into new Entry commands (with description) `Function key F1`

`ConvertEntryNew` converts all new Entry commands (with description) into old Entry commands (without description) `Function key F2`

`ConvertTrackOld` converts all old Track commands (without description) into new Track commands (with description) `Function key F3`

`ConvertTrackNew` converts all new Track commands (with description) into old Track commands (without description) `Function key F4`

`ConvertDisplayZone <dgc> [<dgc> [<dgc> ...]]` `Function key F5`

converts the display commands `DisplayPicture`, `DisplayAnimation` and `DisplayBuffer` into the display zone commands `DisplayPictureZone`, `DisplayAnimationZone` resp. `DisplayBufferZone`.

If a block is selected in the active Trickfilm, the command will only affect the display commands within the selected block. Otherwise the display commands in the whole Trickfilm will be converted.

`ModifyDisplayZones <dgc> [<dgc> [<dgc> ...]]` `Function key F6`

modifies the DGC zones entered in the display zone commands `DisplayPictureZone`, `DisplayAnimationZone` and `DisplayBufferZone`.

The following subcommands are available:

`ModifyDisplayZones.Set <dgc> [<dgc> [<dgc> ...]]`

The specified DGC zones are entered in the display zone commands.
All other DGC zones are removed.

`ModifyDisplayZones.Add <dgc> [<dgc> [<dgc> ...]]`

The specified DGC zones are added to the display zone commands.
All other DGC zones remain unchanged.

`ModifyDisplayZones.Remove <dgc> [<dgc> [<dgc> ...]]`

The specified DGC zones are removed from the display zone commands.
All other DGC zones remain unchanged.

`ModifyDisplayZones.Copy <from> <to>`

Within the display zone commands, the DGC zone <from> is copied to the DGC zone <to>.
The previous state of the DGC zone <to> is lost.

`ModifyDisplayZones.SWap <1st> <2nd>`

Within the display zone commands, the two DGC zones <1st> and <2nd> are swapped.

If a block is selected in the active Trickfilm, the "ModifyDisplayZones" commands only affect the display zone commands within the selected block. Otherwise the display zone commands in the whole Trickfilm will be modified.

`[more]`	jumps to the first menu level	`Function key F7`
`[Exit]`	changes into the main menu (Root)	`Function key F8`

3. The Film Commands

The **Lasergraph DSP** offers you a large selection of film commands.

In this chapter we will describe all film commands sorted by groups, just as you will find them in the *command window.*

3.1 Comment

The command `Comment` is a very useful orientation guide in your films.

It has no influence on the film and you can insert it as often and wherever you like.

You can never have too many `comments`.
If after months or years you look for something or simply look through your old films you will be grateful for every comment! ☞

For instance, begin every film with a `comment`. Make a note of the purpose for which the film is programmed, which music is used, date and anything else that seems important to you.

It is also advisable to provide each *entry* with a short description of the sequence. That makes every trickfilm more clear and it makes show programming easier.

You can fit more than one line of text into a `comment`, but the *program line* will only show the first words. If it is important to you that you can read the complete commentary with one look at the displayed section of the film without clicking on the *comment* line, you can insert as many *comments* consecutively as you want and divide your text accordingly.

Displayed command line in the film:
Beginning of comment
Position in the program: Wherever you like
Keyboard shortcut: `c`

3.2 Absolute Jump Commands

3.2.1 Entry

Entries are used to divide the trickfilm into sequences which can be called up independently.

You can use up to 1000 *entries* per trickfilm (0-999).

Entries are the entry points into the film. They allow you to call up individual film sequences in different ways and on different levels.

- You can program jumps to *entries* within a trickfilm with `Goto`, `ConditionalGoto` or `OnGoto`.

→ Chapter 3.2.5 `Goto`

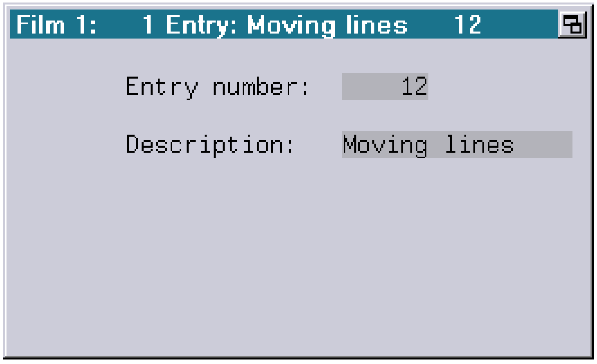
- While the film is running you can call up specific film sequences with the `Goto` command in the *menu bar* at the bottom (F5) + desired *entry* number or jump one *entry* back or forward with `GotoNext` (F6) resp. `GotoPrev` (F7).

`Goto(F5)`
`GotoNext(F6)`
`GotoPrev(F7)`

- In the `time script` you select the desired film sequence via the *entry* number.
- When programming the *keyboard configuration* you assign the keys by entering the *entry* number of the corresponding film sequence.

Parameter: `Description`

In the Entry command an explanatory description can be entered in addition to the Entry number.



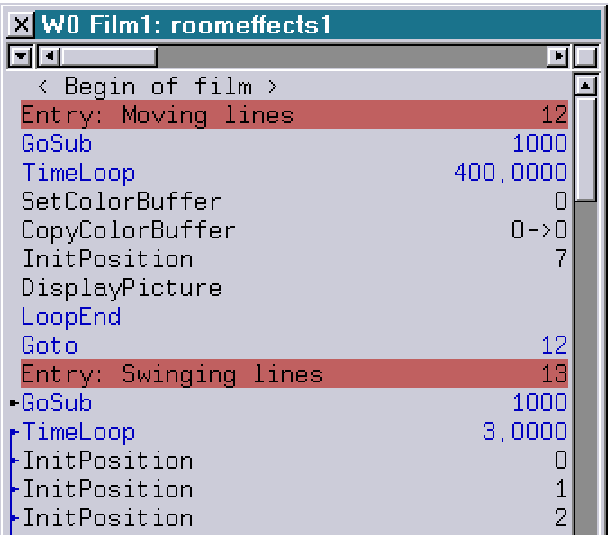
Film 1: 1 Entry: Moving lines 12

Entry number: 12

Description: Moving lines

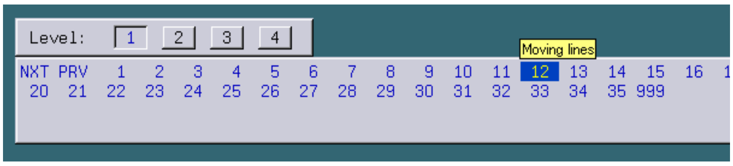
The description entered there will be displayed at the following points:

1) In the Trickfilm editor:



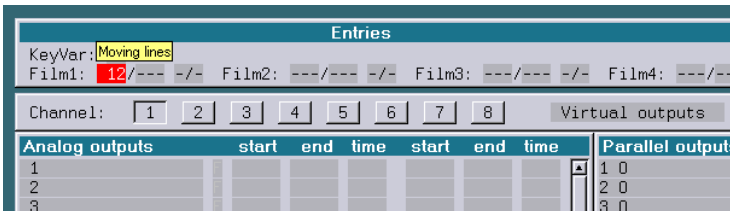
```
W0 Film1: roomeffects1
  < Begin of film >
Entry: Moving lines            12
GoSub                        1000
TimeLoop                 400,0000
SetColorBuffer                  0
CopyColorBuffer              0->0
InitPosition                    7
DisplayPicture
LoopEnd
Goto                           12
Entry: Swinging lines          13
GoSub                        1000
TimeLoop                   3,0000
InitPosition                    0
InitPosition                    1
InitPosition                    2
```

2) As a tooltip in the Entry table:



```
Level:  1   2   3   4
                                                    Moving lines
NXT PRV  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  1
 20  21 22  23  24  25  26  27  28  29  30  31  32  33  34  35 999
```

3) As a tooltip in the Keyconfig:



```
                              Entries
KeyVar:Moving lines
Film1:  12/--- -/-   Film2: ---/--- -/-   Film3: ---/--- -/-   Film4: ---/--

Channel:  1   2   3   4   5   6   7   8     Virtual outputs

Analog outputs       start   end   time   start   end   time    Parallel output
1                                                              1 0
2                                                              2 0
3                                                              3 0
```

Attention!

- Show producers should be aware of the fact that the Entry command with description is only available on the Mark 2 starting with version 2009/09/23.
- A show containing this Entry command will not run on older systems.
- If a show is to be programmed for such older systems, the old Entry command (`EntryOld`) has to be used!

Displayed command line in the film: `Entry + Description` and number (0-999)
Position in the program: Beginning of a sequence
Keyboard shortcut: `e`

3.2.2 EntryOld

The command `EntryOld` is similar to the command `Entry`. The difference beween `Entry` and `EntryOld` is that `EntryOld` does not contain a description.

`EntryOld` has to be used when a show is produced for an old Lasergraph DSP.

Displayed command line in the film:
`EntryOld` and number (0-999)
Position in the program: Beginning of a sequence
Keyboard shortcut: `eo`

3.2.3 Subroutine

The command `Subroutine` marks the beginning of a subroutine.

Programming a subroutine always makes sense when the same sequence is needed several times in a film.

Subroutines save you a lot of programming work because they are always executed during the course of the program when they are called up with a `GoSub`, `ConditionalGoSub` or `OnGoSub` command and the respective *subroutine* number. After a subroutine has been executed the film will continue at the position it has been before.

→ Chapter 3.2.6
`GoSub`

It is important to know this because it is common practice and advisable to set the default *values* in a subroutine and to call this subroutine at the beginning of every film sequence.

The number of a *subroutine* can be in range from 1000 to 1499.

Each subroutine has to be concluded with a `RETurn`.

→ Chapter 3.2.7
`RETurn`

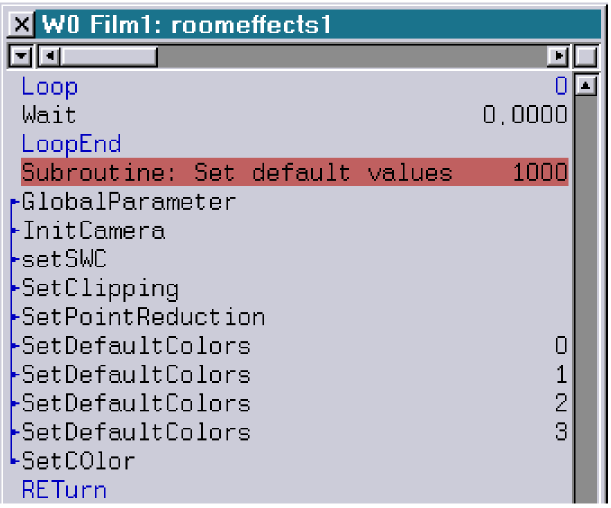
All subroutines have to be placed at the end of the film behind all other *entries*! 💣

`Description`

In the `Subroutine` command an explanatory description can be entered in addition to the subroutine number.

The description entered there will be displayed at the following places:

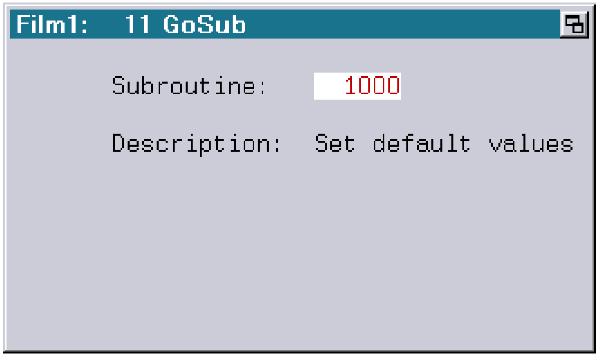
1) In the Trickfilm editor:



W0 Film1: roomeffects1

```
Loop                                    0
Wait                               0.0000
LoopEnd
Subroutine: Set default values       1000
GlobalParameter
InitCamera
setSWC
SetClipping
SetPointReduction
SetDefaultColors                        0
SetDefaultColors                        1
SetDefaultColors                        2
SetDefaultColors                        3
SetCOlor
RETurn
```

2) In the command GoSub:

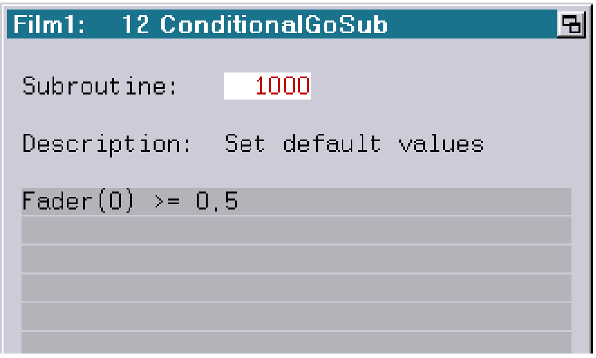


Film1: 11 GoSub

Subroutine: 1000

Description: Set default values

3) In the command ConditionalGoSub:



Film1: 12 ConditionalGoSub

Subroutine: 1000

Description: Set default values

Fader(0) >= 0.5

Attention!

- Show producers should be aware of the fact that the `Subroutine` command with description is only available on the Mark 2 starting with version 2014/03/03.
- A show containing the `Subroutine` command will not run on older systems.
- If a show is to be programmed for such older systems, the old `EntrySubroutine` command has to be used!

Displayed command line in the film:
`Subroutine` + `Description` and number (1000-1499)
Position in the program: Beginning of a sequence, behind all other *entries*
Keyboard shortcut: `s`

3.2.4 EntrySubroutine

The command `EntrySubroutine` is similar to the command `Subroutine`.
The difference beween `Subroutine` and `EntrySubroutine` is that `EntrySubroutine` does not contain a description.

`EntrySubroutine` has to be used when a show is produced for an old Lasergraph DSP.

Displayed command line in the film:
`EntrySubroutine` and number (1000-1499)
Position in the program: Beginning of a sequence, behind all other *entries*
Keyboard shortcut: `es`

3.2.5 Goto

The **`Goto`** command makes the program jump to an *entry* which was selected accordingly.

→ Chapter 3.2.1 `Entry`

The `Goto` command has to be handled carefully. It can lead to very confusing and unesthetic program structures. You should therefore use it as sparingly as possible.

The command actually only makes sense if you want your program to run endlessly and you use the `Goto` command at the end of the endless loop to get back to the first *entry*.

Displayed command line in the film:
`Goto` with selected *entry* number (0-999)
Position in the program: Always at the end of a sequence and outside of the time loop
Keyboard shortcut: `g`

3.2.6 GoSub

With **`GoSub`** you tell the computer to jump to the selected `subroutine`.

→ Chapter 3.2.3 `Subroutine`

Displayed command line in the film:
`GoSub` with selected *entry* number (1000-1499)
Position in the program: After the *entry* and outside of the time loop
Keyboard shortcut: `gs`

3.2.7 RETurn

The **`RETurn`** command has to end each `subroutine`.

→ Chapter 3.2.3 `Subroutine`

As soon as the `subroutine` has been completed and the command `RETurn` is reached the program jumps back to the place in the film where the `subroutine` was called up.

Displayed command line in the film:
`RETurn`
Position in the program: always at the end of a subroutine
Keyboard shortcut: `ret`

3.3 Time Control

3.3.1 Loop

The **Loop** command lets one or more connected sequences of your film run repeatedly.

Every Loop command has to be ended with LoopEnd at the end of the sequence which is to be repeated.

→ 3.3.6 LoopEnd

Number of Repeats:
Here you specify how many times the program sequence is to be run. Repeats from 1 to 65535 are possible.

With the repeat number 0 you program an endless loop.

Displayed command line in the film:
Loop with the number of repeats
Position in the program: At the beginning of the sequence which is to be repeated
Keyboard shortcut: l

3.3.2 TimeLoop

With the **TimeLoop** command you program a time loop for which you specify a fixed duration.

Like every other loop, the TimeLoop also **has to** be concluded with LoopEnd.

→ 3.3.6 LoopEnd

The sequence between the commands TimeLoop and LoopEnd runs for the period of time which you enter.
The display window shows the time in seconds which you can change:

Time in seconds:
Here you enter the time in seconds during which the sequence is be executed.

The possible period of time ranges from 0.0 to 400.0 seconds.

Displayed command line in the film:
TimeLoop with time in seconds
Position in the program: At the beginning of the sequence which is to be repeated
Keyboard shortcut: tl

3.3.3 ExternTimeLoop

With the **ExternTimeLoop** command the running time of the loop is determined through the time script.
In contrast to the TimeLoop you cannot enter any time parameters for this command.

Using the ExternTimeLoop is surely the most elegant and simplest way to program a music synchronous timecode controlled laser show.

You can only use the ExternTimeLoop in connection with a time script and an external or internal timecode.

The ExternTimeLoop also has to be concluded with LoopEnd.

→ 3.3.6 LoopEnd

Displayed command line in the film:
ExternTimeLoop
Position in the program: At the beginning of the sequence which is to be repeated
Keyboard shortcut: etl

3.3.4 Track

With the command **Track** you can divide each loop into individual segments. These segments can be shown independent of each other and with different running times within the loop.

All commands following a `track` are time controlled by the `track` up to the next `track` or `LoopEnd`.

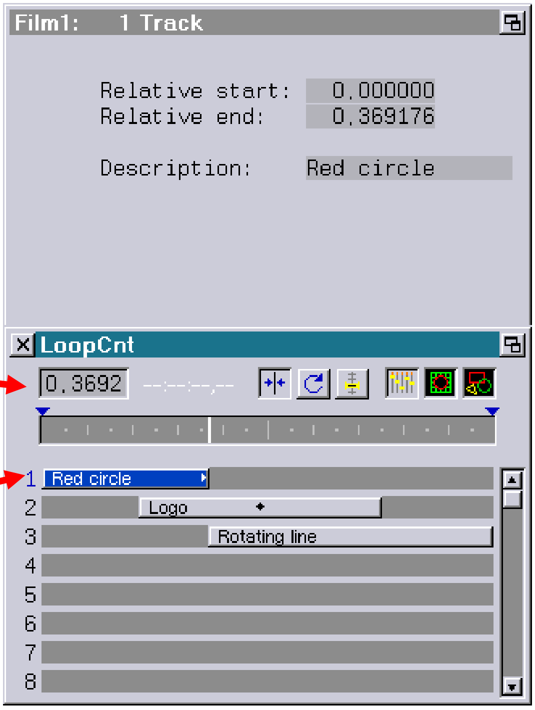
You can work with up to 99 `tracks` in one loop.

The `tracks` are displayed in the lower part of the `LoopCount` window in the form of numbered horizontal bars, let us call them *sliders*.

→ `LoopCnt window`

The number of the `track` you are currently editing in the trickfilm is marked by a different color in the `LoopCnt` window.

Regardless of that the whole *slider* of the `track` which you are currently editing in the LoopCnt window has a different color.



Relative start: 0,000000
Relative end: 0,369176

Description: Red circle

Input options in the display window:
- Beginning and end of the running time of the track within the loop (0-1)
- Explanatory description

Adjustment of the tracks by *mouse poti*

Moving the tracks by mouse

Using the *sliders* you can lengthen or shorten the running time of each `track` and/or move it within the loop.

When you move the mouse pointer over the *sliders* small black triangles appear at each side and a small black rhomb in the center.

When you activate one of the *sliders* with a mouse click it is marked by its color and triangle or rhomb are highlighted.

Using the triangles you can change the relative start and end times of the track by grabbing the left or right end of a `track` *slider* with pressed left mouse button and dragging it in the desired direction with the mouse.

While grabbing the beginning resp. the end of a track these will snap to the position of the beginning or the end of a near-by track. The snap function can be disabled temporarily by keeping <Shift> or <Alt> pressed during positioning.

When you click on the rhomb and keep the mouse button pressed you can move the whole *slider* back and forth within the `TimeLoop` without changing the running time.

Timing Several `Tracks` Simultaneously

If you want to move several `tracks` simultaneously activate the corresponding *sliders* either with the middle mouse button or with the left mouse button while pressing the *Shift* key. Then drag the mouse in the desired direction.

!!!

When you keep the <Control> key pressed while you move several tracks simultaneously, the start and end times of the selected tracks are automatically aligned.
You have to release the left mouse button before the <Control> key, otherwise the old relations are restored.

☜

If you realize that you have made a mistake while moving the sliders you can undo the process while the left mouse button is still pressed by additionally pressing the right mouse button or the <ESC> key once.

To deactivate the selection of `tracks` simply click anywhere in the window beside or below the tracks with the left mouse button.

Another possibility is to activate the edge of a *slider* with a mouse click and then click with the mouse on the `LoopCnt` display field at the top to the left and change the value by *mouse poti*. You immediately see the corresponding changes to the *slider* of the activated `track`.

You can also use this method to time several tracks simultaneously by keeping the *Shift* key pressed.

The third possibility is to go to the command line of the `track` which you want to change and to specify the start and end values by *mouse poti* or keyboard input.
You see the changes in the `LoopCnt` window as soon as you release the mouse button resp. confirm the input.

`Relative start:`
Starting point of the track within the loop (0 - 1).

`Relative end:`
End point of the track within the loop (0 - 1).

`Description:`
An explanatory description can be entered.

Displayed command line in the film: `Track`
Position in the program: Within a loop before the sequence which is to be part of the track.
Keyboard shortcut: `t`

3.3.4.1 InsertTracks

In order to divide a loop into several sections it is necessary to insert the corresponding number of track commands and then adjust the start- and end times accordingly.

This procedure can be done automatically using one of the following commands:

```
InsertTracks.ByNumber
InsertTracks.FromTimescript
```

3.3.4.1.1 InsertTracks.ByNumber (or short: InsertTracks)

This variant of the command **`InsertTracks`** devides a loop into several equal sections:

```
InsertTracks  <number>  [<start offset>
                        [<end offset>]]
```

Parameter:

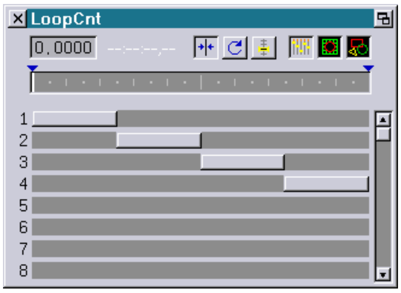
`Number` number of track commands to be entered

`start offset` optional offset for modifying the start times of all tracks

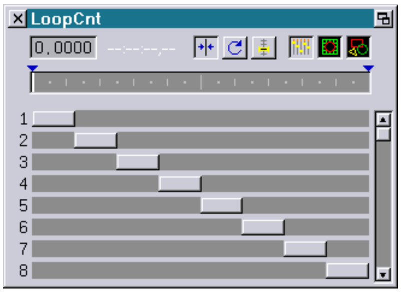
`end offset` optional offset for modifying the end times of all tracks

Examples:

```
InsertTracks  4
```



```
InsertTracks  8
```



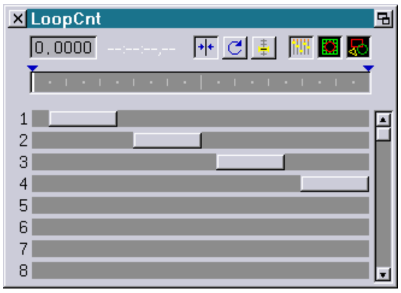
From time to time it may be necessary to start the tracks not at the exact moment described above but minimally earlier or later. For this case the optional parameters `<start offset>` and `<end offset>` are available.

The values relate to the length of the tracks and have to be in a range from -1.000 to +1.000.

Examples:

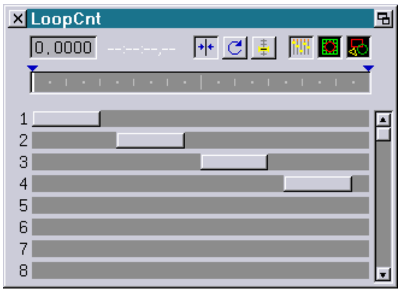
If a loop is to be divided into 4 equal sections with each track starting with a delay of 20% the corresponding command would be:

```
InsertTracks  4  0.2
```



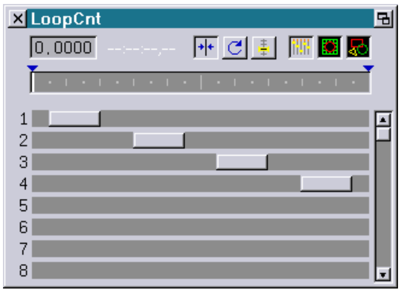
If a loop is to be divided into 4 equal sections with each track ending of 20% earlier the corresponding command would be:

```
InsertTracks  4  0.0  -0.2
```



It is also possible to combine both cases mentioned above:

```
InsertTracks  4  0.2  -0.2
```

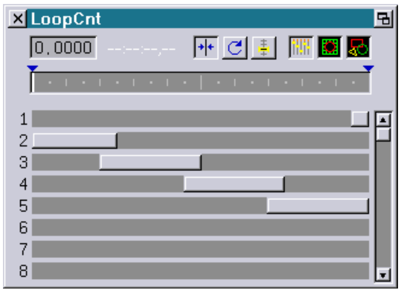


If a negative start offset or a positive end offset were to be specified, the track would begin before 0.000 resp. end after 1.000.

To account for this in this case an additional track would be inserted before the first track or after the last track.

Example:

```
InsertTracks  4  -0.2
```



If the divided track contains code which is LoopCnt-dependent (e.g. InitPosition with curves) the LoopCnt has to be corrected at the beginning of the track with a "`SetVariables`".

For the first track in this example this would be:

```
LoopCnt = LoopCnt * (0.2 / 1.2)
```

And for the second track:

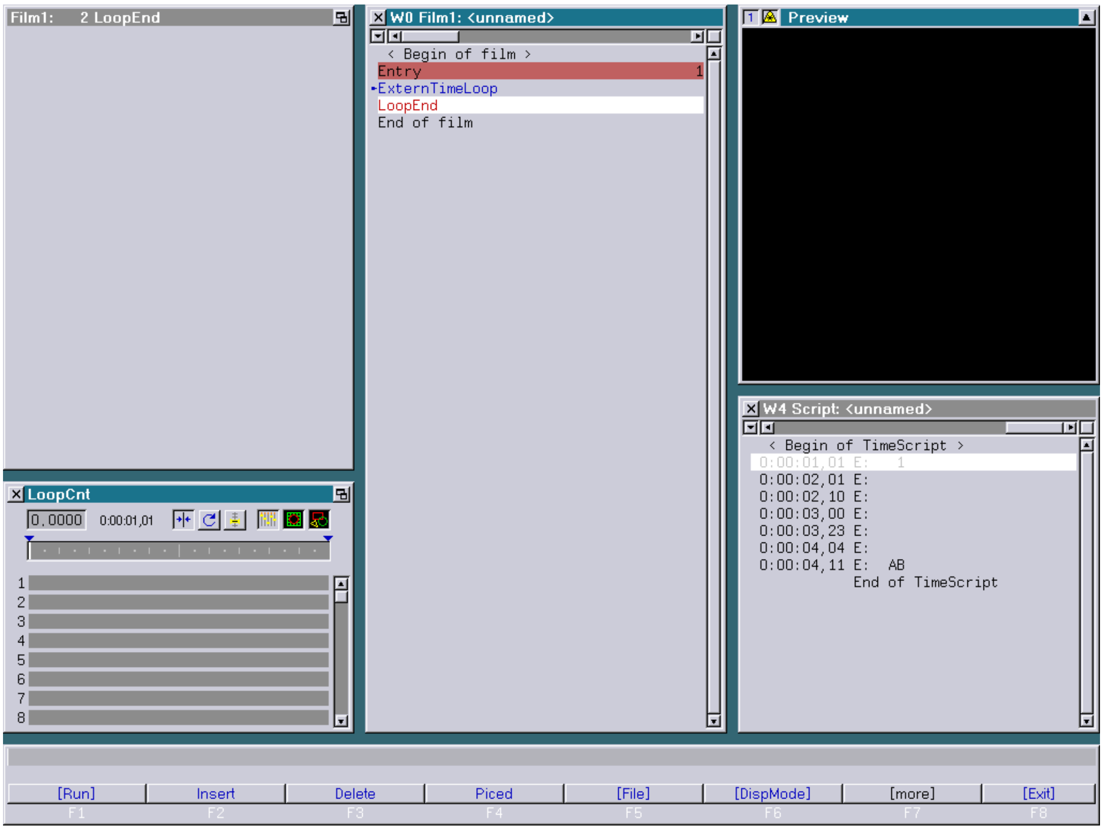
```
LoopCnt = (0.2 / 1.2) + (LoopCnt / 1.2)
```

3.3.4.1.2 InsertTracks.FromTimescript

With this variant of the command **`InsertTracks`**, the number of the tracks to be inserted and their positions are determined by **`Entry`** comamands in the Timescript.

Example:

The **`ExternTimeLoop`** in Entry 1 shall be divided by **`Track`** commands based on **`Entry`** commands within the Timescript:



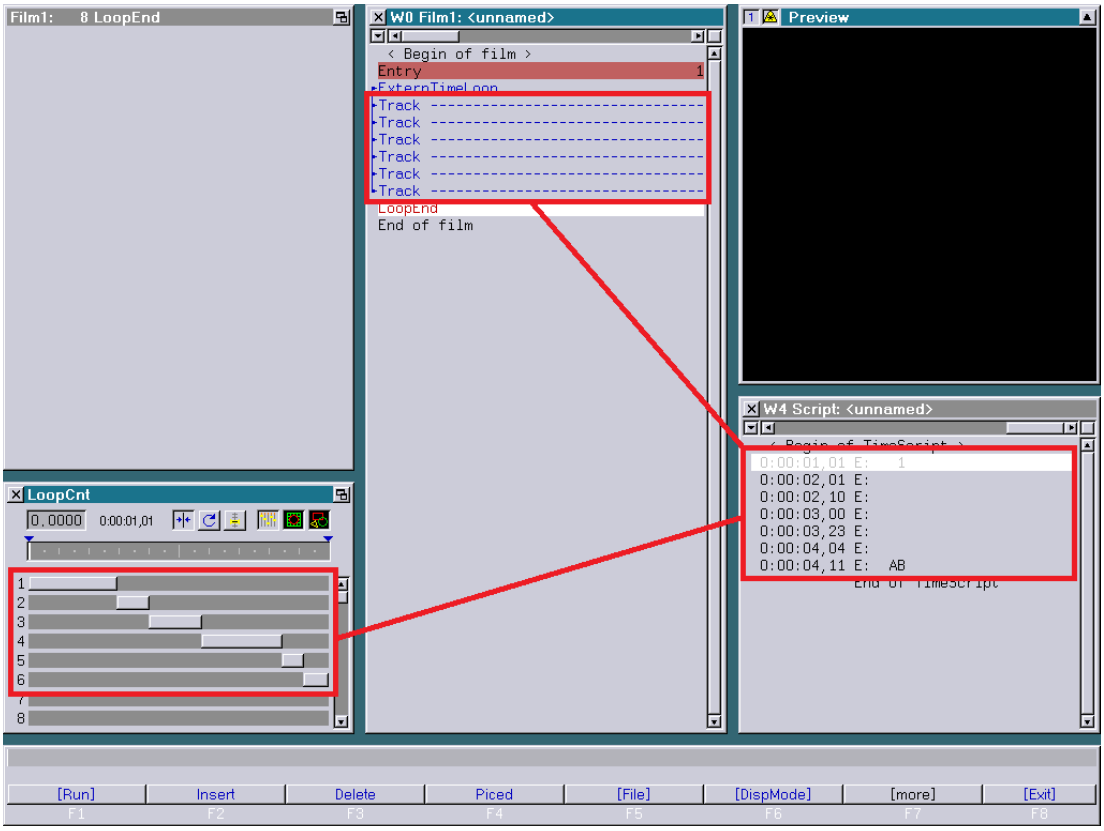
```
 < Begin of film >
Entry                                     1
•ExternTimeLoop
LoopEnd
End of film
```

```
 < Begin of TimeScript >
0:00:01,01 E:    1
0:00:02,01 E:
0:00:02,10 E:
0:00:03,00 E:
0:00:03,23 E:
0:00:04,04 E:
0:00:04,11 E:   AB
           End of TimeScript
```

By entering

`InsertTracks.FromTimescript`

`Track` commands based on the **`Entry`** commands in the Timescript are inserted at the current position in the Trickfilm:



```
W0 Film1: <unnamed>
  < Begin of film >
Entry                                  1
ExternTimeLoop
Track ------------------------------
Track ------------------------------
Track ------------------------------
Track ------------------------------
Track ------------------------------
Track ------------------------------
LoopEnd
End of film
```

```
W4 Script: <unnamed>
  < Begin of TimeScript >
0:00:01,01 E:   1
0:00:02,01 E:
0:00:02,10 E:
0:00:03,00 E:
0:00:03,23 E:
0:00:04,04 E:
0:00:04,11 E:  AB
       End of TimeScript
```

Note:

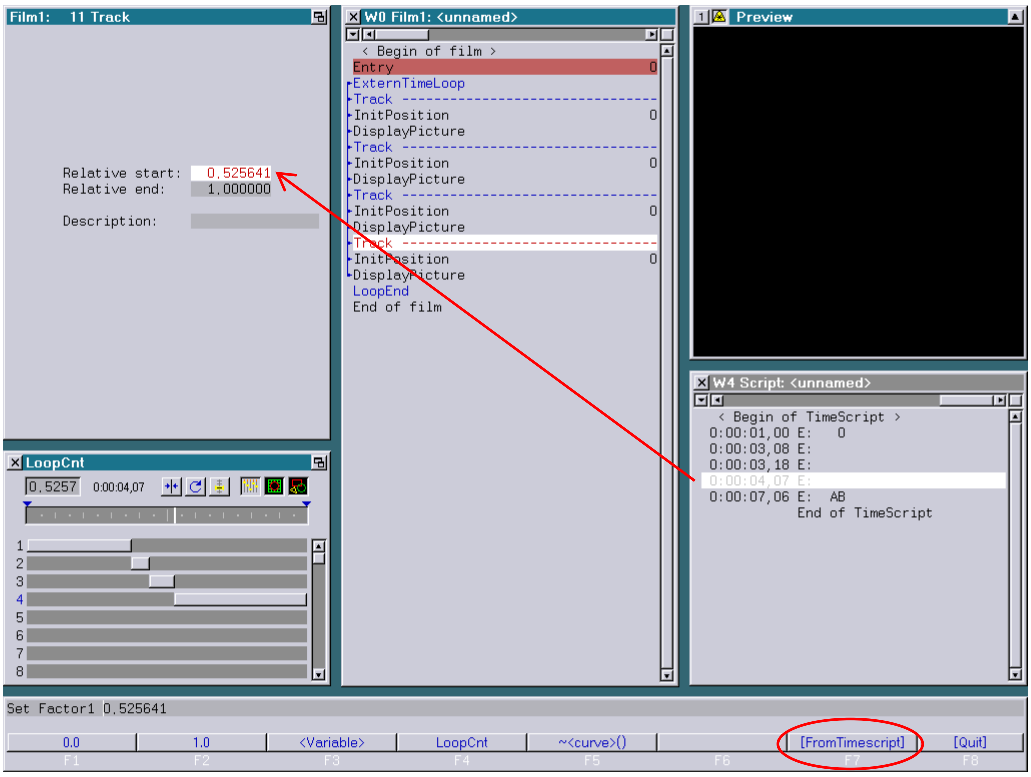
To enable **`InsertTracks.FromTimescript`** to calculate the desired track positions, the Entry in which the current line is located has to be called from the Timescript and the time of the current line in the Timescript has to be within the running time of the Entry.

3.3.4.2 FromTimescript

In practice most tracks are meant to start resp. end at a specific time.

Since the start- and end times in the track command have to be specified relatively, i.e. in a range from 0.000 to 1.000, it would be necessary to calculate the relative time on the basis of the absolute time in the timescript or to determine the relative time through trial and error.

The function "[FromTimescript]" calculates the relative track time from the absolute time of the current command from the timescript and enters this into the parameter "Relative start" resp. "Relative end":



Film1: 11 Track

```
Relative start:  0.525641
Relative end:    1.000000

Description:
```

W0 Film1: <unnamed>

```
 < Begin of film >
Entry                              0
ExternTimeLoop
Track ------------------------------
InitPosition                       0
DisplayPicture
Track ------------------------------
InitPosition                       0
DisplayPicture
Track ------------------------------
InitPosition                       0
DisplayPicture
Track ------------------------------
InitPosition                       0
DisplayPicture
LoopEnd
End of film
```

W4 Script: <unnamed>

```
 < Begin of TimeScript >
0:00:01,00 E:   0
0:00:03,08 E:
0:00:03,18 E:
0:00:04,07 E:
0:00:07,06 E:  AB
           End of TimeScript
```

Set Factor1 0.525641

For "[FromTimescript]" to work the entry containing the track command has to be called up from the timescript and the time of the current line has to be within the entry's running time in the timescript.

3.3.5 TrackOld

The command `TrackOld` is similar to the command `Track`. The difference beween `Track` and `TrackOld` is that `TrackOld` does not contain a description.

`TrackOld` has to be used when a show is produced for an old Lasergraph DSP.

Displayed command line in the film: `TrackOld`
Position in the program: Within a loop before the sequence which is to be part of the track.
Keyboard shortcut: `to`

3.3.6 LoopEnd

Each individual `Loop` command has to be ended with the command `LoopEnd`. Otherwise the *compiler* will display an error message and your film will not run.

Displayed command line in the film: `LoopEnd`
Position in the program: End of the sequence which is to be repeated
Keyboard shortcut: `le`

3.3.7 Wait

With the `Wait` command you interrupt the program for the time you specify. During this time no picture is displayed.

You can interrupt the `Wait` command at any time by calling up the next *entry*.

→ 3.2.1 `Entry`

The display window shows the time in seconds which you can change:

`Time in seconds:`
Here you enter the time in seconds for which the program is to be interrupted.

The possible time span ranges from 0.0 to 400.0 seconds.

Displayed command line in the film: `Wait` with time in seconds
Position in the program: At the end of a time loop
Keyboard shortcut: `w`

3.4 Color Control

3.4.1 SetColorBuffer

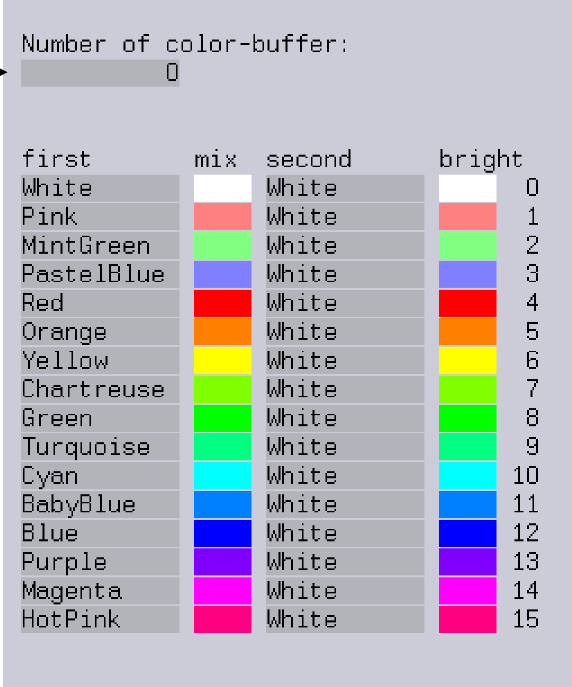
With the **SetColorBuffer** command you can define 16 colors for your picture output. You can set up to 8 different *color buffers*.

→ *Picture Editor* (Piced)

With this command you can easily recolor your graphics at any time.
Even if a graphic was digitized with different colors in the *picture editor*, with the **Lasergraph DSP** you can turn red lips blue or have the same woman dance with a yellow dress once and with a violet dress the next time.

When you enter the SetColorBuffer command and click on it you will see the following default *color buffer* in the *display window*.

Here you enter the number of the 8 possible *color buffers* (0 - 7)



Number of color-buffer:
0

first	mix	second	bright	
White	⬜	White	⬜	0
Pink	🟥	White	🟥	1
MintGreen	🟩	White	🟩	2
PastelBlue	🟦	White	🟦	3
Red	🟥	White	🟥	4
Orange	🟧	White	🟧	5
Yellow	🟨	White	🟨	6
Chartreuse	🟩	White	🟩	7
Green	🟩	White	🟩	8
Turquoise	🟩	White	🟩	9
Cyan	🟦	White	🟦	10
BabyBlue	🟦	White	🟦	11
Blue	🟦	White	🟦	12
Purple	🟪	White	🟪	13
Magenta	🟪	White	🟪	14
HotPink	🟥	White	🟥	15

first

In the column titled first you can change the basic colors.

Primary colors Window

When you click on one of the fields another window will open, the Primary colors window. You can look through all available colors using the *scroll bar* on the right side of the window.

When you click on one of the colors it will immediately be transferred to your *color buffer*, overwriting the color you clicked on there.

In the third column, titled `second`, you can define another color in the same way which can be mixed with the first one. `second`

For this you need the second column, titled `mix`. `mix`
Here you define the ratio between the `first` and the `second` color.

If you move the mouse pointer over this column in the default setting you will see a black `1.00` appear in the color fields. This means that the first color is displayed with 100%.

You can use the *mouse poti* for mixing. Click on the color field with the left or middle mouse button and drag the mouse upward or downward with the mouse button pressed.
You will immediately see how the mixed color changes in the corresponding color field in the last column.

Alternatively you can also enter a numeric value by keyboard: simply click on the color field, type in the number and confirm with *Shift/Return*.

The `mix` color field will now display the respective ratio of the two colors.

In the corresponding color field in the last column (`bright`) you will see the resulting mixed color. `bright`

However, the column `bright` has more to offer. Here you can define the brightness of the mixed color. The procedure is the same as for mixing colors.

The default *value* is 1.00 meaning 100% brightness, 0 = dark.

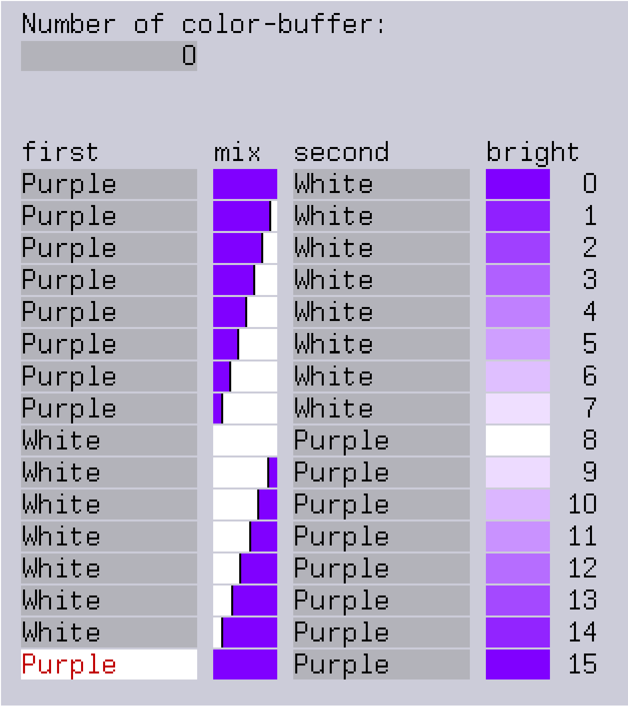
The **Lasergraph DSP** offers you two very effective tools called "REVERSE" (F6) and "`MIX`" (F7). REVERSE `MIX`
You will find these tools in the *menu bar* at the bottom as soon as you have clicked on one of the colors in the *color buffer*.

The REVERSE command changes the order of the colors:

```
REVERSE [<from> <to>]
```

If REVERSE is called without parameters, the order of all 16 colors is reversed. Alternatively, the order of colors in a specific range can be reversed by specifying the optional parameters `<from>` and `<to>`.

With the `MIX` command you can effortlessly achieve perfect color and brightness gradients.



Number of color-buffer:
0

first	mix	second	bright	
Purple	■	White	■	0
Purple	▨	White	■	1
Purple	▨	White	■	2
Purple	▨	White	■	3
Purple	▨	White	■	4
Purple	▨	White	■	5
Purple	▨	White	■	6
Purple	▨	White	■	7
White	□	Purple	□	8
White	▨	Purple	■	9
White	▨	Purple	■	10
White	▨	Purple	■	11
White	▨	Purple	■	12
White	▨	Purple	■	13
White	▨	Purple	■	14
Purple	■	Purple	■	15

To program a color gradient click on the first line (0) of the first column and in the `Primary colors` window click on `Purple`.

Then click on `MIX` in the *menu bar* at the bottom or press `F7`. Now you have to enter the color numbers at which the gradient is to start and end.

The *menu bar* already offers a few color numbers which divide the *color buffer* into quarters.

In the example printed here `Purple` runs once from 0 - 8 and once from 8 - 15.

Either you click on the corresponding numbers in the *menu bar* or press the *function keys* shown underneath or you enter the numbers with spaces between them and confirm with *Return*. For the second `Purple` gradient click on the last line in the first column and again click on `Purple` in the `Primary colors` window.

Then enter a `MIX` from 0 - 8 as described above.
Handle the white gradient accordingly.

The basic principle is: After choosing the option `MIX` use the numbers 0 – 15 to choose the beginning and the ending of the mix.

Grades in the brightness are programmed accordingly.

bright

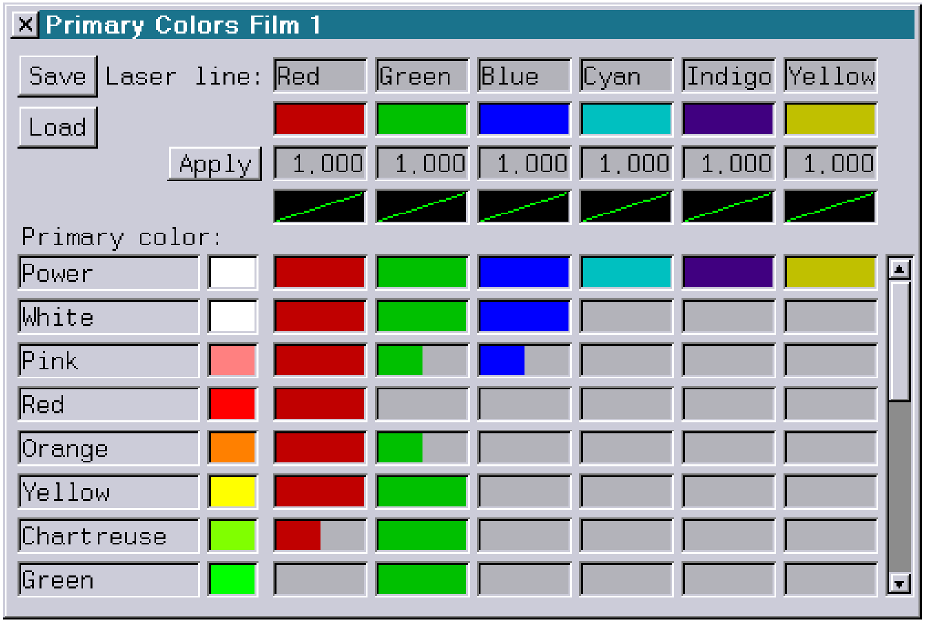
In the column bright select the line which is to have the least brightness, set the desired percentage number by *mouse poti* - and continue as before.

These color and brightness gradients are especially effective in connection with the ColorCYcle command!

Now you know how to create mixed colors from two basic colors in the SetColorBuffer.

However, you can even mix or change primary colors and give them an individual name.

For this click on the small black triangle in the upper right corner of the Primary colors window and you will see the following window:



Laser line:		Red	Green	Blue	Cyan	Indigo	Yellow
		🟥	🟩	🟦	🟦	🟪	🟨
		1.000	1.000	1.000	1.000	1.000	1.000
Primary color:							
Power	⬜	🟥	🟩	🟦	🟦	🟪	🟨
White	⬜	🟥	🟩	🟦			
Pink	🟥	🟥	🟩	🟦			
Red	🟥	🟥					
Orange	🟧	🟥	🟩				
Yellow	🟨	🟥	🟩				
Chartreuse	🟩	🟥	🟩				
Green	🟩		🟩				

It is important to know that the color you see on the monitor is almost never exactly identical with the color emitted by the laser. The intensity of the individual color lines emitted by the laser varies from manufacturer to manufacturer and from model to model.

← Primary colors

The six columns "Red", "Green", "Blue", "Cyan", "Indigo" and "Yellow" represent the six laser color lines and serve to adjust the laser colors to the monitor colors. Please see the chapter about Advanced Parameters, Primary Colors for more information.

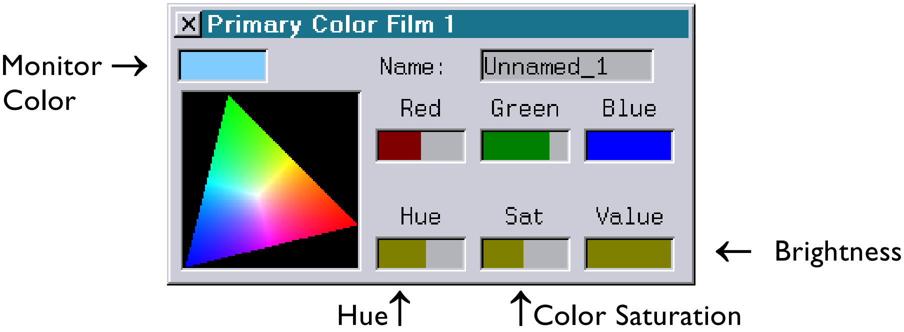
Here we will only discuss how the 16 basic monitor colors can be changed and renamed and how new colors can be added.

If you want to add a new color of your own to the basic colors *scroll* all the way through the basic colors in the `Primary colors` window.
At the end you will then see a black color field without a name.

Monitor Color Selection Window

When you click on it the following new window will open, the *monitor color selection window.*

In our example window a new color has already been mixed but no name has been entered yet.


Primary Color Film 1

Name: Unnamed_1

With the help of this window you can very easily select colors and change them in the finest nuances.

Simply click on the desired color in the color triangle. As soon as you have clicked on a color it is transferred to the *monitor color field* above the color triangle.

You can move the mouse pointer over the triangle with pressed left mouse button and release the mouse button when you have found the right color.

`Red`
`Green`
`Blue`

The mixing ratio of the selected color is immediately displayed in the `Red Green Blue` fields.

The scale ranges from 0 - 1.

The three fields at the bottom have the following meanings:

`Hue`
`Sat`
`Value`

`Hue:` hue of the color in the triangle,
`Sat:` color saturation,
`Value:` brightness.

The values of all six fields can be changed by *mouse poti* or by keyboard entry.

When you click on one of the fields a value between 0 and 1 appears, with the exception of the `Hue` field with a scale from 0 - 360 (degree value equivalent to the position on a color circle).

By changing these values you can fine-adjust your color or, as an alternative to the color triangle, define a new color.

When you are satisfied with the result click on the field Unnamed which is then marked by its color and enter the name you want your creation to have.
The name and the monitor color are now added to the basic colors in the Primary colors window.

However, this color is not yet displayed by your laser. As you can see the six fields next to the monitor color are all empty.

Using these fields you now mix the six color lines in the correct ratio until the laser color is equivalent to the new monitor color.

For this it may help to enter the RGB values displayed in the *monitor color selection window* into the Primary colors window and to then fine-adjust and if necessary use the other three lines additionally.

If you just want to slightly change an existing color or to rename it, click on the corresponding color field. The values of this color and it's name will then immediately appear in the Primary colors window where you can then change nuances and the name as described above.

The altered color table is automatically saved along with the trickfilm, it therefore is part of this specific trickfilm.

If you want to use the same color table for different films you can also save it as an individual file.

Primary colors
save

For this click on save in the Primary colors window and give the new color table an adequate name.

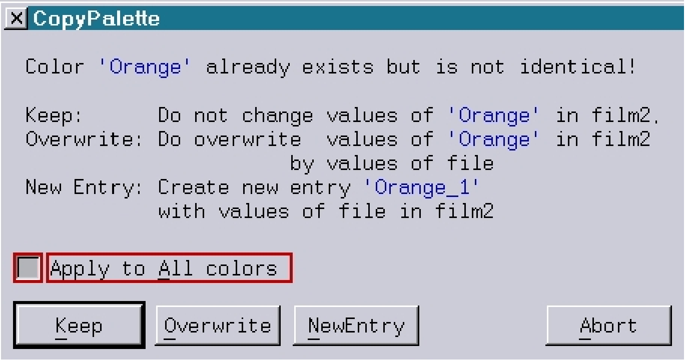
This way the color table is saved as an individual file and is displayed in the File window as <name.COL>.
This has the advantage that you can easily load it and copy it to different films.

You can load saved color tables by clicking on load in the Primary colors window. The File window will then open automatically.

Primary colors
load

After you have clicked on the adequate color table and confirmed the load command, the computer will compare the settings of the basic colors of the current color table with the one you want to load.

If one or more settings are different the following window called *CopyPalette* will open:



```
CopyPalette

Color 'Orange' already exists but is not identical!

Keep:      Do not change values of 'Orange' in film2.
Overwrite: Do overwrite  values of 'Orange' in film2
                     by values of file
New Entry: Create new entry 'Orange_1'
           with values of file in film2
```

☐ Apply to All colors

For each differing color with the same name you will be asked whether you want to keep or overwrite the current settings or if you want to keep both.

Keep

To keep the current settings click on Keep.

Overwrite

If you want to replace the old settings with the new ones click on Overwrite.

NewEntry

For keeping both versions click on NewEntry. The new setting of the respective color will be added to the color table under the same name with the addition of _1, or if already existing, _2 and so on.

If there are more than one differing settings of colors with the same name, the computer will jump to the next color after each decision.

Apply to All Colors

You can now decide for each color individually or click on Apply to All colors beforehand if you want to keep or overwrite all settings or keep all double settings.
Then your decision will affect all identically named colors with differing settings.

Abort

With a click on Abort you cancel the loading process

Displayed command line in the film:
SetColorBuffer with the *color buffer* number (0 - 7)
Position in the program: Before the time loop
Keyboard shortcut: scb

3.4.2 CopyColorBuffer

With the **CopyColorBuffer** command you copy the *color buffer* which you have set before into the *palette* used by your graphic.

→ 3.4.1. SetColorBuffer

Without the CopyColorBuffer command the SetColorBuffer command is ineffective.
The only exception is the ColorCrossFade command which replaces the CopyColorBuffer command.
With the CopyColorBuffer command you assign one of the eight possible *color buffers* to one of the four possible *palettes*.

→ Chapter 3.4.3 Color CrossFade

You can use four *palettes* with 16 colors each for a graphic. Each individual graphic can therefore consist of a total of 64 colors.

In the *display window* you can enter the following parameters:

Brightness:	0 - 1 or *variable* or *curve*

Here you can influence the brightness of the projection.
Brightness: 0 = dark, 1 = full power

You can also fade the picture in or out with a *curve* or a *variable* or make it flash within the time loop.

→ *Curve Editor*

→ Chapter 3.12 *Variables*

Number of color-buffer:	0 - 7

Here you enter the number of the *color buffer* which is to be copied into the *palette* stated below.

Number of palette:	0 - 3

Here you enter the number of the *palette* for which the *color buffer* stated above is to be effective.

Displayed command line in the film:
CopyColorBuffer with *palette* number (0-3)
Position in the program:
Within the time loop, before the picture output
Keyboard shortcut: ccb

3.4.3 ColorCrossFade

With the `ColorCrossFade` command you can program a continuous color change.

This command copies two preciously set *color buffers* into the *palette* used by the respective graphic and allows you to mix the colors of both *color buffers*.

It therefore already executes the function of the `CopyColorBuffer` command so that you do not have to set it additionally.

Imagine you want to program a sunset.
You set two *color buffers*. The first one colors your sun yellow, the second one red.

With the ColorCrossFade command you achieve a soft change from the first to the second color *buffer* and your sun turns red without interruption.

For this enter the following in the *display window*:

`Portion of buffer 2:`

Here you specify how and if a transition from `buffer 1` to `buffer 2` is executed.

For one you have the choice from 0 to 1, i.e. any influence of `color buffer 2` on the displayed color in percent is possible.

`0` means `color buffer 2` has no part in the color display, your sun would only appear yellow.

`1` means `color buffer 2` has 100% portion, your sun would be displayed only in red.

`0.5` would result in a 50% mix of color buffer 1 and color buffer 2, your sun would look orange.

For what we want to do we have to enter a *variable* or a *curve*.

→ *Curve Editor*

→ Chapter 3.12 *Variables*

When you click on the *input field* next to `Portion of buffer 2` the *menu bar* at the bottom will offer several options. We will not discuss all of these in detail here, please read the chapters about *variables* and *curves* mentioned in the *margin column*.

Here and now only this:

For the sunset you need an even color change. You can achieve this for instance by entering the offered *variable*: `LoopCnt`.

Simply click on `LoopCnt` at the bottom or press `F4`.

Then you can influence the brightness.

`Brightness:`

The input possibilities for this parameter are the same as described before.

For one you can set the level of brightness on a scale between 0 and 1 (0 = dark, 1 = full power), or you can fade the picture in and/or out with a *variable* or a *curve*.

→ *Curve Editor*

→ Chapter 3.12 *Variables*

Of the *variables* offered in the *menu bar* at the bottom, `LoopCnt` will for instance fade the picture in continuously, with `DECAY0()` you fade it out.

`Number of color-buffer 1:`

Here you enter the *color buffer* which is to be used for the starting color.
So if `color buffer 0` colors your sun yellow, enter a `0` here.

`Number of color-buffer 2:`

Here you enter the number of the *color buffer* which is responsible for the final color.

For the sunset you would have to enter a 1 provided that `color buffer 1` colors your sun red.

Should you decide to program a sunrise instead, simply enter the *source buffers* in reverse order and the sun will change color from red to yellow.

`Number of palette:`

Here you specify in which *palette* the color transition is to be saved.

Displayed command line in the film:
`ColorCrossFade` with *palette* number (0-3)
Position in the program:
Within the time loop, before the picture output
Keyboard shortcut: `ccf`

3.4.4 LogoShutter

There are systems where the brightness of the displayed picture can be adjusted additionally by a special output pin, the so called `LogoShutter` (the name is historic and comes from a time when this was a mechanically shutter which could be adjusted).

Using the `LogoShutter` will not make sense for most users because only few systems utilize the `LogoShutter` signal.

The function of the `LogoShutter` is very similar to that of an analog output. It is possible to set constant values as well as fades.

For DSPs which do not support the `LogoShutter` there is a possibility to simulate it using an analog (adjustment through "`AdvancedParameters ⇨ DisplayParameters ⇨ Logoshutter as analog`").

`From` defines the start value

`To` defines the end value

`Time` defines how long the fade `From` / `To` will take (if used outside a loop).

You can use the command `LogoShutter` in a timeloop with a curve, too. Then insert a curve into the parameter `To` instead of a value.

Displayed command line in the film:
`LogoShutter`
Position in the program:
Outside the time loop with fixed values, inside the loop with a curve
Keyboard shortcut: `ls`

3.4.5 ColorCYcle

With the `ColorCYcle` command you can program color cycles, for instance to simulate movement in static pictures.
You can also add colors to pictures digitized with one color.

You have influence on the following parameters:

`Number of color segments` (scale from 1-64)

Here you enter how many colors from the respective *palette* should be used.
If you set only one *palette* you can enter a maximum of 16 colors.
However, you can also set four *color buffers*, copy them into the four possible *palettes* with the command `CopyColorBuffer` and then display a graphic with 64 colors.

This is especially effective if you work with color and/or brightness gradients because that way you can achieve soft, flowing transitions.

`Length of color segments:`

Here you define the length of the individual color segments.

The best way is to click into the *input* field and use the *mouse poti* to find out which length looks best.
The value depends on the number of selected colors and the complexity of the graphic (points, blanking and repeats) and on the effect you want to achieve.

`Phase:`

In this field you specify if and if yes then how the colors are to cycle.

→ *Curve Editor*

You can define the direction and speed of the color cycle with *variables* or *curves*.

→ Chapter 3.12 `Variables`

Displayed command line in the film:
`ColorCYcle`
Position in the program:
Within the time loop, after `CopyColorBuffer`
Keyboard shortcut: `ccy`

3.4.6 SetPaletteEntry

With the `SetPaletteEntry` command you can define the colors in a palette without using the "fixed" values from the primary color table.

You define the ratio of the six laser lines within the time loop for a specific color number. The difference to the commands `SetColorBuffer` and `CopyColorBuffer` is, here you can use variables, e.g. a sinewave or a fader, instead of fixed values.

If for instance you digitized a face with blue eyes which have the color number 11 and red lips with the color number 4, you can make the blue eyes smoothly turn green or the red lips violet - or if you want to program a horror film, turn the blue eyes red and the red lips green.

When you click on the trickfilm line `SetPaletteEntry` you will see the following input options in the *display window*.

`Number of palette:`

Enter the corresponding *palette* number (0-3).

`Index of entry:`

Here you enter the color number with which the part of the picture which you want to recolor was digitized.

For the example mentioned above you have to enter 11 here to change the color of the eyes or 4 to change the color of the lips.

`Red:`

In this field you define the fraction of the red line.

If you want to color the red lips green you have to reduce the red component from 1 to 0 using a *curve* or a *variable*.

`Green:`

Here the green fraction of the projection is defined.

In our example you would have to set the green component from 0 to 1 with a *curve* or a *variable*.

`Blue:`

In this *input* field you determine the fraction of the blue line.

In the lip example you can set this parameter to 0.

C4: C5: C6:

These three parameters stand for the remaining three laser lines. This is only of interest for you if your laser supports more than three lines.

The colors are not specified by name because they depend on the laser you use.

The `SetPaletteEntry` command ignores all settings in the primary color table, therefore you cannot change the color balance for all effects that uses this command. ☜

The values set here for red, green and blue will also be used as screen colors in the preview.
In case you want to set the screen colors separately, you can use the command `SetPaletteSEntry`, see description below.

Displayed command line in the film:
`SetPaletteEntry`
Position in the program:
Within the time loop, after the time loop or SetVariables
Keyboard shortcut: `spe`

3.4.7 SetPaletteSEntry

With the `SetPaletteSEntry` command you can define the screen colors used by the Preview Display for the specified palette entry. This command does not affect the laser output.

Note: Using the `SetPaletteEntry` command again after this command will overwrite the settings made here (see description of `SetPaletteEntry`).

`Number of palette:`

Enter the corresponding palette number (0-3).

`Index of entry:`

Here you enter the color number (0-15) within the specified palette.

`Red:`

In this field you define the fraction of the red line.

`Green:`

Here the green fraction is defined.

`Blue:`

Here you determine the fraction of the blue line.

Displayed command line in the film:
`SetPaletteSEntry`
Position in the program:
Within or outside the time loop
Keyboard shortcut: `spse`

3.4.8 SetDefaultColors

This command initializes all 16 colors of the specified palette using the following default colors:

	Red	Green	Blue	C4	C5	C6
White	1.00	1.00	1.00	0.00	0.00	0.00
Pink	1.00	0.50	0.50	0.00	0.00	0.00
MintGreen	0.50	1.00	0.50	0.00	0.00	0.00
PastelBlue	0.50	0.50	1.00	0.00	0.00	0.00
Red	1.00	0.00	0.00	0.00	0.00	0.00
Orange	1.00	0.50	0.00	0.00	0.00	0.00
Yellow	1.00	1.00	0.00	0.00	0.00	0.00
Chartreuse	0.50	1.00	0.00	0.00	0.00	0.00
Green	0.00	1.00	0.00	0.00	0.00	0.00
Turquoise	0.00	1.00	0.50	0.00	0.00	0.00
Cyan	0.00	1.00	1.00	0.00	0.00	0.00
BabyBlue	0.00	0.50	1.00	0.00	0.00	0.00
Blue	0.00	0.00	1.00	0.00	0.00	0.00
Purple	0.50	0.00	1.00	0.00	0.00	0.00
Magenta	1.00	0.00	1.00	0.00	0.00	0.00
HotPink	1.00	0.00	0.50	0.00	0.00	0.00

Similar to the command `SetPaletteEntry` the command `SetDefaultColors` does not take the primary color table into account.

`Brightness:` 0.0 - 1.0, *variable* or *curve*

This parameter can be used to change the intensity of the colors written into the palette.
Value ranges from 0.0 (off) to 1.0 (full on).

`Number of palette:` 0 - 3

Number of the palette to be initialized.

Displayed command line in the film:
`SetDefaultColors` with *palette* number (0-3)
Position in the program:
Within or outside the loop, before the picture output
Keyboard shortcut: `sdc`

3.4.9 COlorControl

This command is used when true color support is needed.

`Intensity:`

Value: 0 – 2, default: 1

Value 0: the picture will be displayed with no intensity (black).
Value 1: each color will be displayed with its original intensity.
Value 2: all colors will be displayed with maximum intensity.

`Saturation:`

Value: 0 – 2, default: 2

Value 0: no saturation is used, colors and saturation of the picture will be ignored (picture will be white).
Value 1: each color will be displayed with its original saturation.
Value 2: maximum saturation is used, the original saturation of the picture will be ignored.

Displayed command line in the film:
`COlorControl`
Position in the program:
outside the time
Keyboard shortcut: `COC`

3.5 Display Control

3.5.1 SParkle

The `SParkle` command is a special effect with which you can make individual points of your graphic flash briefly.

To create the 'sparkling', different points are repeated during the picture projection as if you had increased the *repeats* of these points when generating the graphic.

You cannot influence which of these points sparkle but you can influence the frequency, the number of points and the intensity.

The `SParkle` command is only effective in connection with the `setSWC` command.

→ Chapter 3.5.2 `setSWC`

In order to define the `SParkle` command make the following settings in the `setSWC` command window:

```
(Pre)Set SWC mask:        AllInvisible
SWC use mode:             Sparkle
SWC record mode:          Disabled
SWC suppress display:     No
```

`Probability:` (scale from 0-1)

Here you influence the frequency and number of flashing points.
If you enter 0, the sparkling is turned off.
When you enter 1 all points used in the graphic flash simultaneously. (Only makes sense if the graphic only has a small number of points.)

`Repeat:` (scale from 1-65535)

Here you influence the intensity of the flashing points. The higher the value which you set here the more often the respective points will be repeated.

For both parameters the optimal value depends on the complexity of the graphic and the effect which you want to achieve in the end. Just try it!

Displayed command line in the film:
`SParkle`
Position in the program:
`SParkle` outside the time loop if you enter constant values, within the time loop if you enter *curves* or *variables*
`setSWC` command within the time loop after the Loop
Keyboard shortcut: `sp`

3.5.2 setSWC

The `setSWC` command makes different special effects possible which you can place specifically on previously defined parts of a graphic. With the `setSWC` command, also called *image masking*, you define a graphic as a mask which divides another graphic into 'visible' and 'invisible' areas.

The `setSWC` command is especially useful to create an impression of three-dimensional display by defining fore- and background by overlapping two or more pictures. The picture in front then serves as a mask which is either superimposed over parts of the picture in the background or through which parts of the otherwise covered picture are visible.

First you define the mask.

There are two ways to use the mask. Either everything covered by the mask is invisible (fig. SWC 1) or everything outside the mask is invisible (fig. SWC 2).

You define the type of mask in the first `setSWC` command, placed before the picture that is to determine the shape of the mask.

`setSWC` command before first graphic (square)

`setSWC` command before second graphic (DSP)

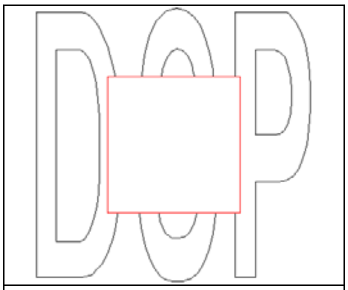
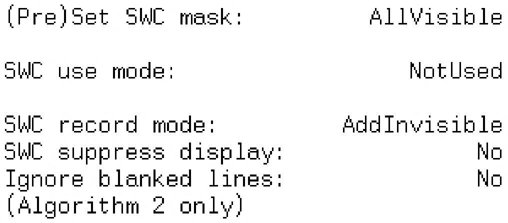
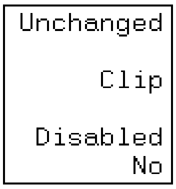


Fig. SWC 1



```
(Pre)Set SWC mask:          AllVisible

SWC use mode:                  NotUsed

SWC record mode:          AddInvisible
SWC suppress display:               No
Ignore blanked lines:               No
(Algorithm 2 only)
```



```
Unchanged

     Clip

 Disabled
       No
```

After you have defined the mask, in the second `setSWC` command you define what is to happen to the picture in the so-called invisible areas.

You set this command before position and output of the picture on which the mask (or the masks) is (are) to have an effect.

'Invisible' (`AllInvisible`) in this case has to be understood as a definition for the area in which the effect you selected is executed, it actually is not always invisible. Do not let the term confuse you!

!!!

You can either make the area actually invisible (Clip) as shown in the example above (fig. SWC 1), let it sparkle (Sparkle) or change colors (SwapColors). Additionally you can choose if the picture which defines the mask is to be visible or invisible as in fig. SWC 3 (→ SWC suppress display).

→ Chapter 3.5.1 SParkle

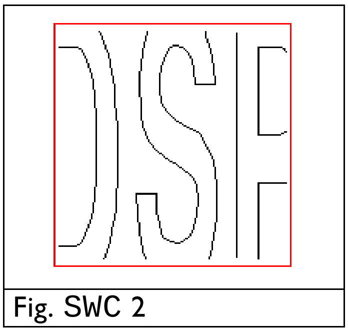


Fig. SWC 2

	setSWC command before first graphic (square)	setSWC command before second graphic (DSP)
(Pre)Set SWC mask:	AllInvisible	Unchanged
SWC use mode:	NotUsed	Clip
SWC record mode:	AddVisible	Disabled
SWC suppress display:	No	No
Ignore blanked lines: (Algorithm 2 only)	No	

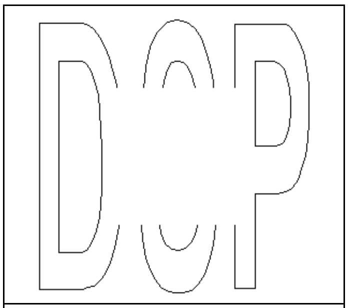


Fig. SWC 3

	setSWC command before first graphic (square)	setSWC command before second graphic (DSP)
(Pre)Set SWC mask:	AllInvisible	Unchanged
SWC use mode:	NotUsed	Clip
SWC record mode:	AddVisible	Disabled
SWC suppress display:	Yes	No
Ignore blanked lines: (Algorithm 2 only)	No	

Now for the input options in the display window of the setSWC command in detail:

SWC Algorithm: Unchanged/1/1+/2

SWC Algorithm

Algorithm 2 is the default setting, it is the best one.

Algorithm 1+ and 1 are only needed when producing a show for "old" Lasergraph DSP systems.

Unchanged means that the algorithm which was chosen last is not changed.

(Pre) Set SWC mask:

(Pre) Set SWC mask

Here you define the properties of your mask.

If the lower picture is to be covered by the upper one(s):
`AllVisible`

AllVisible

If you want to cut holes into the mask:
`AllInvisible`

AllInvisible

If you do not want to change the current setting:
`Unchanged`

Unchanged

`Inverted` inverts the current setting.

Inverted

SWC use mode:

With this parameter you activate the defined mask by specifying what is to happen in the 'invisible' area:

With `Clip` you suppress the projection of the areas defined as 'invisible'. This is the only case where these areas really do become invisible.

Clip

This way you can achieve an effect of optical depth, whether you position two pictures over each other or define a section where one picture becomes visible behind another (e.g. a glance or a flight through a keyhole).

`Sparkle` makes individual points flash brightly in the 'invisible' areas. You can control frequency and intensity with the additional command `SParkle`.

Sparkle

→ Chapter 3.5.1 SParkle

`SwapColors` causes a color exchange in the areas defined as 'invisible'.
To achieve this the color numbers used in the picture from *palette* 0 are exchanged with the respective color numbers from *palette* 2 and those from *palette* 1 with those from *palette* 3

SwapColors

Let us assume that your picture was digitized with color number 2 from *palette* 0, the default for this is pink. You now want a certain area of the picture, which you of course have to define as 'invisible' with a suitable mask, to be projected in blue.

→ Chapter 3.4.1 SetColorBuffer

You now set two *color buffers*. In the first one color number 2 stays pink. In the second one you make color number 2 blue. Then you copy the first *buffer* into *palette* 0 and the second one into *palette* 2.

→ Chapter 3.4.2 CopyColorBuffer

If you now enter the `SWC use mode: SwapColors` the 'invisible' area will be displayed in blue.

`NotUsed`

`NotUsed` is the default setting. The mask is inactive in this setting.
You can keep this default setting as long as you do not want to use the mask (yet).

💣

As soon as you have selected another setting, however, this setting will be kept until it is revoked by another `setSWC` command.

→ Chapter 3.2.3 `Subroutine`

You should therefore make it a habit to set a `setSWC` command with the default settings in your *subroutine* and to call up the *subroutine* with each new *entry*.

→ Chapter 3.2.6 `GoSub`

Otherwise you can have gaps or color faults which you never programmed appear in pictures when calling up different *entries* in variable order.

SWC record mode:

Here you influence the mask defined in the `(Pre) Set SWC mask` and introduce the picture following this `setSWC` command as a kind of stencil.

`AddInvisible`

If you decided in the `(Pre) Set SWC mask` to make the rear picture completely visible, entering `AllVisible`, usually the only thing that makes sense here is to enter `AddInvisible`. Otherwise the effect will not be very spectacular because nothing will happen.

The three possible effects which you can choose in the `use` mode are only executed in the areas defined as `'invisible'`.

`AddVisible`

On the other hand, if you defined the rear picture as invisible in the `(Pre) Set SWC mask` entering `AllInvisible` you should enter `AddVisible` here. You thereby cut a hole in the form of the following picture into the mask through which you can see the picture behind. (Fig. SWC 2)

`Disabled`

With the default setting `Disabled` the mask you defined before stays unchanged.

SWC suppress display: No/ Yes SWC suppress display

With this parameter you can turn off the display of the picture which defines the form of the mask (Fig. SWC 3).

The default setting is No, the picture is visible.

If you enter Yes the picture output will be suppressed.
This parameter always affects the output of the picture which follows this setSWC command.

Remember the following as a rule of thumb: Suppress display: Yes only makes sense if the record mode is not set to Disabled because this is usually only the case for the last setSWC command of the sequence which is set before the picture which you want to modify using one or more masks and where you should in any case define a use mode.

So if you enter Suppress display: Yes here, output of the picture for which you did all the work would be suppressed.

Ignore blanked lines: No/ Yes Ignore blanked lines

The default setting is No so the blanked lines in the mask will be used for masking, too

If the setting is Yes the blanked lines will be ignored for masking.

Displayed command line in the film:
setSWC
Position in the program:
Within the time loop, before InitPosition and the picture output
Keyboard shortcut: SWC

3.5.3 SetClipping

With this command you can limit a graphic horizontally as well as vertically.

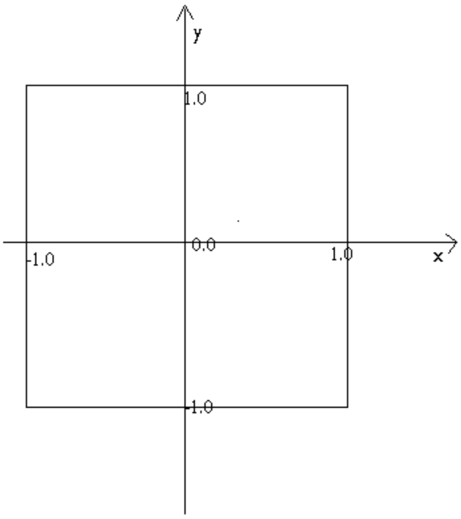
All picture elements located outside this imaginary border line will be 'cut off', they will no longer be visible.

The default setting is:

```
Clip top:      1.000000
Clip left:    -1.000000
Clip bottom:  -1.000000
Clip right:    1.000000
```

With this setting the whole picture is visible.

This is easy to understand if you imagine your picture in a square within a coordinate system. The origin of the coordinate plane is the center of the square and thereby also of the picture.



If you want only the right half of the picture to be visible set the left clipping edge (`Clip left`) to 0.000000. The other parameters keep their default settings.

It is a very nice effect to 'wipe away' the picture.

→ *Curve Editor*

→ Chapter 3.12.1 `SetVariables`

You can achieve this either with a *variable* or with a *Curve*. If for instance you want to wipe away the picture from top to bottom, click on the input field of `Clip top`, select `~<curve>()` from the *menu bar* at the bottom and define a *curve* from 1.0 to -1.0.

It is also very effective to let a second picture appear behind the first one.

Suppose you want to create the impression of wiping the first picture away from left to right while the second picture appears from behind it:

Set a *curve* from -1.0 to 1.0 in the input field next to `Clip left` under `SetClipping` for the first picture.

Under `SetClipping` for the second picture also set a *curve* from -1.0 to 1.0 in the input field next to `Clip right`.

For the input activate the desired field by mouse click and enter a value by keyboard or by *mouse poti* or set a *variable* or a *curve* (selection from the *menu bar* at the bottom).

Predefined *variables* such as LoopCnt, ATTACK0() or DECAY0() cannot be used in this case because they count from 0 - 1 while we have a scale from -1 to 1 here. ☜

`Clip top:`

Here you enter the limit for the upper part of the graphic on a scale from -1.0 to 1.0.
-1.0 : the picture is invisible.
0.0 : reduces the picture to the lower half.
1.0 : the complete picture is visible.

`Clip left:`

Here you enter the limit for the left part of the graphic on a scale from -1.0 to 1.0.
-1.0 : the complete picture is visible.
0.0 : reduces the picture to the right half.
1.0 : the picture is invisible.

`Clip bottom:`

Here you enter the limit for the lower part of the graphic on a scale from -1.0 to 1.0.
-1.0 : the complete picture is visible.
0.0 : reduces the picture to the upper half.
1.0 : the picture is invisible.

`Clip right:`

Here you enter the limit for the right part of the graphic on a scale from -1.0 to 1.0.
-1.0 : the picture is invisible.
0.0 : reduces the picture to the left half.
1.0 : the complete picture is visible.

`X & Y` second input field next to `Clip top`:

Here you can set the limits for the upper and the left part of the graphic simultaneously by *mouse poti*.

`X & Y` second input field next to `Clip bottom`:

Here you can set the limits for the lower and the right part of the graphic simultaneously by *mouse poti*.

Displayed command line in the film:
`SetClipping`
Position in the program:
Within the time loop, before `InitPosition` and the picture output
Keyboard shortcut: `SC`

3.5.4 SetSoftBlanking

The `SetSoftBlanking` command allows you to let lines run out smoothly.

To achieve this effect you have to digitize the graphics accordingly resp. for existing pictures change the *repeats* in the *picture editor*. Normally Blanking ON/OFF points have a repeat of 1.

→ *Picture Editor* `Piced`

The definition for a softblanking point is a repeat of 0.

With the `SetSoftBlanking` command you only influence points whose *repeats* are set to 0 on the blank change.

With the `SetSoftBlanking` command you determine how fast the blanking is turned on or off.

Of course you can also manipulate both values if the picture was digitized accordingly.

`Delay down:`

Here you determine how fast the *blanking* is turned on.
In this case the *repeats* of the BlankOn points have to be set to 0 in the graphic.

Scale from 0.0 to 1.0

`Delay up:`

Here you determine how fast the *blanking* is turned off.
In this case the *repeats* of the BlankOff points have to be set to 0 in the graphic.

Scale from 0.0 to 1.0

Displayed command line in the film:
`SetSoftBlanking`
Position in the program:
Within the time loop, before `InitPosition` and the picture output
Keyboard shortcut: `ssb`

3.5.5 SetBeamBrush

If your projector is equipped with appropriate hardware, such as LA.brush, you can use the `SetBeamBrush` command to define the desired line width.

Beam Width:

The parameter "Beam Width" must be in range from 0.0 to 1.0. Where 0.0 is the smallest possible line width and 1.0 is the maximum possible line width.

Wait:

As the optical system requires a certain time until the desired line width has been reached, you can use the parameter "Wait" to specify a time to wait before starting with the next picture.

Notes:

- Both parameters can also contain curves or variables.
- The line width set with the command `SetBeamBrush` remains valid for all subsequent pictures until it is changed by another `SetBeamBrush`.

  To have lines with different widths within a scene, you can achieve this by using several display commands.

  Example:

  ```
  Loop
    SetBeamBrush
    DisplayPicture
    SetBeamBrush
    DisplayPicture
  LoopEnd
  ```

Displayed command line in the film:
`SetBeamBrush`
Position in the program:
Before the display command
Keyboard shortcut: `sbb`

3.5.6 setDePthCueing

With this command you can give your pictures an impression of spatial depth by making points which are located farther away from the viewer on the z-axis run out darkly and so make them optically disappear in the distance.

For the three possible settings you can enter the numeric values either by keyboard or by *mouse poti*.

`Distance:`

Here you determine the distance between the viewer and the point on the z-axis where the points are supposed to start getting darker. Along this distance all points still have 100% brightness.

`Depth:`

Here you define the distance between the point on the z-axis you specified in the previous parameter and the point where the lowest brightness value is to be reached.

`LowLevel:`

Here you enter the brightness value which is to be reached at the point farthest away from the viewer. The computer here assumes a point which goes beyond the sum of the values of `Distance` and `Depth`.

The scale reaches from 0.0 to 1.0
1.0 is equivalent to 100% brightness, i.e. the effect is 'turned off'
0.0 = black

Here is a tip how you can set the parameters most easily:
First set `Depth` and `LowLevel` to 0.0.
Now specify the `Distance` up to the point where the projection is to get darker. The easiest way is to drag the *mouse poti* to the desired point.
Then set the `Depth` in the same way.
Finally you set the desired lowest brightness value for the farthest point (`LowLevel`).

Displayed command line in the film:
`setDePthCueing`
Position in the program:
Within the time loop, before `InitPosition` and the picture output
Keyboard shortcut: `dpc`

3.5.7 setXYCueing

This command is generally equivalent to the command `setDePthCueing` with the difference that here you can fade out points on the x/y axis.

You can influence six parameters.

Length:

Here you determine the length of the stretch along which the brightness slowly decreases.

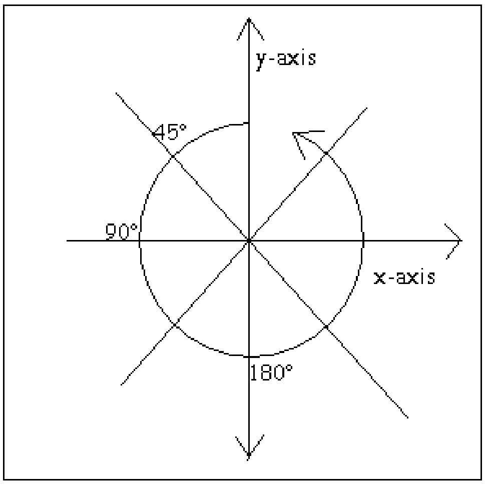
LowLevel:

Here you enter the brightness value which is to be reached at the darkest point.

The scale reaches from 0.0 to 1.0
1.0 is equivalent to 100% brightness, meaning the effect is 'turned off'.

RotZ:

With this parameter you determine the shadow direction.
The value you enter denotes the angle of the shadow axis.
Sounds complicated, but it is really simple - see the graphic and the examples:



0.0 darkens the picture vertically, i.e. upwards on the y- axis,
180.0 darkens it vertically downwards,
90.0 throws the shadow along the x-axis, i.e. horizontally to the left,
270.0 shades the picture horizontally to the right.

MoveX:

Here you move the shadow axis along the x-axis, i.e. to the right or left.

MoveY:

Here you define the vertical position of the shadow axis (upwards or downwards).

Offset:

With the `Offset` you move the shadow axis in the direction which you defined with `RotZ`.

Displayed command line in the film:
`setXYCueing`
Position in the program:
Within the time loop, before `InitPosition` and the picture output
Keyboard shortcut: `xyc`

3.5.8 Pattern

With the Pattern command you modulate the brightness of a projection.

Similar to the way you can influence colors with the ColorCYcle command you can influence the brightness with the Pattern command.

This effect is cannot be seen on the Preview.

☜

Frequency

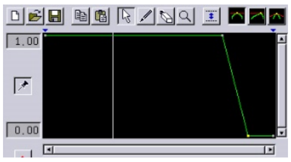
Frequency

With the Frequency you determine how often the *pattern* is displayed within the picture. However, the result depends on the number of points in the picture.

In general one can say: the higher the Frequency the more often the brightness gradient will be applied to the graphic.

!!! Points

As mentioned before the result depends on the number of picture points in the graphic and the number you enter under Points. The actual number of points of a graphic includes the number of *repeats* as well as all interpolations.



In our example fig. Pattern 1-4 we have a circle of exactly 100 picture points (including *repeats and realtime interpolation*) with this curve in Pattern 1.
To achieve the result in fig. 1 we entered 100 for Points and 1 for Frequency.

For fig. 2 we set the frequency to 2, the rest stays unchanged. As we can see the *pattern* is now shown twice in the graphic.
We would, however, have achieved the same result if we had left the Frequency at 1 and set the Points from 100 to 50 (further information under parameter Points).

→ Points

Frequency **1** Points 100 Picture Points	Frequency **2** Points 100 Picture Points	Frequency **4** Points 100 Picture Points	Frequency **16** Points 100 Picture Points

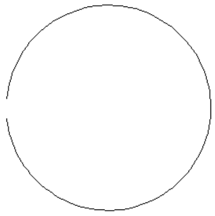
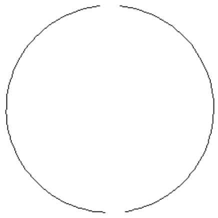
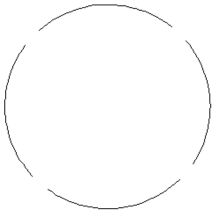
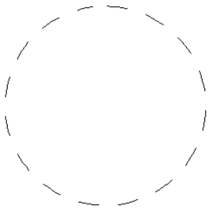

			

fig. Pattern 1	fig. Pattern 2	fig. Pattern 3	fig. Pattern 4

If you apply the Pattern command to Abstracts the frequency, as for all modulators, refers to the base frequency.

For instance, if you generate a circle with two sine functions offset by 90° with a frequency of 1 each, a *pattern* with a frequency 1 will be displayed exactly once within the circle, twice with a frequency of 2 etc.

Phase

Phase

→ *Curve Editor*

Like in the ColorCYcle command, here you can

- either define a specific starting point within the course of the pattern by entering a value between -1 and 1 or
- let the modulation run through the picture using a *curve* or a *variable* such as LoopCnt.

Points

Points

With the number of Points you determine how often the *curve* you enter for Pattern1 and / or Pattern2 is scanned and onto how many picture points the pattern is distributed.

Let us assume you enter Points 3. This would mean that the *curve* of your *pattern* is divided into three sections (see fig. Pattern 5). The *curve* will then only be scanned at three points, namely at 0, 0.33 and 0.66, which will have the effect that important parts of your *curve* will not be evaluated and therefore will not be visible (s. fig. Pattern 7).

In comparison fig. Pattern 8 shows the same *curve* with Points 20. As you can see the result would be much more faithful to the *curve*.

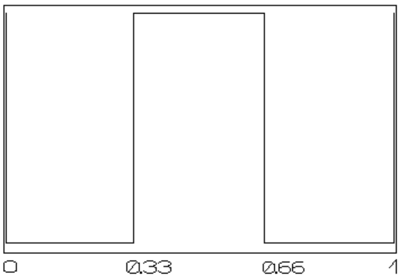


fig. Pattern 5

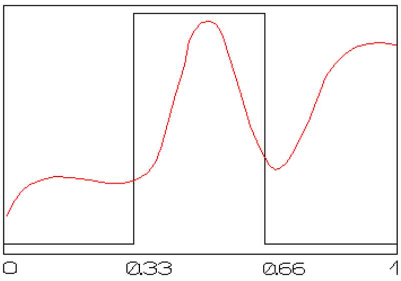


fig. Pattern 6

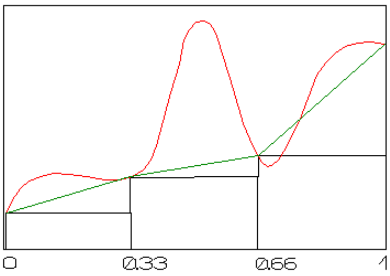


fig. Pattern 7

The more points you enter the finer the *curve* is scanned.

The more complex the *curve* is the more points you should enter to ensure a high resolution and the more simple the *curve* is the fewer points are necessary.

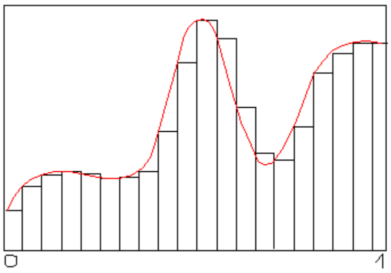


fig. Pattern 8

In addition the Points value determines onto how many picture points the *pattern* is distributed.

→ Frequency

So if we enter Points 100 as in our example fig. Pattern 1, the pattern is distributed to 100 picture points which is exactly the number of points in the circle. With a Frequency of 1 the pattern will be distributed on the graphic exactly once.

If we only enter 50 Points instead of 100, the *pattern* will be distributed to 50 picture points and then start over again. It would therefore run exactly twice through the circle of 100 points. With a Frequency of 1 the result would look like fig. Pattern 2.

So if at some time you want to increase the number of points to achieve a higher resolution for the scanning of the *pattern curve* while keeping the same number of bright- / dark spots you have to increase the Frequency by the same factor as the Points.

This results in the simple formula:

Frequency x actual number of picture points / Points = number of repetitions of the *pattern* in the graphic

For fig. Pattern 1 the formula will look like this:

1 x 100 : 100 = 1
or
2 x 50 : 100 = 1

For the result in fig. Pattern 3 there would for instance be the following possibilities:

4 x 100 : 100 = 4
8 x 100 : 200 = 4
1 x 100 : 25 = 4

!!!

An exception to this are the Abstracts.
When you use the Pattern command on Abstracts the input under Points only influences the scanning of the *curve*, not the number of repetitions in the Abstracts.

Of course with the *mouse poti* and a good eye for effects you can simply change the values for Frequency and Points until you like the result!

Fade

Fade

→ Chapter 3.4.3 ColorCrossFade

Fade is generally equivalent to the parameter Portion of buffer 2 in the ColorCrossFade with which you can fade between two *color buffers* defined by you.

In this case you can fade between two *patterns* which you define with the following parameters (Pattern 1 + 2).

You can enter constant values, *curves* or *variables*.

→ Chapter 3.12 SetVariables

0 = Pattern 1 is displayed
1 = Pattern 2 is displayed

For instance if you want a linear smooth transition from Pattern 1 to Pattern 2 enter the *variable* LoopCnt or define a straight *curve* from 0 to 1.

→ *Curve Editor*

Pattern 1

Here you define the *pattern*, i.e. you determine the type of modulation.

You can enter a constant value, a *variable* or a *curve*. However, constant values or *variables* such as the LoopCnt do not make much sense here. If you enter a 0 everything is dark, with 1 everything is evenly bright, LoopCnt would simply fade in the picture and after all you can do that with CopyColorBuffer.

→ Chapter 3.4.2 CopyColorBuffer

Pattern 2

With Pattern 2 you can define a second modulation if you want to change the displayed *pattern* dynamically with Fade. Otherwise: see Pattern 1.

Displayed command line in the film:
Pattern
Position in the program:
Within the time loop
Keyboard shortcut: p

3.6 Controlling Optical Bench and other Peripherals

3.6.1 ChaseDelay

This command allows you to switch different relays and/or analogs in regular intervals, i.e. to program effect cycles.

→ Chapter 3.6.2 ChaseSection

The different relay blocks are separated with the command ChaseSection. In general each ChaseSection command is equivalent to a Wait command whose waiting time can be set with the command ChaseDelay.
You could just as well enter:

```
  Set Relais
  Wait
  Set Relais
  Wait ...
```

However, if you then want to change the speed you have to change the time for every Wait, with ChaseDelay you have to change it only once.

!!!

Scanner output is not possible at the same time with this command.

Delay in seconds:

Delay in Seconds

Here you enter the time delay after which the effects are supposed to be switched. With 1.0 the effects will be switched every second, with 0.5 every half second and so on.

Speed multiplier:

Speed multiplier

The value you entered for the previous parameter is divided by the value entered here.

The default setting is 1.0, it does not influence the speed defined before. If, however, you enter 2.0 the cycle speed will be doubled:

Example:

```
  Delay in Seconds: 1.0000
  Speed multiplier: 2.0000
```

1 : 2 = 0.5 The change of effects will occur every half second.

Displayed command line in the film:
ChaseDelay and Delay in seconds
Position in the program:
Outside the time loop
Keyboard shortcut: cd

3.6.2 ChaseSection

→ Chapter 3.6.1 ChaseDelay

ChaseSection is an internal Wait command.
It is used together with ChaseDelay to separate the individual effect blocks.

After each effect block you have to set a ChaseSection command which will then wait before switching to the next effect for exactly the period of time which you defined for ChaseDelay.

A scanner output is not possible during this waiting period.

A chaser *entry* would then look like this:

```
Entry
ChaseDelay     Delay in Seconds:    0.3000
               Speed multiplier:    1.0000
TimeLoop
SetRelais
SetAnalog
ChaseSection   holds the block set before this command
               for 0.3 seconds before switching to the
               next block
SetRelais
SetAnalog
ChaseSection   holds the block set before this command
               for 0.3 seconds before switching to the
               next block
SetRelais
SetAnalog
ChaseSection   holds the block set before this command
               for 0.3 seconds
LoopEnd
```

Displayed command line in the film:
ChaseSection
Position in the program:
after a SetRelais and/or SetAnalogs block
Keyboard shortcut: cs

3.6.3 SetRelais

With `SetRelais` you can switch DMX outputs on or off digitally.

General Information about Relays and Analogs:

The **Lasergraph DSP** provides 256 DMX outputs. This number results from eight `channels` with 32 outputs each.
You can control digital as well as analog external devices with the **DSP** such as lamps, screens, fog machines and bench effects.

To avoid confusion you can and should mark the different outputs. In the *bank configuration* you can assign names to all used outputs.

You can control each of these outputs with the commands `SetRelais`, `SetAnalogs` and `SetAnalogSingle`.

→ Chapter 3.6.4/5
`SetAnalogs`
`SetAnalogSingle`

It has to be noted that an output switched on with `SetRelais` is always set to 100% even if you set the same output to 50% power with `SetAnalogs` afterwards.
However, if you switch this output off again with `SetRelais` it is not really turned off but will output the analog value.

💣

Sounds like programmed chaos and can lead to one, therefore make a habit of switching digital devices which only know the two states "on" or "off" exclusively with `SetRelais`.
All analog devices which for instance can be dimmed or whose speed can be controlled should be actuated with `SetAnalogs` or `SetAnalogSingle` - always!

👍

There are more potential errors that can be made.
You can distribute relay- and analogs programming to a total of four films, the `timescript` and the *keyboard configuration*.

Therefore it is best to program DMX Channels exclusivly in one Film or in the Timescript !

💣

A relay for instance counts as activated if it was turned on at any of the possibilities mentioned above - even if you turn it off somewhere else.

For the analogs the possible problems can be accurately described as follows:

- The last one wins -.

If for instance you set an output to 1.0 with `SetAnalogs` in the trickfilm and set the same one to 0.5 in the `timescript`, the value called up last will be output.

So decide:

Set the relays or analogs only at one point, **either** in the trickfilm, **or** in the `timescript`, **or** in the *keyboard configuration*.

Recommendation: In the trickfilm. The *entries* can be assigned to keyboard keys as well as called up in the `timescript`.

`Channel:`

`Channel`

Here you enter the number of the channel (1-8) containing the desired effects.

Underneath you see the 32 outputs belonging to the selected channel.

Each of these eight channels (1-8) is numbered from 1-32 resulting in a total of 256 outputs resp. DMX channels.

So if you use output 25 of channel 3 the number of the resulting DMX channel naturally is not 25 but 89.

You calculate:

DMX = (channel - 1) x 32 + output

Used on our specific example from above the formula would look like this: (3 - 1) x 32 + 25 **= 89**

You can derive the formulas for determining either the channel or the output accordingly:

Channel = Int (DMX : 32) + 1

For non-mathematicians: "Int" means that only the digits before the decimal point are of interest.

Our established example: Int (89 : 32) + 1 **= 3**

Output = DMX – (channel - 1) x 32

In our example: 89 – (3 – 1) x 32 **= 25**

The default setting is All Unchanged.
That means all relays you set in the previous *entry* are unchanged, they are taken over from *entry* to *entry* even though they are not noticeably turned on.

!!!

You should set relays which you set in the film but which you do not need anymore to Off in the next *entry* by clicking on it twice.

All OFF

All OFF

When you click on All OFF all relays will be turned off.

All ON

All ON

When you click on All ON all relays will be turned on.

All Unchanged

All Unchanged

Sets all analogs back to their original states.

Get actual state

Get actual state

This is similar to GetActPos for InitPosition.
However, with the SetRelais command it only works retroactively within one *entry*.

Displayed command line in the film:
SetRelais and Channel number
Position in the program:
Outside the time loop, exception: Chaser
Keyboard shortcut: sr

→ Chapter 3.6.1/2
ChaseDelay
ChaseSection

3.6.4 SetAnalogs

With `SetAnalogs` you can turn DMX channels on and off, like with `SetRelais`. The `SetAnalogs` command differs from `SetRelais` in so far as it not only outputs 0 and 1 but intermediate values as well. You can for instance dim a lamp or turn a rotary motor on with half or quarter speed.

→ Chapter 3.6.3 `SetRelais`

Please read 'General Information about Relays and Analogs' in the previous chapter concerning possible errors and their prevention in connection with this command.

'General Information about Relays and Analogs' 💣

`Channel:`

`Channel`

Here you enter the number of the channel (1-8) containing the desired effects.

Each of these eight channels (1-8) is numbered from 1-32 resulting in a total of 256 outputs resp. DMX channels.

So if you use output 25 of channel 3 the number of the resulting DMX channel naturally is not 25 but 89.

You calculate:

DMX = (channel - 1) x 32 + output

Used on our specific example from above the formula would look like this: (3 - 1) x 32 + 25 **= 89**

You can find derivatives of this formula in the previous chapter about `SetRelais`.

→ Chapter 3.6.3 SetRelais, Channel

The default setting is `All Unchanged`.

That means all analogs you set in the previous *entry* are unchanged, they are taken over from *entry* to *entry* even though they are not noticeably turned on.

You should set analogs which you set in the film but which you do not need anymore to 0 in the next *entry*. For this click on the corresponding field and press F3 or click on the 0.0 in the *menu bar* at the bottom.

`All 0.00`

`All 0.00`

When you click on `All 0` all analogs will be turned off.

All 1.00

All 1.00

When you click on `All 1` all analogs will be turned on undimmed.

All Unchanged

All Unchanged

Sets all analogs back to their original states.

Get actual state

Get actual state

This is similar to `GetActPos` for `InitPosition`. However, with the `SetAnalogs` command it only works retroactively within one *entry*.

Displayed command line in the film:
`SetAnalogs` and `Channel` number
Position in the program:
Outside the time loop, exception: Chaser
Keyboard shortcut: `sa`

→ Chapter 3.6.1/2
ChaseDelay
ChaseSection

3.6.5 SetAnalogSingle

With this command you can control one DMX output analogously. In contrast to `SetAnalogs` you can only select only one output at a time here, but you can dim this one in or out continuously.

'General Information about Relays and Analogs'

Please read 'General Information about Relays and Analogs' in the previous chapter concerning possible errors and their prevention in connection with this command.

`Channel:`

`Channel`

Here you enter the number of the channel (1-8) containing the desired effect.

`Analog #:`

`Analog #`

Here you enter the number (1-32) of the desired output.
The name you gave the effect automatically appears in the line underneath.

You can only enter a number between 1 and 32 here. The number of the DMX channel, however, is calculated from the number of the channel and the output specified here.

For instance, if you selected channel 3 in the previous parameter and now `Analog` # 25 the number of the DMX channel is 89, namely 2x32 (from channel 1 + 2) + 25, the output of channel 3.

This can be reduced to the following formula:

DMX = (channel - 1) x 32 + output

Used on our specific example from above the formula would look like this: (3 - 1) x 32 + 25 **= 89**

You can find derivatives of this formula in the chapter about `SetRelais`.

→ Chapter 3.6.3 SetRelais, Channel

`From:`

`From`

Here you specify the starting value. Let us assume it is a lamp. If you want to dim it in you enter a 0.0 here and a 1.0 in the next parameter and then you specify the time it is supposed to take.

If the lamp is to go out slowly enter a 1.0 here. Of course all intermediate steps are also possible.

In addition you have the possibility of setting a dash (-) here instead of a value. You can simply click on it in the *menu bar* at the bottom or press F1. This dash has the effect that the starting value for whatever you enter for the following two parameters depends on the setting made before.

So regardless if the device was on, off or dimmed before, it will be set to the value you enter for the following parameter.

`To:`

`To`

Here you enter what is to happen to with device.
The starting point is the value you entered for `From`.
For dimming it to half power enter 0.5.
If you simply want to switch it off, enter 0.0.

Instead of a constant value you can also define a *curve* or enter a *variable*, e.g. `Fader0`.

→ `CurveEditor`

In this case whatever you entered for `From` has no influence anymore.

→ Chapter 3.12. `Set Variables`

If a curve is used the dimming process would start with the starting value of the *curve* and dim to its final value according to the *curve*.

With `Fader0` you take the dimming process entirely into your own hand, i.e. you can dim the device on or off continuously with the first fader on the fader board.

`Time:`

`Time`

Here you can specify how long the dimming process is to take.

The default setting is a dash (-). You have to keep this setting if you entered a *curve* or a *variable* under `To`. The `SetAnalogSingle` command then has to be located inside the time loop.

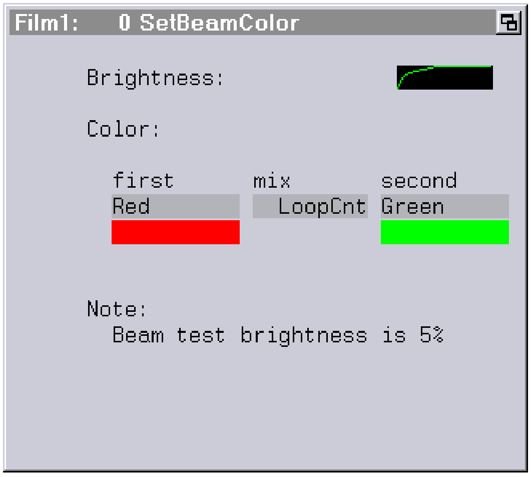
If you entered a constant value for `To` you can specify the time period during which the dimming process is to be executed at this point.
In this case the `SetAnalogSingle` command can be outside the time loop.

Displayed command line in the film:
`SetAnalogSingle`
Position in the program:
depending on the application inside or outside the time loop
Keyboard shortcut: `sas`

3.6.6 SetBeamColor

→ Chapter 3.6.8
`SetBeamTimed`

The command “SetBeamColor” is used to specify the color of scan beams. It is a direct replacement of the old command "SetCOlor" (see 3.6.9).



Film1: 0 SetBeamColor

Brightness:

Color:

first	mix	second
Red	LoopCnt	Green
🟥		🟩

Note:
Beam test brightness is 5%

In contrast to the command "SetCOlor" where each of the six laser lines has to be adjusted separately, the "SetBeamColor" command uses colors from the primary color table.

This has numerous advantages:

1) Color transitions between two colors can easily be realized.
2) Colors can be adjusted centrally even if the film which is to be changed is not editable.
3) Scan beams which use only the color lines 4 to 6 will now be displayed in the preview window (this was not the case with "SetCOlor" because there the first three laser lines where used as monitor colors).

!!!

Hint:
It is advisable to define new colors for the beams in the primary color table. This allows adjusting the beam colors independently of the other colors.

Note:
As with the command "SetCOlor", for safety reasons the test level set in the beam table will be used when editing the command "SetBeamColor".

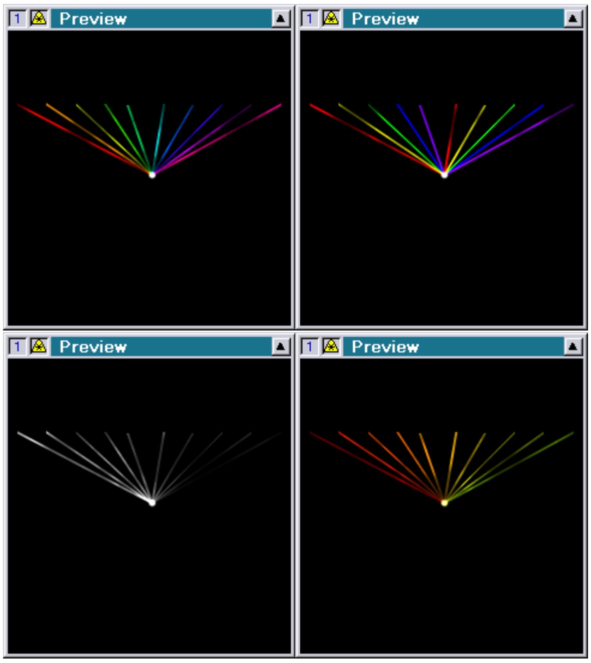
Displayed command line in the film:
`SetBeamColor`
Position in the program:
Before `SetBeamTimed`
Keyboard shortcut: `sbc`

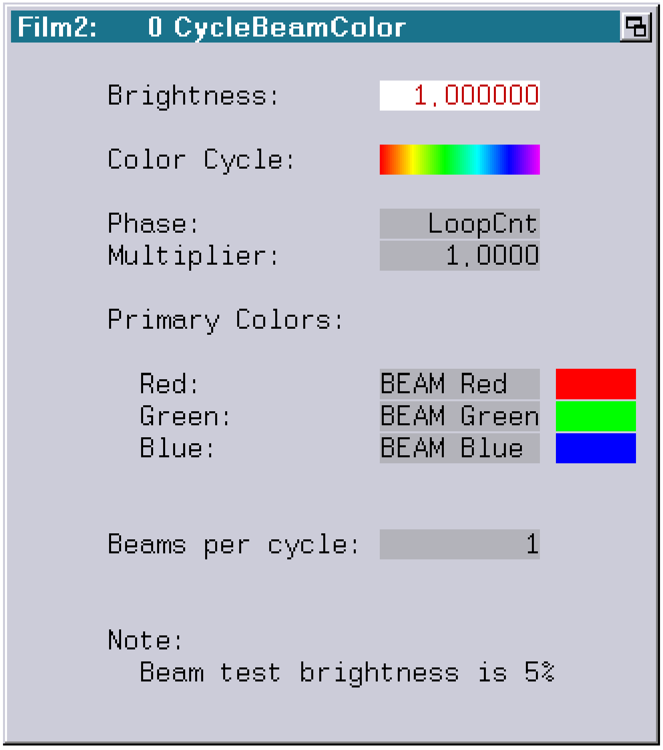
3.6.7 CycleBeamColor

The trickfilm command `CycleBeamColor` offers a simple way to color scan beams and to realize beam chases.

→ Chapter 3.6.8 `SetBeamTimed`

Here are a few examples:





Film2: 0 CycleBeamColor

Brightness:	1.000000	
Color Cycle:		
Phase:	LoopCnt	
Multiplier:	1.0000	
Primary Colors:		
Red:	BEAM Red	🟥
Green:	BEAM Green	🟩
Blue:	BEAM Blue	🟦
Beams per cycle:	1	

Note:
Beam test brightness is 5%

Brightness: Determines the brightness of the beams colored using the `CycleBeamColor` command

ColorCycle: Enter or edit the color cycle used by the command

Phase: The parameter Phase is used to select the desired color within the color cycle. Available value range: 0.0000 (first color) to 1.0000 (last color).

Multiplier: If the color cycle is to run more than once within the loop (parameter Phase = LoopCnt) this can easily be achieved by increasing the multiplier. A value of 2 will for instance result in two runs of the color cycle within the loop.

Red: Primary color which is used for the red portion in the color cycle

Green: Primary color which is used for the green portion in the color cycle

Blue: Primary color which is used for the blue portion in the color cycle

Beams per Cycle: If all following scan beams are to have the same color this value has to be set to 1. Values >1 will lead to changing colors from scan beam to scan beam. For this the color cycle is divided into a corresponding number of sections of equal size. If the value entered here is equal to the number of scan beams the colors used by the scan beams will be distributed evenly across the complete color cycle.
If the value entered is higher than the number of the actually used scan beams only a portion of the color cycle will be visible at a time. Changing the phase will then shift the visible portion.

Displayed command line in the film:
`CycleBeamColor`
Position in the program:
Before `SetBeamTimed`
Keyboard shortcut: `cbc`

3.6.8 SetBeamTimed

With this command you can position individual beams on mirrors or any other laser effect.

If you use this command frequently you should create a `Beam table`.

You can color the beams by setting the command `SetBeamCOlor` or `CycleBeamColor` before `SetBeamTimed`.

→ Chapter 3.6.6, 3.6.7

This command has to be handled with the utmost caution. Static beams can cause serious eye damages up to blindness. Therefore never point beams at areas where people may be. While in Edit Mode, the beam test brightness from the beamtable will be used (default is 5 %). You can change this setting in the beamtable window.

!!!!
💣
!!!!

`Adjust beam:`

`Adjust beam`

The default setting for this parameter is `BeamAdj`.

`BeamAdj`

With this setting you can either define the position of the beam in the trickfilm or select a *beam* from your `Beam table`.

When you click on this field the *menu bar* at the bottom will also offer the alternative `Globals`. You can only select this setting if you are not using a `Beam table`.

`Globals`

With the setting `Globals` the position of the beams defined in the trickfilm is changed according to possible changes to the *Global Parameters*. However, this setting actually only makes sense if the *beam* is part of a graphics projection.

If you want to access mirrors or other effects you should always use a `Beam table`.

`Time in seconds:`

`Time in seconds`

The default setting is 0.0016. With this setting the human eye perceives the *beam* as a standing effect even if you are displaying more than one simultaneously. You can increase up to approx. 0.01. The beam will then be brighter.

Position X:

Position X

Here you define the horizontal position of the *beam* along the X-axis.

Position Y:

Position Y

Here you define the vertical position of the *beam* along the Y-axis.

X & Y:

X & Y

Here you can set the horizontal and vertical position of the *beam* simultaneously with the *mouse poti*.
If you click on the field and enter a number, this value will automatically be entered for `Position X` as well as Y.

Beam number:

Beam number

If you want to use a *beam* from a `Beam table` click on this field. A new window will open with the name `Beam table for DSP`. There you click on the desired *beam* and it will automatically be assigned.

When you have entered a *beam* here you cannot enter anything for `Position X` and Y. Anything you may have entered before will automatically be set back to 0. If you enter anything afterwards the selected *beam* will be deleted and replaced by the newly entered position.

!!!

Displayed command line in the film:
`SetBeamTimed` and `Time in seconds`
Position in the program:
Within the time loop, if existent
Keyboard shortcut: `sbt`

3.6.9 SetCOlor

This command was used to color scan beams before the command `SetBeamColor` has been introduced.

→ Chapter 3.6.6
`SetBeamColor`

In contrast to the command `SetBeamColor` the command `SetCOlor` directly controls the output of the six color lines ignoring the Primary Color Table.

You can choose between RGB (**R**ed, **G**reen, **B**lue) and three additional lines if you operate your laser system with six lines.

The additional lines are labeled `C4-C6` because these color lines are not identical for all lasers. For instance not every laser has a yellow line.

The default setting for all lines is 1.0, i.e. full power and - depending of the laser used - white light.

Red:
Green:
Blue:

Red
Green
Blue

For those of you who are not familiar with the theory of colors we present an overview to show which combinations result in which colors.
Of course you can mix the whole color palette using intermediate values. Simply click into the number field next to the color line which you want to adjust and experiment with the *mouse poti*. In any case you should adjust the colors with your laser because what you see on the monitor is not necessarily equivalent to the laser output.

☞

Instead of using a fixed value you can use a curve or a variable to calculate the colors.
Note that you cannot change the balance of these colors in the primary color table.

Red:	1.0	
Green:	0.0	= red
Blue:	0.0	

Red:	1.0	
Green:	1.0	= yellow
Blue:	0.0	

Red:	1.0	
Green:	1.0	= white
Blue:	1.0	

Red:	0.0	
Green:	1.0	= cyan
Blue:	1.0	

Red:	0.0	
Green:	0.0	= blue
Blue:	1.0	

Red:	1.0	
Green:	0.0	= magenta
Blue:	1.0	

Red:	0.0	
Green:	1.0	= green
Blue:	0.0	

C4:	C4
C5:	C5
C6:	C6

With most lasers these lines stand for cyan, indigo and yellow.

Displayed command line in the film:
`SetCOlor`
Position in the program:
before `SetBeamTimed`
Keyboard shortcut: `sco`

3.7 Conditional Jump Commands

3.7.1 ConditionalGoto

→ Chapter 3.2.5 Goto

The ConditionalGoto command, like the Goto command, makes the program jump to a selected *entry*.

With the ConditionalGoto, however, the command is only executed when a condition defined by you is fulfilled.
This command is mainly used for interactive show programming.

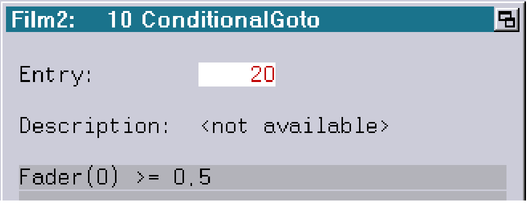
Entry number

Entry number:

Here you enter to which *entry* the program is to jump when the condition defined below is fulfilled.

To enter a condition, click on the field below Entry number. In the *input* line at the bottom you will see the text you write which will be copied to the upper field as soon as you confirm with *Return*.
If you want to change an existing text, click on it and it will be copied to the *input* line at the bottom. There you can make your changes and confirm again with *Return*.

Example 1:



Film2: 10 ConditionalGoto

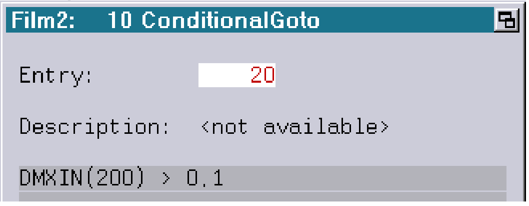
Entry: 20

Description: <not available>

Fader(0) >= 0.5

Here, the film will jump to Entry 20 if the first fader is set to 50% or higher.

Example 2:



Film2: 10 ConditionalGoto

Entry: 20

Description: <not available>

DMXIN(200) > 0.1

Here, the film will jump to Entry 20 if the DMX-In Channel 200 ist higher than 10%.

Displayed command line in the film:
ConditionalGoto and the selected *entry* number
Position in the program:
depending on the application
Keyboard shortcut: cg

3.7.2 ConditionalGoSub

This command, like the `ConditionalGoto` command, is mainly used for interactive show programming.

→ Chapter 3.2.6 `GoSub`

When a condition defined by you is fulfilled the program will jump to the specified *subroutine*.

→ Chapter 3.2.3 `Subroutine`

`Subroutine number:`

`Subroutine number`

Here you enter the number of the *subroutine* which is to be executed whenever the condition defined below is fulfilled.

To enter a condition, click on the field below `Subroutine number`. In the *input* line at the bottom you will see the text you write which will be copied to the upper field as soon as you confirm with *Return*.

If you want to change an existing text, click on it and it will be copied to the *input* line at the bottom. There you can make your changes and confirm again with *Return*.

See `ConditionalGoto` 3.7.1 for an example of a condition.

Displayed command line in the film:
`ConditionalGoSub` and the selected *subroutine* number
Position in the program:
within the time loop
Keyboard shortcut: `cgs`

3.7.3 OnGoto

This command is also mainly used for interactive show programming.
It is similar to the command `ConditionalGoto`.

→ Chapter 3.7.1 `ConditionalGoto`

You can define up to ten *entries* to which the program jumps when the corresponding value between 1 and 10 is fulfilled.

`Entry 1-10:`

`Entry 1-10`

Here you enter the numbers of the *entries*.

In the field below you specify the *variable* with which you call up the *entries* which are to be executed e.g. "Fader(0) * 10"

- if the value of the variable is <0.5, nothing happens
- if the value of the variable is >=0.5 and <1.5, Entry 1 will be executed
- if the value of the variable is >=1.5 and <2.5, Entry 2 will be executed
- if the value of the variable is >=2.5 and <3.5, Entry 3 will be executed

and so on.

Displayed command line in the film:
`OnGoto`
Position in the program:
depending on the application
Keyboard shortcut: `og`

3.7.4 OnGoSub

This command is similar to the `ConditionalGoSub` command.

→ Chapter 3.7.2 `ConditionalGoSub`

However, here you can specify up to ten different *subroutines* which are to be called up when the corresponding value between 1 and 10 is fulfilled.

`Subroutine 1-10:`

`Subroutine 1-10`

Here you enter the numbers of the *subroutines*.

In the field below you specify the *variable* with which you call up the *subroutines* which are to be executed:

- if the value of the variable is <0.5, nothing happens
- if the value of the variable is >=0.5 and <1.5, Subroutine 1 will be executed
- if the value of the variable is >=1.5 and <2.5, Subroutine 2 will be executed
- if the value of the variable is >=2.5 and <3.5, Subroutine 3 will be executed

and so on.

Displayed command line in the film:
`OnGoSub`
Position in the program:
depending on the application
Keyboard shortcut: `ogs`

3.7.5 Repeat / Until

With the commands `Repeat` and `Until` you can program loops which are independent of the time and are executed until a defined condition is fulfilled.

`Repeat` then replaces `TimeLoop`, `Until` takes the place of `LoopEnd`.

You enter the condition when to leave the loop at `Until`. When the condition is fulfilled the program continues with the next command after the command `Until`.

☜

In order to see the pictures in a `Repeat / Until` loop on the monitor you have to enter a `Wait` command before the `Until` command!

Displayed command line in the film:
`Repeat / Until`
Position in the program:
instead of the time loop
Keyboard shortcut: `r / u`

3.8 Point Of View Definitions

3.8.1 InitCamera

With the `InitCamera` command you position the camera and adjust its focal length.
You now view the corresponding picture through this camera.

This setting should not be confused with the `InitPosition` command with which you determine the position of the picture.

→ Chapter 3.8.3 `InitPosition`

Position #:

Here you can reference to a position list to define the camera setting.
The `Position #` specified here allows positioning a camera using a position defined previously with `InitPosition`.
If "none" is entered the camera is at the origin.
This for instance makes it easy to realize a camera course through an existing setup.

Horizontal offset:

Here you position the camera horizontally, i.e. the displayed picture is shifted to the right or to the left. This happens without the angle distortions that would occur when shifting the picture with `InitPosition`.

The scale reaches from – 256.0 to + 256.0.

Vertical offset:

Here you position the camera vertically, i.e. the displayed picture is shifted up or down. Other than that everything is equivalent to the `Horizontal offset`.

X & Y:

This *input* field makes setting the camera position easier because you can position the camera simultaneously along the x- and y-axis.

The easiest way is to set the desired position with the *mouse poti*. Click on the field with the left mouse button and shift the picture wherever you want to with the mouse button pressed.

`Focal length hor: / Focal length vert:`

This is the horizontal and the vertical focal length of the camera.
A small value creates the impression of a wide angle lens, a high value that of a telephoto lens.
With the wide angle you have a more effective perspective!
You can enter the vertical and horizontal focal length separately, thereby distorting the picture in height or width.

`X = Y:`

With this *input* field you can set the height and width for the camera lens simultaneously.

!!!

When you click on this field and move the mouse the horizontal value is automatically adapted to the vertical value, it is therefore not possible to assign different values to x and y.

You also have a scale from – 256.0 to + 256.0 for the focus settings. When a negative value is entered the picture will be mirrored horizontally and vertically.

`ZClipping at:`

With this parameter you define a section of the z-axis where the picture is to be projected. All points positioned beyond this section will be invisible.
This way you can make objects or parts of the picture which move too far away from the camera "disappear" in the distance.

☞

With the `ZClipping` you can create the impression of "Hidden Lines" for convex objects.
The best example is the revolving earth globe.
Using the `ZClipping` you can make the part of the globe which is on the far side of the viewer invisible.

You can also use a *variable* or a *curve* instead of a constant value for all parameters and that way make the camera move.
This way you do not need the command `ChangeCamera`.

→ `CurveEditor`

→ Chapter 3.12 `SetVariables`

Displayed command line in the film:
`InitCamera` and - if existing - `Position #`
Position in the program:
If the camera is to move within the time loop, otherwise before the time loop
Keyboard shortcut: `ic`

3.8.2 ChangeCamera

With this command you change the settings you made previously with the command `InitCamera`.

You can for instance pan the camera or zoom an object closer. However, this command is not necessary if you work with *variables* or *curves* for `InitCamera`.

→ Chapter 3.8.1 `InitCamera`

`Factor:`

Here you enter the type of movement.
For instance, you can start the movement fast and then let it run out slowly, or the other way around, or...
This you determine either with a *variable* or a *curve*.

→ `CurveEditor`

→ Chapter 3.12 `SetVariables`

`Horizontal offset:`

Here you determine how far the picture is to be shifted horizontally, i.e. to the right or to the left.

This happens without the angle distortions you would have when shifting the picture with `ChangePosition`.

→ Chapter 3.8.4 `ChangePosition`

The scale reaches from – 256.0 to + 256.0.

Entry by *mouse poti* or keyboard.
Numeric values, *variables* or *curves* are possible.
This is also true for all following parameters.

`Vertical offset:`

Here you shift the picture vertically, i.e. the displayed picture is shifted up or down. Other than that everything is equivalent to the `Horizontal offset`.

`X & Y:`

This *input* field allows you to enter the movement for the x- and y direction simultaneously.
The easiest way is to set the desired position with the *mouse poti*. Click on the field with the left mouse button, keep the mouse button pressed and move the picture to the desired end position.

Focal length hor: / Focal length vert:

Here you change the focal length. You can zoom an object closer or farther within the time loop.

You can change the zoom separately vertically and horizontally, creating interesting special effects.

If you want to change the focus with the same ratio the easiest way is to use the *mouse poti* in the *input* field:

X = Y:

See X & Y for handling.
However, different values for x and y are then no longer possible. **!!!**

You also have a scale from – 256.0 to + 256.0 for the focus settings. If the value entered is negative the picture is mirrored horizontally and vertically.

ZClipping at:

With this parameter you change the section of the z-axis along which the picture is visible.

Displayed command line in the film:
`ChangeCamera`
Position in the program:
within the time loop, after `InitCamera`
Keyboard shortcut: `CC`

3.8.3 InitPosition

With the `InitPosition` command you define a position list for your graphics.

You determine the size, spatial orientation and position of the graphic on the screen or in space.

The following fig. shows the default values.

`Position #:`

Here you enter the identification number for the position list.

You can define up to 16 (0 - 15) position lists per time loop. You can then assign these different position lists to different pictures or up to eight to one single picture, if it was digitized with different matrixes. You then assign the numbers of the different lists to the different matrix numbers at the command for the picture output.

→ *Picture Editor* `Piced`

→ Chapter 3.9 Picture Output

`Master Position #:`

You can use this to arrange position lists in a hierarchical order.

Let us assume you want to have three different graphics execute the same Z-rotation, but you want to assign an additional individual movement to each one.

In this case you enter four position lists, the first one with

```
Position #:         0
Master Position #:  None.
```

This position list dominates the following lists if they get the `Master Position #: 0`.

However, these lists still position their assigned pictures independent of each other.

You can also configure the following position lists hierarchically by numbering the `Master Position` numbers from `0 - 2`. The first list is still the `Master Position` of all others, but the second one in turn dominates the following two, the third one dominates the fourth one.

The first two lines of your four `InitPosition` commands would then look like this:

```
Position #:         0      dominates all
Master Position #:  None

Position #:         1      dominates the following
Master Position #:  0

Position #:         2      dominates the following
Master Position #:  1

Position #:         3
Master Position #:  2
```

Just like the command within a time loop, the parameters of a position list are executed in the given order.

→ Chapter 1.2 Structure of a Film

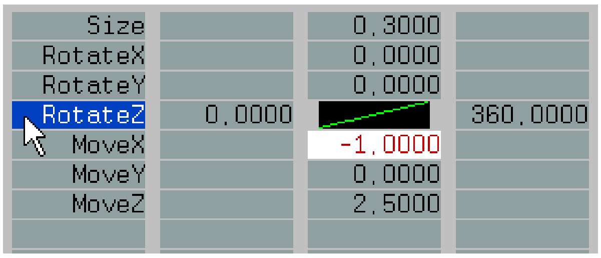
A rotation turns an object around the x-, y- or z-axis of the coordinate system.

If you rotate an object first and then shift it, it will be moved to its new position in the angle you defined with the rotation.

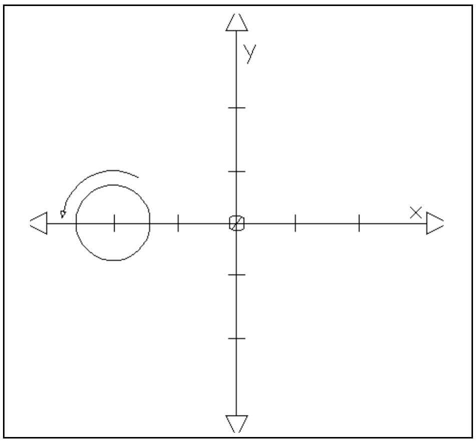
However, if you shift the object first and then enter the rotation, the rotation will still be around the same axis, but in an orbit whose radius is the distance of the shift.

This may sound complicated, so here is a simple example:

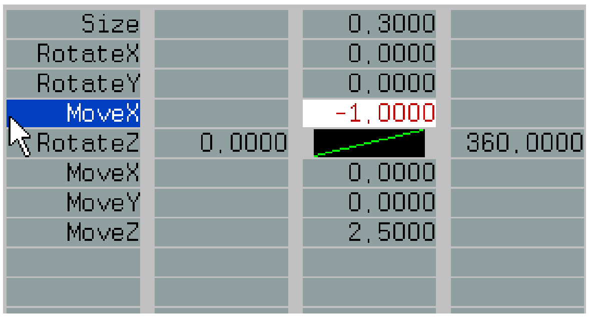
Imagine a centric circle. If you rotate it around the z-axis, i.e. its own center, the effect will be little spectacular.
If you then shift it a little to the left afterwards it will still rotate around its center in exactly this spot.



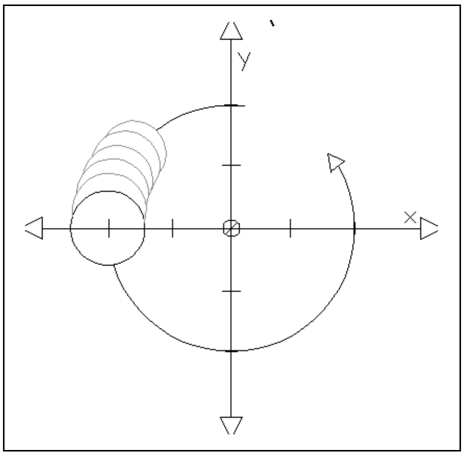
Size		0.3000	
RotateX		0.0000	
RotateY		0.0000	
RotateZ	0.0000		360.0000
MoveX		-1.0000	
MoveY		0.0000	
MoveZ		2.5000	



However, if you shift the circle to the left **first** and then enter the z-rotation, it will execute its rotation in a circular orbit around the center of the coordinate system starting from the point to which you shifted it.



Size		0.3000	
RotateX		0.0000	
RotateY		0.0000	
MoveX		-1.0000	
RotateZ	0.0000		360.0000
MoveX		0.0000	
MoveY		0.0000	
MoveZ		2.5000	



As you can see it makes a lot of difference if you rotate an object first and then shift it or the other way around.

When you insert the `InitPosition` command you will see the standard configuration in the *display window* showing seven parameters with which you can influence the total size of the graphic and execute rotations and shifts around resp. along the three axes.

Here at first the explanations of the parameters of the standard configuration:

For all sets of parameters you can change values by entering a number or by *mouse poti*.

`Size:`

`Size`

Enlarges or downsizes the picture on a scale of +/- 256.0.
By entering negative values you mirror a two-dimensional picture vertically as well as horizontally.

`RotateX:`

`RotateX`

Rotates the picture around the x-axis. The numerical values you enter here are equivalent to the rotation angle (+/- 360.0°).

90.0 = a two-dimensional graphic becomes a horizontal line,
180.0 = the graphic is mirrored vertically.

A positive value is equivalent to an anti-clockwise rotation.

RotateY:

RotateY

Rotates the picture around the y-axis. The numerical values you enter here are equivalent to the rotation angle (+/- 360.0°).

90.0 = a two-dimensional graphic becomes a vertical line,
180.0 = the graphic is mirrored horizontally.

A positive value is equivalent to an anti-clockwise rotation.

RotateZ:

RotateZ

Rotates the picture around the z-axis, i.e. the center of the drawing area.

The numerical values you enter here are equivalent to the rotation angle (+/- 360.0°).

A positive value is equivalent to an anti-clockwise rotation.

MoveX:

MoveX

Shifts the graphic along the x-axis on a scale of +/- 256.0.
A positive value shifts the graphic to the right, a negative to the left.

MoveY:

MoveY

Shifts the graphic along the y-axis on a scale of +/- 256.0.
A positive value shifts the graphic upwards, a negative downwards.

MoveZ:

MoveZ

Shifts the graphic along the z-axis.
The higher the value you enter the farther the object moves "backwards" from the center, i.e. it becomes optically smaller.

The default value of 1.5 is chosen so that a two-dimensional picture digitized in the Piced fills the entire display area.

If you want to rotate the picture and don't want it to be clipped you should enter a value of 2.5 instead.

Those were the parameters you will always find when you call up `InitPosition`.

Additional Parameters

You can, however, change this list. You can either delete individual parameters, change the order or add or insert additional parameters.

When you click on a parameter in the left column a new window with all available parameters will open.

If you click on a parameter in this window, the one you clicked on before in the `InitPosition` command will be overwritten.

If you do not want that you have to select one of the following from the *menu bar* at the bottom:

`Insert`
`Delete`
`Append`

`Insert` - inserts the new parameter before the selected one
`Delete` - deletes the selected parameter
`Append` - the new parameter is added at the end of the list.

If you do not want any of these, press the right mouse button and you will be back in the normal `InitPosition` command window.

You can add the following parameters to the `InitPosition` command:

`SizeX:`

Compresses or stretches the picture along the x-axis on a scale of +/- 256.0.

The default setting 1.0 means no change to the previous size.
0.0 results in a vertical line.

`SizeY:`

Compresses or stretches the picture along the y-axis on a scale of +/- 256.0.

The default setting 1.0 means no change to the previous size.
0.0 results in a horizontal line.

`SizeZ:`

Compresses or stretches the picture along the z-axis on a scale of +/- 256.0.

The default setting 1.0 means no change to the previous size.

shearXByY:

This parameter changes the x-coordinate depending on the y-coordinate. The values of the y-coordinates stay unchanged.

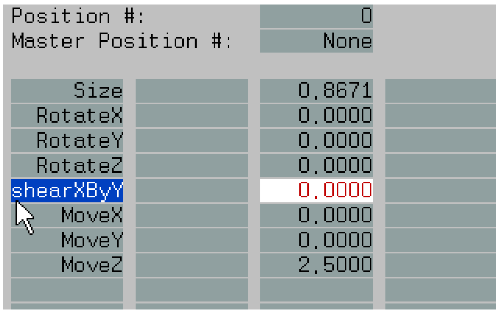
When a positive value is entered the points above the x-axis, i.e. y-positive points, are shifted to the right, i.e. along the x-axis in the positive direction. All points below the x-axis are pulled to the left.
When a negative value is entered the distortion is executed in the opposite direction.

All points located directly on the x-axis, i.e. those which have a value of y=0 in the coordinate system, stay unchanged.

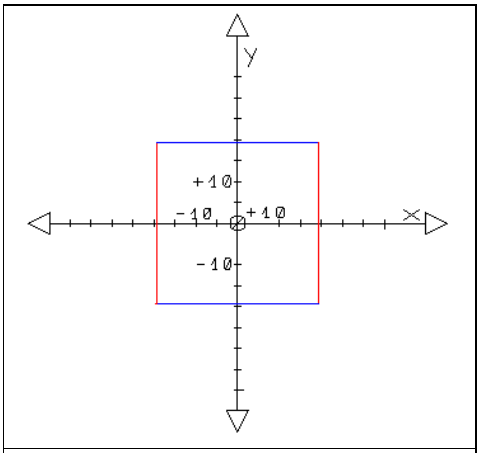
Generally the following is true for all *shearing* versions:

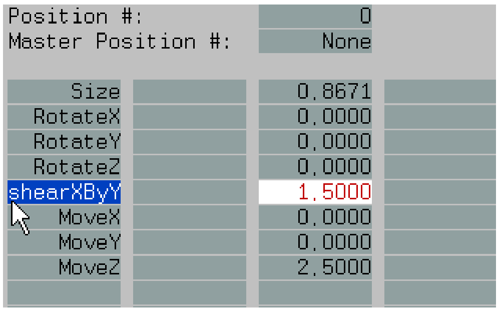
The coordinate value named first is modified depending on the second one. All points where the coordinate value of the axis named second is 0 stay in the same position.

As an illustration here first a square, unchanged in a X-Y coordinate system, then the same square with an XByY **shearing** of 1.5000.

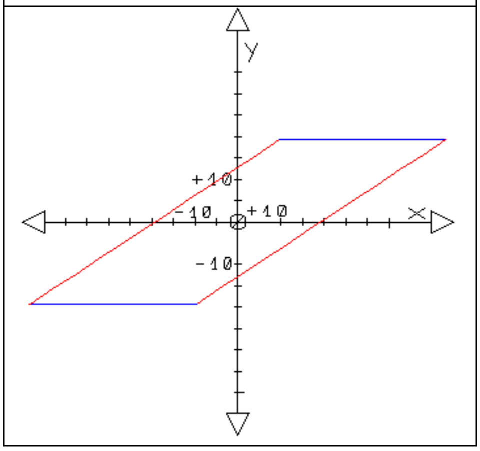


Position #:	0
Master Position #:	None
Size	0.8671
RotateX	0.0000
RotateY	0.0000
RotateZ	0.0000
shearXByY	0.0000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000





Position #:	0
Master Position #:	None
Size	0.8671
RotateX	0.0000
RotateY	0.0000
RotateZ	0.0000
shearXByY	1.5000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000



shearXByZ:

With this parameter you distort the picture in the direction of the x-axis, but in the x-z-plane.

You should always insert the shearXByZ **before** MoveZ, otherwise you will create a little spectacular effect and just shift the picture left and right on the x-axis. !!!

If you process a two-dimensional picture - which usually will have been digitized in the x-y-plane with z-coordinates of 0 - without shifting the positions of some of the points on the z-axis through an x- or y-rotation or a corresponding *shearing* **before** executing the shearXByZ, nothing will happen. !!!

As mentioned before for the shearXByY command, all points whose coordinate value of the axis named second - in this case the z-axis - is 0 will stay in the same position.

shearYByX:

With this parameter you distort the picture in the direction of the y-axis in the x-y-plane. The values of the x-coordinates remain constant.

When a positive value is entered the points to the right of the y-axis, i.e. x-positive points, are shifted upwards, i.e. along the y-axis in the positive direction. All points left of the y-axis are pulled downwards. All points located directly on the y-axis, i.e. those which have a value of x=0 in the coordinate system, stay unchanged.

When a negative value is entered the distortion is executed in the opposite direction.

shearYByZ:

This parameter distorts the picture in the y-direction in the y-z-plane.

Otherwise see shearXByZ.

shearZByX:

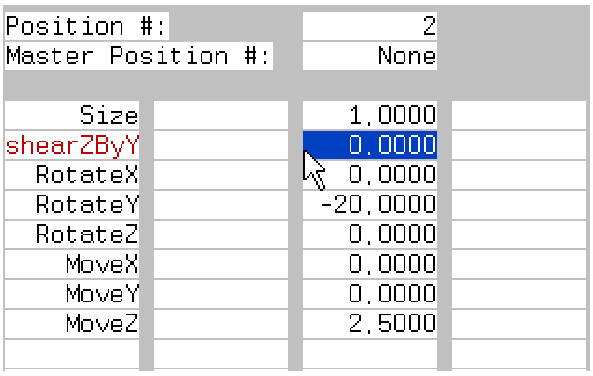
The picture is distorted in the x-z-plane in the direction of the z-axis.

When a positive value is entered it will look as if the points with negative x-values move towards you with a distortion, while x-positive points will be distorted backwards on the z-axis, i.e. they will get smaller.

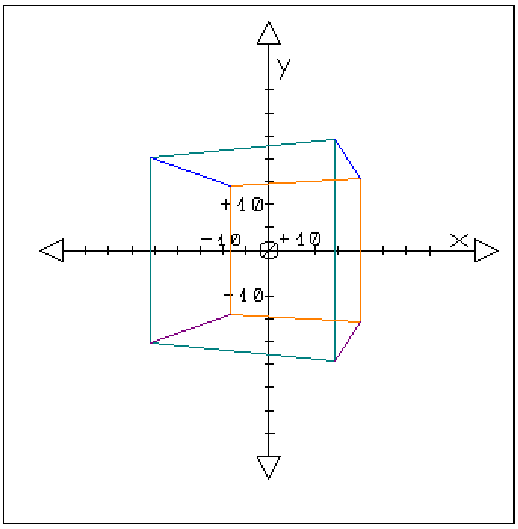
shearZByY:

The picture will be distorted in the direction of the z-axis in the y-z-plane.

The example below shows a cube which was given a slight slant with RotateY.



Position #:	2
Master Position #:	None
Size	1.0000
shearZByY	0.0000
RotateX	0.0000
RotateY	-20.0000
RotateZ	0.0000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000

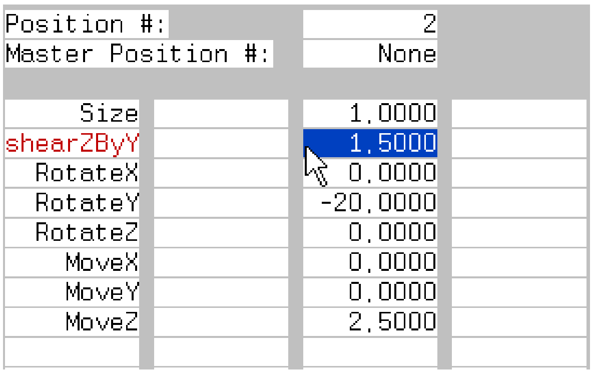


The following two illustrations serve to emphasize how important the order of the commands within the parameter list is.

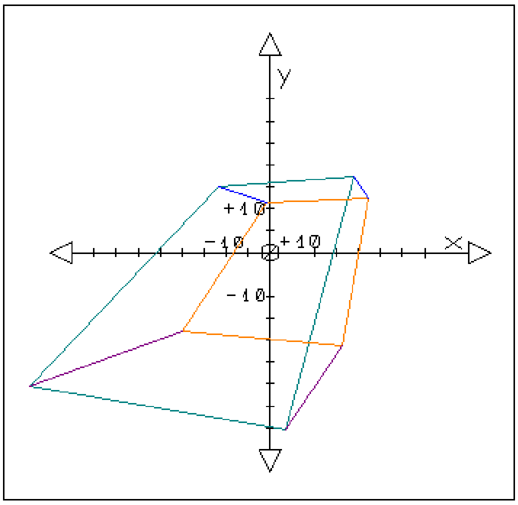
In both cases a shearZByY of 1.5000 was entered.

In the first case the shearZByY command was inserted **before** the rotations and moves so that the *shearing* would not be influenced by any rotations (in this case RotateY) or shifts.

The cube as it was digitized is distorted in the z-y-plane. Then the distorted cube is rotated around the y-axis.

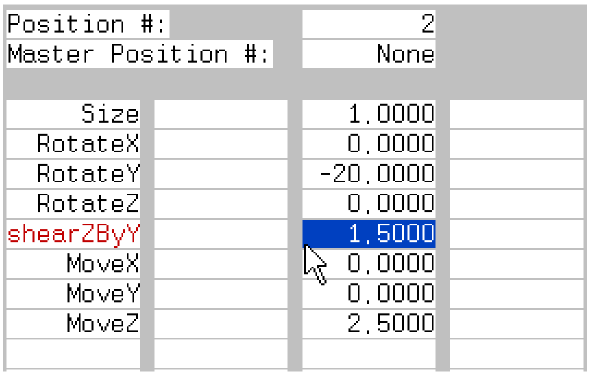


Position #:	2
Master Position #:	None
Size	1.0000
shearZByY	1.5000
RotateX	0.0000
RotateY	-20.0000
RotateZ	0.0000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000

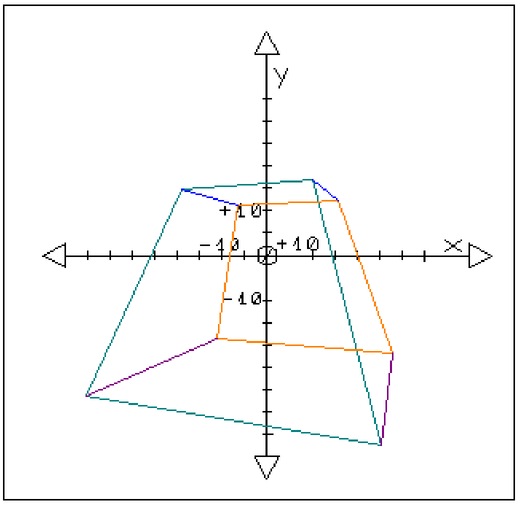


In the second example the same `shear` command is executed **after** the y-rotation. As you can see this leads to a completely different result.

The cube is rotated first and the resulting changed perspective is the basis for the distortion.



Position #:		2	
Master Position #:		None	
Size		1,0000	
RotateX		0,0000	
RotateY		-20,0000	
RotateZ		0,0000	
shearZByY		1,5000	
MoveX		0,0000	
MoveY		0,0000	
MoveZ		2,5000	



Oscillators:

Oscillators

An oscillator is an oscillation generator.

You can make you graphic or parts of it oscillate with the Osc(illator) parameters. These oscillations run in predefined paths, the follow so called wave forms.

You can choose from four different wave forms:

SAW

SAW: is a saw-tooth function. The picture oscillates in one directions, jumps to the opposing end point and oscillates back again.

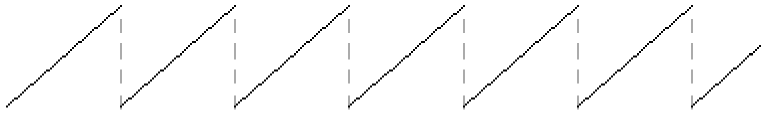


Fig. Saw-Tooth Function (SAW)

SIN

SIN: defines a sine wave. The picture oscillates back and forth, slowing down smoothly towards the end points, similar to an object hanging on an elastic spring and swinging back and forth.

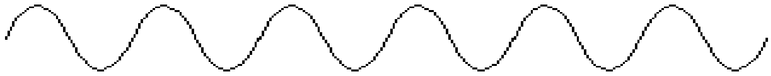


Fig. Sine Function (SIN)

SQuare

SQuare: is a square function. The picture jumps back and forth between the outer end points according to the amplitude which was enteted.

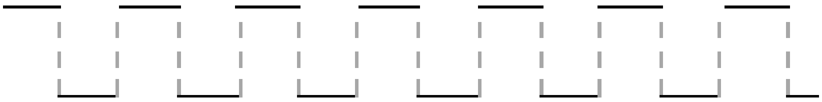


Fig. Square Function (SQuare)

Triangle

Triangle: is a triangle function. The picture swings back and forth steadily with hard stops at the end points.

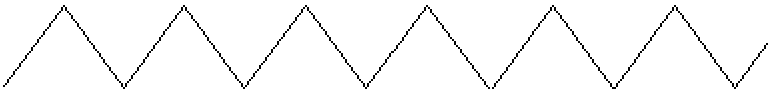


Fig. Triangle Function (Triangle)

If you want to work with the *oscillators* you have to select a wave form first and add it to the position list.

The selected wave form is effective for all oscillators set below until another wave form is selected.

The value you enter determines the amplitude of the oscillation. The default setting is `1.0`.

You can either enter a constant value or change the amplitude using a *curve*, increasing or decreasing the extent of the oscillation.

With the **`OscOffset`** you shift the origin of the oscillator. `OscOffset`

Example:
A sine wave with an amplitude of 1 swings from -1 to +1.
If an *offset* of 1 is set it swings from 0 to 2 instead.

The `OscOffset` has to be entered before the `Osc`-movement.

`OscMoveX` makes the graphic swing in the direction of the x-axis, i.e. horizontally. `OscMoveX`

The entered values are equivalent to the degree numbers:

If you define a *curve* from 0-360 (°) the movement will be executed once.

Starting value 0 = start at the 0-axis
Starting value 90 = start at the end point

`OscMoveY` makes the graphic swing in the direction of the y-axis, i.e. vertically. `OscMoveY`

The entered values are equivalent to the degree numbers:
If you define a *curve* from 0-360 (°) the movement will be executed once.

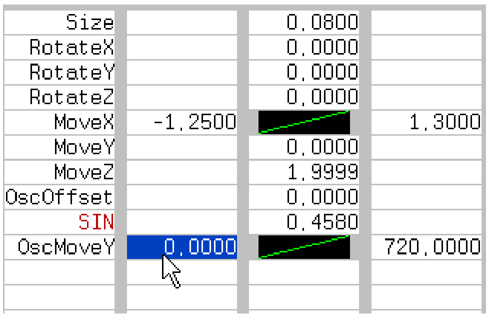
Starting value 0 = start at the 0-axis

In the example shown below we defined the movement with a *curve*.

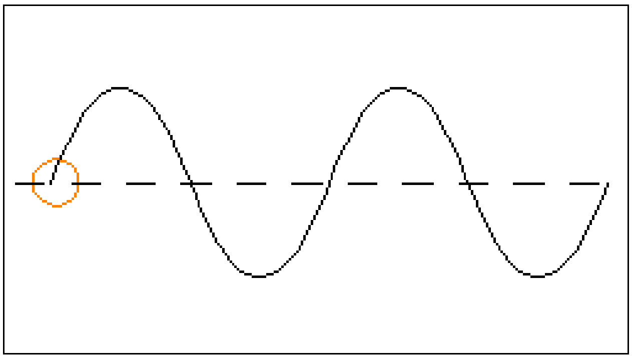
The circle swings from x=1.25 / y=0 upwards by 0.458 (according to the amplitude, i.e. the value we entered for SIN), from there downwards along the y-axis to -0.458 and back to the point of origin.

This movement would be executed twice because we entered a value of 720.

At the same time we entered a `MoveX` so that the circle follows the shown wave movement.

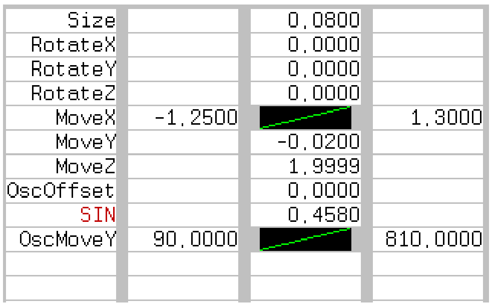


Size		0,0800	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX	-1,2500		1,3000
MoveY		0,0000	
MoveZ		1,9999	
OscOffset		0,0000	
SIN		0,4580	
OscMoveY	0,0000		720,0000

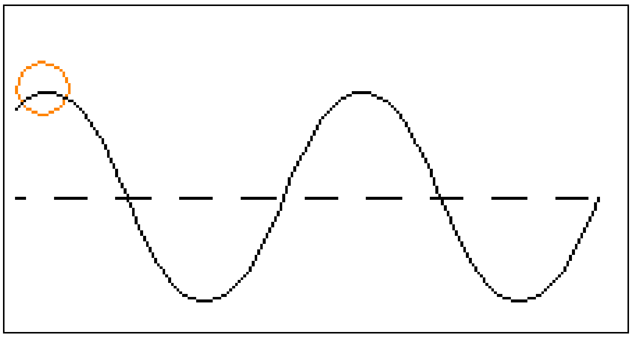


In the following example we have the starting value 90.

The red circle starts the wave movement at the upper end point (y = 0.458), swings down to y = -0.458 and back to the starting point. The movement would be executed twice. (2 x 360° = 720 + 90 = 810)



Size		0,0800	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX	-1,2500		1,3000
MoveY		-0,0200	
MoveZ		1,9999	
OscOffset		0,0000	
SIN		0,4580	
OscMoveY	90,0000		810,0000



OscMoveZ

`OscMoveZ` lets the graphic swing in the direction of the z-axis. This creates the impression that the graphic swings towards the viewer and then away again.

The entered values are equivalent to the degree numbers:
If you define a *curve* from 0-360(°) the movement will be executed once.

Starting value 0 = start at the 0-axis
Starting value 90 = start at the highest end point, in this case the point farthest from the viewer.

With a starting value of -90 the movement will start when the picture is largest, i.e. closest to the viewer on the z-axis.

OscRotX

OscRotX

The graphic executes an oscillating rotation around the x-axis.

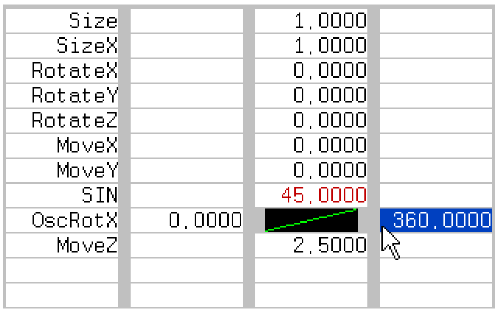
The entered values are equivalent to the degree numbers:
If you define a *curve* from 0-360(°) the selected wave form will be executed once.

Starting value 0 = start at the 0-axis
Starting value 90 = start at the highest end point
In our example this would be at a rotation of 45°.

To achieve a visible effect for the rotations - in contrast to the **OscMove** parameters - the amplitude, i.e. the value you enter for the selected wave form, should be greater than 1.
The entered values are equivalent to the degree numbers. With an amplitude of 45 the graphic will rotate by 45°. For a complete revolution around the x-axis you naturally have to enter 360 (°).

In the example below we defined the movement with a *curve*.

The cube turns from x=0 (see fig. OscRotX 1) by 45° (according to the amplitude, i.e. the value we entered for SIN) around the x-axis (fig. OscRotX 2), swings back to -45° (fig. OscRotX 3) and back to the point of origin.



Size		1,0000	
SizeX		1,0000	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX		0,0000	
MoveY		0,0000	
SIN		45,0000	
OscRotX	0,0000		360,0000
MoveZ		2,5000	

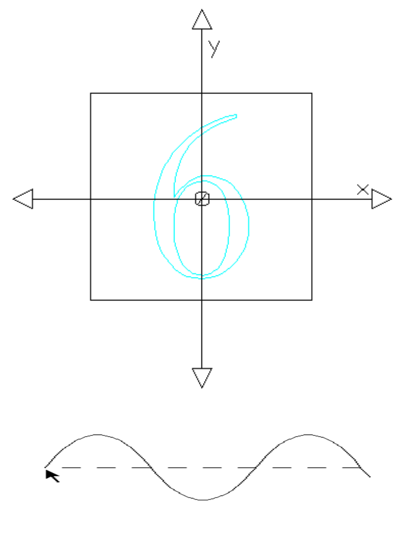


Fig. OscRotX 1

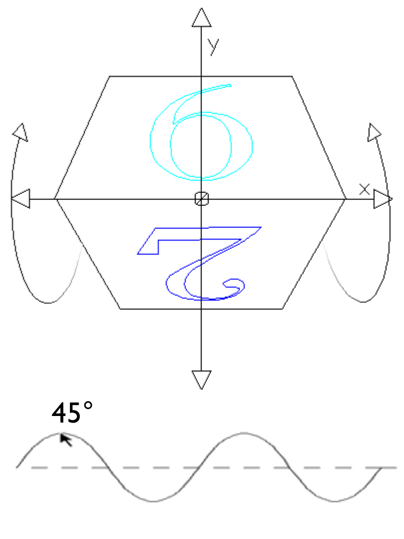


Fig. OscRotX 2

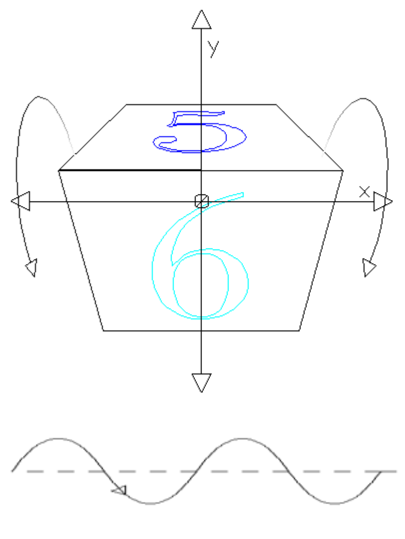


Fig. OscRotX 3

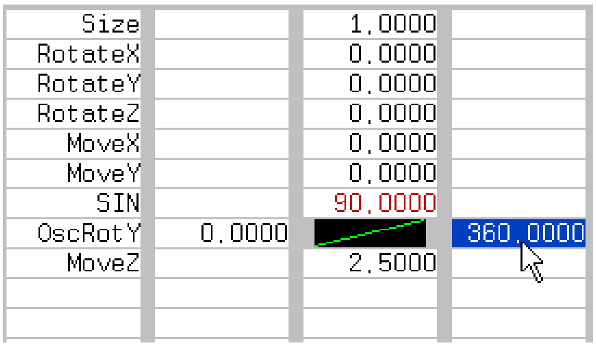
Wave function and OscRotX and OscRotY have to be inserted before MoveZ! **!!!**

OscRotY:

OscRotY

The graphic executes an oscillating rotation around the y-axis.

In the example below the cube swings around the y-axis. It first turns counterclockwise by 90°, slows down the movement before it reaches the highest amplitude (90°, the 4 is in front), accelerates again while it swings back clockwise until it slows down again at -90° (the 3 is in front) and swings back to the starting point.



Size		1,0000	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX		0,0000	
MoveY		0,0000	
SIN		90,0000	
OscRotY	0,0000		360,0000
MoveZ		2,5000	

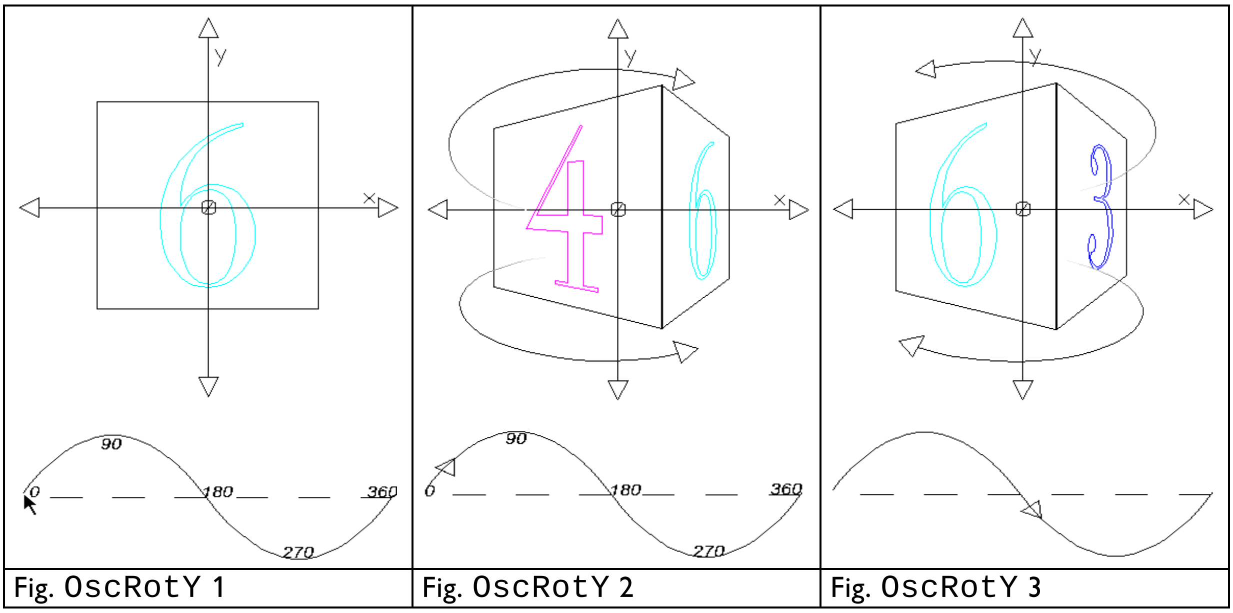


Fig. OscRotY 1 | Fig. OscRotY 2 | Fig. OscRotY 3

OscRotZ

OscRotZ:

With this parameter you give the graphic an oscillating rotation around the z-axis.

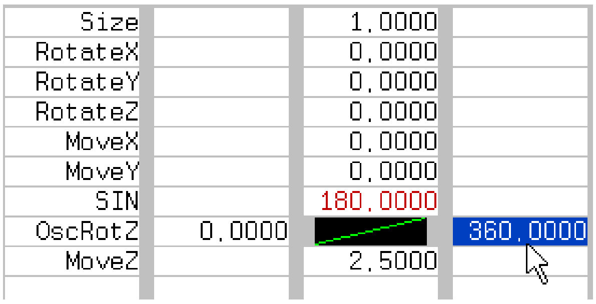
As with the other rotations the entered values are equivalent to the degree numbers.

When you enter a *curve* from 0 to 180 the graphic swings from the point of origin once counterclockwise up to the highest amplitude and then clockwise back to the starting point.

When you define a *curve* from 90 to 270 the movement starts at the highest amplitude and swings clockwise over the point of origin up to the lowest amplitude.

In our example we entered a *curve* from 0 to 360, i.e. the selected wave form is executed once completely.

With an amplitude of 180 (SIN = 180) this means the graphic swings from the point of origin (fig. OscRotZ 1) 180° counterclockwise (fig. OscRotZ 2), then clockwise 360° (fig. OscRotZ 3) and then back again counterclockwise to the origin.



Size		1,0000	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX		0,0000	
MoveY		0,0000	
SIN		180,0000	
OscRotZ	0,0000		360,0000
MoveZ		2,5000	

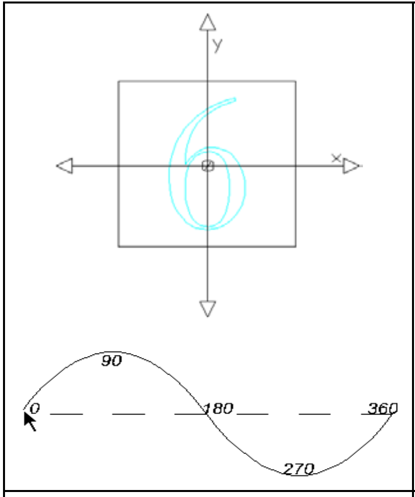


Fig. OscRotZ 1

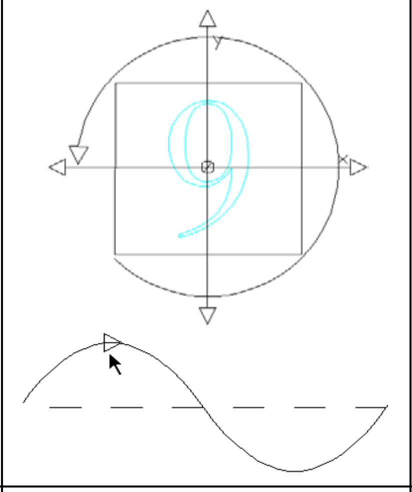


Fig. OscRotZ 2

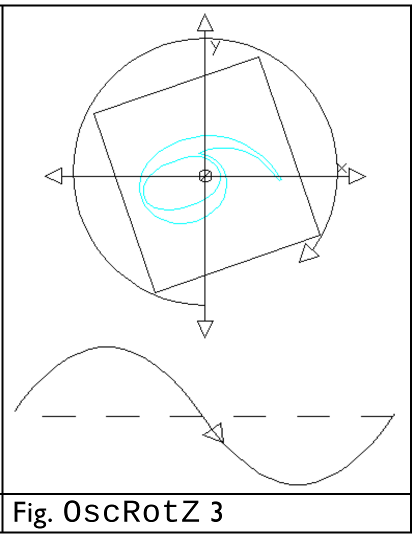
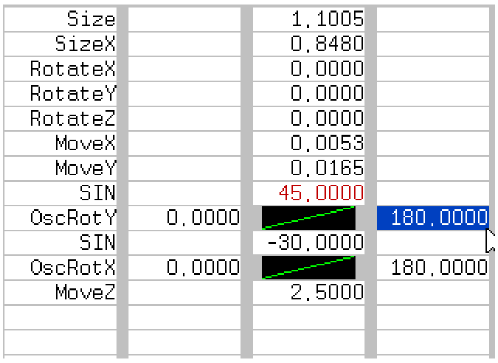


Fig. OscRotZ 3

Of course you can combine all OscRot(ations) with each other and program complete motion sequences using *curves*.

YOU HAVE TO DEFINE A SEPARATE WAVE FORM BEFORE EACH NEW ROTATION PARAMETER.



Size		1,1005	
SizeX		0,8480	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
MoveX		0,0053	
MoveY		0,0165	
SIN		45,0000	
OscRotY	0,0000		180,0000
SIN		-30,0000	
OscRotX	0,0000		180,0000
MoveZ		2,5000	

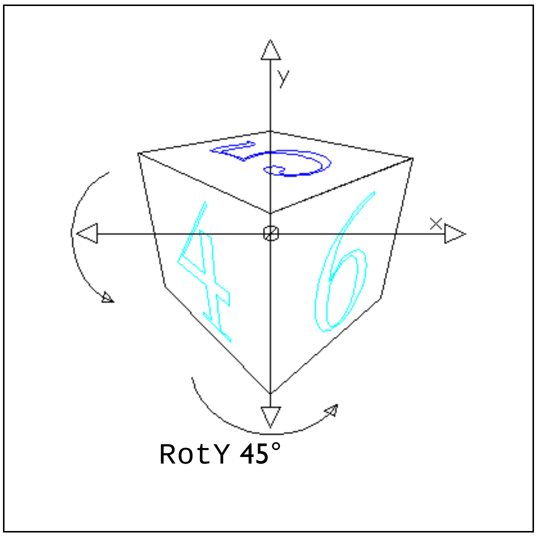


Fig. OscRotX/Y

OscshXByY: OscshXByY → ShearXByY

This is used to apply a wave form to the ShearXByY command.

OscshXByZ: OscshXByZ → ShearXByZ

This is used to apply a wave form to the ShearXByZ command.

OscshYByX: OscshYByX → ShearYByX

This is used to apply a wave form to the ShearYByX command.

OscshYByZ: OscshYByZ → ShearYByZ

This is used to apply a wave form to the ShearYByZ command.

OscSize: OscSize

With this command you can change the size of the graphic with a wave form. The amplitude, i.e. the value you enter for the selected wave form, determines the maximum size of the graphic.

Wave function and `OscSize` have to be inserted <u>before</u> `MoveZ`, otherwise you will not achieve the desired effect. **!!!**

The change of size would not happen in the center of the object and depending on the choice of projection parameters you will not see any change in the display.

When you define a ***curve*** **of 0-90** for `OscSize`, with `SIN` and `Triangle` the graphic will zoom to the defined size once.

With the `SQuare` function nothing will happen because the picture will only jump back and forth between the two end points, it is therefore only displayed in full size.

If you chose SAW the picture will only zoom to half size.

CURVE FROM 0-180:

`SIN` and `Triangle`: The picture zooms in and back out.

SAW: The picture zooms to full size.

`SQuare`: Constant full size.

CURVE FROM 0-360:

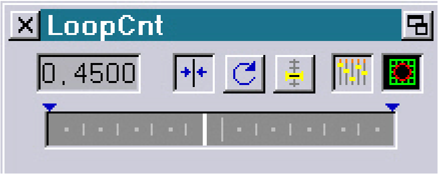
`SIN` and `Triangle`: The picture zooms in, zooms back out and then zooms back in and out mirrored vertically as well as horizontally.

SAW: The picture zooms in to full size, then jumps to the opposite end point, i.e. amplitude -x. That means it then zooms back out mirrored vertically and horizontally.

`SQuare`: The picture jumps back and forth between 'normal' and mirrored display.

For the following examples we selected a sine-, saw-, and square wave function with an amplitude of 1. We run the *curve* once each time by defining a *curve* from 0 - 360.

We stopped the course of the different *curves* twice at the same spot for each example to illustrate the differences.



LoopCnt

0,4500

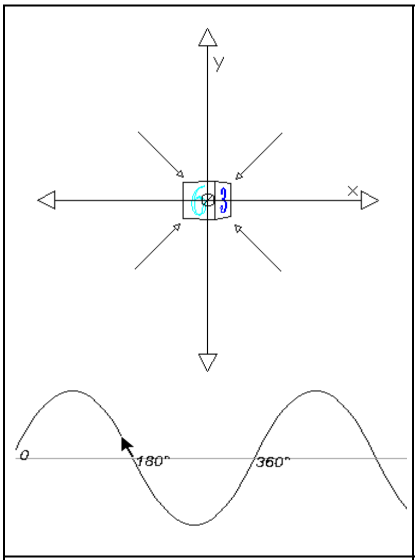


Fig. OscSize SIN 1

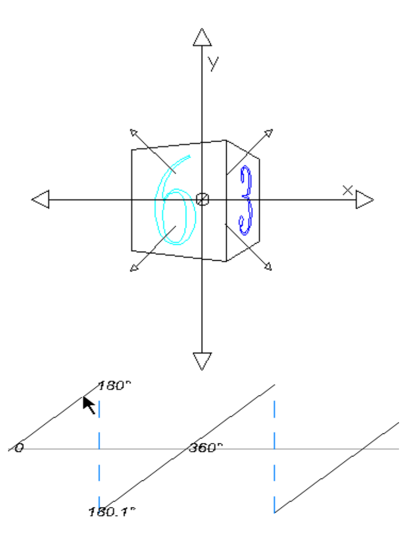


Fig. OscSize SAW 1

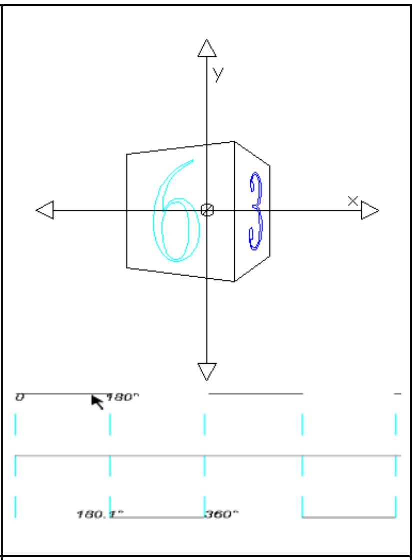
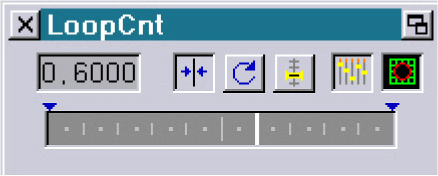


Fig. OscSize Square 1



LoopCnt

0,6000

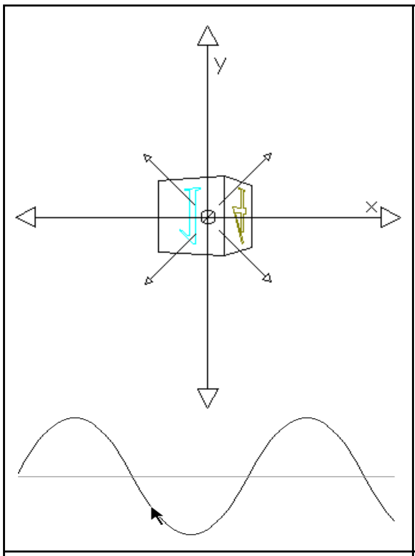


Fig. OscSize SIN 2

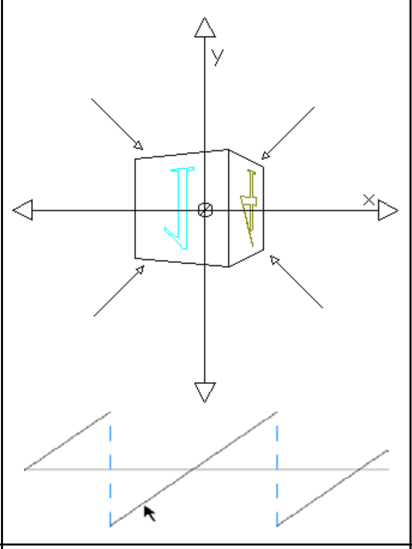


Fig. OscSize SAW 2

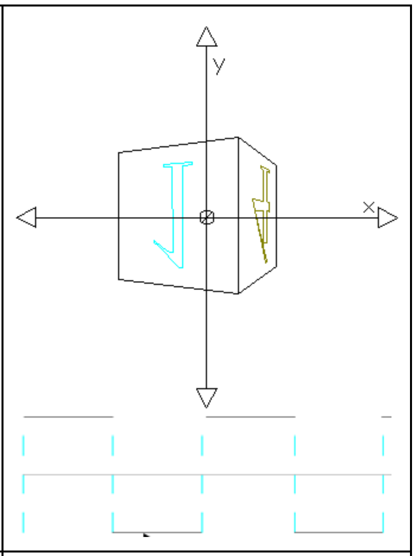


Fig. OscSize SQuare 2

OscSizeX:

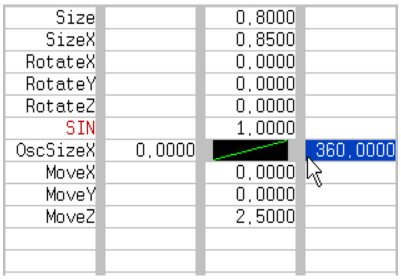
OscSizeX

With this command you compress and/or stretch the graphic along the x-axis with the selected wave form.

The amplitude determines the maximum stretching of the graphic.

In the example below we selected a sine with an amplitude of 1. We run the sine *curve* once by defining a *curve* from 0 - 360.

The figures 3 and 5 show at which point of the *curve* we are in the LoopCount window at that moment.



Size		0,8000	
SizeX		0,8500	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
SIN		1,0000	
OscSizeX	0,0000		360,0000
MoveX		0,0000	
MoveY		0,0000	
MoveZ		2,5000	

Fig. OscSizeX 1

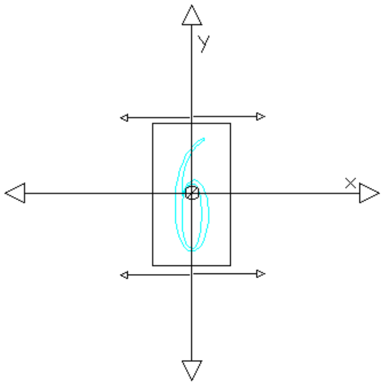
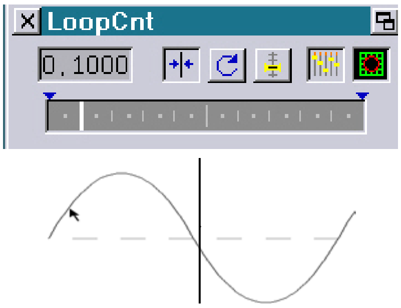


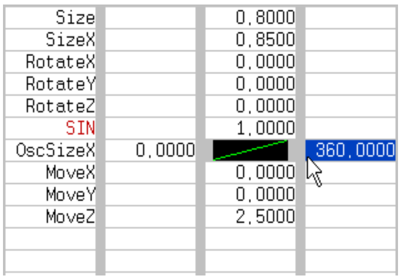
Fig. OscSizeX 2

Curve from 0 – 360



0,1000

Fig. OscSizeX 3



Size		0,8000	
SizeX		0,8500	
RotateX		0,0000	
RotateY		0,0000	
RotateZ		0,0000	
SIN		1,0000	
OscSizeX	0,0000		360,0000
MoveX		0,0000	
MoveY		0,0000	
MoveZ		2,5000	

Fig. OscSizeX 1

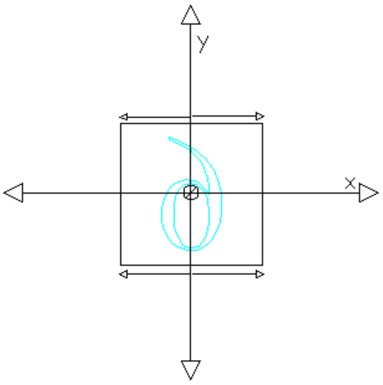
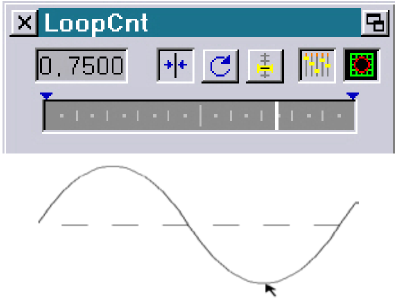


Fig. OscSizeX 4

Curve from 0 – 360



0,7500

Fig. OscSizeX 5

OscSizeY:

OscSizeY

With this command you compress and/or stretch the graphic along the y-axis with the selected wave form.

The amplitude determines the maximum stretching of the graphic.

The default value of 1 stretches the graphic to its original height at the end point.

A higher value elongates the graphic vertically, a smaller value shortens it.

OscSizeZ:

OscSizeZ

With this command you compress and/or stretch the graphic along the z-axis with the selected wave form.

The amplitude determines the maximum stretching of the graphic.

If you process a two-dimensional picture - which usually will have been digitized in the x-y-plane with z-coordinates of 0 - without shifting the positions of some of the picture points on the z-axis through an x- or y-rotation **before** applying the `OscSizeZ`, nothing will happen. !!!

You can, however, influence the movement with a second `MoveZ` which you insert before the wave function and the `OscSizeZ`.

`Getactpos`

Getactpos

With this option you can take over the previous or next values from another InitPosition.

Displayed command line in the film:
`InitPosition`
Position in the program:
depending on the application
Keyboard shortcut: `ip`

3.8.4 ChangePosition

With the ChangePosition command you set a graphic in motion by changing the position defined with InitPosition.

The `ChangePosition` command can only be used in connection with `InitPosition`. It is entered after `InitPosition` and automatically adopts the parameters defined there and the position number.

→ Chapter 3.8.3 `InitPosition`

If you subsequently change or add parameters for `InitPosition` you only have to click into the *input* field of `ChangePosition` to the left afterwards and the number and order of the parameters will be adapted automatically.

Position #:

The `Position#` of `ChangePosition` has to be identical to the `Position#` of the corresponding `InitPosition` command. If you change the `Position#` for `InitPosition` afterwards you also have to change the one for `ChangePosition`.

There are two basic applications for this command.

1. You can define a motion sequence and the resulting end position within the time set in the `TimeLoop`. For this `InitPosition` has to be set within the time loop.

   Let us assume that you enter a 90 for `RotateX` (`Factor: LoopCnt`). The graphic will then turn around the x-axis by 90° in the time you defined in the `TimeLoop`. If the time is short the movement will be executed quickly, if it is long the rotation will be correspondingly slow.

   For this application the `ChangePosition` command has effectively become obsolete since it has become possible to already define the movement for `InitPosition` using a *Curve*.

→ *Curve Editor*

→ Chapter 3.8.3 `InitPosition`

Factor:

`Factor`

The values you enter for `ChangePosition` are multiplied by the `Factor` and then added to the values from `InitPosition`.

The default setting is LoopCnt. Factor LoopCnt

LoopCnt starts at 0 and ends at 1.

Let us assume you entered a 90 for RotateX in the ChangePosition and a 30 in the InitPosition.
The 90 would first be multiplied by 0 and then added to 30. The result is 30, i.e. we start at the position defined in the InitPosition.

At the end of the loop the LoopCnt has reached 1.
We therefore calculate:

90x1+30=120

The graphic has rotated smoothly around the x-axis by 90° to its end position of 120° during the duration of the loop.

Other reasonable choices for this application of the ChangPosition command are ATTACK0(), DECAY0(), or a *curve*. → *Curve Editor*

ATTACK0()and DECAY0() are predefined curves you cannot change.

However, with DECAY0() you exchange the start- and end positions because DECAY0() starts at 1 and ends at 0. **!!!**

A constant value does not make sense for this application of ChangePosition, you might as well define the position with InitPosition.

2. The second possible application is the so called incremental method. Here the InitPosition is located outside the loop and with each cycle of the loop the current position is changed by the position change defined in the ChangePosition multiplied by the factor. In this case the factor usually is a constant value between -1 and +1. Factor: –1 to +1
This creates a movement speed which is independent of the duration of the loop, which however is not constant but depends on several factors, such as the complexity of the graphic and the scanning speed.

As an alternative to a constant value you can also enter a variable. Factor: Variable

Either you choose one of the predefined variables such as Analog, LoopStep..., or you define your own Variable in SetVariables. → Chapter 3.12 SetVariables

Using the variables and the internal time it is also possible to achieve a movement speed which is actually constant.
For this you define the following variables:

!!!
Constant Movement Speed

1. *Variable*: (outside the time loop)
   old_time = Seconds()

2. *Variable*: (within the time loop)
   time = Seconds();
   dt = time – old_time;
   old_time = time

With the additional command SetVAR you can also adjust the movement speed continuously with the mouse poti.

→ Chapter 3.12 SetVAR

In the trickfilm this looks as follows:

	Entry
	InitPosition
old_time = Seconds()	SetVariables
	TimeLoop
Variable Value Speed 1.000	SetVAR

time = Seconds();
dt = time – old_time;
old_time = time
Factor = dt * Speed
	SetVariables

Factor: Factor RotateZ 360.000	ChangePosition
	DisplayPicture
	LoopEnd

In this example the graphic would execute a rotation around the z-axis in exactly one second.

You can enter the time for the movement at SetVAR-Value by mouse poti or keyboard.

If you enter 1.0000 the movement will be executed in exactly one second, higher values increase and lower values decrease the movement speed.

Entering negative values will reverse the direction of the movement.

Displayed command line in the film:
ChangePosition
Position in the program:
Within the time loop
Keyboard shortcut: cp

3.9 Picture Output

3.9.1 DisplayPicture

This command causes the display of a single graphic whose position, color and possible movement you previously defined with commands of the *Point Of View Definitions*, *Color- and Display Control*.

`Catalog:`

Here you define from which catalog you want to select a graphic. Simply click on `Catalog:` or on the empty field with the ? to the right of it. → `File`

A new window will open with the heading:
`Catalogs in memory listed, pick name.`
Below you will see how much memory you have left in the `Picturememory`.

All catalogs which you have already loaded into the picture memory are listed here. Click on the desired catalog. Immediately a window will open: `Pictures in catalog xxx:` (name of the selected catalog)

Now click on the desired graphic.
Afterwards you will again see the input window of `DisplayPicture`.

If the required catalog has not been loaded into the picture memory yet, click on the field with the middle mouse button. You are then on the File level and you can search the hard drive for the desired catalog. By clicking on the catalog all graphics of this catalog are automatically listed and you can load the desired picture directly from the hard drive into the memory and the trickfilm.
Another option to select a picture is to open the `Browser` with `F7` and "browse" through the memory or the directories on the partitions.

`Picture:`

If you realize that you need another graphic from the previously selected catalog click on the field with the displayed graphic or on `Picture:`.
Now you again see all graphics of the catalog listed and you can select one by clicking on it.

Matrix 0-7 set by Pos#:

Each graphic has at least one matrix number and can have up to eight (0-7). This you define in the *picture editor* Piced.

→ Piced

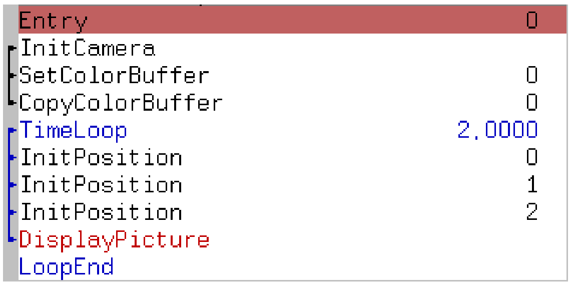
You have to assign a position list to each picture element with a different matrix number.

The advantage is that you can assign these picture elements different position lists and thereby place and move them independently.

In the following example the individual words of the graphic 'do what you want' were each digitized with a different matrix number. You can now assign each word a different position and movement.

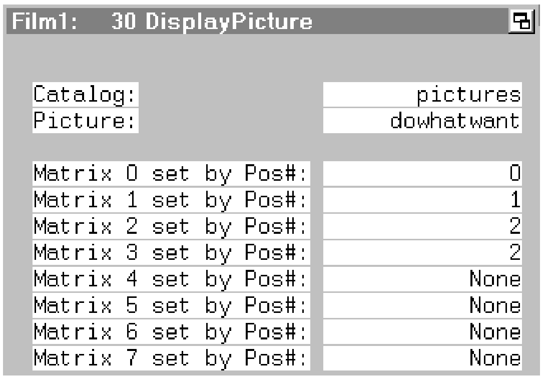
In our example we only want to move three words differently. We therefore need three position lists but only one DisplayPicture (see fig. DisplayPicture 1).

The position numbers are assigned to the corresponding matrix# at DisplayPicture (see fig. DisplayPicture 2).



```
Entry                   0
InitCamera
SetColorBuffer          0
CopyColorBuffer         0
TimeLoop           2.0000
InitPosition            0
InitPosition            1
InitPosition            2
DisplayPicture
LoopEnd
```

Fig. DisplayPicture 1



Film1: 30 DisplayPicture

Catalog:	pictures
Picture:	dowhatwant
Matrix 0 set by Pos#:	0
Matrix 1 set by Pos#:	1
Matrix 2 set by Pos#:	2
Matrix 3 set by Pos#:	2
Matrix 4 set by Pos#:	None
Matrix 5 set by Pos#:	None
Matrix 6 set by Pos#:	None
Matrix 7 set by Pos#:	None

Fig. DisplayPicture 2

We assigned the position list 0 to the word 'do' with the matrix# 0, position list 1 to the word 'what' with the matrix# 1 and position list 2 to the words 'you' with matrix# 2 as well as 'want' with matrix# 3.

The word 'do' does not move, we zoom in to the word 'what' and back out again using a *curve*, 'you want' rotates back and forth around the y-axis (see fig. DisplayPicture 3 + 4).

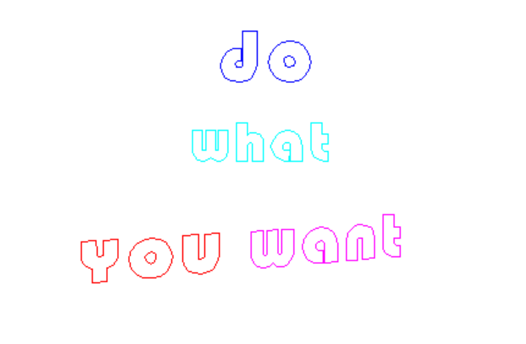


Fig. `DisplayPicture` 3

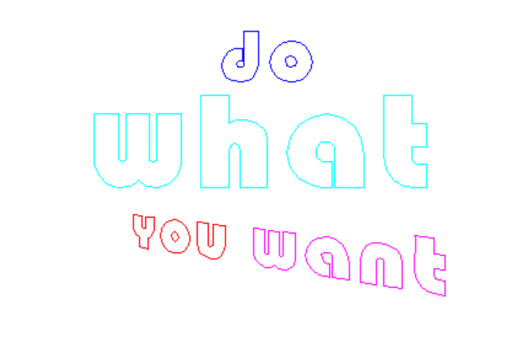


Fig. `DisplayPicture` 4

Displayed command line in the film:
`DisplayPicture`
Position in the program:
Within the time loop after commands of the point of view definition and the color and display control
Keyboard shortcut: `dp`

3.9.2 DisplayBuffer

This command causes the display of a single graphic from one of the eight *Picture Buffers*. Generally this command is similar to the `DisplayPicture` command, however here you do not choose a graphic from a catalog but rather specify the number of the *picture buffer* into which you previously copied the desired graphic. *Buffer* operations are necessary if for instance you want to transform or morph a graphic or want it to be drawn.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Buffer:`

Here you specify the number of the *buffer* which contains the picture you want to project (0-7).

`Transform-par.number:`

If you perform a transformation on a graphic or have already performed it, enter the `Transform-par.number` (0-3) here. Otherwise keep the default value `none`.

→ Chapter 3.11 `Transformations`

`Matrix 0 (-7) set by Pos#:`

Here you determine which matrix number is controlled by which position list. You can find detailed information in the previous chapter `DisplayPicture`.

→ Chapter 3.9.1 `DisplayPicture`

Displayed command line in the film:
`DisplayBuffer` and number of the *buffer*
Position in the program:
Within the time loop like `DisplayPicture`
Keyboard shortcut: `db`

3.9.3 DisplayAnimation

This command causes the display of an animation just as `DisplayPicture` causes the display of an individual graphic. Before this, however, you have to create an animation with the `AnimationEditor`.

→ `AnimationEditor`

`Animation:`

Here you enter the name of the desired animation. When you click on the field a new window will open listing all loaded animations. Click on the desired animation and you will immediately see the input window with the corresponding entry. Should this window be empty click on the field with the middle mouse button. You are then in the File level where you can search for the desired animation. By clicking on it it is automatically loaded and entered under `Animation:`.

`Stream from file:`

This allows you to play an animation without loading it to the memory. This is usefull if the animation consists of a large amount of frames (e.g. 10.000).

☞ Load all animations into memory, save all as a CompressedCat (CCAT), then choose this CCAT. Finally, unload all unused Catalogs.

Notes:

- It is possible to organize the animations in more than one CCAT. However, it is advisable to use one CCAT because opening and closing the individual CCATs takes additional time.
- It is advisable to enter the CCAT in the "DisplayAnimation" command without specifying a path; the system will then first search for the CCAT in the same directory as the loaded show. If there is no CCAT present then it will also search the directories "C:\General" and "C:\General\Streams".
- It is also possible to enter absolute paths as streaming source to enforce streaming from a specific place (e.g. a certain server in the network).

`Frames:`

The number of individual pictures used in the selected animation is automatically displayed here.

Duration:

If an Animation was created in 3Dstudio Max and converted with proTize max, you will get the information about the original length of that animation.

Mode:

Here you can choose between `Relative`, `Absolute`, `Dynamic` and `Timed`.
The default setting is `Relative`. This, in connection with `LoopCnt` for the following parameter, is probably the most commonly used setting because it leads to a smooth progression of the animation in the time defined in the `TimeLoop`.

In the `Absolute Mode` you enter constant values for the following parameter whose value is equivalent to the number of the frame of the animation. So 0 stands for the first picture, 1 for the second, 2 for the third picture of the animation and so on.

"`Dynamic`" and "`Timed`" can be used with TimeLoop and ExternTimeLoop for animations created with proTize max's dynamic rendering. ILDA animations which were not created for a fixed picture rate can also be displayed better with these modes.

Number of picture:

The entry here depends on the selected `Mode`.

The default setting is `LoopCnt` in connection with the `Relative` mode.

`LoopCnt` has the effect that the animation is displayed <u>once</u> smoothly from 0 to 1 during the time specified in the `TimeLoop` or controlled by the `ExternTimeLoop`.

If you want the animation to be executed more than once during the time loop it makes most sense to define a *variable*.

→ Chapter 3.12 *Variables*

You can give whatever names you like to *variables* which you create yourself, but the names of functions are preassigned in the program (e.g. "Frac").

Let us simply call the *variable* 'AnimLoop'.

You program as follows:

```
TimeLoop
SetVariables            AnimLoop = Frac (Loopcnt *4)
InitPosition
DisplayAnimation
   Mode                 Relative
   Number of picture    AnimLoop
LoopEnd
```

The animation will then be executed four times during the time loop. If you write "Frac (Loopcnt*2)" only twice etc.

You can also imitate the *variable* with a *curve*.
If you only want a small number of repetitions this is no problem. Otherwise it becomes complex, and changing the number of loops is easier with a *variable* than with a *curve*.

→ *Curve Editor*

When you enter a constant value in the `Relative Mode` the 0 stands for the first picture, the 1 for the last picture in the animation.

However, in the Absolut Mode the entered number is equivalent to the `Frame` of the animation. Therefore 0 stands for the first picture, 1 for the second, 2 for the third picture of the animation and so on.

Matrix 0-7 set by Pos#:

Here you determine which matrix number is controlled by which position list. You can find detailed information in the chapter `DisplayPicture`.

→ Chapter 3.9.1 `DisplayPicture`

Displayed command line in the film:
`DisplayAnimation`
Position in the program:
Within the time loop like `DisplayPicture`
Keyboard shortcut: `da`

3.9.4 DeFineText

→ Chapter 3.9.5 RunningText

The DeFineText command is an enhancement of the running text command and can only be used in connection with RunningText.

However, a text entered under DeFineText will only become effective if there is no text entered for the RunningText command.

In the trickfilm DeFineText is used as follows:

```
  Entry
  DeFineText
  TimeLoop
  RunningText (without Text)
  LoopEnd
```

DeFineText allows you to work with texts in a show and to change them without manipulating the trickfilm.

That means that you can even change texts if you have no access to the trickfilm, for instance because it was protected and saved with /NoEdit.

You can even change a text while the show is running.

Using DeFineText is always useful when a show contains text sections like names, dates, venues etc. which you may want to resp. have to update or change.

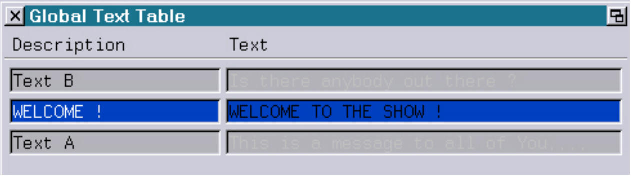
While a show is running you see [GlobalTEXT] in the main menu in the *menu bar* at the bottom - click on it or press F7.

When the film is not running you can find [GlobalTEXT] by clicking on [more].

A new window will then open called Global Text Table (see fig. GlobalText 1). There you see all texts that were programmed with DeFineText and then saved and loaded again with the trickfilm.

LGFD

In the LGFD you open this window by clicking on the button 'Text' which always appears if DeFineText was used in the loaded show.



Global Text Table

Description	Text
Text B	Is there anybody out there ?
WELCOME !	WELCOME TO THE SHOW !
Text A	This is a message to all of You...

← Default text is displayed

← Text was changed with Global text

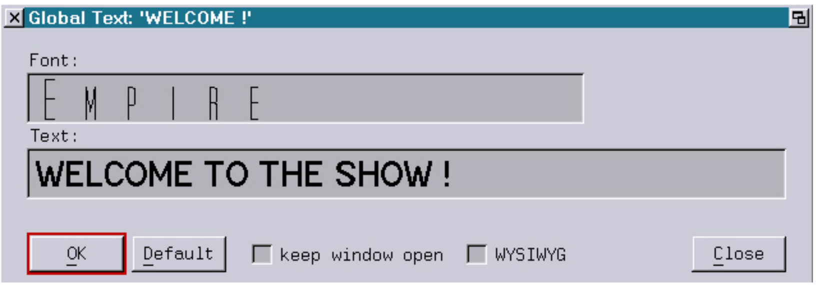
Fig. GlobalText 1

In our example above (fig. GlobalText 1) three different texts were programmed with DeFineText in the loaded trickfilm.

In the column Description you see the name under which the text section was programmed in the trickfilm.

In the column next to it you see the text which is displayed.

If you want to change a text click on the corresponding line. The Global Text window will open.



Global Text: 'WELCOME !'

Font: EMPIRE

Text: WELCOME TO THE SHOW !

Fig. GlobalText 2

In this window you can exchange the default text for another text as well as change the font or reset a changed text back to the default text.

As soon as the default text was changed in GlobalText, the text line will appear with a different color (see fig. GlobalText 1) in the Global Text Table until you click on Default again in the Global Text window.

Changes will be saved to the hard disk under C:\System\GlobalTexts, regardless of the loaded show. They will be called up in any film containing a DeFineText with the corresponding Description. Therefore you always automatically load the latest change.

Animation:

Animation

You can also choose an animation instead of a text and use the RunningText command to position, move and display it. The individual frames will then be displayed in sequence instead of an entered text.

→ Chapter 3.9.5 RunningText

Animations cannot be changed subsequently with GlobalText.

Frames:

Frames

If you selected an animation this will show how many frames (pictures) this animation contains.

AutoSpace:

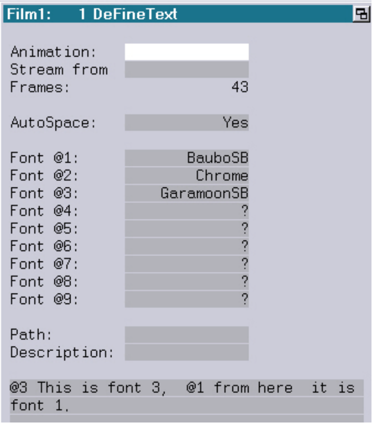
AutoSpace

The default setting of this parameter is **Yes**.
This will cause the appropriate number of blank spaces to be inserted before and after the text so that the text appears in the picture completely and then disappears completely.

Font@1-@9:

Font@1-@9

With DeFineText you can use up to nine different fonts per text.
When you click on the individual fonts a window will open listing all loaded fonts. If you have not yet loaded any or if the desired font is not listed click on the window twice with the middle mouse button. You then are in the File level in the Generals directory and you can see all available fonts. By simply clicking on it the desired font is loaded and copied to the selected font number of DeFineText.
To use different fonts in the same text, use the definition @1-9 before typing the text (see the example below).



Film1: 1 DeFineText

Animation:	
Stream from	
Frames:	43
AutoSpace:	Yes
Font @1:	BauboSB
Font @2:	Chrome
Font @3:	GaramoonSB
Font @4:	?
Font @5:	?
Font @6:	?
Font @7:	?
Font @8:	?
Font @9:	?
Path:	
Description:	

@3 This is font 3, @1 from here it is font 1.

Path:

Path

The parameter "Path" is used to specify the directory where the altered texts will be saved. If no directory is specified here the directory "C:\System\GlobalTexts" will be used. It is a good idea to leave it that way!

Description:

Description

Here you give the following text a name so that you can later find it in the Global Text Table.

Text field:

The following lines are meant for your text. Click on the field and start writing. At first you will only see the text in the *input* line at the bottom. It is copied to the text field as soon as you confirm the entry with [ok] or the return key.

If you want to change something click on the text field again and you will see the text in the *input* line again. You can then click on the desired spot with the mouse and write again. Confirmation: Do not forget [ok], F1 or return!

→ Chapter 2. 3
Input Methods

If you program a running text adapt the duration of the loop to the text. The longer the text the more time it requires to run through the picture legibly.

When using DeFineText, later changes can cause the length of the text to differ largely from the default text. This can have the effect that the scrolling speed becomes too fast or too slow. In the worst case the text will rush through the picture so fast that it cannot be read or it crawls so slowly that the viewers will have forgotten the first word by the time the second one appears.

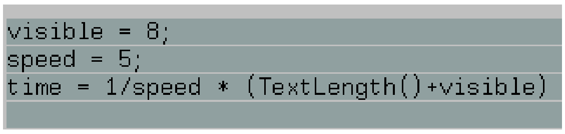
You can prevent this by using *variables* to adapt the length of the loop to the length of the text.

→ Chapter 3.12
SetVariables

In the SetVariables you can use the function "TextLength()" to find out the length of the text which was set with DeFineText and which is to be displayed with RunningText.

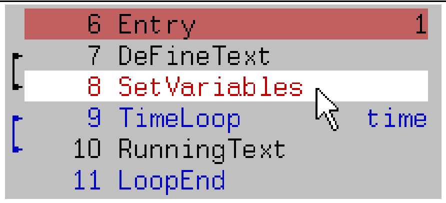
If you entered 'Yes' for AutoSpace in the DeFineText you have to take into account the number of visible characters for the formula.

In the trickfilm the whole thing would look like this:



```
visible = 8;
speed = 5;
time = 1/speed * (TextLength()+visible)
```

Fig. GlobalText 3



```
 6 Entry              1
 7 DeFineText
 8 SetVariables
 9 TimeLoop          time
10 RunningText
11 LoopEnd
```

Fig. GlobalText 4

In our example the number of visible characters is 8 (visible = 8), i.e. there are eight characters at the most in the picture at any time.

The scrolling speed is 5 characters per second (speed = 5).

For TimeLoop, instead of a number you enter the *variable* 'time'.

For this you click on Time in seconds and then on <Variable> in the *menu bar* at the bottom.
A window listing all available *variables* will open.
Among others you should see visible, speed, and time there. Click on the *variable* time.

Of course this only makes sense if the *entry* containing the text display is not time code controlled.

Displayed command line in the film:
DeFineText
Position in the program:
Outside the time loop
Keyboard shortcut: dft

3.9.5 RunningText

With RunningText you can combine letters, numerals and symbols from up to three fonts at will and project them as running text.

You define the text and the way it is to run through the picture with the parameters explained below.

pos of 1st

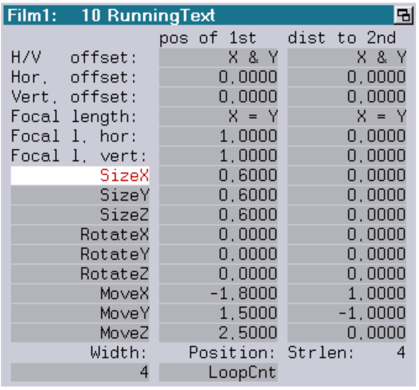
pos of 1st

Everything you enter in this column determines the position, size and possible rotation of the whole text if the dist to 2nd value of the corresponding position is 0. The first character keeps this position, regardless of the position changes made under dist to 2nd.

dist to 2nd

dist to 2nd

With the parameters in this column you determine the position, size and rotation of the second character in relation to the first one. The following characters are positioned automatically in relation to this.



Film1: 10 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	0.0000	0.0000
Vert. offset:	0.0000	0.0000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	0.0000
Focal l. vert:	1.0000	0.0000
SizeX	0.6000	0.0000
SizeY	0.6000	0.0000
SizeZ	0.6000	0.0000
RotateX	0.0000	0.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	-1.8000	1.0000
MoveY	1.5000	-1.0000
MoveZ	2.5000	0.0000
Width:	Position:	Strlen: 4
4	LoopCnt	

Fig. RunningText 1

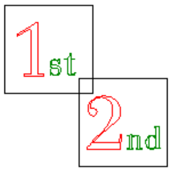


Fig. RunningText 2

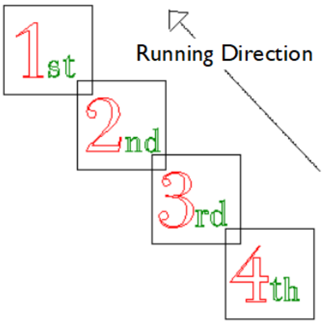


Fig. RunningText 3

In the example above (fig. RunningText 1-3) we moved the first letter along the x-axis to the left (MoveX -1.8000) and upwards along the y-axis (MoveY 1.5000). We shifted the second letter to the right by 1.000 and downwards by -1.000.

Fig. RunningText 3 illustrates that the projection of all additional letters is shifted in the same ratio.

H/V offset: X & Y

H/V offset

The offset parameters are equivalent to those for InitCamera.

→ Chapter 3.8.1 InitCamera

You can use them to shift the text horizontally or vertically without the angle distortion you would get with MoveX and MoveY.

The H/V offset: X & Y practically unites the following two. By clicking on the X & Y field with the left mouse button and keeping the mouse button pressed you can use the *mouse poti* to set the vertical and horizontal offset simultaneously.

If you press the middle mouse button instead of the left one a finer adjustment is possible.

Hor. offset:

Hor. offset

pos of 1st

Here you set the horizontal position of the text. The first character retains this position regardless of the position changes at dist to 2nd.

dist to 2nd

Here you define the horizontal position change of the second character in relation to the first one as well as that of all following characters to each other.

Vert. offset:

Vert. offset

pos of 1st

Here you set the vertical position of the text. The first character retains this position regardless of the position changes at dist to 2nd.

dist to 2nd

Here you define the vertical position change of the second character in relation to the first one as well as that of all following characters to each other.

From here on we will no longer differentiate between pos of 1st and dist to 2nd. We will only discuss the position changes in general because the difference between the two position lists is always the same.

Focal length: X = Y

Focal length

→ H/V offset

Here you can influence the focal length with which the text is to be projected, giving the projection an appearance of spatial depth.

This parameter, like the H/V offset, is a combination of the following two.

By clicking on the X = Y field with the left mouse button and keeping the mouse button pressed you can use the *mouse poti* to set the vertical and horizontal focal length simultaneously.

If you press the middle mouse button instead of the left one a finer adjustment is possible.

The figures RunningText 4 to 9 illustrate the effect of a change of the Focal length.

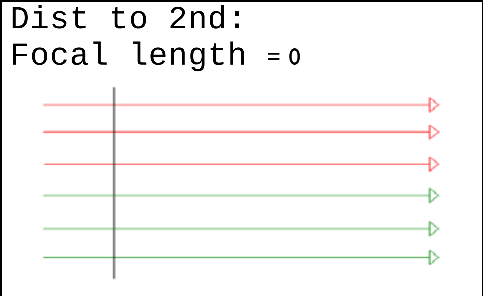


Fig. RunningText 4

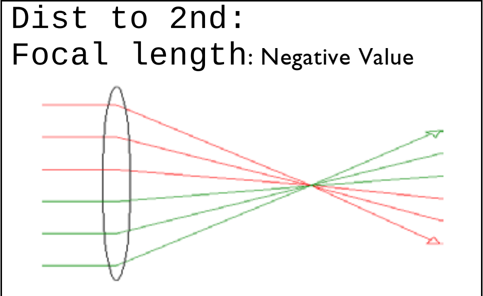
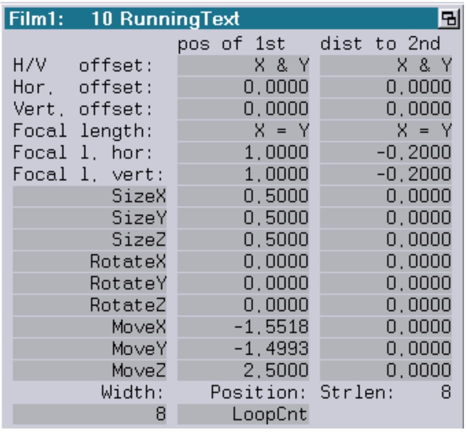


Fig. RunningText 5



Film1: 10 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	0.0000	0.0000
Vert. offset:	0.0000	0.0000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	-0.2000
Focal l. vert:	1.0000	-0.2000
SizeX	0.5000	0.0000
SizeY	0.5000	0.0000
SizeZ	0.5000	0.0000
RotateX	0.0000	0.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	-1.5518	0.0000
MoveY	-1.4993	0.0000
MoveZ	2.5000	0.0000
Width:	Position:	Strlen: 8
8	LoopCnt	

Fig. RunningText 6

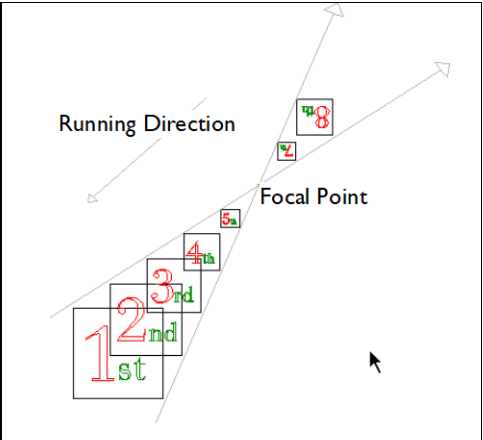
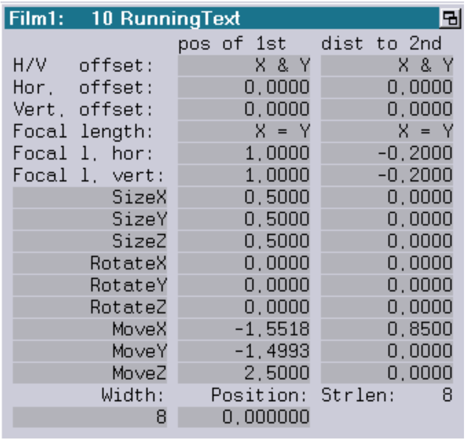


Fig. RunningText 7



Film1: 10 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	0.0000	0.0000
Vert. offset:	0.0000	0.0000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	-0.2000
Focal l. vert:	1.0000	-0.2000
SizeX	0.5000	0.0000
SizeY	0.5000	0.0000
SizeZ	0.5000	0.0000
RotateX	0.0000	0.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	-1.5518	0.8500
MoveY	-1.4993	0.0000
MoveZ	2.5000	0.0000
Width:	Position:	Strlen: 8
8	0.000000	

Fig. RunningText 8

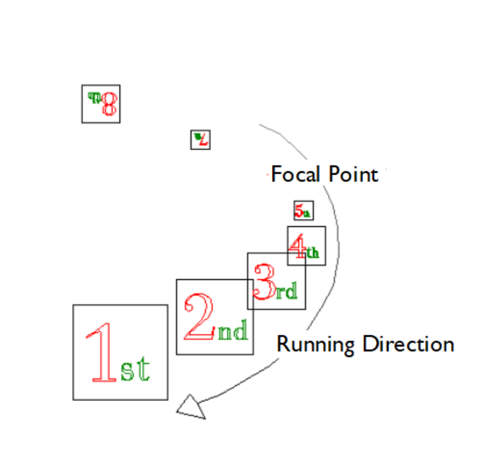


Fig. RunningText 9

Focal l.hor:

Focal l.hor

Here you can change the horizontal focal length, the vertical one remains unchanged.

Focal l.vert:

Focal l.vert

Here you can change the vertical focal length, the horizontal one remains unchanged.

SizeX:

SizeX

With this you change the size of the characters in the direction of the x-axis. This means you can make the characters wider or narrower independent of their height.

SizeY:

SizeY

Here you influence the height of the characters, i.e. you change the size in the direction of the y-axis.

SizeZ:

SizeZ

This parameter increases or reduces the size of the characters in the z-plane.

Usually the fonts will have been digitized two-dimensionally with a z-coordinate of 0. You will only see a change with this parameter if the font consists of three-dimensional characters or if the z-coordinate is not 0 for two-dimensional characters.

RotateX:

RotateX

Here you can rotate the characters around the x-axis.
The entered numbers are equivalent to degree numbers.
The value you enter for dist to 2nd rotates the second character and all the following by this value, each one in relation to the character preceding it.

An example:

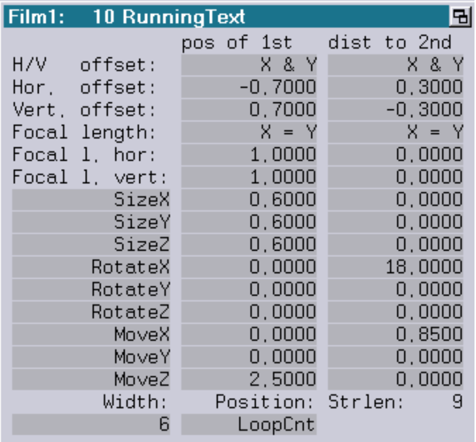
You display six characters at the same time (Width: 6),
the pos of 1st value is 0,
you enter 18.0 for dist to 2nd,
the second character will then be rotated by 18° counterclockwise around the x-axis.

The third character has the 18° rotation and is rotated by another 18°, i.e. 36° in relation to the first character.

The last character will therefore be rotated by 90°.

If the sixth character is to be displayed exactly as a line with the rotation as shown in fig. RunningText 11, vertical shifts of the characters have to be entered with the value Vert. offset because then they do not have the angle distortion they would get with MoveY.

→ Chapter 3.8.3 InitPosition



Film1: 10 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	-0.7000	0.3000
Vert. offset:	0.7000	-0.3000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	0.0000
Focal l. vert:	1.0000	0.0000
SizeX	0.6000	0.0000
SizeY	0.6000	0.0000
SizeZ	0.6000	0.0000
RotateX	0.0000	18.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	0.0000	0.8500
MoveY	0.0000	0.0000
MoveZ	2.5000	0.0000
Width:	Position:	Strlen: 9
6	LoopCnt	

Fig. RunningText 10

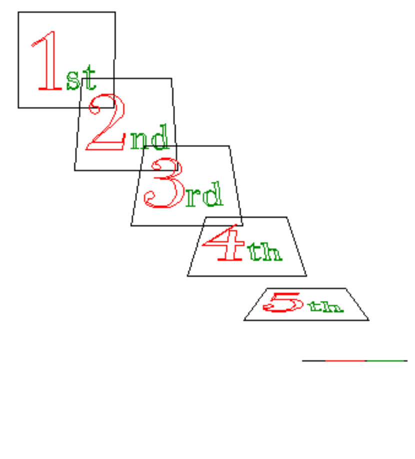


Fig. RunningText 11

RotateY:

RotateY

Here you can rotate the characters around the y-axis.
The entered numbers are equivalent to degree numbers.
The value you enter for dist to 2nd rotates the second character and all the following by this value, each one in relation to the character preceding it.

→ also see RotateX

RotateZ:

RotateZ

Here you can rotate the characters around the z-axis.
The entered numbers are equivalent to degree numbers.
The value you enter for dist to 2nd rotates the second character and all the following by this value, each one in relation to the character preceding it.

→ also see RotateX

MoveX:

MoveX

With MoveX you shift the text along the x-axis to the desired position.
With pos of 1st you position the first character, the entry for dist to 2nd determines the distance between the individual letters.

MoveY:

MoveY

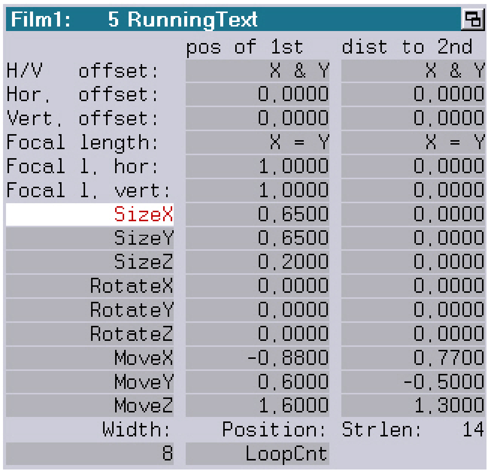
Here you can shift the text along the y-axis.
With dist to 2nd you determine the vertical shift of the second character and of all the following in relation to each other.

MoveZ:

MoveZ

With this parameter you can shift the characters of your running text along the z-axis and so create an impression of spatial depth.

The higher the value you enter the farther the characters will move from the viewer, i.e. they will get optically smaller.
If the distance to 2nd value is positive the text will get smaller starting with the second character (fig. RunningText 12-13). If, however, you enter a negative value it will get bigger (fig. RunningText 14-15).



Film1: 5 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	0.0000	0.0000
Vert. offset:	0.0000	0.0000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	0.0000
Focal l. vert:	1.0000	0.0000
SizeX	0.6500	0.0000
SizeY	0.6500	0.0000
SizeZ	0.2000	0.0000
RotateX	0.0000	0.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	-0.8800	0.7700
MoveY	0.6000	-0.5000
MoveZ	1.6000	1.3000
Width:	Position:	Strlen: 14
8	LoopCnt	

Fig. RunningText 12

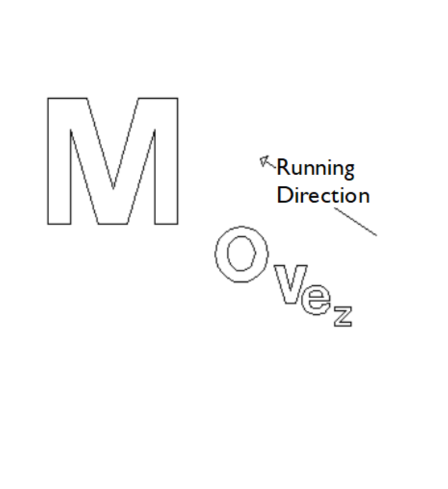
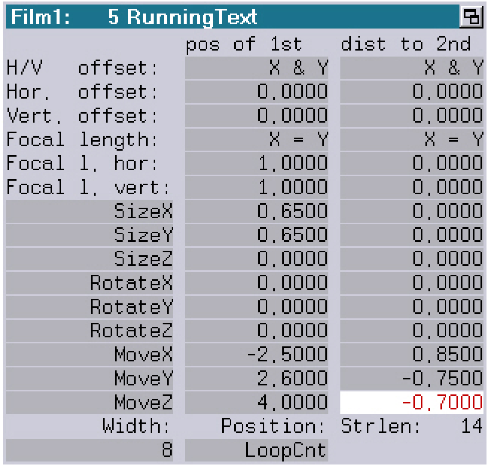


Fig. RunningText 13



Film1: 5 RunningText

	pos of 1st	dist to 2nd
H/V offset:	X & Y	X & Y
Hor. offset:	0.0000	0.0000
Vert. offset:	0.0000	0.0000
Focal length:	X = Y	X = Y
Focal l. hor:	1.0000	0.0000
Focal l. vert:	1.0000	0.0000
SizeX	0.6500	0.0000
SizeY	0.6500	0.0000
SizeZ	0.0000	0.0000
RotateX	0.0000	0.0000
RotateY	0.0000	0.0000
RotateZ	0.0000	0.0000
MoveX	-2.5000	0.8500
MoveY	2.6000	-0.7500
MoveZ	4.0000	-0.7000
Width:	Position:	Strlen: 14
8	LoopCnt	

Fig. RunningText 14

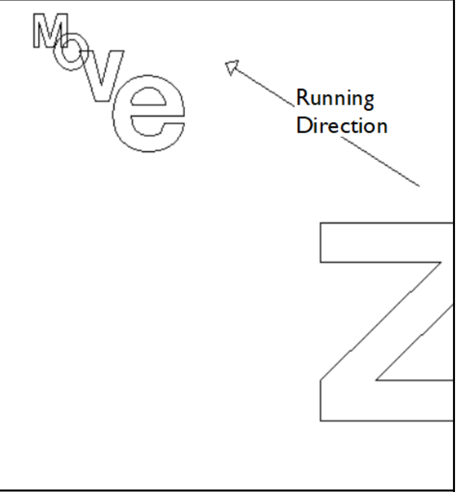


Fig. RunningText 15

Width:

Width

Here you enter the maximum number of characters that are to be visible simultaneously. The actual number of characters visible in the picture at the same time depends on the size, distance and positioning of the individual characters.

Position:

Position

At Position you enter at which point resp. from where to where the text is to be displayed.
The default setting LoopCnt has the effect that the text runs once during the time loop from position 0, i.e. the starting position, to position 1 (end position).

Alternatively you can enter a *curve*, a *variable* or a fixed value.

When you enter a fixed value (0-1) the running text or a part of it will be displayed as a stationary picture.
You can slide back and forth between 0 and 1 with the *mouse poti* and so easily find the right position.

You will then see the actual characters and not just the substituting squares without the text actually running.

Like in the `InitPosition`, it is possible to change order of the parameters to change the behaviour of movements.
→ `InitPosition` 3.8.3

`Strlen:` `Strlen`

The number you enter here states the number of characters making up the entered text.

`<Font @1-3>` `<Font @1-3>`

Here you specify the font/s you want to use for your running text.
You have to enter at least one font unless you do not want to enter any text at all because you already did so with the `DeFineText` command.
When you click on one of the three fields the window with the `Picturememory` will automatically open and you will see all catalogs that were already loaded.
If the desired font was not yet loaded you can reach the `Generals Directory` by clicking twice with the middle mouse button. There you can select a font from the hard drive by simply clicking on it.

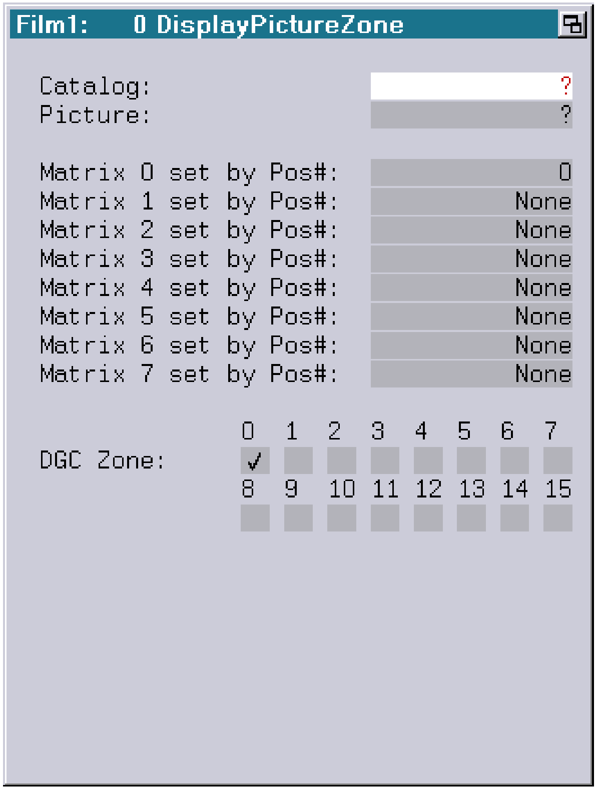
Displayed command line in the film:
`RunningText`
Position in the program:
Within the time loop
Keyboard shortcut: `rt`

3.9.6 DisplayPictureZone

This command `DisplayPictureZone` is an enhancement of the command `DisplayPicture`.

→ Chapter 3.9.1
`DisplayPicture`

The addition is the specification of the desired projection zones where the picture is displayed.



Film1: 0 DisplayPictureZone

Catalog:	?
Picture:	?
Matrix 0 set by Pos#:	0
Matrix 1 set by Pos#:	None
Matrix 2 set by Pos#:	None
Matrix 3 set by Pos#:	None
Matrix 4 set by Pos#:	None
Matrix 5 set by Pos#:	None
Matrix 6 set by Pos#:	None
Matrix 7 set by Pos#:	None

	0	1	2	3	4	5	6	7
DGC Zone:	☑	□	□	□	□	□	□	□
	8	9	10	11	12	13	14	15
	□	□	□	□	□	□	□	□

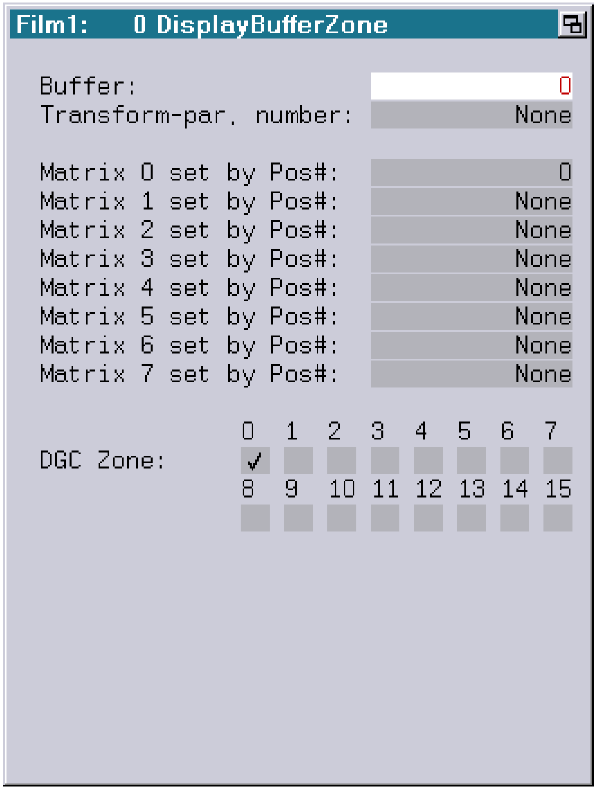
Displayed command line in the film:
`DisplayPictureZone`
Position in the program:
Within the time loop after commands of the point of view definition and the color and display control
Keyboard shortcut: `dpz`

3.9.7 DisplayBufferZone

This command `DisplayBufferZone` is an enhancement of the command `DisplayBuffer`.

→ Chapter 3.9.2 `DisplayBuffer`

The addition is the specification of the desired projection zones where the buffer is displayed.



Film1: 0 DisplayBufferZone

Buffer:	0
Transform-par. number:	None
Matrix 0 set by Pos#:	0
Matrix 1 set by Pos#:	None
Matrix 2 set by Pos#:	None
Matrix 3 set by Pos#:	None
Matrix 4 set by Pos#:	None
Matrix 5 set by Pos#:	None
Matrix 6 set by Pos#:	None
Matrix 7 set by Pos#:	None

	0	1	2	3	4	5	6	7
DGC Zone:	☑	□	□	□	□	□	□	□
	8	9	10	11	12	13	14	15
	□	□	□	□	□	□	□	□

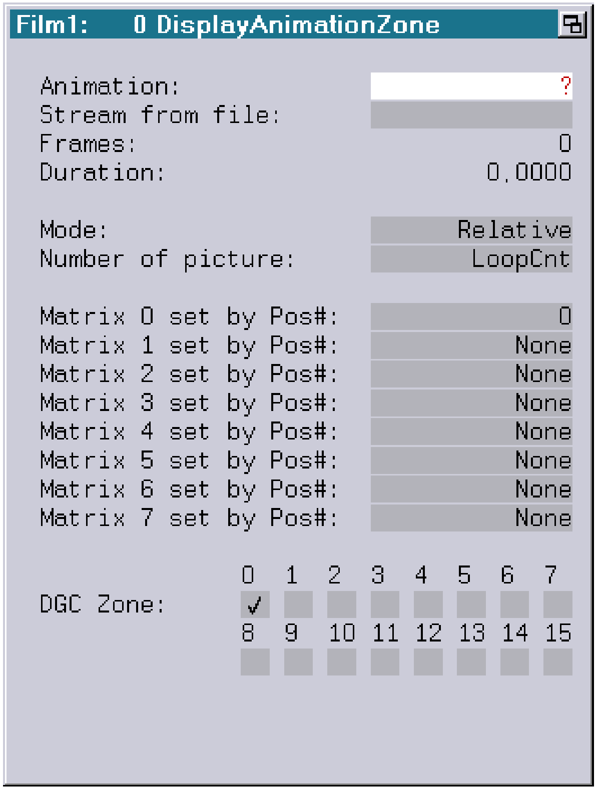
Displayed command line in the film:
`DisplayBufferZone` and number of the *buffer*
Position in the program:
Within the time loop like `DisplayPicture`
Keyboard shortcut: `dbz`

3.9.8 DisplayAnimationZone

This command DisplayAnimationZone is an enhancement of the command DisplayAnimation.

→ Chapter 3.9.3 DisplayAnimation

The addition is the specification of the desired projection zones where the animation is displayed.


Film1: 0 DisplayAnimationZone

Animation:	?
Stream from file:	
Frames:	0
Duration:	0.0000
Mode:	Relative
Number of picture:	LoopCnt
Matrix 0 set by Pos#:	0
Matrix 1 set by Pos#:	None
Matrix 2 set by Pos#:	None
Matrix 3 set by Pos#:	None
Matrix 4 set by Pos#:	None
Matrix 5 set by Pos#:	None
Matrix 6 set by Pos#:	None
Matrix 7 set by Pos#:	None

DGC Zone:

0	1	2	3	4	5	6	7
☑	□	□	□	□	□	□	□
8	**9**	**10**	**11**	**12**	**13**	**14**	**15**
□	□	□	□	□	□	□	□

Displayed command line in the film:
DisplayAnimationZone
Position in the program:
Within the time loop like DisplayPicture
Keyboard shortcut: daz

3.9.9 Abstract

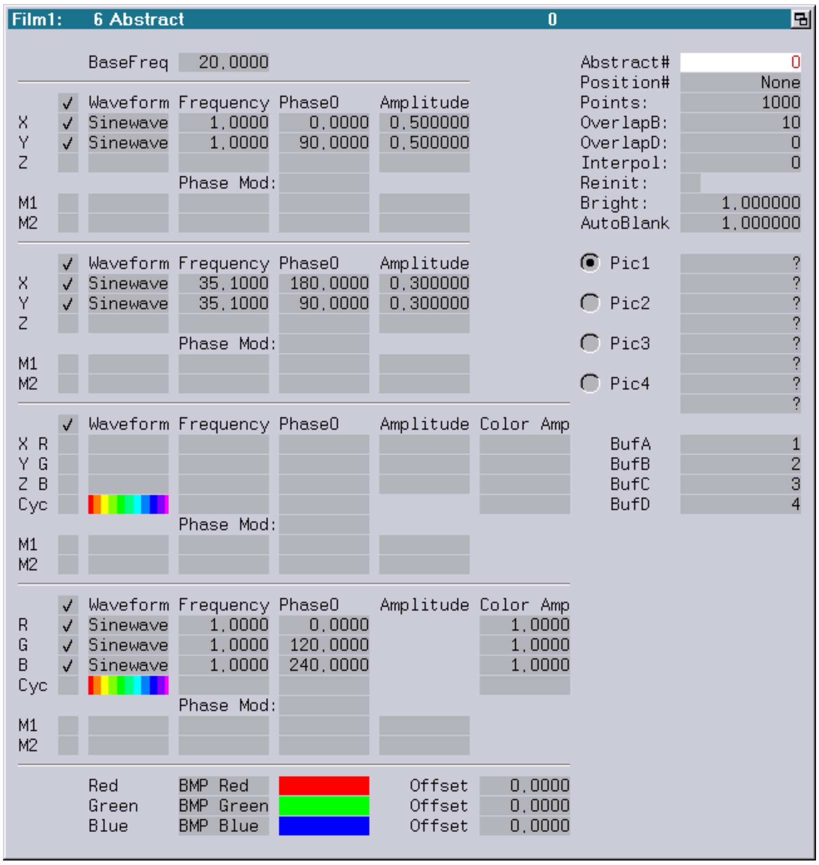
Abstract

The `Abstract` command generates "Analog calculated" pictures (Lissajous figures) by overlaying linear frequencies.

The abstract generator features 22 digital waveform generators grouped into 4 sets of triple oscillators, modified by two amplitude modulators and one phase modulator each.

Waveform, frequency, phase and amplitude can be chosen for every oscillator and modulator. The waveform can be sinewave, triangle, square, saw, or any frame. All other parameters can be fixed values, variables or curves.

Two of the four oscillator sets are graphical (geometrical) X-Y-Z oscillators, one is for R-G-B color modulation, and one can be used for graphical and/or color modulation. Additional color control is provided by a colorcycle editor (Cyc).



Film1: 6 Abstract 0

BaseFreq 20,0000

		Waveform	Frequency	Phase0	Amplitude
	✓	Waveform	Frequency	Phase0	Amplitude
X	✓	Sinewave	1,0000	0,0000	0,500000
Y	✓	Sinewave	1,0000	90,0000	0,500000
Z					
			Phase Mod:		
M1					
M2					

	✓	Waveform	Frequency	Phase0	Amplitude
X	✓	Sinewave	35,1000	180,0000	0,300000
Y	✓	Sinewave	35,1000	90,0000	0,300000
Z					
			Phase Mod:		
M1					
M2					

	✓	Waveform	Frequency	Phase0	Amplitude	Color Amp
X R						
Y G						
Z B						
Cyc		🟥🟧🟨🟩🟦🟪				
			Phase Mod:			
M1						
M2						

	✓	Waveform	Frequency	Phase0	Amplitude	Color Amp
R	✓	Sinewave	1,0000	0,0000		1,0000
G	✓	Sinewave	1,0000	120,0000		1,0000
B	✓	Sinewave	1,0000	240,0000		1,0000
Cyc		🟥🟧🟨🟩🟦🟪				
			Phase Mod:			
M1						
M2						

Red	BMP Red	🟥	Offset	0,0000
Green	BMP Green	🟩	Offset	0,0000
Blue	BMP Blue	🟦	Offset	0,0000

Abstract#	0
Position#	None
Points:	1000
OverlapB:	10
OverlapD:	0
Interpol:	0
Reinit:	
Bright:	1,000000
AutoBlank	1,000000
● Pic1	?
	?
○ Pic2	?
	?
○ Pic3	?
	?
○ Pic4	?
	?
BufA	1
BufB	2
BufC	3
BufD	4

Displayed command line in the film:
`Abstract`
Position in the program:
Inside the loop
Keyboard shortcut: a

3.9.10 InitAbstract

The `InitAbstract` command resets the Abstract.

Displayed command line in the film:
`InitAbstract`
Position in the program:
Outside the time loop
Keyboard shortcut: `ia`

3.9.11 ApplyBitmap

This command can be used to apply the color information of one ore more bitmaps to a laser picture before it is displayed.

Restrictions:

- The maximum resolution that is supported is 256 x 256 pixels.
- The number of bitmaps (or the length of an AVI file) that is supported depends on the memory available.

`Bitmaps:`

Here you enter the bitmap(s) to be used.
You can use a single bitmap or more than one by using “*.*”, "*.BMP", “*.DIB” or an AVI file.

The format "DIB" (device independent bitmap) is an alternative to the "BMP" format.

To enter an AVI file the following conditions have to be fulfilled:

- The LGServer has to be running and the PC on which the LGServer is runnning has to be specified as the server in the I/O Setup of the Mark 2.
- The AVI file can be located in the directory which is specified under “WAV/AVI” in the LGServer.

Alternatively an AVI file can be converted into a DVS file first, using the File command `COnvert.AVItoDVS`.
A “DVS” file can be entered without the LGServer.

`Frames:`

Shows the number of frames which will be used.

Width:

Displays the width of the bitmap(s).

Height:

Displays the height of the bitmap(s).

Mode:

The command "ApplyBitmap" supports two methods of handling cases where the bitmap no longer covers the complete laser picture due to reduced size or shifting:

`Wrap`	The bitmap is repeated indefinitely.
`Clip`	The bitmap is not repeated. Outside the bitmap the color of the edge of the bitmap is continued.

Additionally you can determine how the parameter `Select BMP` is interpreted:

`Relative`	The number of the bitmap used has to be in the range from 0.0 (first bitmap) to 1.0 (last bitmap).
`Absolute`	The number of the bitmap used has to be specified absolutely. 0 stands for the first bitmap, 1 for the second, 2 for the third and so on.

Select BMP:

The entry here depends on the selected `Mode` (see above).

The default setting is `LoopCnt` in connection with the `Relative` mode. This leads to a progression of the bitmaps in the time defined in the `TimeLoop`.

Buffer SRC: (Source Buffer)

Here you specify to which of the eight possible *buffers* (0-7) you want to apply the bitmap.

Default is "0"

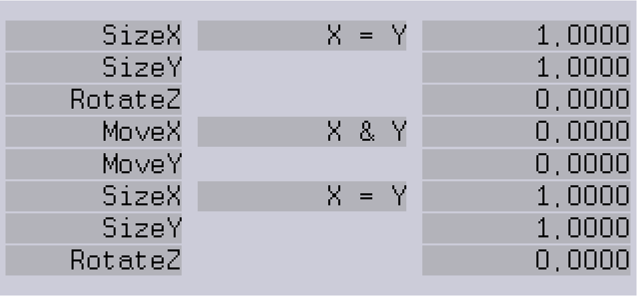
Buffer DST: (Destination Buffer)

Here you specify to which of the eight possible *buffers* (0-7) you want to copy the result of the operation.

This buffer must not be the same as the source buffer!!

Default is "1"

Transformation Parameters:



SizeX	X = Y	1,0000
SizeY		1,0000
RotateZ		0,0000
MoveX	X & Y	0,0000
MoveY		0,0000
SizeX	X = Y	1,0000
SizeY		1,0000
RotateZ		0,0000

Here you define the size and postion of the bitmap on the displayed picture. As in the InitPosition, you can change the order of the parameters.

Brightness:

Defines the brightness of the displayed bitmap(s).

Red:
Green:
Blue:

Defines the assignment of the laser colors to the bitmap colors.

By clicking on the color names you can change the default colors BMP Red, BMP Green and BMP Blue to any color from the primary color table.

By clicking on the color fields you can define the brightness for each color.

Displayed command line in the film:
`ApplyBitmap`
Position in the program:
Inside the time loop
Keyboard shortcut: `ab`

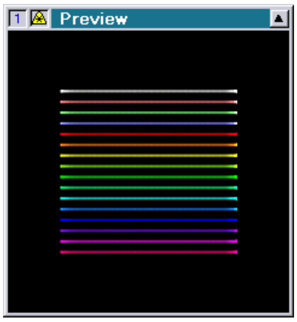
3.9.12 ApplyIntensity

The trickfilm command `ApplyIntensity` works similarly as the command `ApplyBitmap`.

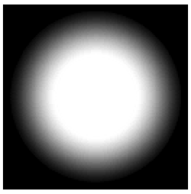
In contrast to the `ApplyBitmap`, where the bitmap is used to colorize the picture, the bitmap in the `ApplyIntensity` only changes the brightness of the picture.

Since the colors are not changed, the command `ApplyIntensity` can be combined with other coloring commands such as `ColorCYcle`, `ApplyBitmap` or `ApplyColorCycle`.

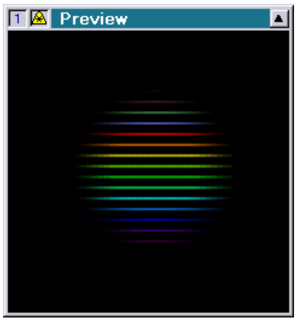
Example:



without ApplyIntensity



Bitmap



with ApplyIntensity

Bitmaps:

Here the bitmap(s) to be used are entered.

Either a single bitmap or several bitmaps can be entered by using wildcards.

Example: *.BMP enters all BMP files in the current directory.

Since only the brightness of the bitmaps is used, the bitmaps should contain only grayscales. If colored bitmaps are entered, they are automatically converted to grayscales.

Since the bitmaps are stored in the trickfilm, the bitmaps should not be unnecessarily large to prevent the loading time from being unnecessarily increased.

The maximum supported resolution is 256x256 pixels.

Frames:

Displays the number of bitmaps that have been entered.

Width:

Displays the width of the bitmap(s) in pixels.

Height:

Displays the height of the bitmap(s) in pixels.

Mode:

The selection `Wrap` / `Clip` defines how to deal with the case that the bitmap does not cover the entire laser picture:

`Wrap`	The bitmap is repeated endlessly on all sides
`Clip`	The bitmap is not repeated. Outside the bitmap the brightness from the edge of the bitmap is continued.

The selection `Relative` / `Absolute` determines how the value `Select BMP` is interpreted to select the desired bitmap:

`Relative` The value must be in range from 0.0 to 1.0

0.0 = first bitmap
...
1.0 = last bitmap

`Absolute` The value is equal to the number of the bitmap

0 = first bitmap
1 = second bitmap
2 = third bitmap
etc.

Select BMP:

The meaning of the value entered here depends on the selected mode `Relative` / `Absolute` (see above).

The default (`LoopCnt` in relative mode) results in an evenly progression of the bitmaps within the loop.

Buffer SRC: (Source Buffer)

Here the number of the picture buffer (0-7) is specified in which the picture is located whose brightness is to be changed by the command ApplyIntensity.

Default is "0"

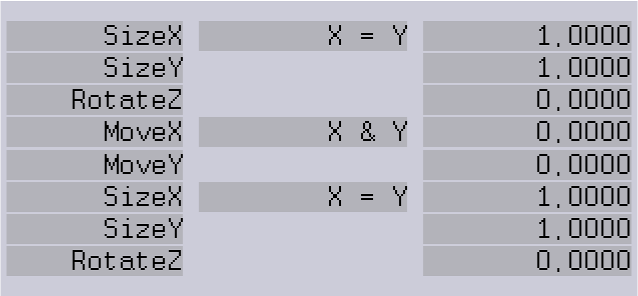
Buffer DST: (Destination Buffer)

Here the number of the picture buffer (0-7) is specified, into which the modified picture is to be written.

This buffer must not be the same as the source buffer!!

Default is "1"

Transformation Parameters:



SizeX	X = Y	1,0000
SizeY		1,0000
RotateZ		0,0000
MoveX	X & Y	0,0000
MoveY		0,0000
SizeX	X = Y	1,0000
SizeY		1,0000
RotateZ		0,0000

The transformation parameters are used to position the bitmap. As with ApplyBitmap or InitPosition, both type and order of the transformation parameters can be freely selected.

Annotations:

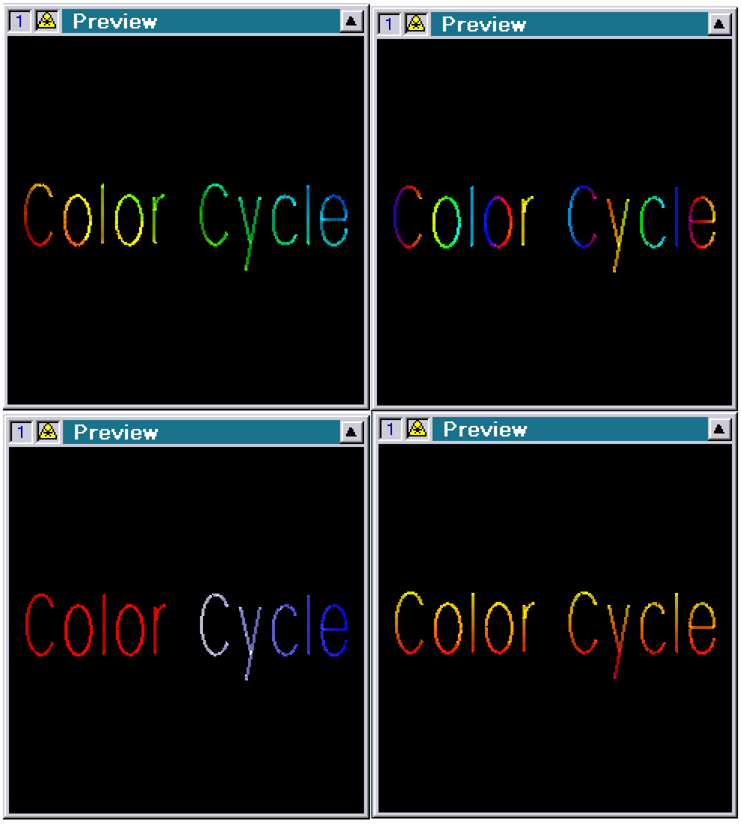
- The command ApplyIntensity affects only the points in the original picture, but not automatic added ones.
  To achieve smooth transitions, it is therefore recommended to interpolate the pictures in the Piced or with the trickfilm command HardInterpolation.
- As with the command ApplyBitmap, it is also important for ApplyIntensity that the color delay is set correctly.

Displayed command line in the film:
ApplyIntensity
Position in the program:
Inside the time loop
Keyboard shortcut: ai

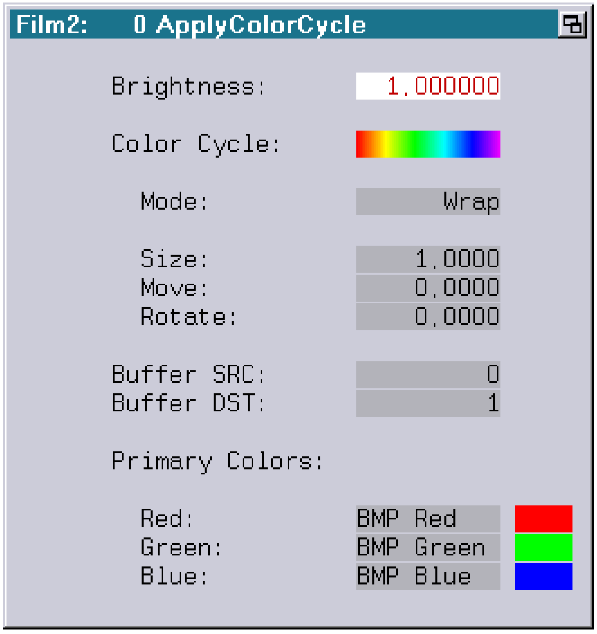
3.9.13 ApplyColorCycle

The command `ApplyColorCycle` can be used to apply the color information of a color cycle to a laser picture before it is displayed.

Here are a few examples:



The command works similar to the command `ApplyBitmap`: Depending on the X/Y-position of a point the desired color is selected. For this the `ApplyBitmap` command uses the color information of a bitmap, the color information for the `ApplyColorCycle` comes from a color cycle created directly on the Lasergraph DSP using the Color Cycle Editor (see manual chapter "Color Cycle Editor").



Film2: 0 ApplyColorCycle

```
Brightness:        1,000000
Color Cycle:
  Mode:            Wrap
  Size:            1,0000
  Move:            0,0000
  Rotate:          0,0000
Buffer SRC:        0
Buffer DST:        1
Primary Colors:
  Red:             BMP Red    🟥
  Green:           BMP Green  🟩
  Blue:            BMP Blue   🟦
```

Brightness: Determines the brightness of the newly calculated picture

ColorCycle: Enter or edit the color cycle used by the command

Mode: There are two different modes for determining which color is to be used when the calculation results in a position before the start resp. after the end of the color cycle:

- Wrap: The color cycle is repeated continuously
- Clip: Before the start of the color cycle the first color of the color cycle is used, after the end of the color cycle the last color of the color cycle is used.

Size:
Move:
Rotate: Determines the size and position of the color cycle within the picture which is to be colored

Buffer SRC: Number of the buffer (0-7) containing the picture which is to be colored

Buffer DST: Number of the buffer (0-7) for storing the colored picture

Red:	Primary color which is used for the red portion in the color cycle
Green:	Primary color which is used for the green portion in the color cycle
Blue:	Primary color which is used for the blue portion in the color cycle

Important! !!!

The `ApplyColorCycle` command only colors points which are actually part of the picture, not automatically interpolated points. In order to achieve smooth color transitions the picture should be interpolated either in the Piced or using the trickfilm command `HardInterpolation`.

Example of an entry using the `ApplyColorCycle` command:

```
Entry
CopyPictoBuffer      <picture to be colored> -> Buffer 0
HardInterpolation    Buffer 0 -> Buffer 1
TimeLoop
  ApplyColorCycle    Buffer 1 -> Buffer 2
  InitPosition
  DisplayBuffer      Buffer 2
LoopEnd
```

Remark:

The parameters Brightness, Size, Move and Rotate can contain variables and curves in addition to fixed values. Since the coloring is executed in real-time these parameters can also be influenced live by Faders or DMX values.

Displayed command line in the film:
`ApplyColorCycle`
Position in the program:
Inside the time loop
Keyboard shortcut: `acc`

3.10 Buffer Operations

3.10.1 CopyPictoBuffer

This command is needed when you want to perform an operation on a graphic which exceeds a simple projection, such as transforming, having it drawn or morphing it.
For these special effects you have to copy the corresponding graphics to one of the eight available *picture buffers*.

Catalog: Catalog

Here you specify the catalog from which you want to select a graphic.

Simply click on Catalog or on the empty field with the ? to the right of it. A window will then open headed Catalogs in memory listed, pick name.
All catalogs you already loaded into the picture memory are now listed.

If the necessary catalog has not been loaded into the picture memory click on the field with the middle mouse button. You are then on the File level and you can search the hard disk drive for the desired catalog.
Another click with the middle mouse button opens the Generals directory.

Clicking on the catalog opens the window Pictures in catalog xxx: (name of the selected catalog) and all graphics of this catalogs are listed. Click on the desired graphic. Then you will be back in the entry window for CopyPictoBuffer.

Picture: Picture

If you find that you want another graphic from the previously selected catalog click on the field with the displayed graphic or on Picture:.

Now you will again see all graphics of the catalog listed and you can choose another one by clicking on it.

`Buffer:`

Buffer

Here you specify into which of the eight *buffers* (0-7) the graphic is to be copied.

If you work with more than one *buffer* at a time all used *buffers* have to have different numbers!

Displayed command line in the film:
`CopyPictoBuffer` and number of the *buffer*
Position in the program:
Outside the time loop
Keyboard shortcut: `cpb`

3.10.2 SelectAnimationPic

With this command you copy a picture from an animation into one of the eight available *picture buffers* (0-7).

Therefore, to do anything with this command an animation either has to exist or you have to create one with the `Animation Editor`.

→ `AnimationEditor`

While the animation is running you can at the same time execute a metamorphosis, an interpolation or an equalization of the number of points between the individual `frames`.

To display the picture resp. the animation afterwards the `SelectAnimationPic` command always has to be followed by a `DisplayBuffer`.

→ Chapter 3.9.2 `DisplayBuffer`

Animation:

`Animation`

Here you enter the name of the desired animation. When you click on the field a new window listing all loaded animations will open. Click on the desired animation and you will immediately see the *entry window* again with the corresponding entry.

If the desired animation is not yet in the memory click on the field with the middle mouse button. You are then on the File level and you can search for the desired animation there.
By clicking on it it is automatically loaded and entered under `Animation`.

Stream from file:

This allows you to play an animation without loading it to the memory.
This is usefull if the animation consists of a large amount of frames (e.g. 10.000).
See `DisplayAnimation` 3.9.3 for details.

☜

Frames:

`Frames`

Here the number of individual pictures (`frames`) that make up the animation appears automatically.

Duration:

If an Animation was created in 3Dstudio Max and converted with proTize max, you will get the information about the original length of that animation.

Mode:

Mode

Here you can choose between Relative and Absolute. The default setting is Relative. This, in connection with LoopCnt for the following parameter, is probably the most commonly used setting because it leads to a smooth progression of the animation in the time defined in the TimeLoop.

In the Absolute Mode you enter constant values for the following parameter whose value is equivalent to the number of the frame of the animation. So 0 stands for the first picture, 1 for the second, 2 for the third picture of the animation and so on.

Number of picture:

Number of picture

The entry here depends on the selected Mode.
The default setting is LoopCnt in connection with the Relative Mode.

Otherwise there are the same possibilities as described in the chapter DisplayAnimation.

→ Chapter 3.9.3 DisplayAnimation

All other settings will only display a single picture or at most lead to a metamorphosis between the first and second pictures of the animation (Absolute Mode and LoopCnt).

Metamorphosis:

Metamorphosis

Here you specify whether or not you want to execute a metamorphosis between the individual pictures of the animation.
If yes enter ON.
If not leave the default setting OFF.

Equalize:

Equalize

If the individual pictures of your animation contain differing numbers of points and you want to execute a metamorphosis you have to adapt the numbers of points to reach an optimal result.
For this enter ON.
This command is equivalent to the command EqualizeBuffers with which you can equalize the numbers of points of two *buffers*.

→ Chapter 3.10.5 EqualizeBuffers

If all pictures of the animation have the same number of points this is not necessary and you enter OFF.

Interpolation:

Interpolation

Interpolation means that you can insert points into the pictures of your animation if the distance between two points exceeds a specific value (MaxDist).
Here you can also choose between yes, I want to interpolate: ON, and no, I don't need it: OFF.

This command and the following one are equivalent to the command HardInterpolation which is for instance used to make transformations cleaner and smoother.

→ Chapter 3.10.4 Hard Interpolation

MaxDist:

MaxDist

Here you define the maximum distance which two consecutive points are to have.
If it is exceeded points will automatically be inserted until the distance between the points is smaller than or equal to the specified maximum distance.
The smaller the value you enter is the more points will be inserted.

→ Chapter 3.10.4 Hard Interpolation

Destination buffer:

Destination buffer

Here you specify into which *buffer* you want to copy the picture (0-7).
This number you have to enter again for DisplayBuffer.

Scratch buffer 1:
Scratch buffer 2:
Scratch buffer 3:

These *buffers* are needed for temporary saving. You cannot display the contents of these *buffers.*

All four *buffers* have to have different numbers. !!!

Displayed command line in the film:
SelectAnimationPic
Position in the program:
Within the time loop, before DisplayBuffer
Keyboard shortcut: sap

3.10.3 CopyBuffer

With this command you copy a graphic from a *source buffer* (`Buffer SRC = Source`) which you specified before in the `CopyPictoBuffer` or `SelectAnimationPic` command to a *destination buffer* (`Buffer DST = Destinaiton`).

You then call up this *destination buffer* for display with `DisplayBuffer`.

You can for instance use the `CopyBuffer` command to have a graphic or a written line drawn forwards or backwards.

`Begin (fraction):`

`Begin (fraction)`

Here you specify from which point on the graphic is to be copied. Values between 0.0 and 1 are possible.
The default value is 0.0, i.e. the graphic is copied from the start.
If you enter a value of 0.5 here the first half of the graphic will not be copied.
Therefore a fixed value of 1.0 would not make a lot of sense here because the graphic would not be visible at all.

The effect of the graphic drawing itself or "undrawing" itself is achieved by using *variables*, e.g. `LoopCnt`, `ATTACK0()`, `DECAY0()` or a *curve*.

→ *Curve Editor*

Example:
If you enter `LoopCnt` here and 1.0 for the following parameter the graphic will "undraw" itself from the first to the last point.
If you want the "undrawing" to start with the last point, enter a 0.0 here and `DECAY0()` for the following parameter.

`End (fraction):`

`End (fraction)`

Here you specify up to which point the graphic is to be copied. Values between 0.0 are 1.0 are possible, but 0.0 makes as little sense here as 1.0 for the previous parameter.
The default setting 1.0 copies the graphic up to the last point, with 0.5 the second half of the graphic would not be copied.

Entering 0.0 for the previous parameter and 0.8 for this one would have the result that the last 20% of the graphic would not be displayed.

You can achieve a drawing effect with *variables*, e.g. `LoopCnt`, `ATTACK0()`, `DECAY0()` or *curves*.

→ *Curve Editor*

Example:
If you enter LoopCnt here and 0.0 for Begin the graphic will draw itself smoothly from the first to the last point.

Buffer SRC:

Buffer SRC

Here you specify from which of the eight possible *buffers* (0-7) you want to copy a graphic.

→ Chapter 3.10.1 CopyPictoBuffer

Buffer DST:

Buffer DST

Here you determine into which of the remaining seven *buffers* you want to copy the graphic. You enter this *buffer* number again for DisplayBuffer.

→ Chapter 3.9.2 DisplayBuffer

The *buffers* have to have different numbers!

An example for the necessary command lines:

	Entry
Catalog: Pictures Picture: Picture1 **Buffer: 0**	CopyPictoBuffer 0
	TimeLoop 5.000 InitPosition 0
Begin (fraction): 0.00 End (fraction): LoopCnt **Buffer SRC: 0** **Buffer DST: 1**	CopyBuffer 1
Buffer: 1 Transform-par.number: None Matrix 0 set by Pos#: 0 Matrix 1 set by Pos#: None and so on	DisplayBuffer 1
	LoopEnd

Displayed command line in the film:
CopyBuffer and number of the *destination buffer*
Position in the program:
within the time loop, before DisplayBuffer
Keyboard shortcut: cb

3.10.4 HardInterpolation

This command gives you the possibility to manually 'interpolate' a graphic at runtime (in comparison to an interpolation done within the Picture Editor). That means you can specify the maximum distance between two consecutive points (`Max distance`). If this distance is exceeded the computer automatically inserts one point here, or more if necessary.

For this you copy a graphic to a buffer(*source buffer* , `Buffer SRC = Source`) and define the *destination buffer* (`Buffer DST = Destination`).

The `HardInterpolation` is for instance used for transformations and morphings to make the movement look smoother and more precise.

→ Chapter 3.11.2/3 `Change/Do Transform`

→ Chapter 3.10.6 `AverageOfBuffers`

The HardInterpolation is usually no longer needed for the drawing effect (→ `CopyBuffer`) because the program executes an automatic interpolation.

`Max distance:`

`Max distance`

Here you determine the maximum distance between the individual points of your graphic.

If this defined distance between consecutive points is exceeded the computer inserts points until the distance is equal to or smaller than the maximum distance set by you.

Values between 0.01 and 1.0 are possible. You can also enter values smaller than 0.01, however in most cases this does not make sense. The smaller the value you enter the more points are inserted and depending on the complexity of the graphic the picture may flicker or the maximum number of points is exceeded and only parts of the graphic are visible.

Here is a simple example:

Let us assume you digitized a simple fan spanning the whole drawing area and you only entered two corner points, one at -1, the other one at +1.
The distance between these points then is 2.

```
o--------------------------------------------------o
-1                                                +1
```

Now if you interpolate this fan and enter 1 for Max distance the computer will insert one point because the maximum distance between two consecutive points will then be exactly 1.

```
o------------------------o------------------------o
-1                       0                        +1
```

If you enter 0.5 for Max distance the result after the interpolation will look as follows:

```
o-----------o-----------o-----------o-----------o
-1          -0.5        0           +0.5        +1
```

With a desired Max distance of 0.01 the computer would insert 199 points, the fan would then consist of 201 points. With a graphic more complex than a straight line a scanner would then soon reach its limits.

Buffer SRC :

Buffer SRC

Here you enter the number of the *buffer* in to which you copied the graphic.

→ Chapter 3.10.1 CopyPictoBuffer

Buffer DST:

Buffer DST

Here you enter the number of the *buffer* into which you want to copy the interpolated graphic (0-7). You call up this *buffer* number again with DisplayBuffer.

→ Chapter 3.9.2 DisplayBuffer

The buffers have to have different numbers!

Displayed command line in the film:
HardInterpolation and number of the *destination buffer* (Buffer DST)
Position in the program:
before the time loop
Keyboard shortcut: hi

3.10.5 EqualizeBuffers

This command is used to equalize the number of points of two graphics.

The `EqualizeBuffers` command mainly serves to prepare for a flawless morphing.

→ Chapter 3.10.6 `AverageOfBuffers`

→ Chapter 3.10.7 `TangramMorphing`

You can only morph two graphics faultlessly when they have an identical number of points.

`EqualizeBuffer` identifies the numbers of points of the graphics in `Buffer 1` and `Buffer 2` and evens out the difference by inserting points.

`Buffer 1:`

`Buffer 1`

Here you enter the number of the *buffer* into which you copied the first graphic.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Buffer 2:`

`Buffer 2`

Here you enter the number of the *buffer* into which you copied the second graphic.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Scratch buffer 1:`
`Scratch buffer 2:`

`Scratch buffer 1`
`Scratch buffer 2`

These buffers are needed for temporary saving. You have no influence on and no access to the contents of these *buffers*.

All four buffers have to have different numbers!

Displayed command line in the film:
`EqualizeBuffers` and number of `Buffer1`
Position in the program:
before the time loop, after `CopyPictoBuffer`
Keyboard shortcut: `eb`

3.10.6 AverageOfBuffers

The `AverageOfBuffers` is one of the two commands allowing a smooth metamorphosis between two graphics. However, the two morphings differ in the way the transformation is executed.

→ Chapter 3.10.7 `TangramMorphing`

With the `AverageOfBuffers` command all points from picture 1 move to the position of the corresponding points of picture 2 simultaneously.

A clean and faultless metamorphosis requires both graphics to have an identical number of points. Since this is rarely the case you set the `EqualizeBuffers` command before the metamorphosis.

→ Chapter 3.10.5 `EqualizeBuffers`

`Portion of buffer2:`

`Portion of buffer2:`

This parameter determines how far the picture points from the first picture have already moved from their original position in the direction of the second picture.

With an entry of 0.0 they would still be in their original position, i.e. picture 1 would be displayed.

With an entry of 1.0 they are in the position of the second picture, i.e. picture 2 would be displayed.

A continuous morphing is achieved by entering `LoopCnt`, `Attack` or a corresponding *curve* (0-1).
When you enter `Decay` you exchange the start picture and the resulting picture, i.e. you morph from picture 2 to picture 1 equivalent to a *curve* from 1-0.

→ *Curve Editor*

`Buffer1:`

`Buffer1:`

Enter the number of the buffer into which you previously copied the first picture for the planned metamorphosis.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Buffer2:`

`Buffer2:`

Here you enter the number of the buffer into which you previously copied the second picture for the planned metamorphosis.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Destination:`

`Destination:`

Here you enter the number of the *buffer* into which you want to copy the result of the metamorphosis (0-7). This *buffer* number you call up again with `DisplayBuffer`.

→ Chapter 3.9.2 `DisplayBuffer`

All buffers have to have different numbers!

Displayed command line in the film:
`AverageOfBuffers` and number of the destination buffer
Position in the program:
within the time loop, before `DisplayBuffer`
Keyboard shortcut: `aob`

3.10.7 TangramMorphing

You can use the TangramMorphing command like the AverageOfBuffers command to program a metamorphosis between two graphics.

→ Chapter 3.10.6 AverageOfBuffers

However, the progress of the metamorphosis is different from the AverageOfBuffers command.

With the TangramMorphing the points of the first picture are not shifted to the corresponding position of the second picture simultaneously as with the AverageOfBuffers command. Instead they are shifted consecutively.

In addition you can determine the number of points that are to morph simultaneously resp. the size of the sections that are to morph in succession.

A clean and faultless metamorphosis requires both graphics to have an identical number of points. If this is not the case set the EqualizeBuffers command before the metamorphosis.

→ Chapter 3.10.5 EqualizeBuffers

Portion of buffer2:

Portion of buffer2

This parameter determines how far the picture points from the first picture have already moved from their original position in the direction of the second picture.

→ Chapter 3.10.6 AverageOfBuffers

The default setting LoopCnt results in a continuous metamorphosis of the picture in Buffer1 to the picture in Buffer2.

Width of transition:

Width of transition

You can use the value Width of transition to influence the characteristic of the morphing.
The value you enter here determines the size of the sections of the picture which successively shift to the position of the second picture. The smaller the value is the smaller are the individual sections of the picture.
It is hard to describe the different effects you can achieve with this parameter. Just give it a try!

Buffer1:

Buffer1

Here you enter the number of the buffer into which you previously copied the first picture for the planned metamorphosis.

→ Chapter 3.10.1 CopyPictoBuffer

Buffer2:

Buffer2

Here you enter the number of the buffer into which you previously copied the second picture for the planned metamorphosis.

→ Chapter 3.10.1 CopyPictoBuffer

Destination:

Destination

Here you enter the number of the buffer into which you want to copy the result of the metamorphosis (0-7). This *buffer* number you call up again with DisplayBuffer.

→ Chapter 3.9.2 DisplayBuffer

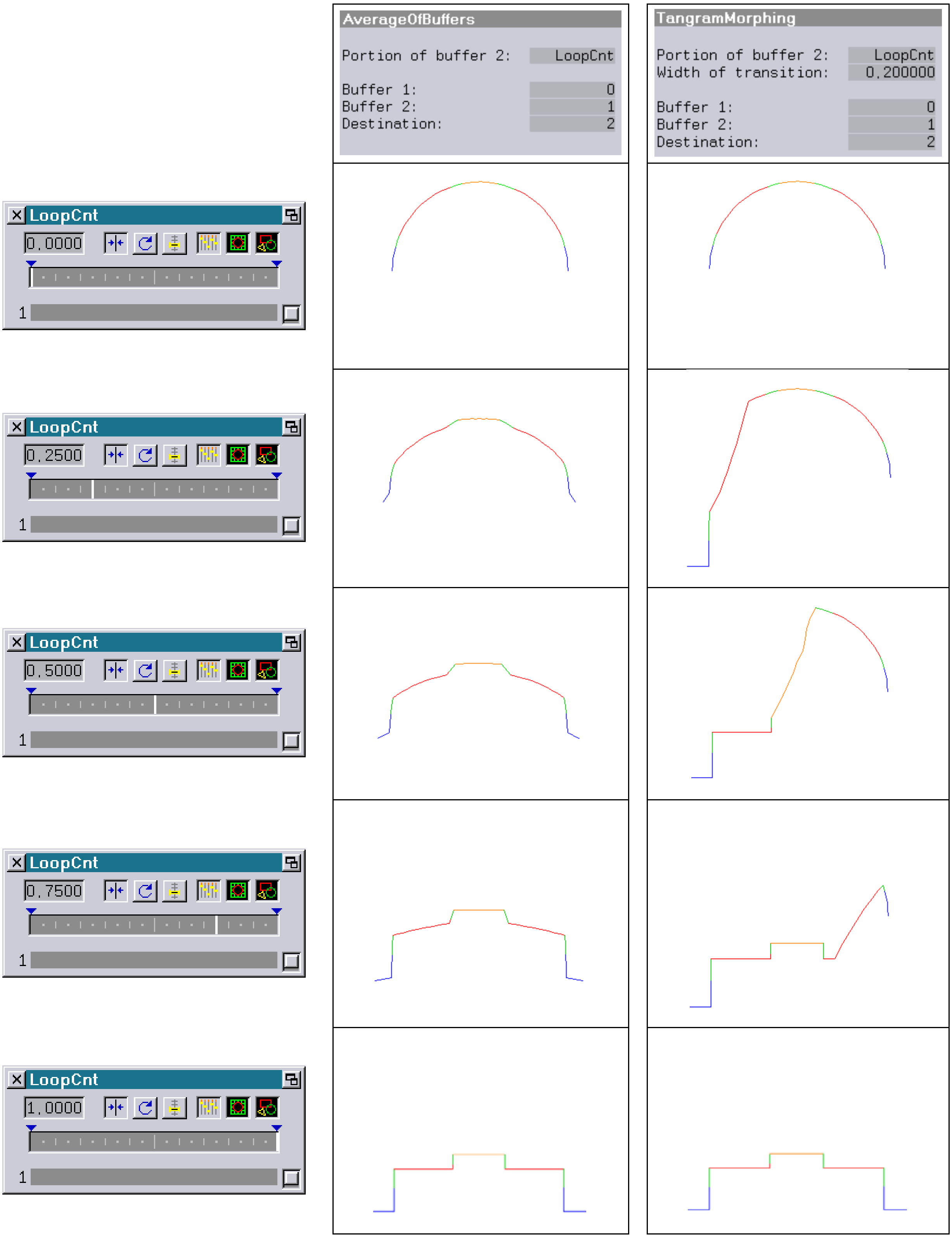
All buffers have to have different numbers!

Displayed command line in the film:
TangramMorphing and number of the destination buffer
Position in the program:
within the time loop, before DisplayBuffer
Keyboard shortcut: tm

Here is another example of the necessary command lines when working with buffers:

	Entry
Catalog: Pictures Picture: SemiCircle **Buffer:** 0	CopyPictoBuffer 0
Catalog: Pictures Picture: Stairs **Buffer:** 1	CopyPictoBuffer 1
Buffer1: 0 **Buffer2:** 1 ScratchBuffer1: 6 ScratchBuffer2: 7	EqualizeBuffers 0
	TimeLoop 5.000 InitPosition 0
Portion of buffer2: LoopCnt Width of transition: 0.200000 **Buffer1: 0** **Buffer2: 1** **Destination: 2**	TangramMorphing/ 2 AverageOfBuffers **2**
Buffer: 2 Transform-par.number: None Matrix 0 set by Pos#: 0 Matrix 1 set by Pos#: None and so on	DisplayBuffer 2
	LoopEnd

Below you see an example for the different types of transformations with `AverageOfBuffers` and `TangramMorphing`:



AverageOfBuffers

Portion of buffer 2: LoopCnt

Buffer 1: 0
Buffer 2: 1
Destination: 2

TangramMorphing

Portion of buffer 2: LoopCnt
Width of transition: 0,200000

Buffer 1: 0
Buffer 2: 1
Destination: 2

LoopCnt 0,0000 1

LoopCnt 0,2500 1

LoopCnt 0,5000 1

LoopCnt 0,7500 1

LoopCnt 1,0000 1

3.11 Transformations

3.11.1 InitTransform

With this command you define transformation masks which you can use to give any graphic another shape, for instance that of a sphere.

Just imagine you have a piece of paper with a suitable picture (your graphic) and you wrap it around a ball or a cylinder (the mask).

The graphic has to be copied into one of the picture buffers first and then displayed with `DisplayBuffer`.

→ Chapter 3.10.1 `CopyPictoBuffer`

`Transform-par.number:`

Here you enter the identification number for this transformation parameter.

You can define up to four (0-3) transformation lists per time loop. You then assign these numbers to the corresponding graphics by entering the `Transform-par.number` at the picture display (`DisplayBuffer`).

→ Chapter 3.9.2 `DisplayBuffer`

`Transformation-type:`

`Transformation -type`

Here you define which of the four possible transformations you want to execute.

When you click on the field you will see the following selection possibilities in the *menu bar* at the bottom:

```
Sphere
Cylinder
NETInt
NET
```

Simply click on the desired transformation or press the corresponding function key.

For each of the four transformations you will see different parameters in the *entry window*.

Below we will present the different settings you can make resp. have to make for the four different transformations and the effects they have.

Transformation-type: Sphere

Sphere

Sphere is the default setting. With this you define the surface of a sphere.

With the following parameters you define how far your graphic is to be applied to the surface of the sphere.

Sphere hor. angle:

Sphere hor. angle

With this parameter you determine by how many degrees your graphic is to form into a sphere in the horizontal plane.

Provided your graphic was digitized over the complete drawing area, the default values of 180.000 for this parameter and 90.000 for the following one will result in a complete sphere.
If you for instance enter 90.000 here and keep the other default settings you will form the graphic into a hemisphere.
(See fig. InitTrans 1-4)

Sphere vert. angle:

Sphere vert. angle

Here you enter the angle for the bending in the vertical plane.
(See fig. InitTrans 1-4)

Sphere RadiusX:

Sphere RadiusX

With this parameter you can influence the diameter of the sphere in the x-plane.
(See fig. InitTrans 6 and 7)

→ Cylinder RadiusX

Sphere RadiusY:

Sphere RadiusY

With this parameter you influence the vertical diameter of the sphere.
(See fig. InitTrans 5)

On the following page you see in four steps how a graphic is applied to the sphere mask. Fig. InitTrans1.1 shows the unchanged graphic as it was digitized, using the complete drawing area.
In fig.2 half of the transformation is completed, two thirds in fig.3, in fig.4 it completely covers the surface of the sphere. The figures 1-4.2 show the same process from another perspective.
The figures 5, 6 and 7 illustrate the changes of the x- resp. y-radius.

Sphere hor. angle Sphere vert. angle Sphere RadiusX Sphere RadiusY	90,000 45,000 Norm:1,00000 Norm:1,00000	135,000 67,500 Norm:1,00000 Norm:1,00000	180,000 90,000 Norm:1,00000 Norm:1,00000

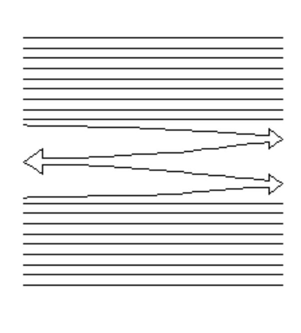


Fig. InitTrans1.1

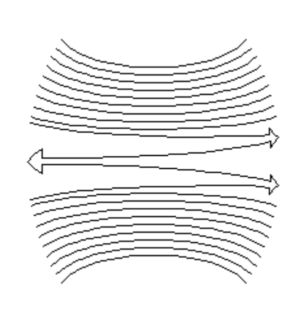


Fig. InitTrans2.1

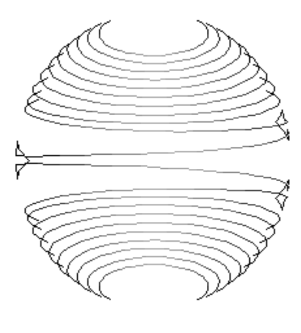


Fig. InitTrans3.1

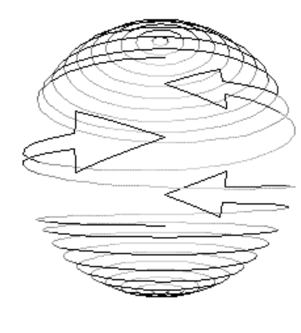


Fig. InitTrans4.1

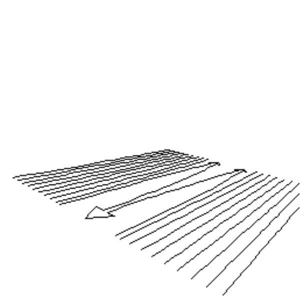


Fig. InitTrans1.2

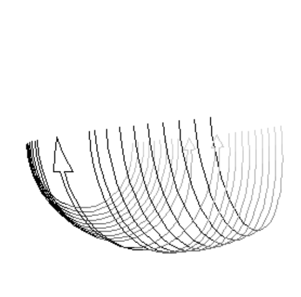


Fig. InitTrans2.2

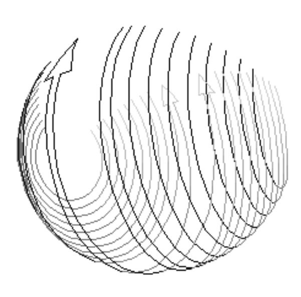


Fig. InitTrans3.2

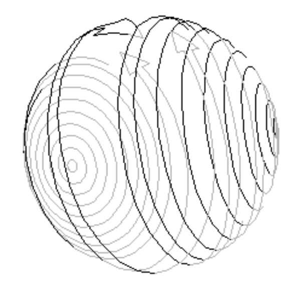


Fig. InitTrans4.2

Sphere hor. angle Sphere vert. angle Sphere RadiusX Sphere RadiusY	180,000 90,000 Norm:1,00000 Norm:0,50000	180,000 90,000 Norm:0,50000 Norm:1,00000	As Fig. InitTrans6, View after 40° y-rotation

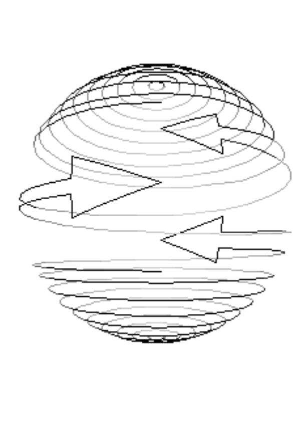


Fig. InitTrans4.1

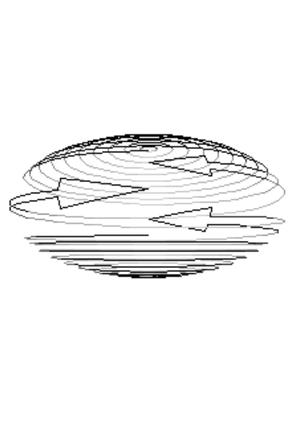


Fig. InitTrans5

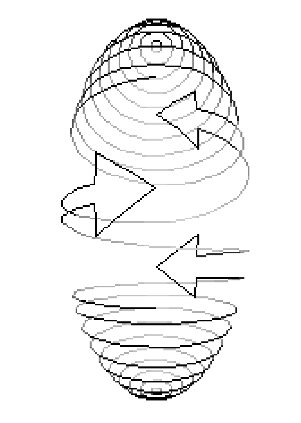


Fig. InitTrans6

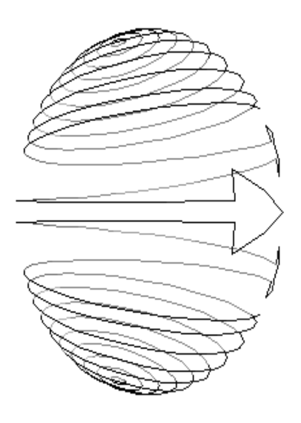


Fig. InitTrans7

As you can see in the graphic examples, in the z-plane a graphic wraps itself around a mask from back to front. The center of the picture is located on the inner rear side of the cylinder or sphere after the transformation is completed.

This has the result that the graphic is mirror-inverted on the side bent outwards.

For graphic structures like in the examples on the previous page this does not make much of a difference. However, if you for instance want to have a line of writing circle around the earth globe it would significantly impair the readability if all the characters appeared inverted (see fig. `InitTransCyl` 3).

Some programmers try to solve this problem by digitizing the writing inverted. This is one way of doing it, but only if you do not show the transformation but only the final result. It is not very nice if the writing appears on the screen inverted at the beginning of the transformation.

Tricks like this are actually not necessary. You can simply reverse the orientation of the transformation.

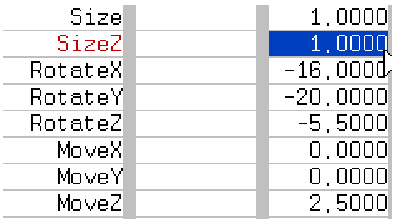
Just imagine you have a transparent foil with printing on it on your desk in front of you with the right side up. You then place a cylinder (or a ball) on the foil and wrap the foil around it.

When you now turn the cylinder you see the back side of the foil. That is what we have done so far and what is illustrated in the figures InitTransCyl 1-3.

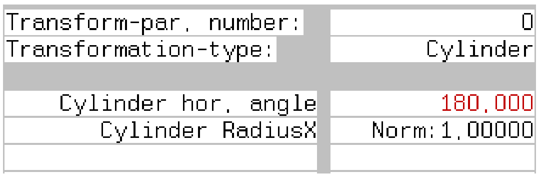
Now you place the cylinder on the table first and then put the foil on the cylinder, again with the right side up, and wrap it around the cylinder toward the back. This time you have the front on the outside and the writing the right way around.
(See fig. `InitTransCyl` 4-6.)

You achieve this by replacing the first parameter "`Size`" of InitPosition with `SizeZ` or by inserting a `SizeZ` and entering a negative value there.

→ Chapter3.8.3 `InitPosition`



Size	1.0000
SizeZ	1.0000
RotateX	-16.0000
RotateY	-20.0000
RotateZ	-5.5000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000



Transform-par. number:	0
Transformation-type:	Cylinder
Cylinder hor. angle	180.000
Cylinder RadiusX	Norm:1.00000

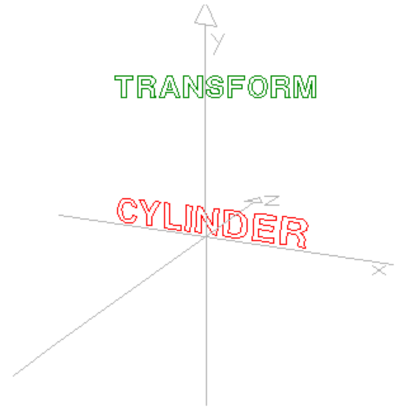


Fig. InitTransCyl 1

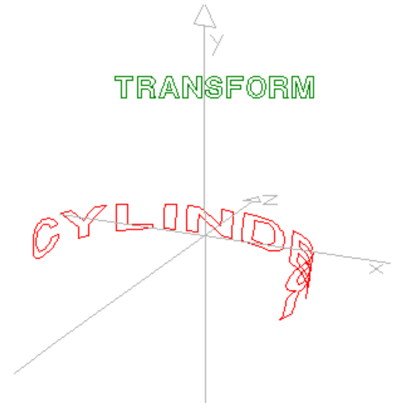


Fig. InitTransCyl 2

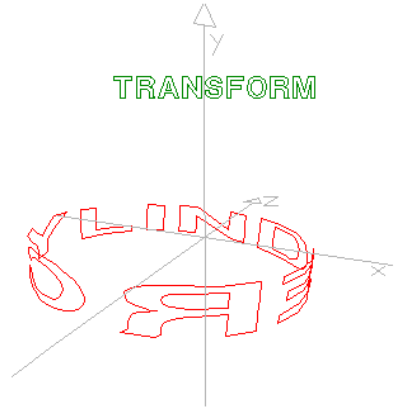
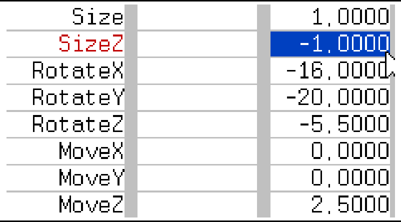
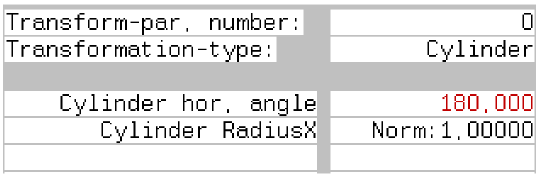


Fig. InitTransCyl 3



Size	1.0000
SizeZ	-1.0000
RotateX	-16.0000
RotateY	-20.0000
RotateZ	-5.5000
MoveX	0.0000
MoveY	0.0000
MoveZ	2.5000



Transform-par. number:	0
Transformation-type:	Cylinder
Cylinder hor. angle	180.000
Cylinder RadiusX	Norm:1.00000

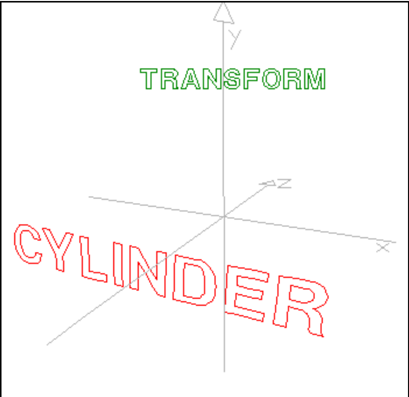


Fig. InitTransCyl 4

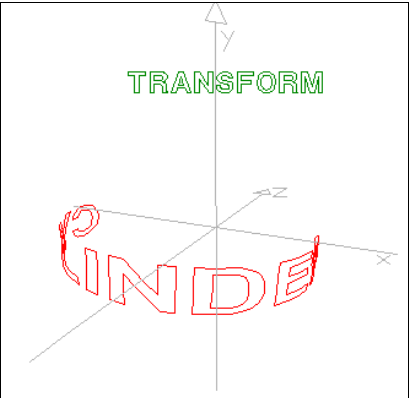


Fig. InitTransCyl 5

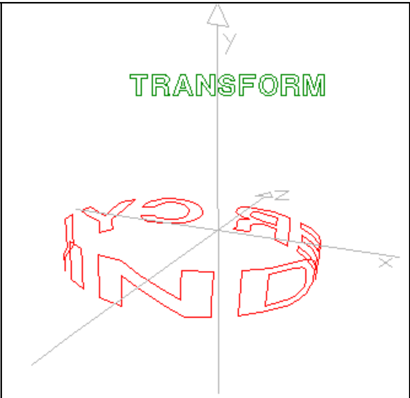


Fig. InitTransCyl 6

This reversal of the orientation of the transformation, shown here for `Cylinder`, of course also works for the function `Sphere`.

The transformations Sphere and Cylinder are ideal for the use of the Zclipping. Depending on the complexity of the transformed graphic the visible lines to the rear can be very confusing and can significantly impair the 3-dimensional impression.
You should especially use the Zclipping if you want to have a graphic or a line of writing rotate on a circular path around another object - unless the object is supposed to be made of glass (see example below).

→ Chapter 3.8.1 InitCamera: ZClipping

In the examples fig. 8 and 9 a SWC mask was programmed in addition. Fig. 7 shows the same projection as fig. 8 without Zclipping and SWC for comparison.

→ Chapter 3.5.2 setSWC

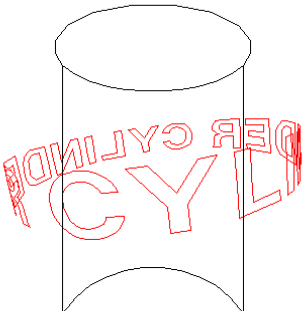
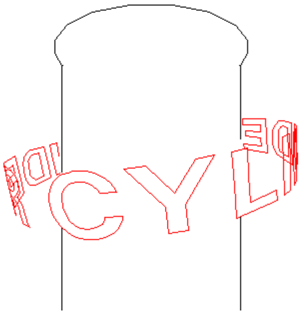
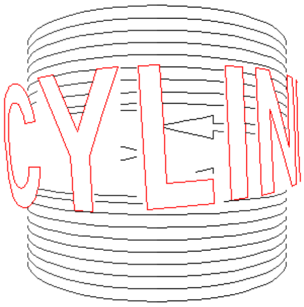
		

Fig. InitTransCyl 7	Fig. InitTransCyl 8	Fig. InitTransCyl 9

Transformation-type: Cylinder

Cylinder

With this entry you define a cylinder surface.

Cylinder hor. angle

Cylinder hor. angle

Here you specify by how many degrees the graphic is to be applied to the cylinder.

Provided your graphic was digitized over the complete drawing area, the default value of 180.000 will result in a complete cylinder (see fig. InitTransCyl 3, 6 and 7).
When you enter 90.000 only half of the cylinder will be covered.

Cylinder RadiusX

Cylinder RadiusX

With this parameter you determine the diameter of the cylinder on the x-plane.

The default setting Norm:1.0 guarantees a perfectly circular cylinder (fig. InitTransCyl 10).

If you reduce the RadiusX the cylinder will only be compressed in the x-plane, i.e. it will become elliptical as illustrated in figures 11 and 12.

However, if you want to have a circular cylinder with reduced diameter you have to add a SizeZ for InitPosition and reduce the default value of 1.0 by the same factor as RadiusX.

→ Chapter 3.8.3 InitPosition

Cylinder hor.angle: 180.0 Cylinder RadiusX:Norm:1.0	Cylinder hor.angle: 180.0 Cylinder RadiusX: Norm:0.5	View from above along the y-axis through the cylinder with RadiusX 0.5

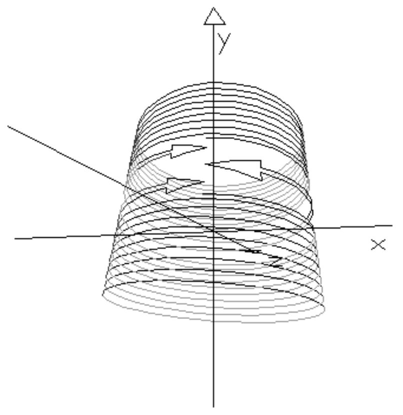
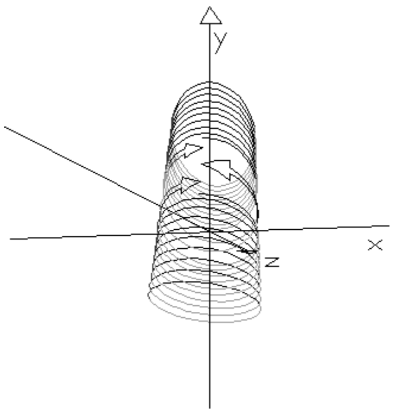
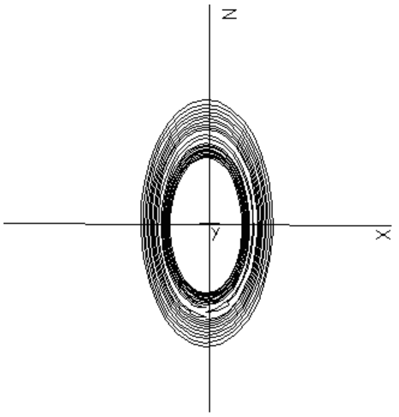

		

Fig. InitTransCyl 10	Fig. InitTransCyl 11	Fig. InitTransCyl 12

Transformation-type: NETInt

NETInt

With this entry you define a transformation mask in the form of a net picture.

Such a net has to consist of at least two lines. !!!
All lines have to have the same number of points.

When digitizing the net graphic, start with the bottom line from -x to +x, i.e. from left to right (see fig. NETInt 1).
With these two lines you then define a clipping where another graphic is deformed.

For this you have to copy the net graphic as well as the picture which is to be deformed into one of the eight *picture buffers* and then display the picture with DisplayBuffer.
Usually only the deformed picture will be displayed (see fig. NETInt 3).
However, you can of course also show the transformation mask as in fig. NETInt 2.

Transformation Mask	Transformation Mask + Deformed Graphic	Deformed Graphic

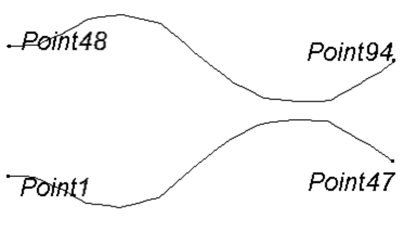
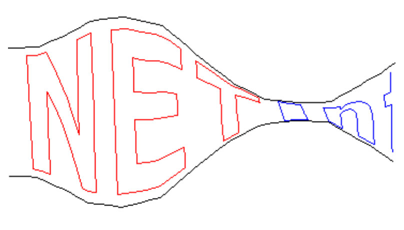
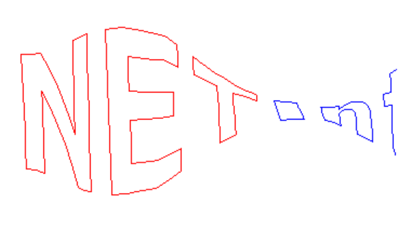
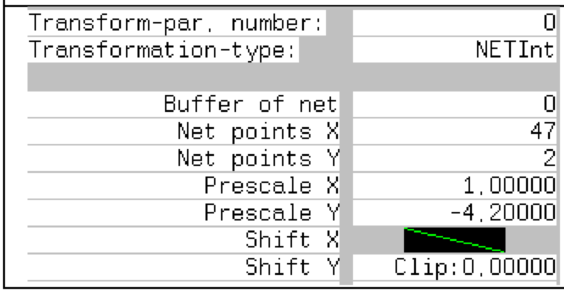
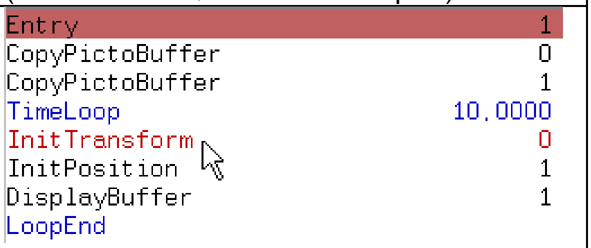

		

Fig. NETInt 1	Fig. NETInt 2	Fig. NETInt 3

The InitTransform Parameters for Fig. NETInt 1+2		The command line if you only want to display the deformed graphic. (Buffer 0 = Net, Buffer 1 = Graphic)
 Transform-par. number: 0 Transformation-type: NETInt Buffer of net 0 Net points X 47 Net points Y 2 Prescale X 1,00000 Prescale Y -4,20000 Shift X Shift Y Clip:0,00000		 Entry 1 CopyPictoBuffer 0 CopyPictoBuffer 1 TimeLoop 10,0000 InitTransform 0 InitPosition 1 DisplayBuffer 1 LoopEnd

Buffer of net

Here you enter the number of the transformation buffer (0-7).

This number you enter again for `Transform-par.number` for `DisplayBuffer`.

Net points X

Here you have to enter the number of points of **one** line along the x-axis.

This number multiplied by `Net points Y` has to be the total number of points in the net.

Example fig. `NETInt` 1-3:

1 line consists of 47 points.
The net consists of 2 lines.
2 x 47 = 94 = total number of points in the net.

Net points Y

Here you enter the number of lines.

The following parameters influence the size and position of the graphic which is to be deformed, not of the net, while you change the size and position of the net and the graphic simultaneously with `InitPosition`, even if you do not display the net graphic at all. **!!!**

Prescale X

Prescale X

With this parameter you can influence the size of the graphic in the x-plane.

Prescale Y

Prescale Y

With this parameter you can influence the size of the graphic in the y-plane.

Shift X

Shift X

Here you determine the horizontal position of the graphic.

The default setting `Clip: 0.00` for this and the following parameter positions your graphic exactly in the center of the net.
If you want to move the graphic you can either define the start position here and set the end position with `ChangeTransform`, or you enter a *variable* or a *curve* here right away.
However, in this case you will not find the *Curve Editor* in the *menu bar* at the bottom. You can still call it up by pressing <~> twice and confirming with <shift return>.
Instead of `Clip`: you can also select `Wrap`: from the *menu bar* at the bottom. This creates an effect as if the part of the graphic would uncoil where it clips and rebuild at the opposite clipping edge.

→ *Curve Editor*

Shift Y

Shift Y

Here you determine the vertical position of the graphic.
All settings and options are equivalent to `Shift X`, only vertically.

→ Shift X

Displayed command line in the film:
`InitTransform` and number of the transformation parameter (`Transform-par. number`)
Position in the program:
(usually) within the time loop, before the picture display
Keyboard shortcut: `it`

3.11.2 ChangeTransform

The `CangeTransform` command is equivalent in significance and application to the `ChangePosition` command.
The positions you entered for InitTransform will be changed by the values you enter here multiplied by the factor.

→ Chapter 3.8.4 `ChangePosition`

`ChangeTransform` can only be used in connection with a previously entered `InitTransform` and a picture buffer. `ChangeTransform` automatically adopts the `Transformation-type` selected for `InitTransform` (`Sphere`, `Cylinder`, `NETInt` or `NET`), provided the `Transform-par. number` is identical.

→ Chapter 3.11.1 `InitTransform`

→ Chapter 3.10. `Buffer Operations`

As with `ChangePosition` there are two basic applications for the `ChangeTransform` command:

1. You can use the `ChangeTransform` command to execute a transformation which you defined previously with `InitTransform`, e.g. apply a picture to the surface of a sphere or a cylinder or, with the net transformations, influence a graphic's size and x/y movement within a net.

   This form of application is actually outdated because you can already enter all this for `InitTransform` using a *Curve*.

   → *Curve Editor*

2. The second possible application is the so called incremental method. Here the `InitTransform` is located outside the loop and with each cycle of the loop the current position is changed by the position change defined in the `ChangeTransform` multiplied by the factor (usually 1).
   This creates a movement speed which is independent of the duration of the loop, which however is not constant but depends on several factors, such as the complexity of the graphic and the scanning speed and your DSP card.
   However, using *variables* and the internals time it is possible to achieve a movement speed which is actually constant.

   → Chapter 3.8.4 `ChangePosition`

`Transform-par. number:`

The `Transform-par. number` of `ChangeTransform` has to be identical to that of the corresponding `InitTransform` command. `ChangeTransform`, if it is entered subsequently, automatically adopts the number entered for `InitTransform`.
If for any reason you subsequently change the number in `InitTransform` you also have to correct it in `ChangeTransform`.

→ Chapter 3.11.1 `InitTransform`

Transformation-type:

ChangeTransform automatically adopts the Transformation-type entered for InitTransform. If you subsequently change it in InitTransform it is automatically updated when you click anywhere on the window.

Factor:

The values you enter for ChangeTransform are multiplied by the Factor and then added to the values from InitTransform.

The default setting is **LoopCnt**.
LoopCnt starts at 0 and ends at 1.

Let us assume you want to wrap a graphic around a sphere and you set the Sphere hor. angle and the Sphere vert. angle to 0 in InitTransform.

For ChangeTransform you now enter 180 for Sphere hor. angle and 90 for Sphere vert. angle.

At the beginning of the loop the 180 as well as the 90 would now be multiplied by 0 and then added to the 0 you entered for InitTransform. The result each time is 0. We therefore start with the position defined in InitTransform.

At the end of the loop the LoopCnt has reached 1. We therefore calculate:

180 x 1 + 0 = 180
90 x 1 + 0 = 90

The graphic has wrapped itself evenly around the surface of the sphere during the duration of the loop.

Other reasonable choices for this application of the ChangTransform command are ATTACK0(), DECAY0() or a *curve*.

→ *Curve Editor*

A constant value does not make sense here, you might as well define the position with InitTransform.
You only enter a fixed value if you use the incremental method (see above).

Sphere hor. angle

Sphere hor. angle

With this parameter you determine by how many degrees your graphic is to form into a sphere on the horizontal plane.

→ Chapter 3.11.1 InitTransform

Sphere vert. angle

Sphere vert. angle

Here you enter the angle change for the bending on the vertical plane. (see fig. InitTrans 1-4)

→ Chapter 3.11.1 InitTransform

Sphere RadiusX

Sphere RadiusX

With this parameter you can change the diameter of the sphere on the x-plane.
(see fig. InitTrans 6 and 7)

→ Chapter 3.11.1 InitTransform

Sphere RadiusY

Sphere RadiusY

With this parameter you change the vertical diameter of the sphere.(see fig. InitTrans 5)

→ Chapter 3.11.1 InitTransform

For the other Transformation-types Cylinder, NETInt and NET you also change the values entered for InitTransform in the described way.

→ Chapter 3.11.1 InitTransform

Displayed command line in the film:
CahngeTransform and number of the transformation parameter (Transform-par. number)
Position in the program:
within the time loop, after InitTransform, before the picture output
Keyboard shortcut: ct

3.11.3 DoTransform

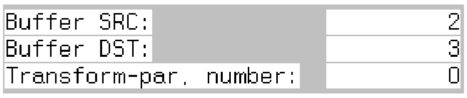
You only need this command if you want to transform a graphic twice within the same time loop.

DoTransform allows you to use the intermediate result after the first transformation as the basis for another transformation.

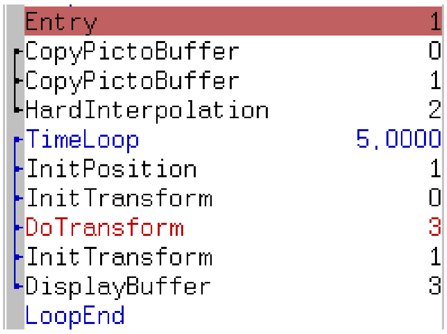
This way you can for instance transform a graphic with a NETInt transformation first and the apply it to the surface of a sphere.
You can also have a graphic turn around a cylinder and in addition use a NET transformation to create the illusion that the graphic is under water or fluttering in the wind.

→ Chapter 3.11.1 InitTransform

In the trickfilm this would look like the following example:



```
Buffer SRC:                 2
Buffer DST:                 3
Transform-par. number:      0
```



```
Entry                  1
CopyPictoBuffer        0
CopyPictoBuffer        1
HardInterpolation      2
TimeLoop          5,0000
InitPosition           1
InitTransform          0
DoTransform            3
InitTransform          1
DisplayBuffer          3
LoopEnd
```

First we copied the net graphic and the graphic which is to be transformed into different buffers. The following HardInterpolation is optional and depends on the composition of your graphic.

→ Chapter 3.10.4 Hard Interpolation

Within the time loop after InitPosition we find the first InitTransform command with the Transform-par. number 0. Afterwards we copied the deformed graphic with DoTransform from Buffer 2 into Buffer 3 in order to alter it again. The second InitTransform has the parameter number 1.

We then call up the final result with DisplayBuffer by entering the number of the destination buffer of DoTransform (3) and the Transform-par. number of the second transformation (1).

Buffer SRC:

Buffer SRC

Here you enter the buffer number of the graphic which has already been transformed once.

Buffer DST:

Buffer DST

Here you enter the number of the buffer into which you want to copy the graphic in order to process it further.
You call up this number again with `DisplayBuffer`.

Transform-par. number:

Transform-par. number

Here you enter the parameter number of the transformation which was already executed.

Displayed command line in the film:
`DoTransform` and number of the destination buffer (`Buffer DST`)
Position in the program:
within the time loop, after `InitTransform`, before the picture output
Keyboard shortcut: `dt`

3.12 Variables

3.12.1 SetVariables

With this command you can program your own *variables*.

It is possible to use up to 384 independent *variables* per trickfilm.

You can use a value range (*floating point*) from 10^{-100} to 10^{100} with an accuracy of up to six decimals. Everything outside this range is not defined.

You can choose the names for the *variables* freely, however, they have to start with a letter and are limited to eight characters. You can also use underscores and numerals, but only after the first letter.

You can use *variables* practically at every point where you achieve any change to the displayed graphic, e.g. moves, color changes etc.

The use of *variables* allows many possibilities to define movements which you could not realize otherwise.

An example:

```
IF  Fader(0) > 0.5 THEN
   MoveX = SIN (LoopCnt * 360)
ELSE
   MoveX = 0
END
```

→ Chapter 3.8.3 InitPosition

/*comment*/

/**/

To insert a comment into the code use the syntax /**/.

Example:

```
x = SIN (Seconds()*360)  /*used for the move x*/
```

In addition to the 384 *variables* which you can define yourself you will find the following predefined *variables* in the DSP:

Fader0-15

Fader0-15

The values are determined by the sliders of the fader board (0.0 - 1.0).

LoopCnt

LoopCnt

Counts from 0.0 to 1.0 in the time that is defined by the "TimeLoop", "ExternalLoop" or "Loop".

LoopStep

LoopStep

At the beginning of a loop "LoopStep" is "0.000".
After each loop cycle "LoopStep" is set to the quotient of the time the loop cycle has taken and the total loop time.

Example:
If a "TimeLoop" runs for 3 seconds and the display of the picture within the loop takes 0.01 second (100 Hz) `LoopStep` will be set to 0.00333 (= 0.01s / 3s).

LoopCnt versus LoopStep

LoopCnt versus LoopStep

Before it was possible to enter curves into the `InitPosition` command a movement was typically performed using the command `ChangePosition`:

```
TimeLoop
InitPosition
ChangePosition
DisplayPicture
LoopEnd
```

The parameter "Factor" within the command `ChangePosition` was set to "LoopCnt".

For every loop cycle the position was reset to the start position as specified in the "InitPosition" command.

Before the picture was displayed the position was modified by the `ChangePosition` command. The values here are multiplied with the "Factor", in this case the "LoopCnt" (0.000-1.000), and added to the position specified in the `InitPosition` command.

A different approach was to use "LoopStep" as "Factor".
In this case the Loop had to look like this:

```
InitPosition
TimeLoop
ChangePosition
DisplayPicture
LoopEnd
```

Here the start position is specified outside of the loop and the `ChangePosition` adds a small portion to the position each time it is executed. At the end of the loop the entire values specified in the `ChangePosition` have been added to the initial position.

Nowadays the command `ChangePosition` with the Factor "LoopCnt" is used only rarely because it is much more convenient to enter start and end value within the `InitPosition` command.

But the use of `ChangePosition` in combination with the Factor "LoopStep" still makes sense:
If you do not want to use variables it is the only possibility to program a loop for performing a specific transformation (for example a 360° rotation in 3 seconds) beginning from an unknown start position.

MouseX

MouseX

If the middle mouse button is not pressed the value of this *variable* is 0.5. By pressing the middle mouse button and simultaneously moving the mouse left/right the value changes between 0.0 and 1.0.

MouseY

MouseY

If the middle mouse button is not pressed the value of this *variable* is 0.5. By pressing the middle mouse button and simultaneously moving the mouse up/down the value changes between 0.0 and 1.0.

To create a *variable* of your own insert `SetVariables`.
On the left you will then see an empty field with horizontal lines.

When you click on the empty field the entry line at the bottom will show `Set Assign0` and the cursor. This means that the computer is waiting for your orders.

The *menu bar* at the bottom provides a range of tools.

Under [`operators`], [`functions`], [`IF-THEN`] and [`delim`] you will find a list of most of the supported operators, functions etc.

[operators] [operators]

a^b example: 2^3 = 8 (equivalent to $2^3 = 8$) **^**

a*b multiplication mark *****

a/b division mark **/**

a+b addition mark **+**

a-b subtraction mark **-**

[relations] [relations]

The following operators deliver only one result - either 1.0 or 0.0.

a=b equals sign:
If a equals b is true the result is 1.0, if it is not true 0.0. **=**

a==b same as a=b **==**

a!=b Here the result is 1.0 if a is not equal b, and when a = b is true it is 0.0. **!=**

a<b Smaller than: When a is smaller than b is true the result is 1.0, if not 0.0. **<**

a<=b Smaller or equal: If a is smaller than or equal to b the result is 1.0, if not 0.0. **<=**

a>b Larger than: When a is larger than b is true the result is 1.0, if not 0.0. **>**

a>=b Larger or equal: If a is larger than or equal to b the result is 1.0, if not 0.0. **>=**

[logical] [logical]

Under [logical] you will find the two logical 'And' or 'Or' operators. These two also only deliver the result 1.0 or 0.0.

a & b Logical 'And': **&**
If a is unequal 0.0 and b is unequal 0.0 the result is 1.0, if not 0.0.

Another example:

a**>**3 **&** a**<**5

If a is larger than 3 and smaller than 5: result 1.0.
If a is smaller than 3 or larger than 5: result 0.0.

a | b Logical 'Or': If a is unequal 0.0 or b is unequal 0.0 the result is 1.0, if not 0.0. **|**

?, : In addition you can write *conditional expressions*.

An example:

a **>** 5 ? 17 : 25

The result is either 17or 25.
When a is larger than 5 the result is 17, otherwise 25.

Priorities Priorities

The order in which the different operators are executed depends on their ranking.

Operators with identical ranks are executed from left to right.

If you want operations with lower priority to be executed before ones with higher priority you have to set them in parentheses:

1+2*3=7
but
(1+2)*3=9

Negations (-a = 0.0-a) have the highest priority.
The second highest priority is given to: ^, *, /, % .
Then come +, - (- for substractions).
After that =, !=, #, <, <=, >, >= will be executed.
The lowest priority is given to: **&, |** (and, or).

[functions] [functions]

When you click on [functions] (F6) in the *menu bar* at the bottom you will find a large range of functions.

SQRT (x) marks the square root of x. SQRT (
A value x < 0 is not defined.

SIN (x) Returns the sine of a number (x) in degrees (360.0). SIN (
SIN(0) = 0.0
SIN(90) = 1.0
SIN(270) = -1.0
SIN(-90) = -1.0
SIN(450) = 1.0
SIN(a+/-360) = SIN(a)

When you click on [trigon] you get different sine, cosine and tangent functions:

FASTSIN (x) Is a considerably faster version of SIN. Although it works much less exactly it is sufficient for most applications. FASTSIN (

FASTCOS (x) Is a considerably faster version of COS. Although it works muh less exactly it is sufficient for most applications. FASTCOS (

TAN (x) Returns the tangent of a number (x). TAN (

TAN(x) = SIN(x)/COS(x) **!!!**
COS(x) = 0 is not defined and leads to unpredictable results.

ATAN (x) Returns the arc tangent of a number (x) in degrees. ATAN (
(-180 to 180)
TAN(ATAN(X)) = x

ATANYX (y,x) Arc tangent of y / x (full circle). ATANYX (
It executes the following calculations:
ATAN(y/x). An angle between 0.0 and 360.0 degrees is returned.
y = x 0 0.0 and 360.0° is not defined for this function.

SIN (x) see above

COS (x) Returns the cosine of a number (x) in degrees (360.0). COS (
COS(a) = SIN(a+90)
COS(0)= 1.0

	COS(90) = 0.0 COS(180) = -1.0 COS(360) = 1.0 COS(450) = 0.0	
TRIANGLE (x)	Triangle function of a number (x) with a period of 360. TRIANGLE(0) = 0.0 TRIANGLE(45) = 0.5 TRIANGLE(90) = 1.0 TRIANGLE(180) = 0.0 TRIANGLE(270) = -1.0 TRIANGLE(360) = 0.0 TRIANGLE(a+/-360) = TRIANGLE(a)	TRIANGLE (
SQUARE (x)	Rectangle function of a number (x) with a period of 360.0. SQUARE(x) = 1.0 for x >= 0 and x < 180 SQUARE(x) = -1.0 for x >= 180 and x < 360	SQUARE (
SAW (x)	Saw-tooth function of a number (x).	SAW (

When you click on [exp/log] you find the following functions:

EXP (x)	Exponential function of x with the basis e. (e = 2.71828). EXP(x) = e ^ x.	EXP (
EXP2 (x)	Exponential function of x with the basis 2. EXP2(x) = 2.0 ^ x.	EXP2 (
EXP10 (x)	Exponential function of x with the basis 10. EXP10(x) = 10.0 ^ x.	EXP10 (
LN (x)	Natural logarithm of x to the base e. The logarithm is the inverse function of the exponential function: LN (EXP(a)) = a. For x<=0 the result is not defined.	LN (
LD (x)	Natural logarithm of x to the base 2. The logarithm is the inverse function of the exponential function: LD (EXP2(a)) = a. For x<=0 the result is not defined.	LD (
LOG (x)	Natural logarithm of x to the base 10. The logarithm is the inverse function of the exponential function: LOG (EXP10(a)) = a. For x<=0 the result is not defined.	LOG (

Under [cut] you will find the following functions:

ABS (x)
Returns the absolute value of a number (x).
ABS (

INT (x)
INT(1) = 1
INT(1.23) = 1
INT(-1) = -1
INT(-1.23) = -2
INT (

TRUNC (x)
Only considers the digits before the decimal point.
TRUNC(1.23) = 1.0
TRUNC(-2.345) = -2.0
TRUNC (

FRAC (x)
Only looks at the digits behind the decimal point of numbers (x).
FRAC(1.0) = 0.0
FRAC(2.2345) = 0. 2345 etc.
FRAC (

MOD (x,y)
Modulo
Returns the remainder of a division x by y.
MOD(7,4) = 3
MOD(12.34,12) = 0.34
MOD (

Under [specials] you will find a large range of Lasergraph DSP specific functions:

TimerTicks ()
Returns the number of timer ticks since the start of the film. The timer runs with a frequency of 20833 Hz.
(→ 20833 timer ticks = 1 second)
TimerTicks ()

Seconds ()
Returns the time in seconds that has passed since the start of the film.
Seconds ()

LoopTime ()
Returns the current time of the time loop in seconds.
LoopTime ()

EndTime ()
While a script is running: Displays the time remaining until the current entry is finished.
EndTime ()

TrueSeconds ()
Returns the time that has passed since the start of the film. The resolution is 1/20000 of a second.
TrueSeconds ()

ChaseDelay ()
Returns the value set in the last used ChaseDelay command.
ChaseDelay ()

XOfBuffer (bn,x)

Returns or defines the x-coordinate of a point (x) from the picture which is located in *buffer* (bn).
The value lies between -1.0 and 1.0.
x=0.0 is the first point and x=1.0 is the last point.
The coordinate values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`XOfBuffer (`

YOfBuffer (bn,x)

Returns or defines the y-coordinate of a point (x) from the picture which is located in *buffer* (bn).
The value lies between -1.0 and 1.0.
x=0.0 is the first point and x=1.0 is the last point.
The coordinate values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`YOfBuffer (`

ZOfBuffer (bn,x)

Returns or defines the z-coordinate of a point (x) from the picture which is located in *buffer* (bn).
The value lies between -1.0 and 1.0.
x=0.0 is the first point and x=1.0 is the last point.
The coordinate values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`ZOfBuffer (`

NOfBuffer (bn)

Returns or defines the number of points in the picture which is located in *buffer* (bn).
The number of the *buffer* (bn) has to be between 0 and 7.

`NOfBuffer (`

MOfBuffer (bn,x)

Returns or defines the matrix of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 and 7.
x=0.0 is the first point and x=1.0 is the last point.
The matrix values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`MOfBuffer (`

ROfBuffer (bn,x)

Returns or defines the repeat value of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 and 15.
x=0.0 is the first point and x=1.0 is the last point.
The repeat values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`ROfBuffer (`

`VOfBuffer (bn,x)`

Returns or defines the vector flag (on/off) of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 (vector on) and 1 (vector off).
x=0.0 is the first point and x=1.0 is the last point.
The vector values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`VOfBuffer (`

`BOfBuffer (bn,x)`

Returns or defines the blank value (on/off) of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 (blank off) and 1 (blank on).
x=0.0 is the first point and x=1.0 is the last point.
The blank values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`BOfBuffer (`

`POfBuffer (bn,x)`

Returns or defines the *palette* value of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 and 3.
x=0.0 is the first point and x=1.0 is the last point.
The palette values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`POfBuffer (`

`COfBuffer (bn,x)`

Returns or defines the color value of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0 and 15.
x=0.0 is the first point and x=1.0 is the last point.
The color values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`COfBuffer (`

`IOfBuffer (bn,x)`

Returns or defines the intensity of a point (x) from the picture which is located in *buffer* (bn).
The value lies between 0.0 and 1.0.
x=0.0 is the first point and x=1.0 is the last point.
The intensity values of all points in-between are interpolated.
The number of the *buffer* (bn) has to be between 0 and 7.

`IOfBuffer (`

`XTFM (pl,x,y,z)`	Returns the x-coordinate of a point with the coordinates x, y, z after the transformation with position list pl. The value has to be between 0 and 9.	`XTFM (`
`YTFM (pl,x,y,z)`	Returns the y-coordinate of a point with the coordinates x, y, z after the transformation with position list pl. The value has to be between 0 and 9.	`YTFM (`
`ZTFM (pl,x,y,z)`	Returns the z-coordinate of a point with the coordinates x, y, z after the transformation with position list pl. The value has to be between 0 and 9.	`ZTFM (`
`HTFM (pl,x,y,z)`	Returns the x-coordinate of a point with the coordinates x, y, z after the transformation with position list pl. The value has to be between 0 and 9. The returned value indicates the position in the output plane and can be used for instance to determine the edges of the picture.	`HTFM (`
`VTFM (pl,x,y,z)`	Returns the y-coordinate of a point with the coordinates x, y, z after the transformation with position list pl. The value has to be between 0 and 9. The returned value indicates the position in the output plane and can be used for instance to determine the edges of the picture.	`VTFM (`

RedCBuf (cb,x)

Returns or defines the value of red of the color "x" from the color buffer "cb".

`RedCBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

GreenCBuf (cb,x)

Returns or defines the value of green of the color "x" from the color buffer "cb".

`GreenCBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

BlueCBuf (cb,x)

Returns or defines the value of blue of the color "x" from the color buffer "cb".

`BlueCBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

C4CBuf (cb,x)

Returns or defines the value of the 4th laser line (typically cyan) of the color "x" from the color buffer "cb".

`C4CBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

C5CBuf (cb,x)

Returns or defines the value of the 5th laser line (typically indigo) of the color "x" from the color buffer "cb"

`C5CBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

C6CBuf (cb,x)

Returns or defines the value of the 6th laser line (typically yellow) of the color "x" from the color buffer "cb".

`C6CBuf (`

cb: color buffer (0 – 7)
x: color index (0 – 15)

RealHour () Returns the current time of day in hours. `RealHour ()`

RealMin () Returns the current time of day in minutes. `RealMin ()`

RealSec () Returns the current time of day in seconds. `RealSec ()`

RealDay () Returns the current date in days. `RealDay ()`

RealMonth() Returns the current date in months. `RealMonth ()`

RealYear () Returns the current date in years. `RealYear ()`

KeyVar (x) *Variable* (0..63) which is assigned to a midi note in the *KeyConfig*. If the corresponding midi note is pressed KeyVar(x) has the value 1. The value 0 stands for no action. `KeyVar (`

Rnd () Returns a random value between 0.0 and 1.0. `Rnd ()`

RndOf (x) Returns a random value between 0.0 and 1.0.
The value you specify is the starting value of the pseudo random generator. `RndOf (`

Array (x) Returns or defines one element of the array. `Array (`

Writing into the array: Array (x) = value
Reading out of the array: variable = Array (x)

The index "x" has to be in the range from 0 to 9999.

Note:
The array uses the same memory area in the DSP as the picture buffer 7. Therefore picture buffer 7 must not be used together with the function "Array()" since this can lead to unwanted behavior.

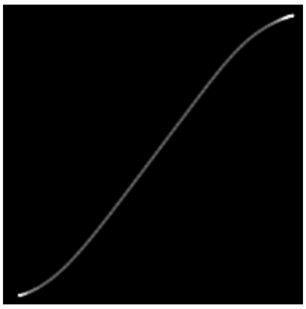
Shared (x) Returns or defines the value of one of the 64 shared variables (0 - 63). Shared variable can be accessed by different Trickfilms and therefore can be used to communicate between different DSPs. `Shared (`

Example:
var = Shared (0)
copies the value of the first shared variable to the variable with the name "var".

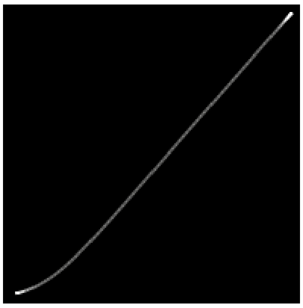
Shared (1) = 12
sets the value of the second shared variable to 12.

GlobalValue (x) Returns the global value number x.
Global values can be defined using the command `SetGlobalValue`. `GlobalValue (`

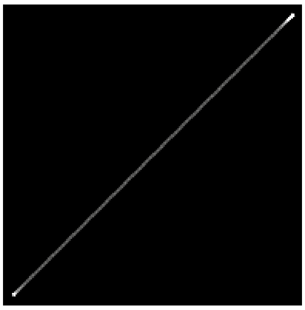
Attack0 (x) Predefined variable with accelaration and deceleration, runs from 0 to 1. Attack0 (



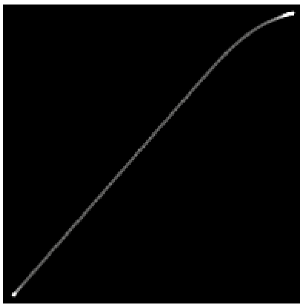
Attack1 (x) Predefined variable with accelaration, runs from 0 to 1. Attack1 (



Attack2 (x) Predefined variable without accelaration and deceleration (linear), runs from 0 to 1. Attack2 (

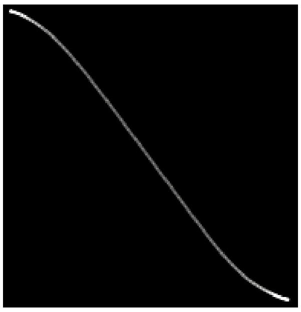


Attack3 (x) Predefined variable with deceleration, runs from 0 to 1. Attack3 (



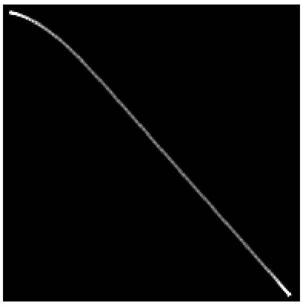
Decay0 (x) Predefined variable with accelaration and deceleration, runs from 1 to 0.

Decay0 (



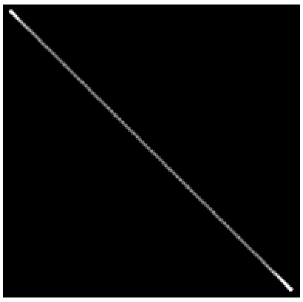
Decay1 (x) Predefined variable with accelaration, runs from 1 to 0.

Decay1 (



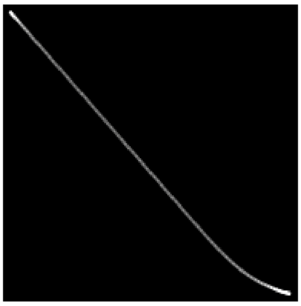
Decay2 (x) Predefined variable without accelaration and deceleration (linear), runs from 1 to 0.

Decay2 (



Decay3 (x) Predefined variable with deceleration, runs from 1 to 0.

Decay3 (



ScaleTo (x, lower_limit, upper_limit)	Scales a value between 0 – 1 from "lower-limit" to "upper-limit" Example: ScaleTo(Fader(0), 1, 360)	`ScaleTo (`
ClipScale (x, lower_limit, upper_limit)	In contrast to the function "ScaleTo()" which scales an input value in the range from 0.0 to 1.0 to the specified value range from "lower limit" to "upper limit", the function "ClipScale()" limits the input value to the specified value range and then scales the value to the range 0.0 to 1.0.	`ClipScale (`
TextLength ()	Returns the number of letters in the text last defined with the command `DeFineText`.	`TextLength ()`
AniPictures ()	Returns the number of frames in the animation last used by the command "`DisplayAnimation`" or "`SelectAnimationPic`"	`AniPictures ()`
SmpteOffset ()	In Run Mode this function returns the value 0. In Play Mode (Timescript) it returns the starting time of the entry in the timescript.	`SmpteOffset ()`
Smpte ()	In Run Mode it returns the value of the time since the film has been started. In Play Mode (Timescript) it returns the value of the time of the timescript.	`Smpte ()`
Fader (x)	Returns the fader value (0.0 - 1.0) of the fader (x).	`Fader (`
FaderB (x)	Returns the value (0, 1) of the button of fader (x).	`FaderB (`
JoyX (x)	Returns the x-value (-1.0 – 1.0) of the joystick (x).	`JoyX (`
JoyY (x)	Returns the y-value (-1.0 – 1.0) of the joystick (x).	`JoyY (`
JoyB (x)	Returns the state of the buttons of the joystick (x): 0 = no button is pressed 1 = button 1 is pressed 2 = button 2 is pressed 3 = both buttons are pressed	`JoyB (`

BeamTabX (x)
Returns the x-position of (-1.0 - 1.0) of the beam (x) from the defined beam table.
`BeamTabX (`

BeamTabY (x)
Returns the y-position of (-1.0 - 1.0) of the beam (x) from the defined beam table.
`BeamTabY (`

BeamTabE (x)
Returns the status (0, 1) whether beam (x) is activated in the beam table.
`BeamTabE (`

DmxIn (ch)
Returns the DMX value (0.0 - 1.0) of the channel (ch) from the DMX input.
`DmxIn (`

value = DmxIn (ch)

ch: DMX channel 1 - 512
value: 0.000 - 1.000

The channel number entered here is affected by the externally adjusted DMX base address and by the "DMX-In offset" within the Advanced Display Parameters.

DmxOut (ch)
Controls one channel of the DMX output.
`DmxOut (`

DmxOut (ch) = value

ch: DMX channel 1 - 256
value: 0.000 - 1.000

If "value" is a decimal value in the range 0-255 the following conversion can be used:

DmxOut (ch) = value / 256

DmxOut16 (ch)
Controls two consecutive channels of the DMX output which are combined into one 16-bit channel.
`DmxOut16 (`

DmxOut16 (ch) = value

ch: DMX channel 1 - 255
value: 0.000 - 1.000

The DMX channel "ch" will get the upper 8-bit (MSB) and the DMX channel "ch+1" the lower 8-bit (LSB) of the 16-bit "value".

If "value" is a decimal value in the range 0-65535 the following conversion can be used:

DmxOut16 (ch) = value / 65536

MidiNote (ch,note)

MidiNote (

Returns the last received value of the Midi commands "Note On" / "Note Off" on the Midi channel "ch".
The returned value indicates the press / release velocity. It has a range from -1.0 to 1.0.
Positive values indicate a pressed key, negative values a released key.

ch:	Midi channel 1 - 16
note:	Note number 0 - 127
result:	-1.0 - 1.0

MidiControl (ch, ctrl)

MidiControl (

Returns the last received value of the Midi command "Control Change" for the controller 0 - 31 (MSB) combined with the controller 32 - 63 (LSB) on the Midi channel "ch".

ch:	Midi channel 1 - 16
ctrl:	0 - 31
result:	0.0 - 1.0

MidiSwitch (ch, sw)

MidiSwitch (

Returns the last received value of the Midi command "Control Change" for the controller 64 - 95 (also known as switches) on the Midi channel "ch".

ch:	Midi channel 1 - 16
sw:	0 - 31 (equivalent to controller 64 - 95)
result:	0.0 - 1.0

MidiProgram (ch)

MidiProgram (

Returns the last received value of the Midi command "Program Change" on Midi channel "ch".

ch:	Midi channel 1 - 16
result:	0 – 127

MidiPitch (ch)

MidiPitch (

Returns the last received value of the Midi command "Pitch Bend Change" on the Midi channel "ch".

ch:	Midi channel 1 - 16
result:	0.0 - 1.0

You can also write instructions which are linked to a condition.
When you click on [**IF-THEN**] you will find: [IF-THEN]

IF With IF you initiate the condition. IF

THEN Initiates the instruction where you define what is to happen when the condition is fulfilled.
An example:
IF a > 360 THEN a = 180 THEN

ELSIF Can be tied in with an IF-THEN instruction to define a second condition which in turn is followed by another THEN with a corresponding instruction. ELSIF

ELSE Can be tied in with an IF-THEN instruction and initiates the instruction defining what is to happen when the IF-condition is not fulfilled. ELSE

END An IF-THEN *variable* has to be concluded with END. END

Displayed command line in the film:
SetVariables
Position in the program:
Usually within the time loop
Keyboard shortcut: SV

3.12.2 SetVAR

The SetVAR command basically is an enhancement of the SetVariables command.

→ Chapter 3.12.1 SetVariables

It simplifies determining the value of a *variable* because under Value you can set the value of the *variable* continuously directly by *mouse poti* and immediately see the change.

On the other hand if you change a value in SetVariables you first have to type, then confirm the entry and only then can you see the result.

An example:

	Entry

Pos = 0;
Time = 0
	SetVariables

	Loop 0
Variable Value Speed 1.0000	SetVAR

NewTime = Seconds ();
dt = NewTime – Time;
Time = NewTime;
Pos = Frac (Pos + Speed * dt);
LoopCnt = Pos
	SetVariables

RotateZ -0.0000 -360.0000	InitPosition
	DisplayPicture
	LoopEnd

In the example above the graphic would rotate around the z-axis exactly once per second depending on how long the *Entry* is called up for instance in the *Timescript*.

With Value 2.0000 the rotation would be executed in a half second, with 0.5000 it would take 2 seconds etc.

If you do not like it this way you can change the Value interactively by *mouse poti* until you are satisfied with the rotation speed.

Of course you could also leave out the `SetVAR` command and for `SetVariables` enter a value instead of 'Speed':

NewTime = Seconds ();
Dt = NewTime – Time;
Time = NewTime;
Pos = Frac (Pos + 1 * dt);
LoopCnt = Pos
	`SetVariables`

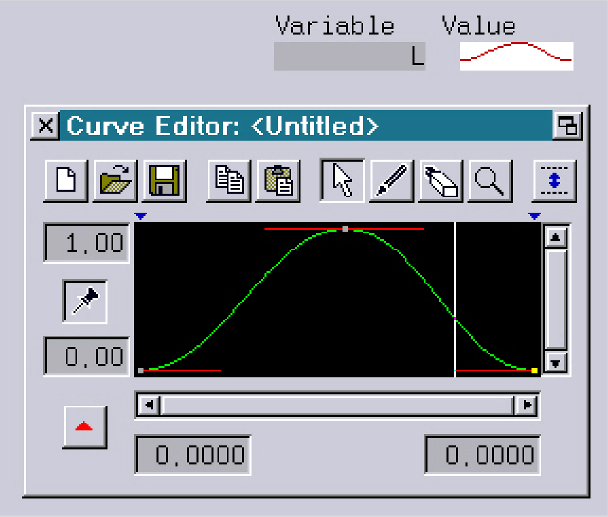
Now each time you want to change the movement speed you have to call up `SetVariables`, find the correct spot in the entry line, enter a different value, press Return, and finally you can see the result. This is much more complicated.

In the following two examples we make a graphic rotate on a circular path with the help of a *curve*.

→ *Curve Editor*

In example 1 we define the *curve* with the `SetVAR` command.

Example 1

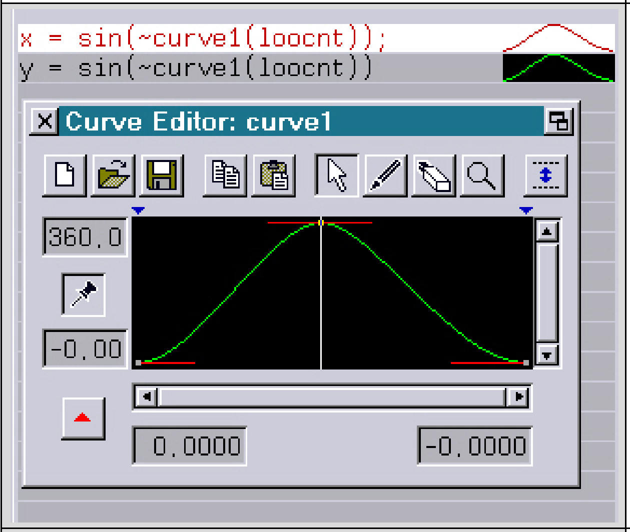
	Entry
	TimeLoop
 Variable Value L Curve Editor: <Untitled> 1,00 0,00 0,0000 0,0000	SetVAR

x = sin(L*360);
y = cos(L*360)
	SetVariables

MoveX x MoveY y	InitPosition
	DisplayPicture
	LoopEnd

Example 2

	Entry
	TimeLoop


x = sin(~curve1(loocnt));
y = sin(~curve1(loocnt))
 	
	SetVariables

MoveX x MoveY y	InitPosition
	DisplayPicture
	LoopEnd

When working with *curves* as *variables* using SetVAR has advantages as well as disadvantages.

If a *curve* is entered directly under SetVariables as in example 1 the *curve* will get a name and it can be used under this name again elsewhere. It then only has to be edited once in order to be altered everywhere. This can be an advantage but it can also become a problem if you change the *curve* and then realize that you would have needed the unchanged *curve* in another spot...

This cannot happen when you use the *curve* under SetVAR because here the *curve* does not get a name.
Of course you can save the *curve* in the *Curve Editor* under a name and load and use it again elsewhere.

Displayed command line in the film:
SetVAR
Position in the program:
within the time loop
Keyboard shortcut: svar

3.13 Global Settings

3.13.1 GlobalParameter

With this command you can downsize and shift the picture output in relation to the *Globals* set in the main menu.

→ MainMenu
`Globals`

Except for a few very special applications you should only make these global settings in the main menu.

💣

Once you have set the `GlobalParameters` in the trickfilm and started the film the size and position set there will be used until they are set elsewhere in the trickfilm.
If you actually use `GlobalParameter` in the trickfilm for any - hopefully good - reason you should definitely put a `GlobalParameter` command into the *subroutine* and there set the default values.
Under normal circumstances you should avoid this command in the trickfilm!

→ Chapter 3.2.3
`Subroutine`

`PictureSize hor:`

`PictureSize hor`

With this you influence the horizontal size of the whole picture output.

The default value is 1.0, i.e. the horizontal values you set in the main menu are displayed unchanged.
If you enter a negative value the picture is displayed mirrored.

`PictureSize vert:`

`PictureSize vert`

With this you influence the vertical size of the whole picture output.

The default value is 1.0, i.e. the vertical values you set in the main menu are displayed unchanged.
Negative values turn the displayed picture upside down.

`X = Y`

`X = Y`

This parameter makes the size adjustment easier because you can set the x- and y-size simultaneously proportionally.

You can only make this adjustment with the *mouse poti* by clicking on the field and moving the mouse while keeping the left mouse button pressed until the desired size is reached. When the middle mouse button is pressed a finer adjustment is possible.

PictureShift hor:

PictureShift hor

With this you influence the horizontal position of the whole picture output.

The default value is 0.0, i.e. the horizontal values you set in the main menu are displayed unchanged.

PictureShift vert:

PictureShift vert

With this you influence the vertical position of the whole picture output.

The default value is 0.0, i.e. the vertical values you set in the main menu are displayed unchanged.

X & Y

X & Y

Here you can change the horizontal and the vertical position simultaneously with the mouse.
The entry method is the same as for X = Y.

Displayed command line in the film:
`GlobalParameter`
Position in the program:
outside the time loop
Keyboard shortcut: `gp`

3.13.2 Global3DParameter

You only need this command to display stereoscopic 3-D projections.

→ Advanced display parameters

If you work with color filter glasses (red/green or red/blue) you have to additionally enter the command `Anaglyph`. For polarized 3-D projections this command is not relevant.

→ Chapter 3.13.4 Anaglyph

Distance of eyes:

Distance of eyes

With this parameter you define the distance between the right and left eye of the viewer.

The larger you choose this distance the farther apart the two pictures will be set.

Projection plane dist:

Projection plane dist

This parameter defines the imaginary distance of the graphic in the object space.

Displayed command line in the film:
`Global3DParameter`
Position in the program: Outside the time loop
Keyboard shortcut: `g3dp`

3.13.3 SET3DMODE

The command "SET3DMODE" enables you to control the external parameters "3D external sync" and "3D eye control" for stereoscopic 3D projections from the trickfilm.

→ Advanced display parameters

3D external sync:

3D external sync

The following options are available:

```
Off
Low=Left
High=Left
#Extern
```

3D eye control:

3D eye control

The following options are available:

```
Both
Left
Right
#Extern
```

An example of use is the display of a 3D animation, rendered with 3ds Max and converted for laser display via proTize max. The converter generates an ongoing animation whose pictures are alternating intended for the left and the right eye.

In this case, the required command sequence would look like this:

```
Output loop
    SET3DMODE  =>  3D eye control = Left
    display left picture
    SET3DMODE  =>  3D eye control = Right
    display right picture
End of loop
```

An example can be found in the Demofiles:

C:\DemoFiles\Templates\proTize3D.FILM

Displayed command line in the film:
`SET3DMODE`
Position in the program:
depending on the application inside or outside the output loop

3.13.4 AnaGlyph

This command is needed for stereoscopic 3-D projections with red/green resp. blue/green filter glasses.

In advance we should mention that the red/green or blue/green filter method only allows a monochromatic 3-D display.

The red filter does not let green or blue through, the blue and green filters stop red. However, the blue filter lets green pass and the green filter blue.

You take a picture which for instance is gray, double it and color one of the resulting pictures red and the other one green. You place these pictures on top of each other and slightly shift them apart horizontally.

If you then put on a pair of red/green filter glasses you see the red picture with the 'red' eye and the green picture with the 'green' eye. The brain combines these two pictures, creating a three-dimensional impression and mixing the two colors into a yellow hue.

For blue/red glasses you simply color one of the pictures blue instead of green. The final result is generally the same.

It is however possible to achieve a certain color impression by changing the color intensity for the right or the left eye.

If you for instance have a red picture to start with, for the 3-D transformation the red picture will be displayed with 100% while the green resp. blue one will only be displayed with something like 15%. This is sufficient to achieve a 3-D effect but the red color impression will dominate.

→ Advanced display parameters

There already is a default setting in the main menu which usually does not have to be changed.
Therefore all parameters are set to `#Extern` when you call up the command in the trickfilm.
You are surely well advised to leave them that way.

Mode

Mode:

Here you can choose between ON and OFF.

Red by green: `Red by green`

Here you determine the intensity for the left eye (red).

If your original picture or a part of it is pure green, a weak red picture will be automatically generated here by the default settings whose intensity is chosen so that the 3-D effect will be retained while green dominates.

The external default setting is 0.13 (13% red by 100% green).

Red by blue: `Red by blue`

Here you determine the intensity for the left eye (red).
The external default setting is 0.09 (9% red by 100% blue).

Green by red: `Green by red`

Here you determine the intensity for the right eye (green).
The external default setting is 0.00.

Blue by red: `Blue by red`

Here you determine the intensity for the right eye (blue).
The external default setting is 0.12 (12% blue by 100% red).

Displayed command line in the film:
`AnaGlyph` and `ON` or `OFF`
Position in the program:
outside the time loop
Keyboard shortcut: `ag`

3.13.5 SetFrequency

With this parameter you influence the frequency of the picture output.

→ Advanced display parameters

When you use this command in the trickfilm the setting in the main menu under `Advanced display parameters` will become ineffective. All following pictures will be displayed with the newly selected frequency until you revoke it elsewhere.

If you change the scanning frequency anywhere in the trickfilm you should definitely put a `SetFrequency` command into a *subroutine* you call at the beginning of every Entry and there set it to `#Extern`.

→ Chapter 3.2.3 Subroutine

Desired frequency (kHz):

Desired frequency (kHz)

This parameter as well as all the following ones is set to a default setting of `#Extern`.
This means that the scanning frequency set in the main menu will be used. The default setting is about 30,000 kHz.

A higher frequency usually only makes sense for special applications such as *abstracts* or *bitmaps*. When using *bitmaps* for instance it is quite possible to achieve significant quality increases with frequencies of up to 120,000 kHz.

Real frequency (kHz):

Real frequency (kHz)

Here the frequency which is actually displayed by the respective DSP type is displayed.

Displayed command line in the film:
`SetFrequency`
Position in the program:
outside the time loop
Keyboard shortcut: `sf`

3.13.6 SCanningParameter

This command offers many possibilities to influence the *scanner output*.

→ Advanced display parameters

Normally these settings should be done in the main menu (like almost all global settings!).

This command should be used carefully in the trickfilm because it turns off the *scanning* parameters of the main menu and influences the subsequent picture output until you revoke it.

☜

If there actually is a reason to program the command in the trickfilm you should not forget to include it in a *subroutine* you call at the beginning of every Entry and set all values to `#Extern` there

→ Chapter 3.2.3 Subroutine

`MaxdistDrawn:`

MaxdistDrawn

The DSP will execute an automatic interpolation for the picture output if necessary.
That means it will insert points if the distance between two points exceeds the value set here.

The default value is 0.0900, whether you chose *Scanner* or *Crystal* in the preset.
If you choose too small a distance between the points there is the risk that the picture will flicker because simply too many points will be inserted.
If you set the distance too far the quality of the picture output can suffer significantly.

→ Advanced display parameters

`MaxdistBlanked:`

MaxdistBlanked

Generally this parameter has the same effect as the previous one, with the difference that here the *blanked* lines are interpolated.

`ToStartPoint:`

ToStartPoint

When the last point of a graphic is reached the scanners jump back to the first point ands start over again. To make sure that the first point has been reached a little delay in the form of *repeats* is inserted.
Here you can set the number of *repeats* according to the speed of your x-y *scanners*.

`PreBlankingOn:` PreBlankingOn

Here you determine the delay of the projection before a *blanked* point in the form of *repeats*.
This way you give the *scanners* enough time to reach the point where they are to fade out the beam.

`PostBlankingOn:` PostBlankingOn

Here you determine the delay of the projection on a *blanked* point in the form of *repeats*.

`PreBlankingOff:` PreBlankingOff

Here you set the delay before the *blanking* is turned off.
This way you give the *scanners* enough time to reach the point where they are to fade the beam in again.

`PostBlankingOff:` PostBlankingOff

With this parameter you can delay the projection to give the *blanker* enough time to fade the beam in again.

`Color delay:` Color delay

Here you determine the delay of the projection before a color change in the form of *repeats*.
This way you give the *scanners* enough time to reach the point where you want the color change to happen.
A very fast color control such as a *crystal* does not need a delay here.

Displayed command line in the film:
`SCanningParameter`
Position in the program:
outside the time loop
Keyboard shortcut: `scp`

3.13.7 SetPointReduction

This command practically has the opposite effect of an *interpolation*. Here you can set the distance between individual points from which on *repeats* will be reduced and points will be eliminated.

This can make sense if you drastically reduce the size of a very complex picture which then gets too bright in some places because the individual points are too close together.

`Reduction distance:`

`Reduction distance`

When the distance reaches or falls below the value you set here, *repeats* will automatically be reduced.

Setting the distance to 0.0 will disable the reduction of repeats.

`Skip distance:`

`Skip distance`

When the distance reaches or falls below the value you set here, points will automatically be dropped.

Setting the distance to 0.0 will disable the dropping of points.

Displayed command line in the film:
`SetPointReduction`
Position in the program:
outside the time loop
Keyboard shortcut: `spr`

3.13.8 SetRepeatMap

With this command you can change individual *repeats* for the picture output, i.e. you change the optical effect of the picture without changing the picture itself.

You can for instance make corners sharper or smoother or project the same circle or fan of a beam show with a structure one time and smoothly another time.

You can also make these settings in the main menu. However, this only makes sense if you want to change certain *repeats* for all used graphics.

→ Advanced Parameters Repeat Mapping

Otherwise you should retain the default values of the Repeat Mapping in the main menu and program changes specifically in the trickfilm.
However, you should not forget to include SetRepeatMap with the external default setting in a *subroutine* you call at the beginning of every Entry!

→ Chapter 3.2.3 Subroutine

You can only influence picture points with a *repeat* of 1 or more.

!!!

In addition repeat mapping is not applied to points of blank change.

!!!

Map repeat 1-15 to:

Map repeat 1-15

Here you assign new values to the actual repeats.
For instance if you enter:

```
Map repeat 1 to:  5
```

all points of the graphic which were digitized with *repeat* 1 will be projected with *repeat* 5.

In addition to fixed values it is also possible to enter variables or curves to achieve a dynamic change of the number of repeats.

Displayed command line in the film:
SetRepeatMap
Position in the program:
outside the time loop (inside the time loop if a dynamic change shall be performed)
Keyboard shortcut: srm

3.13.9 ColorMode

With this command you can adapt the Lasergraph DSP to different color controls, even slow ones.
We recommend that you make these settings once in the main menu and if necessary save them with the globals.

→ Advanced display parameters

Slow color mode:

Slow color mode

This parameter should be set to ON if color and *blanking* are not controlled by a *crystal* but for instance by a *color box* and *scanners*.

It has the result that the color is switched independent of the *blanking*, i.e. the color is not switched to black automatically as soon as you enter a *blanking*.

Color switch repeat:

Color switch repeat

Here you can increase the *repeats* for a color change if you are working with a very slow color control.
This way you give the *scanners* enough time to reach the point where you want the color change to happen.

Color inversion:

Color inversion

Here you can invert the color output. Certain color controls work with inverted outputs and here you can make the adjustment.

NOTE: Color inversion is only available on old DSP-cards but still implemented due of compatibility.

!!!

Displayed command line in the film:
`ColorMode`
Position in the program:
outside the time loop
Keyboard shortcut: `cm`

3.13.10 SelectDGC

→ `Screen Parameter`

With the `SelectDGC` command you can access up to 16 *DGC* settings in the trickfilm which you previously defined in the main menu (`Set 0-15`).

This is especially useful when you have to switch between different projection surfaces within one show.

`Setting:`

`Setting`

Here you enter the number (0-15) of the *DGC setting* you want to use for the specific trickfilm sequence or #Extern.

`Name:`

`Name`

Here the name of the currently selected DGC setting is displayed.

Displayed command line in the film:
`SelectDGC` and `#Extern` or the number of the set you entered
Position in the program:
outside the time loop
Keyboard shortcut: `sdgc`

3.14 Special commands

3.14.1 SendMIDI

The SendMIDI command is used to send MIDI Data through the MIDI-Output (e.g. to control a musical instrument). `SendMIDI`

Please make sure that the "Midi Out Mode" is set to "Midi" in the Midi Out Setup (see chapter "Setup").

Note: **!!!**
The command "SendMIDI" will only be executed during the running Trickfilm. The command will be ignored during editing.

`Midi Command` `Midi Command`
Choose from these command types:

```
NoteON
NoteOFF
Control
Control14
Program
PitchBend
```

`NoteON` `NoteON`
Transmits the Midi command "Note On".

Parameters:	Channel	(1-16)
	Note	(0-127)
	Velocity	(0.0-1.0)
Data sent:	`%1001nnnn`	(nnnn = channel number 0-15)
	`%0kkkkkkk`	(kkkkkkk = note number 0-127)
	`%0vvvvvvv`	(vvvvvvv = key velocity 0-127)

Note: A velocity of "0" is defined as "Note Off"

`NoteOFF` `NoteOFF`
Transmits the Midi command "Note Off".

Parameters:	Channel	(1-16)
	Note	(0-127)
	Velocity	(0.0-1.0)
Data sent:	`%1000nnnn`	(nnnn = channel number 0-15)
	`%0kkkkkkk`	(kkkkkkk = note number 0-127)
	`%0vvvvvvv`	(vvvvvvv = key velocity 0-127)

Control

Control

Transmits the Midi command "Control Change" as 7-bit value.

Parameters:	Channel	(1-16)
	Controller	(0-127)
	Value	(0.0-1.0)

Data sent:	%1011nnnn	(nnnn = channel number 0-15)
	%0ccccccc	(ccccccc = controller number 0-127)
	%0vvvvvvv	(vvvvvvv = controller value 0-127)

Control14

Control14

To increase the resolution the controllers 0-31 can be combined with the controllers 32-63.
If this is done the controllers 0-31 represent the most significant bits (MSB) and the controllers 32-63 the least significant bits (LSB).

To address such a controller the command type "Control14" sends the Midi command "Control Change" as 14-bit value.

Parameters:	Channel	(1-16)
	Controller	(0-31)
	Value	(0.0-1.0)

Data sent:	%1011nnnn	(nnnn = channel number 0-15)
	%0ccccccc	(ccccccc = controller number 0-31)
	%0vvvvvvv	(vvvvvvv = controller value MSB 0-127)
	%1011nnnn	(nnnn = channel number 0-15)
	%0ccccccc	(ccccccc = controller number 32-63)
	%0vvvvvvv	(vvvvvvv = controller value LSB 0-127)

Program

Program

Transmits the Midi command "Program Change" as 7-bit value.

Parameters:	Channel	(1-16)
	Program	(0-127)

Data sent:	%1100nnnn	(nnnn = channel number 0-15)
	%0ppppppp	(ppppppp = program number 0-127)

PitchBend

PitchBend

Transmits the Midi command "Pitch Bend Change" as 14-bit value.

Parameters:	Channel	(1-16)
	Value	(0.0-1.0)

Data sent:	%1110nnnn	(nnnn = channel number 0-15)
	%0vvvvvvv	(vvvvvvv = least significant bits 0-127)
	%0vvvvvvv	(vvvvvvv = most significant bits 0-127)

3.14.2 SENDstring

SENDstring

This command sends ASCII-code or hexadecimal values through the serial outputs or via the TCP/IP network. Note that for serial output the SerCmd Setup has to be set up properly.

Examples:

"TCP://192.168.1.1:200/Hello World"
sends the words "Hello World" through the network to the computer with the IP-address 192.168.1.1, port 200.

“Ser://5/Hello World”
sends the words “Hello World” to the SerCmd-Interface # 5.

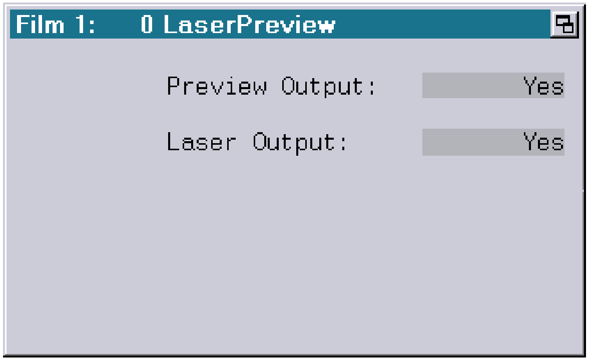
For more details see “Chapter SerCmd”.

Displayed command line in the film:
SENDstring
Position in the program:
outside the time loop
Keyboard shortcut: send

3.14.3 Laser Preview

LaserPreview

The command can be used to determine if the following outputs are to be displayed on the laser and / or in the preview window:



Film 1: 0 LaserPreview

Preview Output: Yes

Laser Output: Yes

Laser Output: "Yes" ⇨ The following outputs are displayed by the projector.
"No" ⇨ The following outputs are not displayed by the projector.

Preview Output: "Yes" ⇨ The following output will appear in the preview window.
"No" ⇨ The following output will not appear in the preview window.

The command `LaserPreview` is for instance used in the Live Mode of the jr.Live!

Here a picture or an effect can be seen in the preview window before it is actually displayed on the laser, independent of the currently running effect.

Example:
The following trickfilm sequence displays a circle on the laser and a square in the preview window:

```
TimeLoop

  LaserPreview     Laser Output = Yes
                   Preview Output = No
  InitPosition
  DisplayPicture        Circle

  LaserPreview     Laser Output = No
                   Preview Output = Yes
  InitPosition
  DisplayPicture        Square

LoopEnd
```

Notes:

- Even when the laser or preview output is disabled the trickfilm commands are still executed.
  Therefore color, position, SWC mask and so on are modified by the corresponding commands even when the output is disabled.
- The number of frames per second (fps) displayed in the Super Preview when "Display Laser Fps" is enabled, is based on the output shown in this preview window.
  The frames per second of the actual laser output might be different.

Attention:

- The command LaserPreview should be used by experienced programmers only. A mistake could cause unpredictable results like no laser output while the preview window still shows the desired output.
- Shows containing the new `LaserPreview` command cannot be run on older systems!

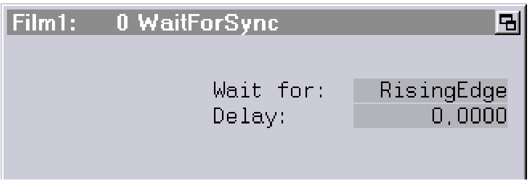
Displayed command line in the film:
`LaserPreview`
Position in the program:
Within the loop, before `Display` command.
Keyboard shortcut: `lp`

3.14.4 WaitForSync

WaitForSync

The command WaitForSync allows the laser output to be synchronized with an external sync signal.

One application example is the synchronization for video recordings:
Without synchronization sampling effects can occur which can lead to shifting pictures or to breaks in the recorded video.



Film1: 0 WaitForSync

Wait for: RisingEdge
Delay: 0.0000

The flank of the sync signal which the DSP is to wait for can be specified in the parameter window of the WaitForSync.

RisingEdge	⇨	The DSP will wait for the sync signal to change its state from low to high.
FallingEdge	⇨	The DSP will wait for the sync signal to change its state from high to low.
AnyEdge	⇨	The DSP will wait for any change in the sync signal state.

In addition to the desired flank a delay in seconds can also be set. Within the trickfilm the command WaitForSync should be placed directly before the picture output command which is to be synchronized.

Example:

```
TimeLoop
 InitPosition
 WaitForSync
 DisplayPicture
LoopEnd
```

Notes:

- In order to use the WaitForSync command, the Mark 2 has to be equipped with a sync input which is not available on all devices.
- If the DSP does not detect the desired flank within one second, the waiting will be aborted automatically.

Displayed command line in the film:
WaitForSync
Position in the program:
Within the loop, before Display command.
Keyboard shortcut: wfs

3.14.5 LimitFrameRate

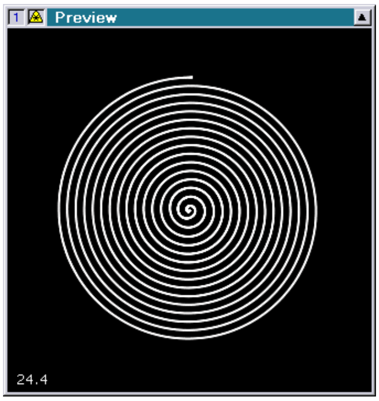
One of the Lasergraph DSP's fortes is to display only the visible parts of a picture.

Certain conditions (drawing / clipping of a picture) may cause certain moments when only very few points are displayed with a high refresh rate.

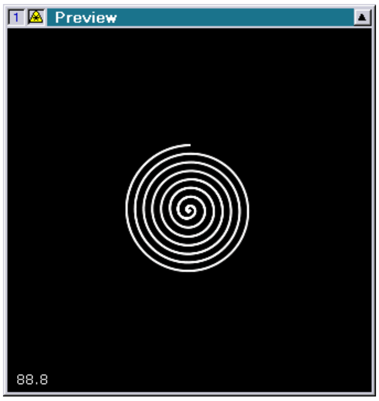
The resulting high power density causes an unpleasant dazzling while watching graphic shows and may be a threat for the viewers' eyes during beam shows.

Example: A spiral is undrawn within a loop

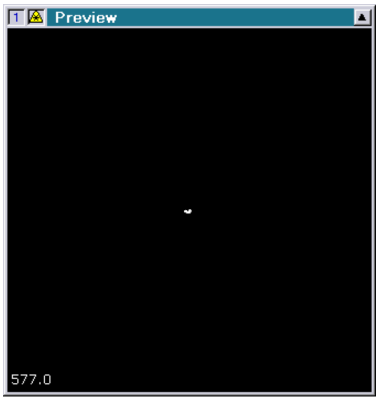
At the beginning the complete graphic is visible. The picture is flickering perceptibly.



Halfway through the loop the flickering is not perceptible any longer.

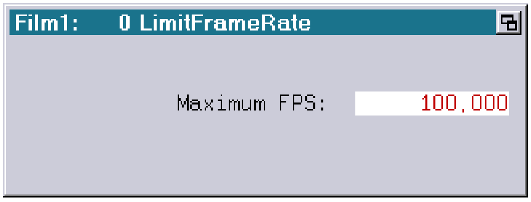


At the end the complete laser power is focused on a very little part of the picture.



`LimitFrameRate`

You can avoid problems like this with the trickfilm command "LimitFrameRate".



Film1: 0 LimitFrameRate

Maximum FPS: 100,000

This command takes care that the picture refresh rate does not exceed a user-definable rate.

The command "LimitFrameRate" calculates the time since the last "LimitFrameRate". If this time span is too short, it is adjusted by waiting accordingly.

Example:

```
ExternTimeLoop
 CopyPictoBuffer
 CopyBuffer
 InitPosition
 DisplayBuffer
 LimitFrameRate
LoopEnd
```

Annotation:

- The display of the current picture refresh rate in the Super Preview ("Display Laser Fps") keeps showing the picture refresh rate resulting from the number of points.
  The waiting time inserted by "LimitFrameRate" will not be taken into account.

Displayed command line in the film:
`LimitFrameRate`
Position in the program:
Within the output loop
Keyboard shortcut: `lfr`

3.15 Sipro Commands

The Sipro (**SI**mple **PRO**jector) is a small and low priced device manufactured by LaserAnimation Sollinger GmbH to control a laser system without a Lasergraph DSP.

This standalone device can be controlled via MIDI or a PC running the Sicon software (**SI**mple **CON**troller).

There are two different Sipro versions:

1. the Sipro I (older hard- and software)
2. the Sipro II / II+

Limitations of the Sipro:

- Only 2D frames are supported.
- The resolution is limited to 8 bit.
- Curves and variables are not supported.
- Clipping, masking and geometric corrections are not supported.

Programming

The programming of the Sipro effects is done using the Lasergraph DSP.

Because the Sipro does not support all DSP commands (that is why it is called Simple!), there is a subset of DSP commands and other commands specially designed for the Sipro which are supported by the Sipro.

These commands can be accessed by clicking into the top area of the Trickfilm Command Insert list.

A mouse click switches from the default window

"*Commands sorted in groups*"

into the

"*All commands in alphabetical order*"

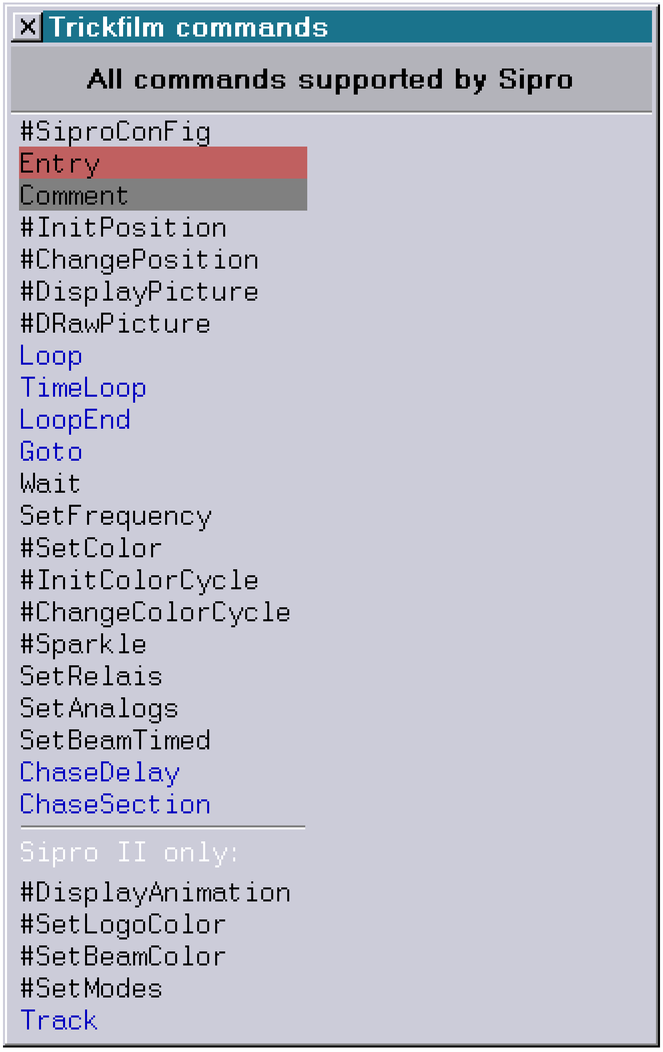
window.

Another click switches to the

"***All commands supported by Sipro***"

window where all commands are displayed that can be used in the Sipro.

The Sipro Insert Window



Trickfilm commands

All commands supported by Sipro

```
#SiproConFig
Entry
Comment
#InitPosition
#ChangePosition
#DisplayPicture
#DRawPicture
Loop
TimeLoop
LoopEnd
Goto
Wait
SetFrequency
#SetColor
#InitColorCycle
#ChangeColorCycle
#Sparkle
SetRelais
SetAnalogs
SetBeamTimed
ChaseDelay
ChaseSection
Sipro II only:
#DisplayAnimation
#SetLogoColor
#SetBeamColor
#SetModes
Track
```

The Sipro can have up to 256 Entries. When programming a Sipro Trickfilm a specific command syntax has to be used.

The Trickfilm Syntax

- The Trickfilm has to start with the command `#SiproConFig`.
- The Trickfilm has to have an Entry number “255”, which is the Entry that is executed after startup.
- Only the commands available in the "Sipro" insert window can be used.

3.15.1 #SiproConFig

`#SiproConFig`

Every Sipro Trickfilm has to start with this command.
It is used to define all the needed settings for the scanning output.

It is similar to the command `SCanningParameter`.

The parameters:

MaxdistDrawn:
The DSP will execute an automatic interpolation for the picture output if necessary. That means it will insert points if the distance between two points exceeds the value set here. The default value is 0.0900, whether you chose Scanner or Crystal in the preset.
If the chosen distance between the points is too small there is the risk that the picture will flicker because too many points will be inserted.

MaxdistBlanked:
Generally this parameter has the same effect as the previous one, with the difference that here the blanked lines are interpolated.

ToStartPoint:
When the last point of a graphic is reached the scanners jump back to the first point and start over again. To make sure that the first point has been reached, a little delay in the form of repeats is inserted.

PreBlankinOn:
Here you determine the delay of the projection before a blanked point in the form of repeats.
This way you give the scanners enough time to reach the point where the beam is to fade out.

PostBlankinOn:
Here you determine the delay of the projection on a blanked point in the form of repeats.

PreBlankinOFF:
Here you set the delay before the blanking is turned off.
This way you give the scanners enough time to reach the point where the beam is to fade in again.

PostBlankinOFF:
With this parameter you can delay the projection to give the projector enough time to fade the beam in again.

Desired frequency:
Defines the scanning frequency.
Value: 5.200 - 20.833 for the Sipro I
Value: 5.200 - 33.333 for the Sipro II

Real frequency:
Can differ from the desired frequency
Value: 5.200 - 20.833 for the Sipro I
Value: 5.200 - 33.333 for the Sipro II

Displayed command line in the film: `#SiproCoFig`
Position in the program: Beginning of the trickfilm
Keyboard shortcut: #scf

3.15.2 Entry

`Entry`

Entries are used to divide the trickfilm into sequences which can be called up independently.

You can use up to 256 entries for a Sipro trickfilm.
Value: 0 - 255

Entries are the entry points into the film. They allow you to call up individual film sequences.

`GotoNext(F6)`
`GotoPrev(F7)`

While the film is running you can call up specific film sequences with the Goto command in the menu bar at the bottom (F5) + desired entry number or jump one entry back or forward with `GotoNext (F6)` resp. `GotoPrev (F7)`.

Displayed command line in the film: Entry and number (0-255)
Position in the program: Beginning of a sequence
Keyboard shortcut: e

3.15.3 Comment

The command is a very useful orientation guide in your films.
It has no influence on the film and you can insert it as often and wherever you like.

Displayed command line in the film: Beginning of comment
Position in the program: Wherever you like
Keyboard shortcut: c

3.15.4 #InitPosition

`Position #:`
Here you enter the identification number for the position list.

The Sipro allows 8 different position lists (0-7).
You can assign these position lists to different pictures.

Parameters:

Size:
Defines the size of the picture.
Value: 0 - 1

RotateZ:
Rotates the picture around its z-axis.
Value: -360,0 - 360,0

RotateY:
Rotates the picture around its y-axis.
Value: -360,0 - 360,0

RotateX:
Rotates the picture around its x-axis.
Value: -360,0 - 360,0

MoveRange:
Defines the amplitude of the following movement.
Value: 0 - 1
A value of 0 will result in no movement.
A value of 1 will result in max. movement.

MoveX:
Moves the picture like a sinus function (oscillation) on the horizontal axis with the amplitude defined in the MoveRange value.
The entered values are equivalent to the degree numbers.
Value: -360,0 - 360,0

MoveY:
Moves the picture like a sinus function (oscillation) on the vertical axis with the amplitude defined in the MoveRange value.
The entered values are equivalent to the degree numbers.
Value: -360,0 - 360,0

Displayed command line in the film: `#InitPosition and number (0-7)`
Position in the program: Inside or outside the loop
Keyboard shortcut: #ip

3.15.5 #ChangePosition

With this command you set a graphic in motion by changing the position defined with the command `#InitPosition`.

The `#ChangePosition` command can only be used in connection with `#InitPosition`.

Position #:
Here you enter the identification number of the corresponding position list.

Mode:
Two modes are available: "Delayed" and "LoopCnt"

Delay:
This value is only relevant in the Mode Delayed!
In this mode, it defines the delay time before another Size, Rotate or Move Value is added to the picture's position.
Value: 0 - 1
Value 0 = no delay
Value 1 = delay of 1 second

Mode Delayed:
In this mode, the `#InitPosition` command should be outside the loop.
The value that is defined in the #ChangePosition command will be added to the value of the
#InitPosition in every cycle of the loop.

That means, e.g. a value of 90 in RotateZ will result in four pictures, each rotated 90 degrees to the one before.

Note: In this mode, the preview might not give you the exact impression of the laser output!

Mode Loopcnt:
In this mode, both commands should be inside the loop.
The value that is defined in the #ChangePosition command will be added to the value of the #InitPosition in the time that is defined in the loop.

Displayed command line in the film: `#ChangePosition and number (0 - 7)`
Position in the program: Within the time loop
Keyboard shortcut: #cp

3.15.6 #DisplayPicture

The command causes the display of a single graphic.

Catalog
Here you define from which catalog you want to select a graphic. Simply click on Catalog: or on the empty field with the '?' to the right of it.

The file window will open.
All catalogs which you have already loaded into the picture memory are listed here. Click on the desired catalog. Immediately a window will open: Pictures in catalog xxx: (name of the selected catalog)

Now click on the desired graphic.

Position #
This value defines to which #InitPosition the picture is assigned.

Displayed command line in the film: `#DisplayPicture`
Position in the program: Within the time loop
Keyboard shortcut: #dp

3.15.7 #DRawPicture

The command causes the drawing of a single graphic.

Catalog
Here you define from which catalog you want to select a graphic. Simply click on Catalog: or on the empty field with the '?' to the right of it.

The file window will open.
All catalogs which you have already loaded into the picture memory are listed here. Click on the desired catalog. Immediately a window will open: Pictures in catalog xxx: (name of the selected catalog)

Now click on the desired graphic.

Position #
This value defines to which #InitPosition the picture is assigned.

Displayed command line in the film: `#DRawPicture`
Position in the program: Within the time loop
Keyboard shortcut: #drp

3.15.8 Loop

The command lets one or more connected sequences of your film run repeatedly.

Every Loop command has to be ended with `LoopEnd` at the end of the sequence which is to be repeated.

→ `LoopEnd`

Number of Repeats:
Here you specify how many times the program sequence is to be run.
Repeats from 1 to 65535 are possible.
With the repeat number 0 you program an endless loop.

Displayed command line in the film: `Loop` with the number of repeats
Position in the program: At the beginning of the sequence which is to be repeated
Keyboard shortcut: l

3.15.9 TimeLoop

With the command you program a time loop for which you specify a fixed duration.

Like every other loop, the `TimeLoop` also **has to** be concluded with `LoopEnd`.

→ `LoopEnd`

The sequence between the commands `TimeLoop` and `LoopEnd` runs for the period of time which you enter.
The display window shows the time in seconds which you can change:

Time in seconds:
Here you enter the time in seconds during which the sequence is to be executed.

The possible period of time ranges from 0.0 to 300.0 seconds.

Displayed command line in the film: `TimeLoop` with time in seconds
Position in the program: At the beginning of the sequence which is to be repeated
Keyboard shortcut: tl

3.15.10 LoopEnd

Each individual `Loop` or `TimeLoop` command has to be ended with the command LoopEnd. Otherwise the compiler will display an error message and your film will not run.

→ `Loop`

Displayed command line in the film: `LoopEnd`
Position in the program: End of the sequence which is to be repeated
Keyboard shortcut: le

3.15.11 Goto

The `Goto` command makes the program jump to a selected entry.

→ Chapter Entry

The command has to be handled carefully. It can lead to very confusing and unaesthetic program structures. You should therefore use it as sparingly as possible.

The command actually only makes sense if you want your program to run endlessly and you use the `Goto` command at the end of the endless loop to get back to the first entry.

Displayed command line in the film: `Goto` with selected entry number (0-255)
Position in the program: Always at the end of a sequence and outside of the time loop
Keyboard shortcut: g

3.15.12 Wait

With the command you interrupt the program for the time you specify. During this time no picture is displayed.
You can interrupt the `Wait` command at any time by calling up the next entry.

→ `Entry`

Time in seconds:
Here you enter the time in seconds for which the program is to be interrupted.
The possible time span ranges from 0.0 to 300.0 seconds.

Displayed command line in the film: Wait with time in seconds
Position in the program: At the end of a time loop
Keyboard shortcut: w

3.15.12 SetFrequency

With this parameter you influence the frequency of the picture output.

When you use this command in the trickfilm the setting in the `#SiproConFig` will become ineffective. All pictures in that Entry will be displayed with the newly selected frequency until you revoke it elsewhere or you start an Entry that has contains no `SetFrequency` command.

Note:
The Sipro II only supports frequencies up to 33.333, the Sipro I up to 20.833. When you select frequencies over 33.333, the Lasergraph DSP will execute these frequencies, the Sipro will not!

Desired frequency (kHz): Desired frequency (kHz)
Defines the frequency.
Value: 5.200 - 20.833 for the Sipro I
Value: 5.200 - 33.333 for the Sipro II

Real frequency (kHz): Real frequency (kHz)
Here the frequency which is actually displayed by the respective DSP type is displayed.

Note:
The Sipro only supports frequencies up to 33.333 !!!

Displayed command line in the film: `SetFrequency`
Position in the program: Outside the loop
Keyboard shortcut: sf

3.15.13 #SetColor

This command overwrites the color information of the picture and displays the whole picture with the color set in the command.

You can choose between these colors:
White - Red - Yellow - Green - Cyan - Blue - Magenta
or the default colors of the picture (picture).

Displayed command line in the film: `#SetColor`
Position in the program: Outside or inside the loop
Keyboard shortcut: #sc

3.15.14 #InitColorCycle

This command should be used in combination with the command `#ChangeColorCycle`.

The `#InitColorCycle` command is used to define values for a color cycle that is initiated with the command #ChangeColorCycle.

You can choose between these colors:
White - Red - Yellow - Green - Cyan - Blue - Magenta
or Black.

Number of colors:
This parameter defines how many of the 16 colors are used for the color cycle.
Value: 1 - 16

Length of segment:
This parameter defines the length of each color segment in the picture.
Value: 1 - 255

Phase:
This parameter defines the position of the selected colors in the picture.
Value: 0 - 255

Color 0 - 15:
Defines the 16 colors.

Displayed command line in the film: `#InitColorCycle`
Position in the program: Outside the loop
Keyboard shortcut: #icc

3.15.15 #ChangeColorCycle

This command should be used in combination with the command `#InitColorCycle`.

The `#ChangeColorCycle` command is used to define the speed of the color cycle.

Delay:
This parameter defines the time in seconds of a delay that is used to slow down the cycle.
Value: 0.0 – 1.0

Increment:
Defines the speed of the color cycle.
1 = slowest possible speed
255 = maximum speed
Value: 0 - 255

Displayed command line in the film:
`#ChangeColorCycle`
Position in the program: Inside the loop
Keyboard shortcut: #ccc

3.15.16 #Sparkle

This command initiates a "sparkle" effect on the picture. It makes individual points of your graphic flash briefly.

Probability:
Here you influence the frequency and number of flashing points.
If you enter 0, the sparkling is turned off.
When you enter 1 all points used in the graphic flash simultaneously.
Value: 0 – 1

Repeat:
Here you influence the intensity of the flashing points.
The higher the value which you set here the more often the respective points will be repeated.
Value: 1 - 255

Note: Although you can define values higher than 255, the Sipro only supports values up to 255.

Displayed command line in the film: `#Sparkle`
Position in the program: Inside or outside the loop
Keyboard shortcut: #s

3.15.17 SetRelais

With `SetRelais` you can switch up to 8 (Sipro I) or 16 (Sipro II) outputs on or off digitally.

Channel: Channel
Underneath you see the 32 outputs belonging to this channel.

Note: The Sipro only supports the first 8/16 outputs of channel 1!

The default setting is *All Unchanged.* **!!!**
That means all relays you set in the previous entry are unchanged, they are taken over from entry to entry even though they are not noticeably turned on.

All OFF All OFF
When you click on All OFF all relays will be turned off.

All ON All ON
When you click on All ON all relays will be turned on.

All Unchanged All Unchanged
Sets all analogs back to their original states.

Get actual state

Get actual state
This is similar to *GetActPos* for `InitPosition`.
However, with the `SetRelais` command it only works retroactively within one entry.

Displayed command line in the film: `SetRelais` and *Channel* number
Position in the program: Outside the time loop, exception: Chaser
Keyboard shortcut: sr

→ Chapter ChaseDelay ChaseSection

3.15.18 SetAnalogs

With SetAnalogs you can control analog outputs.

The Sipro supports the following Analogs (1-4):
Analog 1 = Logoshutter
Analog 2 = Beamshutter
Analog 3 = Motor 1
Analog 4 = Motor 2

Channel: Channel

Note: The Sipro only supports Analog outputs 1-4 of channel 1!

The default setting is *All Unchanged*.
That means all analogs you set in the previous entry are unchanged, they are taken over from entry to entry even though they are not noticeably turned on.

All 0.00 All 0.00

When you click on All 0 all analogs will be turned off.

All 1.00 All 1.00

When you click on All 1 all analogs will be turned on undimmed.

All Unchanged All Unchanged

Sets all analogs back to their original states.

Get actual state Get actual state

This is similar to *GetActPos* for `InitPosition`.
However, with the `SetAnalogs` command it only works retroactively within one entry.

Displayed command line in the film: `SetAnalogs` and Channel number
Position in the program: Outside the time loop, exception: Chaser
Keyboard shortcut: sa

→ Chapter ChaseDelay ChaseSection

3.15.19 SetBeamTimed

With this command you can position individual beams on mirrors or any other laser effect.

You can color the beams by setting the command `#SetC0lor` before `SetBeamTimed`. → `#SetC0lor`

Static beams can cause serious eye damages up to blindness. Therefore never point beams at areas where people may be. **!!!!**

While in Edit Mode, the beam test brightness from the beamtable will be used (default is 5 %).
You can change this setting in the beamtable window. **!!!!**

Adjust beam: Adjust beam
The Sipro only supports the value *BeamAdj*.

Time in seconds: Time in seconds
The default setting is 0.0016.
With this setting the human eye perceives the beam as a standing effect even if you are displaying more than one simultaneously.

Position X: Position X
Here you define the horizontal position of the beam along the X-axis.

Position Y: Position Y
Here you define the vertical position of the beam along the Y-axis.

X & Y: X & Y
Here you can set the horizontal and vertical position of the beam simultaneously with the mouse poti mode.
If you click on the field and enter a number, this value will automatically be entered for Position X as well as Y.

Beam number: Beam number
If you want to use a beam from a Beam table click on this field.
A new window will open with the name Beam table for DSP.
There you click on the desired beam and it will automatically be assigned.
When you have entered a beam here you cannot enter anything for Position X and Y. Anything you may have entered before will automatically be set back to 0. If you enter anything afterwards the selected beam will be deleted and replaced by the newly entered position. **!!!**

Note:
If you use beams from the beamtable, the positions will be derived and used as a fixed value on the Sipro!

Displayed command line in the film: `SetBeamTimed` and Time in seconds
Position in the program: Within the time loop, if existent
Keyboard shortcut: sbt

3.15.20 ChaseDelay

This command allows you to switch different relays and/or analogs in regular intervals, i.e. to program effect cycles.

→ `ChaseSection`

The different relay blocks are separated with the command `ChaseSection`. In general each `ChaseSection` command is equivalent to a Wait command whose waiting time can be set with the command `ChaseDelay`.
You could just as well enter:

```
 SetRelais
 Wait
 SetRelais
 Wait ...
```

However, if you then want to change the speed you have to change the time for every Wait, with ChaseDelay you have to change it only once.

!!!

Scanner output is not possible at the same time with this command.

Delay in Seconds

Delay in seconds:
Here you enter the time delay after which the effects are supposed to be switched. With 1.0 the effects will be switched every second, with 0.5 every half second and so on.

Speed multiplier

Speed multiplier:
The value you entered for the previous parameter is divided by the value entered here.

The default setting is 1.0, it does not influence the speed defined before. If, however, you enter 2.0 the speed will be doubled.

Example:

```
  Delay in Seconds:  1.0000
  Speed multiplier:  2.0000
```

1 : 2 = 0.5 The change of effects will occur every half second.

Displayed command line in the film: `ChaseDelay` and `Delay` in seconds
Position in the program: Outside the time loop
Keyboard shortcut: cd

3.15.21 ChaseSection

ChaseSection is an internal Wait command.
It is used together with ChaseDelay to separate the individual effect blocks.

→ ChaseDelay

After each effect block you have to set a ChaseSection command which will then wait before switching to the next effect for exactly the period of time which you defined for ChaseDelay.

A scanner output is not possible during this waiting period.

A chaser entry would then look like this:

```
Entry
ChaseDelay      Delay in Seconds: 0.3000
                Speed multiplier:   1.0000
TimeLoop
SetRelais
SetAnalog
ChaseSection    holds the block set before this command for
                0.3 seconds before switching to the next
                block
SetRelais
SetAnalog
ChaseSection    holds the block set before this command for
                0.3 seconds before switching to the next
                block

SetRelais
SetAnalog
ChaseSection    holds the block set before this command for
                0.3 seconds
LoopEnd
```

Displayed command line in the film: ChaseSection
Position in the program: After a SetRelais and/or SetAnalogs block
Keyboard shortcut: cs

3.15.22 #DisplayAnimation (Sipro II only)

The `#DisplayAnimation` command causes the display of an animation.

Position #
This value defines to which #InitPosition the picture is assigned.

Animation:
Select an animation by clicking into the Animation field.
The file window will open and show all animations that are in the memory.

Stream from:
Streaming is not supported by the Sipro!

Frames:
The amount of frames used in the animation.

Mode:
Two modes are available: "Loopcnt" and "Repeat"
When "LoopCnt" is selected the animation will be shown once with the duration that is defined by the loop.
When "Repeat" is selected the animation will be repeated again and again as long as the loop is running.

Repeat:
This value will decrease the speed of the animation by increasing the display of each single frame.
Value 0: speed as fast as possible
Value 255: as slow as possible
Value Range: 0 - 255

Displayed command line in the film:
`#DisplayAnimation`
Position in the program: Within the time loop
Keyboard shortcut: #da

3.15.23 #SetLogoColor (Sipro II only)

This command offers the possibility to fade a picture from one color to another or to fade it in or out.
For each of the 3 color lines you can define a start and an end position.
Also a static mixture of colors is done with this command.

Overwrite Colors:
If the checkbox is enabled the colors of the picture/animation will be overwritten by the values set in the following parameters. If it is disabled, the picture will be displayed with its default colors.
This is useful if you want to show one picture with its default and another with modified colors.

Mix:
Define a value between 0 - 1 or choose "LoopCnt" to get a dynamic fade from start to end in the loop.

Start Red / Green / Blue:
Define a starting value for each color.
Value: 0 - 1

End Red / Green / Blue:
Define an end value for each color.
Value: 0 - 1

Displayed command line in the film: `#SetLogoColor`
Position in the program: Inside the loop
Keyboard shortcut: #slc

3.15.24 #SetBeamColor (Sipro II only)

This command defines the color of the optical bench effects (special hardware required). It does not affect the display of pictures and animations.

You can choose between these colors:
White - Red - Yellow - Green - Cyan - Blue – Magenta or Black.

Note:
This command does not affect the colors of the scanning output.

Displayed command line in the film: `#SetBeamColor`
Position in the program: Outside or inside the loop
Keyboard shortcut: #sbc

3.15.25 #SetModes (Sipro II only)

This command is used to enable an external input (e.g. vidal) and to double the size of the displayed pictures.

External input:
values: Unchanged, ON, OFF

Double display size:
Doubles the size of a picture.

Note: This command does not change the size of your projection area, it just doubles the size of the picture and therefore reduces the possible range of movement.

Displayed command line in the film: `#SetModes`
Position in the program: Outside or inside the loop
Keyboard shortcut: #sm

3.15.26 Track (Sipro II only)

With the command `Track` you can divide each loop into individual segments. These segments can be shown independent of each other and with different running times within the loop. All commands following a track are time controlled by the track up to the next track or `LoopEnd`.

Relative start:
Defines the relative start position of the track.

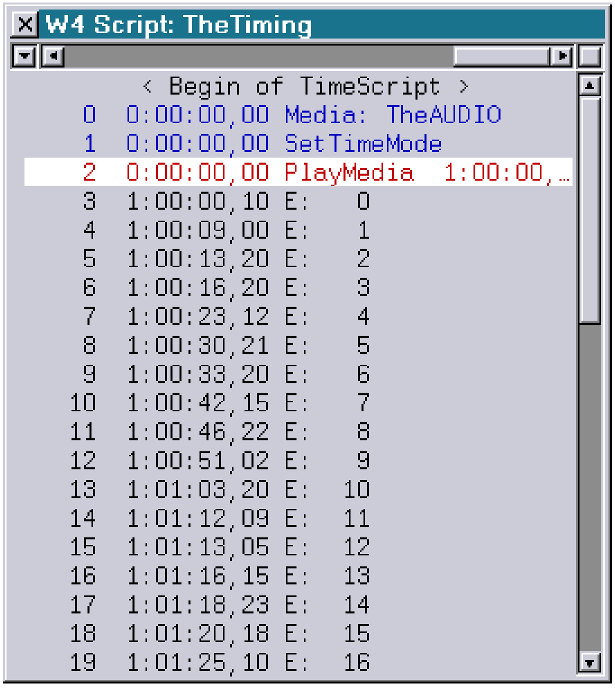
Relative end:
Defines the relative end position of the track.

Displayed command line in the film: `Track`
Position in the program: Inside the loop
Keyboard shortcut: T

4. The TimeScript

The `TimeScript` is used to put the Entries of the film into relation to the time of a timecode.

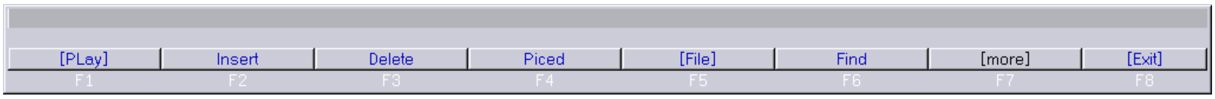
This timecode can be an external or an internal one.
External means, it is the timing information which comes from another device, a video-/audioplayer, a MIDI-device, a PC running LGServer or another Lasergraph DSP.
An internal timecode is a self generated timecode within the Lasergraph itself.



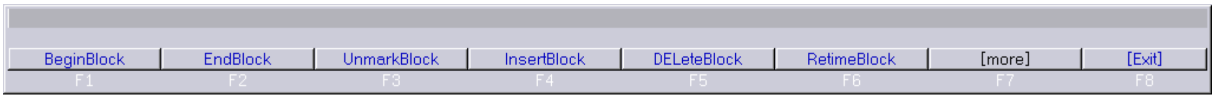
```
      < Begin of TimeScript >
 0  0:00:00,00 Media: TheAUDIO
 1  0:00:00,00 SetTimeMode
 2  0:00:00,00 PlayMedia  1:00:00,…
 3  1:00:00,10 E:   0
 4  1:00:09,00 E:   1
 5  1:00:13,20 E:   2
 6  1:00:16,20 E:   3
 7  1:00:23,12 E:   4
 8  1:00:30,21 E:   5
 9  1:00:33,20 E:   6
10  1:00:42,15 E:   7
11  1:00:46,22 E:   8
12  1:00:51,02 E:   9
13  1:01:03,20 E:  10
14  1:01:12,09 E:  11
15  1:01:13,05 E:  12
16  1:01:16,15 E:  13
17  1:01:18,23 E:  14
18  1:01:20,18 E:  15
19  1:01:25,10 E:  16
```

As you can see, the timescript is based on the same command line syntax as the film editor.
In the first column you see the line number, behind it is the time information and then the command itself.

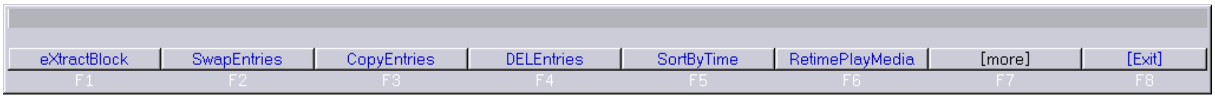
4.1 Commands in the Menu



`[PLay]`	starts the Script	`Function key F1`
`Insert`	inserts a new command at the current position	`Function key F2`
`Delete`	deletes the current command in the script	`Function key F3`
`Piced`	changes to the Picture Editor with 4 options:	`Function key F4`
`[lastedit]`	→ keeps the last picture in the Piced	
`[byfilm]`	→ takes the picture and its position from the current film situation	
`<catalog>`	→ let's you choose a picture	
`[browse]`	→ opens the browser to choose a picture	
`[File]`	opens the File window	`Function key F5`
`Find`	search the Timescript for a certain command	`Function key F6`
`[more]`	jumps to the next menu level	`Function key F7`
`[Exit]`	changes into the main menu (Root)	`Function key F8`



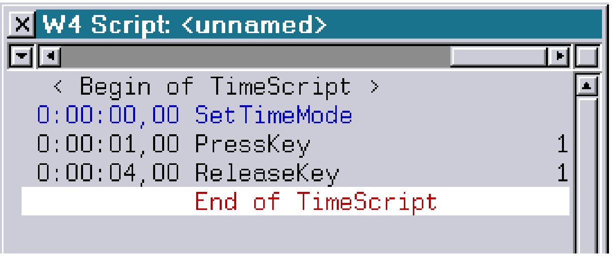
`BeginBlock`	marks the beginning of the selection (block)	`Function key F1`
`EndBlock`	marks the ending of the selection (block)	`Function key F2`
`UnmarkBlock`	umarks the selection (block)	`Function key F3`
`InsertBlock`	inserts the selection (block) before the current position	`Function key F4`
`DELeteBlock`	deletes the selection (block).	`Function key F5`
`RetimeBlock`	changes the start time of the selected block	`Function key F6`
`[more]`	jumps to the next menu level	`Function key F7`
`[Exit]`	changes into the main menu (Root)	`Function key F8`



eXtractBlock Function key F1

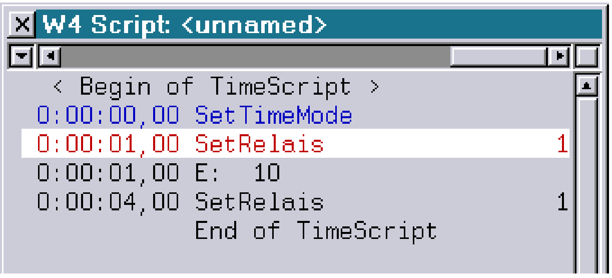
converts all PressKey and ReleaseKey commands in the selected block according to the actions assigned in the Keyconfig.

E. g. if you recorded a PressKey and a ReleaseKey command where Entry 10 and DMX Analog Output 4 were defined



```
      < Begin of TimeScript >
0:00:00,00 SetTimeMode
0:00:01,00 PressKey               1
0:00:04,00 ReleaseKey             1
           End of TimeScript
```

it will extract the commands into



```
      < Begin of TimeScript >
0:00:00,00 SetTimeMode
0:00:01,00 SetRelais              1
0:00:01,00 E:   10
0:00:04,00 SetRelais              1
           End of TimeScript
```

SwapEntries swaps Entries between two films Function key F2

CopyEntries copies the Entries of one film to another Function key F3

DELEntries deletes Entries of the selected film in the script Function key F4

SortByTime sorts all lines into the right order (timing) Function key F5

RetimePlayMedia Function key F6

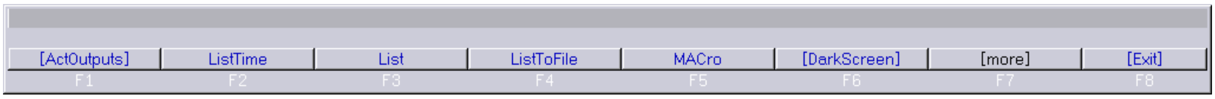
this command reorganizes a show which is allocated to a certain Media track to another track.

E. g. a show programmed to track 12 of a CD will be reorganized to track 3 with the command `RetimePlayMedia 3`.

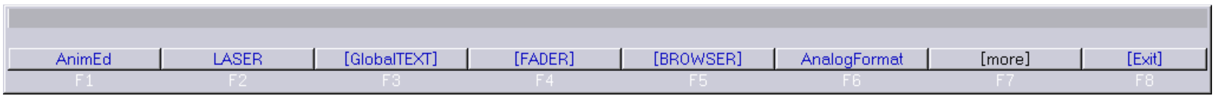
This command changes all 'PlayMedia' commands in the 'TimeScript'.

[more] jumps to the next menu level Function key F7

[Exit] changes into the main menu (Root) Function key F8



[ActOutputs]	informs you about the programmed DMX outputs in the script	Function key F1
ListTime	jumps to a specific time within the time script	Function key F2
List	jumps to specific line, Top or Bottom	Function key F3
ListToFile	saves the script command structure to a textfile	Function key F4
MACro	recording or playing back a Macro.	Function key F5
[DarkScreen]	fades the screen to grey	Function key F6
[more]	jumps to the next menu level	Function key F7
[Exit]	changes into the main menu (Root)	Function key F8



AnimEd	opens up the AnimationEditor	Function key F1
LASER	opens up the Laser Control window	Function key F2
[GlobalTEXT]	opens up the Global Text Table.	Function key F3
[FADER]	opens the FADER emulation.	Function key F4
[BROWSER]	opens the Browser window.	Function key F5
AnalogFormat	switches the numerical format of analog values in the Keyconfig and in the Trickfilm- and Timescript commands SetAnalogs between "Fractional", "Decimal", "Hexadecimal" and "Percent".	Function key F6
[more]	jumps to the first menu level	Function key F7
[Exit]	changes into the main menu (Root)	Function key F8

4.2 The TimeScript Commands

Insert

In order to set commands you click on Insert in the *menu bar* at the bottom or press the function key F2.

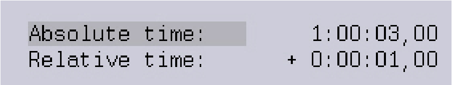
After choosing INSERT you can directly type in a time.

syntax→ hours:minutes:seconds,frames

or choose from 3 predefined options:
F1 <abstime> for absolute time
F2 <reltime> for relative time
F3 for time 0:00:00,00

<abstime> will insert the time of the last command line
<reltime> will insert the time of the last command line with the possibility to add a time.
e.g.
Insert 1:00:16,20+0:01:00,00
will result in 1:01:16,20

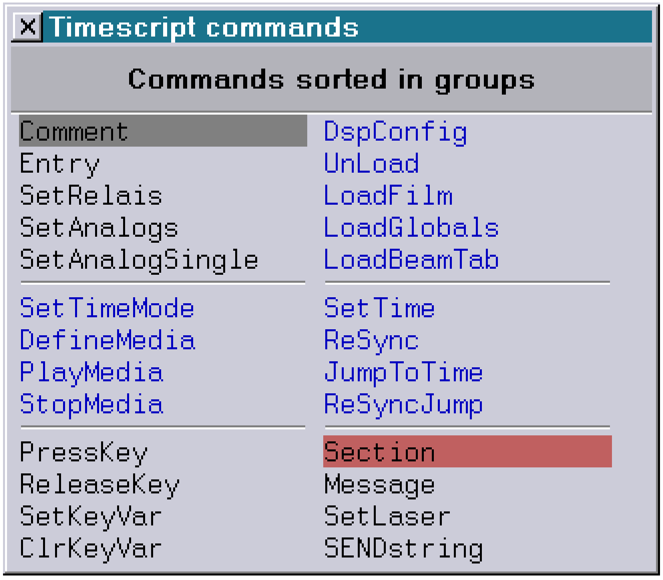
In every command you will find the time information on top of the command options.



```
Absolute time:        1:00:03,00
Relative time:      + 0:00:01,00
```

The absolute time is the time where the command will be executed. The relative time informs you of the time change since the last command.

The following commands are available after defining the time:

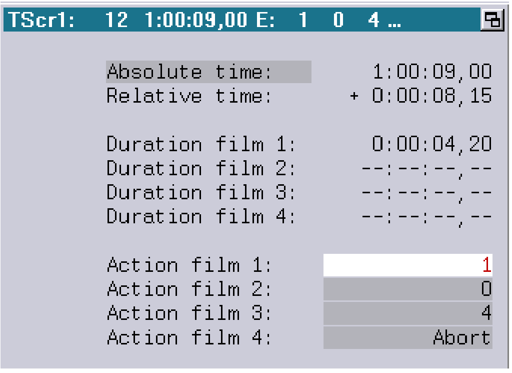


4.2.1 Entry

Entry

The command `Entry` links the time to an Entry of a film.

You can define any existing Entry number or an Abort to stop the film.
For each of the 4 films.



```
TScr1:   12  1:00:09,00 E:  1   0   4 ...

Absolute time:          1:00:09,00
Relative time:        + 0:00:08,15

Duration film 1:        0:00:04,20
Duration film 2:       --:--:--,--
Duration film 3:       --:--:--,--
Duration film 4:       --:--:--,--

Action film 1:                   1
Action film 2:                   0
Action film 3:                   4
Action film 4:               Abort
```

Keyboard shortcut: `e`

4.2.2 Comment

Comment

The command `Comment` is a useful orientation guide in the script, as it is in the film!

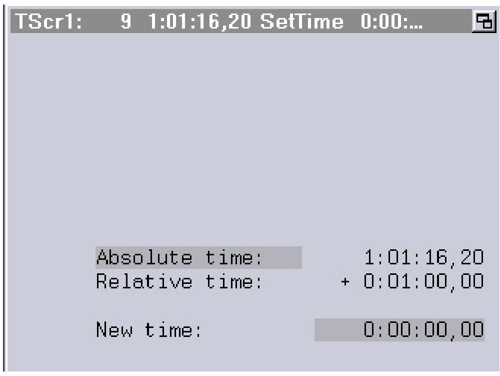
It has no influence on the script and you can insert it as often and wherever you like.

Keyboard shortcut: `c`

4.2.3 SetTime

SetTime

The command SetTime jumps to the time defined in "New time".
Works only in Internal Mode !



TScr1: 9 1:01:16,20 SetTime 0:00:...

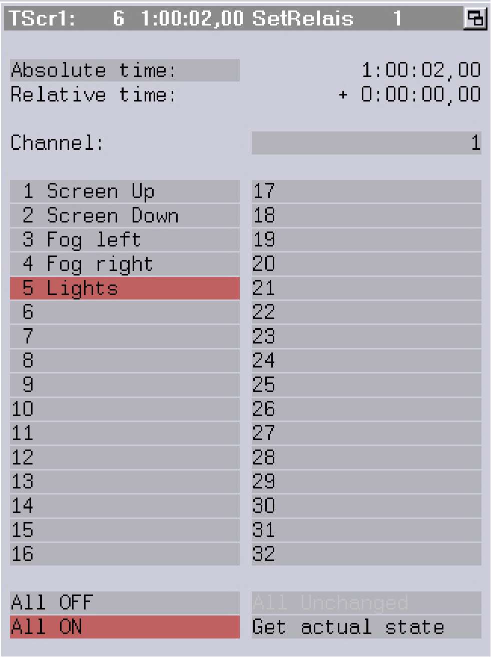
```
Absolute time:        1:01:16,20
Relative time:      + 0:01:00,00

New time:             0:00:00,00
```

Keyboard shortcut: st

4.2.4 SetRelais

SetRelais

The command SetRelais enables or disables the DMX outputs (only ON, OFF or Unchanged), similar to the film command SetRelais.



TScr1: 6 1:00:02,00 SetRelais 1

```
Absolute time:        1:00:02,00
Relative time:      + 0:00:00,00

Channel:                       1
```

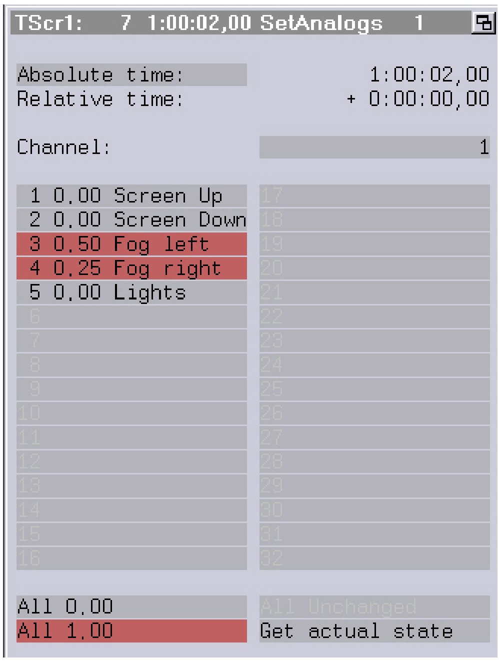
1 Screen Up	17
2 Screen Down	18
3 Fog left	19
4 Fog right	20
5 Lights	21
6	22
7	23
8	24
9	25
10	26
11	27
12	28
13	29
14	30
15	31
16	32
All OFF	All Unchanged
All ON	Get actual state

Keyboard shortcut: sr

4.2.5 SetAnalogs

SetAnalogs

The command SetAnalogs gives you analog control of the DMX Outputs, similar to the film command SetAnalogs.



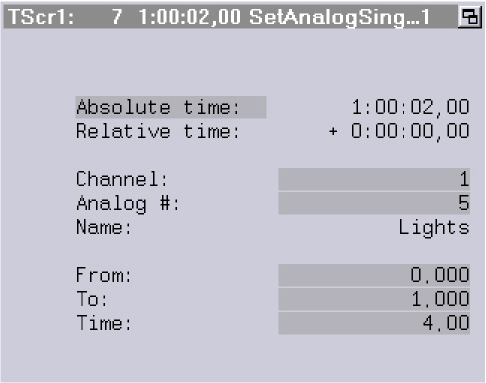
```
TScr1:    7  1:00:02,00 SetAnalogs    1

Absolute time:                 1:00:02,00
Relative time:               + 0:00:00,00

Channel:                                1

 1  0.00 Screen Up          17
 2  0.00 Screen Down        18
 3  0.50 Fog left           19
 4  0.25 Fog right          20
 5  0.00 Lights             21
 6                          22
 7                          23
 8                          24
 9                          25
10                          26
11                          27
12                          28
13                          29
14                          30
15                          31
16                          32

All 0.00                    All Unchanged
All 1.00                    Get actual state
```

Keyboard shortcut: sa

4.2.6 SetAnalogSingle

SetAnalogSingle

The command SetAnalogSingle gives you analog control over a defined time (0-10.2 seconds) of one DMX output, similar to the film command SetAnalogSingle (except for use of a variable).



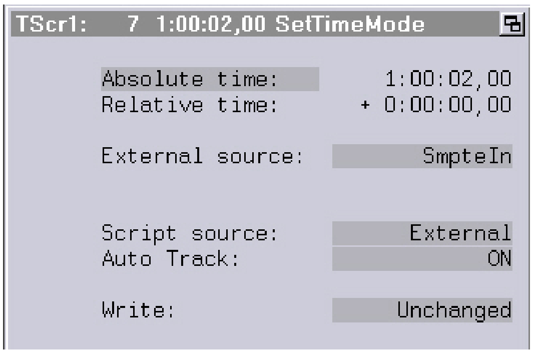
```
TScr1:    7  1:00:02,00 SetAnalogSing…1

Absolute time:             1:00:02,00
Relative time:           + 0:00:00,00

Channel:                            1
Analog #:                           5
Name:                          Lights

From:                           0.000
To:                             1.000
Time:                            4.00
```

Keyboard shortcut: sas

4.2.7 SetTimeMode

The command `SetTimeMode` is used to define the handling of the timecode.

`SetTimeMode`


TScr1: 7 1:00:02,00 SetTimeMode

Absolute time: 1:00:02,00
Relative time: + 0:00:00,00

External source: SmpteIn

Script source: External
Auto Track: ON

Write: Unchanged

External source: Where does the timecode come from?

Unchanged:	The state defined before stays unchanged.
SmpteIn:	Smpte timecode (LTC, Logitudinal Timecode)
Midi:	Midi timecode (MTC)
Media:	Timecode (LGServer)
Net:	Timecode that is send by an other DSP over the network with the option “Write”
NetScript:	Timecode from a DSP defined in the Net I/O
Vitc:	Vertical interval Timecode
Time:	Current time of day

Script source: Select the timecode source the script shall use. Options are Unchanged, Internal and External.
If External is selected and the external source does not deliver a timecode, the script waits until the code is detected. In this case, if AutoTrack is on, it becomes active after the first appearance of the external timecode.

Auto Track: Options are Unchanged, ON and OFF.
If AutoTrack is on, the external timecode is used as soon as it is available. If the external timecode becomes unavailable, the internal timecode is used until it is available again.

Write: Options are Unchanged, ON and OFF.
If enabled, the timecode used by the script is delivered to the network.

NOTE: Make sure that in a network environment only one DSP is set write, otherwise you will flood the network with traffic !!

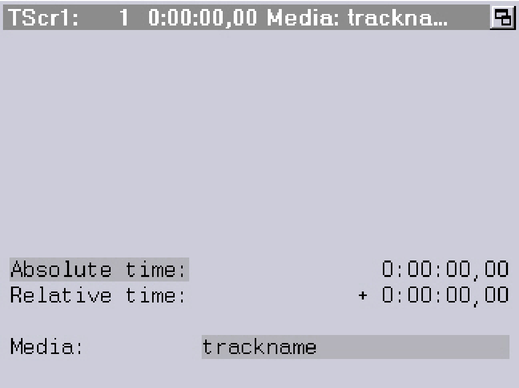
! ! ! !

Keyboard shortcut: `stm`

4.2.8 DefineMedia

DefineMedia

The command DefineMedia defines the source for the LGServer.



TScr1: 1 0:00:00,00 Media: trackna...

```
Absolute time:                0:00:00,00
Relative time:              + 0:00:00,00

Media:          trackname
```

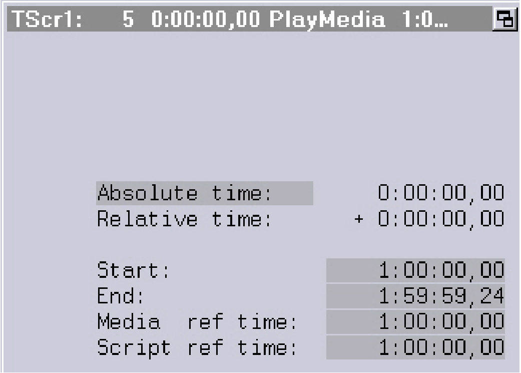
Media: Define the name of the audio-/video file on the LGServer.

Keyboard shortcut: dm

4.2.9 PlayMedia

PlayMedia

The command PlayMedia starts the media player on the PC (LGServer) or the CD-Rom Drive (on old workstations only!).



```
TScr1:    5  0:00:00,00 PlayMedia  1:0…

Absolute time:           0:00:00,00
Relative time:         + 0:00:00,00

Start:                   1:00:00,00
End:                     1:59:59,24
Media  ref time:         1:00:00,00
Script ref time:         1:00:00,00
```

Start: starting point of the media file

End: endpoint of the media file

Media ref time: This time in the media file...

script ref time: ...is equivalent to this time in the script.

The ref-times are useful if the timecode in the script differs from the timecode of the media file. Let's say you have a show with music from tape; you decide to write the show with the music onto CD. The problem is, after the music is written to CD, the timecode from the CD is very likely to differ from the timecode of the tape. You could re-time the show to fit the new timecode, but this is not always the way you want it to be done. With the ref-times you can leave your script as it is and define a reference-point to synchronize script and CD. This point can be anywhere in the music track; that way you can select an easy-to-find part in the music, type the timecode into the field CD ref time, then get the timecode of the script at this point of the show (using GetTime) and put it into the script ref time field. That's it.

Keyboard shortcut: pm

4.2.10 StopMedia

StopMedia

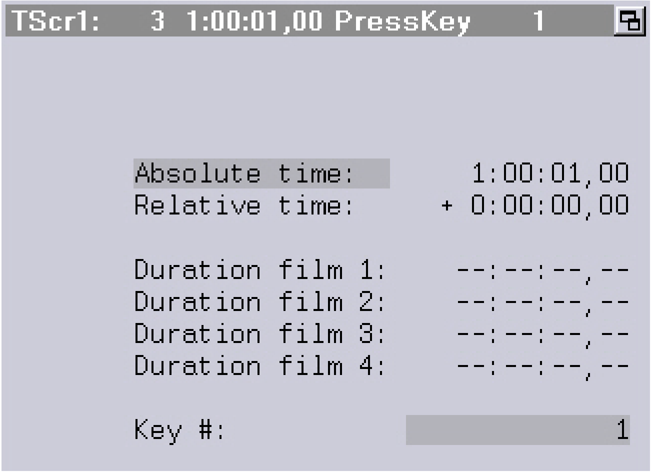
The command StopMedia is used to stop the playback of a media file.

Keyboard shortcut: sm

4.2.11 PressKey

PressKey

The command `PressKey` is used to simulate and record the pressing of a Midi key on the Midi keyboard.



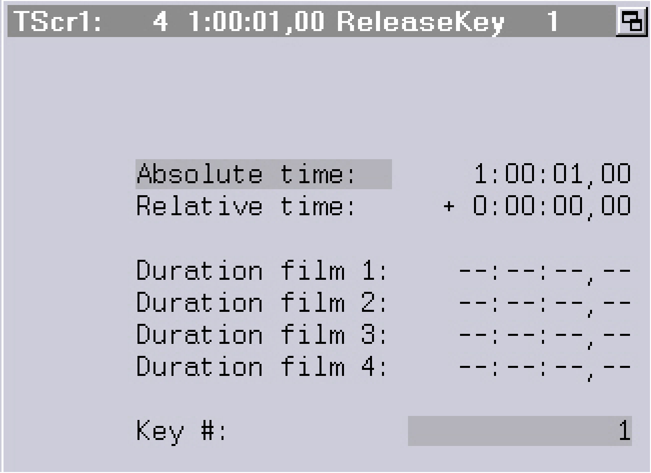
```
TScr1:     3  1:00:01,00 PressKey      1

Absolute time:          1:00:01,00
Relative time:        + 0:00:00,00

Duration film 1:       --:--:--,--
Duration film 2:       --:--:--,--
Duration film 3:       --:--:--,--
Duration film 4:       --:--:--,--

Key #:                           1
```

Keyboard shortcut: `pk`

4.2.12 ReleaseKey

ReleaseKey

The command `ReleaseKey` is the opposite of `PressKey`, it releases the defined Midi key.



```
TScr1:     4  1:00:01,00 ReleaseKey    1

Absolute time:          1:00:01,00
Relative time:        + 0:00:00,00

Duration film 1:       --:--:--,--
Duration film 2:       --:--:--,--
Duration film 3:       --:--:--,--
Duration film 4:       --:--:--,--

Key #:                           1
```

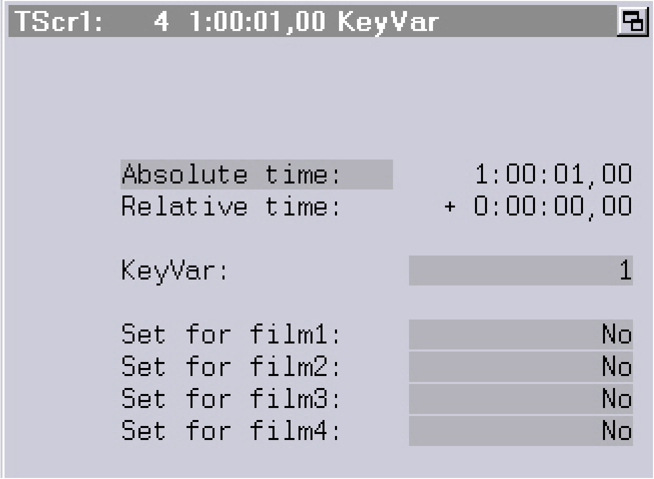
Keyboard shortcut: `rk`

4.2.13 SetKeyVar

The command `SetKeyVar` is used to simulate the pressing of a key with an assigned KeyVariable. The number of the variable can be 0 to 63.

`SetKeyVar`

The following example command sets the key variable # 1 to 1.



```
TScr1:    4  1:00:01,00 KeyVar

Absolute time:          1:00:01,00
Relative time:        + 0:00:00,00

KeyVar:                          1

Set for film1:                  No
Set for film2:                  No
Set for film3:                  No
Set for film4:                  No
```

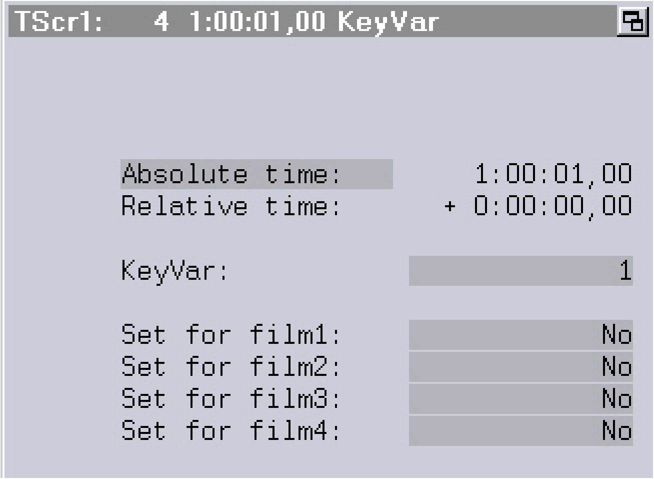
Keyboard shortcut: `skv`

4.2.14 ClrKeyVar

The command `ClrKeyVar` is the opposite of `SetKeyVar`, it releases the defined KeyVar.

`ClrKeyVar`

The following example command sets the key variable # 1 to 0.



```
TScr1:    4  1:00:01,00 KeyVar

Absolute time:          1:00:01,00
Relative time:        + 0:00:00,00

KeyVar:                          1

Set for film1:                  No
Set for film2:                  No
Set for film3:                  No
Set for film4:                  No
```

Keyboard shortcut: `ckv`

4.2.15 SetParallel

The command `SetParallel` is used to control old devices. It should not be used anymore.

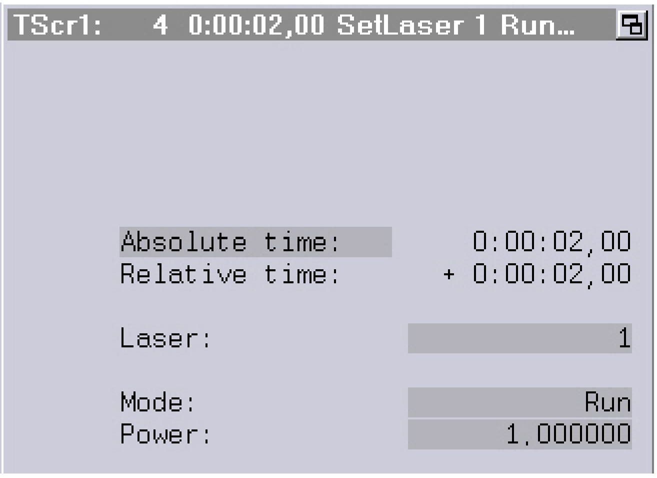
`SetParallel`

Keyboard shortcut: `sp`

4.2.16 SetLaser

The command `SetLaser` is used to control Laser that are connected via the serial interface.
Note that the laser type and the serial interface have to be defined in the "SETUP".

`SetLaser`



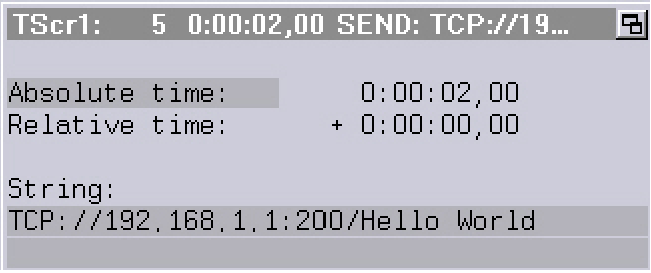
```
Absolute time:        0:00:02,00
Relative time:      + 0:00:02,00

Laser:                         1

Mode:                        Run
Power:                  1,000000
```

Keyboard shortcut: `sl`

4.2.17 SENDstring

This command sends ASCII-code or hexadecimal values through the serial outputs or via the TCP/IP network.
For more details see Chapter “SerCmd”.

`SENDstring`



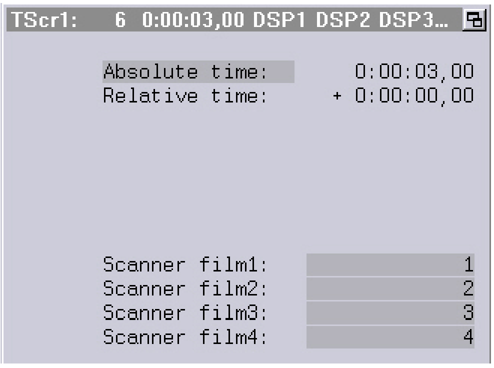
```
Absolute time:        0:00:02,00
Relative time:      + 0:00:00,00

String:
TCP://192.168.1.1:200/Hello World
```

Keyboard shortcut: `send`

4.2.18 DspConfig

DspConfig

The command DspConfig defines the assignment of films to the DSP-Output Cards.
Note that the number of Channels available depends on the hardware platform.



```
TScr1:    6  0:00:03,00 DSP1 DSP2 DSP3...

Absolute time:        0:00:03,00
Relative time:      + 0:00:00,00

Scanner film1:                 1
Scanner film2:                 2
Scanner film3:                 3
Scanner film4:                 4
```

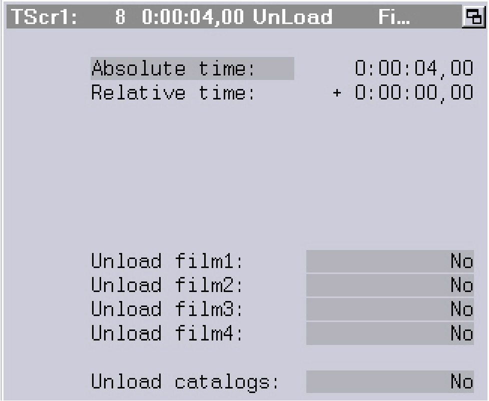
Note: This command will be ignored in a Master/Slave setup on a Compact. The workstation will still be executing this command in Master/Slave mode.

Keyboard shortcut: `dc`

4.2.19 UnLoad

UnLoad

With the command UnLoad you can unload specified films and the catalogs used in these films.



```
TScr1:    8  0:00:04,00 UnLoad    Fi...

Absolute time:        0:00:04,00
Relative time:      + 0:00:00,00

Unload film1:                 No
Unload film2:                 No
Unload film3:                 No
Unload film4:                 No

Unload catalogs:              No
```

Note: When “Unload catalogs” is selected all animations and catlogs that are not used by any film will be unloaded after the selected films have been unloaded.
Tip: While editing the Script you can disable all Load and Unload commands with the option "Suppress Load by script".

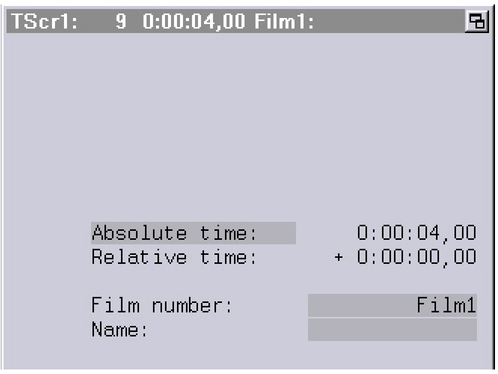
Keyboard shortcut: `ul`

4.2.20 LoadFilm

`LoadFilm`

With the command `LoadFilm` you can load a new film while the script is running.

Note that the laser output is stopped while loading.



TScr1: 9 0:00:04,00 Film1:

```
Absolute time:       0:00:04,00
Relative time:     + 0:00:00,00

Film number:              Film1
Name:
```

Film number: defines the Film number (1-4) for the film

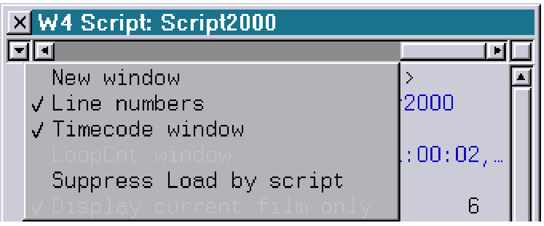
Name: Defines the name of the film

Notes:

- The file has to be in the current directory.
- Pictures and animations used by the film are also loaded when they are available in the current directory.

TIPS:

- Save the film in the same directory where the BOOT is saved.
- While editing the Script you can disable all Load and Unload commands with the option "Suppress Load by script".



W4 Script: Script2000

```
  New window
✓ Line numbers
✓ Timecode window
  LoopCnt window
  Suppress Load by script
✓ Display current film only
```

Keyboard shortcut: `lf`

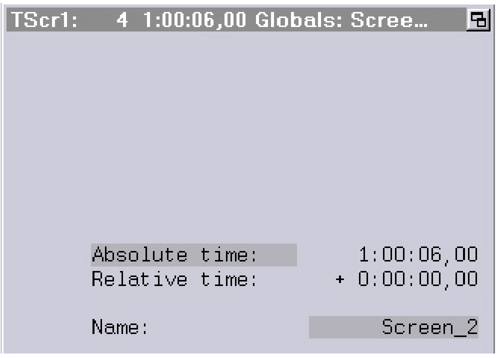
4.2.21 LoadGlobals

With the command `LoadGlobals` you can load new Global Settings (Screenparameters etc.) while the script is running.

`LoadGlobals`

This file has to be located inside the "current directory".
Note that the laser output is stopped while loading.

Tip: While editing the Script you can disable all Load and Unload commands with the option "Suppress Load by script".



TScr1: 4 1:00:06,00 Globals: Scree...

```
Absolute time:        1:00:06,00
Relative time:      + 0:00:00,00

Name:                   Screen_2
```

Keyboard shortcut: `lg`

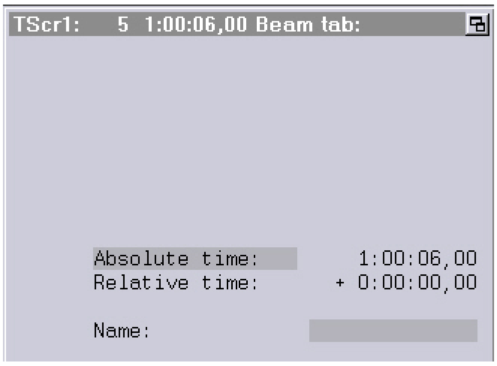
4.2.22 LoadBeamTab

With the command `LoadBeamTab` you can load a new Beamtable while the script is running.

`LoadBeamTab`

This file has to be located inside the "current directory".
Note that the laser output is stopped while loading.

Tip: While editing the Script you can disable all Load and Unload commands with the option "Suppress Load by script".



TScr1: 5 1:00:06,00 Beam tab:

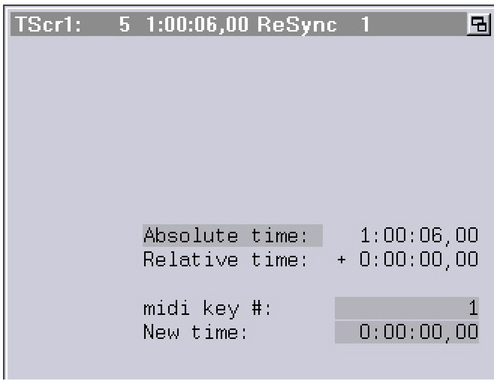
```
Absolute time:        1:00:06,00
Relative time:      + 0:00:00,00

Name:
```

Keyboard shortcut: `lbt`

4.2.23 ReSync

ReSync

The command ReSync is used if a show is displayed to music which is played live. This poses the problem that there is no external timecode to synchronize to. There are only scenes which have to start at certain measures of the music, but the time of these scenes is determined by the speed of the conductor. To have the show synchronous to the music, use internal timecode to run the show. Insert the command "ReSync" before every scene which is supposed to be synchronous to the music. Define the number of a Midi key in the field 'Midi key #:' which is to be used to synchronize the show with the conductor. Set the field 'New time:' to the time of the scene to be started. After this command is executed, the internal timecode will be set to the specified time whenever the specified Midi key is hit. This means, by pressing the specified Midi key at the moment the conductor gives the signal, the internal time will be set to the time of the scene for this measure, thus resynchronizing the show with the music.



```
TScr1:    5  1:00:06,00 ReSync   1

Absolute time:     1:00:06,00
Relative time:   + 0:00:00,00

midi key #:                 1
New time:          0:00:00,00
```

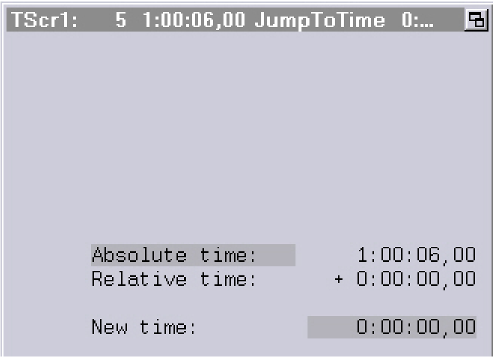
Keyboard shortcut: rs

4.2.24 JumpToTime

JumpToTime

The command JumpToTime sets the time code to the specified time. Entries between the current time and the specified time will not be executed.

In contrast to the TimeScript command "SetTime", with "JumpToTime" entries between the current time and the new time will be skipped.



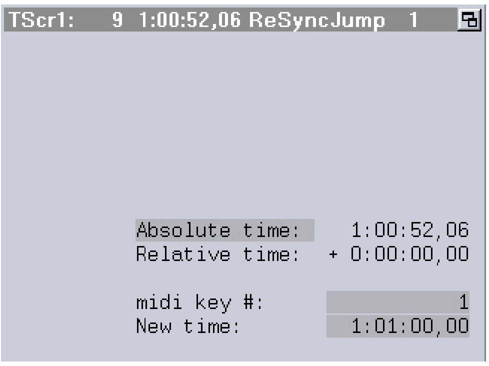
```
TScr1:    5  1:00:06,00 JumpToTime  0:...

Absolute time:          1:00:06,00
Relative time:        + 0:00:00,00

New time:               0:00:00,00
```

Keyboard shortcut: jtt

4.2.25 ReSyncJump

ReSyncJump

The command ReSyncJump makes a running TimeScript jump to the time "New time:" specified in the command window when a certain MIDI key is pressed.



```
TScr1:    9  1:00:52,06 ReSyncJump   1

Absolute time:     1:00:52,06
Relative time:   + 0:00:00,00

midi key #:                 1
New time:          1:01:00,00
```

Keyboard shortcut: rsj

4.2.26 Section

Section

When starting a timescript from a certain specified time usually the complete timescript is executed up to the time of entry. There are certain applications where this is not wanted, e.g. when controlling DMX devices.

To prevent the execution of the complete script the timescript can be separated into individual sections using the command `Section`. When entering the script at a certain point in time only the section starting at the last command `Section` will be executed.

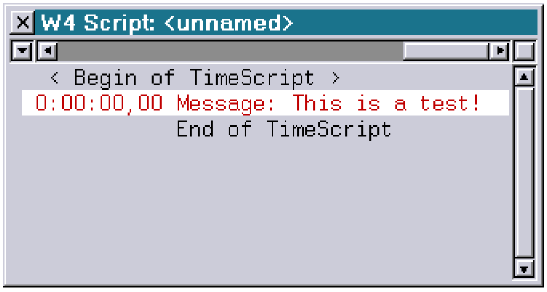
Keyboard shortcut: `s`

4.2.27 Message

Message

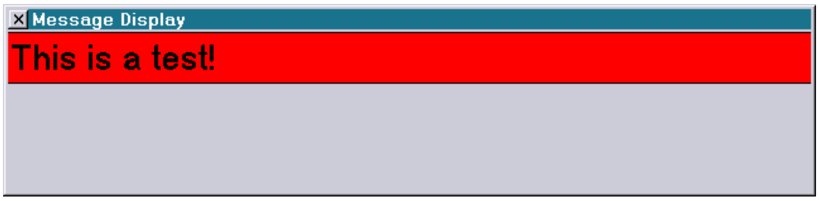
This command can be used to trigger an output in the message window.

The following command



```
< Begin of TimeScript >
0:00:00,00 Message: This is a test!
End of TimeScript
```

will lead to the following output when executed:



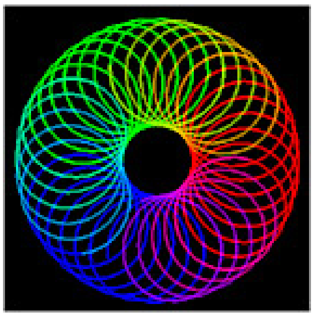
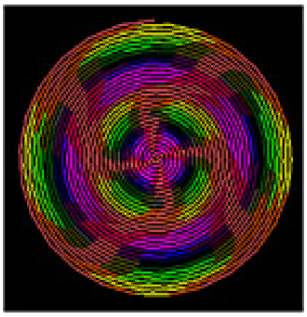
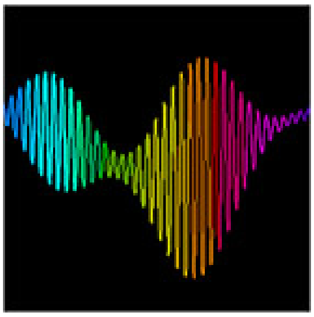
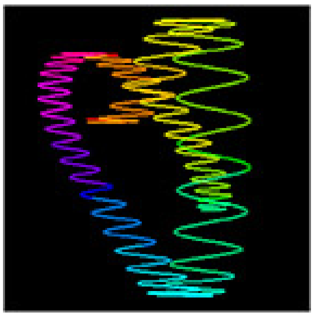
This is a test!

Keyboard shortcut: `m`

Chapter Abstract

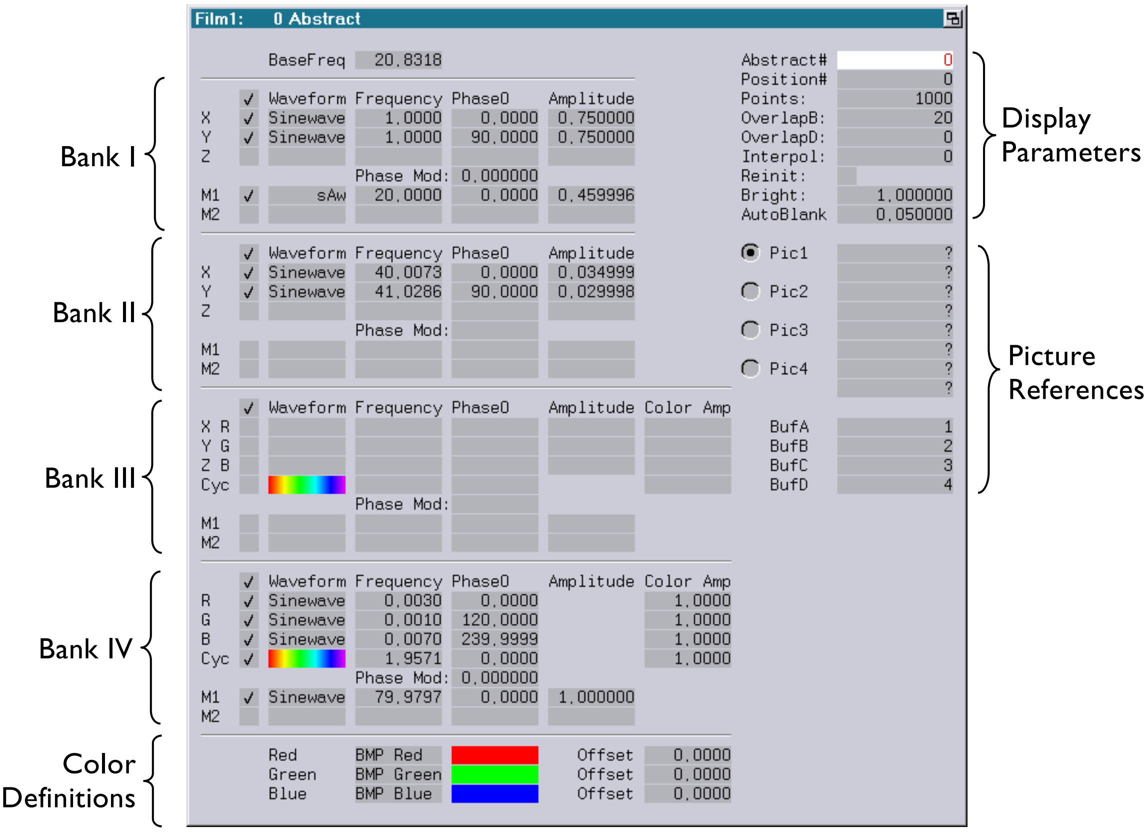
The command Abstract activates the mathematical picture generator.
The pictures created with Abstract are not stored on the computer. They are calculated in real time from the mathematical definitions specified in the parameter window of the Abstract command. It is difficult to find words to describe the resulting pictures. Some look like psychedelic color patterns or they resemble a picture of a colorful loosely wound ball of yarn.

The following pictures show a few examples:

The basis for these picture calculations are oscillations set by the user, defined through specification of the waveform, the frequency, the starting value (Phase0) and the amplitude resp. the color amplitude (Color Amp). The pictures resulting from these multiple superimposed oscillations can only be predicted up to a certain number of active parameters, even by mathematically trained people. Therefore an intuitive use of the provided parameter fields is often the best way to achieve the desired (or even completely different) pictures.

In order to retain a certain overview even with all the intuition, the parameter window is divided into several sections:



```
Film1:    0 Abstract

        BaseFreq  20.8318                                 Abstract#           0
                                                          Position#           0
     ✓ Waveform Frequency Phase0   Amplitude              Points:          1000
X    ✓ Sinewave    1.0000   0.0000 0.750000               OverlapB:          20
Y    ✓ Sinewave    1.0000  90.0000 0.750000               OverlapD:           0
Z                                                         Interpol:           0
                Phase Mod: 0.000000                       Reinit:
M1   ✓      sAw   20.0000   0.0000 0.459996               Bright:      1.000000
M2                                                        AutoBlank    0.050000

     ✓ Waveform Frequency Phase0   Amplitude              ● Pic1              ?
X    ✓ Sinewave   40.0073   0.0000 0.034999                                   ?
Y    ✓ Sinewave   41.0286  90.0000 0.029998               ○ Pic2              ?
Z                                                                             ?
                Phase Mod:                                ○ Pic3              ?
M1                                                                            ?
M2                                                        ○ Pic4              ?
                                                                              ?
     ✓ Waveform Frequency Phase0   Amplitude Color Amp
X R                                                         BufA              1
Y G                                                         BufB              2
Z B                                                         BufC              3
Cyc                                                         BufD              4
                Phase Mod:
M1
M2

     ✓ Waveform Frequency Phase0   Amplitude Color Amp
R    ✓ Sinewave    0.0030   0.0000             1.0000
G    ✓ Sinewave    0.0010 120.0000             1.0000
B    ✓ Sinewave    0.0070 239.9999             1.0000
Cyc  ✓             1.9571   0.0000             1.0000
                Phase Mod: 0.000000
M1   ✓ Sinewave   79.9797   0.0000 1.000000
M2

       Red      BMP Red     🟥        Offset   0.0000
       Green    BMP Green   🟩        Offset   0.0000
       Blue     BMP Blue    🟦        Offset   0.0000
```

The sections labeled Bank 1-4 contain several frequency generators (oscillators), they will therefore hereafter be referred to as oscillator banks 1-4. The pictures created by the Abstract command show the sum of the calculations from the four provided oscillator banks. Banks 1+2 create geometries, bank 3 creates colors and geometries and bank 4 only colors.

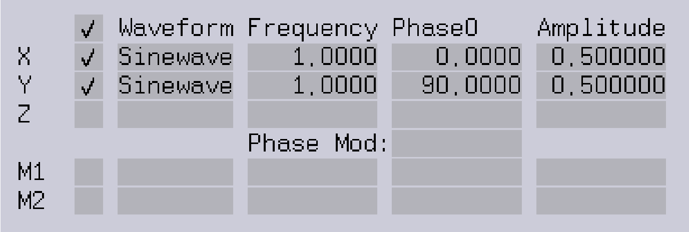
The Display Parameters section contains parameters which influence the display of the calculated pictures. Also, the number of the activated dataset is displayed here.

The Picture References section, Pic1 - Pic4 and BufA - BufD, contains the names and catalogs of graphics which can be used instead of a wave form.

The color definitions refer to an additive three-component color model. The preset RGB color model can be redefined by changing the basic colors.

The Oscillator Banks:

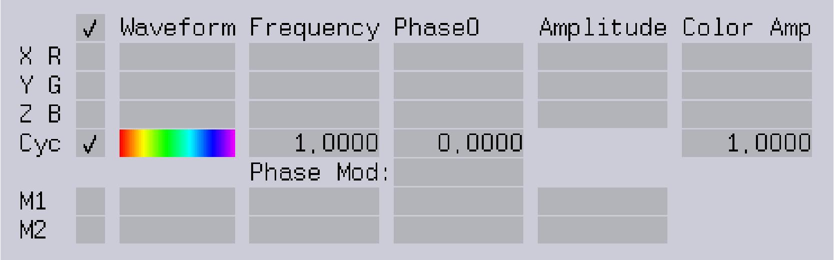
Bank I/II



	✓	Waveform	Frequency	Phase0	Amplitude
X	✓	Sinewave	1.0000	0.0000	0.500000
Y	✓	Sinewave	1.0000	90.0000	0.500000
Z					
			Phase Mod:		
M1					
M2					

The four provided oscillator banks are structured almost identically. Each bank contains 5 resp. 6 oscillators. The on/off buttons on the far left determine whether a function is active. The topmost button turns the whole oscillator bank on and off, the buttons below control the individual parameter lines.
The values can be changed in the usual way by clicking on the desired parameter field with the mouse. The three lines labeled X, Y and Z affect the corresponding coordinates. M1 and M2 create two more oscillations for further changing amplitude and phase. The parameter Phase Modulation allows using the oscillation created with M1 and M2 to modulate the phase of the previously generated oscillation.
Simply turning on the selected function will not lead to any visible results. At least the parameter Amplitude must not be zero to achieve any visible effect, otherwise all results are multiplied by zero and no lines are visible.

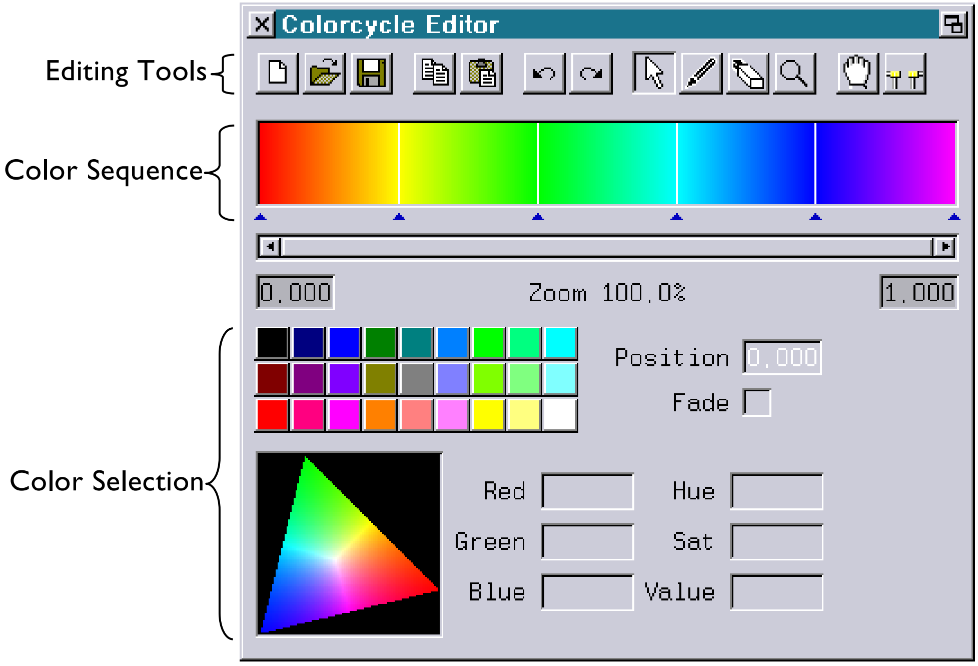
Bank III/IV



	✓	Waveform	Frequency	Phase0	Amplitude	Color Amp
X R						
Y G						
Z B						
Cyc	✓		1.0000	0.0000		1.0000
			Phase Mod:			
M1						
M2						

Banks 3 + 4 contain an additional parameter line marked Cyc (ColorCycle) and a parameter column labeled Color Amplitude. These functions calculate colors for the calculated geometry. Color Amplitude computes colors according to an additive three-component color model. The oscillations generated on bank 3 are used to calculate geometry data (X, Y and Z coordinates) as well as the corresponding three color components. If only one of the two functions is to be used, the amplitude for the geometry or the color amplitude simply have to be set to zero.
The oscillations generated by bank 4 are only used for color calculation.

The ColorCycle Editor



0.000

Zoom 100.0%

1.000

Position 0.000

The second possibility for generating colors is the <Cyc> function.
This will not calculate the individual color components. Instead the complete generated geometry will be colored over its total length with a sequence of colors.
In order to change this sequence it is first necessary to select a color position (marked by the blue triangles below the sequence). Then the colors can be changed with the color triangle or by clicking on the color buttons.

Detailed information about the use of the ColorCycle editor can be found in the separate manual chapter "Color Cycle Editor".

Hint:
Of course here the parameter Color Amp also has to be unequal zero in order to generate colors.
It is worth experimenting with the settings, the colors generated by the laser have a larger bandwidth and higher intensity than the usual monitor colors.

The Picture Calculation Parameters:

The picture calculation within an oscillator bank always follows the same formula for the X, Y and Z coordinate. The result for a coordinate is the sum of the calculated values from the three possible oscillator banks for the geometry calculation.
A coordinate within the oscillator banks is calculated using the formula:

Coordinate = (Waveform (Base Frequency * Frequency * (Time + Phase0)) * Amplitude) * M1 * M2

The lines generated by the sequence of the calculated points often seem two-dimensional, however, they are real 3D lines with X, Y and Z coordinates. It is therefore possible to shift and rotate the resulting patterns in 3D space, just like any other picture used in the program. Sometimes a graphic generated with the Abstract can only truly be appreciated in a moving 3D view.

Each oscillator bank contains four resp. five parameter columns for setting the values mentioned above. Except for the Waveform column, the individual parameter fields are operated according to the usual procedure for numerical values.

Waveform

This field is used to define the wave function which is to be applied. Possible wave forms are:

OFF	(oscillator is deactivated)
Sinewave	
Triangle	
sQuare	(rectangle)
sAw	(sawtooth)
Pic1X	(X-coordinate of the picture specified under Pic1)
Pic1Y	(Y-coordinate of the picture specified under Pic1)
Pic1Z	(Z-coordinate of the picture specified under Pic1)
Pic1I	(Intensity of the picture specified under Pic1, 0 if blanked)
Pic1XB	(like Pic1X, additionally blanked segments will also be blanked in the abstract)
Pic2X	(X-coordinate of the picture specified under Pic2)
Pic2Y	(Y-coordinate of the picture specified under Pic2)
Pic2Z	(Z-coordinate of the picture specified under Pic2)
Pic2I	(Intensity of the picture specified under Pic2, 0 if blanked)
Pic3X	(X-coordinate of the picture specified under Pic3)
Pic3Y	(Y-coordinate of the picture specified under Pic3)
Pic3Z	(Z-coordinate of the picture specified under Pic3)
Pic3I	(Intensity of the picture specified under Pic3, 0 if blanked)
Pic4X	(X-coordinate of the picture specified under Pic4)
Pic4Y	(Y-coordinate of the picture specified under Pic4)
Pic4Z	(Z-coordinate of the picture specified under Pic4)
Pic4I	(Intensity of the picture specified under Pic4, 0 if blanked)

BufAX	(X-coordinate of the picture located in buffer BufA)
BufAY	(Y-coordinate of the picture located in buffer BufA)
BufAZ	(Z-coordinate of the picture located in buffer BufA)
BufAI	(Intensity of the picture located in buffer BufA, 0 if blanked)
BufAXB	(like BufAX, additionally blanked segments will also be blanked in the abstract)
BufBX	(X-coordinate of the picture located in buffer BufB)
BufBY	(Y-coordinate of the picture located in buffer BufB)
BufBZ	(Z-coordinate of the picture located in buffer BufB)
BufBI	(Intensity of the picture located in buffer BufB, 0 if blanked)
BufCX	(X-coordinate of the picture located in buffer BufC)
BufCY	(Y-coordinate of the picture located in buffer BufC)
BufCZ	(Z-coordinate of the picture located in buffer BufC)
BufCI	(Intensity of the picture located in buffer BufC, 0 if blanked)
BufDX	(X-coordinate of the picture located in buffer BufD)
BufDY	(Y-coordinate of the picture located in buffer BufD)
BufDZ	(Z-coordinate of the picture located in buffer BufD)
BufDI	(Intensity of the picture located in buffer BufD, 0 if blanked)

Frequency (-256.0 to 256.0)

The abstract generator creates geometries based on overlapping oscillations, therefore the frequency is an important parameter. The parameters for setting the frequencies within the individual oscillator banks are multipliers of the base frequency. Entering 2 into the corresponding field will therefore result in a doubling of the base frequency.

Phase0 (-360.0 to 360.0)

The parameter Phase0 determines the starting value of the calculations. This value "shifts" the selected wave. A value of 180 "shifts" the wave by half the duration of the oscillation.
Example: Two sine oscillations with the same frequency and the same amplitude, one for the X coordinate and one for the Y coordinate, where the oscillation of the Y coordinate has an offset of 90 compared to the offset of the X coordinate will result in a circle.

Amplitude (-1.0 to 1.0)

The amplitude determines the maximum value of the wave. An amplitude of 0 always leads to a result of 0. In practice this value changes the size of the displayed picture. If the maximum value of 1.0 is reached and does not result in a sufficiently large display size, the whole picture has to be scaled.

Color Amp (-256.0 to 256.0)

Similar to the geometry this value determines the maximum of the corresponding color component. A value of 256 means that the exact color specified for the respective color component in the color definition area will be used.

Color Definitions

The preset base colors ("BMP Red", "BMP Green" and "BMP Blue") can be replaced with user defined color definitions by clicking on the desired color field with the mouse.
The values calculated for colors in the oscillator banks and the ColorCycle-Editor are only calculative values until the colors are actually displayed. In contrast to calculated geometries the final result for colors is always positive. In case the result for one color component should be negative this would mean a value of 0 for the respective color component. The specified offset value will be added to the calculated result of the color component. This way, as an example, line segments which were black before (red=0, green=0 and blue=0) can have colors again in the end.

BaseFreq (Base Frequency)

When interpolation is deactivated (either in the "Advanced Parameters" or within the trickfilm using the command "SCanningParameter") the frequency and the base frequency will be directly related to the displayed output frequency:

Output Frequency (Hz) = (Base Frequency * Oscillator Frequency * Scanning Frequency (kHz)) / 20.8333

The Display Parameters

Abstract# <0 - 3>
Displays the number of the activated data set. Up to four different definitions are possible within one program.

Position# <0 - 15>
The generated picture will be placed at position <0 - 15> in 3D space. The selected position must be defined before it can be used.
In case no Position# is specified (None) the abstract command will use an implicit position where the Z-component will be set to 0. Therefore specifying a Position# is obligatory for three-dimensional abstracts.

Points <0 -X>
If two or more pictures are to be displayed simultaneously, this value determines the number of points displayed by the abstract command. When point count X is reached the display will switch to the next picture.

OverlapB <0 - X>
The abrupt change of direction of the laser beam necessary when two or more pictures are displayed will sometimes lead to display errors in the picture geometry due to the mass inertia of the scanner mirrors. The value set with OverlapB counteracts this effect because a certain number of points are drawn additionally with deactivated laser. X is the number of points which will be drawn additionally with deactivated (blanked) laser in order to conclude the drawn picture exactly.

OverlapD <0 - X>
By activating OverlapB the part of the picture which was drawn with deactivated laser will be left out of the display when the laser beam returns to the picture. OverlapD counteracts this effect by causing the repeated display of X points at the end of the previous part of the picture. Otherwise there will be visible gaps in the display of the picture.

Interpol <0 - X>
This option allows a simplified display of the calculated geometry. Every X^{th} point will be calculated, changing the geometry's character by giving it more edges and angles. Under certain conditions this can save computing time, sometimes improving the display quality.

Reinit <On/Off>
When this option is activated the abstract generator will start all its calculations with the specified values resp. zero. Specifying the number of points X allows displaying the calculations up to point X and then restarting.

Bright
The global brightness of the displayed picture can be adjusted with this option.

AutoBlank <0 - X>
This value specifies a distance. When the distance between two consecutive points is larger than specified here the visible beam will automatically be deactivated.

Picture References

The text fields labeled Pic1 - Pic4 contain the names of pictures which are to be used instead of a predefined wave form. Clicking on the desired field will open a selection list where an existing picture can be selected. This picture will then be available as a wave form.

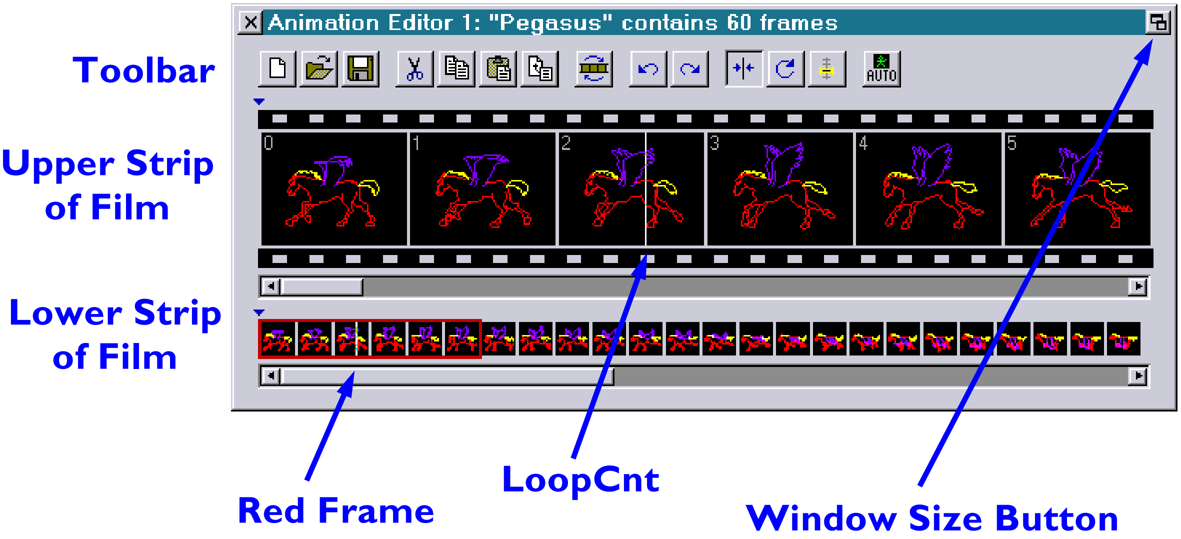
The fields BufA - BufD allow setting a "Picture Buffer" as a wave form in the abstract.

Hint:

Finally it should be noted that "non-mathematicians" will frequently find it complicated to deal with the many functions and possibilities. This can be overcome by intensively working with the possibilities through "trial and error".
It is helpful to load the Boot "Abstracts.BOOT" from the demo files and use the figures as the basis for the first attempts.

Overview

The Animation Editor is a tool which was designed for conveniently creating new animations and editing existing ones.



Workflow

The Animation Editor contains two strips of film which show the course of the animation. The lower strip shows an excerpt of an animation made up of small pictures and provides an overall impression of the sequence of the animation. The upper strip contains a smaller excerpt of the same animation, consisting of a few larger pictures, which makes detailed viewing of single frames possible. A red frame in the lower strip marks the portion of this strip which is visible in the upper strip.

Size and position of the Animation Editor window can be changed as you like. By clicking into the window size button the window will be moved and sized to an alternate window position with an alternate size. Clicking again restores the window to it's original size and position.

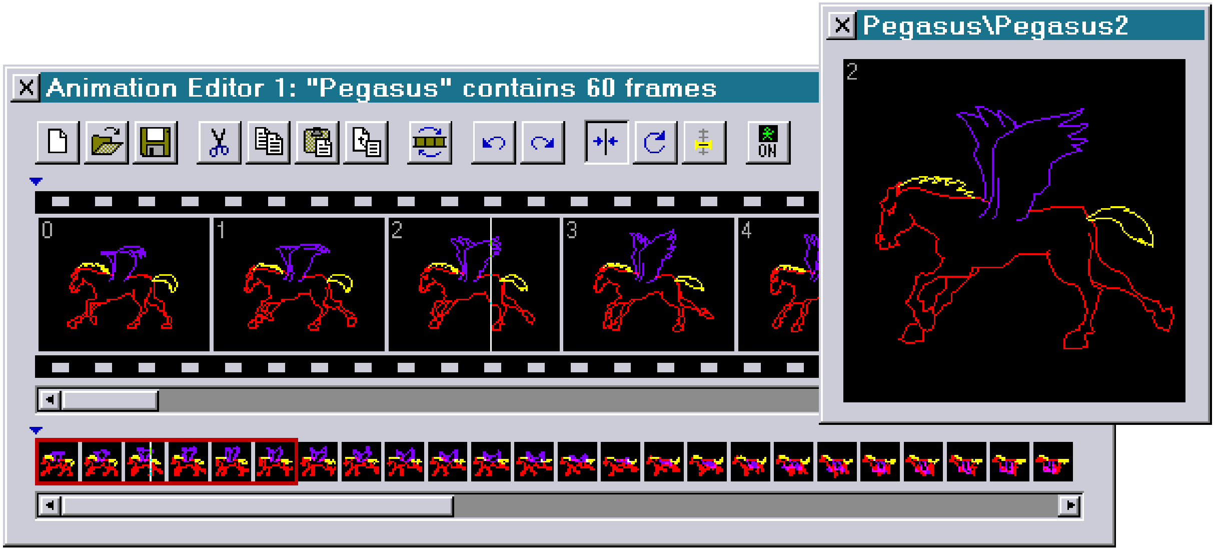
In order to work on an animation, an existing catalog of pictures can be loaded and then altered by cut, copy and paste. For adding pictures from different catalogs, the Animation Editor can be simultaneously opened more than once. Pictures can then be copied or cut from one editor and pasted into another.

Clicking into any frame in one of the strips of film with the left mouse key marks this frame as the active frame (for later insertions), and the LoopCnt, visible as a thin white line, is set at this position. While the mouse key is pressed, the LoopCnt and the active frame can be changed. If the end of the visible strip of film is reached by doing this, the visible section is shifted (scrolled).

In order to select certain passages of an animation, the middle mouse key can be used as in the Trickfilm Editor. By pressing, holding and dragging, a section is defined directly. Single clicks will expand or reduce the active section; double clicking will delete the selection.

Double clicking of the left mouse key into a picture of the animation causes this picture to be opened in the Picture Editor.

A special display window can be opened which shows only the active frame, see picture.



When the left mouse key is pressed and the mouse pointer is dragged through the animation in one of the strips of film, the display window shows each frame as the mouse pointer reaches it. This makes it possible to examine transitions between frames and to detect possible discrepancies immediately.

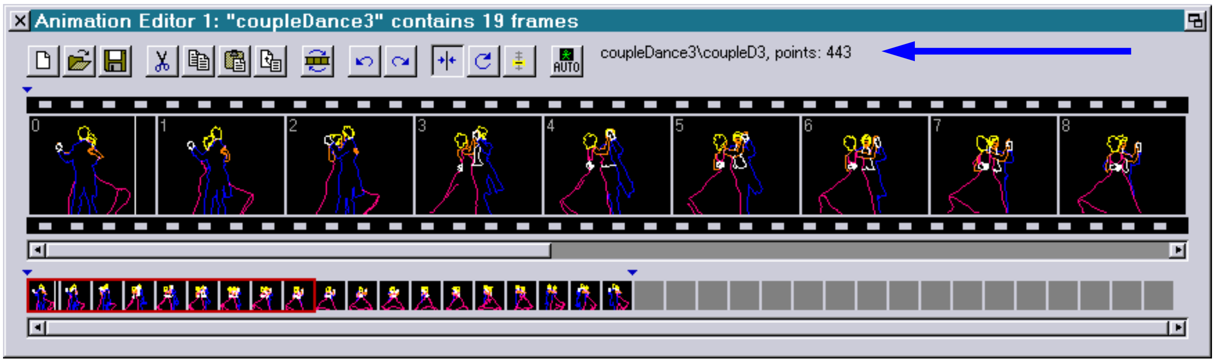
Meaning of the icons

🗋	New Animation	Creates a new animation.
📂	Open Animation/Catalog	Loads an animation from the memory or from the hard disk into the Animation Editor. If a catalog is to be loaded and converted into an animation, the option "/ImportCat" can be activated. This happens automatically if a catalog is selected in the file selection window.
💾	Save Animation	When saving an animation, you can select several options: **/CreateCat**. This saves the animation and automatically creates a catalog of the same name containing copies of all the pictures used in the animation, numbered consecutively as they appear. Pictures used multiple times in the same animation are only saved once. This option is available for saving to disk only, therefore it will not work combined with the **/Nodisksave** option. If you want every single picture copied to the catalog, no matter how often one picture appears in the animation, select the **/NoCollapse** option. This only works in combination with **/CreateCat**. If **/Nodisksave** is selected, the animation is only saved in the memory, not on hard disk. **/Overwrite** has to be used if there is an existing animation with the same name which is to be replaced with the current one.
✂	Cut Animation To Clipboard	Cuts (removes) a selected section from the animation and copies it to the clipboard.
⧉	Copy Animation To Clipboard	Copies a selected section from the animation to the clipboard without removing it from the animation in the editor.
📋	Paste Animation From Clipboard	Inserts the contents of the clipboard in front of the active frame in the Animation Editor.
⎘	Insert Animation/Catalog/Picture	Inserts a picture, a catalog or an animation right after the active frame. First click on a frame to determine where the new frame is to go, then click the insert button. A window opens to let you select the picture, catalog or animation to insert.
🔄	Flip selected frames	With the flip button the order in which the selected frames appear in the animation is reversed.

↶	Undo	Cancels the last editing action. It is possible to undo the last eight steps.
↷	Redo	Revokes one undo action, the last step which was canceled is executed again.
→|←	Default LoopCnt	Enable this button to move the LoopCnt position marker manually by dragging it with the mouse.
↻	Run Loop	When the "Run Loop" function is enabled, the LoopCnt will be executed with the time given in the loop and it will be looped, too.
	LoopCnt by Fader 7	This button will enable the control of the LoopCnt by the Fader 7 of the Faderboard.
AUTO	Frame Display Mode	Determines the state of the single-frame display window. Three different modes are possible: - ***Off*** The window is not opened. - ***Auto*** The window opens when any frame is clicked into. It stays open as long as the mouse key is held down; it closes as soon as the mouse key is released. - ***On*** The window is permanently open.

Information for frames

While the mouse pointer is over one frame, you will get some information about that frame:



Animation Editor 1: "coupleDance3" contains 19 frames

coupleDance3\coupleD3, points: 443

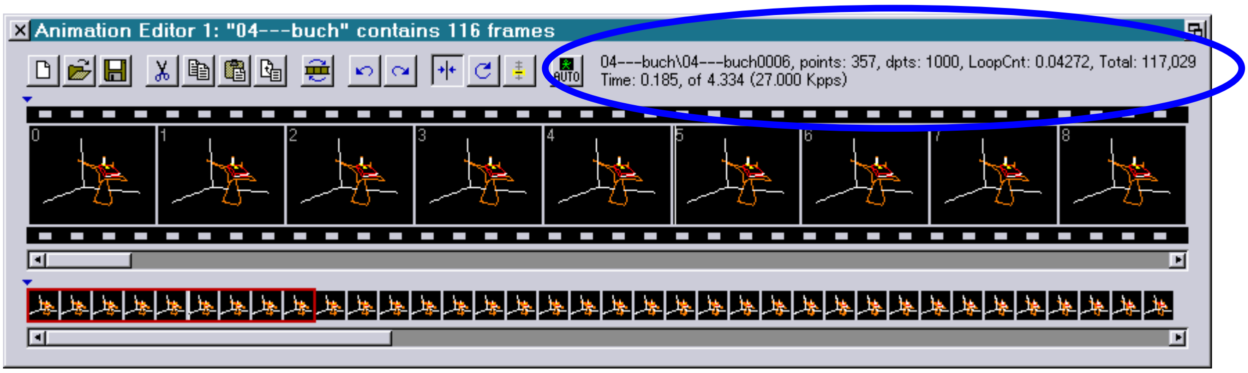
Format: Catalog \ Picture \ Points

Catalog: The name of the catalog the frame has been saved in

Picture: The name of the picture

Points: The number of digitized points

If the frame has been created with "proTize", you will get additional information:



Animation Editor 1: "04---buch" contains 116 frames

04---buch\04---buch0006, points: 357, dpts: 1000, LoopCnt: 0.04272, Total: 117,029
Time: 0.185, of 4.334 (27.000 Kpps)

Format: Catalog \ Picture \ Points \ Display Points \ LoopCnt \ Total Points \ Time

Catalog: The name of the catalog the frame has been saved in.

Picture: The name of the picture.

Points: The number of digitized points.

Display Points: The number of points which will be sent to the projector. This also includes automatically inserted points due to interpolation or repeats.

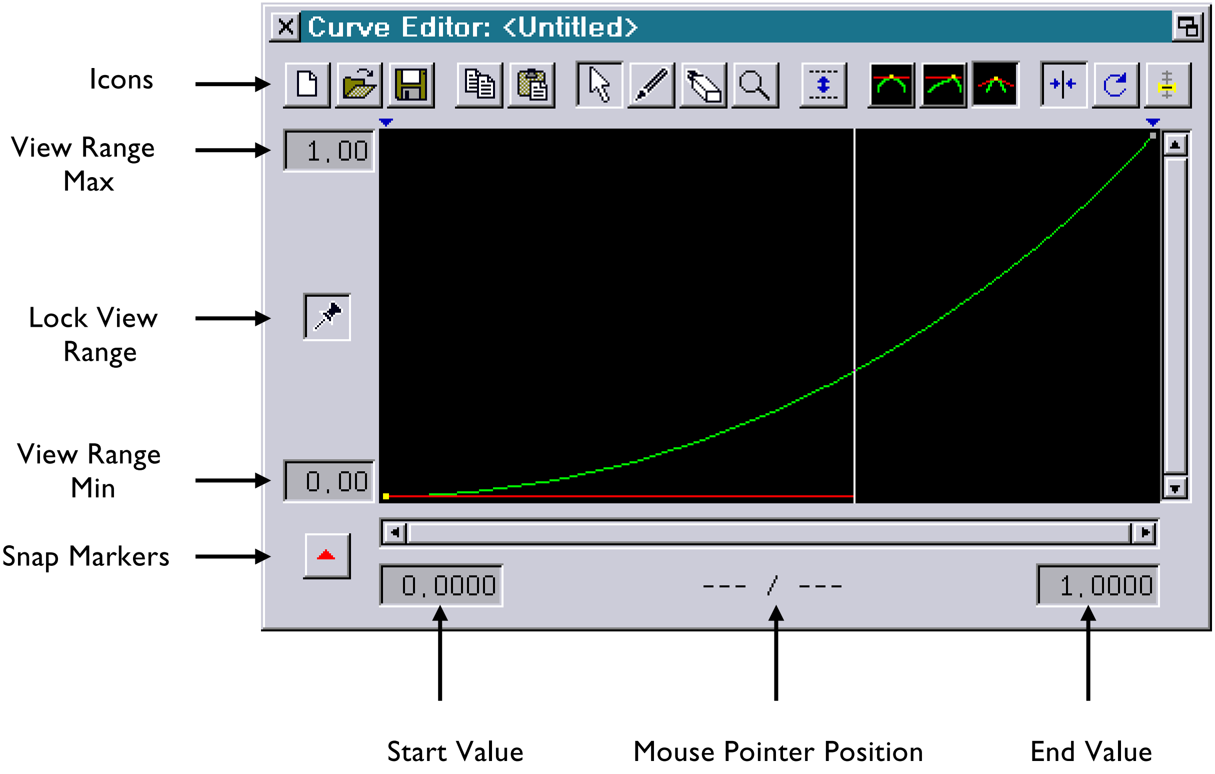
LoopCnt: Relative position between 0.0 and 1.0 where the frame starts.

Total Points: The sum of the display points of all frames within the animation.

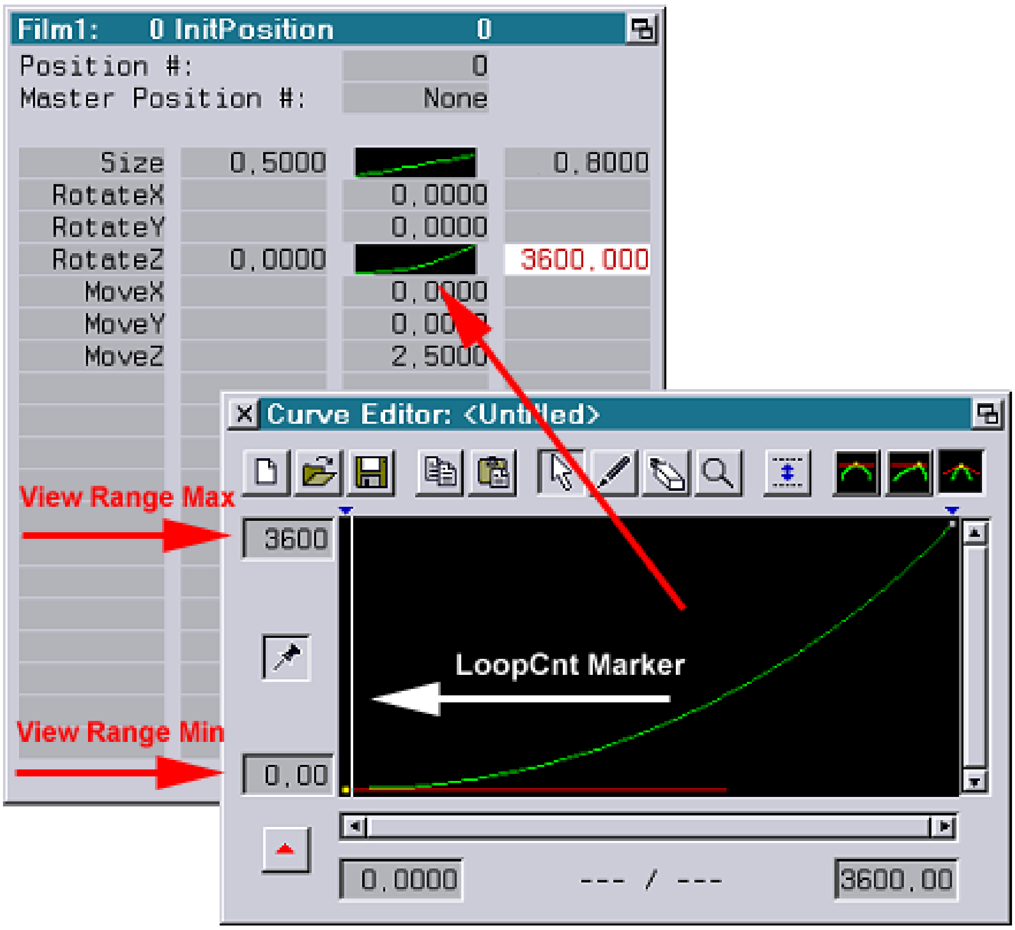
Time: The absolute time of the frame in seconds, the duration of the entire animation and the scanning frequency on which the time calculation is based.

Chapter Curve Editor

The Curve Editor is a tool to create and edit curves.



The Curve Editor window opens up automatically if a curve is entered into a parameter field.



Values shown in the Curve Editor Window

The two values "View Range Max" and "View Range Min" show the highest and lowest values in the range of the curve view. These are gadgets, so values can be changed by editing the value or using the mouse poti mode.

"Start Value" and "End Value" show the Y-values of the left and right fixed control points. These values may also be changed by changing the values in the gadgets.

If the mouse pointer is in the curve area, its current position is displayed as pair of the X- and Y-values between the two gadgets containing the Y-values of the left and right control points. This is only an information display.

Icons of the Curve Editor

New Curve

A new curve can be created by clicking into this icon. A confirmation is necessary to replace the current curve with the new one.
When starting a new curve, only the two fixed control points are available.

Open Curve

Curves can be loaded ...

Save Curve

... and saved separately by clicking the corresponding icons.

Copy Curve To Clipboard

Clicking into the icon will place the curve currently edited into the clipboard.
A curve in the clipboard stays there until it is replaced by another one, even if the Curve Editor is closed and reopened later.

Paste Curve From Clipboard

A curve can be reproduced anywhere else by clicking into this icon. This will not delete the contents of the clipboard.

Edit Pointer

The position of points and tangents can be edited using the edit tool.
Pressing and holding the left mouse button next to a point will change the position of the point if the mouse is moved. The same applies to a tangent if it is next to the mouse position.
If next to a point, the tool will always move this point, even if tangents are nearer to the mouse position.
The middle mouse button can be used similarly to edit tangents, even to pull them out of a point.

Insert Points

Additional points can be inserted by double clicking into the area where the curve is displayed, or by activating the insert tool.
If the insert tool is active, each click into the curve will create a new point at the mouse pointer position until the tool is switched off.

Delete Points

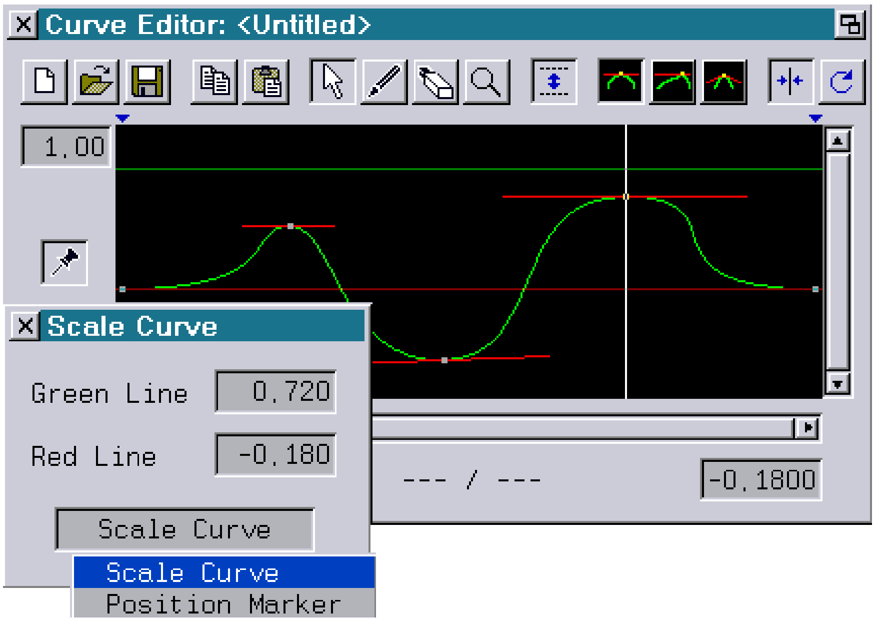
If the delete tool is active, every click next to a point will delete that point.

Zoom In/Out

Clicking into the curve window with the left button will zoom into the curve, centered around the position clicked to.
Clicking with the middle mouse button will zoom out of the curve. Double clicking into the icon will show the curve filling the whole window.

Scale Curve

A red and a green auxiliary line will appear. These auxiliary lines can by moved by dragging them by mouse or by entering values in the fields 'Red Line' and 'Green Line':



1,00

Green Line 0,720

Red Line -0,180

-0,1800

'Scale curve' stretches the whole curve in the direction of the Y-axis according to the shift of the red and green auxiliary lines.
'Position marker' allows the setting up of the red and green auxiliary lines without changing the curve itself. This is used to set the auxiliary lines to the desired starting position for scaling.

Tangent Connected

The two tangents of a point are moved simultaneously. Their length and direction stay unchanged.

Tangent Angle Connected

The two tangents of a point are moved independently. Their lengths can be altered, but their directions stay the same.

Tangent Free

The two tangents of a point are moved completely independently. Lengths and directions of the tangents can be changed.

Default LoopCnt

Enable this button to move the LoopCnt position marker manually by dragging it with the mouse.

Run Loop

When the “Run Loop” function is enabled, the LoopCnt will be executed with the time given in the loop and it will be looped, too.

LoopCnt by Fader 7

This button will enable the control of the LoopCnt by the Fader 7 of the Faderboard.

Lock View Range

This tool is a switch. If it is enabled, points and tangents cannot be moved out of the current view range by mouse.
If disabled, points and tangents can be moved everywhere and the window will automatically adapt to the new view size.

Snap Markers

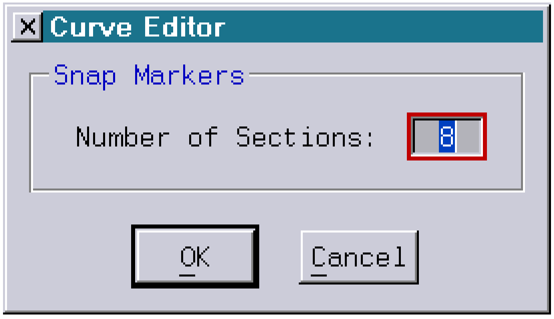
Snap markers can be set by clicking into the bar between the curve and the horizontal scrollbar.
They are displayed as small red triangles.

Control points will snap to a snap marker if moved near enough to one.

Snap markers can be moved by pressing and holding the left mouse button and moving the mouse. If a snap marker is not released inside the bar, it is removed.

Snap markers can also be set by pressing <Space> while a control point or a tangent is moved (see Hotkeys).

A double click on the icon "Snap Markers" removes all previously defined snap markers and sets up snap markers at default positions according to the number of sections entered in the following dialog:



Curve Editor

Snap Markers

Number of Sections: 8

Hotkeys in the Curve Editor

Some hotkeys are available for comfortable use. They are only operational if a point is active and the mouse button is held down:

<Del>	Deletes the active point.
<Space>	Sets a "Snap Marker" to the current X-position.
<+> and <->	Zooms in and out around the current mouse pointer position.
<Esc>	Cancels an edit, identical to clicking the right mouse button.

Curves used in command "SetVariables":

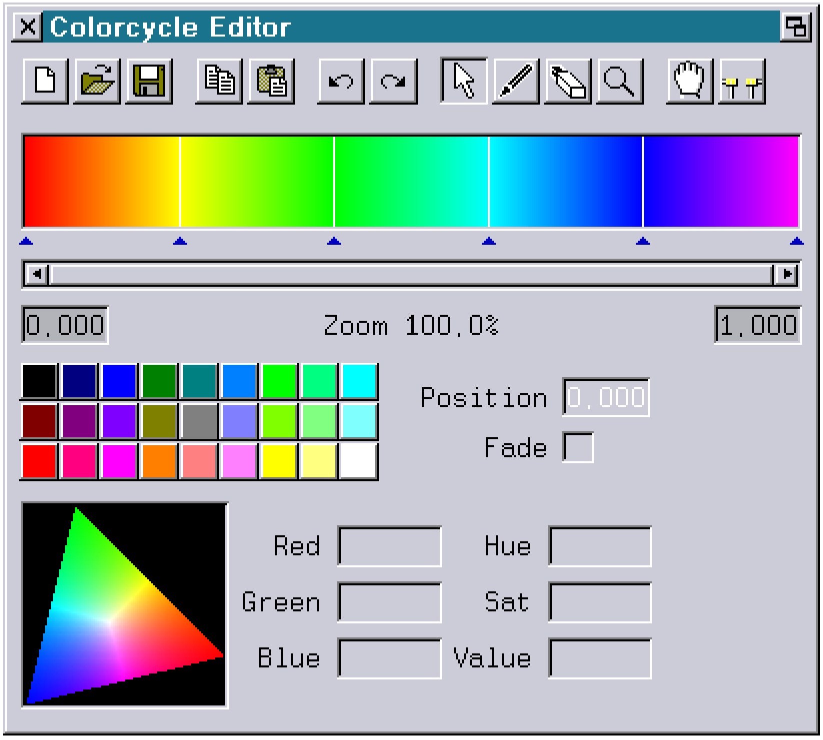
Curves used in a parameter field are unnamed. They exist only at the position where they are used.

Curves used in a "SetVariables" command must have a name, and they exist in the whole trickfilm in which they are used.
They expect an input parameter for which the curve value is to be calculated.
This input value should be in the range [0.0; 1.0], the allowed input range for curves. In most cases this parameter will be the variable "LoopCnt", but also "Attack0 ()" and its variants may be used like any other arithmetical expression defined by the user.

An example: **var : = ~MyCurve(LoopCnt)*100**

var	is the name of a variable which is to be calculated.
~	is a special symbol for the Lasergraph DSP telling him that a curve is following.
MyCurve	is the name of the curve.
LoopCnt	is the input parameter used by the curve to calculate its output value.
*100	is a scaling factor.

The Color Cycle Editor is opened by clicking a color cycle field in the trickfilm parameter window as it can be found for example in the `Abstract` command.



0.000 Zoom 100.0% 1.000

Position 0.000

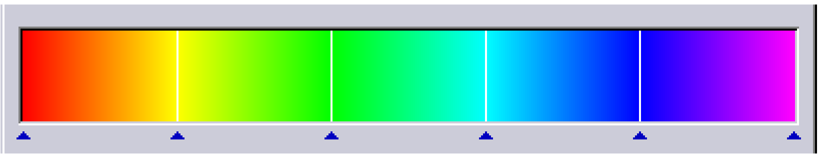
Fade □

The Toolbar

🗋	New	Opens a window with default color cycles. Click one to load it into the main window.
📂	Open	Opens a saved color cycle.
💾	Save	Saves the color cycle to disk.
📄	Copy	Copies the complete color cycle to the clipboard.
📋	Paste	Replaces the complete color cycle with the one from the clipboard.
↶	Undo	Multiple level Undo.
↷	Redo	After an Undo, the step can be restored with Redo.

	Pointer	Single (short) click to activate a point (the little triangle below the color bar turns red), click and hold to move a point. When a point is moved over other points, they are deleted as soon as the mouse button is released. Double click to insert a new point.
	Insert	Insert new points.
	Delete	Delete points.
	Zoom	Clicking the color bar with the left mouse button zooms in, middle mouse button zooms out. See the end of this chapter for keyboard shortcuts. Double click this button to restore 100% view.
	Adjust	Test mode to check how a color turns out on the laser. When activated, clicking (and keeping the mouse button clicked) into the color palette, color triangle or color adjusting area displays the whole picture in the clicked color until the mouse button is released. Note: This only applies to the Trickfilm command Abstract
	Connect	Connects Start and End point of the color cycle, so that the color bar starts and ends with the same color. This gives a visible result only if the first and/or the last segment is faded, not solid.

The Color Bar



The color bar contains at least two points, the Start point (at the far left) and the End point (at the right edge). Every point is marked with a vertical white line and a little triangle below the color bar, which turns red when the point is selected.

While the Start point and the End point have fixed positions, any additional point can be moved within the color bar by dragging it with the left mouse button. If a point is dragged over other points, these are deleted as soon as the mouse button is released.

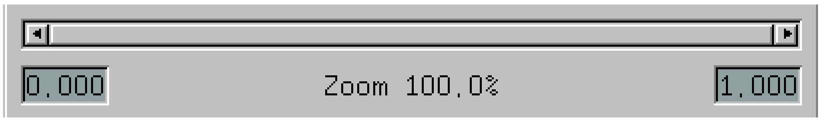
To add a new point, double click with the point tool selected, or single click with the insert tool selected.

To delete a point, single click it with the delete tool selected, or grab it with the mouse, keep the mouse button pushed and press the <Del> key.

A color is defined for every point. When a point is selected, the color values (RGB and HSV) are displayed in the lower part of the window. A new color can then be chosen by clicking a button of the color palette, clicking a point inside the color triangle or by changing values in the adjustment section.

The color of the point affects the segment to the right of the point. It can either contain the solid color of the point or a fade from the color of the point to the color of the next point to the right. Clicking the segment with the middle mouse button toggles between solid color and fade.

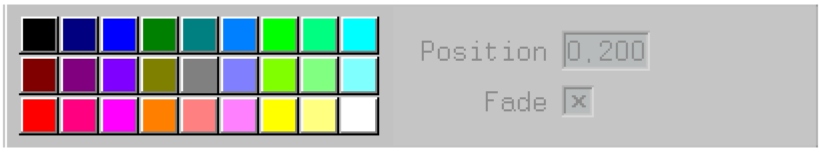
The Zoom Bar



0.000 Zoom 100.0% 1.000

Size and position of the visible part of the color bar are symbolized by the handle in the scrollbar. The two values indicate the position of the left and right edge of the visible part, where 0.0 is the far left end and 1.0 is the far right end of the color bar. These values can be changed either by keyboard or by click-dragging with the mouse. See the description of the zoom button above for different ways of zooming. The zoom factor is displayed in the middle.

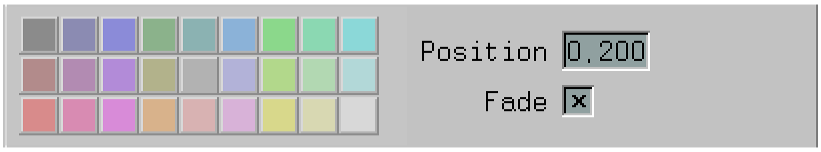
The Color Palette



Position 0.200

Fade ☒

Some fixed colors can be selected by simply clicking them on the color palette.



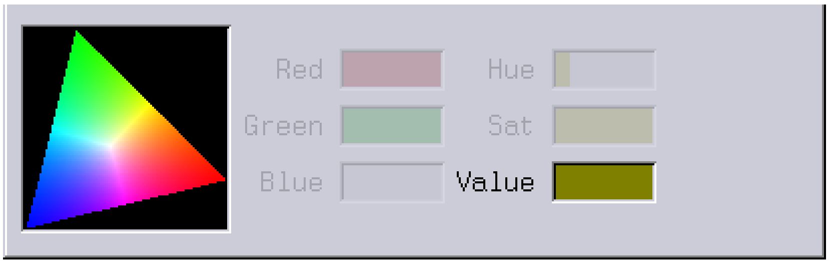
Position 0.200

Fade ☒

In the Position field, the exact position of the selected point between 0.0 (left end) and 1.0 (right end of the color bar) is indicated. It can be changed manually.

If Fade is checked, the color fades from the color of the selected point to the color of the next point to the right. If it's unchecked, the complete segment to the right of the selected point gets the color of the selected point. Changing this checkbox is equivalent to clicking the segment on the color bar with the middle mouse button.

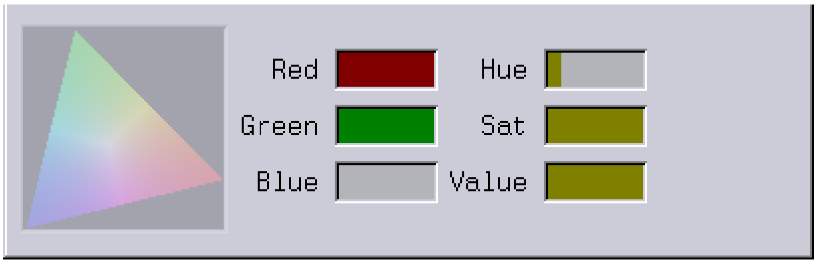
The Color Triangle



With this tool, colors can be selected and fine tuned in a very convenient way. First, you can simply click inside the triangle to select a color. Please note that the Value setting on the right affects the brightness of the selected color, so if the color looks too dark, this setting must be changed. The second way to change the color is to click a color and hold the mouse button, then move the mouse to fade to other colors.

In combination with the Adjust button, this feature can be used to perfectly work out colors on your laser.

Color Adjusting



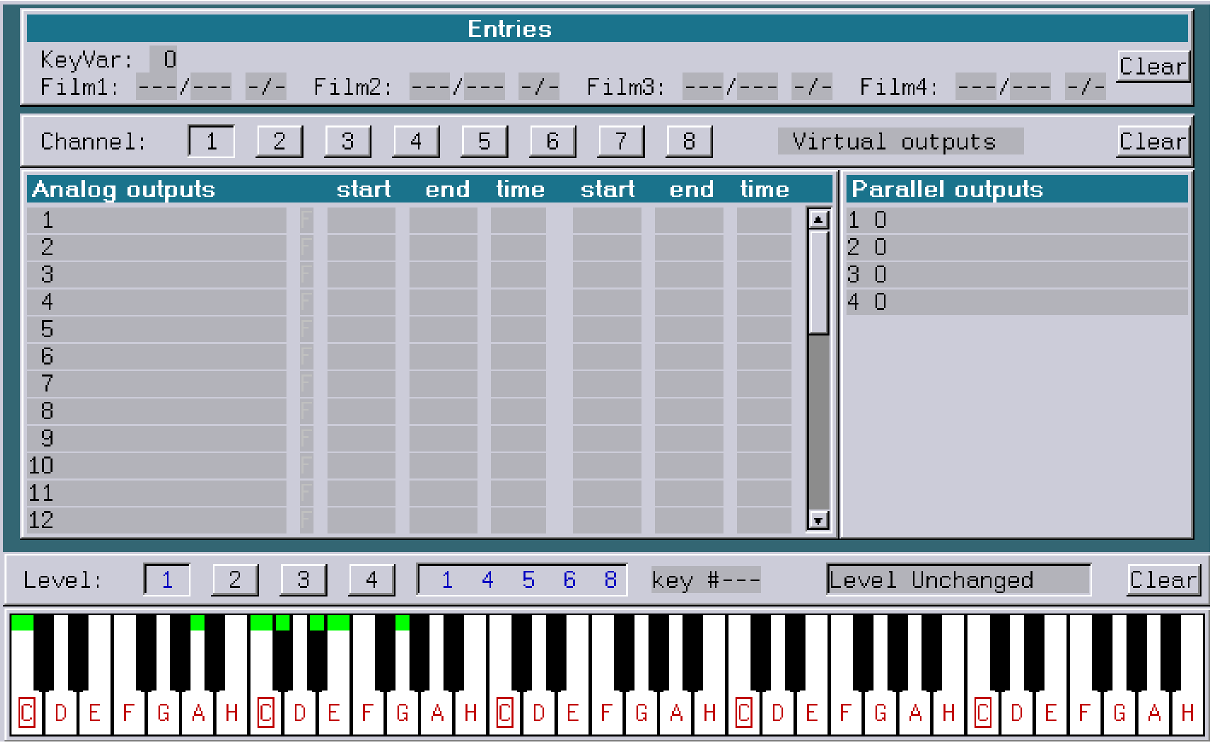
The Red, Green, Blue and the Hue, Saturation and brightness values are displayed as soon as a point is selected. All values can be changed either by keyboard or by click-dragging with the mouse.

Keyboard Shortcuts

Click and hold a point in the color bar, then press one of the following keys:

Key	Action
+	Zoom in
-	Zoom out
Del	Delete point

Introduction



KeyVar: 0
Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

1
2
3
4
5
6
7
8
9
10
11
12

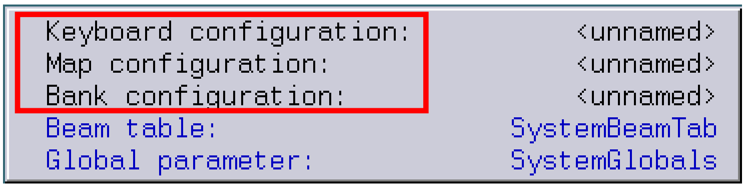
1 0
2 0
3 0
4 0

1 4 5 6 8 key #--- Level Unchanged

The Keyboard / Map / Bank Configuration is the section where you can define and name the Analog and Parallel outputs and the assignment of a Midi-Keyboard.

When a Midi key is assigned to a function, it is marked with a green rectangle.

Key-, Map- and Bank configuration can be accessed by clicking on the corresponding name in the info window on the main page.

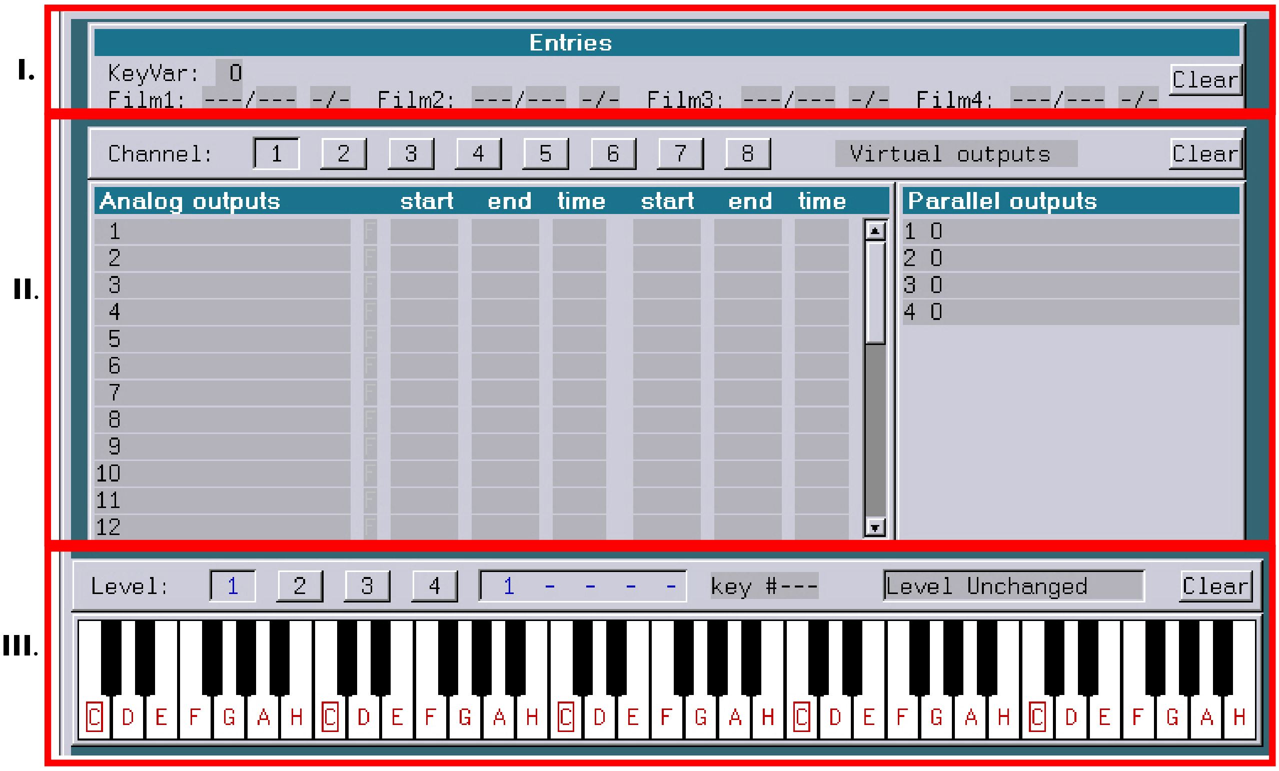


Keyboard configuration: <unnamed>
Map configuration: <unnamed>
Bank configuration: <unnamed>
Beam table: SystemBeamTab
Global parameter: SystemGlobals

The Keyconfig can also be accessed by the function key F5 from the start screen.

Keyconfig

The Keyconfig consists of three main windows.



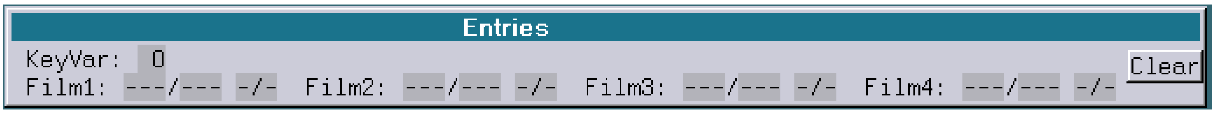
KeyVar: 0
Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

Analog outputs	start	end	time	start	end	time
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						

Parallel outputs	
1	0
2	0
3	0
4	0

1 - - - - key #--- Level Unchanged

I. The Entry Window

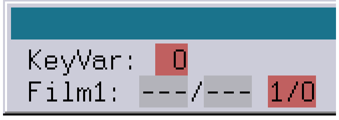


KeyVar: 0
Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

The entry window is where the Entries are assigned to a Midi key.
To assign these parameters, a Midi-key has to be selected with the mouse or directly pressed on the Midi keyboard.
For each Film four parameters can be set:

1) KeyVar On/Off (for each Film)
2) KeyVar number (one for all Films)
3) Entry – pressed Midi key
4) Entry – released Midi key

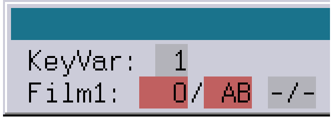
KeyVar:



```
KeyVar:  0
Film1: ---/--- 1/0
```

The KeyVar is a special Variable that can be recalled inside the film.
It enables you to retrieve the state of a Midi key (pressed or not).
This variable is enabled by activating the field "1/0". In the parameter field the KeyVar number is then highlighted. By clicking on the number you can change the value in the command line.
Range: 0 - 63

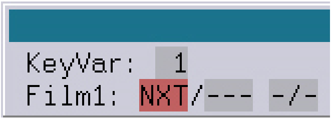
Entry:



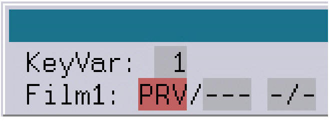
```
KeyVar:  1
Film1:   0/ AB -/-
```

The two parameters beside the Film number (here Film1) define the Entry for pressing/releasing the selected Midi key.
In the example screenshot above, pressing the selected Midi key will start Entry 0 in Film 1 and releasing the key will "Abort" the Film.
Leaving the release parameter empty will lead to no action when releasing the key.
Entries are defined by clicking into the "pressed key" or "released key" field and then by typing the Entry number into the command line.
"AB" means that the Film will be stopped.

Instead of fixed entry numbers “Next” and “Prev” can be entered to start the next resp. preceding entry:



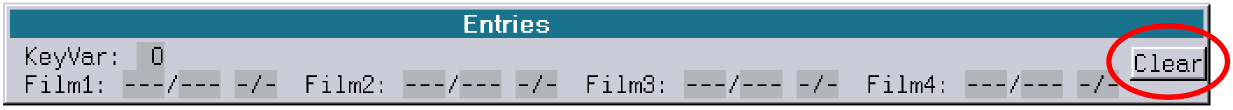
```
KeyVar:  1
Film1: NXT/--- -/-
```



```
KeyVar:  1
Film1: PRV/--- -/-
```

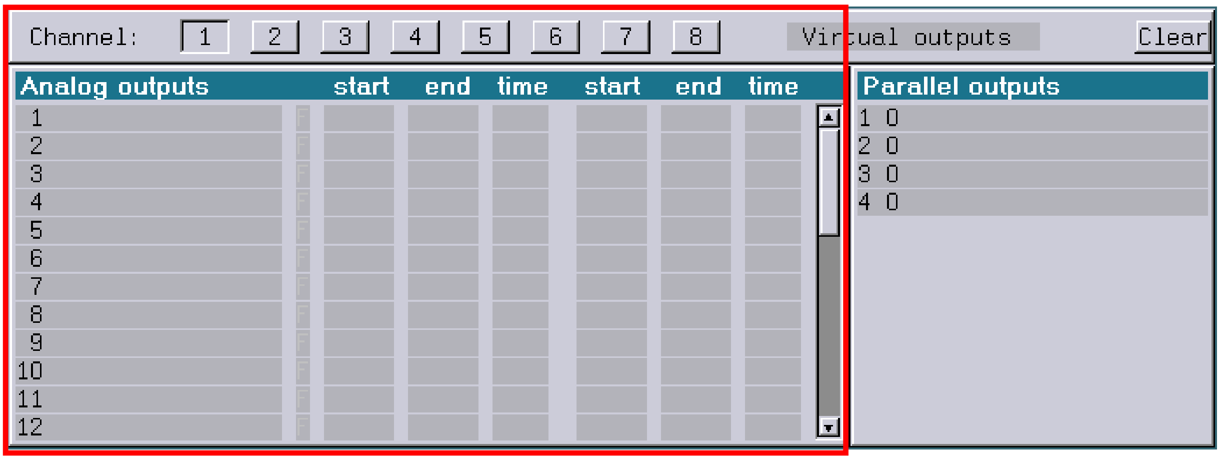
Deleting settings

The use of the button "Clear" will erase all settings made for the selected key.



```
                                  Entries
KeyVar:  0                                                                   Clear
Film1: ---/--- -/-  Film2: ---/--- -/-  Film3: ---/--- -/-  Film4: ---/--- -/-
```

II. Analog outputs



The Analog output window is used to assign an analog value to an Analog output (DMX / Bankbox).

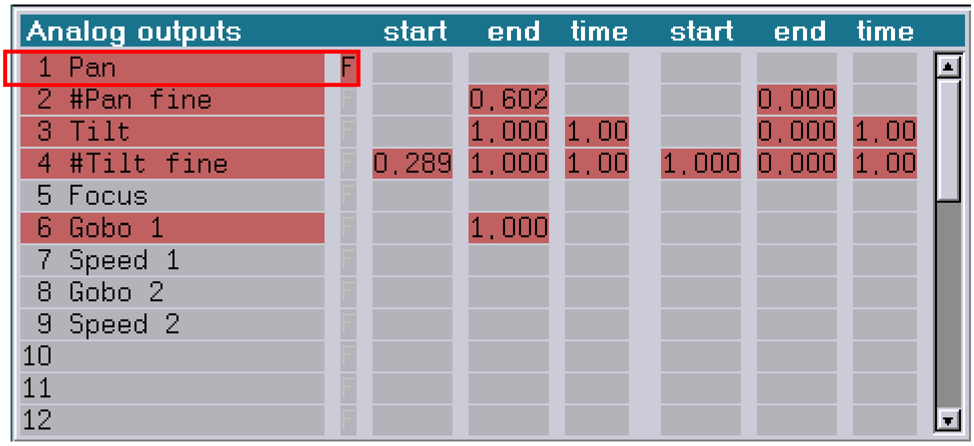
Above this window you will find the channel selection.

In the Lasergraph DSP the 256* Analog outputs are divided into eight channels with 32 virtual outputs to make the access to separate devices easier.
Virtual output means that these outputs are mapped to the 256 physical outputs inside the MapConfig.

Default mapping:
Channel 1 includes virtual output 1 – 32 → physical output 1 – 32
Channel 2 includes virtual output 33 – 64 → physical output 33 – 64
Channel 3 includes virtual output 65 – 96 → physical output 65 – 96
Channel 4 includes virtual output 97 – 128 → physical output 97 – 128
Channel 5 includes virtual output 129 – 160 → physical output 129 – 160
Channel 6 includes virtual output 161 – 192 → physical output 161 – 192
Channel 7 includes virtual output 193 – 224 → physical output 193 – 224
Channel 8 includes virtual output 225 – 256 → physical output 225 – 256

** to increase the update frequency of the DMX protocol, the DSP Output only supports 256 DMX channels.*
This doubles the refresh rate of the DMX signal and allows a much faster response of e.g. DMX controlled optical benches.

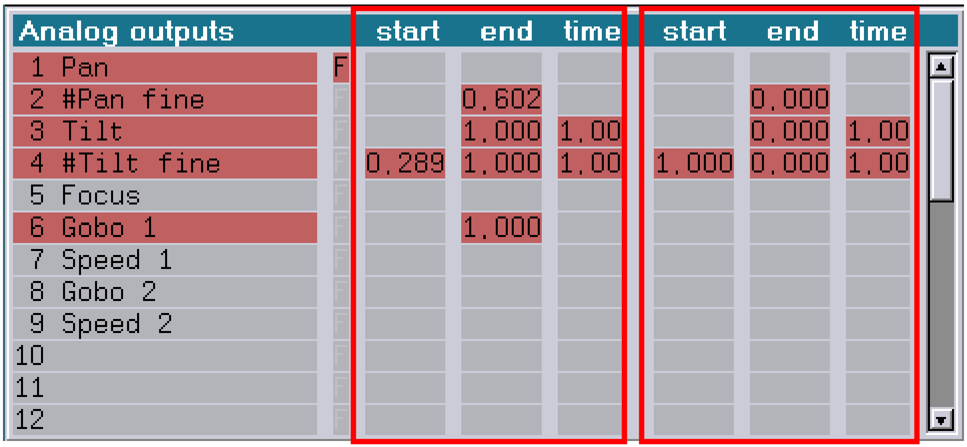
Defining values



Analog outputs		start	end	time	start	end	time
1 Pan	F						
2 #Pan fine	F		0,602			0,000	
3 Tilt	F		1,000	1,00		0,000	1,00
4 #Tilt fine	F	0,289	1,000	1,00	1,000	0,000	1,00
5 Focus	F						
6 Gobo 1	F		1,000				
7 Speed 1	F						
8 Gobo 2	F						
9 Speed 2	F						
10	F						
11	F						
12	F						

Output

By clicking on the output field you select an output and set it to 100 % (FLASH).
This Flash function can be changed by clicking on the "F" button (FLASH).



Analog outputs		start	end	time	start	end	time
1 Pan	F						
2 #Pan fine	F		0,602			0,000	
3 Tilt	F		1,000	1,00		0,000	1,00
4 #Tilt fine	F	0,289	1,000	1,00	1,000	0,000	1,00
5 Focus	F						
6 Gobo 1	F		1,000				
7 Speed 1	F						
8 Gobo 2	F						
9 Speed 2	F						
10	F						
11	F						
12	F						

press key action release key action

start – end – time → press key

By clicking on one of these three fields you assign an Analog value to its output. This can by done by using the Mouse Poti mode or by typing a value into the command line.

Start → This is the start value when you press the Midi key.
End → This is the end value when you press the Midi key.
Time → This is the time it takes for the fade from start to end.

When no start value and no time is specified (parameters are empty), then the output will instantly be the assigned value.

start – end – time → release key

By clicking onto one of these three fields you assign an Analog value to its output.

Start → This is the start value when you release the Midi key.
End → This is the end value when you release the Midi key.
Time → This is the time it takes for the fade from start to end.

When no start value and no time is specified (parameters are empty), then the output will instantly be the assigned value.

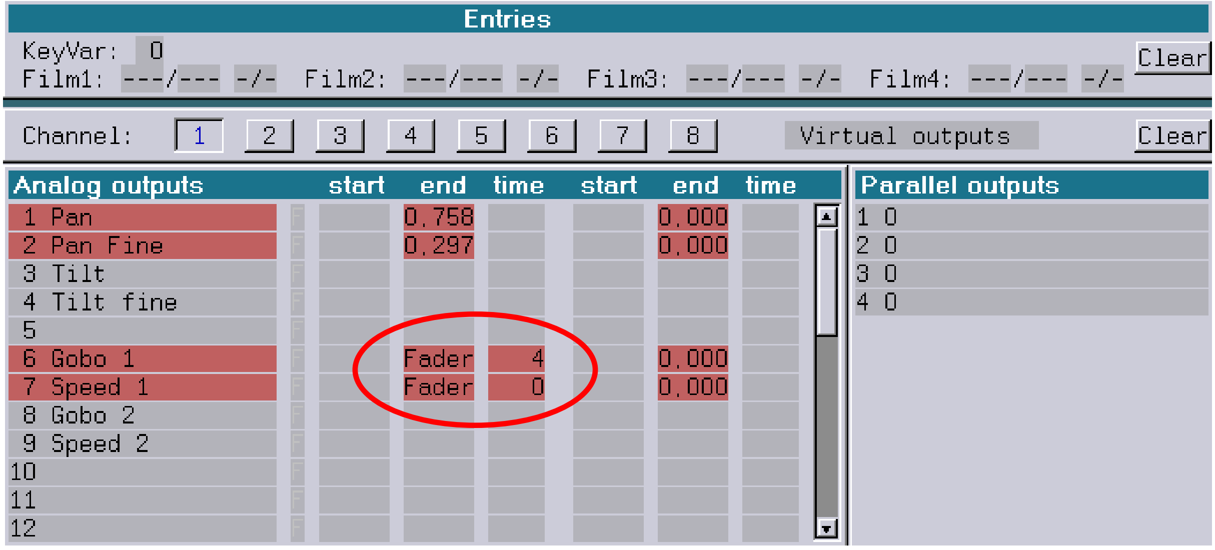
In the example above, **pressing** the corresponding Midi key will do the following:

Output 1 (Pan) will be set to 100%
Output 2 (#Pan fine) will be set to 60%
Output 3 (Tilt) will fade from its current value to 100% in 1 second
Output 4 (#Tilt fine) will fade from 29% to 100% in 1 second
Output 6 (Gobo 1) will be set to 100%

In the example above, **releasing** the corresponding Midi key will do the following:

Output 1 (Pan) will be set to 0%
Output 2 (#Pan fine) will be set to 0%
Output 3 (Tilt) will fade from its current value to 0% in 1 second
Output 4 (#Tilt fine) will fade from 100% to 0% in 1 second
Output 6 (Gobo 1) stays at 100% until it is overwritten by another key

Assigning a Fader to an Analog output



Entries

KeyVar: 0 Clear

Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

Channel: 1 2 3 4 5 6 7 8 Virtual outputs Clear

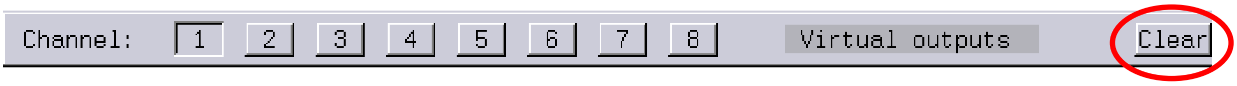
Analog outputs	start	end	time	start	end	time
1 Pan		0,758			0,000	
2 Pan Fine		0,297			0,000	
3 Tilt						
4 Tilt fine						
5						
6 Gobo 1		Fader	4		0,000	
7 Speed 1		Fader	0		0,000	
8 Gobo 2						
9 Speed 2						
10						
11						
12						

Parallel outputs
1 0
2 0
3 0
4 0

There is also the option to assign a Fader to an Analog output.
This is done by clicking into the "end" field of the output and then by choosing the option "Fader" and a Fader number from the menu.
Now the output will be controlled by moving the Fader (either on screen, on the Faderboard or by a Midi controller).

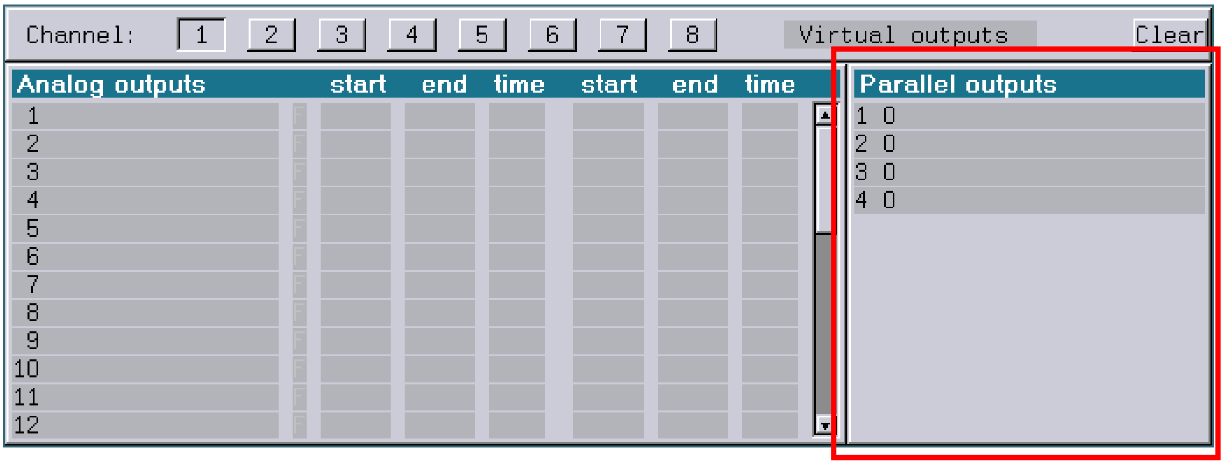
Deleting settings

The use of the button "Clear" will erase all settings made for the selected key.



Channel: 1 2 3 4 5 6 7 8 Virtual outputs Clear

II. Parallel outputs



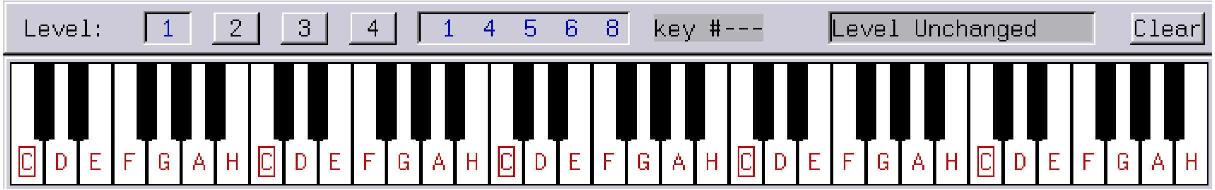
Analog outputs		start	end	time	start	end	time
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							

Parallel outputs	
1	0
2	0
3	0
4	0

The Parallel outputs are used with a special peripheral card.
This card is connected to the Midi out of the Lasergraph DSP.

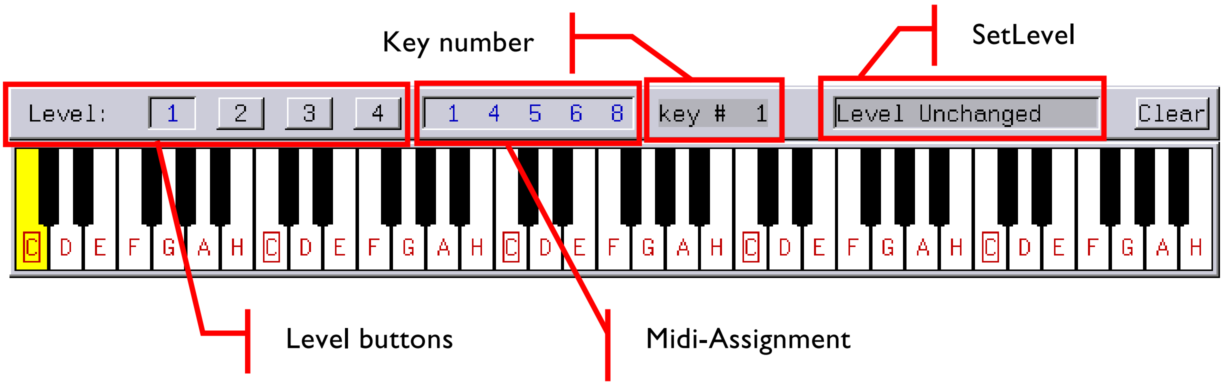
Each channel provides access to four parallel outputs with 8-bit (0-255) each.

III. Virtual MIDI keyboard / Level assignment


Level:
1
2
3
4
1 4 5 6 8
key #---
Level Unchanged
Clear
C D E F G A H C D E F G A H C D E F G A H C D E F G A H C D E F G A H

The Midi keyboard supports five octaves (60 keys).
To expand the number of available keys, you can use the same key on four different levels. In addition you can control the different levels with different Midi-channels.
Every key can be assigned to trigger an Entry and/or set an Analog and/or Parallel output. The level can by switched by keys, too.
When a key is selected, it will be highlighted (yellow color).

Every key can be identified by a number (e.g. key #1).

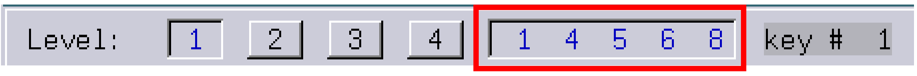


Level Assignment

To switch between levels, click on the Level buttons or assign a Midi key to switch between levels. This is done by clicking into the "SetLevel field, or by using the menu command "SetLevel".

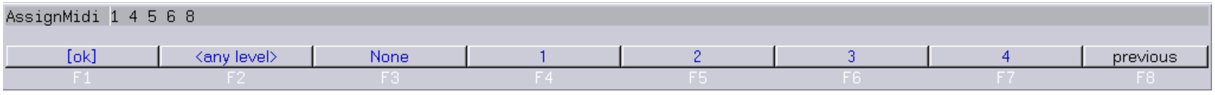
Midi Assignment

The Midi assignment is displayed to the right of the Level buttons.



By clicking into the field you can edit the given assignment.

The command line will show the following:



```
AssignMidi 1 4 5 6 8
```

In our example you see the numbers "1 4 5 6 8".

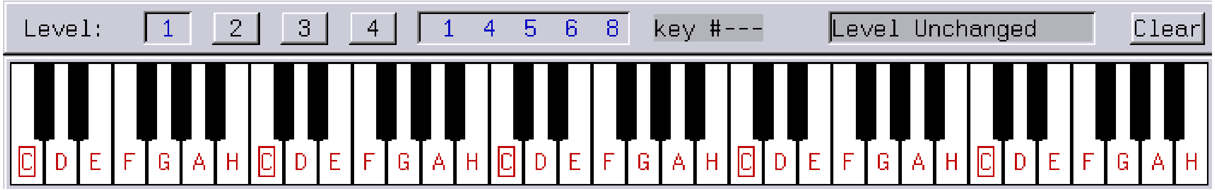
These numbers stand for the Midi channel.

The first number defines the Midi channel that controls the currently active Level (<any level>).

The second number defines the Midi channel that controls Level 1 regardless which Level is currently active (<level 1>).
The third number defines the Midi channel that controls Level 2 regardless which Level is currently active (<level 2>).
The fourth number defines the Midi channel that controls Level 3 regardless which Level is currently active (<level 3>).
The fifth number defines the Midi channel that controls Level 4 regardless which Level is currently active (<level 4>).

This means for our example, that a Midi keyboard sending on Midi channel 1 can control any currently active Level. A keyboard that sends on Midi channel 4 controls Level 1 only, Midi channel 5 controls Level 2, Midi channel 6 controls Level 3 and Midi channel 8 controls Level 4.

Deleting settings



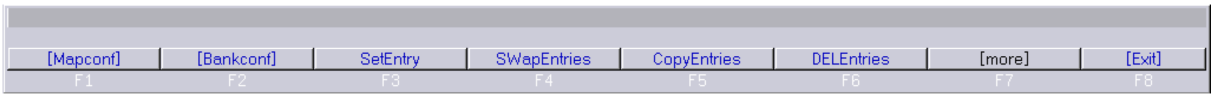
Level:
1
2
3
4
1 4 5 6 8
key #---
Level Unchanged
Clear
C D E F G A H C D E F G A H C D E F G A H C D E F G A H C D E F G A H

The use of the button "Clear" will erase all settings made for the selected key.

The menu bar

All described procedures can also be executed by using the menu bar and the function keys.

page 1



[Mapconf]
Switches into the Map configuration section.

[Bankconf]
Switches into the Bank configuration section.

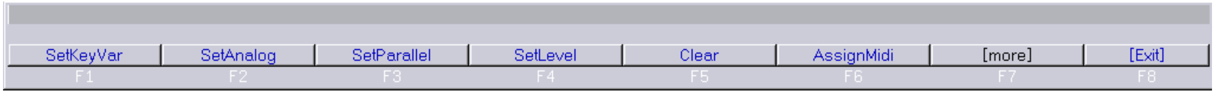
SetEntry
Command: SetEntry
Command to assign an Entry to a Midi key.
Syntax:
SetEntry Film(1-4) press entry (0-999, Next, Prev, Abort or None) release entry (0-999, Next, Prev, Abort or None)

SWapEntries
Command to swap all Entries of Film (x) to Film (y).
Command: SWapEntries
Syntax:
SWapEntries Film(1-4) Film(1-4)

CopyEntries
Command to copy all Entries of Film (x) to Film (y).
Command: CopyEntries
Option: /Overwrite
Syntax:
CopyEntries Film(1-4) Film(1-4) /Overwrite

DELEntries
Command to delete all Entries of Film (x).
Command: DELEntries
Option: /Overwrite
Syntax:
DELEntries Film(1-4) /Overwrite

page 2



SetKeyVar

Command to assign a key variable to a Midi key.
Command: SetKeyVar
Syntax:
SetKeyVar Value (0-63)

SetAnalog

Command to assign a value to an analog output.
Command: SetAnalog
Subcommands: Press.Start / Press.End / Press.Time / Release.Start / Release.End / Release.Time
Syntax:
SetAnalog.Press.Start Number (1-32) Value (0.0-1.0, '-', 'Off' or 'Fader')
SetAnalog.Press.End Number (1-32) Value (0.0-1.0, 'Off' or 'Fader')
SetAnalog.Press.Time Number (1-32) Time (0.0-10.2 or '-')
SetAnalog.Release.Start Number (1-32) Value (0.0-1.0, '-', 'Off' or 'Fader')
SetAnalog.Release.End Number (1-32) Value (0.0-1.0, 'Off' or 'Fader')
SetAnalog.Release.Time Number (1-32) Time (0.0-10.2 or '-')

SetParallel

Command to define the parallel output.
Command: SetParallel
Syntax:
SetParallel Number (1-4) Value (0-255 or 'Off')

SetLevel

Command to assign a level change to a Midi key.
Command: SetLevel
Syntax:
SetLevel Level (1-4 or 'Unchanged')

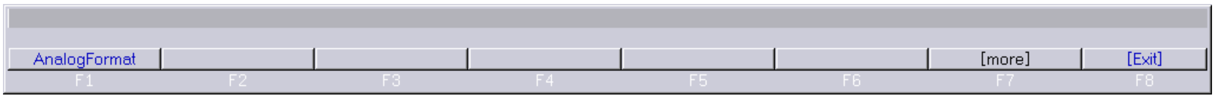
Clear

Command to delete all assignments from one or all keys.
Command: Clear
Subcommands: Key.Entries / Key.Channel / Key.ALL / ALL
Syntax:
Clear.Key.Entries
Clear.Key.Channel.Analogs
Clear.Key.Channel.Parallels
Clear.Key.Channel.ALL
Clear.Key.ALL
Clear.ALL

AssignMidi

Command to assign the Midi channels to the Levels.

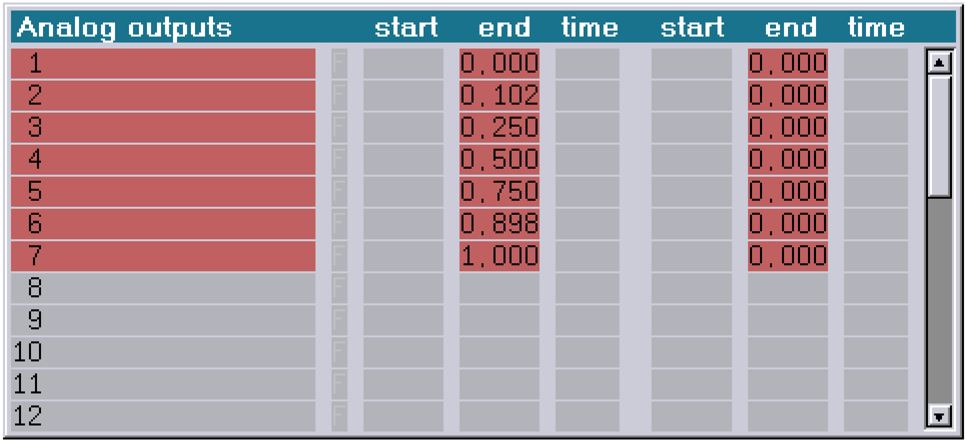
page 3



AnalogFormat

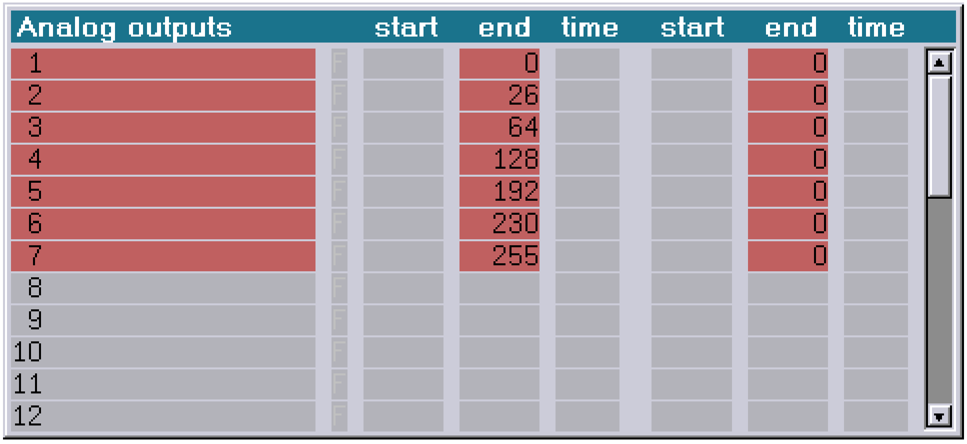
Command to switch between different analog formats.
Command: AnalogFormat
Syntax:
AnalogFormat Format ('Fractional', 'Decimal', 'Hexadecimal' or 'Percent')

‘Fractional’ Value Range 0.000 – 1.000 (Default)



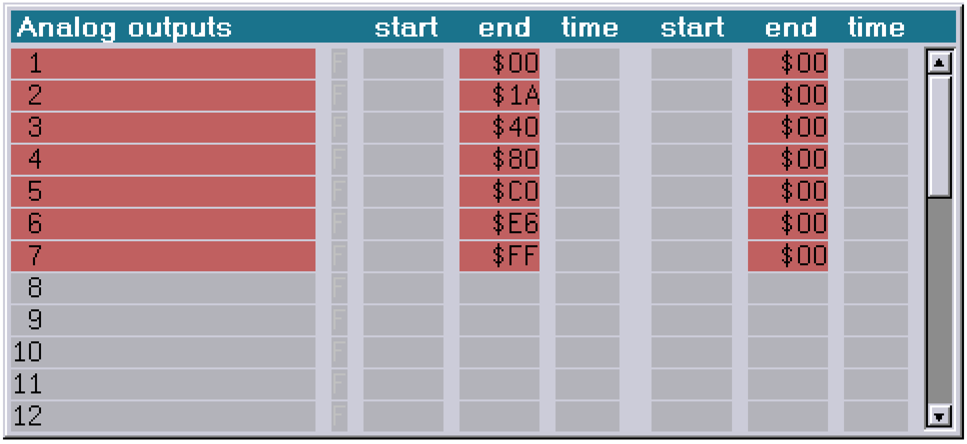
Analog outputs		start	end	time	start	end	time
1	F		0,000			0,000	
2	F		0,102			0,000	
3	F		0,250			0,000	
4	F		0,500			0,000	
5	F		0,750			0,000	
6	F		0,898			0,000	
7	F		1,000			0,000	
8	F						
9	F						
10	F						
11	F						
12	F						

‘Decimal’ Value Range 0 – 255



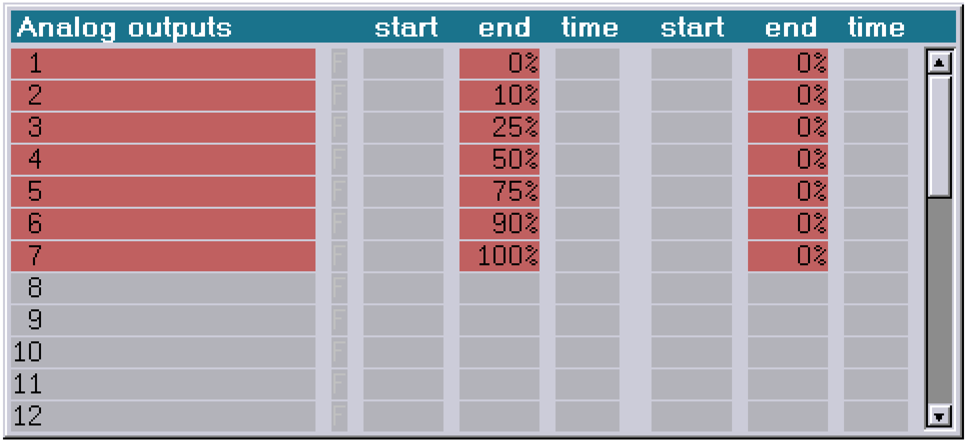
Analog outputs		start	end	time	start	end	time
1	F		0			0	
2	F		26			0	
3	F		64			0	
4	F		128			0	
5	F		192			0	
6	F		230			0	
7	F		255			0	
8	F						
9	F						
10	F						
11	F						
12	F						

‘Hexadecimal’ Value Range $00 - $FF



Analog outputs		start	end	time	start	end	time
1	F		$00			$00	
2	F		$1A			$00	
3	F		$40			$00	
4	F		$80			$00	
5	F		$C0			$00	
6	F		$E6			$00	
7	F		$FF			$00	
8	F						
9	F						
10	F						
11	F						
12	F						

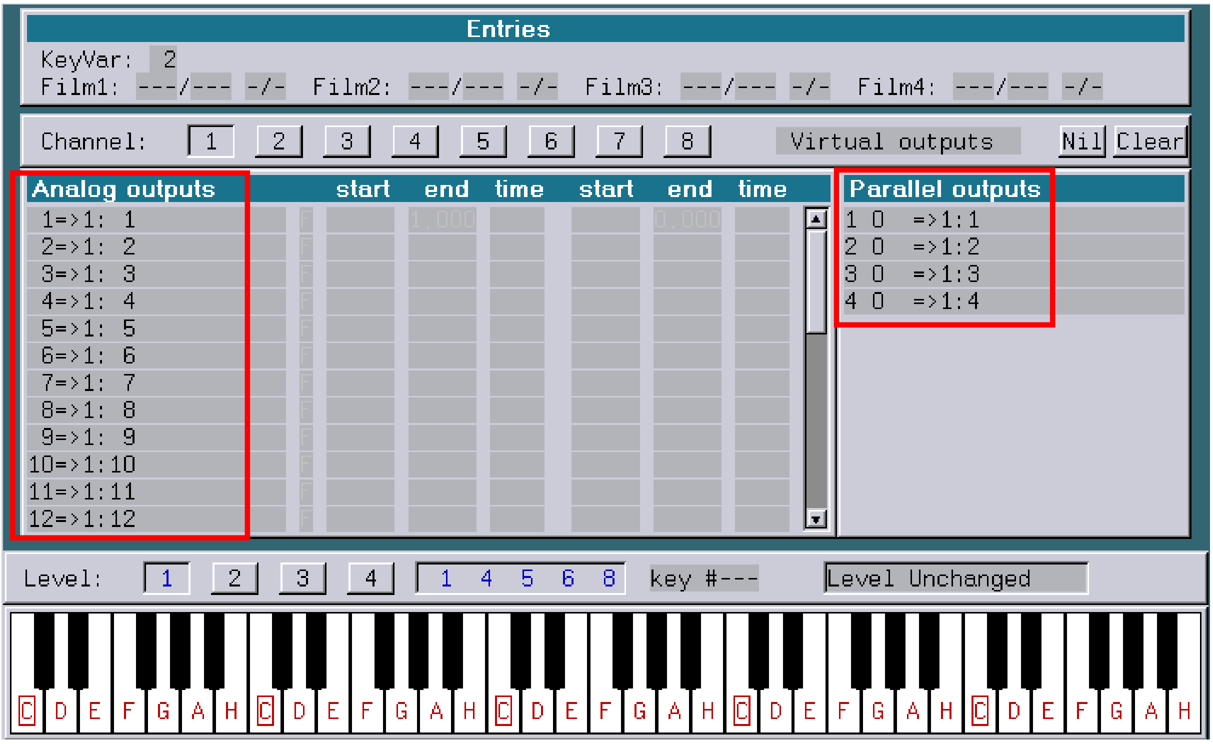
‘Percent’ Value Range 0% - 100%



Analog outputs		start	end	time	start	end	time
1	F		0%			0%	
2	F		10%			0%	
3	F		25%			0%	
4	F		50%			0%	
5	F		75%			0%	
6	F		90%			0%	
7	F		100%			0%	
8	F						
9	F						
10	F						
11	F						
12	F						

Note:
Irrespective of the set value format the hexadecimal input can be forced by a preceding '$' and percentage values by a closing '%'.

Mapconfig



Entries

KeyVar: 2

Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

Channel: 1 2 3 4 5 6 7 8 Virtual outputs

```
Analog outputs      start  end    time  start  end    time    Parallel outputs
  1=>1:  1          F      1,000               0,000          1 0  =>1:1
  2=>1:  2          F                                         2 0  =>1:2
  3=>1:  3          F                                         3 0  =>1:3
  4=>1:  4          F                                         4 0  =>1:4
  5=>1:  5          F
  6=>1:  6          F
  7=>1:  7          F
  8=>1:  8          F
  9=>1:  9          F
 10=>1:10           F
 11=>1:11           F
 12=>1:12           F
```

Level: 1 2 3 4 1 4 5 6 8 key #--- Level Unchanged

Nil
Clear
C D E F G A H C D E F G A H C D E F G A H C D E F G A H C D E F G A H C D E F G A H

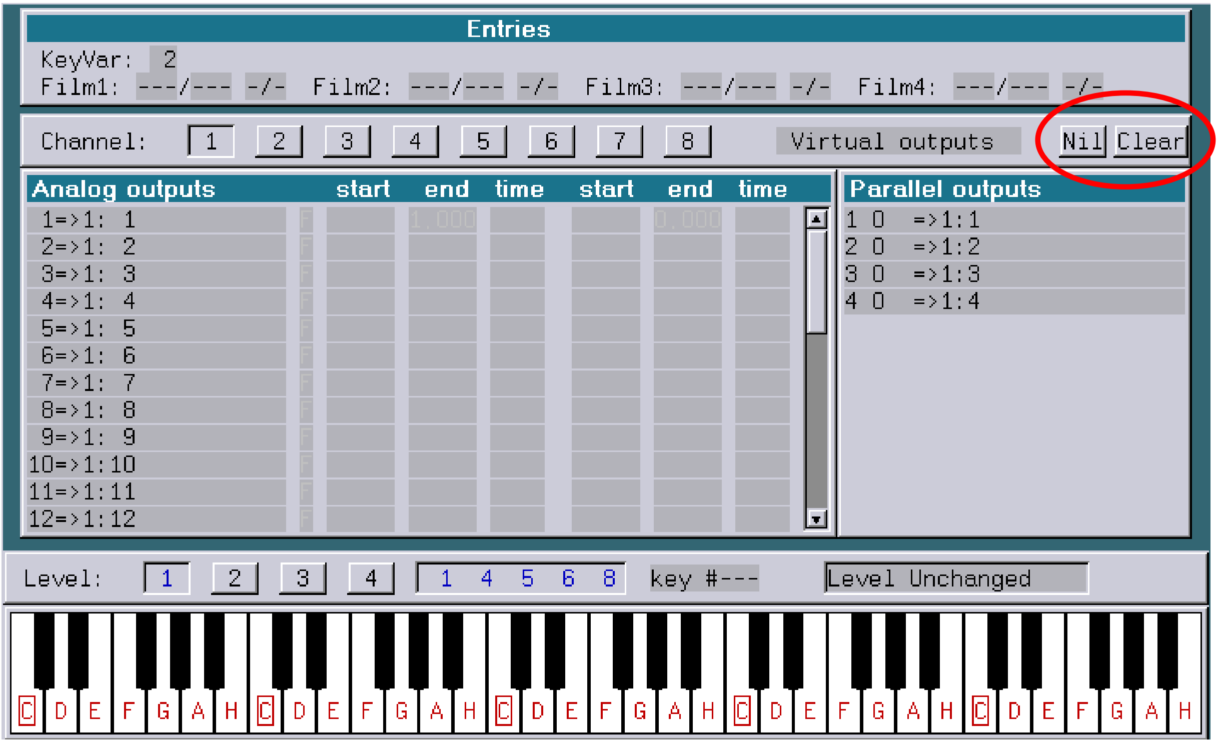
In the Mapconfig the mapping of the analog outputs (DMX) is done.

What is "mapping"?

The Mapconfig assigns the virtual outputs to the physical outputs.
The default mapping is 1:1, which means the virtual output is routed one to one to the physical output (e.g. analog output 5 is the physical output 5 (DMX).

In the screenshot above you can see display of the Analog outputs and the parallel outputs have changed.

The description
"1 => 1:1"
means that virtual output 1 is routed to the physical channel 1 output 1.
"12 => 1:12"
means that virtual output 12 is routed to the physical channel 1 output 12.



KeyVar: 2
Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

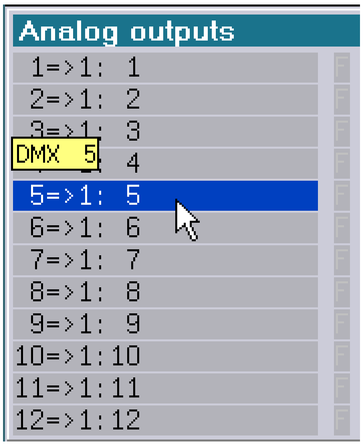
```
1=>1: 1
2=>1: 2
3=>1: 3
4=>1: 4
5=>1: 5
6=>1: 6
7=>1: 7
8=>1: 8
9=>1: 9
10=>1:10
11=>1:11
12=>1:12
```

```
1 0  =>1:1
2 0  =>1:2
3 0  =>1:3
4 0  =>1:4
```

key #---
Level Unchanged

By clicking on the button "Nil" you can set the current channel to be disconnected.

The use of the button "Clear" will reset the current channel (1:1).

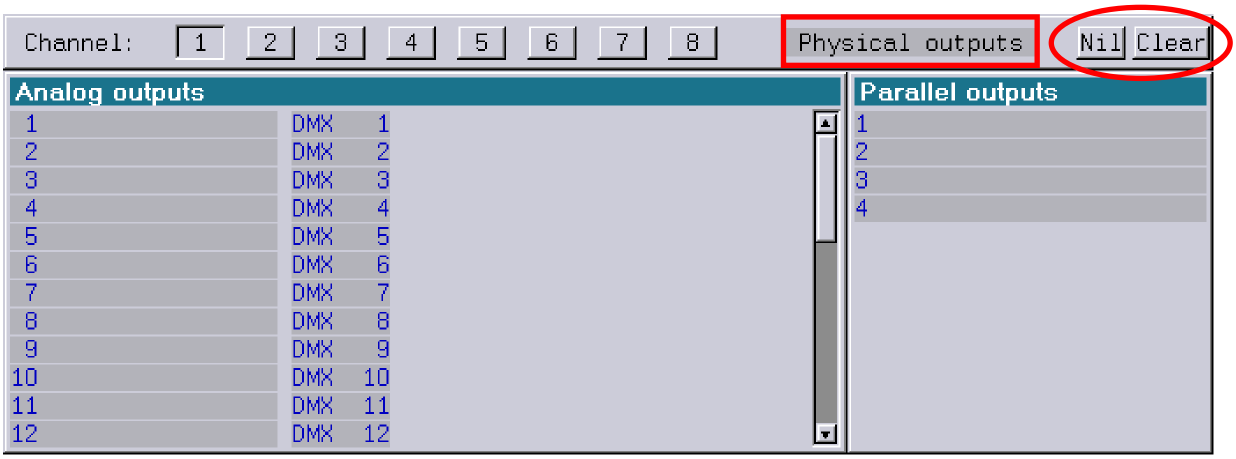


```
1=>1: 1
2=>1: 2
3=>1: 3
4=>1: 4
5=>1: 5
6=>1: 6
7=>1: 7
8=>1: 8
9=>1: 9
10=>1:10
11=>1:11
12=>1:12
```

When the mouse pointer is over an output row, the popup window will show you the physical DMX output channel.

Assignment of outputs

By clicking into a virtual output field, the display will change to the physical output window (see screenshot below).



Analog outputs

1	DMX 1
2	DMX 2
3	DMX 3
4	DMX 4
5	DMX 5
6	DMX 6
7	DMX 7
8	DMX 8
9	DMX 9
10	DMX 10
11	DMX 11
12	DMX 12

Parallel outputs

1
2
3
4

Now you can assign the virtual output to one of the 256 available physical outputs. Once you have selected the physical output, the display will change back to the virtual output window.

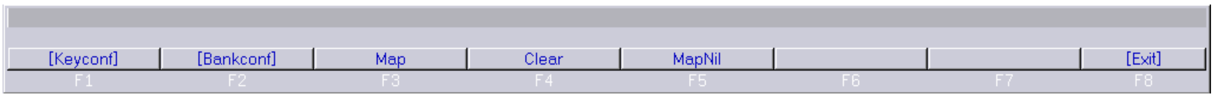
Deleting settings

The use of the button "Clear" will reset the mapping of the current channel (1:1).

The button "Nil" will set all outputs of the current channel to be disconnected.

The menu bar

All described procedures can also be executed by using the menu bar and the function keys.



[Keyconf]

Switches into the Key configuration section.

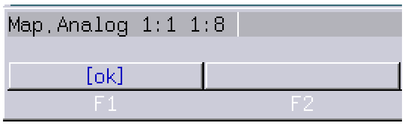
[Bankconf]

Switches into the Bank configuration section.

Map

Command to assign the virtual output to a physical output.
With the subcommand you decide whether to map an Analog or a Parallel output.

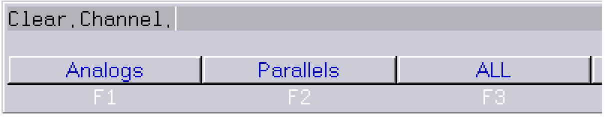
Command: Map
Subcommands: Analog / Parallel
Syntax:
Map.Analog Channel (1-8) : output (1-32) Channel (1-8) : output (1-32)
Map.Parallel Channel (1-8) : output (1-4) Channel (1-8) : output (1-4)


Map.Analog 1:1 1:8

In the example screenshot above, the virtual Analog output 1 of Channel 1 will be mapped to output 8 of channel 1.

Clear

Command to reset a channel or all channels to the default 1:1 mapping.

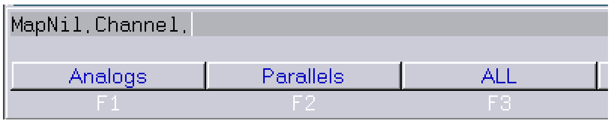

Clear.Channel.

You can choose whether you want to reset all Analog or Parallel outputs in a single channel or all outputs of all channels.

Command: Clear
Subcommands: Channel.Analogs / Channel.Parallels / Channel.ALL / ALL
Syntax:
Clear.Channel.Analogs
Clear.Channel.Parallels
Clear.Channel.ALL
Clear.ALL

MapNil

Command to set one or all channels to be undefined.

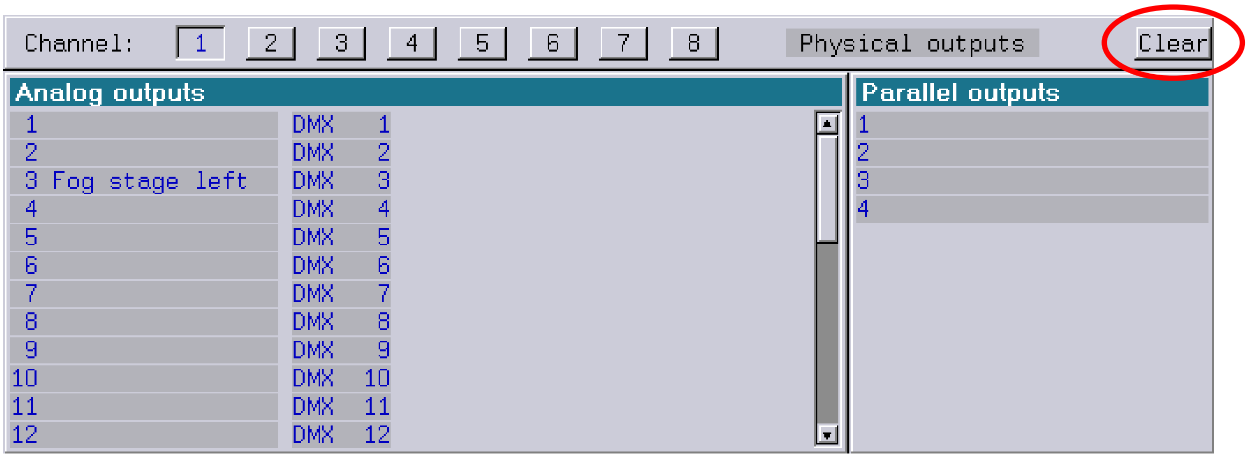


MapNil.Channel.

You can choose whether you want to disconnect all Analog or Parallel outputs in a single channel or all outputs of all channels.

Command: MapNil
Subcommands: Channel.Analogs / Channel.Parallels / Channel.ALL / ALL
Syntax:
MapNil.Channel.Analogs
MapNil.Channel.Parallels
MapNil.Channel.ALL
MapNil.ALL

Bankconfig



Analog outputs

1	DMX	1
2	DMX	2
3 Fog stage left	DMX	3
4	DMX	4
5	DMX	5
6	DMX	6
7	DMX	7
8	DMX	8
9	DMX	9
10	DMX	10
11	DMX	11
12	DMX	12

Parallel outputs

1
2
3
4

Labeling of outputs

The Bankconfig (comes from the name "Bankbox") is used to give DMX output channels a dedicated name.

To do so, just click on an output field and type in a name for that output and finish the input with <Return>.

Note that you are labeling the physical outputs!!

Deleting settings

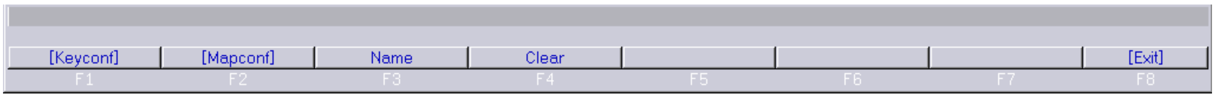
The use of the button "Clear" will delete all output names.

You can choose whether you want to delete all Analog or Parallel names in a single channel or all names of all channels.

Command: Clear
Subcommands: Channel.Analogs / Channel.Parallels / Channel.ALL / ALL
Syntax:
Clear.Channel.Analogs
Clear.Channel.Parallels
Clear.Channel.ALL
Clear.ALL

The menu bar

All described procedures can also be executed by using the menu bar and the function keys.



[Keyconf]

Switches into the Key configuration section.

[Mapconf]

Switches into the Map configuration section.

Name

You can specify a name for an "Analog" or a "Parallel" output.

Command: Name
Subcommands: Analog / Parallel
Syntax:
Name.Analog Number (1-32) Name
Name.Parallel Number (1-4) Name

Clear

Command to delete specified names.

Command: Clear
Subcommands: Channel.Analogs / Channel.Parallels / Channel.ALL / ALL
Syntax:
Clear.Channel.Analogs
Clear.Channel.Parallels
Clear.Channel.ALL
Clear.ALL

Tips

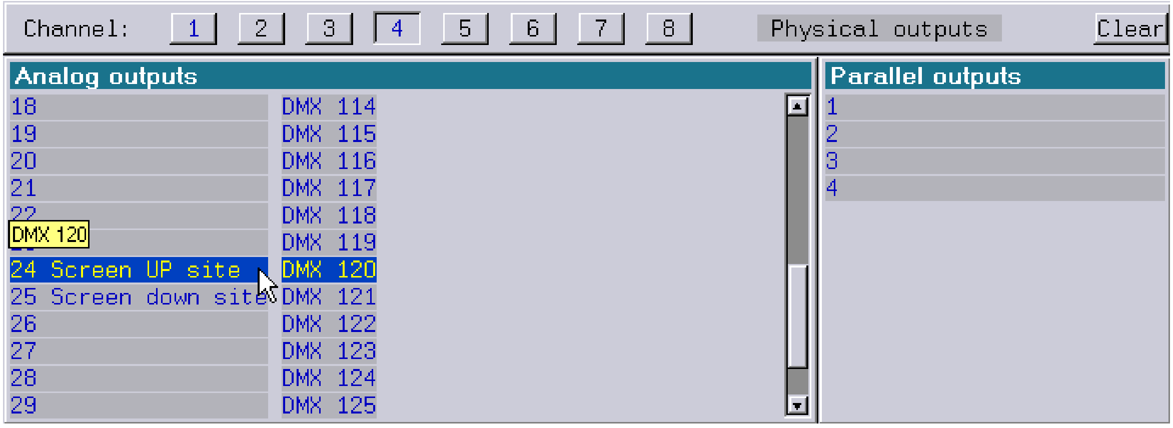
Workflow

Always start with labeling the physical outputs, then, if necessary, change the mapping of the virtual outputs.

Here is an example:

Let us assume you programmed a show which uses two analog outputs (DMX) to control the motorized screen. One channel for "screen up" and one for "screen down".
In your programming environment these two channels are, for example, 10 and 11.

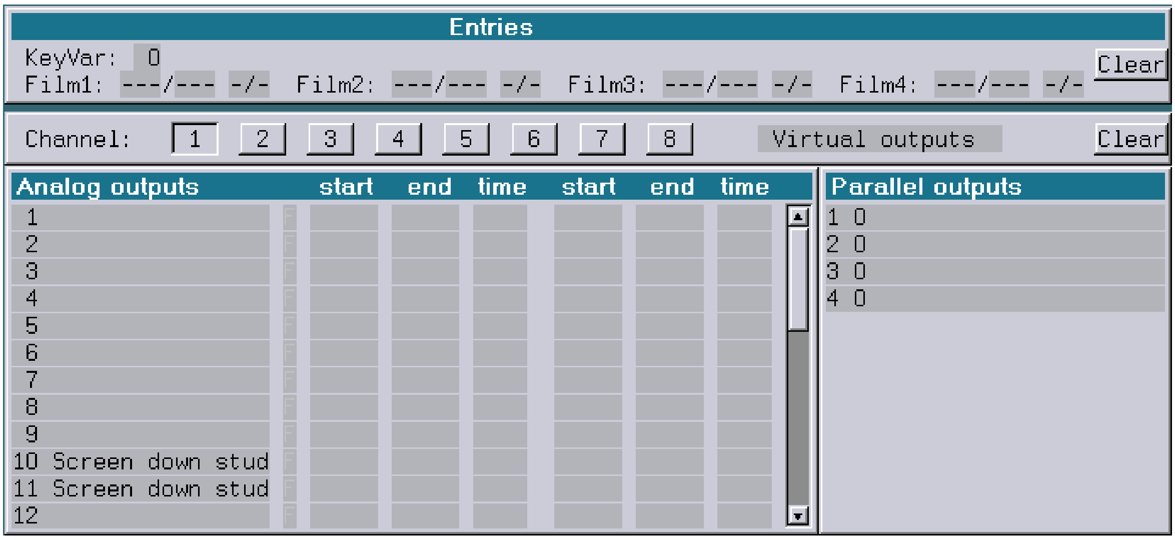
Now, being on the show site, the screen used has the DMX address 120 and 121.
Instead of reprogramming the Film and /or Timescript, it is much easier to change the mapping.
First go to the BankConfig and give output 120 and 121 a name.
You find DMX address 120 in channel 4, output 24.



Analog outputs	
18	DMX 114
19	DMX 115
20	DMX 116
21	DMX 117
22	DMX 118
	DMX 119
24 Screen UP site	DMX 120
25 Screen down site	DMX 121
26	DMX 122
27	DMX 123
28	DMX 124
29	DMX 125

Parallel outputs: 1, 2, 3, 4

Entering the Keyconfig, you will see the old assignment of channel 10/11.



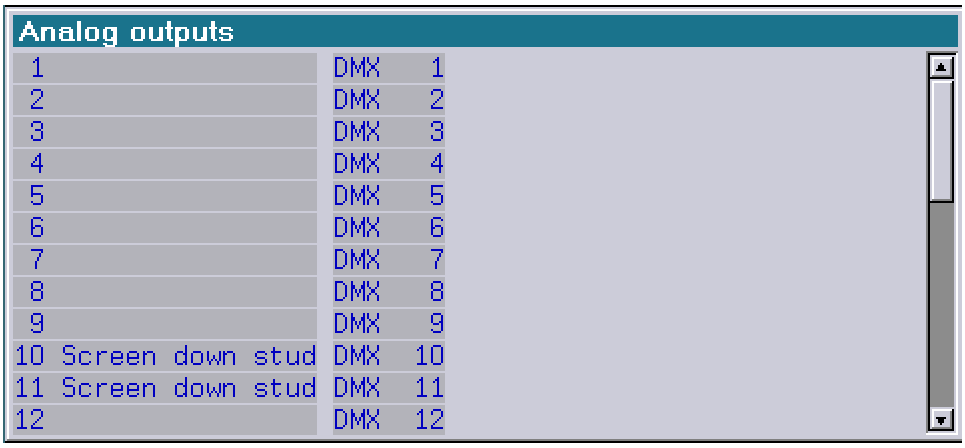
KeyVar: 0
Film1: ---/--- -/- Film2: ---/--- -/- Film3: ---/--- -/- Film4: ---/--- -/-

Analog outputs	start	end	time	start	end	time
1						
2						
3						
4						
5						
6						
7						
8						
9						
10 Screen down stud						
11 Screen down stud						
12						

Parallel outputs	
1	0
2	0
3	0
4	0

Now switch into the MapConfig and click onto (virtual) channel 10.

The display will change to the physical outputs.

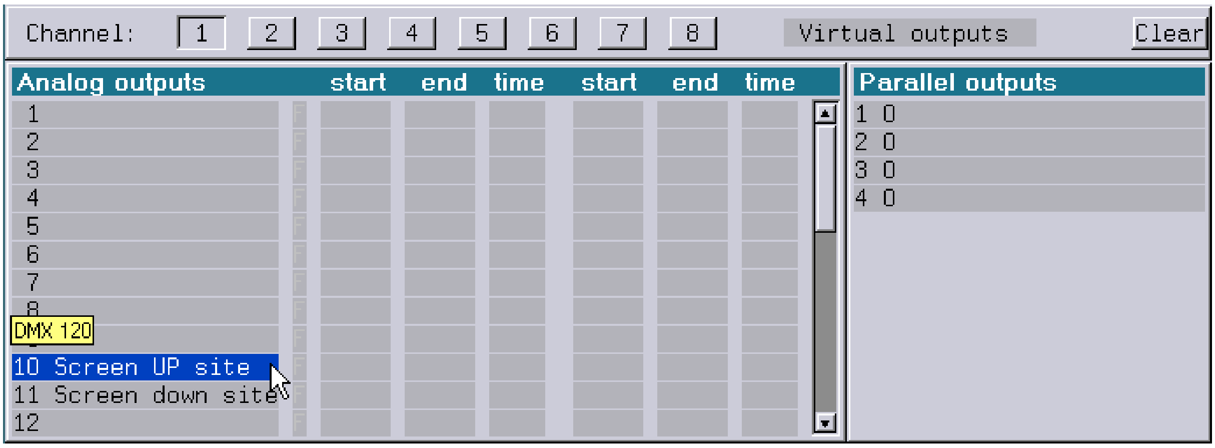


Analog outputs

1	DMX 1
2	DMX 2
3	DMX 3
4	DMX 4
5	DMX 5
6	DMX 6
7	DMX 7
8	DMX 8
9	DMX 9
10 Screen down stud	DMX 10
11 Screen down stud	DMX 11
12	DMX 12

As you can see, output 10 is routed to the physical output 10 (DMX address 10). By changing into channel 4 and scrolling the window down to output 24, you will find the new DMX channel 120.
Just click on it and the virtual output 10 is now routed to the physical output 120.
Do the same with the output 11 and you are done.

Now, being back in the Keyconfig you can see that output 10 is now routed to DMX address 120.



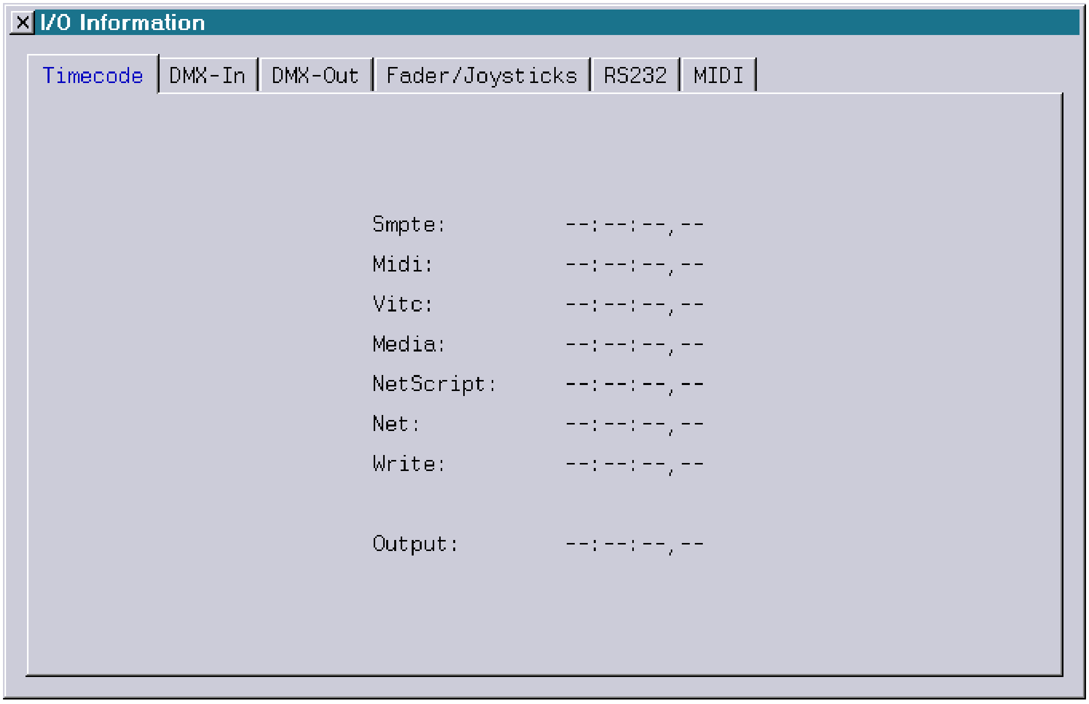
Analog outputs	start	end	time	start	end	time
1						
2						
3						
4						
5						
6						
7						
8						
DMX 120						
10 Screen UP site						
11 Screen down site						
12						

Parallel outputs

1	0
2	0
3	0
4	0

So, mapping makes it easy to adjust an existing programming to a different DMX setting on site.

Chapter I/O Information



I/O Information

```
Smpte:        --:--:--,--
Midi:         --:--:--,--
Vitc:         --:--:--,--
Media:        --:--:--,--
NetScript:    --:--:--,--
Net:          --:--:--,--
Write:        --:--:--,--

Output:       --:--:--,--
```

The I/O Information is a useful tool to check and visualize most of the incoming and outgoing data of the Lasergraph DSP.

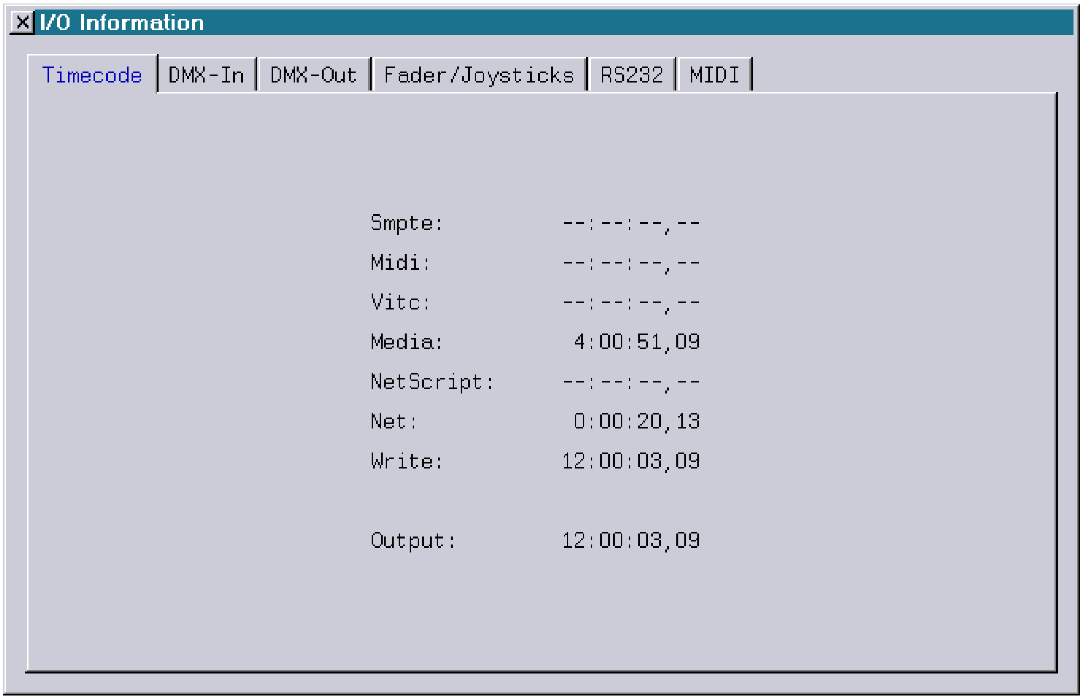
The I/O Info window is opened with the command "IOINFO" in the command line.
The command can also be found in the main menu bar.

When it opens you will see the Timecode Information tab.

Over all, the I/O Info consists of 6 different tabs:

- Timecode
- DMX-In
- DMX-Out
- Fader/Joysticks
- RS232
- MIDI

Timecode



```
Smpte:        --:--:--,--
Midi:         --:--:--,--
Vitc:         --:--:--,--
Media:         4:00:51,09
NetScript:    --:--:--,--
Net:           0:00:20,13
Write:        12:00:03,09

Output:       12:00:03,09
```

The Timecode tab shows all incoming and outgoing timecodes, either from the physical inputs or from the I/O server assigned in the "Net I/O" .

Smpte: displays the incoming Smpte timecode (LTC)

Midi: displays the incoming Midi timecode (MTC)

Vitc: displays the incoming Video timecode (VITC)

Media: displays the incoming timecode from the Media player

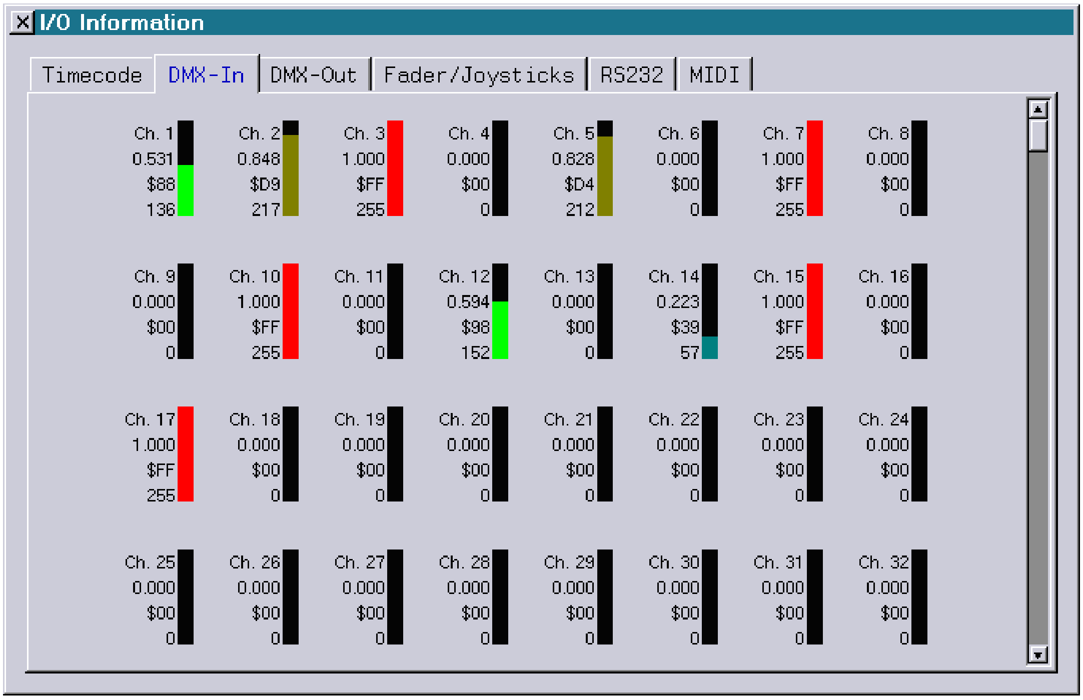
NetScript: displays the incoming script timecode that is sent by another Lasergraph DSP defined as server in the I/O-Setup (TCP)

Net: displays the incoming script timecode that is sent by another Lasergraph DSP with "Write" enabled (UDP broadcast)

Write: displays the outgoing timecode when write is enabled

Output: displays the outgoing timecode of the physical output (LTC)

DMX-In

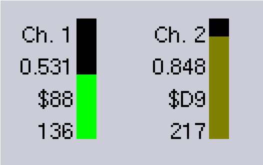


```
Ch. 1   Ch. 2   Ch. 3   Ch. 4   Ch. 5   Ch. 6   Ch. 7   Ch. 8
0.531   0.848   1.000   0.000   0.828   0.000   1.000   0.000
$88     $D9     $FF     $00     $D4     $00     $FF     $00
136     217     255     0       212     0       255     0

Ch. 9   Ch. 10  Ch. 11  Ch. 12  Ch. 13  Ch. 14  Ch. 15  Ch. 16
0.000   1.000   0.000   0.594   0.000   0.223   1.000   0.000
$00     $FF     $00     $98     $00     $39     $FF     $00
0       255     0       152     0       57      255     0

Ch. 17  Ch. 18  Ch. 19  Ch. 20  Ch. 21  Ch. 22  Ch. 23  Ch. 24
1.000   0.000   0.000   0.000   0.000   0.000   0.000   0.000
$FF     $00     $00     $00     $00     $00     $00     $00
255     0       0       0       0       0       0       0

Ch. 25  Ch. 26  Ch. 27  Ch. 28  Ch. 29  Ch. 30  Ch. 31  Ch. 32
0.000   0.000   0.000   0.000   0.000   0.000   0.000   0.000
$00     $00     $00     $00     $00     $00     $00     $00
0       0       0       0       0       0       0       0
```

The DMX-In tab displays the DMX values that are received by the Lasergraph DSP. These values may be received by the physical DMX input or the DMX input of another Lasergraph DSP that is assigned as I/O server or by Art-Net.

Next to the colored line that indicates the value of the channel you see the

- channel number (1–512)
- the DSP value (0.00-1.00)
- the hexadecimal value ($00-$FF)
- the decimal value (0-255)

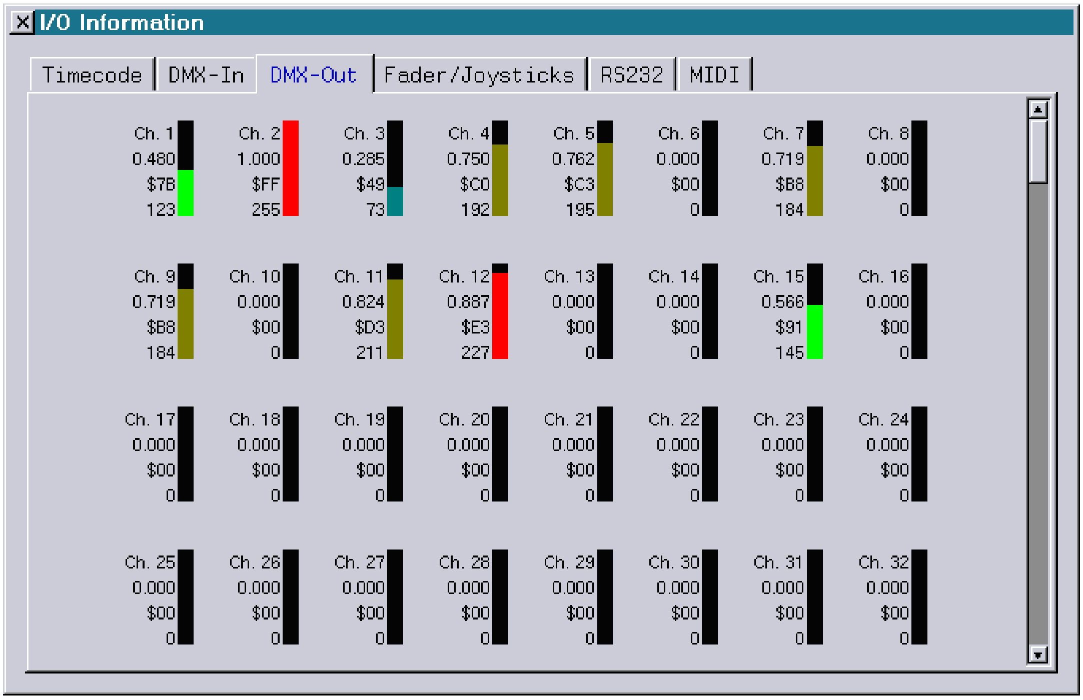


```
Ch. 1   Ch. 2
0.531   0.848
$88     $D9
136     217
```

Note:

The DMX values displayed here are neither affected by the externally adjusted DMX base address nor by the "DMX-In offset" within the Advanced Display Parameters.

DMX-Out

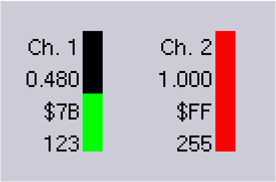


Ch. 1 0.480 $7B 123	Ch. 2 1.000 $FF 255	Ch. 3 0.285 $49 73	Ch. 4 0.750 $C0 192	Ch. 5 0.762 $C3 195	Ch. 6 0.000 $00 0	Ch. 7 0.719 $B8 184	Ch. 8 0.000 $00 0
Ch. 9 0.719 $B8 184	Ch. 10 0.000 $00 0	Ch. 11 0.824 $D3 211	Ch. 12 0.887 $E3 227	Ch. 13 0.000 $00 0	Ch. 14 0.000 $00 0	Ch. 15 0.566 $91 145	Ch. 16 0.000 $00 0
Ch. 17 0.000 $00 0	Ch. 18 0.000 $00 0	Ch. 19 0.000 $00 0	Ch. 20 0.000 $00 0	Ch. 21 0.000 $00 0	Ch. 22 0.000 $00 0	Ch. 23 0.000 $00 0	Ch. 24 0.000 $00 0
Ch. 25 0.000 $00 0	Ch. 26 0.000 $00 0	Ch. 27 0.000 $00 0	Ch. 28 0.000 $00 0	Ch. 29 0.000 $00 0	Ch. 30 0.000 $00 0	Ch. 31 0.000 $00 0	Ch. 32 0.000 $00 0

The DMX-Out tab displays the DMX values that are sent from the physical DMX-output of the Lasergraph DSP.

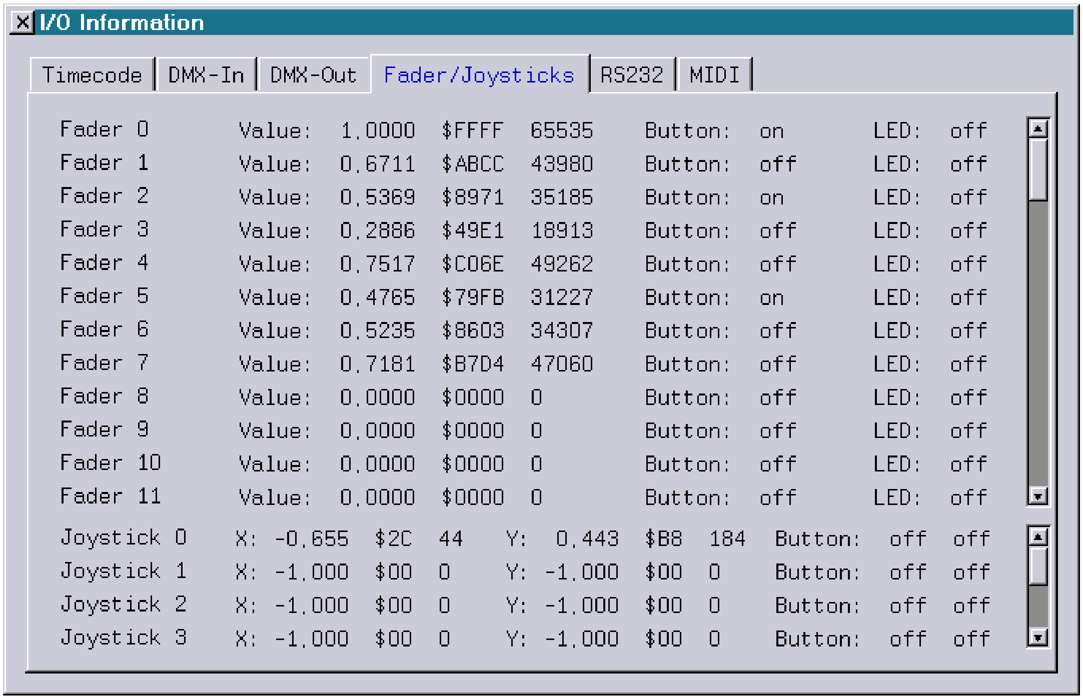
Next to the colored line that indicates the value of the channel you see the

1) channel number (1–256)
2) the DSP value (0.00-1.00)
3) the hexadecimal value ($00-$FF)
4) the decimal value (0-255)



Ch. 1 0.480 $7B 123	Ch. 2 1.000 $FF 255

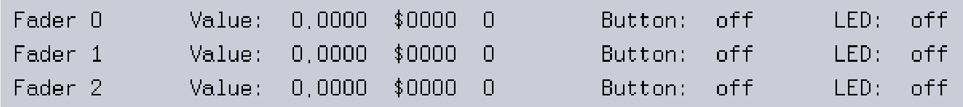
Fader/Joysticks



```
Fader 0      Value:  1.0000  $FFFF  65535    Button:  on       LED:  off
Fader 1      Value:  0.6711  $ABCC  43980    Button:  off      LED:  off
Fader 2      Value:  0.5369  $8971  35185    Button:  on       LED:  off
Fader 3      Value:  0.2886  $49E1  18913    Button:  off      LED:  off
Fader 4      Value:  0.7517  $C06E  49262    Button:  off      LED:  off
Fader 5      Value:  0.4765  $79FB  31227    Button:  on       LED:  off
Fader 6      Value:  0.5235  $8603  34307    Button:  off      LED:  off
Fader 7      Value:  0.7181  $B7D4  47060    Button:  off      LED:  off
Fader 8      Value:  0.0000  $0000  0        Button:  off      LED:  off
Fader 9      Value:  0.0000  $0000  0        Button:  off      LED:  off
Fader 10     Value:  0.0000  $0000  0        Button:  off      LED:  off
Fader 11     Value:  0.0000  $0000  0        Button:  off      LED:  off

Joystick 0   X: -0.655  $2C  44    Y:  0.443  $B8  184   Button:  off  off
Joystick 1   X: -1.000  $00  0     Y: -1.000  $00  0     Button:  off  off
Joystick 2   X: -1.000  $00  0     Y: -1.000  $00  0     Button:  off  off
Joystick 3   X: -1.000  $00  0     Y: -1.000  $00  0     Button:  off  off
```

The Fader/Joysticks tab displays the values resulting from the Faders and Joysticks assigned in the Net I/O.

Fader



```
Fader 0      Value:  0.0000  $0000  0        Button:  off      LED:  off
Fader 1      Value:  0.0000  $0000  0        Button:  off      LED:  off
Fader 2      Value:  0.0000  $0000  0        Button:  off      LED:  off
```

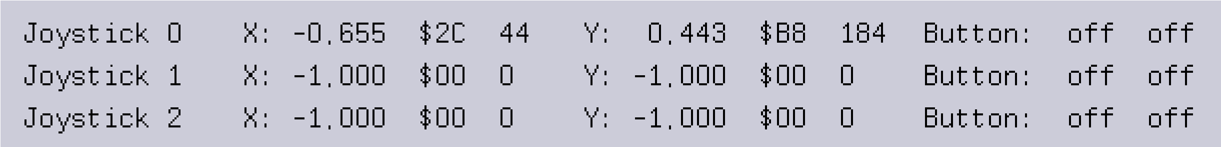
Fader: Fader number (0-63)

Value: DSP value (0.00–1.00)
hexadecimal value ($0000-$FFFF)
decimal value (0-65535)

Button: State of the Fader button (on /off)

LED: State of the LED (on /off)

Joysticks



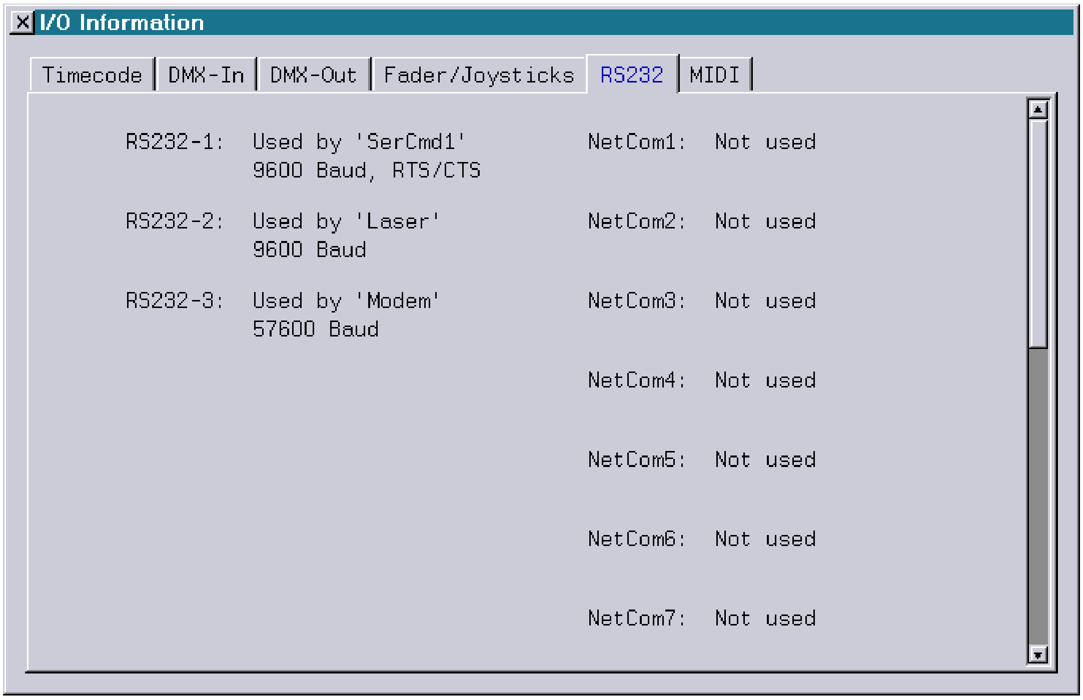
```
Joystick 0   X: -0.655  $2C  44   Y:  0.443  $B8  184  Button:  off  off
Joystick 1   X: -1.000  $00  0    Y: -1.000  $00  0    Button:  off  off
Joystick 2   X: -1.000  $00  0    Y: -1.000  $00  0    Button:  off  off
```

Joystick: Joystick number (0-7)

X/Y: DSP value (-1.00 to 1.00)
hexadecimal value ($00-$FF)
decimal value (0-255)

Button: state of 2 buttons (on /off)

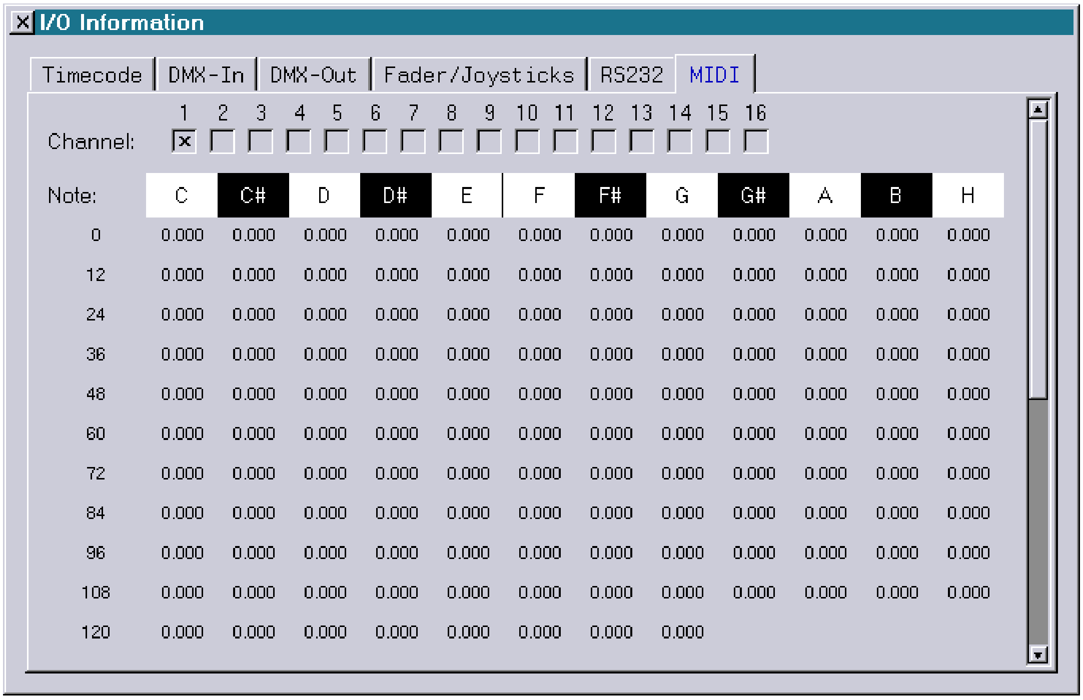
RS232



```
RS232-1:  Used by 'SerCmd1'        NetCom1:  Not used
          9600 Baud, RTS/CTS

RS232-2:  Used by 'Laser'          NetCom2:  Not used
          9600 Baud

RS232-3:  Used by 'Modem'          NetCom3:  Not used
          57600 Baud

                                   NetCom4:  Not used

                                   NetCom5:  Not used

                                   NetCom6:  Not used

                                   NetCom7:  Not used
```

The RS232 tab displays the use and the current Baud rates of the serial interfaces.

MIDI

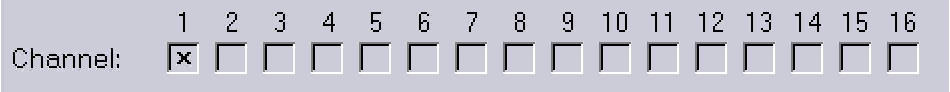


Channel: 1 ☒ 2 □ 3 □ 4 □ 5 □ 6 □ 7 □ 8 □ 9 □ 10 □ 11 □ 12 □ 13 □ 14 □ 15 □ 16 □

Note:	C	C#	D	D#	E	F	F#	G	G#	A	B	H
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
12	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
36	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
48	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
60	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
72	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
84	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
96	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
108	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
120	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000				

The Midi tab displays the incoming Midi data from the local physical Midi interface or the data from another DSP or PC assigned as a Midi server in the "Net I/O" Setup.

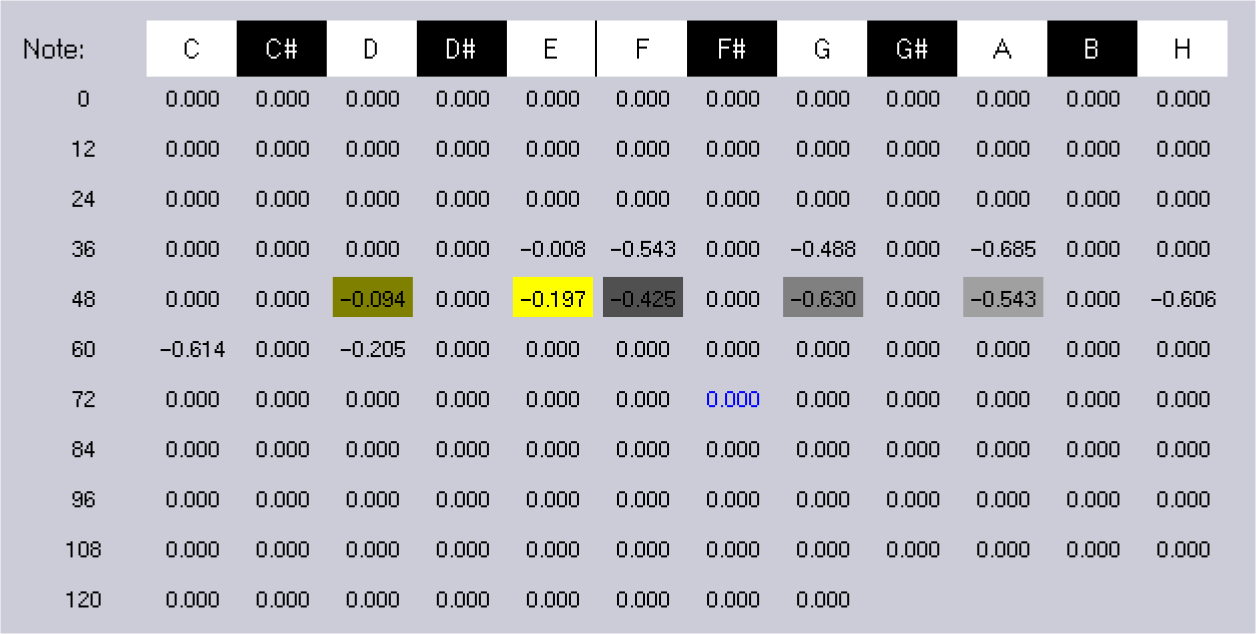
Channel



Channel: 1 ☒ 2 □ 3 □ 4 □ 5 □ 6 □ 7 □ 8 □ 9 □ 10 □ 11 □ 12 □ 13 □ 14 □ 15 □ 16 □

With the channel selection you define which of the 16 Midi channels will be displayed. Only one channel can be shown at a time.

Note



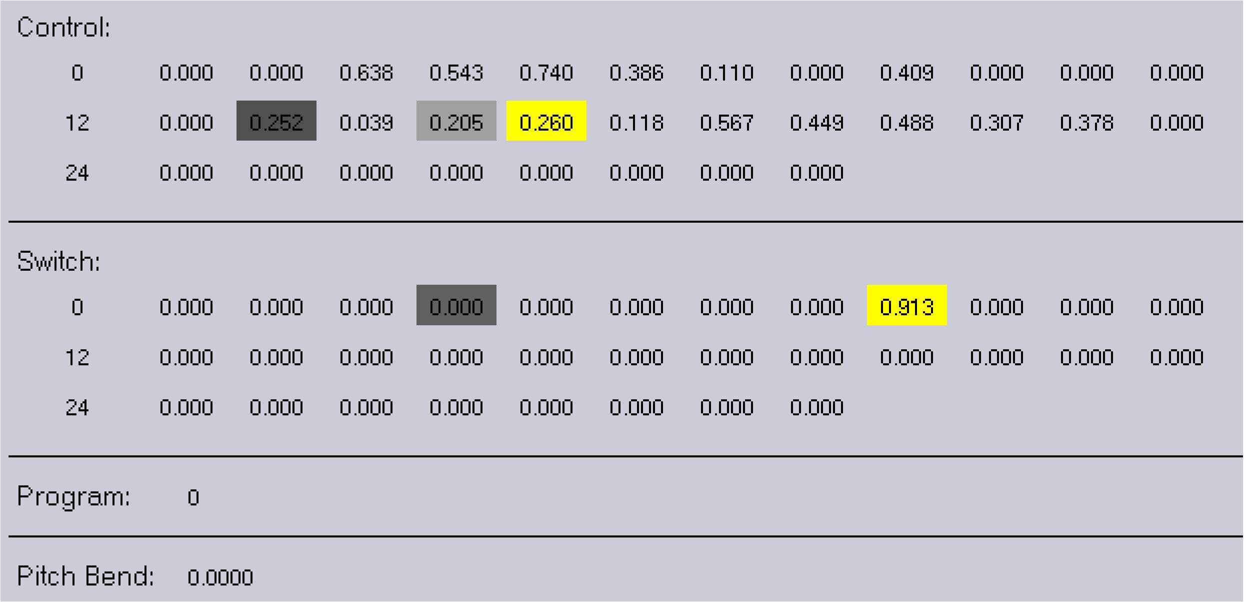
Note:	C	C#	D	D#	E	F	F#	G	G#	A	B	H
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
12	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
36	0.000	0.000	0.000	0.000	-0.008	-0.543	0.000	-0.488	0.000	-0.685	0.000	0.000
48	0.000	0.000	-0.094	0.000	-0.197	-0.425	0.000	-0.630	0.000	-0.543	0.000	-0.606
60	-0.614	0.000	-0.205	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
72	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
84	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
96	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
108	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
120	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000				

Each row shows one octave of the midi notes starting with the note number shown under the column "Note:".
The first row starts with note number 0 and ends with note number 11.
Every value displays the midi note velocity with a value from -1.00 to 1.00.
A value greater than 0 means this note is pressed and a value smaller than 0 means the note is released.
When you move the mouse over one of the values, the value will be highlighted (blue) and its note number will be displayed in the upper right corner.

Color definitions:

Yellow	last received value
Dark yellow	second last received value
Gray	third last received value
Light gray	fourth last received value

Control / Switch / Program / Pitch Bend



Control:

0	0.000	0.000	0.638	0.543	0.740	0.386	0.110	0.000	0.409	0.000	0.000	0.000
12	0.000	0.252	0.039	0.205	0.260	0.118	0.567	0.449	0.488	0.307	0.378	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000				

Switch:

0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.913	0.000	0.000	0.000
12	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000				

Program: 0

Pitch Bend: 0.0000

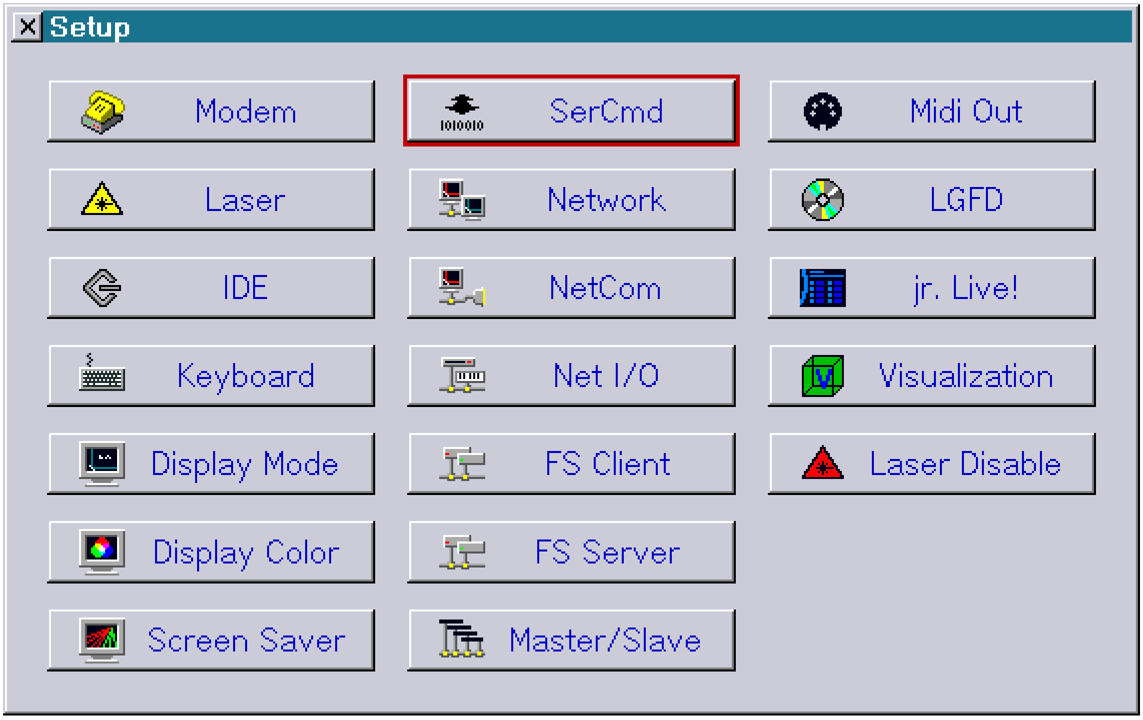
When you increase the size of the window or when you scroll down, you will get information on the values of the midi control, switch, program and pitch bend.

The Mark 2 can remote control other devices over RS232 or Ethernet (TCP/UDP/OSC) or it can be remote controlled by other devices over RS232 or TCP/IP.

Setup

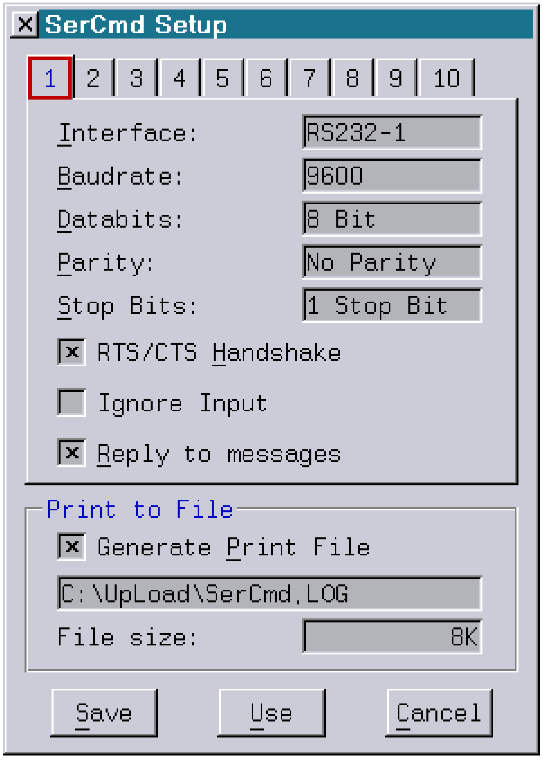
In order to be able to remote control other devices from the Mark 2 over RS232 or have the Mark 2 be remote controlled over RS232, the RS232 interfaces which are to be used have to be configured in the SerCmd setup first.

The SerCmd setup can be reached either by entering "SETUP" and then selecting "SerCmd"



or directly using the command "SERCMD.Setup".

The following window will open which is used to configure the up to 10 RS232 interfaces used by the SerCmd.



Interface: RS232-1
Baudrate: 9600
Databits: 8 Bit
Parity: No Parity
Stop Bits: 1 Stop Bit
☒ RTS/CTS Handshake
☐ Ignore Input
☒ Reply to messages

Print to File
☒ Generate Print File
C:\UpLoad\SerCmd.LOG
File size: 8K

Configuration options for each RS232 interface:

Interface: Determines the interface for sending and receiving data.
The following options are available:

compact Mark 2

None:	No interface => SerCmd is deactivated
RS232-1 to RS232-3:	One of the three local RS232 interfaces
NetCom1 to NetCom16:	One of the 16 possible NetCom interfaces

workstation Mark 2

None:	No interface => SerCmd is deactivated
Modem:	The local modem interface
Tablet:	The local tablet interface Attention: Pin 9 must not be connected because this carries the +12V supply voltage for the tablet!
NetCom1 to NetCom16:	One of the 16 possible NetCom interfaces

Baudrate: Sets the transmission speed in Bit/s.
Options are: 9600, 19200, 38400, 57600, 76800, 115200, 153600, 230400 and 250000.

Default: 9600 Baud

Databits: Determines how many data bits are to be transmitted per data word. Options are "7 Bit" or "8 Bit".

Default: 8 Bit

Parity: Indicates if, and if yes which parity check is to be used for detecting errors.

Options are:

No Parity	No parity check.
Even Parity	Parity bit will be set if the sum of all ones is odd.
Odd Parity	Parity bit will be set if the sum of all ones is even.

Default: No Parity

Stop Bits: Determines how many stop bits are to be transmitted at the end of a data word. Options are "1 Stop Bit" and "2 Stop Bits".

Default: 1 Stop Bit

RTS/CTS Handshake:

Determines whether a data flow control using the signals RTS and CTS is to be used during data transmission. Data flow control should always be used when large amounts of data are transmitted at high speeds. The data flow control ensures that no buffer overflow with the associated loss of data occurs.
Note: In order to use the data flow control the signals RTS/CTS have to be connected.

Default: Enabled

Ignore Input: When "Ignore Input" is activated the Lasergraph DSP Mark 2 will ignore all commands received over RS232.

Default for the first interface:	Disabled
Default for all other interfaces:	Enabled

Reply to messages:

As long as "Reply to messages" is active the Lasergraph DSP Mark 2 will respond to incoming commands and queries.
When "Reply to messages" is turned off the Lasergraph will not send any data of its own accord.

The setting concerns the replies after the command "<execmd", all status messages, the echo outputs and the replies to the queries "<alive?" and "<date?".

Default for the first interface:	Enabled
Default for all other interfaces:	Disabled

Settings affecting all RS232 interfaces:

Generate Print File:

As long as "Generate Print File" is checked all text received with the command "<print" will be appended to the specified text file.

Default: Enabled

File size: Determines the maximum size of the text file. Options are:

8K	The maximum size for the text file is 8KByte.
16K	The maximum size for the text file is 16KByte.
unlimited	The text file has no size limit.

Once the file exceeds the maximum size the text which was received first will be deleted.
The text received most recently will thereby be retained.

Note:
The size limit only works on the local hard drive of the Lasergraph DSP Mark 2. It will not work if the text file is located on a network drive.

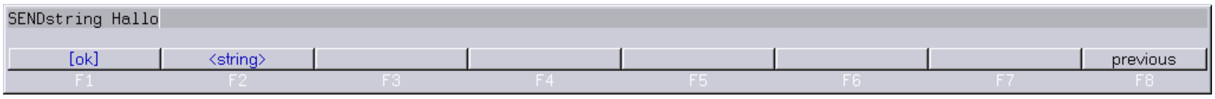
Default: 8K

Controlling Devices

The command "SENDstring" is available for controlling external devices.

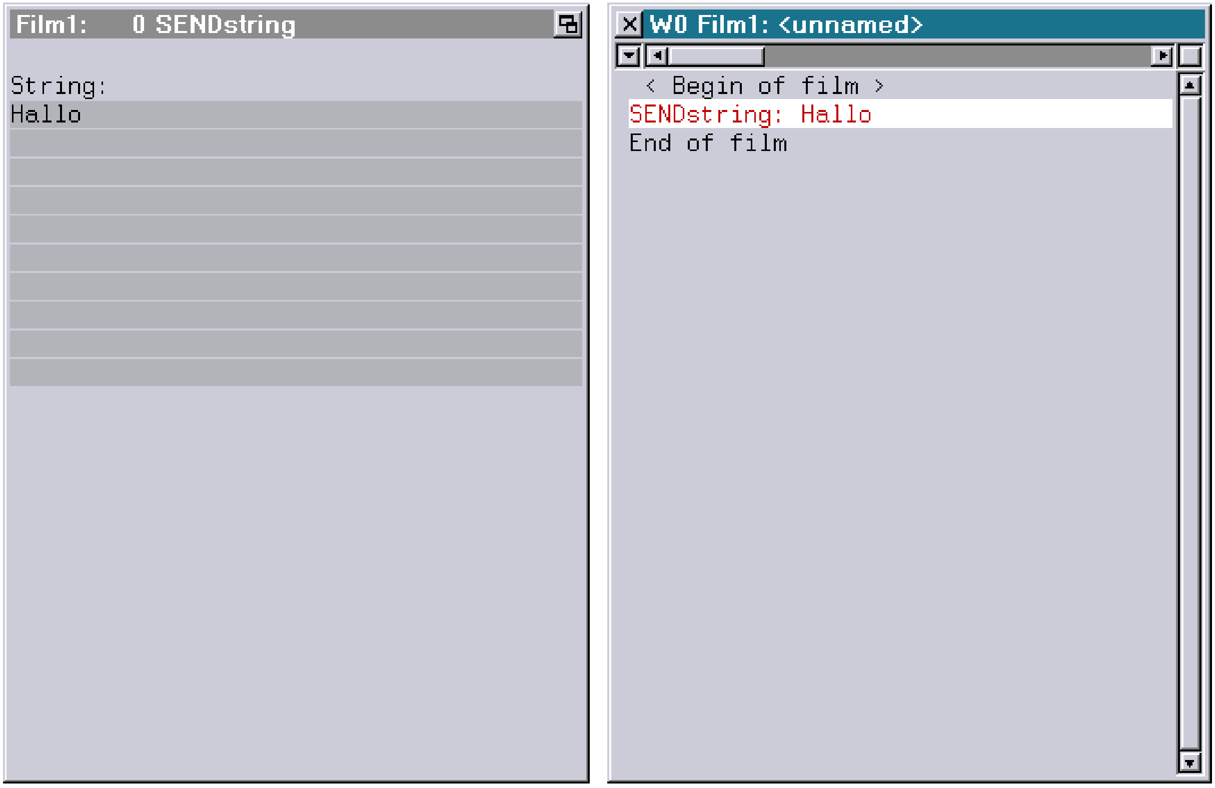
The command can be called up:

1) through the command line:



```
SENDstring Hallo
```

2) from the trickfilm:



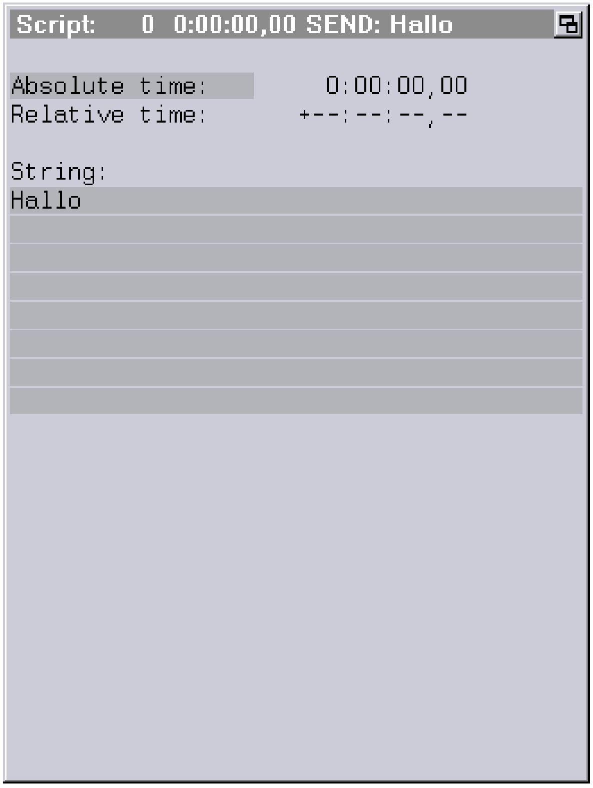
Film1: 0 SENDstring

String:
Hallo

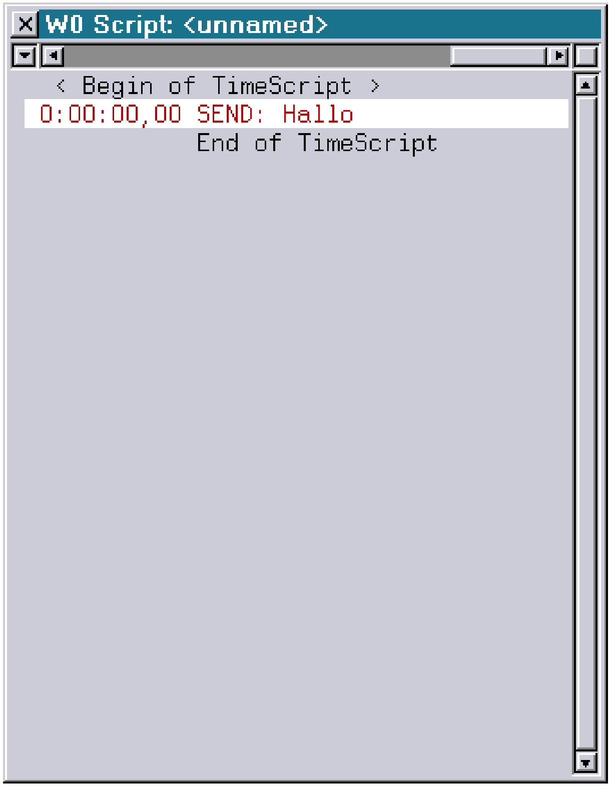
W0 Film1: <unnamed>

```
 < Begin of film >
SENDstring: Hallo
End of film
```

3) from the timescript:



```
Script:     0  0:00:00,00 SEND: Hallo

Absolute time:        0:00:00,00
Relative time:       +--:--:--,--

String:
Hallo
```



```
W0 Script: <unnamed>
  < Begin of TimeScript >
0:00:00,00 SEND:  Hallo
           End of TimeScript
```

The command "SENDstring" expects the character sequence which is to be sent as its parameter.

```
SENDstring  <string>
```

A CR/LF will automatically be added at the end of the character sequence.

In case character sequences cannot be used for control the command "SENDstring" can also be used to send hexadecimal values.
Two backslashes at the beginning of the sequence signal that one or more hex values will follow, separated by commas.

The hex values have to be in the range from 00-FF (decimal: 0-255). They can be entered in one of the following formats alternatively:

$00 - $FF
or 0x00 - 0xFF

Example:

`SENDstring \\$12`	sends the decimal value 18
`SENDstring \\0x12`	sends the decimal value 18
`SENDstring \\$11,$22,$33`	sends the decimal values 17, 34 and 51
`SENDstring \\0x11,0x22,0x33`	sends the decimal values 17, 34 and 51

Selection Of The Interface For Sending

Certain key words at the beginning of the string can be used to specify the interface through which the sequence of characters is to be sent.

Options are the 10 RS232 interfaces which can be configured in the SerCmd, TCP and UDP.

If no corresponding key word is entered the SerCmd interface no. 1 will be used automatically.

The following key words are possible:

SER://<no>/ sends the string to a specific SerCmd interface, with <no> being the number of the desired SerCmd interface (1-10).
The interface has to be configured in the SerCmd setup first.

SER://*/ sends the string to all configured SerCmd interfaces.

TCP://<ipaddr>:<port>/ sends the string as a TCP packet over the Ethernet to the specified IP address and the specified port.

UDP://<ipaddr>:<port>/ sends the string as a UDP packet over the Ethernet to the specified IP address and the specified port.

Notes concerning TCP / UDP:

- TCP and UDP packets can always be sent, irrespective of the setup.
- <ipaddr> if '*' is specified instead of an IP address, the string will be sent to the IP address which was used last.
- <port> if '0' is specified instead of a valid port number, the string will be sent to the same port used previously.

OSC://<ip-address>:<port>/<osc-address>\<osc-type>\<osc-parameter>

Depending on the selected <osc-type> different kinds of OSC packets can be sent:

string	a character string is sent
int	up to eight integer values (32-Bit) can be sent
float	up to eight floating point values (32-Bit) can be sent

If no <osc-type> is specified, an OSC packet without arguments is sent to the specified address.

Examples

Example 1:

```
SENDstring SER://2/Hello World
```

Sends the character sequence "Hello World" through the SerCmd interface no. 2.

If this has not been configured the following error message will be displayed: "SENDstring failed, check 'SETUP'!".

Example 2:

```
SENDstring UDP://192.168.1.1:8210/\\$01,$02,$03
```

Sends the values 1, 2 and 3 as a UDP packet to the IP address 192.168.1.1 and the port number 8210.

Example 3:

```
SENDstring OSC://192.168.1.126:8000/Lasergraph-
DSP/1/Label\string\Hello World!
```

Sends the character string "Hello World" as an OSC packet over UDP to the IP address 192.168.1.126 and the port 8000.

The OSC address is "/Lasergraph-DSP/1/Label".

Example 4:

```
SENDstring OSC://192.168.1.126:8000/Lasergraph-
DSP/1/Slider\int\17
```

Sends the integer value 17 as an OSC packet over UDP to the IP address 192.168.1.126 and the port 8000.

The OSC address is "/Lasergraph-DSP/1/Slider".

Example 5:

```
SENDstring OSC://192.168.1.126:8000/Lasergraph-
DSP/1/Slider\float\0.123\14.2\-3.5
```

Sends the floating point values 0.123, 14.2 and -3.5 as an OSC packet over UDP to the IP address 192.168.1.126 and the port 8000.

The OSC address is "/Lasergraph-DSP/1/Slider".

Controlling the Lasergraph DSP Mark 2

Control over RS232

There are numerous different commands for controlling the Lasergraph DSP Mark 2 over RS232. All commands begin with a '<' and have to be concluded with a CR/LF.

For the control to work over RS232 the checkbox "Ignore Input" must not be active in the SerCmd setup.

The following commands are supported:

`<execmd <command>` enters the command into the command line of the Lasergraph DSP Mark 2 and executes it as if it was entered locally.

Example:

`<execmd ReBoot Test` loads the show called "Test.BOOT"

If "Reply to messages" is active in the setup the Lasergraph DSP Mark 2 will reply to the command as follows:

`>ack`	confirmation that the command was received
`>busy`	command was rejected because another command is being processed
`>fail`	command was aborted with an error
`>done`	command was executed without errors

`<key <keycode>` simulates pressing a key. The ASCII code of the corresponding key has to be specified.

`<status on/off` turns the output of status messages on or off.
The following status messages are possible:

`>info <text>` An information was displayed in the status bar.
`>error <text>` An error was displayed in the status bar.

Note:
Status messages will not only be sent as the result of commands sent over RS232 but also when commands are entered locally.

Status messages will only be sent if "Reply to messages" is active.

`<echo on/off`

turns the echo on or off.
When the echo is turned on all commands received over RS232 will be acknowledged as follows (provided "Reply to messages" is active):

```
>echo '<command>'
```

`<alive?`

triggers the following reply (provided "Reply to messages" is active):

```
>alive
```

`<date?`

queries the date and time from the Lasergraph DSP Mark 2.
The reply will be (provided "Reply to messages" is active):

```
>date <date> <time>
```

`<print <text>`

writes the specified text into the print file specified in the SerCmd setup.

`<wopen`

opens the SerCmd console window.
Note: The console window can also be opened locally using the command "SERCMD.Console"

`<wcls`

clears the content of the SerCmd console window and sets the cursor to the top left position.
Note: The console window can also be cleared locally using the command "SERCMD.CLS".

`<wprint <text>`

writes the specified text to the current cursor position in the SerCmd console window.
Note: This command will not open the console window.

`<wprintln <text>`

writes the specified text to the current cursor position in the SerCmd console window. Afterwards the cursor will be positioned at the beginning of the following line.
Note: This command will not open the console window.

`<wset <x> <y>`

sets the cursor of the SerCmd console window to the position specified by x and y.

The parameters mean:
x=0 first column, x=1 second column, x=2 third column etc.
y=0 first line, y=1 second line, y=2 third line etc.

```
<fader <no> <v> [<v> [<v> ... <v>]]
```

sets one or more of the 16bit fader values.
The parameter <no> specifies the number of the first fader which is to be se t (0-63). The fader values following this have to be in the range from 0 to 65535. They can be entered in hexadecimals alternatively ($0000 - $FFFF or 0x0000 - 0xFFFF).

The values set here can be queried in the trickfilm using the function "Fader(no)". However, it has to be noted that the values returned there are standardized to the range 0.000 - 1.000.

Example:

```
<fader 16 0 32768 65535
```

The Lasergraph DSP Mark 2 will then supply the following values:

Fader(16) = 0.0
Fader(17) = 0.5
Fader(18) = 1.0

```
<faderb <no> <b> [<b> [<b> ... <b>]]
```

defines the state of one or more fader buttons.
The parameter <no> specifies the number of the first fader button which is to be defined. It has to be in the range from 0 to 63.

The values following this specify whether the button is to be pushed or not.

<b> = 0	not pushed
<b> != 0	pushed

In the trickfilm this state can be queried using the function "FaderB(no)".

Example:

```
<faderb 16 0 1 0
```

The Lasergraph DSP Mark 2 will then supply the following values:

FaderB(16) = 0
FaderB(17) = 1
FaderB(18) = 0

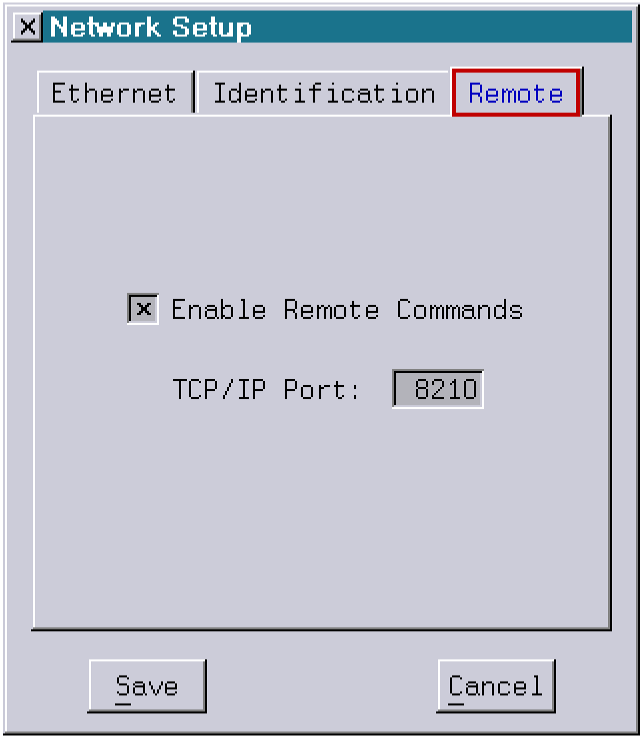
```
<sharedvar <no> <v> [<v> [<v> ... <v>]]
```

sets one or more Shared Variables.
The parameter <no> specifies the number of the first Shared Variable which is to be defined. It has to be in the range from 0 to 63.

The values set here can be queried in the trickfilm using the function "Shared(no)".

Control over TCP/IP

In order to allow a Mark 2 to be controlled over TCP/IP the checkbox "Enable Remote Commands" has to be activated in the network setup under "Remote" and the desired port has to be selected.



Network Setup

☒ Enable Remote Commands

TCP/IP Port: 8210

It is then possible to send commands in the form of TCP/IP packets to the IP address of the Lasergraph DSP Mark 2 and the specified port number.

The commands have to be concluded with CR/LF.
After receipt they will be entered into the command line of the Lasergraph DSP Mark 2 and executed exactly as if they were entered locally.

Introduction

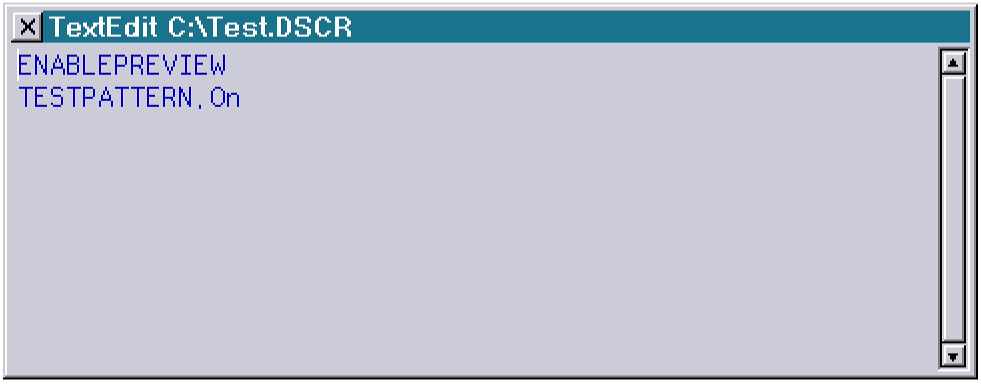
The Lasergraph DSP offers the possibility to put together any commands which would normally be entered into the command line in a text file in order to execute them at a later time.

Such a text file is called "Script" and should have the file extension ".DSCR".

A script can be called up by the user either with the command "startSCRIPT", automatically when the system is started, after a show is loaded or with a ScriptButton (see below).

After the script is called up the commands contained in the script file will be copied to the command line one after the other and executed there.

Script files can be created and edited either on another computer or on the DSP's own text editor "TextEDIT".



```
ENABLEPREVIEW
TESTPATTERN,On
```

When the script shown here is called up it would first activate the preview window and then the test picture.

Important!

When writing a script it is important to remember that not all commands can be entered in all operating modes of the Lasergraph DSP.
For instance the "Compile" command will only work in the edit mode.
If such mode-dependent commands are to be used in a script it may be necessary to select the corresponding mode first with the appropriate commands.

A script can also start another script using the command "startSCRIPT <filename>".
In this case the calling script will be finished immediately and all commands following the "startSCRIPT" will be ignored!

startSCRIPT

```
startSCRIPT <filename>
```

The command "startSCRIPT" executes the specified script.

Parameter: "filename" ⇨ file name of the script which is to be executed

Notes:

- The file name can be specified with or without the path.
- Specifying the file extension (".DSCR") is optional.

ScriptContinueOnError

```
ScriptContinueOnError <Yes>/<No>
```

If an error occurs while a script is executed the script will be stopped at that point and the remaining commands will not be executed.

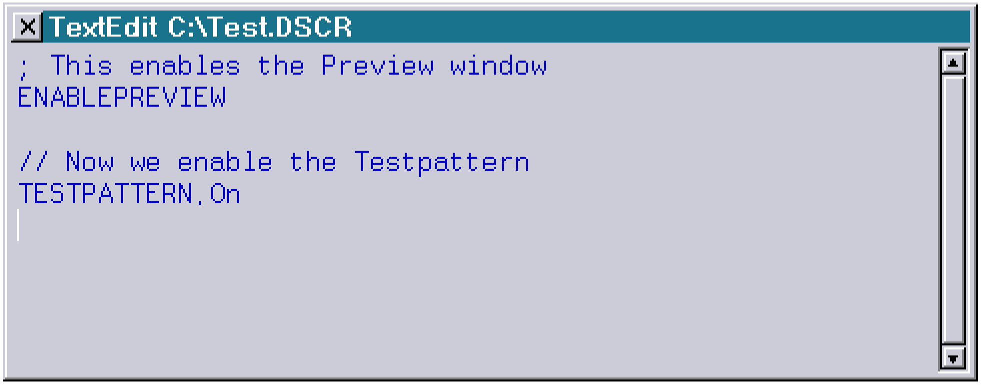
Often it can be useful to continue the execution of the script nonetheless.

The command "ScriptContinueOnError Yes" can be used to determine that scripts are to be continued even when an error occurs with one of the commands within the script.

The setting specified with "ScriptContinueOnError" will remain unchanged until the Lasergraph DSP is restarted.

Comments

If a line in a script starts with ";" or with "//" this line will be recognized as a comment and will be ignored during execution of the script:



```
; This enables the Preview window
ENABLEPREVIEW

// Now we enable the Testpattern
TESTPATTERN.On
```

Special Scripts

AutoScript

The AutoScript is executed immediately after the Lasergraph DSP is started, even before a possible AutoBoot is loaded. It has to be in the following location:

```
C:\System\AutoScript.DSCR
```

Application Example:
There are settings in the Lasergraph DSP which will be reset after a restart. If for instance the preview window is supposed to always be active this can be achieved by entering the following line in the AutoScript:

```
ENABLEPREVIEW
```

BootScript

Immediately after a show is loaded using "ReBoot" the system checks if there is a script with the same name as the BOOT present in the same directory.

If this is the case the script will be loaded and started.

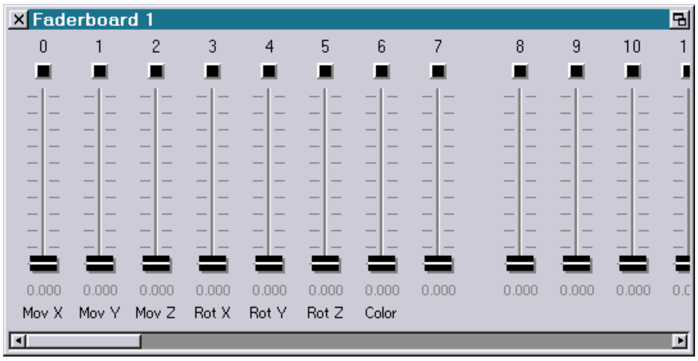
Example:

```
MyShow.BOOT
MyShow.DSCR
```

Application Example:
If, after a live show is loaded, the virtual fader board is to open immediately and the faders marked according to the assignments used in the show, this can be done using the following BootScript:

```
FADERboard.Name 0 "Mov X" "Mov Y" "Mov Z" "Rot X" "Rot Y" "Rot Z" "Color"
FADERboard.Open 1
```

Immediately after the show is loaded the fader board would open as follows:



ScriptButtons

ScriptButtons make it possible to enhance the existing graphic user interface (GUI) of the Lasergraph DSP Mark 2 by adding more buttons or to create completely new interfaces for special applications.

To allow this a script will be executed when a ScriptButton is clicked.

ScriptBUTTON.DefineWindow

```
ScriptBUTTON.DefineWindow <number> <x-pos> <y-pos> <width> <height>
<headline> <bitmap> /HighPriority
```

The command "ScriptBUTTON.DefineWindow" is used to define a ScriptButton which is similar to a normal window with a headline and a closing symbol.

Parameters:

number	number of the ScriptButton (0-79)
x-pos / y-pos	position of the ScriptButton
width / height	size of the ScriptButton
headline	text displayed in the headline of the window
bitmap	file name of the background bitmap

Options:

/HighPriority Unlike the other ScriptButtons a ScriptButton created with "DefineWindow" is displayed with normal priority, i.e. it can be concealed partly or completely by other windows. The option "/HighPriority" always opens the window with high priority so it is always visible above the other windows.

Notes:

- In order to make a ScriptButton defined with "DefineWindow" visible it is necessary to also call up "Show" (see "ScriptBUTTON.Show").
- If only the number is specified as a parameter, the ScriptButton will be deleted and the resources used will be freed.
- <x-pos> / <y-pos> as well as <width> / <height> relate to the client are, i.e. the area within the window frame.

- If <width> resp. <height> is zero the width resp. height of the background bitmap will be used.
- If negative values are specified for position or size the previous values will be retained.
- The bitmap will always be displayed in its original size:
  In case the ScriptButton is larger than the bitmap the bitmap will be displayed several times.
  If the ScriptButton is smaller than the bitmap the bitmap will be clipped on the right resp. the bottom.
- The file name for the bitmaps should contain an absolute path. If a relative path is used the result will depend on the current directory.

ScriptBUTTON.DefineBitmap

```
ScriptBUTTON.DefineBitmap <number> <x-pos> <y-pos> <width> <height>
<script> <bitmap 1> <bitmap 2> <bitmap 3>
```

The command "ScriptBUTTON.DefineBitmap" is used to define the size and look of the ScriptButton as well as the action which is to be executed when the button is clicked.

Parameters:

number	number of the ScriptButton (0-79)
x-pos / y-pos	position of the ScriptButton
width / height	size of the ScriptButton
script	file name of the associated script
bitmap 1	file name of the bitmap which is displayed in idle state
bitmap 2	file name of the bitmap which is displayed when the mouse pointer is positioned over the button
bitmap 3	file name of the bitmap which is displayed when the mouse key is pressed

Notes:

- In order to make a ScriptButton defined with "DefineBitmap" visible it is necessary to call up "Show" (see "ScriptBUTTON.Show").
- If only the number is specified as a parameter, the ScriptButton will be deleted and the memory used by the bitmaps will be freed.
- If <width> or <height> are zero the width resp. height of the first bitmap will be used.

- If negative values are specified for position or size the previous values will be retained.
- If no script is specified ("" as file name) no action will be executed when the button is clicked.
- It is possible to specify 1, 2 or 3 bitmaps:

One bitmap is specified:	The idle state bitmap is used in all three states.
Two bitmaps are specified:	Optical differentiation is only between the idle state and the state with the mouse pointer over the button. It does not make a difference whether the mouse button is pressed or not.
Three bitmaps are specified:	There is an individual bitmap for each of the three states.

- Bitmaps will always be displayed in their original size:

  In case the ScriptButton is larger than the bitmap the bitmap will be displayed several times. If the ScriptButton is smaller than the bitmap the bitmap will be clipped on the right resp. the bottom.

- The file names for the script and the bitmaps should contain absolute paths. If relative paths are used the result will depend on the current directory.

ScriptBUTTON.DefineText

```
ScriptBUTTON.DefineText <number> <x-pos> <y-pos> <width> <height>
<script> <text> <font> /Left / Right /NoFrame /NoHighlight
```

The command "ScriptBUTTON.DefineText" is used to define the size, the displayed text as well as the action which is to be executed when the ScriptButton is clicked.

Parameters:

number	number of the ScriptButton (0-79)
x-pos / y-pos	position of the ScriptButton
width / height	size of the ScriptButton
script	file name of the associated script
text	text which is to be displayed
font	desired font (see below)

Options:

/Left displays the text left-aligned

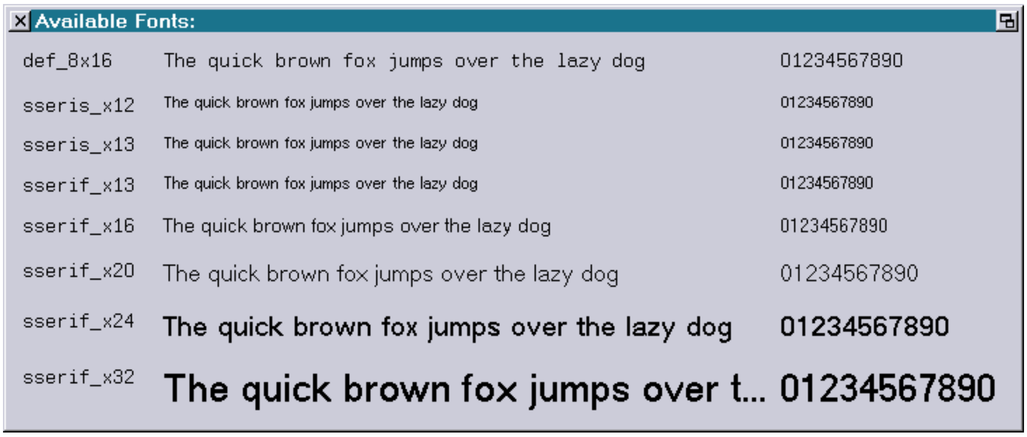
/Right displays the text right-aligned

/NoFrame suppresses the display of the normal button frame

/NoHighlight suppresses the highlighting of the button when the mouse pointer is moved over it

<u>Notes:</u>

- In order to make a ScriptButton defined with "DefineText" visible it is necessary to call up "Show" (see "ScriptBUTTON.Show").
- If only the number is specified as a parameter, the ScriptButton will be deleted and the resources used will be freed.
- If negative values are specified for position or size the previous values will be retained.
- If no script is specified ("" as file name) no action will be executed when the button is clicked.
- The file name for the script should contain the absolute path. If relative paths are used the result will depend on the current directory.
- In case the text contains a space it has to be enclosed in quotation marks.
- Fonts which can be used are the normal DSP fonts which can be displayed using the command "FONTTEST":



def_8x16	The quick brown fox jumps over the lazy dog	01234567890
sseris_x12	The quick brown fox jumps over the lazy dog	01234567890
sseris_x13	The quick brown fox jumps over the lazy dog	01234567890
sserif_x13	The quick brown fox jumps over the lazy dog	01234567890
sserif_x16	The quick brown fox jumps over the lazy dog	01234567890
sserif_x20	The quick brown fox jumps over the lazy dog	01234567890
sserif_x24	The quick brown fox jumps over the lazy dog	01234567890
sserif_x32	The quick brown fox jumps over t...	01234567890

If no font is specified "sserif_x20" will be used as the default.

ScriptBUTTON.Show

```
ScriptBUTTON.Show <number> [<parent>]
```

The command "ScriptBUTTON.Show" makes a ScriptButton which was previously created with "ScriptBUTTON.DefineBitmap" resp. "ScriptBUTTON.DefineText" visible.

Parameter:

number	number of the ScriptButton (0-79)
parent	number of a parent ScriptButton (0-79)

Notes:

- Only ScriptButtons which were previously defined with "DefineBitmap" resp. "DefineText" can be made visible.

- ScriptButtons can overlap partially or completely. In this case the order in which the ScriptButtons were made visible with "ScriptBUTTON.Show" determines which ScriptButton will be hidden by which other ScriptButton.
  If for instance several ScriptButtons are to be placed on a background which is also defined as a ScriptButton, the background has to be made visible first.

- If a parent ScriptButton is specified the ScriptButton will be opened within the parent ScriptButton and the specified positions will be relative to the parent ScriptButton.

ScriptBUTTON.Hide

```
ScriptBUTTON.Hide <first> [<last>]
```

The command "ScriptBUTTON.Hide" hides one or more ScriptButton without freeing the resources used by it. These ScriptButtons can therefore be made visible again at any time with "ScriptBUTTON.Show" without having to use "ScriptBUTTON.DefineBitmap" resp. "ScriptBUTTON.DefineText" again.

Parameter:

first	number of the first ScriptButton to be closed (0-79)
last	optional number of the last ScriptButton to be closed (0-79)

ScriptBUTTON.HideAll

```
    ScriptBUTTON.HideAll
```

The command "ScriptBUTTON.HideAll" closes all open ScriptButtons without freeing the loaded resources so the ScriptButtons can be made visible again with "ScriptBUTTON.Show" without having to use "ScriptBUTTON.DefineBitmap" resp. "ScriptBUTTON.DefineText" again.

ScriptBUTTON.Remove

```
    ScriptBUTTON.Remove <first> [<last>]
```

The command "ScriptBUTTON.Remove" is similar to the command "ScriptBUTTON.Hide":
The ScriptButton(s) specified will be closed. In contrast to "Hide", when using "Remove" all resources used (bitmaps and texts) will be freed.

This may be useful in case a ScriptButton is to be defined with "ScriptBUTTON.DefineBitmap" which has previously already been defined with "ScriptBUTTON.DefineText".
In this case the old text would otherwise be displayed on top of the bitmap.

Parameter:

first	number of the first ScriptButton to be removed (0-79)
last	optional number of the last ScriptButton to be removed (0-79)

ScriptBUTTON.RemoveAll

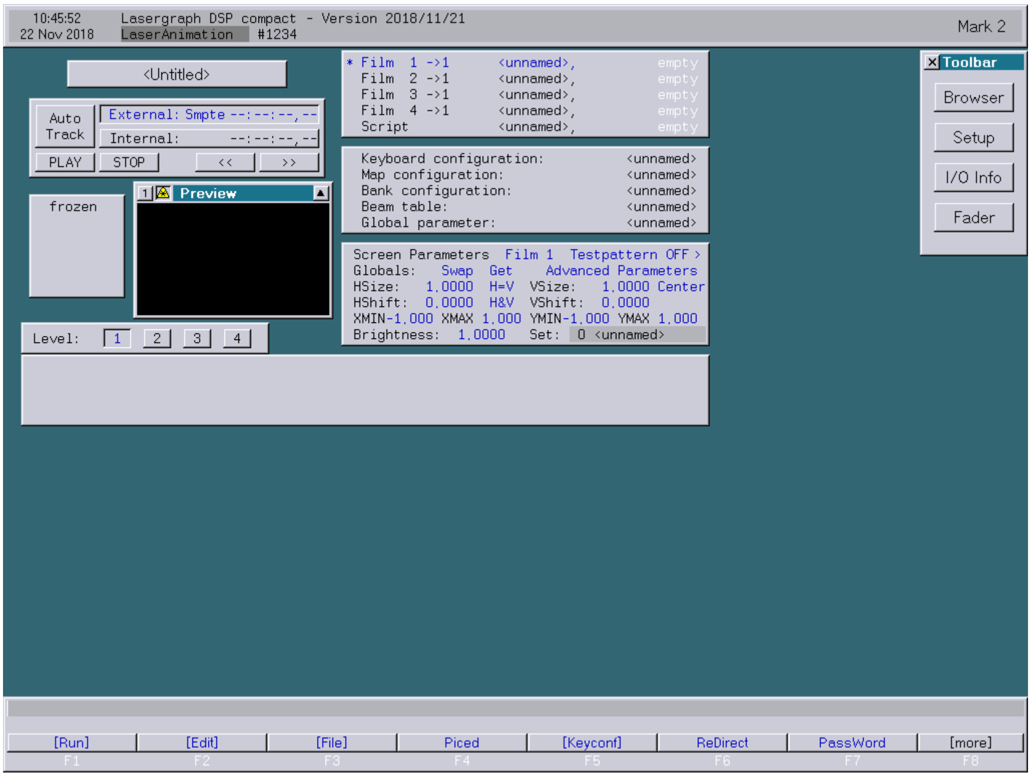
```
    ScriptBUTTON.RemoveAll
```

The command "ScriptBUTTON.RemoveAll" closes all ScriptButtons and frees the resources (bitmaps and texts) used by them.
This basically recreates the initial state.

Examples

Toolbar

ScriptButtons can be used to create a Toolbar for easy access to often used functions:



```
10:45:52     Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018  LaserAnimation  #1234

* Film  1 ->1      <unnamed>,        empty
  Film  2 ->1      <unnamed>,        empty
  Film  3 ->1      <unnamed>,        empty
  Film  4 ->1      <unnamed>,        empty
  Script           <unnamed>,        empty

  Keyboard configuration:            <unnamed>
  Map configuration:                 <unnamed>
  Bank configuration:                <unnamed>
  Beam table:                        <unnamed>
  Global parameter:                  <unnamed>

Screen Parameters   Film 1   Testpattern OFF >
Globals:    Swap  Get     Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:  0.0000  H&V  VShift:  0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000    Set:  0 <unnamed>

External: Smpte --:--:--,--
Internal:       --:--:--,--
```

The script used for this example ("Toolbar.DSCR") looks like this:

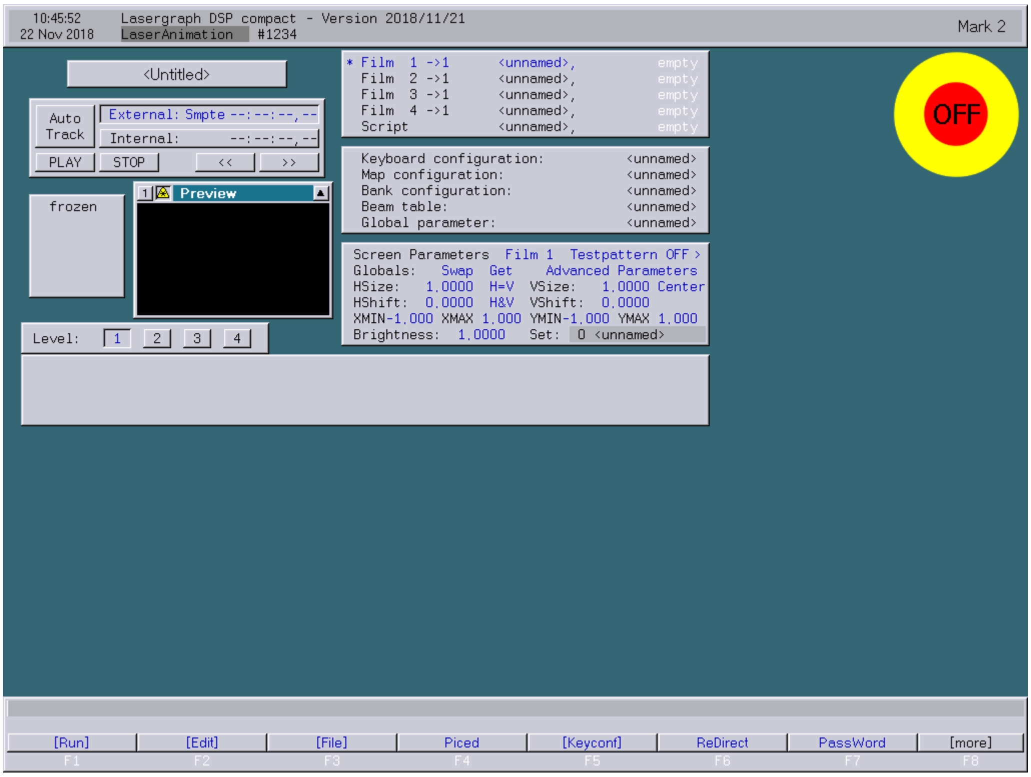
```
ScriptBUTTON.DefineWindow 50 1200 68 100 180 "Toolbar"

ScriptBUTTON.DefineText 51 10 10  80 30  "C:\System\ScriptButtons\files\Browser.dscr"  "Browser"
ScriptBUTTON.DefineText 52 10 50  80 30  "C:\System\ScriptButtons\files\Setup.dscr"    "Setup"
ScriptBUTTON.DefineText 53 10 90  80 30  "C:\System\ScriptButtons\files\IOInfo.dscr"   "I/O Info"
ScriptBUTTON.DefineText 54 10 130 80 30  "C:\System\ScriptButtons\files\Fader.dscr"    "Fader"

ScriptBUTTON.Show 50
ScriptBUTTON.Show 51 50
ScriptBUTTON.Show 52 50
ScriptBUTTON.Show 53 50
ScriptBUTTON.Show 54 50
```

Using keyboard macros it is possible to open the Toolbar with a single key (e.g. <F12>).

Button for Disabling the Laser Output

Frequently it may be necessary to disable the laser output on several Lasergraph DSP which are connected via network simultaneously.

The ScriptButtons offer a simple solution for this:



```
10:45:52     Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018  LaserAnimation   #1234
```

```
* Film  1 ->1      <unnamed>,       empty
  Film  2 ->1      <unnamed>,       empty
  Film  3 ->1      <unnamed>,       empty
  Film  4 ->1      <unnamed>,       empty
  Script           <unnamed>,       empty

  Keyboard configuration:          <unnamed>
  Map configuration:               <unnamed>
  Bank configuration:              <unnamed>
  Beam table:                      <unnamed>
  Global parameter:                <unnamed>

Screen Parameters  Film 1   Testpattern OFF >
Globals:    Swap  Get    Advanced Parameters
HSize:    1.0000  H=V  VSize:    1.0000 Center
HShift:   0.0000  H&V  VShift:   0.0000
XMIN-1.000 XMAX 1.000 YMIN-1.000 YMAX 1.000
Brightness:   1.0000   Set:  0 <unnamed>
```

The following script is started on the Lasergraph DSP which is controlled by LGRemote:

```
ScriptBUTTON.DefineBitmap 0 890 48 0 0  C:\ScriptButtons\TurnLaserOff.dscr  C:\ScriptButtons\LaserOff.bmp
ScriptBUTTON.Show 0
```

After the script is called up the button pictured above will appear.
The script which is called up by clicking on the button could look like this:

```
; Disable local laser output
ENABLELASEROUTPUT No

; Disable laser output of Lasergraph DSP with ip address 192.168.178.138 using Remote Commands
SEND tcp://192.168.178.138:8210/ENABLELASEROUTPUT No
```

The script first disables the local laser output.

Then the laser output on another Lasergraph DSP is disabled with a remote command (in this example the other Lasergraph DSP has the IP address 192.168.178.138).

It is important that **remote commands are activated** on the Lasergraph DSP which is to be addressed using remote commands (see network setup).

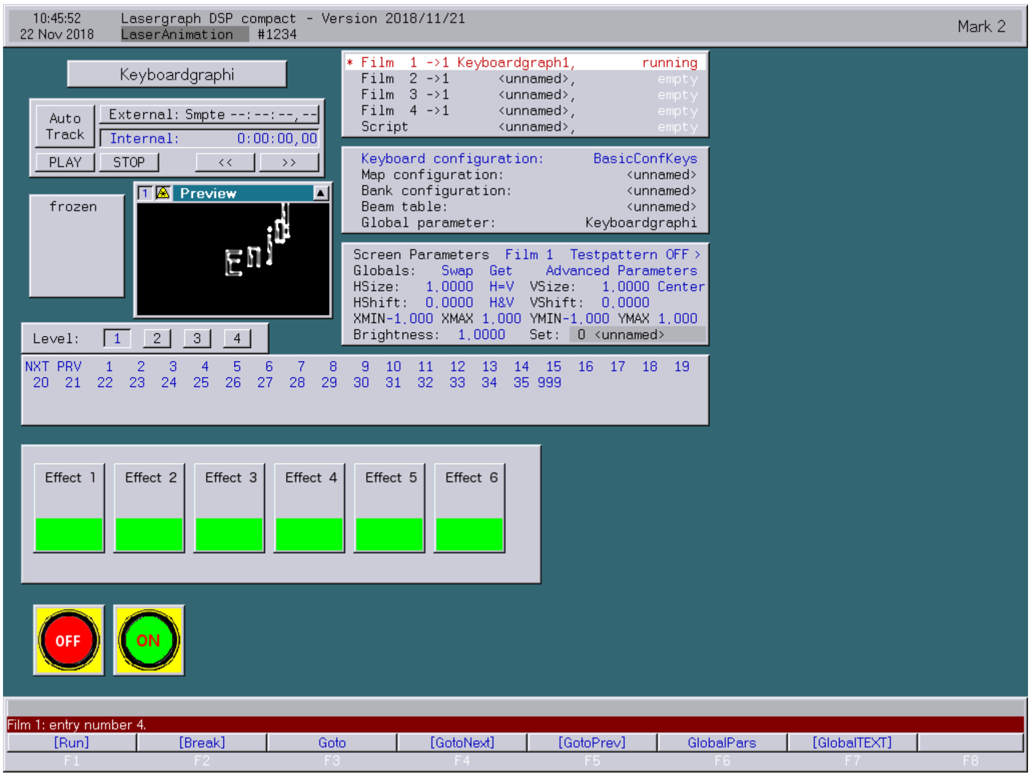
Using corresponding additional "SEND" commands the laser output can be disabled on any number of Lasergraph DSP with one click.

Attention!
In case the network connection to a Lasergraph DSP is interrupted the execution of the script can be delayed.
Moreover, the laser output on this Lasergraph DSP will stay enabled.
Therefore the ScriptButton shown here **can never replace an emergency-off button**.

Note:
The button created with ScriptButton is always positioned on top of all other windows.
If this is irritating in the other operating modes the button "LaserOff" can easily be activated or deactivated using keyboard macros.

Individual Show Interface

The ScriptButtons can be used to create any kind of show interface.



```
10:45:52      Lasergraph DSP compact - Version 2018/11/21
22 Nov 2018   LaserAnimation  #1234                                    Mark 2

* Film  1 ->1 Keyboardgraph1,         running
  Film  2 ->1       <unnamed>,          empty
  Film  3 ->1       <unnamed>,          empty
  Film  4 ->1       <unnamed>,          empty
  Script            <unnamed>,          empty

  Keyboard configuration:       BasicConfKeys
  Map configuration:                <unnamed>
  Bank configuration:               <unnamed>
  Beam table:                       <unnamed>
  Global parameter:             Keyboardgraphi

 Screen Parameters  Film 1  Testpattern OFF >
 Globals:   Swap  Get    Advanced Parameters
 HSize:   1,0000  H=V  VSize:   1,0000 Center
 HShift:  0,0000  H&V  VShift:  0,0000
 XMIN-1,000 XMAX 1,000 YMIN-1,000 YMAX 1,000
 Brightness:  1,0000   Set:  0 <unnamed>

External: Smpte --:--:--,--
Internal:         0:00:00,00

Film 1: entry number 4.
```

The script used in this example ("Keyboardgraphi.DSCR") is called up automatically after the BOOT "Keyboardgraphi.BOOT" is loaded:

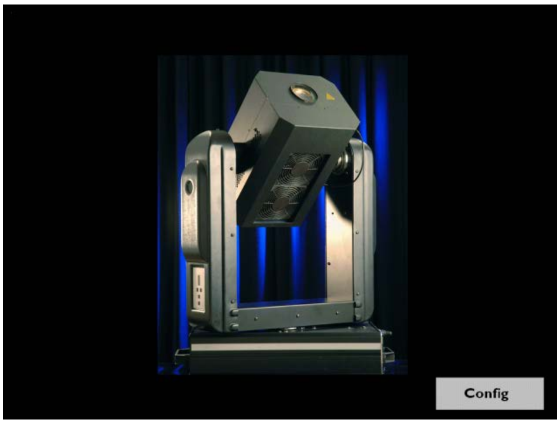
```
ScriptBUTTON.DefineBitmap   0   18    440 0 0   ""                  files\Trigger-base.bmp
ScriptBUTTON.Show           0

ScriptBUTTON.DefineBitmap   1   30    458 0 0   files\Trigger-01.dscr   files\Trigger-01.bmp
ScriptBUTTON.Show           1
ScriptBUTTON.DefineBitmap   2   110   458 0 0   files\Trigger-02.dscr   files\Trigger-02.bmp
ScriptBUTTON.Show           2
ScriptBUTTON.DefineBitmap   3   190   458 0 0   files\Trigger-03.dscr   files\Trigger-03.bmp
ScriptBUTTON.Show           3
ScriptBUTTON.DefineBitmap   4   270   458 0 0   files\Trigger-04.dscr   files\Trigger-04.bmp
ScriptBUTTON.Show           4
ScriptBUTTON.DefineBitmap   5   350   458 0 0   files\Trigger-05.dscr   files\Trigger-05.bmp
ScriptBUTTON.Show           5
ScriptBUTTON.DefineBitmap   6   430   458 0 0   files\Trigger-06.dscr   files\Trigger-06.bmp
ScriptBUTTON.Show           6
ScriptBUTTON.DefineBitmap   7   30    600 0 0   files\OFF.dscr          files\OFF.bmp
ScriptBUTTON.Show           7
ScriptBUTTON.DefineBitmap   8   110   600 0 0   files\ON.dscr           files\ON.bmp
ScriptBUTTON.Show           8
```

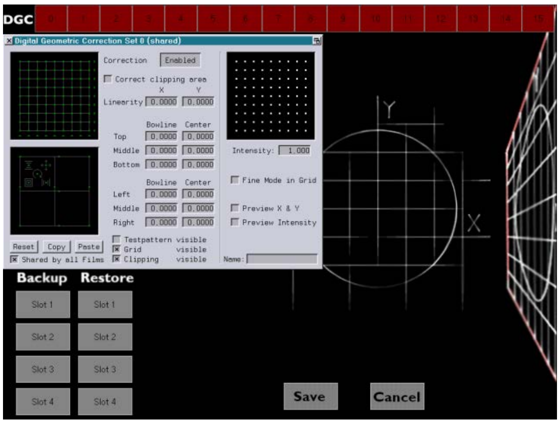
The script for deleting the buttons ("Reset.DSCR") looks like this:

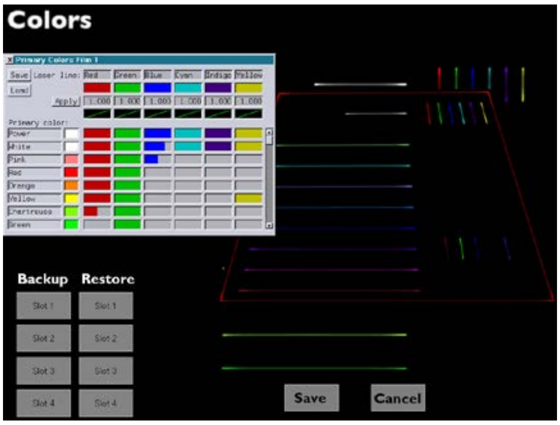
```
ScriptBUTTON.HideAll
```

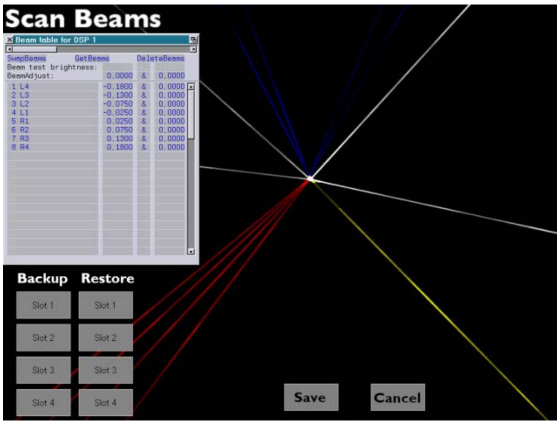
Stage Interface

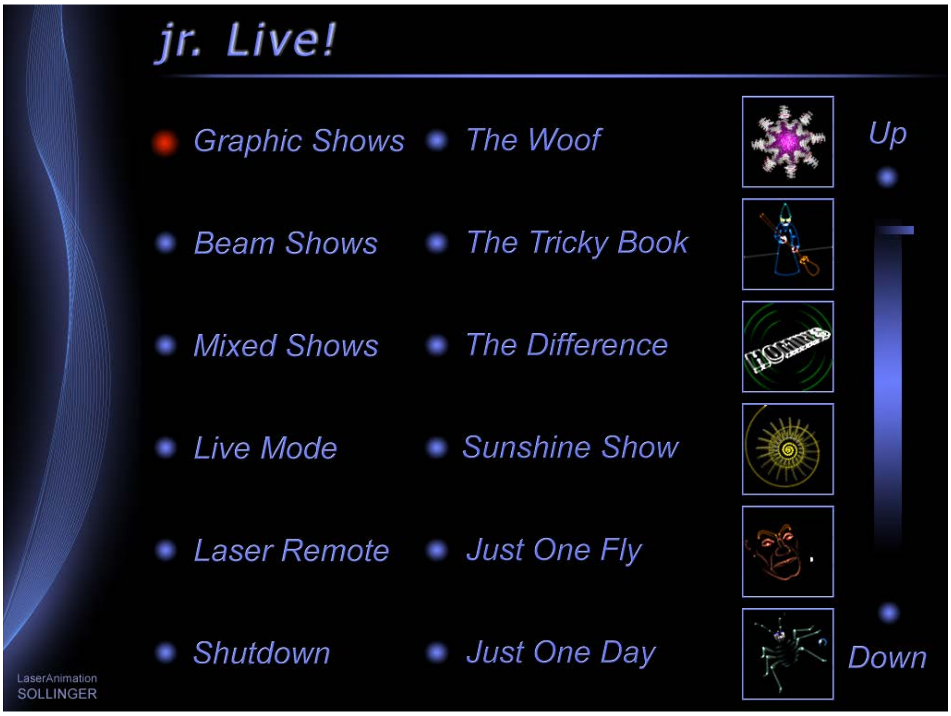
The Stage interface which is installed on our BLITZ Stage projectors was created with a previous version of the ScriptButtons:











Introduction

The "jr. Live!" is a simple graphical user interface (GUI) which is optically attractive and can be used intuitively. With "jr. Live!" you do not need special previous knowledge to select and start shows and call up live effects.

In contrast to the LGFD, the Laser-Preview is a fixed part of the new interface.

The "jr. Live!" is designed to be operated completely with the use of a touch-screen display. The shows of the "jr. Live!" are sorted into three categories:

- Graphic Shows
- Beam Shows
- Mixed Shows

For live applications there is a separate

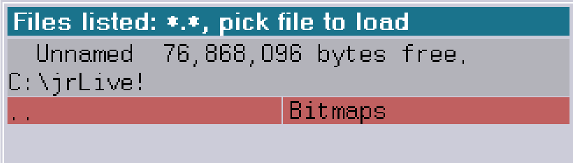
- Live Mode

Setting Up jr.Live!

The following steps are necessary to use the "jr. Live!" interface:

On the DSP

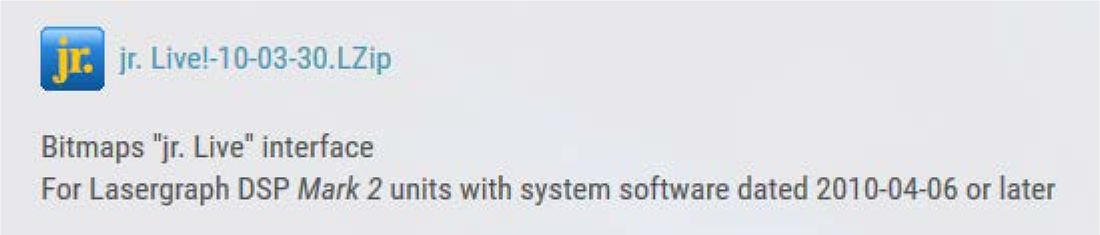
On the partition “C:” there is a directory named "jrLive!", containing a directory with all the basic bitmaps that are needed for the GUI.



```
Files listed: *.*, pick file to load
  Unnamed  76,868,096 bytes free.
C:\jrLive!
..                   Bitmaps
```

If you want to create your own directory, you can change the path to your own directory in the Setup of the jr.Live!.

You can download the bitmaps from our homepage in the customer area.

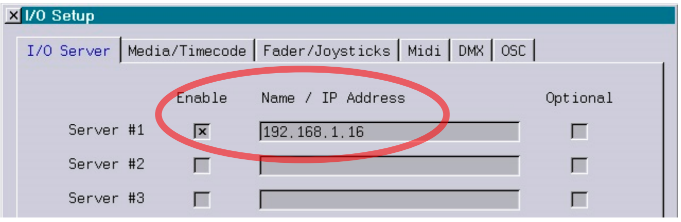


jr. Live!-10-03-30.LZip

Bitmaps "jr. Live" interface
For Lasergraph DSP *Mark 2* units with system software dated 2010-04-06 or later

Net I/O

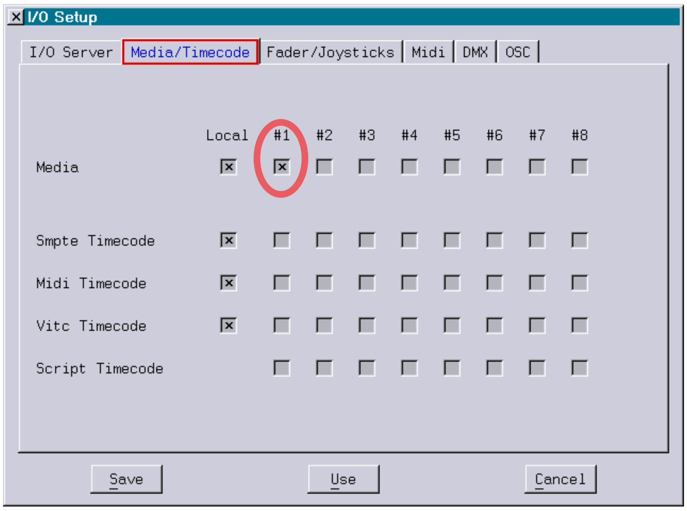
In the Net I/O setup, the PC which will be the server for the media has to be assigned. Type either the IP-address or its name into the server list and enable the Media.



I/O Setup

I/O Server | Media/Timecode | Fader/Joysticks | Midi | DMX | OSC

	Enable	Name / IP Address	Optional
Server #1	☒	192.168.1.16	□
Server #2	□		□
Server #3	□		□



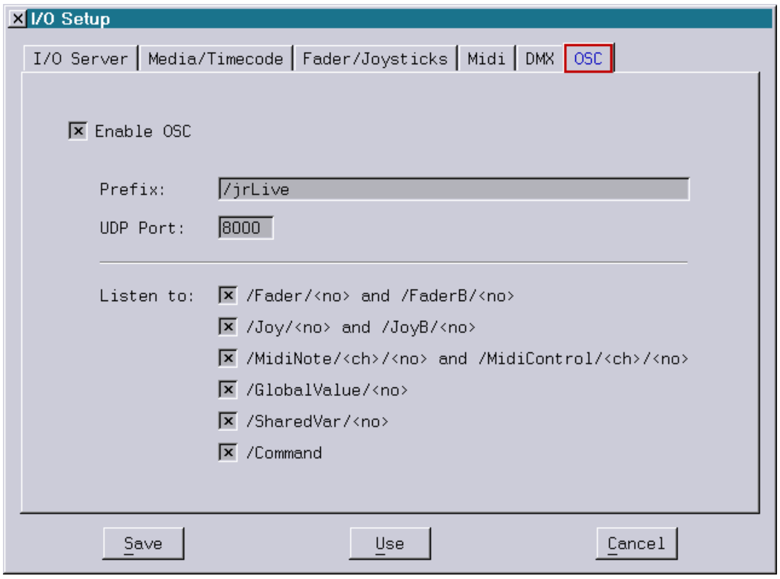
I/O Setup

I/O Server | Media/Timecode | Fader/Joysticks | Midi | DMX | OSC

	Local	#1	#2	#3	#4	#5	#6	#7	#8
Media	☒	☒	□	□	□	□	□	□	□
Smpte Timecode	☒	□	□	□	□	□	□	□	□
Midi Timecode	☒	□	□	□	□	□	□	□	□
Vitc Timecode	☒	□	□	□	□	□	□	□	□
Script Timecode		□	□	□	□	□	□	□	□

If you intend to use the Live mode with the OHM64 or another Midi Controller, you have to enable the Fader and the Midi channel for the corresponding server.

When using an iPad with TouchOSC, OSC with the corresponding port and prefix has to be enabled.



I/O Setup

I/O Server | Media/Timecode | Fader/Joysticks | Midi | DMX | OSC

☒ Enable OSC

Prefix: /jrLive

UDP Port: 8000

Listen to:
☒ /Fader/<no> and /FaderB/<no>
☒ /Joy/<no> and /JoyB/<no>
☒ /MidiNote/<ch>/<no> and /MidiControl/<ch>/<no>
☒ /GlobalValue/<no>
☒ /SharedVar/<no>
☒ /Command

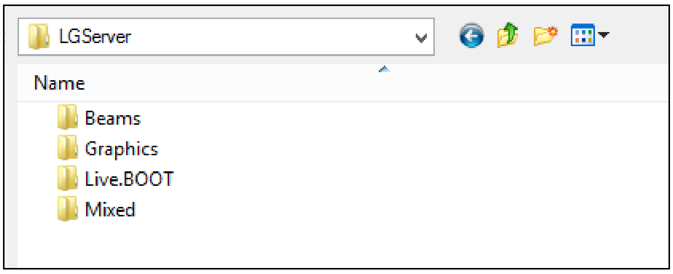
On the PC

All media content that is used by the jr.Live! GUI has to be located on the PC that is used as a server for the DSP.

Files that are necessary:
Three directories
- Graphics
- Beams
- Mixed

a DSP Boot named "Live.BOOT"

The file structure looks like this:



LGServer

Name

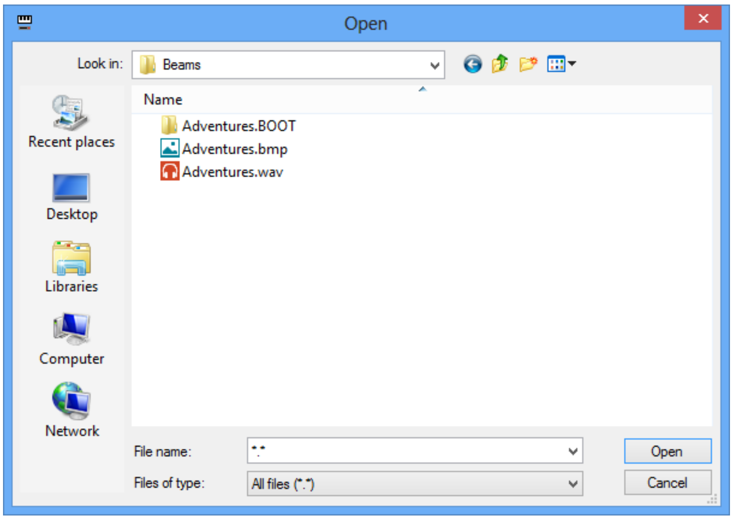
Beams
Graphics
Live.BOOT
Mixed

DSP shows that are in these directories will be displayed when the respective directory is active.

Beside the DSP-Boot the following files have to be in the same directory as the show boot:
- an audio file with the same name as the DSP-Boot
- a bitmap with the same name as the DSP-Boot

Example:

If you have created a show named "Adventures.BOOT", save the corresponding audio file with the name "Adventures.wav" (or "Adventures.mp3"). The corresponding bitmap should be named "Adventures.bmp".



Open

Look in: Beams

Name

Adventures.BOOT
Adventures.bmp
Adventures.wav

File name: *.*

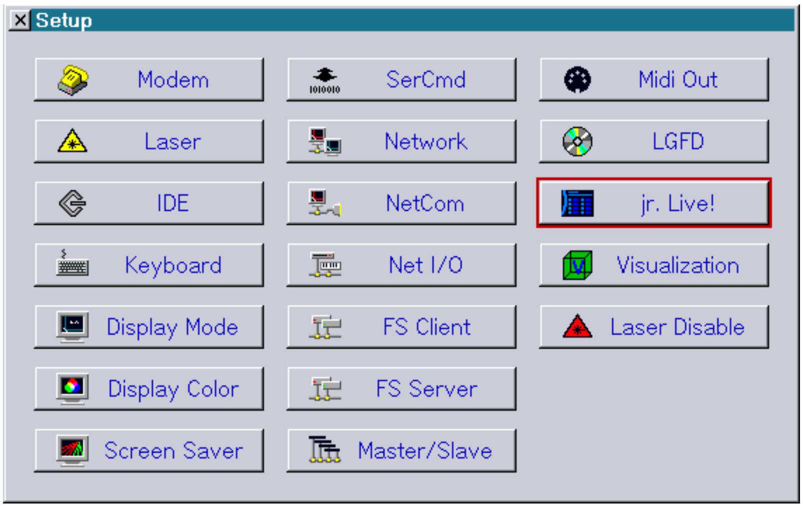
Files of type: All files (*.*)

LGServer

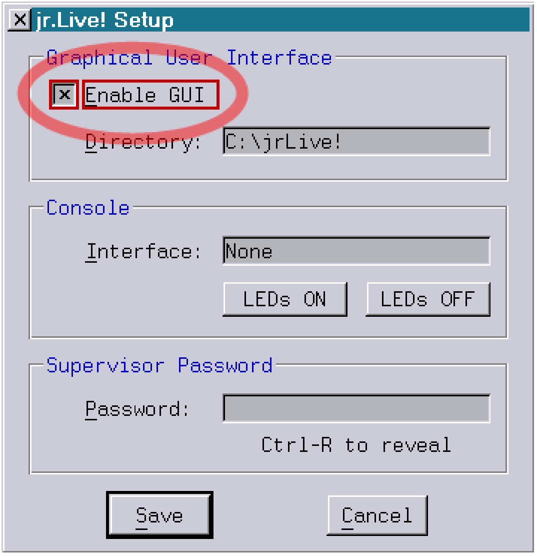
Now start the LGServer and assign the directory where all the files are.
Be sure to select all files either by typing the "*.*" jokers or by choosing "All files" in the File type menu.

Finish the settings with the button "save".

Now you can enable the jr.Live! GUI in the SETUP on the DSP. Click on the "jr.Live!" button.



Then enable the GUI with the checkbox.


jr.Live! Setup

Graphical User Interface

☒ Enable GUI

Directory: C:\jrLive!

Console

Interface: None

Supervisor Password

Password:

Ctrl-R to reveal

By defining a password you can restrict exiting the jr.Live! GUI to authorized persons. If a password is set you will be asked for a password any time you try to exit the jr.Live! GUI for the normal Lasergraph DSP interface.

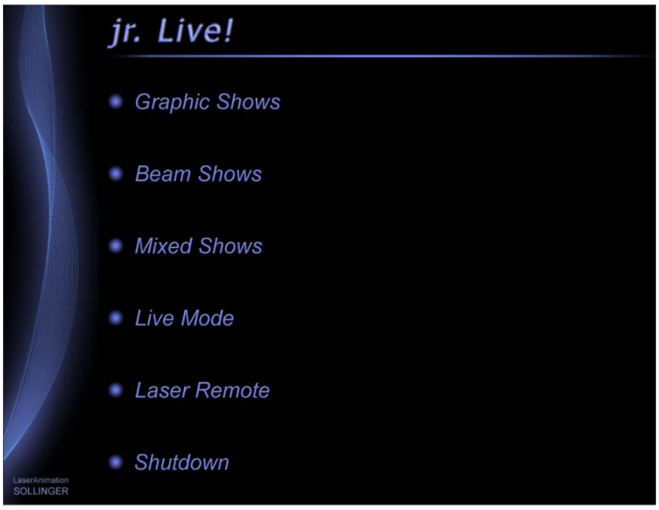
Exit the jr.Live GUI by using the keys <ctrl + alt + backspace> or <ctrl + F12>

Note: The "Console" input is for future use only.

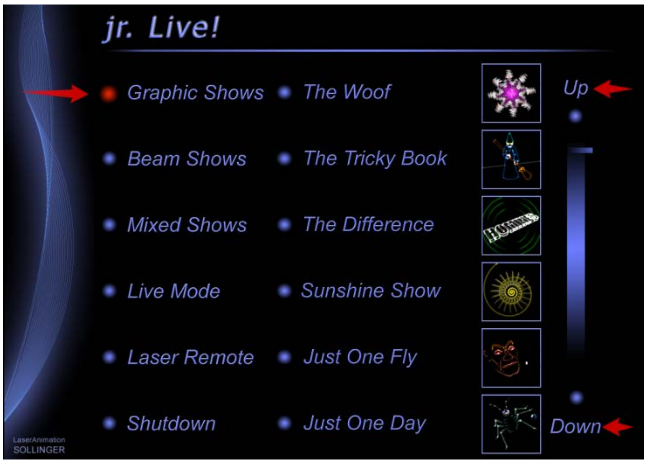
Using the jr.Live!

When the jr.Live! Interface has started you will see six categories to choose from:

- Graphic Shows
- Beam Shows
- Mixed Shows
- Live Mode
- Laser Remote
- Shutdown



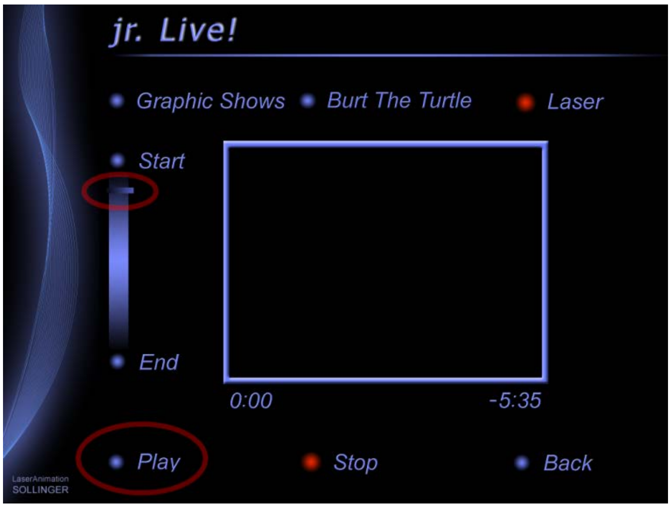
The currently active category is marked with a red dot.



In each category there can be one or more shows.
By clicking on the show picture or the name the show will be loaded into the memory.

If there are more than 6 shows in one category you can scroll through the category by clicking on the Up / Down buttons.

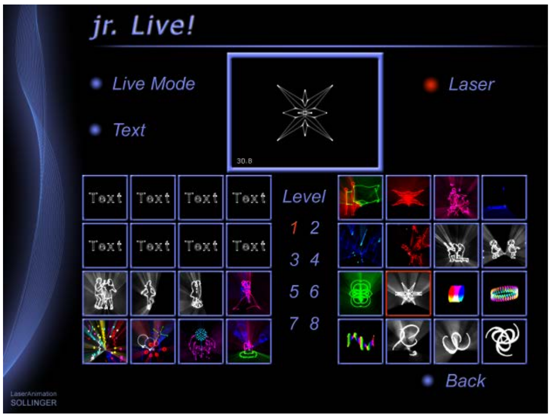
Once the loading is finished, you can press the "Play" button to start the show.



By dragging the position marker on the side bar you can scroll forward and backward through the show.

In addition to the three show directories containing all the shows you can select the following:

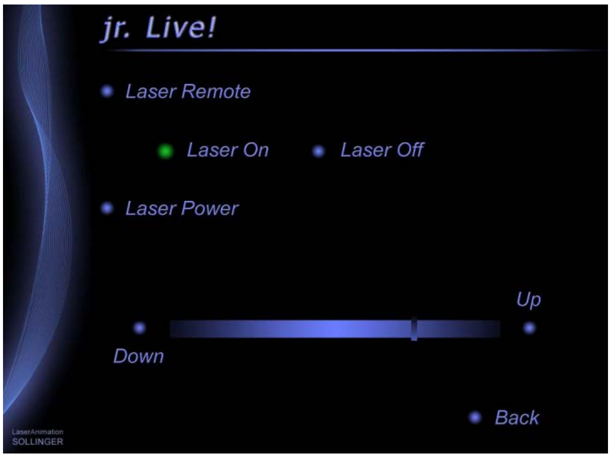
Live Mode
→ loads the Live.BOOT into the memory and shows the Live interface.



The currently active effect is marked with a red border.

If the Live Mode is not needed, you can delete the bitmap in the directory “C:\jr.Live!\Bitmaps” so it will not be displayed and the mode is not selectable.

Laser Remote
→ opens the page where a laser which is assigned in the Lasergraph DSP Setup can be turned on/off and the power can be set. The first ("Laser 1") of the four possible lasers will be used in the jr.Live! GUI.



If the option of controlling the laser in the "jr. Live!" is not desired, the "main_laser.bmp" bitmap can be deleted from the "C:\jrLive!\Bitmaps" directory. In addition to the "Laser Remote" selection this will also delete the laser status displays from the show playback screen and the Live Mode.

Shutdown
→ This menu entry allows shutting down the "jr. Live!" following a safety requester.

Attention!
In addition to the Lasergraph DSP this will also shut down the PC!
If this option is not desired, the "main_shutdown.bmp" bitmap can be deleted from the "C:\jrLive!\Bitmaps" directory.

Note:
With the keyboard shortcut <ctrl> + <alt> + <i> it is possible to call up the "Net I/O Setup" window in the jr.Live! GUI.
This enables the user to change I/O - settings etc. without leaving the GUI (e.g. when the GUI is locked with a password).

Creating "jr. Live!" compatible Shows

As described previously the shows are divided into three categories. A show must be sorted into one of the three subdirectories “Graphics”, “Beams” or “Mixed”.

In addition to the BOOT itself this directory must also contain the music and a bitmap which, except for the file name extension, have to have the same name as the BOOT.

The size of the bitmap must be 450x100 pixels. In order that the bitmap fits in with the other shows and the overall design of the interface we recommend basing the new bitmap on one of the supplied show bitmaps and just changing the text and the preview picture (in the frame).

The texts were created in "Adobe® Photoshop" as follows:

- Font: Arial, Italic, 16Pt. at 300dpi
- Effects: Drop shadow (120°, 75% opacity) plus inner bevel, smooth
- Color: RGB values 103, 119, 243

Subsequently the text was reduced to 20% using "Paint".

For creating the preview pictures, LGPreview screen-shots can be used.
LGPreview has to be set to 96x96 Pixel. For graphics the line width should be set to 2.

In order for a show to be played back without problems and for the progress display to work correctly, the show should meet the following requirements:

1. The first line of the timescript should contain a comment stating the name of the show. In contrast to the name of the boot, the full name can be given here, i.e. more than 14 characters can be used. Spaces or special characters are possible.

2. The show itself should start with the following commands:

DefineMedia	Media: <File Name>
SetTimeMode	External source: Media Script source: Internal Auto Track: ON Write: Unchanged
PlayMedia	Parameter depending on the Show, usually: Start: 1:00:00,00 End: 1:59:00,00 Media ref time: 1:00:00,00 Script ref time: 1:00:00,00

3. The Show should end with the following commands:

SetTimeMode	External source: Unchanged Script source: Internal Auto Track: OFF Write: Unchanged
StopMedia	

In order to determine the running time of the show, all Entries, PressKey and ReleaseKey contained in the timescript are detected. If any intro or fading-out music is to be included in the progress display an entry command should be included at the beginning resp. the end of the show which states "Nochange" for all films.
This has no influence on the course of the show but it will be considered in determining the running time of the show.

Running Text

If you want to integrate one or more RunningTexts with the GlobalText function to let the jr.Live! user change the text inside the GUI, follow the instructions below:

Create the Trickfilm syntax as usual together with the GlobalText command.
Use the numbers 1 – 8 for the description in the GlobalText.

Because the jr.Live GUI shows the text in the preview bitmap, the GUI has to know which Entries have been used for the text effects.
This is done with a definition file that has to be stored in the Live.BOOT with the name "Text.cfg".

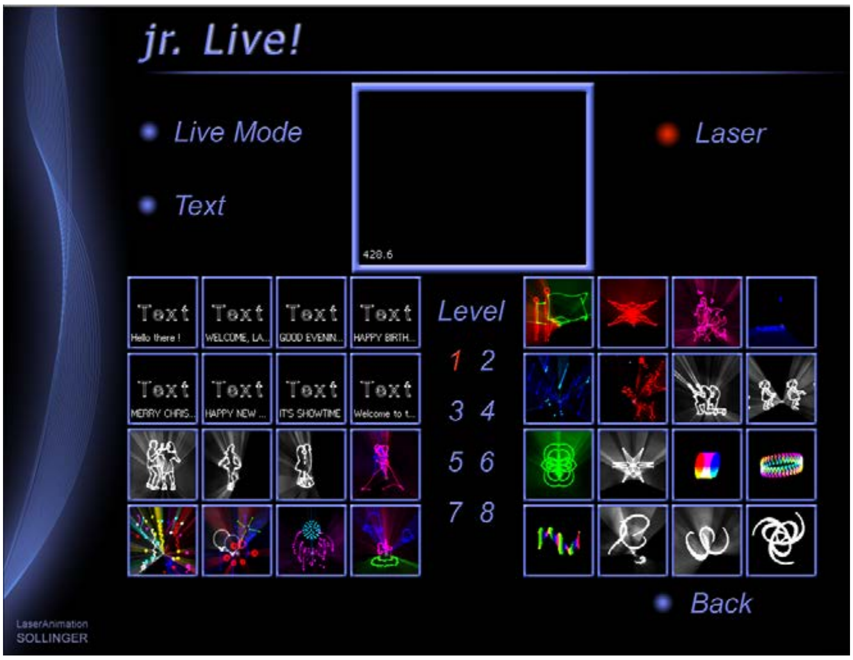
You can find the configuration file of the demo Live.Boot in the "Live.BOOT".
To change into a BOOT directory, use the command "ChangeDirectory" (shortcut "cd").

The Live.Boot in the jr.Live! supports 8 GlobalTexts.
In the default configuration the Live.Boot has 8 Text Entries in Level I (Entry 1 – 8) and 8 Text Entries in Level III (Entry 65 - 72).
The GlobalText is defined by entering the value 1 - 8 into the equivalent line and deleting the "-".

text syntax:
";" marks a comment
"-" marks an Entry without text
"1"-"8" marks an Entry with text and the value defines the GlobalText entry

For details see the "text.cfg" that can be found in the Live.BOOT.

Using the Live Mode that is delivered with the Mark 2



The Live Mode can contain up to 256 effects, organized in 8 levels with 32 effects each.

The “Live.Boot” that comes with Mark 2 is a free to use example for the Live Mode.
It contains 32 graphic effects in Level 1, 32 beam effects in Level 2 and 32 beat effects in Level 3.

By clicking on a Level number you select the corresponding Level.
By clicking on one of the pictures you start the corresponding effects.
Because the Live Mode is mainly made for "live" operation, a Midi controller (or a faderboard) is necessary to take advantage of all features of the Live Mode.

Midi / Fader Assignment:
16 faders are used to control brightness, speed etc. and 8 fader buttons are used to change colors.
Midi notes are used to call up the different effects through the Keyconfig.

Note:

There is a MIDI controller (CC 81) that toggles the Laser output On/Off.
With MIDI Controller 80 you enable/disable the frame selection for the beam effects.
This means, for all beam effects you can choose between 23 different pictures when the frame selection is enabled!
To get a preview of the pictures while an effect is running, use the MIDI controller 82, the frame preview.

Beat Level 3:
On Level 3 all effects move on a beat counter. The beat can be manually “tabbed” by pushing Midi Note #8 (the speed Flash button on Level 1 & 2).

With the Midi-Controller #31 you can define the direction of the effect moving (on the OHM64 it is the knob above the speed fader).

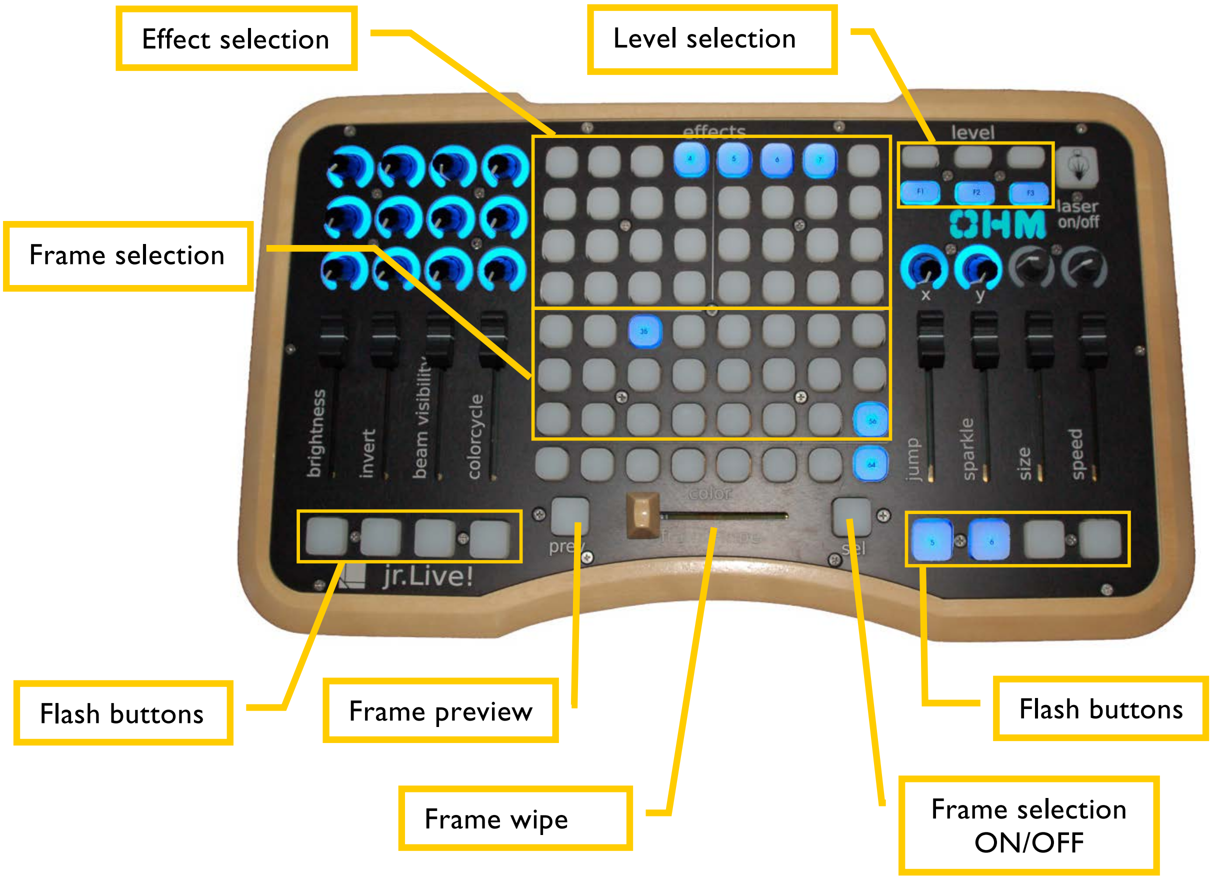
When the controller is < 0.5, the effect will move in one direction, when the controller is > 0.5 the effects moves back and forth.

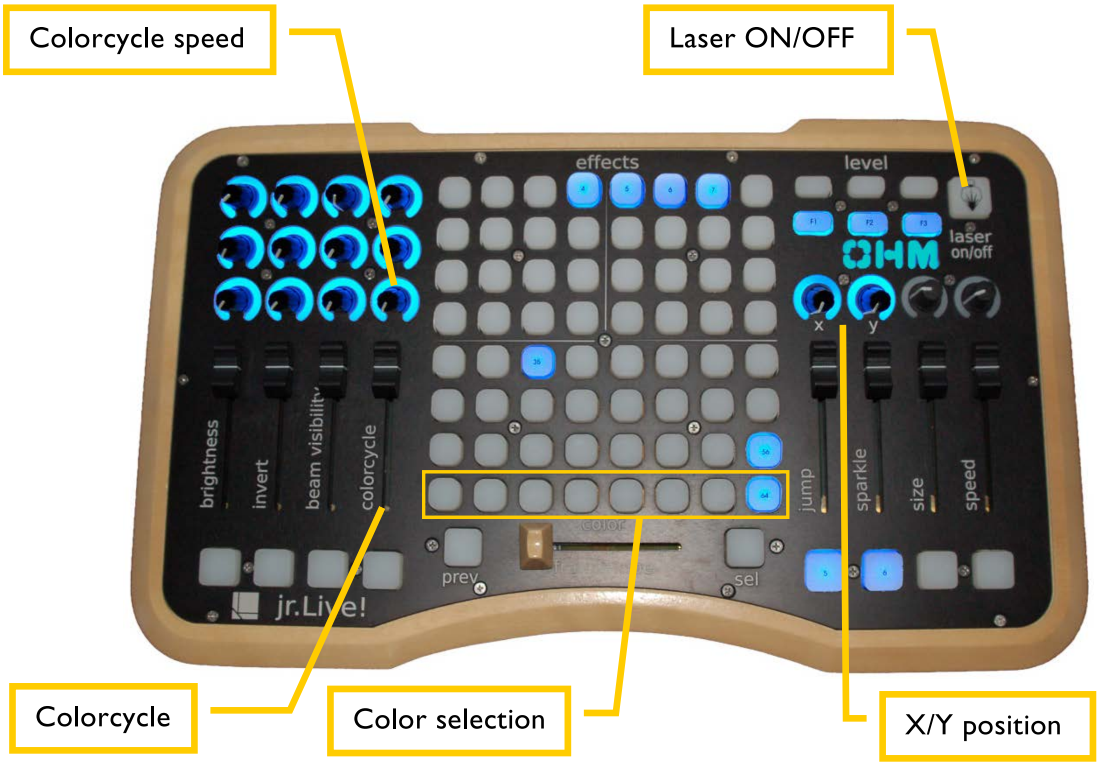
With Controller 7 (Effect speed on Level 1 & 2) you can choose between 3 different Beats. When the Controller is 1, the counter uses every bar, when it is 0.5 it uses every second bar and when controller 7 is 0 it uses every 4th bar.

MIDI Note #6 stops the movement, Midi Note #5 and #4 continue with different directions.

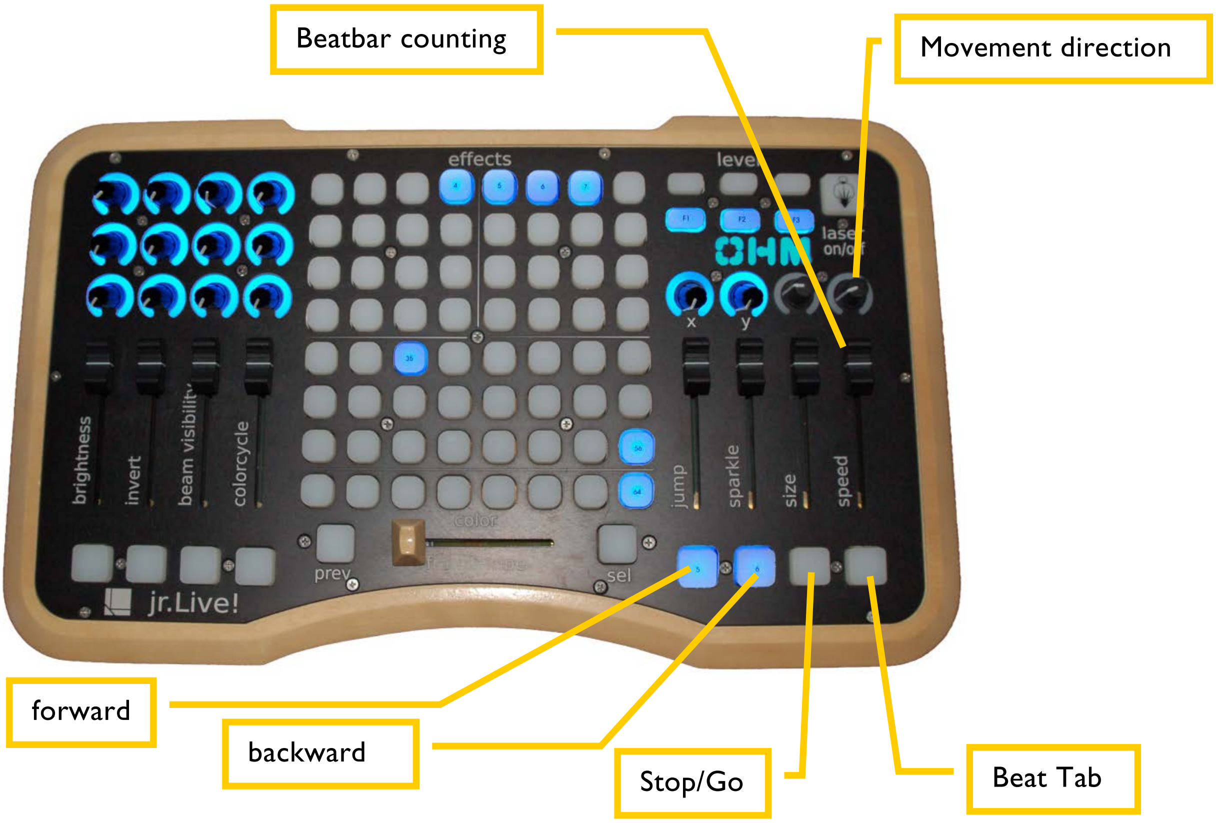
On the OHM64 controller these buttons correspond with the “size” Flash button, the “jump” Flash button and the “sparkle” Flash button.

See the assignment details in the attached pictures.

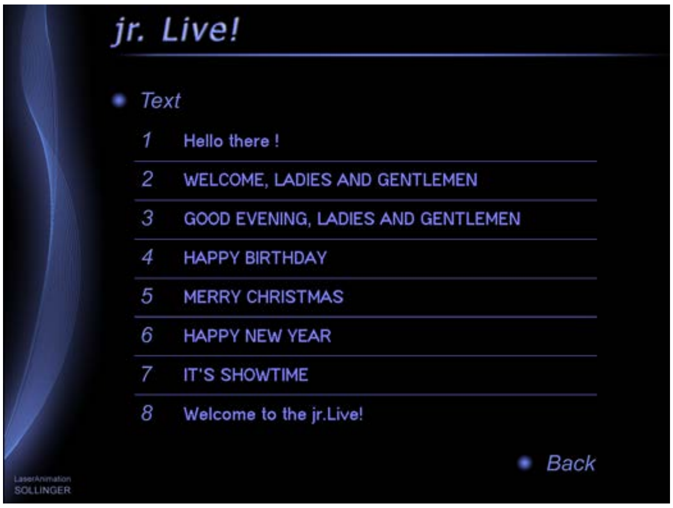




Special functions on Level 3:

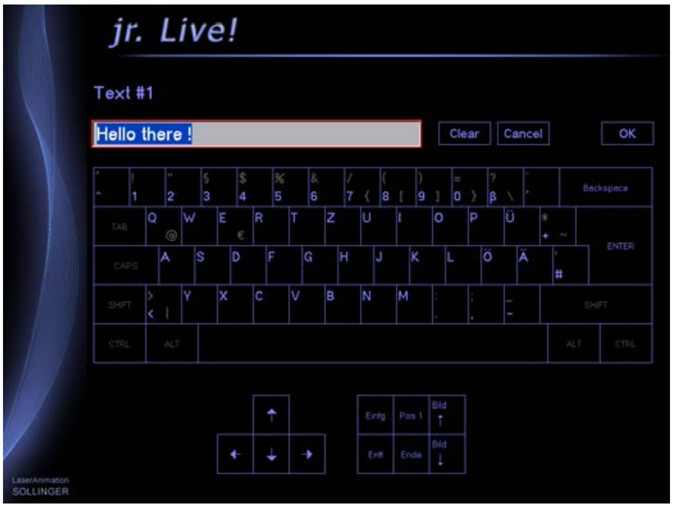


The RunningText Editor



1 Hello there !
2 WELCOME, LADIES AND GENTLEMEN
3 GOOD EVENING, LADIES AND GENTLEMEN
4 HAPPY BIRTHDAY
5 MERRY CHRISTMAS
6 HAPPY NEW YEAR
7 IT'S SHOWTIME
8 Welcome to the jr.Live!

By clicking on the button "Text" you activate the Text editor.
On the first page you choose which of the 8 texts you want to edit.
Just click on the text and you will get to the second page.



Text #1

Hello there !

Now you can use the computer keyboard or the virtual keyboard on screen to edit the text.
Once you press "enter" or "OK" the text will be saved.

MIDI / Fader Assignment

Faders / Controller Syntax

Faders	Midi-Controller	Action
0	0	Brightness 0 – 100 %
1	1	0 - 25 % Size X = 1 ; 25 - 50 % Size X = -1 50 – 75 % move forward ; 75 – 100 % move reverse
2	2	Beam visibility 0 – 100 %
3	3	Colorcycle: 0 – 75 % ; 75 – 100 % Cycle Speed both in combination with Fader 4
4	4	Number of colors in the cycle
5	5	Sparkle 0 – 100 %
6	6	Frame size 0 – 100 %
7	7	Effect speed 0 – 100 % (Level 1/2)
7	7	Beat counter 1-2-4 (Level 3)
8	8	Abstract Shape 1
9	9	Abstract Shape 2
10	10	Abstract color 1
11	11	Abstract color 2
14	14	Position X
15	15	Position Y
16	16	Frame wipe (between default and selected frame)
18	18	Invert Jump
31	31	Beat direction (one way / back and forth) only in Level 3

When Fader 4 (Number of color) is set to 0 and Fader 3 is somewhere between 75 – 100% the effect will get white.

By pressing a Fader button you can then choose a color which will be used for a color change from white to the selected color.

Fader Buttons	Midi-Controller	Action
0	64	Color Red
1	65	Color Orange
2	66	Color Yellow
3	67	Color Green
4	68	Color Cyan
5	69	Color Blue
6	70	Color Hot Pink
7	71	Color Default
16	80	Frame selection enable / disable
17	81	Laser Output enable / disable
18	82	Frame selection Preview !!!

Note:
When the Frame selection is off, the Frame Wipe will wipe between two different positions of the default frame.
When it is on, it wipes between the default and the user selected frame.
The preview button will show the selected frame on the preview output on the computer screen, it will not be shown on the real laser output!

Midi Notes Syntax

Level I Graphic Level

Midi-Note	Action	Midi-Note	Action
#36	DSP Keyboard Level I	#66	---------
#37	---------	#67	Preset Level 1 Effect 16
#38	DSP Keyboard Level II	#68	---------
#39	---------	#69	Preset Level 1 Effect 17
#40	DSP Keyboard Level III	#70	---------
#41	Preset Level 1; Effect 1 (Text 1)	#71	Preset Level 1 Effect 18
#42	Frame selection	#72	Preset Level 1 Effect 19
#43	Preset Level 1; Effect 2 (Text 2)	#73	---------
#44	---------	#74	Preset Level 1 Effect 20
#45	Preset Level 1; Effect 3 (Text 3)	#75	---------
#46	---------	#76	Preset Level 1 Effect 21
#47	Preset Level 1; Effect 4 (Text 4)	#77	Preset Level 1 Effect 22
#48	Preset Level 1; Effect 5 (Text 5)	#78	---------
#49	---------	#79	Preset Level 1 Effect 23
#50	Preset Level 1; Effect 6 (Text 6)	#80	---------
#51	---------	#81	Preset Level 1 Effect 24
#52	Preset Level 1; Effect 7 (Text 7)	#82	---------
#53	Preset Level 1; Effect 8 (Text 8)	#83	Preset Level 1 Effect 25
#54	---------	#84	Preset Level 1 Effect 26
#55	Preset Level 1 Effect 9	#85	---------
#56	---------	#86	Preset Level 1 Effect 27
#57	Preset Level 1 Effect 10	#87	---------
#58	---------	#88	Preset Level 1 Effect 28
#59	Preset Level 1 Effect 11	#89	Preset Level 1 Effect 29
#60	Preset Level 1 Effect 12	#90	---------
#61	---------	#91	Preset Level 1 Effect 30
#62	Preset Level 1 Effect 13	#92	---------
#63	---------	#93	Preset Level 1 Effect 31
#64	Preset Level 1 Effect 14	#94	---------
#65	Preset Level 1 Effect 15	#95	Preset Level 1 Effect 32

Midi-Note	Action	Midi-Note	Action
#01	Flash brightness	#05	Flash jump
#02	Flash invert	#06	Flash sparkle
#03	Flash Beam visibility	#07	Flash size
#04	Flash colorcycle	#08	Flash speed

Level 2 Beam Level

Midi-Note	Action	Midi-Note	Action
#36	DSP Keyboard Level I	#66	Frame selection Frame 11
#37	------	#67	Preset Level 2 Effect 16
#38	DSP Keyboard Level II	#68	Frame selection Frame 12
#39	------	#69	Preset Level 2 Effect 17
#40	DSP Keyboard Level III	#70	Frame selection Frame 13
#41	Preset Level 2 Effect 1	#71	Preset Level 2 Effect 18
#42	Frame selection Frame 1	#72	Preset Level 2 Effect 19
#43	Preset Level 2 Effect 2	#73	Frame selection Frame 14
#44	Frame selection Frame 2	#74	Preset Level 2 Effect 20
#45	Preset Level 2 Effect 3	#75	Frame selection Frame 15
#46	Frame selection Frame 3	#76	Preset Level 2 Effect 21
#47	Preset Level 2 Effect 4	#77	Preset Level 2 Effect 22
#48	Preset Level 2 Effect 5	#78	Frame selection Frame 16
#49	Frame selection Frame 4	#79	Preset Level 2 Effect 23
#50	Preset Level 2 Effect 6	#80	Frame selection Frame 17
#51	Frame selection Frame 5	#81	Preset Level 2 Effect 24
#52	Preset Level 2 Effect 7	#82	Frame selection Frame 18
#53	Preset Level 2 Effect 8	#83	Preset Level 2 Effect 25
#54	Frame selection Frame 6	#84	Preset Level 2 Effect 26
#55	Preset Level 2 Effect 9	#85	Frame selection Frame 19
#56	Frame selection Frame 7	#86	Preset Level 2 Effect 27
#57	Preset Level 2 Effect 10	#87	Frame selection Frame 20
#58	Frame selection Frame 8	#88	Preset Level 2 Effect 28
#59	Preset Level 2 Effect 11	#89	Preset Level 2 Effect 29
#60	Preset Level 2 Effect 12	#90	Frame selection Frame 21
#61	Frame selection Frame 9	#91	Preset Level 2 Effect 30
#62	Preset Level 2 Effect 13	#92	Frame selection Frame 22
#63	Frame selection Frame 10	#93	Preset Level 2 Effect 31
#64	Preset Level 2 Effect 14	#94	Frame selection Frame 23
#65	Preset Level 2 Effect 15	#95	Preset Level 2 Effect 32

Midi-Note	Action	Midi-Note	Action
#01	Flash brightness	#05	Flash jump
#02	Flash invert	#06	Flash sparkle
#03	Flash Beam visibility	#07	Flash size
#04	Flash colorcycle	#08	Flash speed

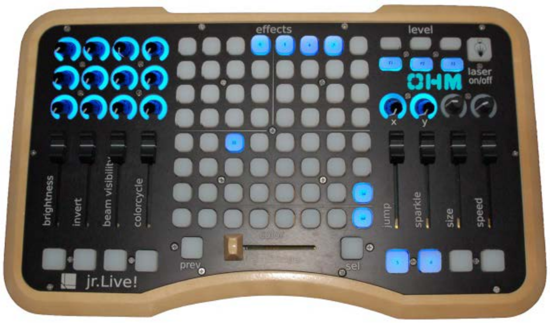
Level 3 Beat Level

Midi-Note	Action	Midi-Note	Action
#36	DSP Keyboard Level I	#66	Frame selection Frame 11
#37	------	#67	Preset Level 3 Effect 16
#38	DSP Keyboard Level II	#68	Frame selection Frame 12
#39	------	#69	Preset Level 3 Effect 17
#40	DSP Keyboard Level III	#70	Frame selection Frame 13
#41	Preset Level 3 Effect 1	#71	Preset Level 3 Effect 18
#42	Frame selection Frame 1	#72	Preset Level 3 Effect 19
#43	Preset Level 3 Effect 2	#73	Frame selection Frame 14
#44	Frame selection Frame 2	#74	Preset Level 3 Effect 20
#45	Preset Level 3 Effect 3	#75	Frame selection Frame 15
#46	Frame selection Frame 3	#76	Preset Level 3 Effect 21
#47	Preset Level 3 Effect 4	#77	Preset Level 3 Effect 22
#48	Preset Level 3 Effect 5	#78	Frame selection Frame 16
#49	Frame selection Frame 4	#79	Preset Level 3 Effect 23
#50	Preset Level 3 Effect 6	#80	Frame selection Frame 17
#51	Frame selection Frame 5	#81	Preset Level 3 Effect 24
#52	Preset Level 3 Effect 7	#82	Frame selection Frame 18
#53	Preset Level 3 Effect 8	#83	Preset Level 3 Effect 25
#54	Frame selection Frame 6	#84	Preset Level 3 Effect 26
#55	Preset Level 3 Effect 9	#85	Frame selection Frame 19
#56	Frame selection Frame 7	#86	Preset Level 3 Effect 27
#57	Preset Level 3 Effect 10	#87	Frame selection Frame 20
#58	Frame selection Frame 8	#88	Preset Level 3 Effect 28
#59	Preset Level 3 Effect 11	#89	Preset Level 3 Effect 29
#60	Preset Level 3 Effect 12	#90	Frame selection Frame 21
#61	Frame selection Frame 9	#91	Preset Level 3 Effect 30
#62	Preset Level 3 Effect 13	#92	Frame selection Frame 22
#63	Frame selection Frame 10	#93	Preset Level 3 Effect 31
#64	Preset Level 3 Effect 14	#94	Frame selection Frame 23
#65	Preset Level 2 Effect 15	#95	Preset Level 3 Effect 32

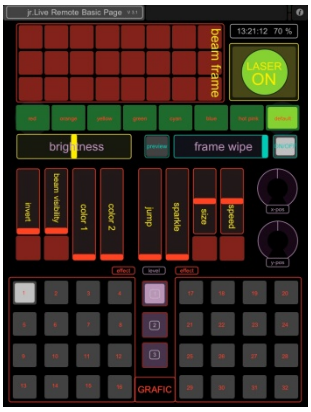
Midi-Note	Action	Midi-Note	Action
#01	Flash brightness	#05	Flash jump
#02	Flash invert	#06	Flash sparkle
#03	Flash Beam visibility	#07	Flash size
#04	Flash colorcycle	#08	Beat Tab

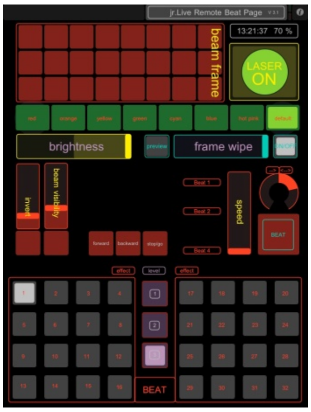
As a MIDI controller we recommend the OHM64 or the iPad with TouchOSC installed.

OHM 64



iPad with TouchOSC





For details how to control the Live Mode with the OHM64 and iPad see our video-tutorial on our web page.

Notes:

- Ohm64 is a registered trademark of Livid Instruments.
- iPad is a registered trademark of Apple computers.

Overview

The LGFD (Lasergraph for Discotheques) was created to provide an easy to use CD-playback system.
You activate the LGFD by entering "Setup" and clicking on the button "LGFD". The LGFD mode can be terminated using the key combination <CTRL + F12>.

CD as Music and Showmaster (Workstation only)

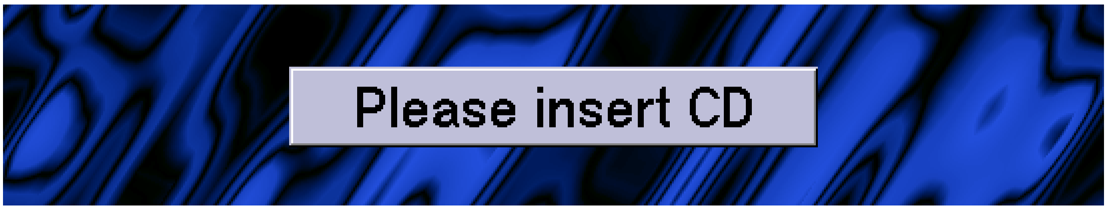
A user who wants to display a show only needs to insert a CD and press a play button.
Each of these CDs can contain one or more shows including the corresponding music. In order for the shows to be correctly recognized they have to be stored as BOOTs in the root directory of the first data track of the CD or in subdirectories.

PC as a Music and Fileserver

As an alternative to the CD it is also possible to use a PC with an activated WAV-server (LGServer - 'Wav/Avi') as the source for music and shows. From the point of view of the user, the LGFD acts in this case exactly as it would with a CD. Therefore there will be no differentiation between CD and WAV-server in the following explanations.

Operation

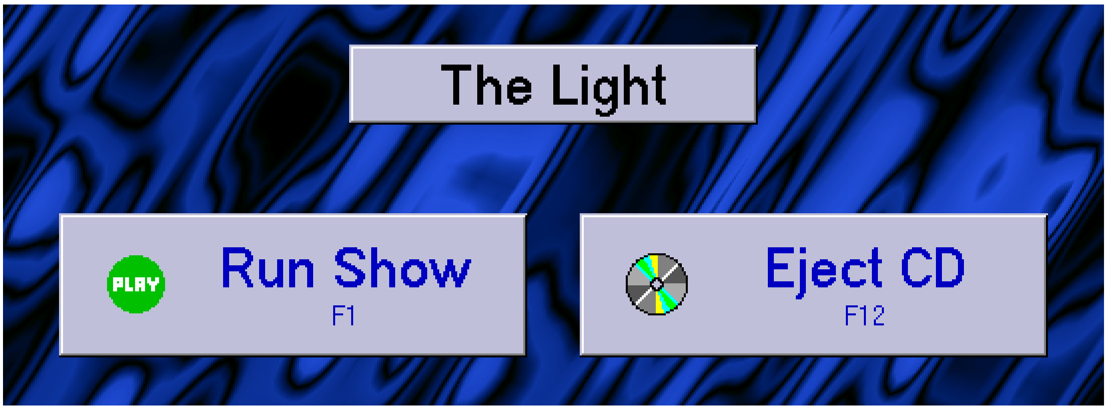
A Lasergraph DSP with activated LGFD asks for a CD after it is turned on:



Please insert CD

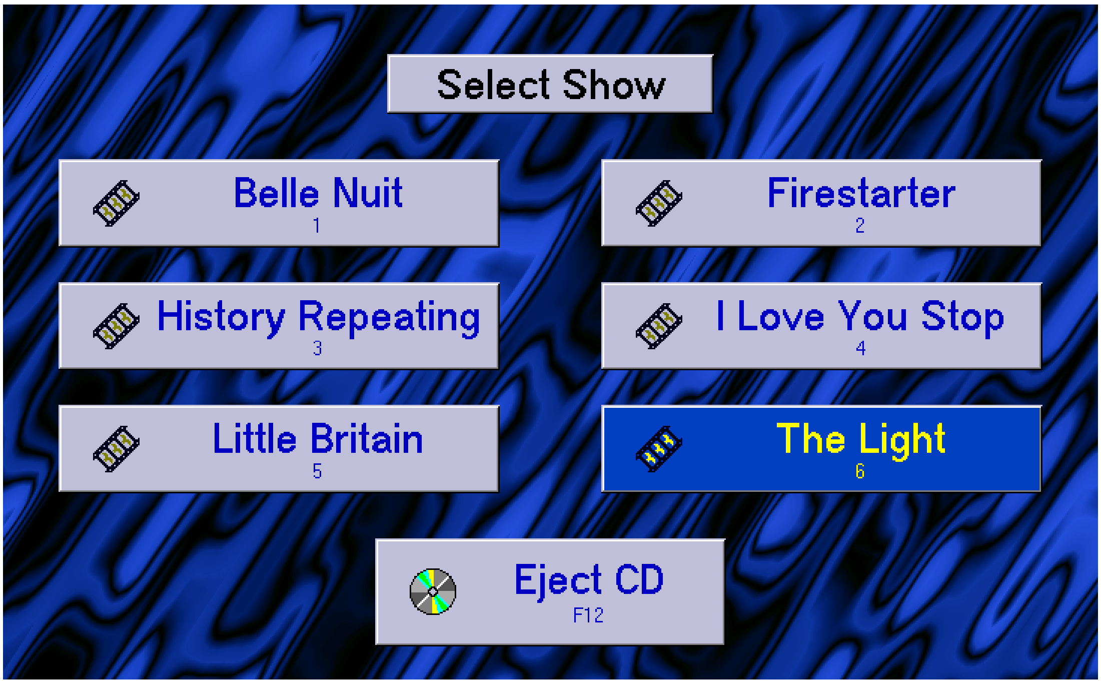
CD contains one show

If the CD contains only one show, this show is automatically loaded and the user can start the show or eject the CD:

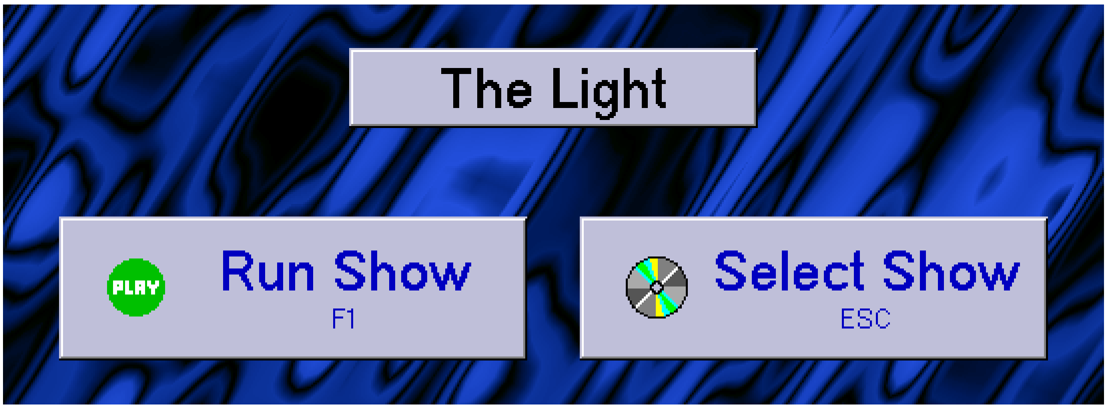


CD contains more than one show

If the CD contains more than one BOOT in the root directory of the CD, the user has to select the desired show:

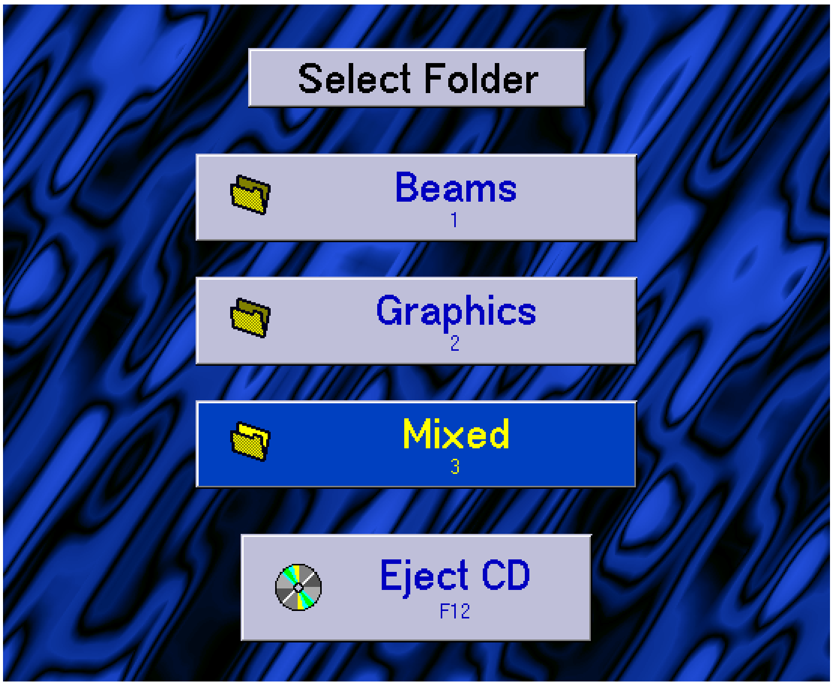


After loading the show, the user has the possibility to start the show or to return to the show selection ("Select Show"):

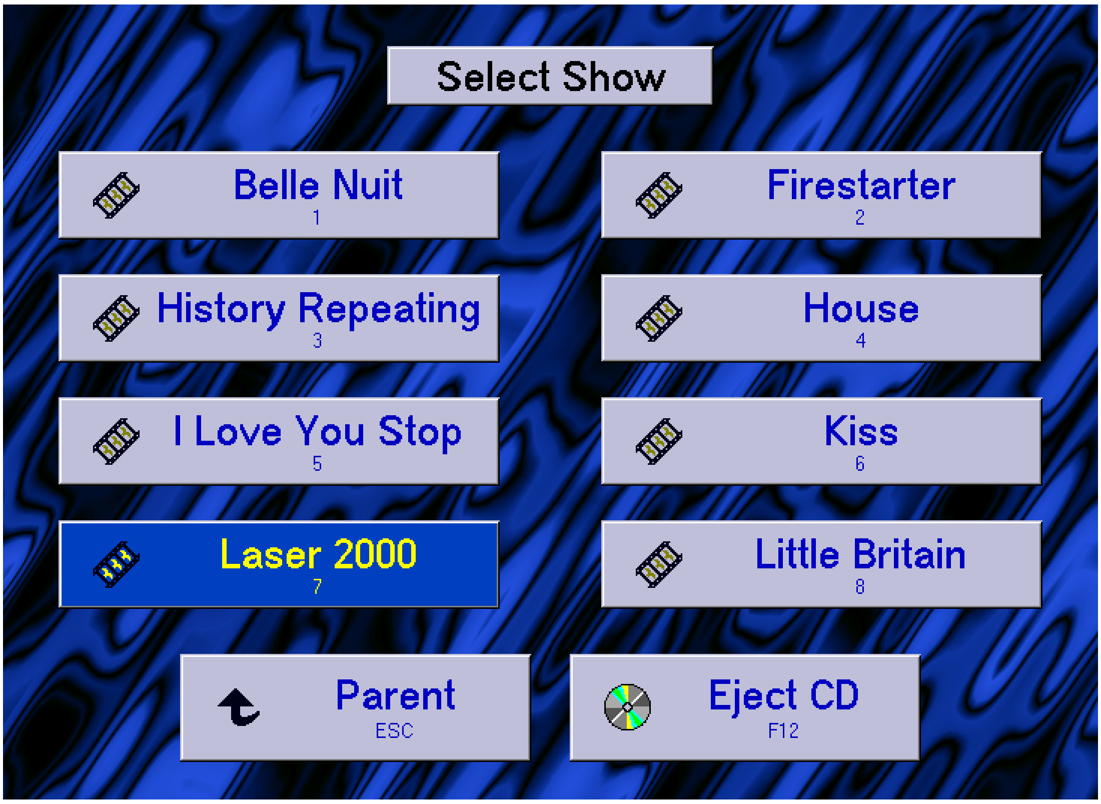


CD contains shows in subdirectories

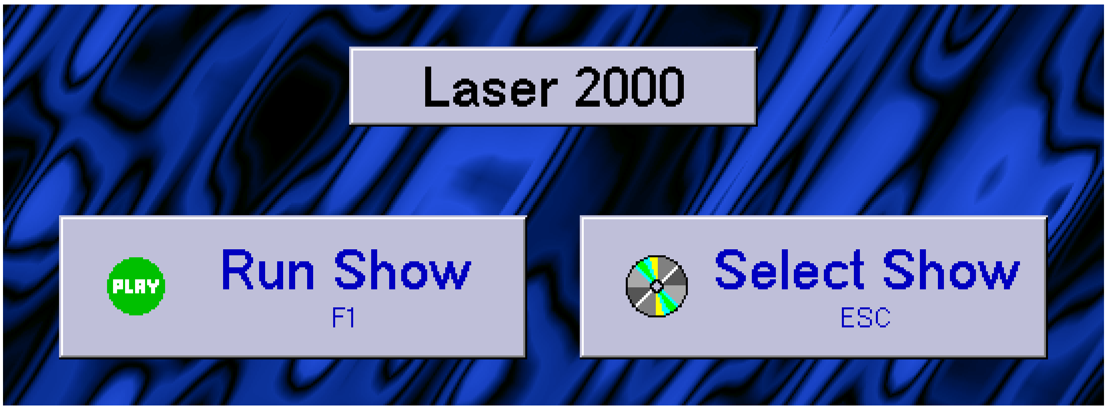
If the CD contains no BOOTs in the root directory but there are subdirectories, a directory selection list ("Select Folder") appears.



After selection of the subdirectory (e. g. "Mixed") which contains shows stored as BOOTs you can select the desired show:



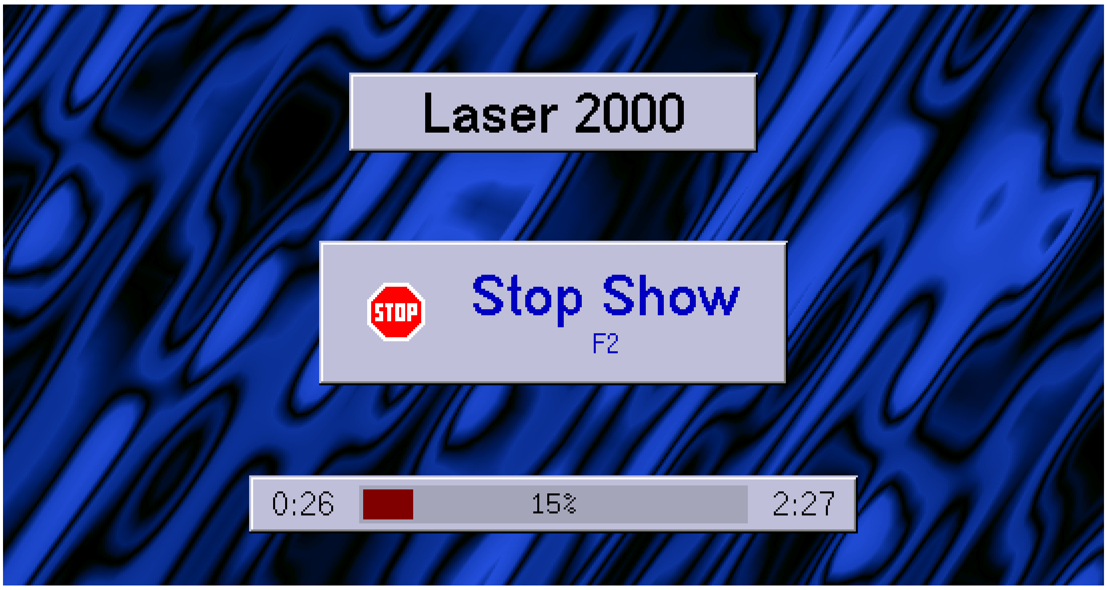
The show will be started with "Run Show" or you can return to "Select Show" respectively:



Note: The directory selection will only be available when the current directory contains no shows but only subdirectories. Then the subdirectories can be browsed until no more subdirectories are available or until a subdirectory contains at least one show. In the latter case the browsing ends, which means that no further subdirectories will be displayed. There is only the option to return to a higher level ("Parent").

While the show is running, a progress bar displays the current position within the show.
To the left of the progress bar the elapsed time is displayed and to the right you will see the remaining time.

In addition there is always the possibility to interrupt the show with "Stop Show":



Some Tips for Creating a CD for LGFD

The name of a show as it is displayed in the selection list and as title can be defined with the first "Comment" in the timescript. If this option is not used, the file name of the show will be used instead.

If a BOOT is stored with the option "/AutoPlay", the show is started immediately after it is loaded, the selection "Run Show" does not appear.

The display of the progress bar depends on the commands used in the timescript. Therefore there should not be an unnecessarily long period of time between the last timescript command and the end of the show.

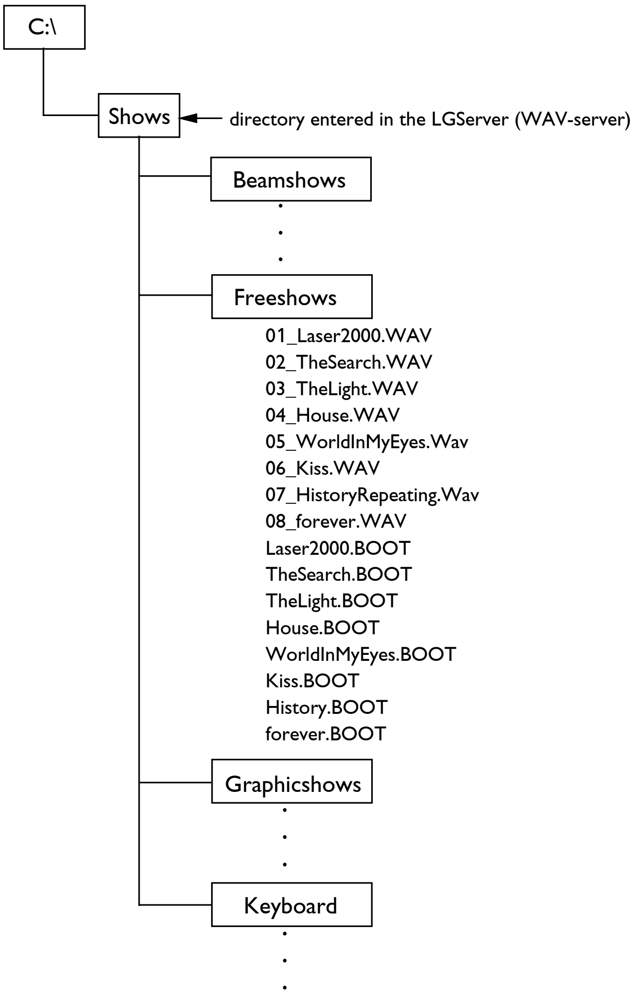
If a show contains a KeyConfig the usual display of the Midi keyboard appears at the lower edge of the screen. The user can use the Midi keyboard to start effects as usual.

If a show contains texts using the command "DeFineText" the button "Text" appears, which can be used to open the text table.

Since it is not possible to make any changes to a CD once it has been burned, there is a special mechanism which allows adapting a show to local conditions:
If the directory "C:\CD-Shows" contains a BOOT with the same file name as the one which is about to be loaded from CD, the BOOT stored on the harddisk will be used instead. This makes it possible e.g. to adapt globals without having to burn a new CD.

Music and show data are provided a little differently when using the WAV-server from the LGServer than with a CD:
A path which has a similar directory- / show structure as the one described for the CD is entered into the LGServer. The only difference is that the audio files (in WAV-, MP3 or Midi format) have to be located in the same directory as the show itself. After selecting such a directory a virtual CD is created based on the audio files found in the directory, which is then used to play back the shows. The order of the tracks is determined by the alphabetical sorting of the tracks on the PC. In order to guarantee a defined sequence it is recommended to add the desired track number preceding the track name.

A possible file structure on the PC is shown in the following example:



Introduction

The Lasergraph DSP supports two different concepts of color handling:

- Indexed colors
- True colors

Indexed Colors

When using indexed colors, the color of a point is defined by an index in a color table.
The DSP uses 64 different indexes, 4 palettes with 16 colors each

Advantage:
Color effects and color changes can easily be applied.

Disadvantage:
In one picture only 64 colors can be used simultaneously.

True Colors

Each point has its individual color.

Advantages:

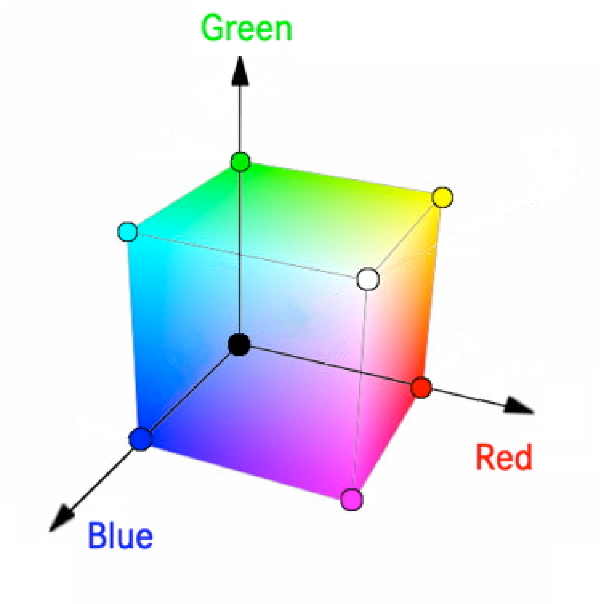
- A picture can have more than 64 colors.
- The colors used by an animation can differ from frame to frame.
- The color information is embedded in the picture. (Transfer between different systems)

Disadvantage:

- Color effects and color changes cannot be easily applied.

Color Models

R G B Model



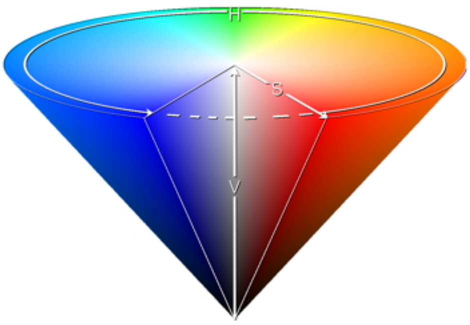
where R stands for red, G for green and B for blue.

H S V Model

where H stands for hue, S for saturation and V for value.

The Lasergraph DSP uses the HSV model.

Because the Piced does not support creation and editing of true color pictures, the only way to use true color pictures with the Lasergraph DSP is by importing pictures from ILDA files or by converting them by using the proTize converter or proTize max.



Importing True Color Pictures

ILDA frames

There are two commands to import ILDA frames into the DSP.

`LoadILDA` and `COnvert.ILDAtoDSP`

When you import ILDA pictures, choose from these options:

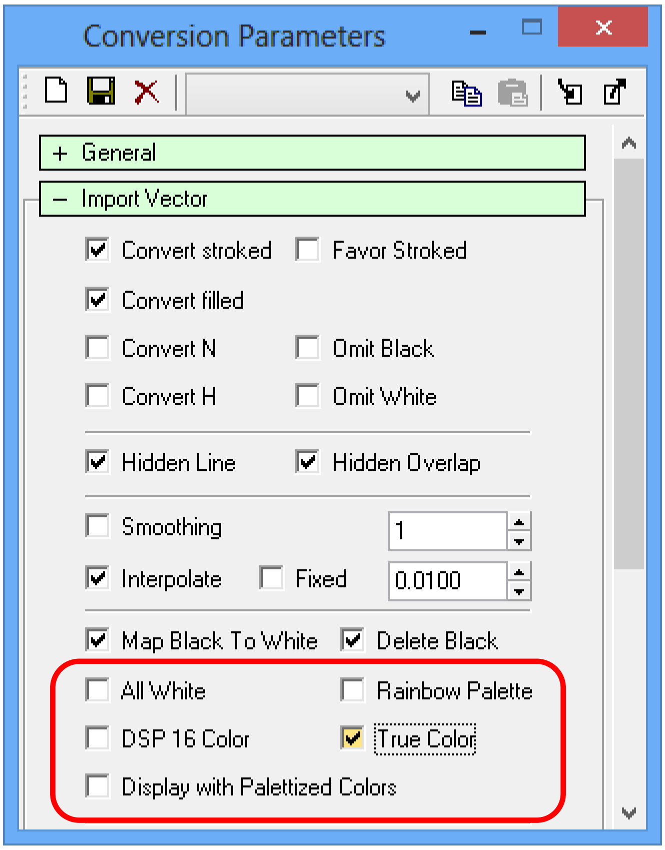
`/IgnorePalette`
to ignore the color palettes contained in the ILDA file

`/IgnoreRgb`
to ignore the color information of true color pictures

`/NoRgb`
to ignore any true color information

proTize converter

With the proTize converter you can create pictures with true color information.



☑ Convert stroked □ Favor Stroked
☑ Convert filled
□ Convert N □ Omit Black
□ Convert H □ Omit White
☑ Hidden Line ☑ Hidden Overlap
□ Smoothing 1
☑ Interpolate □ Fixed 0.0100
☑ Map Black To White ☑ Delete Black
□ All White □ Rainbow Palette
□ DSP 16 Color ☑ True Color
□ Display with Palettized Colors

In the Conversion Parameters you will find the parameter which determines the color handling during conversion.

In default mode (no option is enabled), the converter creates pictures with 64 colors - 16 colors in 4 palettes. The 4 palettes are used as follows:

1. Palette = Default Color Buffer 100%,
2. Palette = Default Color Buffer 75%,
3. Palette = Default Color Buffer 50%,
4. Palette = Default Color Buffer 25%

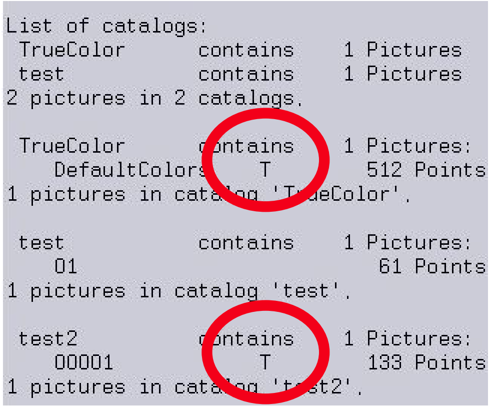
You can use the following options:

- **All White** will ignore all colors and will result in a white picture.
- **Rainbow Palette** will convert the picture to fit the default Rainbow palette of the Pangolin system.
- **DSP 16 Color** will reduce the usage of color to 16.
- **True Color** will create a True Color picture.

Note: Only the use of the option True Color will result in a picture with true color information.

You can identify a true color picture by using the command `LIst`. Choose a single catalog whose content will then be shown.

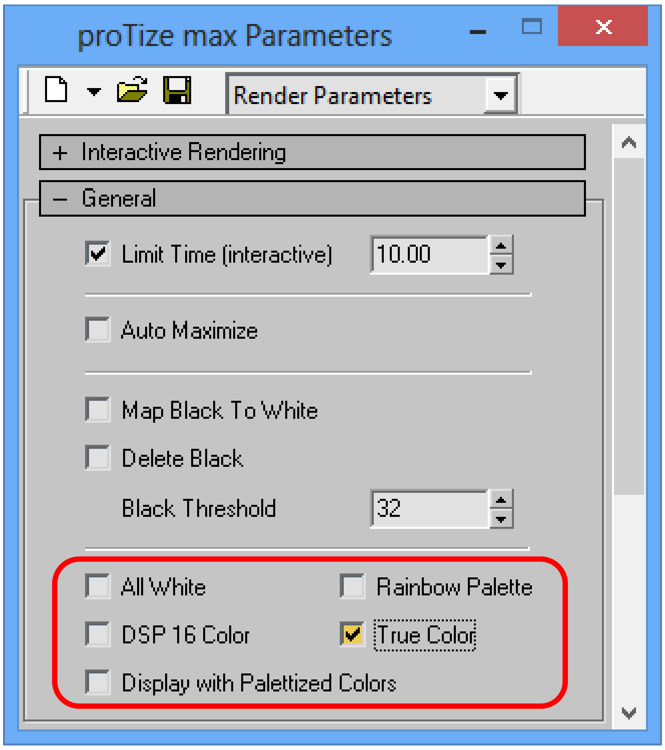
The "T" indicates that a picture contains true color information:



```
List of catalogs:
 TrueColor      contains    1 Pictures
 test           contains    1 Pictures
2 pictures in 2 catalogs.

 TrueColor      contains    1 Pictures:
    DefaultColors     T       512 Points
1 pictures in catalog 'TrueColor'.

 test           contains    1 Pictures:
    01                         61 Points
1 pictures in catalog 'test'.

 test2          contains    1 Pictures:
    00001             T       133 Points
1 pictures in catalog 'test2'.
```

proTize max

With the proTize max you can create pictures with true color information.
Enable the "True Color" option in the proTize max Parameters.



☑ Limit Time (interactive) 10.00

□ Auto Maximize

□ Map Black To White

□ Delete Black

Black Threshold 32

□ All White □ Rainbow Palette

□ DSP 16 Color ☑ True Color

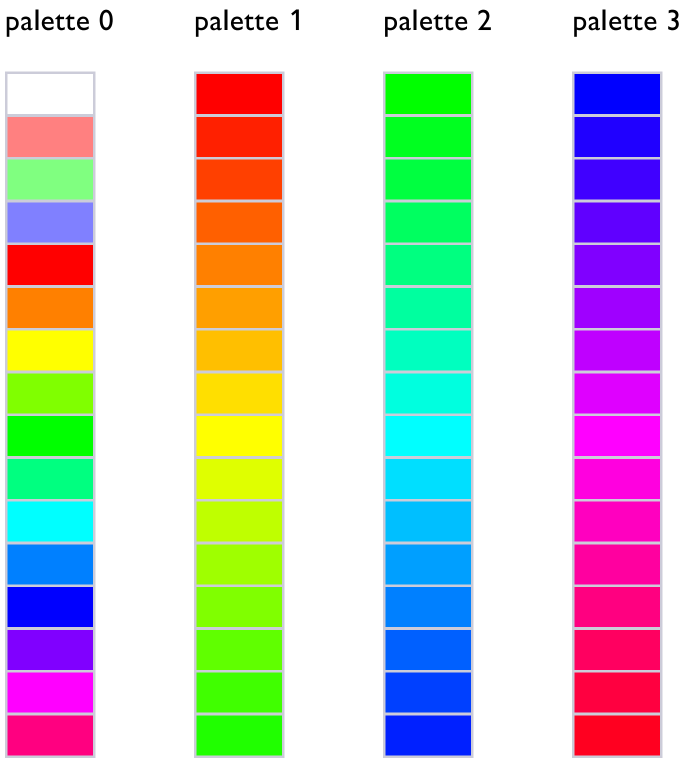
□ Display with Palettized Colors

Using True Color Pictures

To use true color pictures in the **Trickfilm** you have to use certain commands to define the true color.

define the palettes 0 - 3
define the saturation and intensity

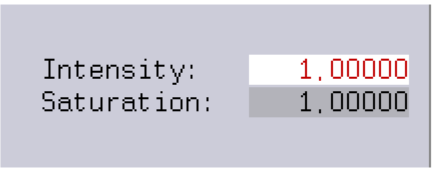
The palettes have to be set like this:



The colors in the buffers are used as follows:

palette 0:	only the first color (white) is used.
palette 1 - 3:	all colors are used red - yellow - green - cyan - blue - magenta - red

The saturation and intensity is set with the command `COlorControl`



```
Intensity:    1.00000
Saturation:   1.00000
```

Intensity: Defines the brightness of the colors.
value: 0 – 1

1 = max. brightness (default)
0 = black

Saturation: Defines the amount of gray in these colors.
Colors with more gray in it are less saturated.
value: 0 – 2

2 = max. saturation
1 = default saturation
0 = min. saturation (all white)

Note: Only the use of these palettes and the COlorControl command will result in the accurate display of a true color picture!

You can find templates for these settings here:

“C:/Demofiles/Templates” ⇨ “TrueColorAni.FILM” and “TrueColorPic.FILM”

or in the directory "DSP-Stuff" in the proTize installation directory.

Because the R/G/B components of the resulting colors are not necessarily assigned to the R/G/B laser lines, true color pictures can easily be adapted to non RGB laser systems like the BLITZ, BLIZZARD or PHAENON RGBY.

Limitations Using True Color Pictures

- True color pictures cannot be edited or created by the Piced.
- Some "CATalog" functions do not work correctly with true color pictures.
- The resolution of the intensity is limited to 7-bit.
- The resolution of the hue value is limited to 48 steps.

Mark 2 Manual

Chapter Tips & Tricks

Table of Contents

Commands which are not listed in the Menu

Commands which can be entered anywhere

`BEEP <time>`
activates the buzzer for the time which can be specified optionally (0.1s - 2.5s).

`CDROM.Info`
displays information about the current medium (number of tracks, length of tracks etc.).

`CDROM.PlayAudio <time>`
starts playback of the medium at the specified time.

`CDROM.STop`
stops playback of the medium.

`CDROM.Load`
loads a medium in the drive which was active last.

`CDROM.Eject`
ejects the current medium.

`CDROM.Pause`
pauses playback of the medium.

`CDROM.Resume`
resumes playback of the medium at the current position.

`CDROM.Select <devices>`
selects a medium. It is also possible to specify a PC running LGServer as the medium.

Example:

CDROM.Select \\#christian\CDRom.E
CDROM.Select \\#christian\Wav

`CLeaRHISTORY`
deletes all entries in the command line history.

`ColorSetup`
opens the "Display Color Setup".

`DGC.Open`
opens the DGC (Digital Geometric Correction) window.

`DGC.Close`
closes the DGC (Digital Geometric Correction) window.

`DGC.Select <set>`
selects the DGC setting for the current film.

`DGC.Reset [<hsize> [vsize]]`
resets the DGC to a user selectable size.

`<hsize>` horizontal size in range from -1.0 to +1.0
`<vsize>` vertical size in range from -1.0 to +1.0

```
DGC.Define.TopLeft <x> <y>
DGC.Define.TopRight <x> <y>
DGC.Define.BottomLeft <x> <y>
DGC.Define.BottomRight <x> <y>
```

Each of these commands positions one of the four corner points of the currently opened DGC.

`<x>` and `<y>` must be in range from -1.0 to +1.0

`DISABLEPREVIEW <dsp>` deactivates the output of the specified DSP in the preview window. If `<dsp>` is not specified the preview will be deactivated for all DSPs.

`DISABLEVISUALIZATION` stops the visualization in "LightConverse", "Depence V:3D", "Capture", "Realizzer 3D" or "wysiwyg".

`DMXBASE` displays the externally set DMX base address. If there is no external selector the base address is '1'.

`DMXOUTput` on the Compact Mark 2 this determines whether the DMX output puts out the values set locally by the Keyconfig and the corresponding trickfilm and timescript commands or the values from the DMX inputs of the servers specified in the "I/O Setup".

`ENABLELASEROUTPUT` activates resp. deactivates the laser output. Identical function as the corresponding button in the preview window.

`ENABLEPREVIEW <dsp>` activates the output of the specified DSP in the preview window. If `<dsp>` is not specified the preview will be activated for all DSPs.

`ENABLEVISUALIZATION <no> [<no> ... <no>]`

starts the visualization in "LightConverse", "Depence V:3D", "Capture", "Realizzer 3D" or "wysiwyg" depending on the selection in the Visualization Setup (see chapter "Setup").

`<no>` is the virtual laser number which can be set in the visualization software. Up to four laser numbers can be specified, one for each DSP.

`Film1` activates Film 1.
`Film2` activates Film 2.
`Film3` activates Film 3.
`Film4` activates Film 4.

`IOSETUP` opens the "I/O Setup".

`jrLIVE` opens the "jr.Live! Setup".

`LANtoSerial <interface>` establishes a connection between the LA-Toolbox and a PHAENON X (Pro or AT) or a Basic laser projector (PHAENON-, BLIZZARD- or BLITZ Basic) connected to an RS232 of the Lasergraph DSP:

LA-Toolbox $\xleftrightarrow[\text{Ethernet}]{}$ Lasergraph DSP $\xleftrightarrow[\text{RS232}]{}$ Laser

`<interface>` is the name of the RS232 interface to which the laser is connected (e.g. "RS232-2") or "None" to terminate the connection.

Notes:
- LANtoSerial communicates via TCP port 60000.
- It requires LA-Toolbox 2015/06/02 (or newer).

`LGFD.Open` starts the LGFD. If the LGFD has not been configured, the "LGFD Setup" will be opened.

`LGFD.Setup` opens the "LGFD Setup".

`LICenseINFO` displays the serial number, the software level, and the date up to which software updates can be installed. Also additionally installed licenses (e.g. "Lpv Creator", "proTize converter") will be displayed.

`MESSAGE.Open [<string>]` opens the window "Message Display".
Optionally a new message can be entered.

`MESSAGE.Close` closes the window "Message Display".

`MODEM.Window` opens the modem window.

`MODEM.DisConnect` disconnects an existing modem connection.

`MODEM.DIAL <number>` dials the specified number and establishes a connection to the modem at the other end of the line.

`MODEM.Upload` sends data over an existing modem connection.

`MODEM.Download` receives data from an existing modem connection.

`MODEM.TTY` activates the terminal emulation.

`MODEM.Config` selects the interface and the baud rate for the modem.

`PRImaryCOLors.Open` opens the primary colors window.

`PRImaryCOLors.Close` closes the primary colors window.

`PROTINFO` shows for each trickfilm, the timescript and the loaded catalogs whether they are protected and if they are, in which way.

`RequiredSystemSoftware` shows for each trickfilm and the timescript which version of the System Software is required to run this trickfilm resp. timescript.

`ROOT` takes you from any mode of operation straight to the start screen.

`Script1` activates the timescript.

`ScriptBUTTON` for more information about ScriptButtons see chapter "Scripting".

`SENDMIDI` sends up to 20 characters continuously over the Midi output to the connected periphery.
Note: This only works if "Midi" is selected under "Midi Out Mode" in the "Midi Out Setup".

`SENDstring <string>` sends a character sequence over RS232 or TCP/IP (see chapter "SerCmd").

`SETANALOGOUT <channel> <number> <value> [<value> ... <value>]`

sets up to 8 Analog outputs (Midi and DMX).

`<channel>` specifies the desired channel (1-8).

`<number>` defines the number of the first Analog output to be set (1-32).

`SetDateTime <date> <time>` sets the clock in the Lasergraph DSP.

`SetGlobalText <path> <description> <font> <text>`

enters a text into the global text table.

The parameters have to be entered with quotation marks.

Example:

SetGlobalText "" "Text 1" "2D_Alphabet" "Test!"

`SetGlobalValue <number> <value> [<value> ... <value>]`

sets up to 16 global values.

`<number>` defines the number of the first global value to be set.

Global values can be accessed in the trickfilm command "SetVariables" by using the function "GLOBALVALUE ()".

`SETIPADDR <addr> <mask>` provides the Lasergraph DSP with a new IP address and optionally a new subnet mask.

Note: The change will only take effect after a restart.

`SHAREDvar.Set <number> <value> [<value> ... <value>]`

sets up to 16 shared variables.

`<number>` defines the number of the first shared variable to be set (0-63).

`SHAREDvar.Name.Set <number> [<name> [<name> ... <name>]]`

labels up to 8 shared variables.

`<number>` defines the number of the first shared variable to be labeled (0-63).

`SHAREDvar.Name.Clear <number>`

removes the label from one shared variable.

`<number>` defines the number of the shared variable whose label is to be removed (0-63).

`SHAREDvar.Name.ClearAll` removes the labels from all shared variables.

`StartScreenSaver` starts the screen saver specified in the "Screen Saver Setup".

`VARINFO` opens the window "Variable Information" which can be used to display the states of the 64 shared variables.

Commands Available on the Start Screen

`JumpToTime <time>` sets the internal timecode to the specified time.

`SetTimeMode.Internal` selects "Internal" with AutoTrack disabled.

`SetTimeMode.External` selects "External" with AutoTrack disabled.

`SetTimeMode.AutoTrack` activates "AutoTrack" without changing the selection "Internal" / "External".

`SetTimeMode.AutoInt` selects "Internal" with AutoTrack enabled.

`SetTimeMode.AutoExt` selects "External" with AutoTrack enabled.

`SetTimeMode.ExtSource` selects the external timecode source (SmpteIn, Midi, Media, etc.)

Commands Available in the Trickfilm Edit

`CompileOld <film>` works similarly to the command "Compile". However, additional errors will be displayed if commands are used in the trickfilm which are not supported by old Lasergraph DSPs.

`ConvertSetColorBufferOld` converts all old SetColorBuffer commands (without primary colors) into new SetColorBuffer commands (with primary colors). The old SetColorBuffer commands will be retained for review purposes.

`DeleteDisplayZones` deletes all display zone commands in which no DGC zone is selected.
If a block is selected in the active trickfilm, the command will only affect the display zone commands within the selected block. Otherwise, the display zone commands in the whole trickfilm will be deleted.

`DeleteSetColorBufferOld` deletes all "SetColorBufferOld" commands in the current trickfilm.

`LOOPCNT` opens the "LoopCnt" window.

`W0.Open [FilmX or Script1]` opens the edit window W0 and brings it to the foreground.
`W0.Close` closes the edit window W0.

`W1.Open [FilmX or Script1]` opens the edit window W1 and brings it to the foreground.
`W1.Close` closes the edit window W1.

`W2.Open [FilmX or Script1]` opens the edit window W2 and brings it to the foreground.
`W2.Close` closes the edit window W2.

`W3.Open [FilmX or Script1]` opens the edit window W3 and brings it to the foreground.
`W3.Close` closes the edit window W3.

`W4.Open [FilmX or Script1]` opens the edit window W4 and brings it to the foreground.
`W4.Close` closes the edit window W4.

Commands Available in the Timescript Edit

`CompileOld`

works similarly to the command "Compile". However, additional errors will be displayed if commands are used in the timescript which are not supported by old Lasergraph DSPs.

`FindEntry [<entry> [<film>]] /Up`

finds an entry command in the timescript.

`<entry>` is the number of the entry to be found. If no number is specified, any entry command will be searched for.

`<film>` is the trickfilm in which the specified entry is called up. If no film is specified a call for the specified entry in any trickfilm will be searched for.

The option `/Up` causes the timescript to be scanned towards the top instead of the bottom.

`ImportEntries <filename>`

inserts empty Entry commands with timecode coming from a text file (e.g. *.CSV) at the current position in the timescript.
It supports timecode with 25fps in the following format: `hh:mm:ss:ff` or `hh:mm:ss,ff`

`W0.Open [FilmX or Script1]`

opens the edit window W0 and brings it to the foreground.

`W0.Close`

closes the edit window W0.

`W1.Open [FilmX or Script1]`

opens the edit window W1 and brings it to the foreground.

`W1.Close`

closes the edit window W1.

`W2.Open [FilmX or Script1]`

opens the edit window W2 and brings it to the foreground.

`W2.Close`

closes the edit window W2.

`W3.Open [FilmX or Script1]`

opens the edit window W3 and brings it to the foreground.

`W3.Close`

closes the edit window W3.

`W4.Open [FilmX or Script1]`

opens the edit window W4 and brings it to the foreground.

`W4.Close`

closes the edit window W4.

Commands Available in the File Mode

`BURNRomImage <filename>` updates the system software.

`ChangeDevDir <directory>` changes to a directory on the specified drive.

`DiskCheck <device>` checks the specified drive for logical errors and, if desired, searches for defective sectors.

`Info.Device <device>` displays information about the specified logical drive.

`Info.Scsi <device>` displays information about the specified SCSI drive.

`Info.IDE <device>` displays information about the specified IDE drive.

`LIstR <catalog>` works similarly to the command "List". However, with "LIstR" all pictures in all catalogs will be listed recursively.

`SUBST <device> <path>` temporarily assigns a drive letter to a network drive.

`Verify <source> <dest>` verifies files and directories together with their subdirectories.

How to Create Downwards Compatible Shows

When creating shows on a Mark 2 that will be played back on an old Lasergraph DSP (*easy*, *progressive*, *superior*) or on a Mark 2 with an older version of the System Software, you have to pay attention to the commands that you use. Some of the commands featured in the Mark 2 are not available on older Lasergraph DSPs and older Mark 2 units.

Nevertheless, it is quite easy to create downwards compatible shows.

Here is a list of commands that must not be used in the **Film**:

- `Entry` (with description), available since System Software Version 2009-09-23
  use the command `EntryOld` (shortcut `eo`) instead.
  The Trickflm Edit command `ConvertEntryNew` converts all new `Entry` commands (with description) into old `Entry` commands (without description).

- `Track` (with description), available since System Software Version 2010-11-18
  use the command `TrackOld` (shortcut `to`) instead.
  The Trickflm Edit command `ConvertTrackNew` converts all new `Track` commands (with description) into old `Track` commands (without description).

- `LaserPreview`, available since System Software Version 2009-09-23

- `SetRepeatMap`, available since System Software Version 2011-03-28
  As long as no variables or curves are used the command `SetRepeatMapOld` (shortcut `srmo`) can be used instead.

- `WaitForSync`, available since System Software Version 2011-03-28
  The selection `AnyEdge` requires System Software Version 2016-03-22 or later.

- `SetBeamColor`, available since System Software Version 2011-05-17
  use the command `SetCOlor` (shortcut `sco`) instead.

- `ApplyColorCycle`, available since System Software Version 2011-09-27

- `CycleBeamColor`, available since System Software Version 2011-10-24

- `Subroutine` (with description), available since System Software Version 2014-03-03
  use the command `EntrySubroutine` (shortcut `es`) instead.

- `DisplayPictureZone`, available since System Software Version 2014-03-03

- `DisplayBufferZone`, available since System Software Version 2014-03-03

- `DisplayAnimationZone`, available since System Software Version 2014-03-03

- `LimitFrameRate`, available since System Software Version 2016-10-17

- `ApplyIntensity`, available since System Software Version 2021-09-20

Also the following variable functions must not be used in the Film:

- "Fader()" and "FaderB()", available since System Software Version 2012-09-06
  If a show is to be programmed for an older DSP it is advisable to keep using the old query functions "Analog()" and "AnalogB()" which are still available for compatibility reasons. That is the only way to ensure that the show will be entirely editable on the old system.

- "DmxOut()" and "DmxOut16()", available since System Software Version 2015-01-15
  As long as no variables or curves are used the Trickfilm commands "SetAnalogs" and "SetAnalogSingle" can be used instead.

- "Array()", available since System Software Version 2015-01-15

- "ClipScale()", available since System Software Version 2016-03-22

In the **Timescript**, the following commands must not be used:

- `SetTimeMode` with External source "Time", available since System Software Version 2008-11-28.

- `Message`, available since System Software Version 2010-04-06.

To make sure that the show is executable on old Lasergraph DSPs, there is a special compile function that will check all used commands.
This special compile can be used by typing the command

```
CompileOld
```

into the command line followed by a <Return>.

To take advantage of the new `Entry` and `Track` commands even when programming a show for older Lasergraph DSPs we recommend the following procedure:

1. Program the show with new `Entry` and `Track` commands.

2. Save the show.

3. Convert all new `Entry` and `Track` command into old ones using the commands `ConvertEntryNew` and `ConvertTrackNew`.

4. Check if everything else is compatible with old systems using the command `CompileOld`.

5. Save the show again with a different name.

Beside the new Film and Timescript commands other enhancements are not available on old Lasergraph DSPs and might affect the behaviour of the show:

- The External Timecode source "Time” must not be used in the Root.
- The naming of the faders does not work on old Lasergraph DSPs.
- In the Keyconfig “Next” and “Prev” must not be entered as key action to start the next resp. preceding entry.
- External sync for 3D projection must not be selected in the ”Advanced display parameters”.
- Scanner Ratings other than 30k in the Super Preview setup are not supported by old Lasergraph DSPs and Mark 2 with a System Software Version older than 2012-01-27.
- Disabling "Auto Maximize" in the Super Preview setup is not supported by old Lasergraph DSPs and Mark 2 with a System Software Version older than 2015-09-07.

Streaming

What is streaming?

Streaming means that the pictures which are used by a show are not loaded into the memory together with the trickfilms and the timescript, but only at the time when they are actually needed during the course of the show. As soon as they are no longer needed they will be removed from the memory.

When does streaming make sense?

Using streaming always makes sense when a show contains large numbers of frame-by-frame animations, causing unnecessarily long loading times, or when the pictures used by the show will not fit into the memory of the Lasergraph DSP.

Animations with large numbers of pictures can for instance be the result of rendering scenes with proTitze.

Example: The show "BurtTheTurtle" delivered with the Mark 2 contains more than 9000 frames. Loading this show on a Compact takes approx. 45 seconds. The streaming version "BurtTheStream" (also included in the directory "C:\Shows\Streaming") takes less than 3 seconds!

It may even make sense to use streaming during the production of a show, even if the final show is not supposed to use streaming: Saving intermediate versions is much faster this way and takes up less space on the hard drive since it is not necessary to save all pictures each time.

How is streaming used?

In order to allow streaming an animation from hard disk or over the network it is necessary to create a Compressed Catalog (CCAT) which contains the animations which are to be streamed as well as all pictures referenced in the animation. This CCAT has to be located in the same directory as the film resp. the BOOT which is to access the pictures.

In order to change an existing show in such a way that it uses streaming simply load the show and use the command "SaveCOmpr" to create a CCAT containing all animations and pictures used by the show.

Afterwards the name of this CCAT has to be entered into all DisplayAnimation commands under the parameter "Stream from file".

Finally the pictures which are no longer needed can be deleted from the memory ("UnLoadAllcat") and the show can be saved with "CreateBoot" (preferably under a new name).

What has to be kept in mind for streaming

Streaming will only work properly if the data can be loaded fast enough. The local hard drive is usually faster than network access. This is particularly true for Lasergraph DSP workstations that are limited to 10 MBit network connections. With the Lasergraph DSP compact the network is still slower than the local hard drive, but the difference is small.

Streaming should only be used on one DSP at a time on a Lasergraph DSP workstation. Simultaneously streaming more than one DSP will reduce bandwidth and lead to visible interruptions.

Streaming only makes sense for animations with more than 50 images because, even if streaming is used, the DSP will preload about 50 images into the memory before they are needed. This is required to prevent dropouts. For the same reason streaming cannot be used if the images of an animation are accessed randomly. In that case the DSP has no chance of knowing in advance what is needed next and will fail to preload the images. This is one of the limitations of streaming. Fortunately most animations are played from the first image to the last image in a linear way.

If an animation is used several times within a film (or even across films) it might actually be better not to use streaming because each reference to an animation (e.g. DisplayAnimation command) is handled separately. For each reference the number of 50 preloaded images and other overhead applies. Therefore it might be better to avoid streaming in such cases.

It is possible to use different CCATs for different animations. But in general it is more efficient if the animations (and the corresponding images) are all combined into one CCAT. Otherwise the DSP needs to close and open files while playing back a show, which adds some overhead.

Streaming works best if images are not too small. It usually takes longer to load 100 images with 10 points than 10 images with 100 points, even though the total number of bytes to load is the same in both cases. There is a certain overhead involved in loading a frame that even applies if a frame contains only one point.

If no CCAT with the name specified under "Stream from file" is found at the location of the film resp. the BOOT, the directory "C:\General" will be scanned for it.

Soft Edge Projection

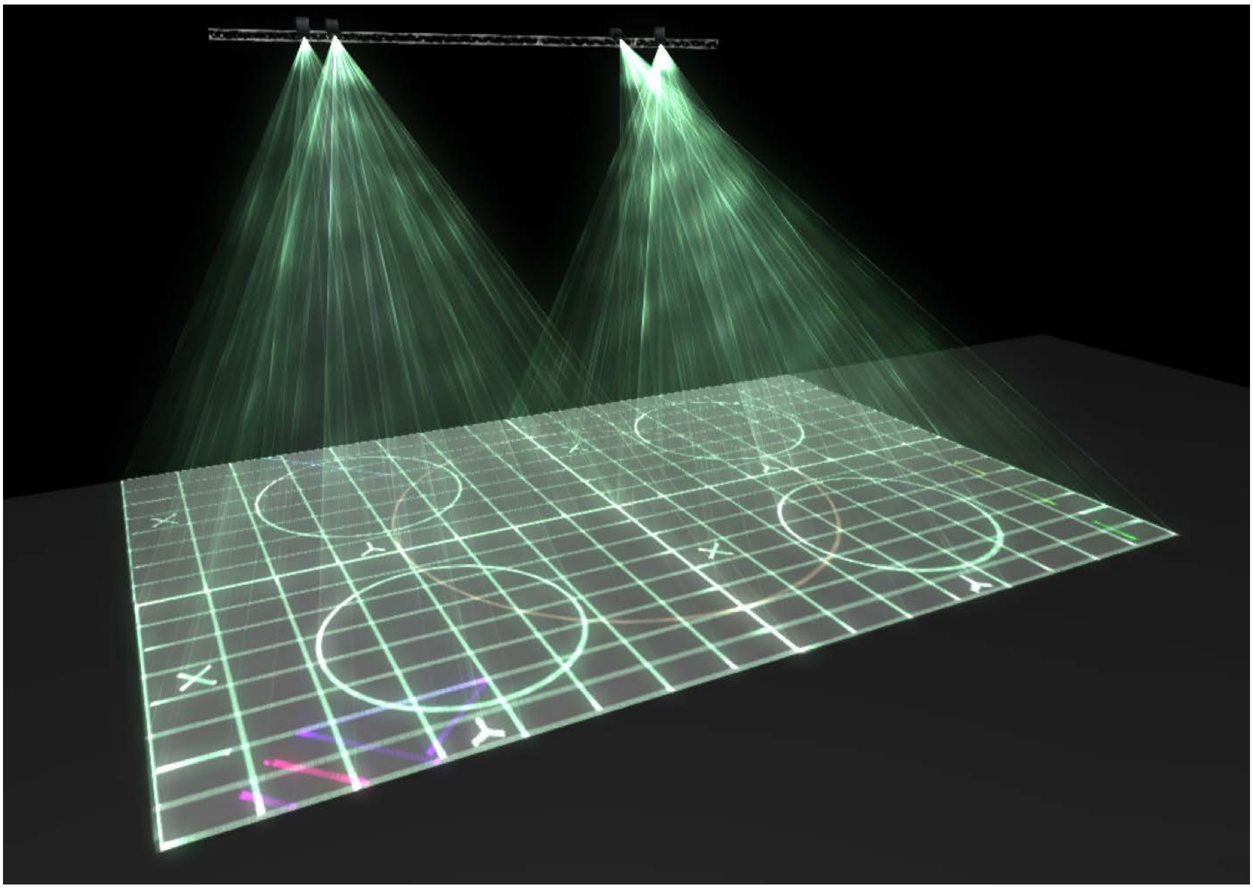
This guide will help you to combine two or more channels to project one larger picture with soft edge overlay.

In what cases is a combination of two or more channels indicated to achieve one larger image?

- short projection distances
- obstacles in the projection area

What is "Soft Edge"?

Soft edge is a blending technique to overlay two or more projections without seeing a hard edge where the projections overlay.

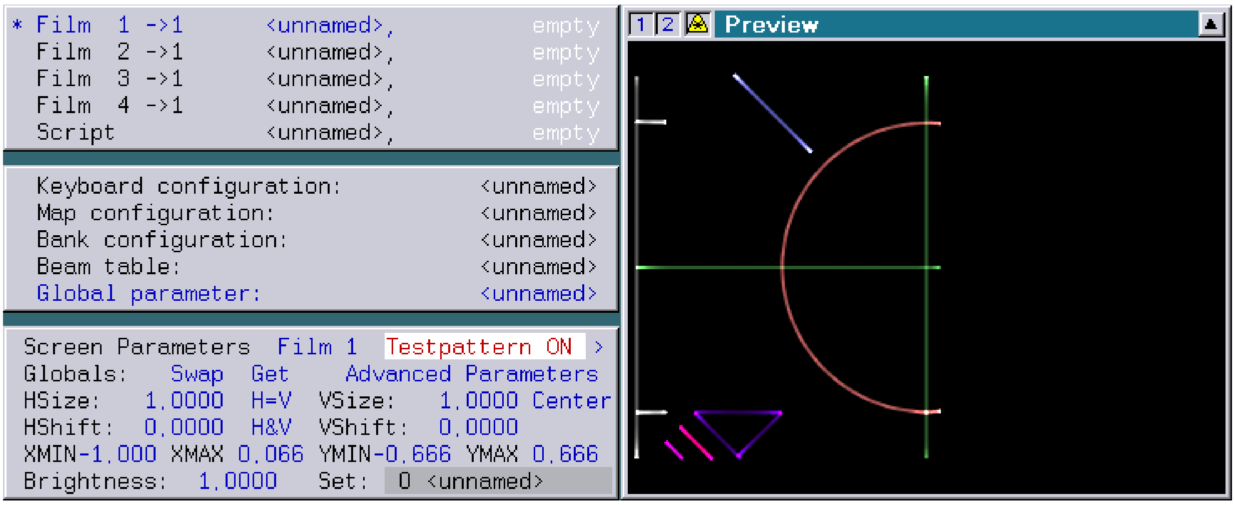


Example: Two Channel Projection with Soft Edge

Step 1

Define the desired Testpattern aspect ratio by adjusting the XY min / max values in the "Screen Parameters".

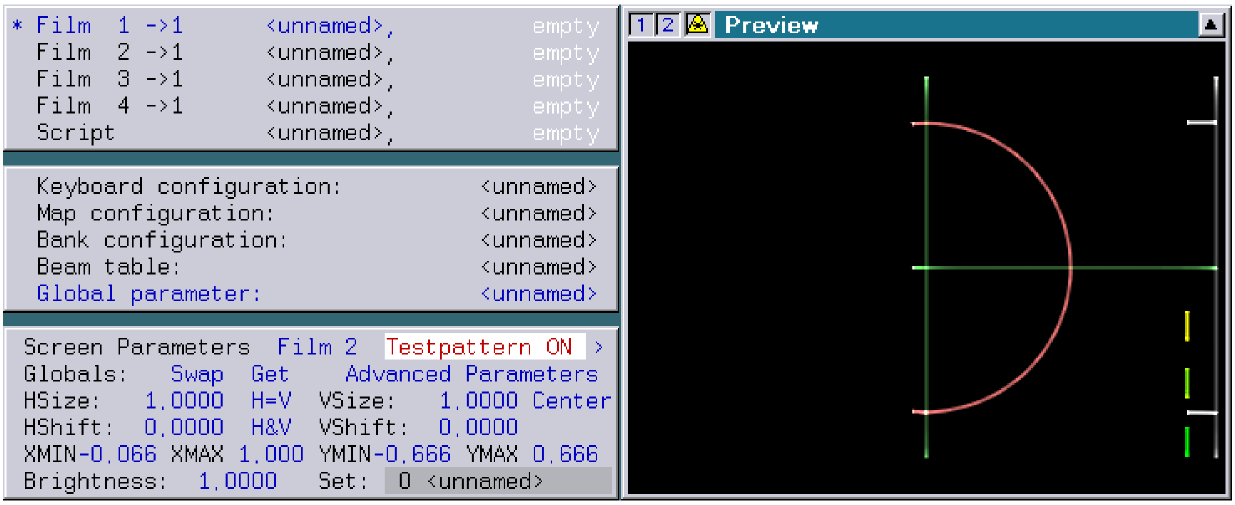
Example for a 3:2 projection and two projectors split:



```
* Film  1 ->1       <unnamed>,          empty
  Film  2 ->1       <unnamed>,          empty
  Film  3 ->1       <unnamed>,          empty
  Film  4 ->1       <unnamed>,          empty
  Script            <unnamed>,          empty

  Keyboard configuration:            <unnamed>
  Map configuration:                 <unnamed>
  Bank configuration:                <unnamed>
  Beam table:                        <unnamed>
  Global parameter:                  <unnamed>

 Screen Parameters  Film 1  Testpattern ON  >
 Globals:   Swap  Get    Advanced Parameters
 HSize:   1,0000  H=V  VSize:   1,0000 Center
 HShift:  0,0000  H&V  VShift:  0,0000
 XMIN-1,000 XMAX 0,066 YMIN-0,666 YMAX 0,666
 Brightness:  1,0000   Set:  0 <unnamed>
```

Left projector: Xmin -1.000, Xmax 0.066, Ymin -0.666, Ymax 0.666

Right projector: Xmin -0.066, Xmax 1.000, Ymin -0.666, Ymax 0.666



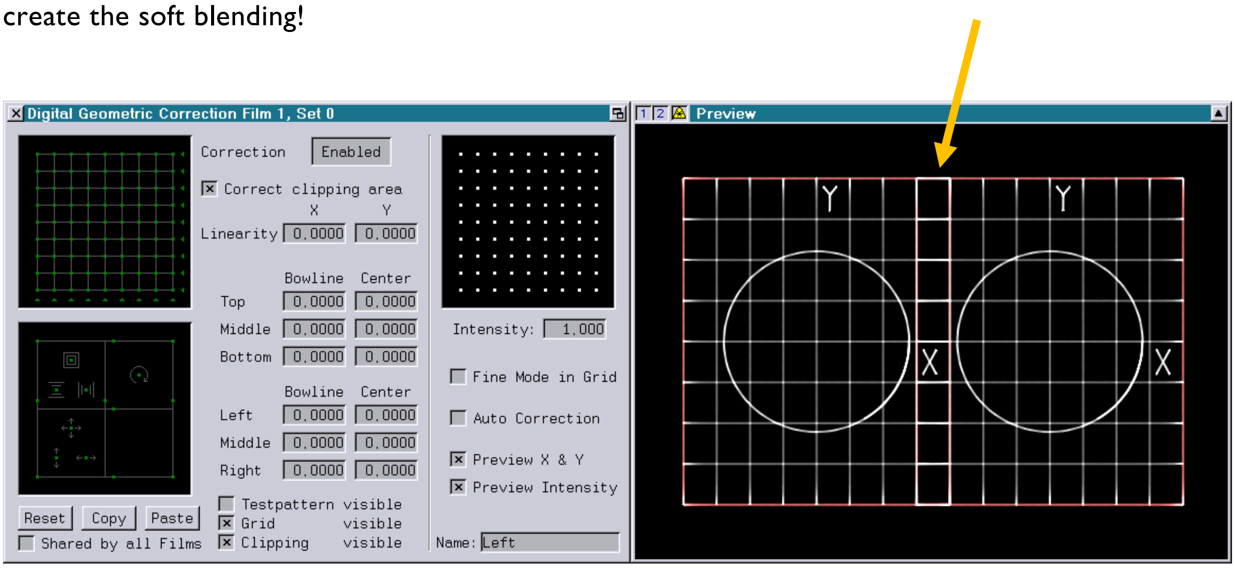
```
* Film  1 ->1       <unnamed>,          empty
  Film  2 ->1       <unnamed>,          empty
  Film  3 ->1       <unnamed>,          empty
  Film  4 ->1       <unnamed>,          empty
  Script            <unnamed>,          empty

  Keyboard configuration:            <unnamed>
  Map configuration:                 <unnamed>
  Bank configuration:                <unnamed>
  Beam table:                        <unnamed>
  Global parameter:                  <unnamed>

 Screen Parameters  Film 2  Testpattern ON  >
 Globals:   Swap  Get    Advanced Parameters
 HSize:   1,0000  H=V  VSize:   1,0000 Center
 HShift:  0,0000  H&V  VShift:  0,0000
 XMIN-0,066 XMAX 1,000 YMIN-0,666 YMAX 0,666
 Brightness:  1,0000   Set:  0 <unnamed>
```

Step 2

Open the "Digital Geometric Correction" and activate the check box "Correct clipping area".

Use the "Grid" pattern to scale and move the Testpattern to the desired size and position. Make sure to have a perfect circle in the end.

Do the same with the second projector. Note that at least one row of squares must overlap to create the soft blending!



Digital Geometric Correction Film 1, Set 0

Correction Enabled

☒ Correct clipping area

	X	Y
Linearity	0.0000	0.0000
	Bowline	Center
Top	0.0000	0.0000
Middle	0.0000	0.0000
Bottom	0.0000	0.0000
	Bowline	Center
Left	0.0000	0.0000
Middle	0.0000	0.0000
Right	0.0000	0.0000

Intensity: 1.000

□ Fine Mode in Grid

□ Auto Correction

☒ Preview X & Y

☒ Preview Intensity

□ Testpattern visible

☒ Grid visible

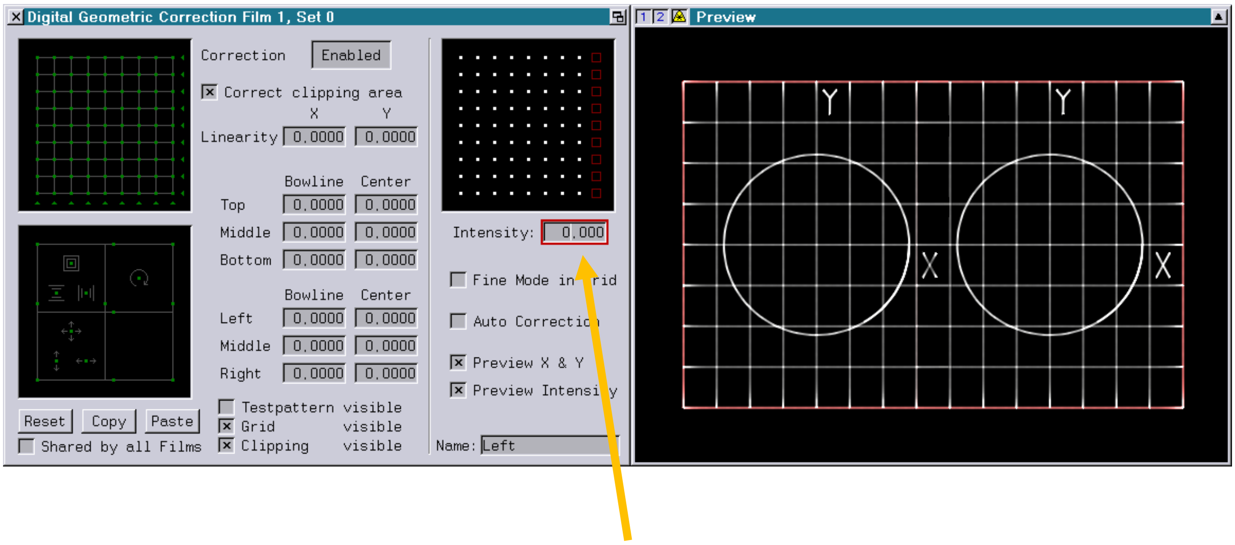
☒ Clipping visible

□ Shared by all Films

Name: Left

Step 3

Adjust the brightness of the overlapping areas to "0" using the intensity grid (<shift>+drag selects multiple points):



Digital Geometric Correction Film 1, Set 0

Correction Enabled

☒ Correct clipping area

	X	Y
Linearity	0.0000	0.0000
	Bowline	Center
Top	0.0000	0.0000
Middle	0.0000	0.0000
Bottom	0.0000	0.0000
	Bowline	Center
Left	0.0000	0.0000
Middle	0.0000	0.0000
Right	0.0000	0.0000

Intensity: 0.000

□ Fine Mode in Grid

□ Auto Correction

☒ Preview X & Y

☒ Preview Intensity

□ Testpattern visible

☒ Grid visible

☒ Clipping visible

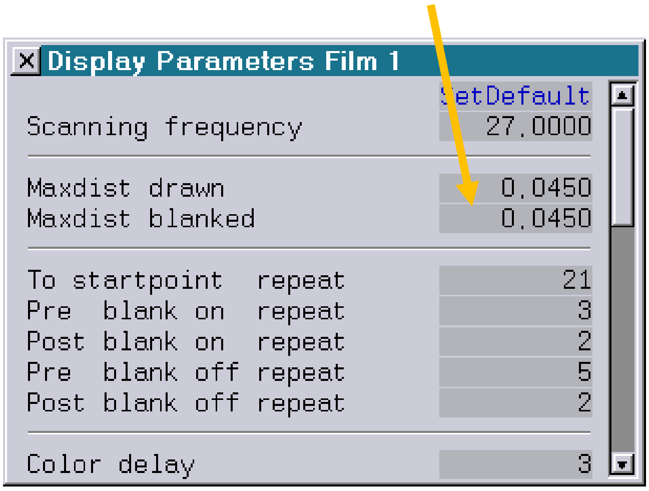
□ Shared by all Films

Name: Left

Step 4

Adjust the interpolation values in the "Display Parameters" proportional to the amount of split zones. In this example divide the default value by 2:

Maxdist drawn:	0.09 / 2 = 0.045
Maxdist blanked:	0.09 / 2 = 0.045



Display Parameters Film 1

SetDefault

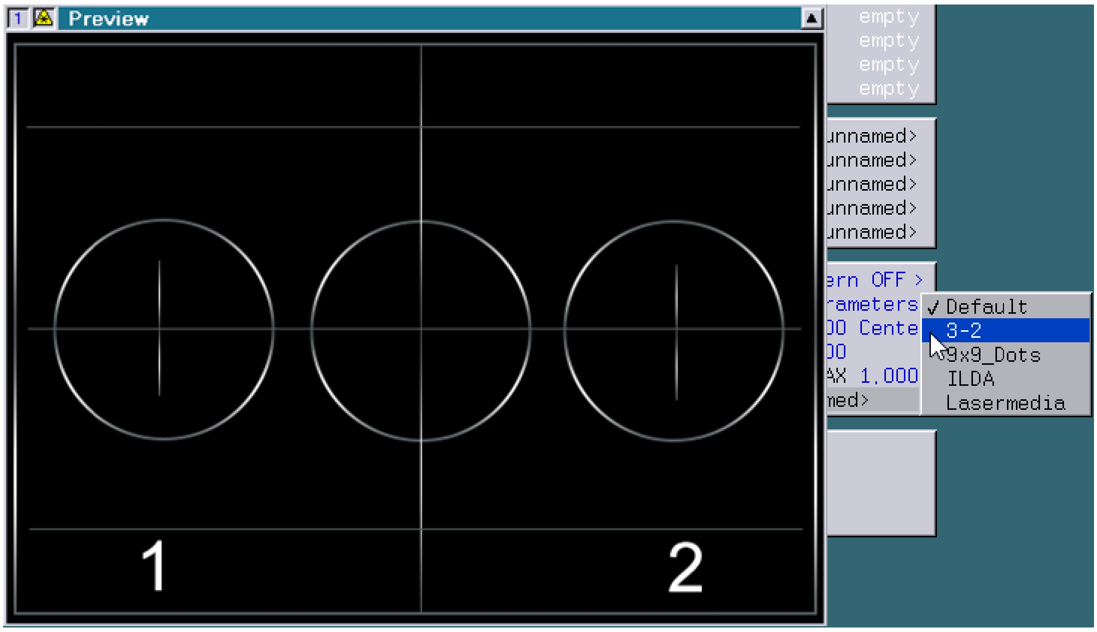
Scanning frequency	27.0000
Maxdist drawn	0.0450
Maxdist blanked	0.0450
To startpoint repeat	21
Pre blank on repeat	3
Post blank on repeat	2
Pre blank off repeat	5
Post blank off repeat	2
Color delay	3

Annotation:

Create your own Testpattern that fits into the ratio and maps the split zones. Adjusting the projectors will be easier then!

Save the new Testpattern inside "C:/System/Testpattern.CATL" e.g. as"3-2" and restart the Lasergraph DSP.

After the restart you can directly use the picture from the Testpattern selection menu:



Net- vs. NetScript-Timecode

In a setup with several Lasergraph DSPs the following question arises:
How shall the Lasergraph DSPs be synchronized to each other?

There are two different solutions:

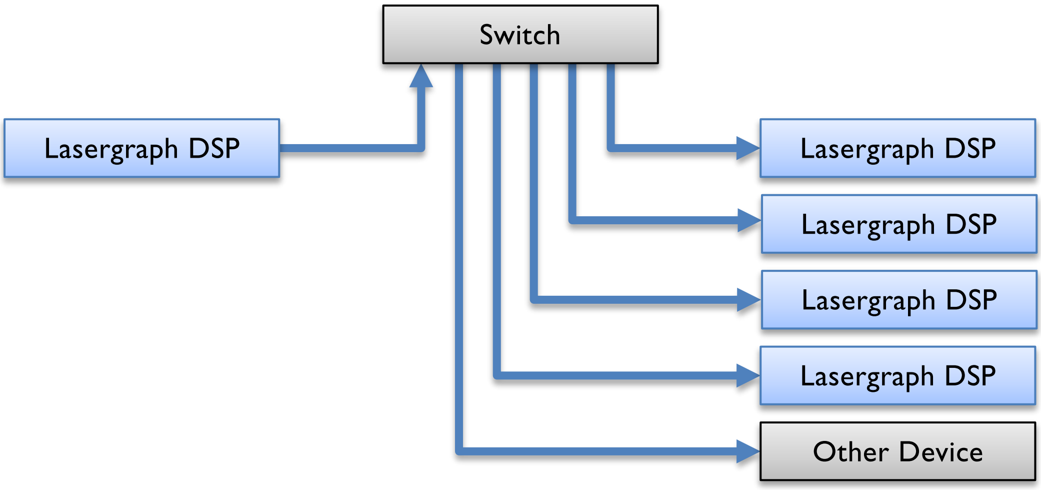
1) One Lasergraph DSP (master) writes its script timecode as Net-Timecode into the network. To enable this, it's Timescript has to begin with the command "SetTimeMode" in which "Write" is enabled.
   All other Lasergraph DSPs have to listen to the external timecode source "Net".

2) The other Lasergraph DSPs listen directly to the script timecode of the master.
   To do this the master has to be entered as server in the I/O-Setup of the other Lasergraph DSPs and the other Lasergraph DSPs have to listen to the external timecode source "NetScript".

To decide which way is the best for a specific project, it is necessary to understand how the transmission of the different timecodes work.

Net-Timecode

Net-Timecode is sent using UDP broadcasts.
Therefore all devices within the network will receive the Net-Timecode:



Advantages:

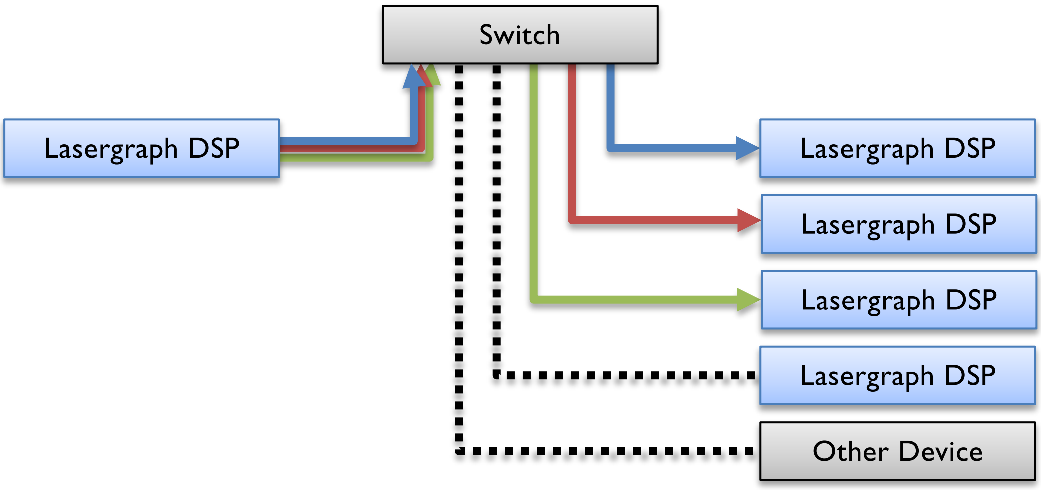
- ✓ No configuration required
- ✓ Net-Timecode can be displayed on a PC or Mac with the software "LGTimecode"

Disadvantages:

- Only one device within the network may send Net-Timecode
- All devices receive the Net-Timecode, regardless of whether they need it or not

NetScript-Timecode

NetScript-Timecode is only sent to registered devices using individual TCP connections:



This mechanism is also used for the transmission of other timecode sources like "Media", "Smpte" or "Midi".

Advantages:

- ✓ More than one timecode can be sent independently within the same network
- ✓ Less overall network traffic

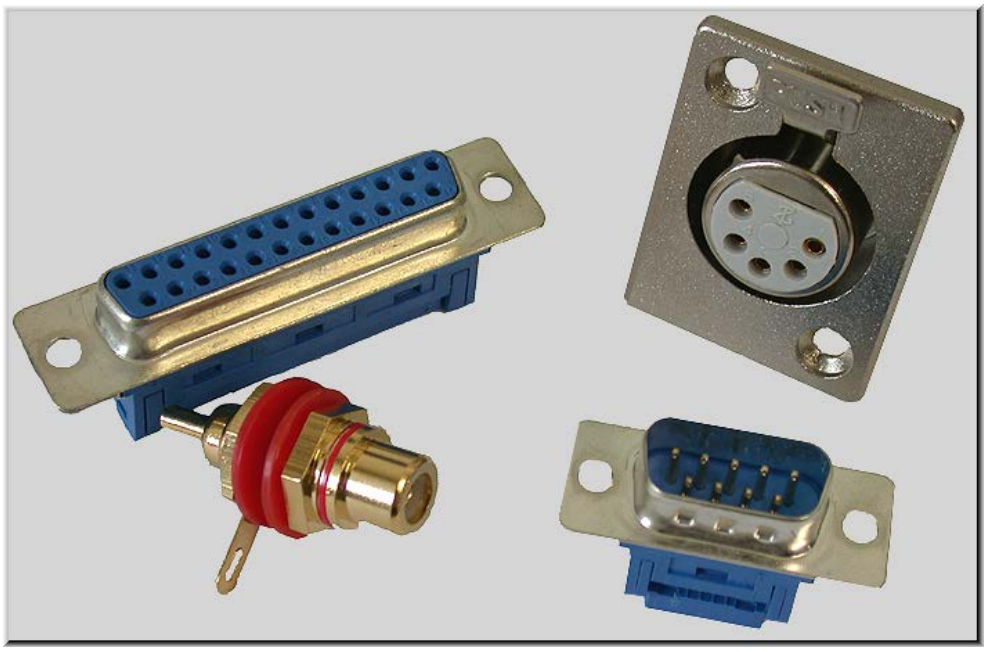
Disadvantages:

- The master has to be entered in the I/O-Setup of the other Lasergraph DSPs
- Higher network traffic for the master

Pinouts

Technical Data

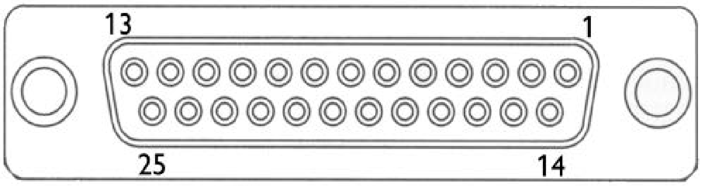
Pinouts



ILDA Out (ILDA Standard)	Connection of a laser projector according to ILDA standard
Scanner RGB (LaserAnimation Standard)	To connect a X,Y-scanner unit
Color 6-line (LaserAnimation Standard)	To connect a color control unit (6-line support)
RS232	Connection for media controller…
RS485/Fader	For connection of a faderboard unit
Ethernet	Connection with PC or laptop via Ethernet network
DMX In	For connection of a DMX lighting console
DMX Out	To connect fog machines, dimmers or other effect devices
MIDI In	Connection of a MIDI Keyboard
MIDI Out	To connect e.g. a MIDI peripheral card
SMPTE In/Out	To connect e.g. a audio player (ADAT)
VITC	To connect e.g. a video player
Sync	Connection of an external sync signal

ILDA Out Connector

25pin D-Sub connector female

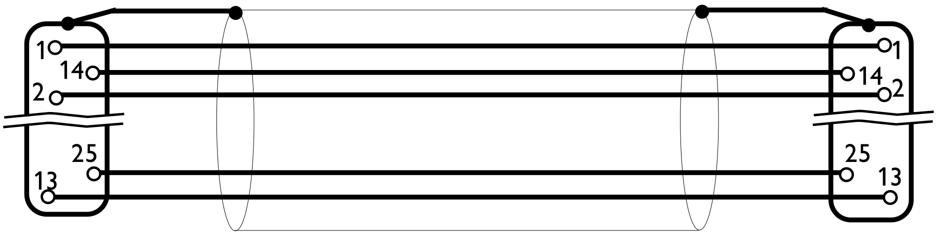


PIN No.	Signal	Level	Meaning
1	X+	± 10V measured against X-	Beam position: +10V : right 0V : center - 10V : left
2	Y+	± 10V measured against Y-	Beam position: +10V : top 0V : center - 10V : bottom
3	Intensity+	0V, 5V against Intensity-	0V: blanked 5V: beam on
4	Interlock A	Interlock loop	Connected with Interlock B
5	Red+ (Color 1)	0V ... +5V against Red-	0V : 0% red 5V : 100% red
6	Green+ (Color 2)	0V ... +5V against Green-	0V : 0% green 5V : 100% green
7	Blue+ (Color 3)	0V ... +5V against Blue-	0V : 0% blue 5V : 100% blue
8	Indigo+ (Color 5)	0V ... +5V against Indigo-	0V : 0% indigo 5V : 100% indigo
9	Yellow+ (Color 6)	0V ... +5V against Yellow-	0V : 0% yellow 5V : 100% yellow
10	Cyan+ (Color 4)	0V...+5V against Cyan-	0V : 0% cyan 5V : 100% cyan
11	3D L/R+	0V, 5V against 3D L/R-	3D left/right: 0V: left picture 5V: right picture
12	reserved	Return lead for the projector	
13	Shutter	0V to + 5V	0V: Shutter closed 5V: Shutter open

14	X-	Return lead for X+	
15	Y-	Return lead for Y+	
16	Intensity-	Return lead for Intensity+	
17	Interlock B	Interlock loop	Connected with Interlock A
18	Red-	Return lead for Red+	
19	Green-	Return lead for Green+	
20	Blue-	Return lead for Blue+	
21	Indigo-	Return lead for Indigo+	
22	Yellow-	Return lead for Yellow+	
23	Cyan-	Return lead for Cyan+	
24	3D L/R-	Return lead for 3D L/R+	
25	GND		

GND: Ground

Recommended cable (maximum length: 50m)



25pin D-Sub male

shielded cable

25pin D-Sub female

Attention!

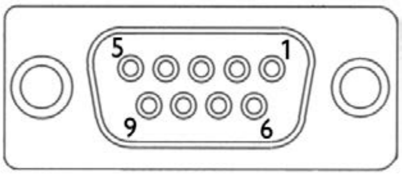
On older versions of the Lasergraph DSP *compact Mark 2* both pins of a differential output are each driven active at the ILDA output.
If a projector with a non-differential input will be connected to those units it is forbidden to connect the negative outputs together as well as to put them to GND!

Instead of that the non-differential outputs (D-Sub 9pin) should be used in this case.

Scanner RGB

(LaserAnimation Standard)

9pin D-Sub female

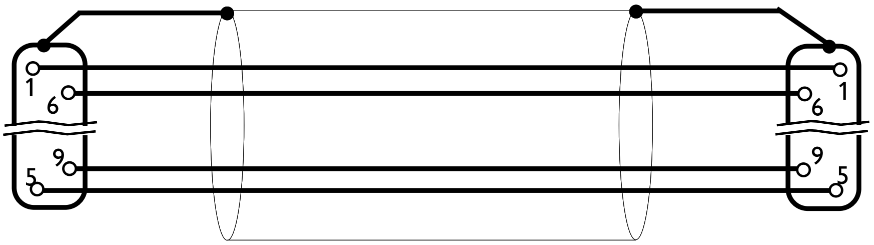


Note:
Newer versions of Lasergraph DSP *compact Mark 2* units are not equipped with this connector!

PIN No.	PIN Assignment	Level
1	X	-10V ... +10V
2	Y	-10V ... +10V
3	Blank (beam on/off)	TTL (digital 0V or +5V) 0V beam off, +5V beam on
4	Blue (Color 3)	0V ... +5V
5	Green (Color 2)	0V ... +5V
6	Red (Color 1)	0V ... +5V
7	P0 (3D L/R)	TTL (0V: 3D left, +5V: 3D right)
8	P1 (3D on/off)	TTL (0V: 3D on, +5V: 3D off)
9	GND	0V

TTL: 0V or +5V
GND: Ground

Recommended cable (maximum length; 30m)

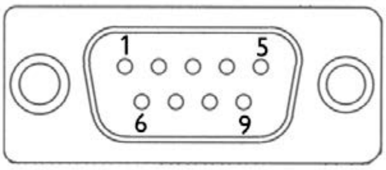


9pin D-Sub male shielded cable 9pin D-Sub female

Color 6-line

(LaserAnimation Standard)

9pin D-Sub male

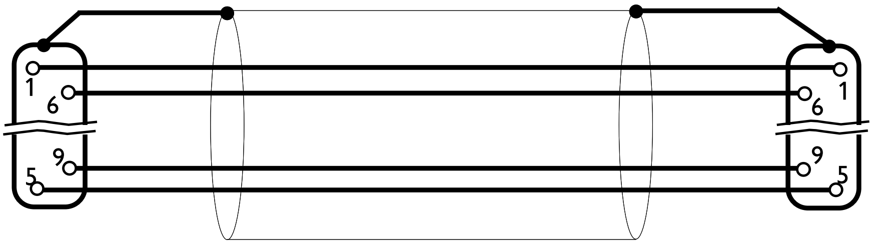


Note:
Newer versions of Lasergraph DSP *compact Mark 2* units are not equipped with this connector!

PIN No.	PIN Assignment	Level
1	Red (Color 1)	0V ... +5V
2	Blue (Color 3)	0V ... +5V
3	Indigo (Color 5)	0V ... +5V
4	Blank (beam on/off)	TTL (digital 0V or +5V) 0V beam off, +5V beam on
5	Logoshutter	0V ... +5V
6	Green (Color 2)	0V ... +5V
7	Cyan (Color 4)	0V ... +5V
8	Yellow (Color 6)	0V ... +5V
9	GND	0V

TTL: 0V or +5V
GND: Ground

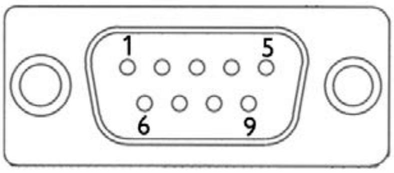
Recommended cable (maximum length: 30m)



9pin D-Sub male shielded cable 9pin D-Sub female

RS232

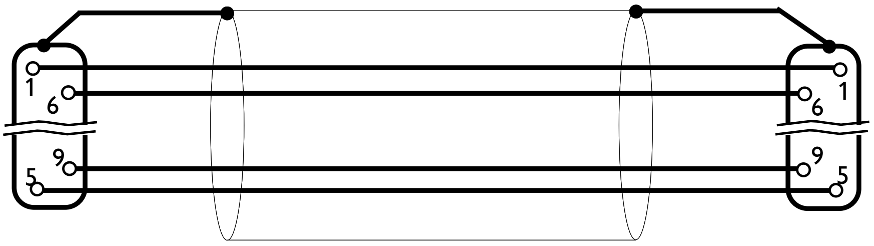
9pin D-Sub connector male



PIN No.	PIN Assignment	Meaning
1	CD*	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI*	Ring Indicator

*The signals CD and RI will be used by modems only.
GND: Ground

Recommended cable (maximum length: 15m*)

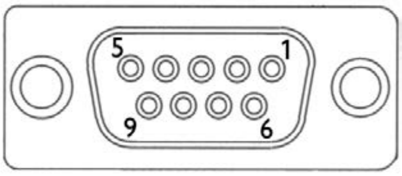


9pin D-Sub male shielded cable 9pin D-Sub female

(* By using low-capacitance cables, the maximum distance can be increased.)

RS485/Fader

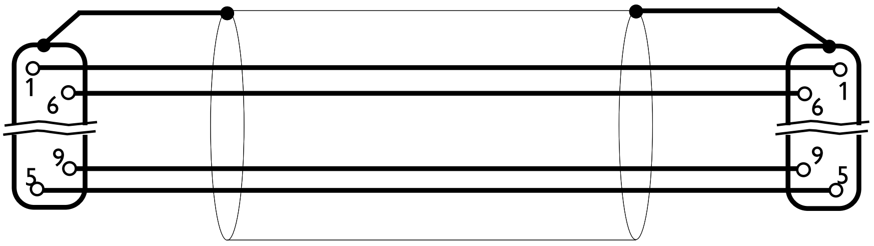
9pin D-Sub-connector female



PIN No.	PIN Assignment	Meaning
1	TXD+	Transmit Data not inverted
2	RXD+	Receive Data not inverted
3	TXD-	Transmit Data inverted
4	RXD-	Receive Data inverted
5	GND	Ground (shield)
6	+12V	Supply voltage
7	+12V	Supply voltage
8	GND	
9	GND	

GND: Ground

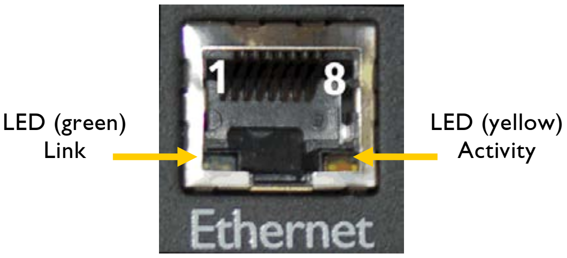
Recommended cable (maximum length: 30m)



9pin D-Sub male shielded cable 9pin D-Sub female

Ethernet

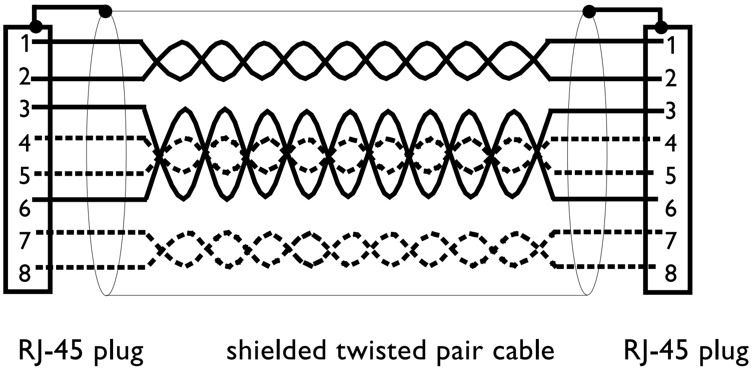
8pin RJ-45 Jack



For connection to a switch or to another Lasergraph DSP *Mark 2* CAT-5 Patch network cables with RJ-45 connectors should be used.

PIN No.	PIN Assignment
1	TXD+
2	TXD-
3	RXD+
4	not used
5	not used
6	RXD-
7	not used
8	not used

Recommended cable (minimum length: 2m; maximum length: 50m)

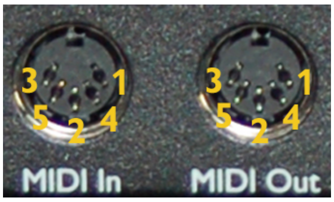


Note:

The pairs 4/5 and 7/8 are not used by the *compact Mark 2.* So it is not necessary to connect them.

MIDI In, MIDI Out

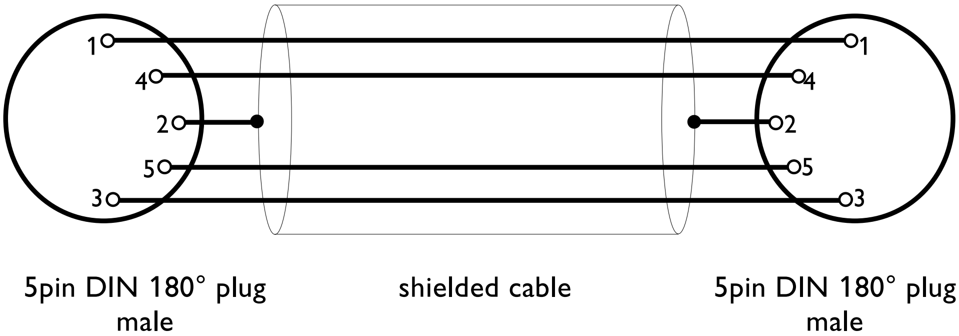
5pin DIN Receptacles



PIN No.	PIN Assignment MIDI In	PIN Assignment MIDI Out
1	+12V (250mA)	NC
2	GND	GND
3	+12V (250mA)	NC
4	Signal +	Signal +
5	Signal -	Signal -

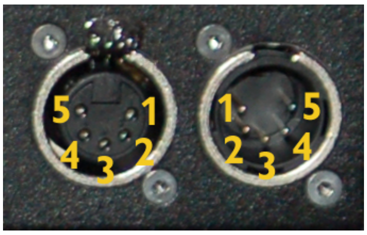
NC: Not Connected
GND: Ground

Recommended cable (maximum length 15m):



DMX In, DMX Out

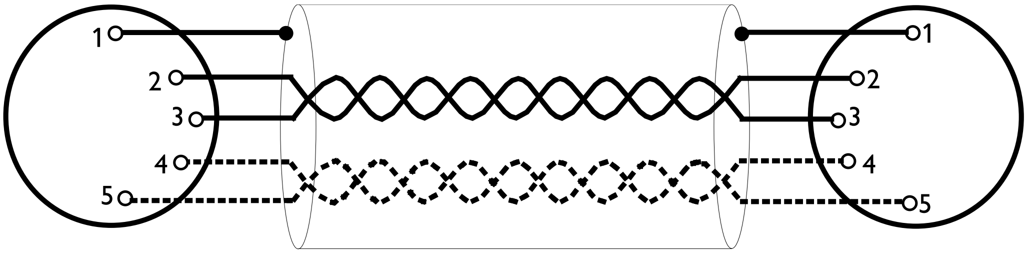
5pin XLR male/female



PIN No.	PIN Assignment
1	GND
2	Signal inverted
3	Signal not inverted
4	NC
5	NC

NC: Not Connected
GND: Ground

Recommended cable (maximum length: 500m)



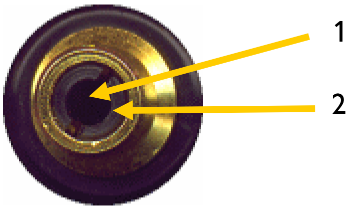
5pin XLR male shielded twisted pair cable 5pin XLR female

Notes:

- Connection of pin 4 and 5 is not necessary because the signals are not used.
- Some DMX systems use 3pin XLR-connectors. Please use the corresponding manual of the DMX system for pin assignment.

VITC, SMPTE In, SMPTE Out

RCA Jacks

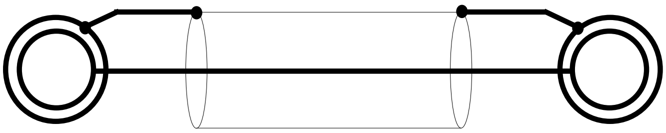


Note:
Newer versions of Lasergraph DSP *compact Mark 2* units are not equipped with a VITC input!

PIN No.	PIN Assignment
1	Signal
2	GND

GND: Ground

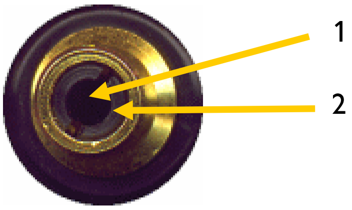
Recommended cable (maximum length 10m):



RCA plug shielded cable RCA plug

Sync

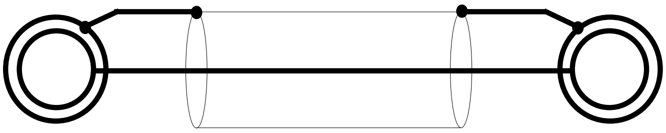
RCA Jack



PIN No.	PIN Assignment
1	Signal
2	GND

GND: Ground

Recommended cable (maximum length: 10m):



RCA plug shielded cable RCA plug

Notes:

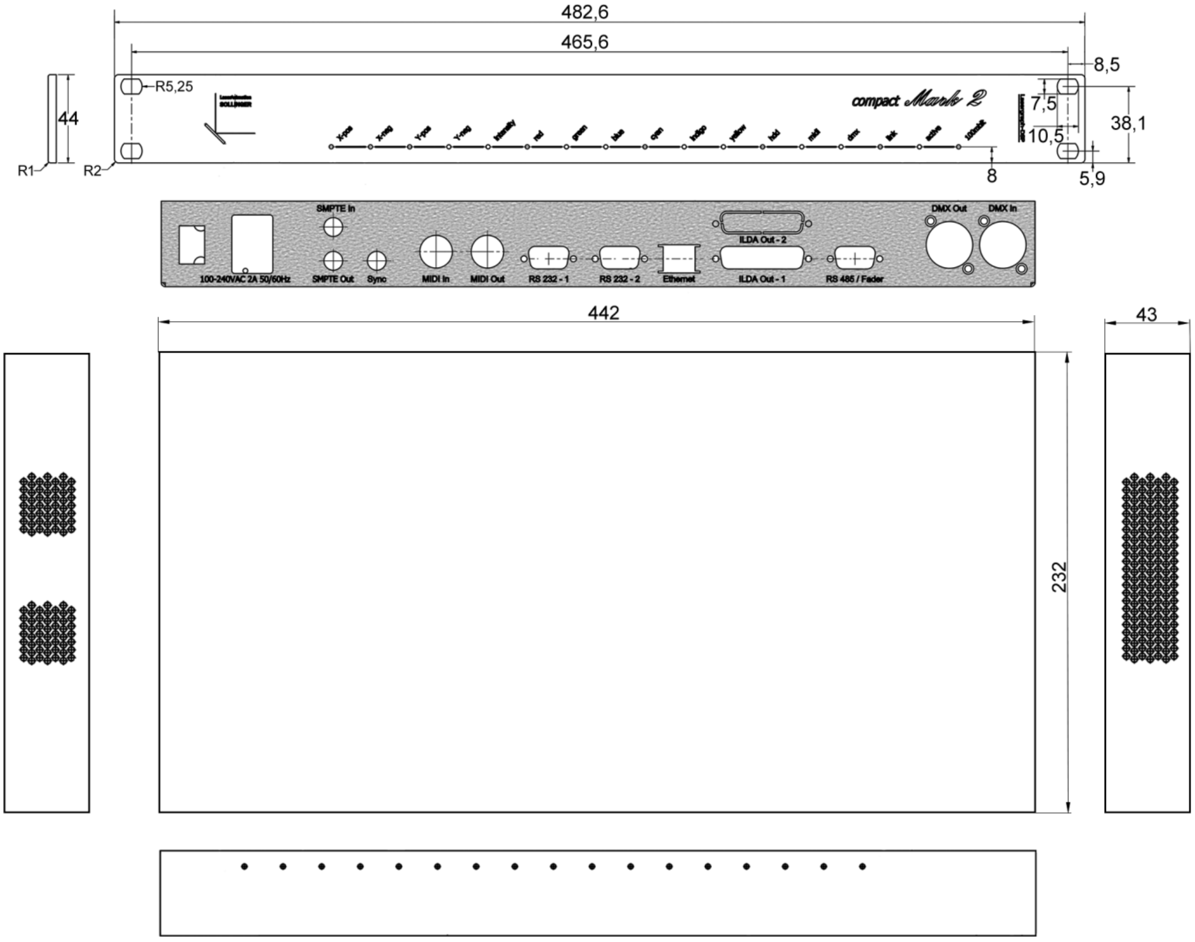
- Input signal in the range of 3.3V to 12V (typical: 5V)
- The following hysteresis is available:
  Rising edge > 1.9V
  Falling edge < 1.5V
- The "Sync" input is not available on older Lasergraph DSP *compact Mark 2* units

Technical Data

Electrical	
AC power input	Main voltages 100-240 VAC
Input frequency	50-60 Hz
Input power	75 W maximum, typical < 30 W

Connectors	
AC Mains Connector	3-pin IEC socket male
ILDA Out	D-sub 25 female for connection of one scanning unit
Ethernet	Standard RJ-45 jack
DMX In/Out	5-pin XLR male/female
SMPTE In/Out	RCA Jacks
MIDI In/Out	5-pin DIN Receptacles
RS485 Fader	9-pin D-sub female
2 x RS 232	9-pin D-sub male (standard RS232)
VITC	RCA Jack
Sync	RCA Jack

Physical	
Ambient temperature range	+5°C to +40°C
Degree of protection	IP 20
Weight	4 kg
Dimensions	19" rack mount, 1 unit height, depth 235 mm



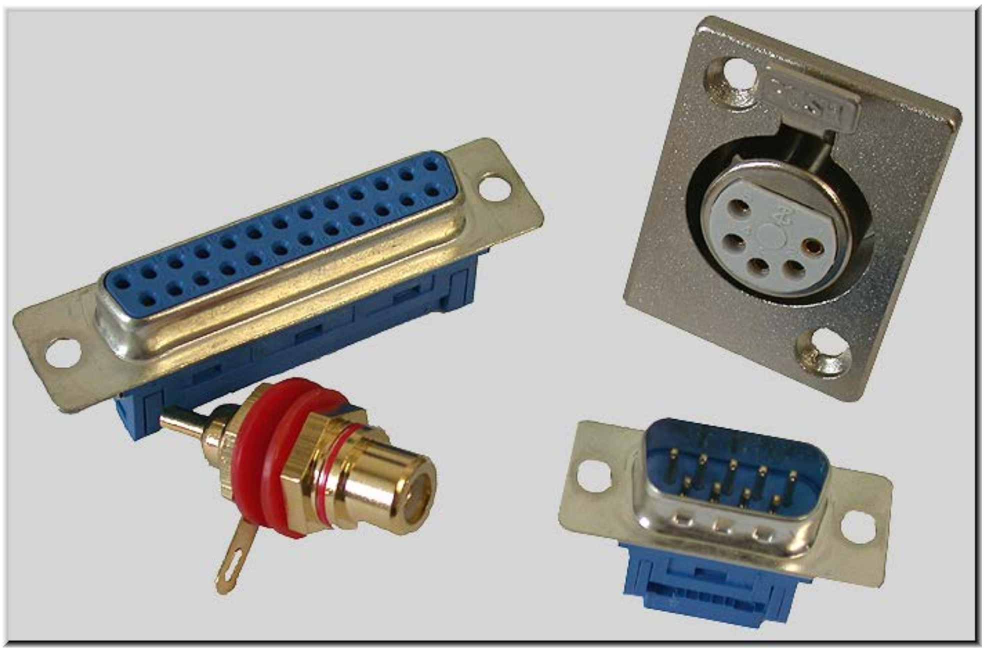
Operation Displays	
X-pos X-neg Y-pos Y-neg intensity red green blue cyan indigo yellow	Indicate the state of the laser output
hdd	Indicates storage access
midi	Indicates an active Midi input
dmx	Indicates an active DMX input
link active 100mbit	Indicate the state of network connection

Technical specifications are subject to change without prior notice.

Pinouts

Technical Data

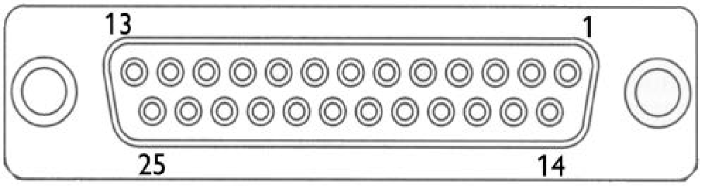
Pinouts



ILDA Out (ILDA Standard)	Connection of a laser projector according to ILDA standard
RS232	Connection for media controller…
RS485/Fader	For connection of a faderboard unit
Ethernet	Connection with PC or laptop via Ethernet network
DMX In	For connection of a DMX lighting console
DMX Out	To connect fog machines, dimmers or other effect devices
MIDI In	Connection of a MIDI Keyboard
MIDI Out	To connect e.g. a MIDI peripheral card
SMPTE In/Out	To connect e.g. a audio player (ADAT)
VITC	To connect e.g. a video player
Sync	Connection of an external sync signal

ILDA Out Connector

25pin D-Sub connector female

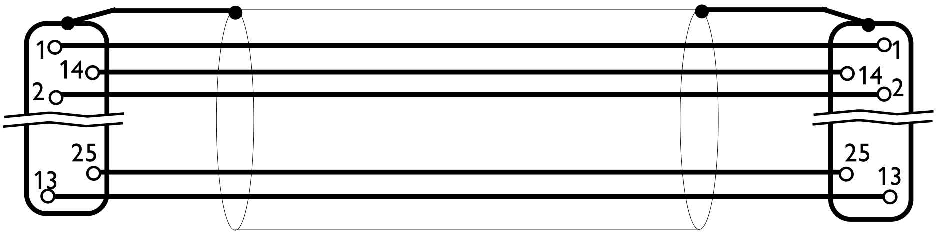


PIN No.	Signal	Level	Meaning
1	X+	± 10V measured against X-	Beam position: +10V : right 0V : center - 10V : left
2	Y+	± 10V measured against Y-	Beam position: +10V : top 0V : center - 10V : bottom
3	Intensity+	0V, 5V against Intensity-	0V: blanked 5V: beam on
4	Interlock A	Interlock loop	Connected with Interlock B
5	Red+ (Color 1)	0V ... +5V against Red-	0V : 0% red 5V : 100% red
6	Green+ (Color 2)	0V ... +5V against Green-	0V : 0% green 5V : 100% green
7	Blue+ (Color 3)	0V ... +5V against Blue-	0V : 0% blue 5V : 100% blue
8	Indigo+ (Color 5)	0V ... +5V against Indigo-	0V : 0% indigo 5V : 100% indigo
9	Yellow+ (Color 6)	0V ... +5V against Yellow-	0V : 0% yellow 5V : 100% yellow
10	Cyan+ (Color 4)	0V...+5V against Cyan-	0V : 0% cyan 5V : 100% cyan
11	3D L/R+	0V, 5V against 3D L/R-	3D left/right: 0V: left picture 5V: right picture
12	reserved	Return lead for the projector	
13	Shutter	0V to + 5V	0V: Shutter closed 5V: Shutter open

14	X-	Return lead for X+	
15	Y-	Return lead for Y+	
16	Intensity-	Return lead for Intensity+	
17	Interlock B	Interlock loop	Connected with Interlock A
18	Red-	Return lead for Red+	
19	Green-	Return lead for Green+	
20	Blue-	Return lead for Blue+	
21	Indigo-	Return lead for Indigo+	
22	Yellow-	Return lead for Yellow+	
23	Cyan-	Return lead for Cyan+	
24	3D L/R-	Return lead for 3D L/R+	
25	GND		

GND: Ground

Recommended cable (maximum length: 50m)



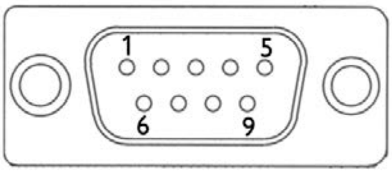
25pin D-Sub male

shielded cable

25pin D-Sub
female

RS232

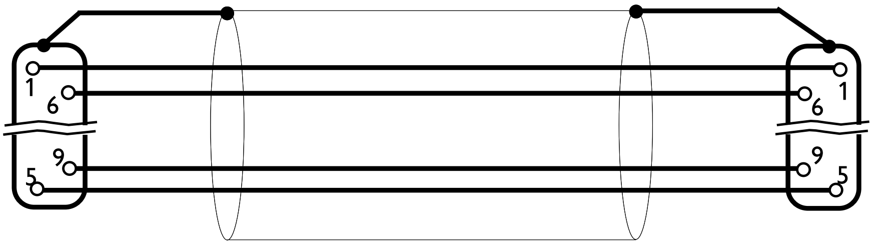
9pin D-Sub connector male



PIN No.	PIN Assignment	Meaning
1	CD*	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI*	Ring Indicator

*The signals CD and RI will be used by modems only.
GND: Ground

Recommended cable (maximum length: 15m*)

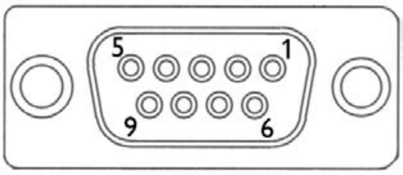


9pin D-Sub male shielded cable 9pin D-Sub female

(* By using low-capacitance cables, the maximum distance can be increased.)

RS485/Fader

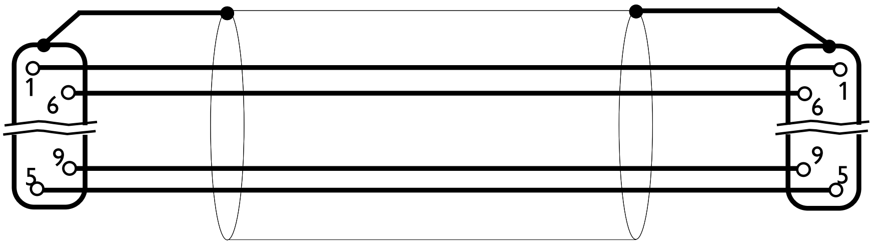
9pin D-Sub-connector female



PIN No.	PIN Assignment	Meaning
1	TXD+	Transmit Data not inverted
2	RXD+	Receive Data not inverted
3	TXD-	Transmit Data inverted
4	RXD-	Receive Data inverted
5	GND	Ground (shield)
6	+12V	Supply voltage
7	+12V	Supply voltage
8	GND	
9	GND	

GND: Ground

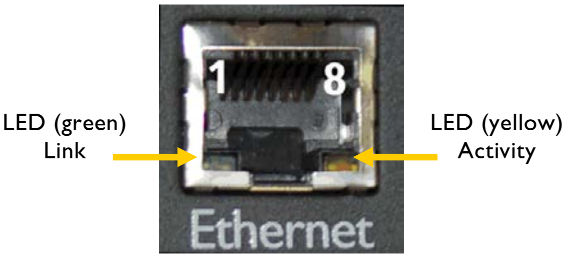
Recommended cable (maximum length: 30m)



9pin D-Sub male shielded cable 9pin D-Sub female

Ethernet

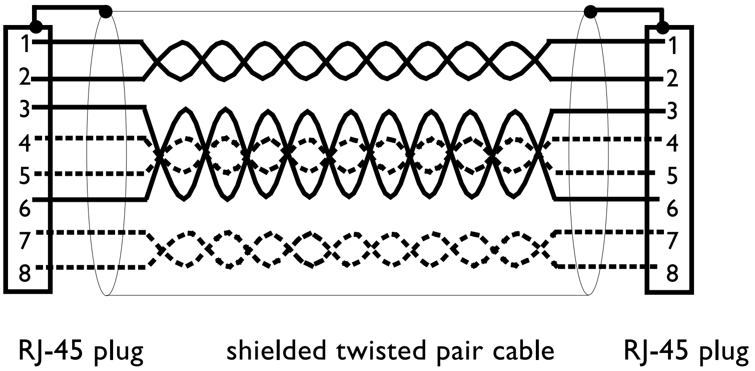
8pin RJ-45 Jack



For connection to a switch or to another Lasergraph DSP *Mark 2* CAT-5 Patch network cables with RJ-45 connectors should be used.

PIN No.	PIN Assignment
1	TXD+
2	TXD-
3	RXD+
4	not used
5	not used
6	RXD-
7	not used
8	not used

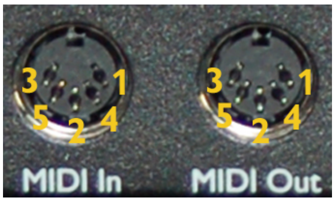
Recommended cable (minimum length: 2m; maximum length: 50m)



Note:
The pairs 4/5 and 7/8 are not used by the *compact dual Mark 2.* So it is not necessary to connect them.

MIDI In, MIDI Out

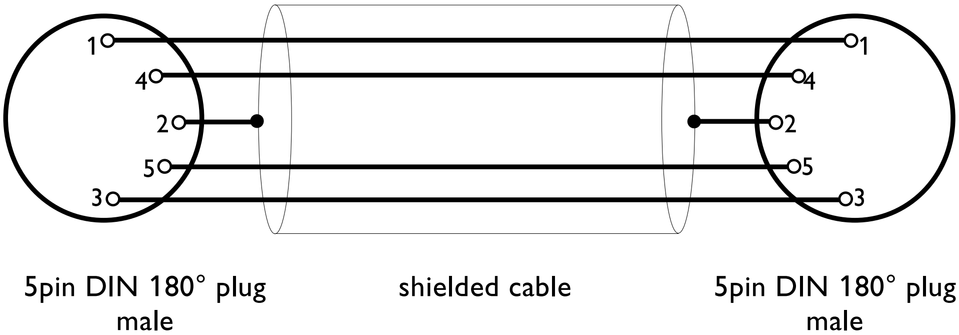
5pin DIN Receptacles



PIN No.	PIN Assignment MIDI In	PIN Assignment MIDI Out
1	+12V (250mA)	NC
2	GND	GND
3	+12V (250mA)	NC
4	Signal +	Signal +
5	Signal -	Signal -

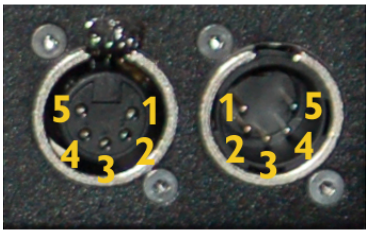
NC: Not Connected
GND: Ground

Recommended cable (maximum length 15m):



DMX In, DMX Out

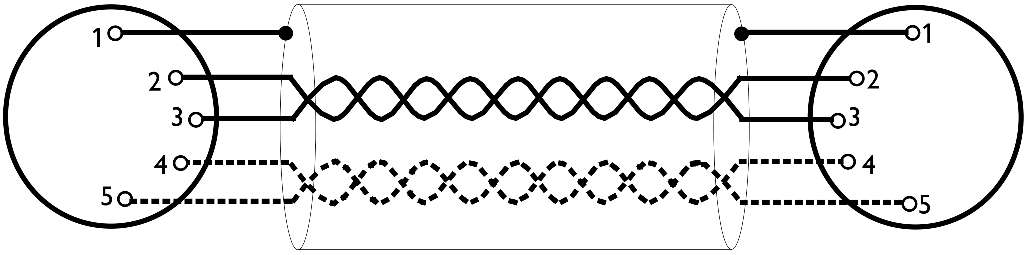
5pin XLR male/female



PIN No.	PIN Assignment
1	GND
2	Signal inverted
3	Signal not inverted
4	NC
5	NC

NC: Not Connected
GND: Ground

Recommended cable (maximum length: 500m)



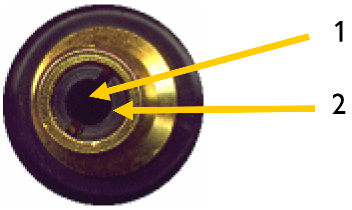
5pin XLR male shielded twisted pair cable 5pin XLR female

Notes:

- Connection of pin 4 and 5 is not necessary because the signals are not used.
- Some DMX systems use 3pin XLR-connectors. Please use the corresponding manual of the DMX system for pin assignment.

VITC, SMPTE In, SMPTE Out

RCA Jacks

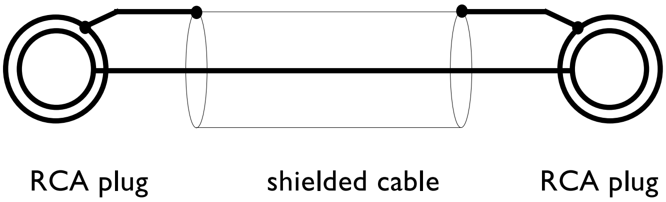


Note:
Newer versions of the Lasergraph DSP *compact dual Mark 2* are not equipped with a VITC input!

PIN No.	PIN Assignment
1	Signal
2	GND

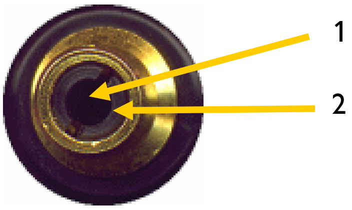
GND: Ground

Recommended cable (maximum length 10m):



Sync

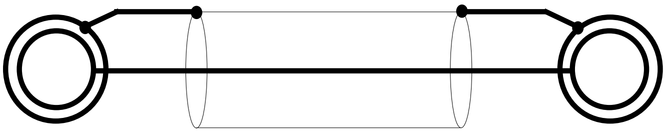
RCA Jack



PIN No.	PIN Assignment
1	Signal
2	GND

GND: Ground

Recommended cable (maximum length: 10m):



RCA plug shielded cable RCA plug

Notes:

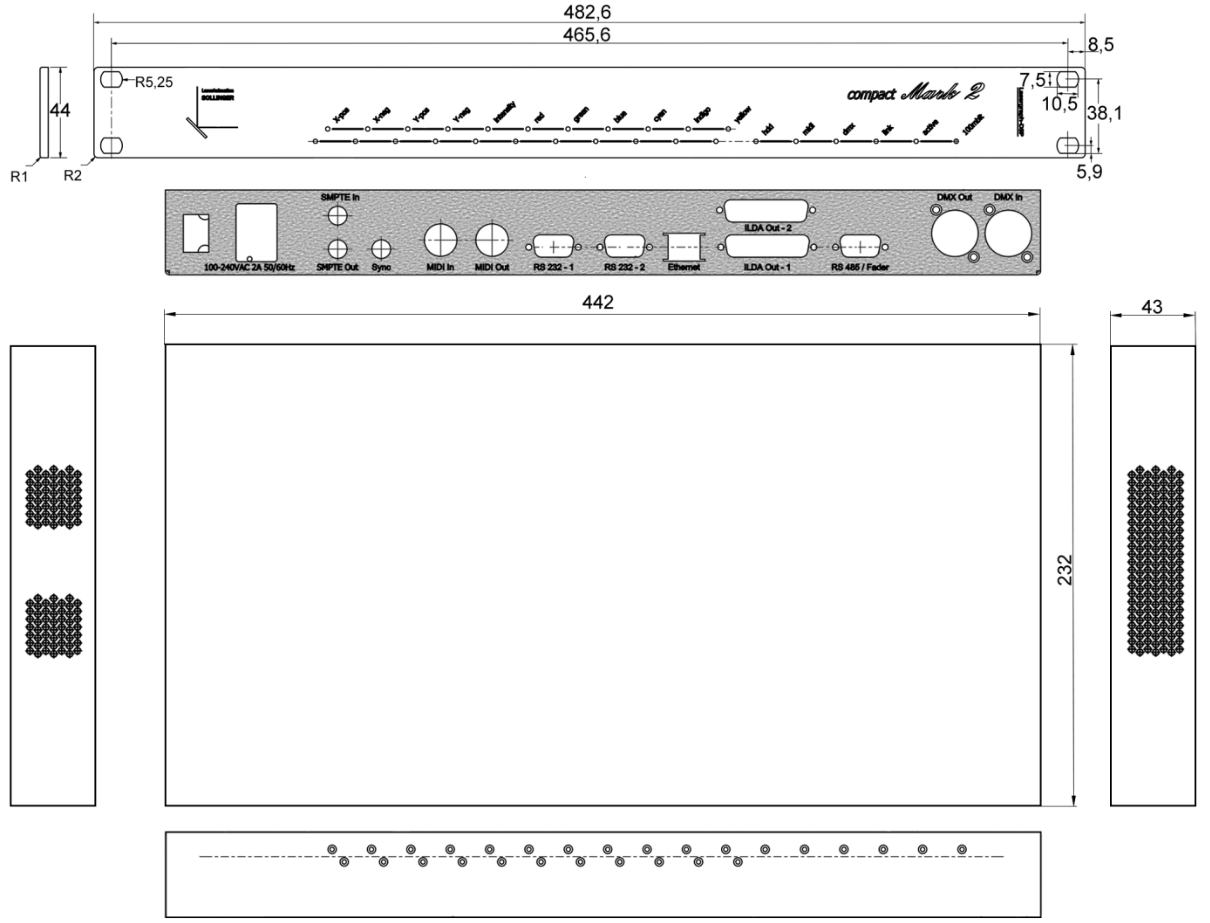
- Input signal in the range of 3.3V to 12V (typical: 5V)
- The following hysteresis is available:
  Rising edge > 1.9V
  Falling edge < 1.5V

Technical Data

Electrical	
AC power input	Main voltages 100-240 VAC
Input frequency	50-60 Hz
Input power	90 W maximum, typical < 35 W

Connectors	
AC Mains Connector	3-pin IEC socket male
2 x ILDA Out	D-sub 25 female for connection of one scanning unit
Ethernet	Standard RJ-45 jack
DMX In/Out	5-pin XLR male/female
SMPTE In/Out	RCA Jacks
MIDI In/Out	5-pin DIN Receptacles
RS485 Fader	9-pin D-sub female
2 x RS 232	9-pin D-sub male (standard RS232)
VITC	RCA Jack
Sync	RCA Jack

Physical	
Ambient temperature range	+5°C to +40°C
Degree of protection	IP 20
Weight	4.3 kg
Dimensions	19" rack mount, 1 unit height, depth 235 mm



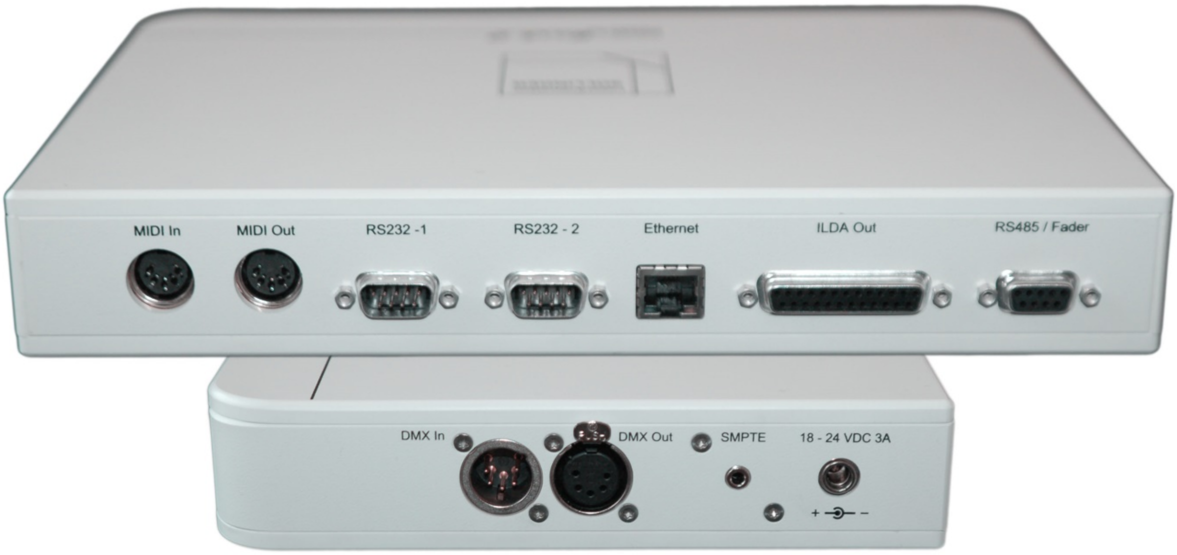
Operation Displays	
X-pos X-neg Y-pos Y-neg intensity red green blue cyan indigo yellow	Indicate the state of the laser output
hdd	Indicates storage access
midi	Indicates an active Midi input
dmx	Indicates an active DMX input
link active 100mbit	Indicate the state of network connection

Technical specifications are subject to change without prior notice.

Pinouts

Technical Data

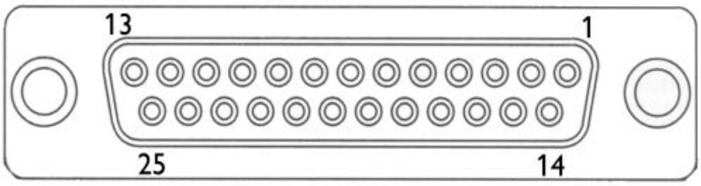
Pinouts



ILDA Out	Connection of a laser projector according to ILDA standard
RS232 (2 pcs.)	Connection for media controller…
RS485 / Fader	For connection of a faderboard unit
Ethernet	For connection with PC or Mac via Ethernet network
DMX In	To connect a DMX lighting console
DMX Out	To connect fog machines, dimmers or other effect devices
MIDI In	Connection of a MIDI Keyboard
MIDI Out	To connect e.g. a MIDI peripheral card
SMPTE (In/Out)	To connect e.g. an audio device (ADAT)
18-24VDC 3A	To connect the power adapter (included)

ILDA Out Connector

25pin D-Sub connector female

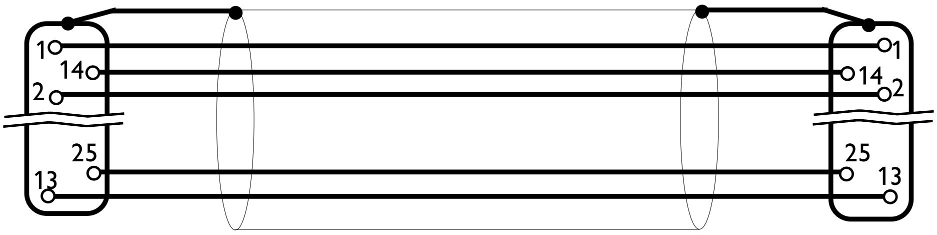


PIN No.	Signal	Level	Meaning
1	X+	± 10V measured against X-	Beam position: +10V : right 0V : center - 10V : left
2	Y+	± 10V measured against Y-	Beam position: +10V : top 0V : center - 10V : bottom
3	Intensity+	0V, 5V against Intensity-	0V: blanked 5V: beam on
4	Interlock A	Interlock loop	Connected with Interlock B
5	Red+ (Color 1)	0V ... +5V against Red-	0V : 0% red 5V : 100% red
6	Green+ (Color 2)	0V ... +5V against Green-	0V : 0% green 5V : 100% green
7	Blue+ (Color 3)	0V ... +5V against Blue-	0V : 0% blue 5V : 100% blue
8	Indigo+ (Color 5)	0V ... +5V against Indigo-	0V : 0% indigo 5V : 100% indigo
9	Yellow+ (Color 6)	0V ... +5V against Yellow-	0V : 0% yellow 5V : 100% yellow
10	Cyan+ (Color 4)	0V...+5V against Cyan-	0V : 0% cyan 5V : 100% cyan
11	3D L/R+	0V, 5V against 3D L/R-	3D left/right: 0V: left picture 5V: right picture
12	reserved	Return lead for the projector	
13	Shutter	0V to + 5V	0V: Shutter closed 5V: Shutter open

14	X-	Return lead for X+	
15	Y-	Return lead for Y+	
16	Intensity-	Return lead for Intensity+	
17	Interlock B	Interlock loop	Connected with Interlock A
18	Red-	Return lead for Red+	
19	Green-	Return lead for Green+	
20	Blue-	Return lead for Blue+	
21	Indigo-	Return lead for Indigo+	
22	Yellow-	Return lead for Yellow+	
23	Cyan-	Return lead for Cyan+	
24	3D L/R-	Return lead for 3D L/R+	
25	GND		

GND: Ground

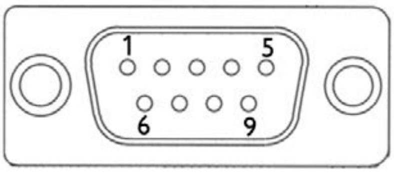
Recommended cable (maximum length: 50m)



25pin D-Sub male shielded cable 25pin D-Sub female

RS232

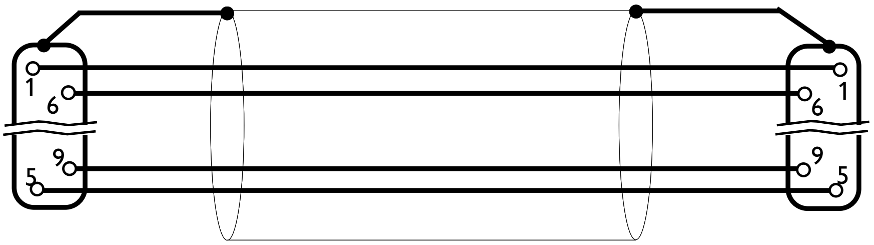
9pin D-Sub connector male



PIN No.	PIN Assignment	Meaning
1	CD*	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI*	Ring Indicator

*The signals CD and RI will be used by modems only.
GND: Ground

Recommended cable (maximum length: 15m*)

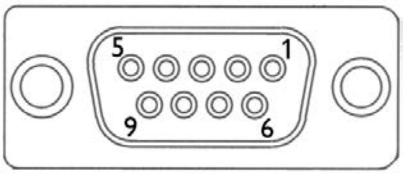


9pin D-Sub male shielded cable 9pin D-Sub female

(* By using low-capacitance cables, the maximum distance can be increased.)

RS485/Fader

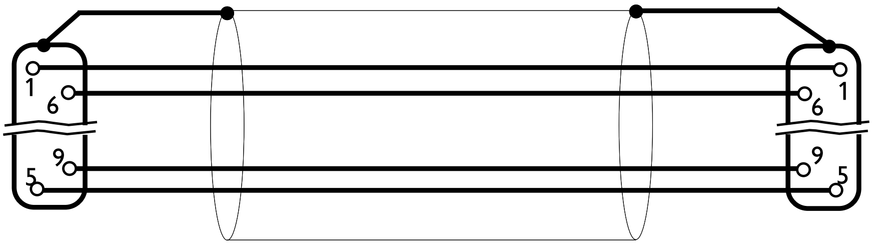
9pin D-Sub-connector female



PIN No.	PIN Assignment	Meaning
1	TXD+	Transmit Data not inverted
2	RXD+	Receive Data not inverted
3	TXD-	Transmit Data inverted
4	RXD-	Receive Data inverted
5	GND	Ground (shield)
6	+12V	Supply voltage
7	+12V	Supply voltage
8	GND	
9	GND	

GND: Ground

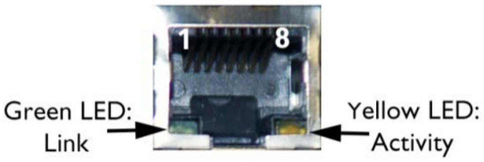
Recommended cable (maximum length: 30m)



9pin D-Sub male shielded cable 9pin D-Sub female

Ethernet

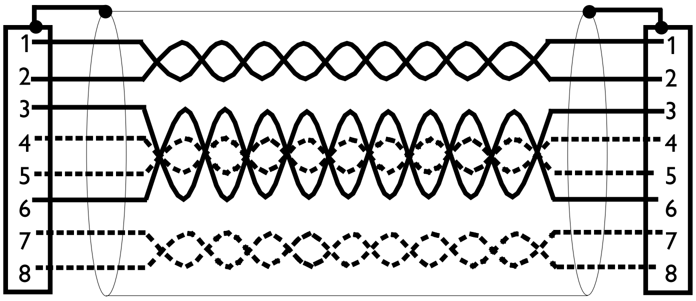
8pin RJ-45 Jack



For connection to a switch or to another Lasergraph *Mark 2*.
CAT-5 Patch network cables with RJ-45 connectors should be used.

PIN No.	PIN Assignment
1	TXD+
2	TXD-
3	RXD+
4	not used
5	not used
6	RXD-
7	not used
8	not used

Recommended cable (minimum length: 2m; maximum length: 50m)



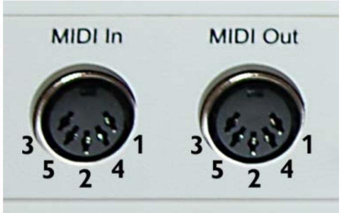
RJ-45 plug shielded twisted pair cable RJ-45 plug

Note:

The pairs 4/5 and 7/8 are not used by the *travel Mark 2***.**
It is not necessary to connect them.

MIDI In, MIDI Out

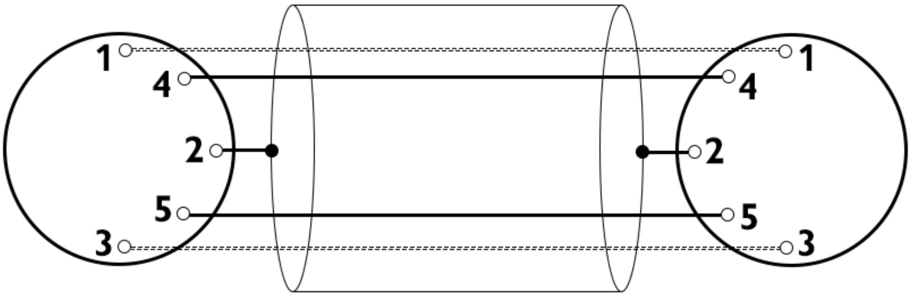
5pin DIN Receptacles



PIN No.	PIN Assignment MIDI In	PIN Assignment MIDI Out
1	DU	NC
2	GND	GND
3	DU	NC
4	Signal +	Signal +
5	Signal -	Signal -

DU: Don't Use
NC: Not Connected
GND: Ground

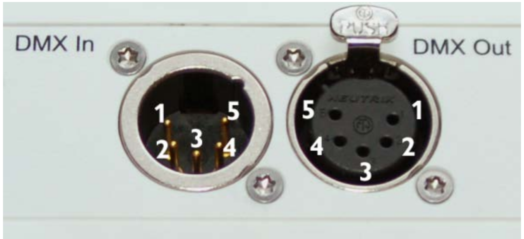
Recommended cable (maximum length 15m):



5pin DIN 180° plug male shielded cable 5pin DIN 180° plug male

DMX In, DMX Out

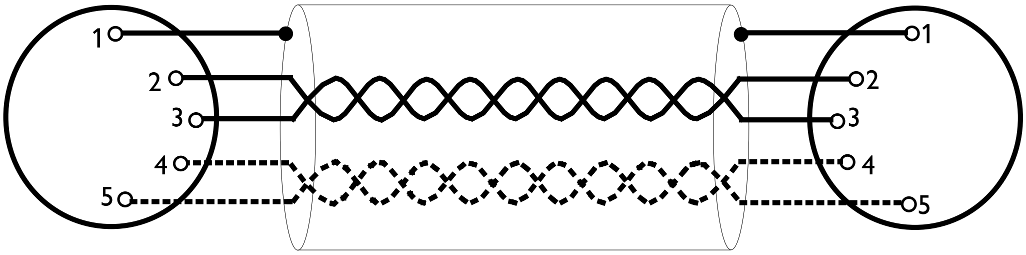
5pin XLR male/female



PIN No.	PIN Assignment
1	GND
2	Signal inverted
3	Signal not inverted
4	NC
5	NC

NC: Not Connected
GND: Ground

Recommended cable (maximum length: 500m)



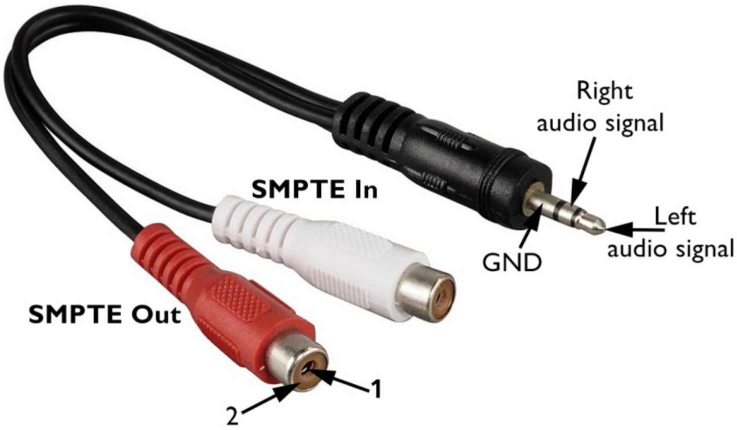
5pin XLR male shielded twisted pair cable 5pin XLR female

Notes:

- Connection of pin 4 and 5 is not necessary because the signals are not used.
- Some DMX systems use 3pin XLR-connectors. Please use the corresponding manual of the DMX system for pin assignment.

SMPTE In/Out

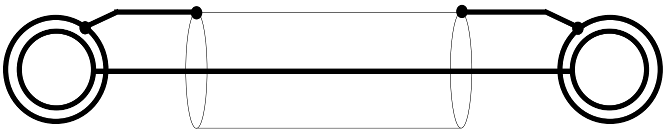
Audio jack 3,5mm to connect the supplied adapter: Jack to 2 x RCA/cinch e.g. as shown in the picture:



PIN No.	PIN Assignment
1	Signal
2	GND

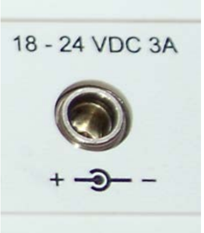
GND: Ground

Recommended cable (maximum length 10m):

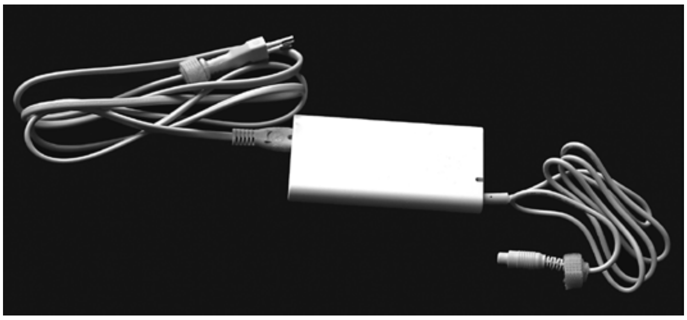


RCA plug shielded cable RCA plug

18-24VDC 3A



Connector for the power supply: Please connect only the included power adapter.



Technical Data

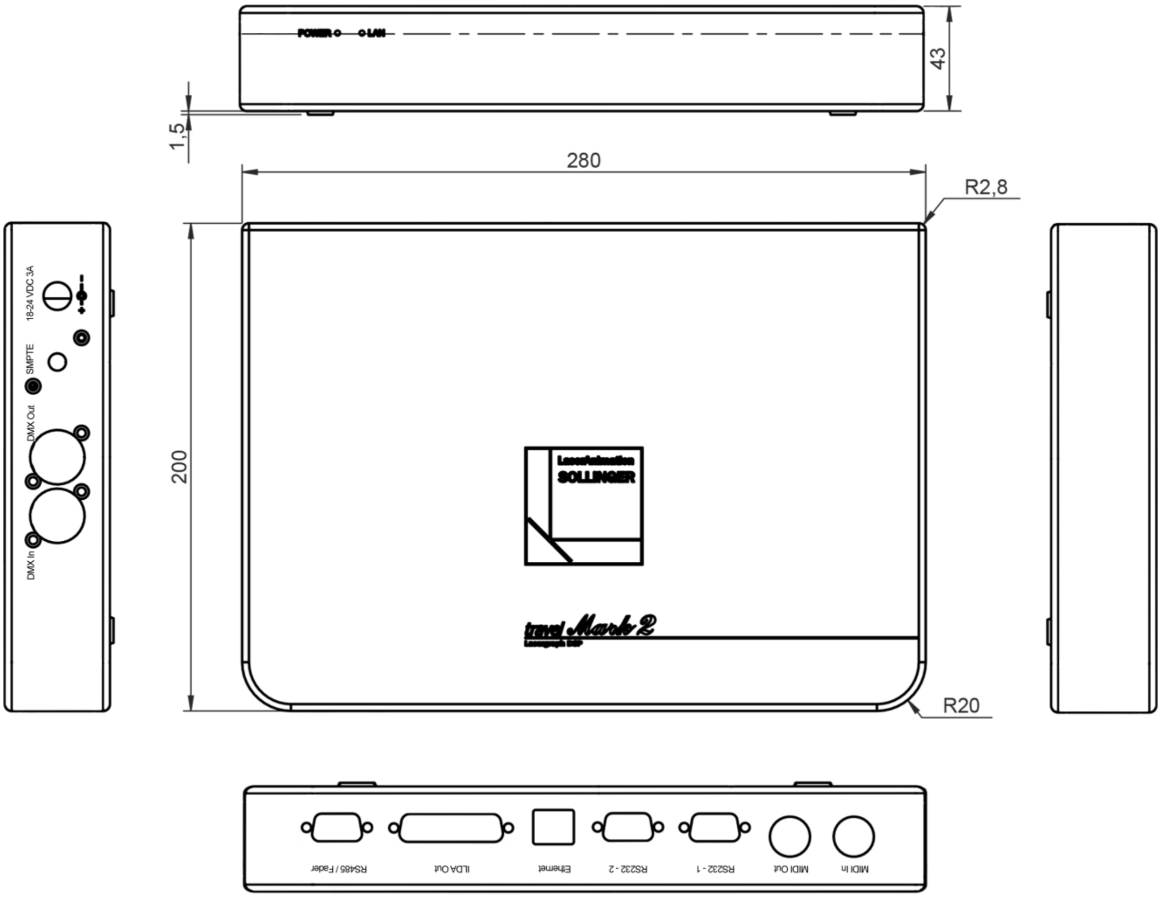
Electrical	
Power adapter	AC input: 100-240 VAC /1.5A
	Input frequency: 50-60 Hz
Input power	50 W maximum, typical < 20 W

Connectors	
18-24 VDC 3A	DC Input for connection of the included power adapter
ILDA Out	D-sub 25 female for connection of one scanning unit
Ethernet	Standard RJ-45 jack
DMX In/Out	5-pin XLR male/female
SMPTE (In/Out)	Audio jack 3,5mm to connect adapter: Jack to 2 x RCA/cinch
MIDI In/Out	5-pin DIN Receptacles
RS485 / Fader	9-pin D-sub female
2 x RS 232	9-pin D-sub male (standard RS232)

Operation Displays	
POWER	If mains power is available the LED will light up
LAN	Indicates network activity

Physical	
Ambient temperature range	+5°C to +40°C
Degree of protection	IP 20
Weight	1.3 kg
Dimensions (L x W x H)	280 mm x 200 mm x 40 mm

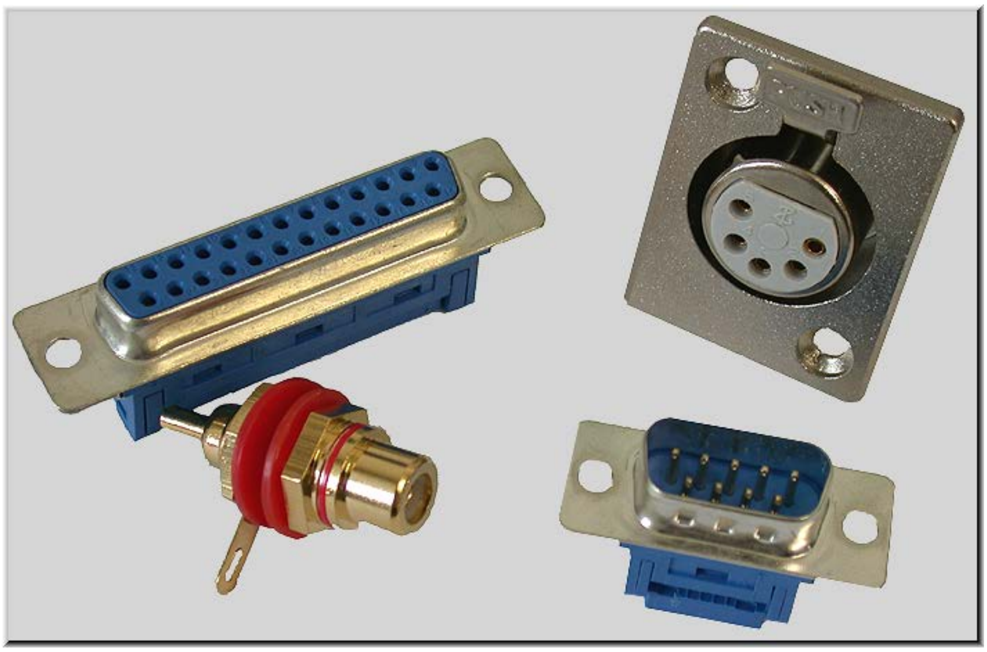
Technical specifications are subject to change without prior notice.



Pinouts

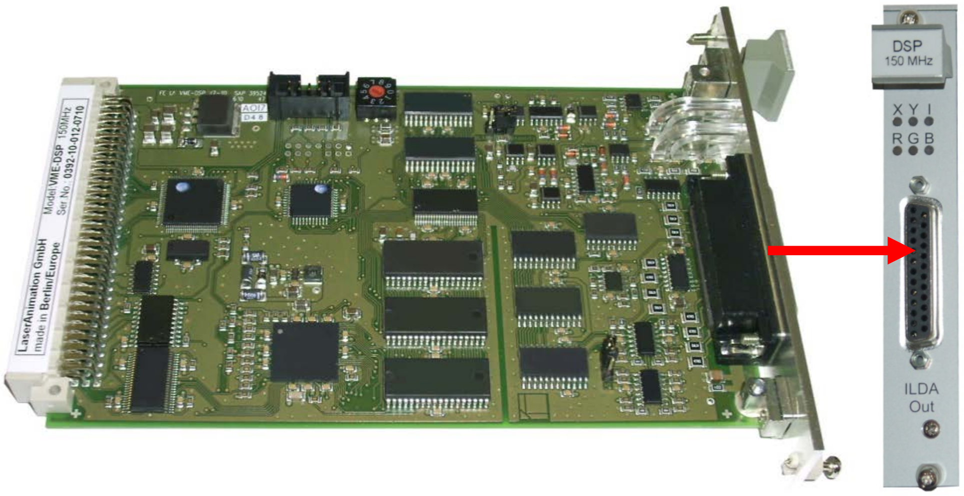
Technical Data

Pinouts



DSP Card 150MHz (ILDA Standard)	Connection of one projection unit with an ILDA compliant output
DSP Card 150 MHz (LaserAnimation Standard)	Connection for adapter “DSP to ILDA” to control one projection unit
RS232	Connection for media controller (or laser power supply)
RS485 (Fader)	Connection of a faderboard unit
LAN (TCP/IP)	Connection with PC or laptop via Ethernet network
MIDI In	Connection of a MIDI keyboard
MIDI Out	To connect e.g. a MIDI peripheral card
DMX 512	For DMX splitter containing DMX In and Out DMX In: For connection of a DMX lighting console DMX Out: To connect fog machines, dimmers or other effect devices
SMPTE In, Out	E.g. for connection of a video player

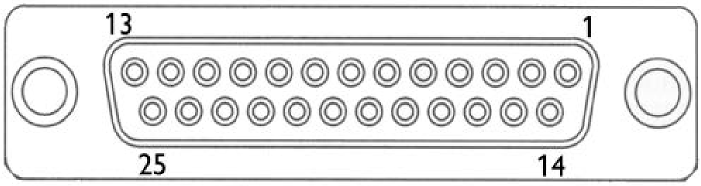
DSP Card 150 MHz ILDA Standard



1. LEDs

X	X-signal of the scanner (duo LED: red / green)
Y	Y-signal of the scanner (duo LED: red / green)
I	Intensity (yellow LED)
R	Red (red LED)
G	Green (green LED)
B	Blue (blue LED)

2. "ILDA Out" 25pin D-Sub female

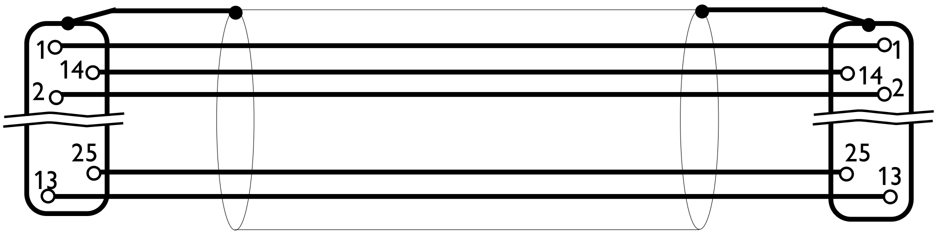


PIN No.	Signal	Level	Meaning
1	X+	-10V ... +10V against X-	Beam position: +10V : right 0V : center - 10V : left
2	Y+	-10V ... +10V against Y-	Beam position: +10V : top 0V : center - 10V : bottom
3	Intensity+	0V / +5V against Intensity-	0V: blanked 5V: beam on
4	Interlock A	Interlock loop	Connected with Interlock B
5	Red+ (Color 1)	0V ... +5V against Red-	0V : 0% red 5V : 100% red
6	Green+ (Color 2)	0V ... +5V against Green-	0V : 0% green 5V : 100% green
7	Blue+ (Color 3)	0V ... +5V against Blue-	0V : 0% blue 5V : 100% blue
8	Indigo+ (Color 5)	0V ... +5V against Indigo-	0V : 0% indigo 5V : 100% indigo
9	Yellow+ (Color 6)	0V ... +5V against Yellow-	0V : 0% yellow 5V : 100% yellow
10	Cyan+ (Color 4)	0V...+5V against Cyan-	0V : 0% cyan 5V : 100% cyan
11	3D L/R+	0V / +5V against 3D L/R-	3D left/right: 0V: left picture 5V: right picture
12	reserved	Return lead for the projector	
13	Shutter	0V ... +5V	0V: Shutter closed 5V: Shutter open
14	X-	Return lead for X+	
15	Y-	Return lead for Y+	

16	Intensity-	Return lead for Intensity+	
17	Interlock B	Interlock loop	Connected with Interlock A
18	Red-	Return lead for Red+	
19	Green-	Return lead for Green+	
20	Blue-	Return lead for Blue+	
21	Indigo-	Return lead for Indigo+	
22	Yellow-	Return lead for Yellow+	
23	Cyan-	Return lead for Cyan+	
24	3D L/R-	Return lead for 3D L/R+	
25	GND		

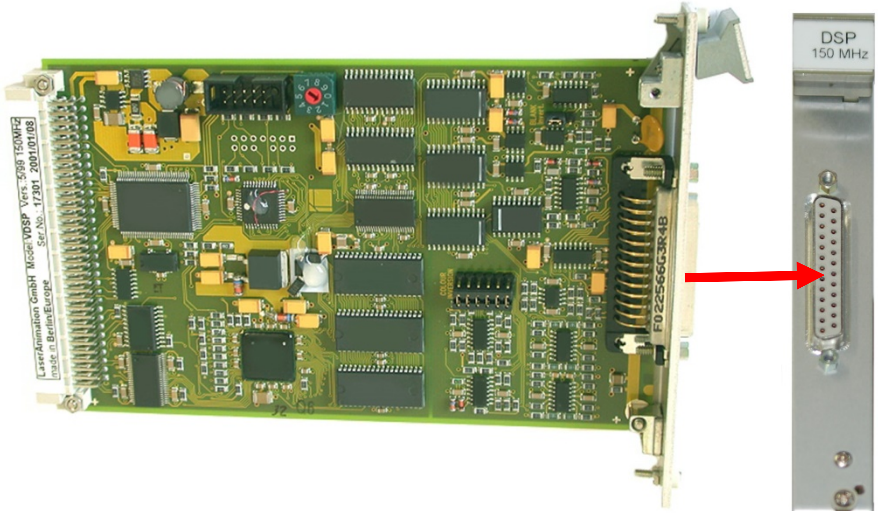
GND: Ground

Recommended cable (maximum length: 50m)

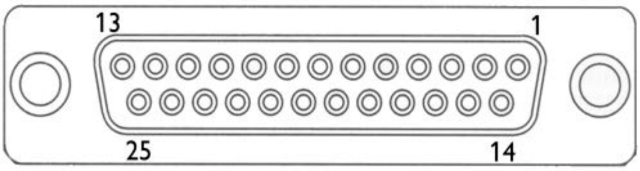


25pin D-Sub male shielded cable 25pin D-Sub female

DSP Card 150 MHz LaserAnimation Standard



25pin D-Sub female output



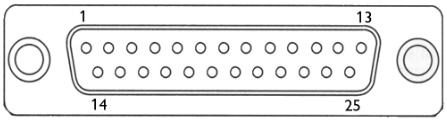
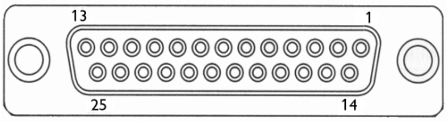
PIN No.	PIN Assignment	Level
1	X	-10V ... +10V
2	Y	-10V ... +10V
3	Blank (beam on/off)	TTL (0V beam on, +5V beam off)
4	Blue (Color 3)	0V ... +5V
5	Green (Color 2)	0V ... +5V
6	Green (Color 2)	0V ... +5V
7	Cyan (Color 4)	0V ... +5V
8	Yellow (Color 6)	0V ... +5V
9	GND	
10	Logoshutter	0V ... +5V
11	Beambrush	0V ... +5V
12	DU	
13	P7	TTL
14	Red (Color 1)	0V ... +5V

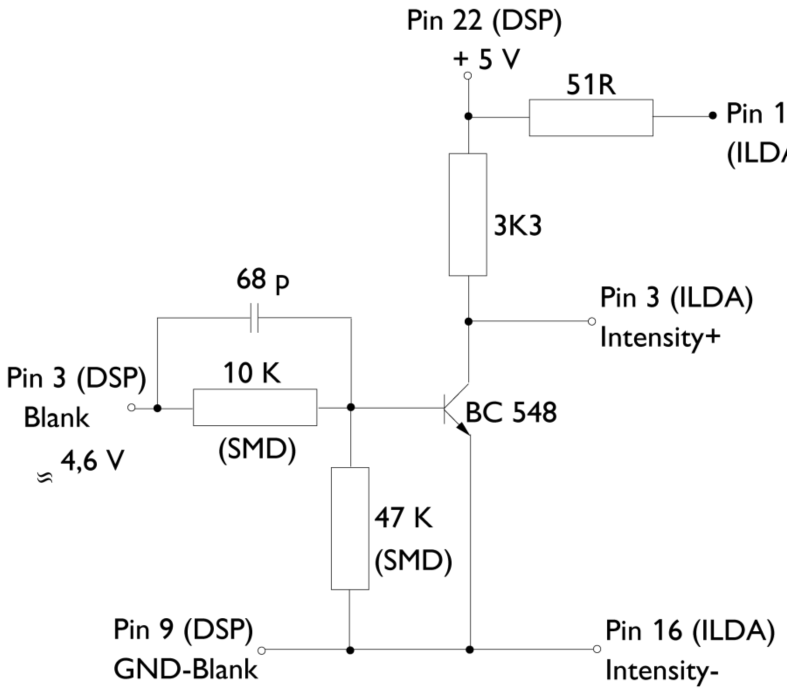
15	3D L/R	TTL (0V: 3D left, +5V: 3D right)
16	3D on/off	TTL (0V: 3D on, +5V: 3D off)
17	GND	
18	Red (Color 1)	0V ... +5V
19	Blue (Color 3)	0V ... +5V
20	Indigo (Color 5)	0V ... +5V
21	Blank (beam on/off)	TTL (0V beam on, +5V beam off)
22	+5V (maximum 500 mA) Supply voltage	
23	GND	
24	DU	
25	DU	

TTL: 0V or +5V
DU: Don't use!
GND: Ground

Adapter DSP to ILDA

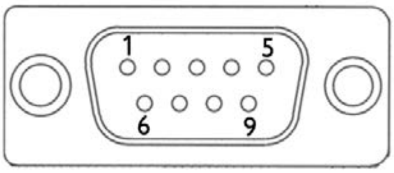
This adapter is only necessary for DSP cards with LaserAnimation Standard.

PIN	25pol. male DSP	25pol. female ILDA	PIN
			
1	X	X+	1
2	Y	Y+	2
3	Blank	Intensity+	3
4	Blue (Color 3)	Blue+	7
5	Green (Color 2)	Green+	6
7	Cyan (Color 4)	Cyan+	10
8	Yellow (Color 6)	Yellow+	9
9	GND Blank	Intensity-	16
14	Red (Color 1)	Red+	5
17	GND X/Y	X-	14
	GND X/Y	Y-	15
20	Indigo (Color 5)	Indigo+	8
22	+5V	Shutter via Rv51R	13
23	GND Color	3D L/R+	11
	GND Color	Red-	18
	GND Color	Green-	19
	GND Color	Blue-	20
	GND Color	Indigo-	21
	GND Color	Yellow-	22
	GND Color	Cyan-	23
	GND Color	3D L/R-	24
	GND Color	GND	25
		Interlock A	4
		Interlock B	17
Connector Housing			Shield


Pin 22 (DSP)
+ 5 V
51R
Pin 13
(ILDA) Shutter
3K3
68 p
Pin 3 (ILDA)
Intensity+
Pin 3 (DSP)
Blank
≈ 4,6 V
10 K
(SMD)
BC 548
47 K
(SMD)
Pin 9 (DSP)
GND-Blank
Pin 16 (ILDA)
Intensity-

RS232 (Modem)

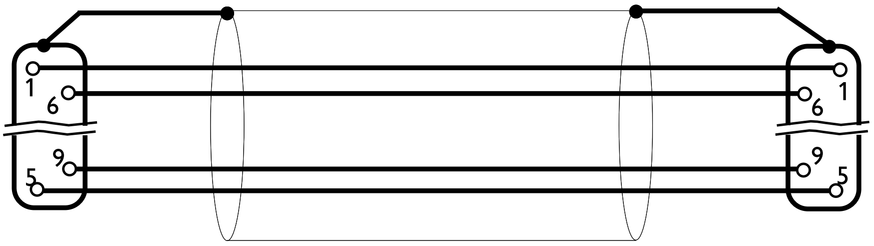
9pin D-Sub connector male



PIN No.	PIN Assignment	Meaning
1	CD*	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI*	Ring Indicator

*The signals CD and RI will be used by modems only.
GND: Ground

Recommended cable (maximum length: 15m*)

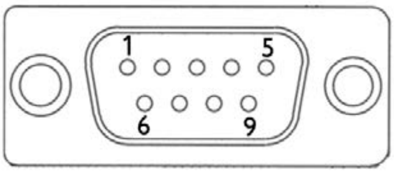


9pin D-Sub male shielded cable 9pin D-Sub female

(*By using low-capacitance cables, the maximum distance can be increased)

RS232 (Tablet)

9pin D-Sub connector male



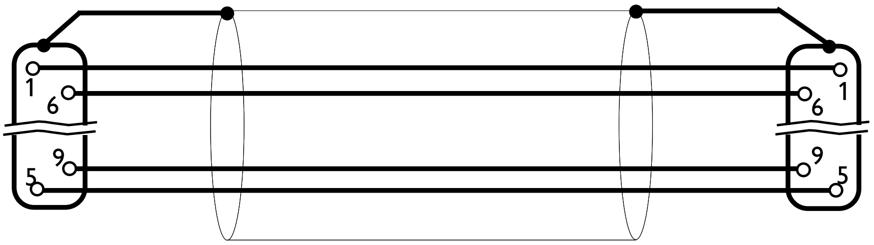
Notes:

- This connector was originally provided to connect a digitizer tablet.
- **If another RS232 device will be connected to this interface (e. g. laser power supply) pin 9 must not be connected!**

PIN No.	PIN Assignment	Meaning
1	NC	
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	+12V (250 mA)	Supply Voltage

NC: Not connected
GND: Ground

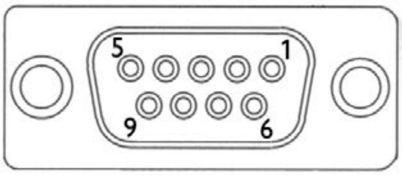
Recommended cable (maximum length: 10m)



9pin D-Sub male shielded cable 9pin D-Sub female

RS485 (Fader)

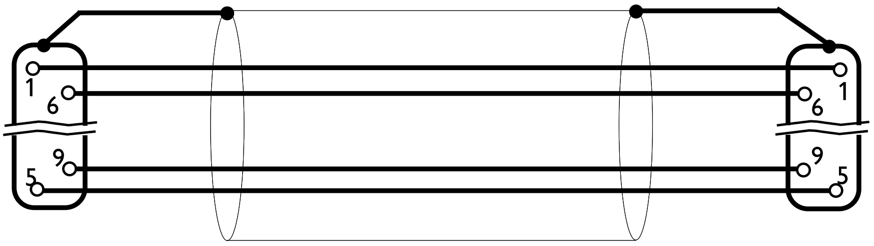
9pin D-Sub-connector female



PIN No.	PIN Assignment	Meaning
1	TXD+	Transmit Data not inverted
2	RXD+	Receive Data not inverted
3	TXD-	Transmit Data inverted
4	RXD-	Receive Data inverted
5	GND	Ground (shield)
6	+12V	Supply voltage
7	+12V	Supply voltage
8	GND	
9	GND	

GND: Ground

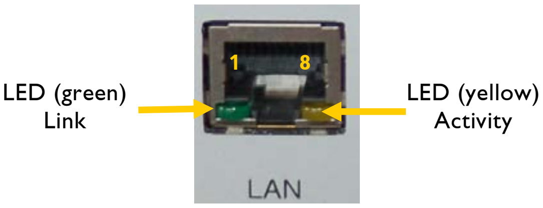
Recommended cable (maximum length: 30m)



9pin D-Sub male shielded cable 9pin D-Sub female

LAN

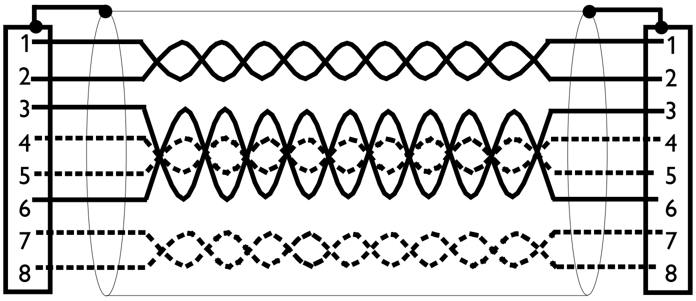
8pin RJ-45 Jack



For connection to a switch UTP (Unshielded Twisted Pair) Ethernet cable can be used.

PIN No.	PIN Assignment
1	TXD+
2	TXD-
3	RXD+
4	not used
5	not used
6	RXD-
7	not used
8	not used

Recommended cable (minimum length 2m, maximum length 50m)



RJ-45 plug shielded twisted pair cable RJ-45 plug

Note:

The pairs 4/5 and 7/8 are not used by the *workstation Mark 2.* So it is not necessary to connect them.

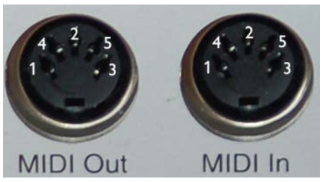
Note:

Two units can be directly connected without a hub. In this case a crossed twisted pair cable is required and have to be connected as follows:

	RJ-45	RJ-45	
PIN	Assignment	Assignment	PIN
1	TXD+	RXD+	3
2	TXD-	RXD-	6
3	RXD+	TXD+	1
6	RXD-	TXD-	2

MIDI In, MIDI Out

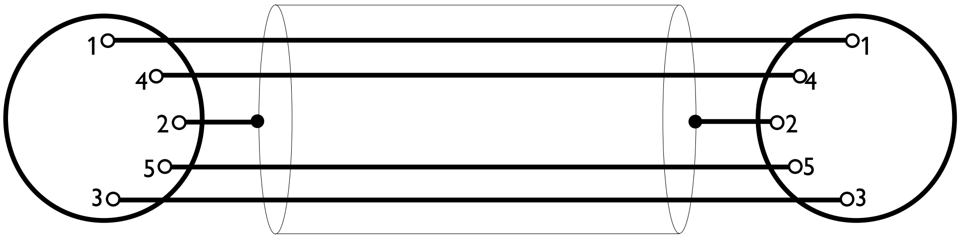
5pin DIN Receptacles



PIN No.	PIN Assignment MIDI In	PIN Assignment MIDI Out
1	+12V (250mA)	NC
2	GND	GND
3	+12V (250mA)	NC
4	Signal +	Signal +
5	Signal -	Signal -

NC: Not Connected
GND: Ground

Recommended cable (maximum length 15m):



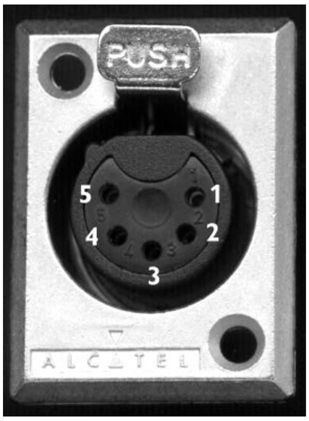
5pin DIN 180° plug male

shielded cable

5pin DIN 180° plug male

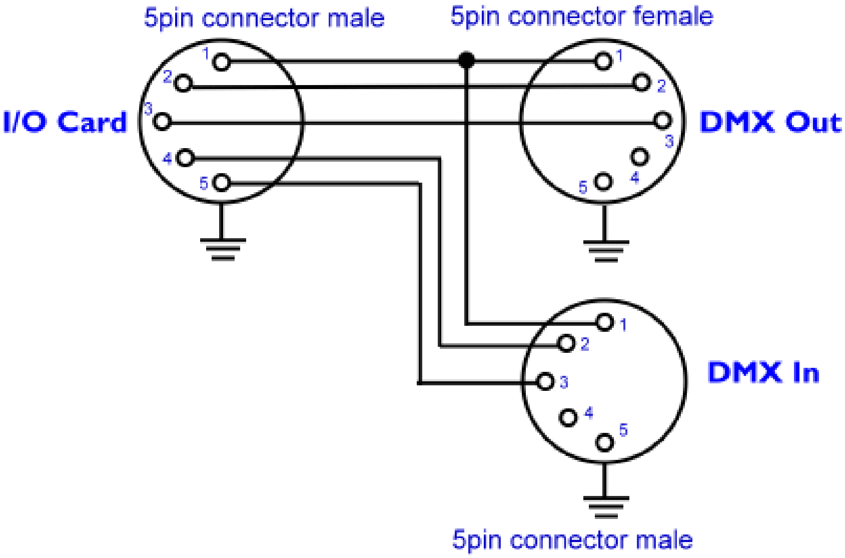
DMX512

5pin XLR female

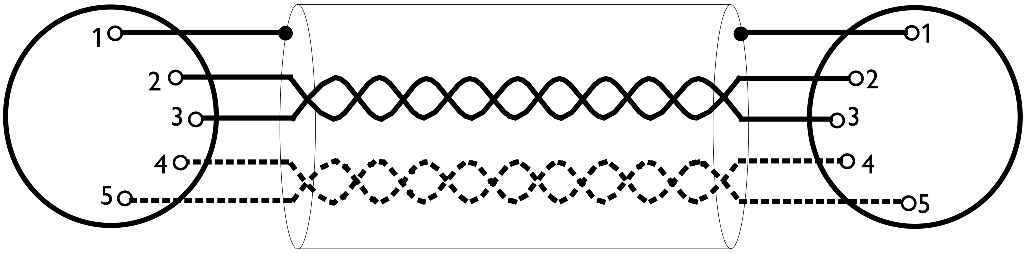


Plug to connect the DMX splitter containing DMX In and DMX Out connectors.

PIN No.	PIN Assignment
1	GND
2	DMX Out: Signal inverted
3	DMX Out: Signal not inverted
4	DMX In: Signal inverted
5	DMX In: Signal not inverted



Recommended cable (maximum length 500m)

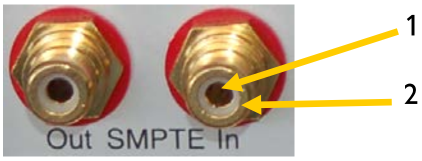


5pin XLR plug male   shielded twisted pair cable   5pin XLR plug female

Note:
Some DMX systems use 3pin XLR-connectors. In that case we recommend using the manual of the corresponding device for the pin assignment.

SMPTE In, SMPTE Out

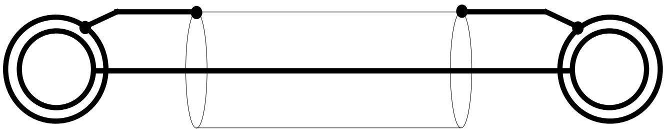
RCA Jacks



PIN No.	PIN Assignment
1	Signal
2	GND

GND: Ground

Recommended cable (maximum length 10m):



RCA plug shielded cable RCA plug

Technical Data

Electrical	
AC power input	Main voltages 100-240VAC
Input frequency	50-60Hz
Input power	*workstation Mark 2* with 2 DSP cards: 110 W maximum, typical < 40 W *workstation Mark 2* with 4 DSP cards: 150 W maximum, typical < 55 W

Physical	
Ambient temperature range	+5°C to +40°C
Degree of protection	IP 20
Weight	Approx. 6.5 kg
Dimensions	19" rack mount, 4 unit height, depth 285 mm

Connectors / Displays / Switches	
AC Mains Connector	3-pin IEC socket male
2 ILDA Outputs (optionally 4 ILDA outputs)	D-sub 25 female for connection of 2 laser projectors
LEDs XYI	Display the states of X and Y signal as well as intensity
LEDs RGB	Display the states of the colors RGB
LAN	8-pin RJ-45 jack
DMX	5-pin XLR female for connection of the included DMX In / Out splitter
SMPTE In/Out	RCA Jacks
MIDI In/Out	5-pin DIN Receptacles
RS485 Fader	9-pin D-sub female
RS232	9-pin D-sub male (standard RS232)
Tablet	9-pin D-sub male (standard RS232 if pin 9 is not connected)

Technical specifications are subject to change without prior notice.