SAMLight Manual

04.11.2022

Table of Contents

1 Introduction

1.1 Overview

Welcome to the SAMLight scanner application. This documentation describes the standard scanner application SAMLight. SAMLight is an application to control scan heads and lasers in order to do marking on different materials, to do 3D marking, welding, cutting and many more. The user interacts using the graphical user interface including dialogs and editors. The options Optic3D, SAM3D, Marking On The Fly, Flash, Client Control Interface, and MultiHead marking are also explained. In more detail:

1. Standard SAMLight functionality:

Standard features are marking of objects and entities either created in the SAMLight editor or loaded / imported using the Load or Import functionality. Standard objects and entities are:

- Geometries like rectangle, ellipse, lines, points
- Barcodes
- Bitmaps
- Serial Numbers
- Date Time Objects
- Motion Controllers
- Control objects which are:
  - Set Output Bits (digital signals)
  - DAC output (analog signals)
  - Timer
  - Wait for Input
  - Override Power / Frequency / Speed
  - Start an executable
  - Motion Control Objects
  - MOTF Control Objects
- Import / Export of various file formats (Adobe Illustrator, DXF, etc...)
- Mark Preview Window
- Assign parameters like Speed, Frequency, Power, Laser Delays to Pens for marking

Additionally there are several marking modes, as:

- Using external Trigger to start marking
- Teach / Relocate Reference Points
- Step / Repeat
- Bitmap Marking
- Marking On The Fly
- Splitting modes: 1D, 1D MOTF, 2D, Angular, Bitmap Splitting

For Real Time entity manipulation there is a tool called Data Wizard. This tool can prepare entities to be grouped for hatching or to sort the marking order if these entities were imported using the Import functionality. The Data Wizard can hugely increase marking performance.

- Create / Edit language resource files to adapt the SAMLight GUI to your language
- Create custom fonts with the laser font converter

2. Optic3D Marking Functionality:

Load and generate 3D data. The marking focus will be shifted in Z-direction by an additional focal lens. This can be used for Deep Engraving and marking 3D surfaces. More details will be given in the chapter 'Option Optic3D'.

3. SAM3D Marking - Rapid Prototyping / Stereolithography:

Import / Export 3D data called STL or CLI files. The 3D will be done by slicing the source file then moving the marking target object in Z-direction with a motion device between marking the slices. This means the object is marked layer for layer. Possible features are Up- and Downskin Marking and Beam Compensation. More details will be given in the chapter 'Option SAM3D'.

4. Marking On The Fly:

This functionality will mark data on a target object that is constantly moving in X or Y direction (in some cases it is also possible to deflect in X and Y direction simultaneously). More details will be given in the chapter 'Option MOTF'.

5. Flash:

USC-2 cards can be run in Standalone Mode. Therefore they have to be connected via Ethernet Telnet Interface or RS232 plain ASCII terminal. This will allow to give commands to the USC-2 like Start Mark, Get Time, ... without an instance of SAMLight is running. It is also possible to access more than one scanner card from the PC. This is called MultiCard. More details will be given in the chapter 'Option Flash'.

6. Remote Control - Client Control Interface:

The Client Control Interface is based on ActiveX / COM calls sent to SAMLight from an external program on the same PC or even from an application on a different PC connected via Ethernet. The Client Control Interface supplies a huge amount of functions used to load, save, edit and mark the jobs in SAMLight and to work with all kinds of entities inside that job. More details will be given in the chapter 'Client Control Interface'.

7. Multiple Scanner Controller Cards:

It is possible to control more than one scanner card at the same time. Differentiate between MultiHead and Head2. In MultiHead up to six different Scanner Controller Cards can be controlled by one instance of SAMLight. Each scan head having its own working field in the View2D and its own laser source. Head2 is used for USC-2, RTC4 and RTC5 cards. Then a second scan head can be connected to the same card and will mark the same content as the primary head.

1.2 Position within the system

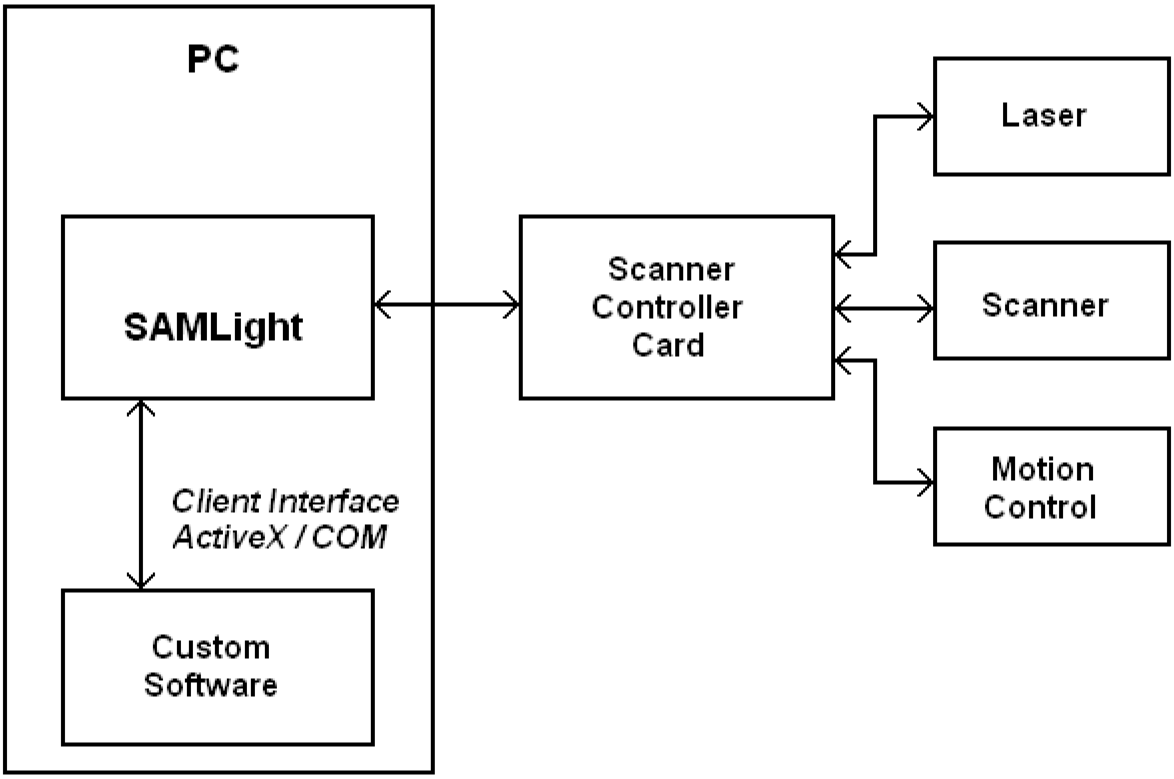


Figure 1: Position within the system

Possible configurations for using SAMLight are:

1. One USC or RTC card with SAMLight:

A single scanner controller card can be included in the system. For SAMLight standard mode simply connect the USC card via USB cable or if using a USC-2 it is also possible to connect via Ethernet. The RTC card has to be fitted in one of the PCI slots of the PC. Configuration of the hardware is explained in chapter "Hardware Settings".

2. One USC-2 card using Flash mode:

Flash mode is only available for USC-2 cards. For the Flash mode the USC-2 doesn't have to be connected to the PC via USB cable but it is necessary to connect it via an RS232 terminal to send ASCII commands to the card or connect via a Telnet Client using an Ethernet connection. However it is not necessary to run SAMLight.

3. Remote control of SAMLight:

It is possible to communicate with SAMLight via the Client Control Interface. The Client Control application can be run inside the same PC or it can be run on a remote PC and then communicate via TCP or plain ASCII commands. The Client Control commands are explained in the chapter Client Control Interface.

4. Using more than one Scanner Controller Card:

It is possible to access more than one scanner controller card at the same time. This feature is called Multi-Head. One can address up to 6 scanner controller cards with SAMLight. Then each of the cards control a unique laser. If using the Head2 feature one scanner controller card can control 2 scan heads using one single laser. The marking of the secondary scan head is the same as for the primary scan head. This mode is also possible with the Flash. If there are more than one USC-2 card, each card can be addressed using for e.g. the visible USC-server. If using the visible USC server with more than one USC-2 card please be careful when using the button InfoView, because it is not always clear which of the cards is used for this button.

5. Using Job IO Selection mode:

The Job IO Selection mode is a kind of half SAMLight plus a half of Remote control. The job that should be marked is issued via the Input pins of the card. These are the Opto_Ins or Digi_Ins of USC cards or the Digital_Ins of RTC cards. This mode can be enabled when SAMLight is already running or it can be enabled while in Flash mode. Detailed description will be given in chapter Mark for SAMLight and chapter Option Flash for the Flash mode.

1.3 Safety

Operational safety:

The goods supplied by SCAPS GmbH are designed to control a laser scanner system. The machine builder is solely responsible for strictly complying with all applicable and relevant country-specific safety regulations regarding installation and operation of the system.

The goods delivered by SCAPS GmbH must not take over any emergency stop and other safety relevant functions!

The Machinery Directive (2006/42/EC) does not apply to the products supplied by SCAPS GmbH, as there are no mechanical moving parts. Furthermore, no safety function according to the mentioned definition is realized via the USC and DSC/DSD modules.

Laser Safety:

Laser radiation can affect a person's health or otherwise cause irreparable damage. Compliance with all relevant laser safety regulations must be ensured prior to installation and operation.

Safe turn on procedure to avoid uncontrolled laser emission. Turn off procedure conversely (4 to 1), unless otherwise needed by the device.

1. Scanner controller
2. Peripheral devices
3. Scan system
4. Laser

Electrical Safety/EMC:

The product is supplied without an enclosure. The machine builder is solely responsible for the strict compliance with all relevant country-specific safety regulations for the integration and operation of the delivered goods.

The Low Voltage Directive (2014/35/EU) does not apply to the products supplied by SCAPS GmbH, as the voltage supply of all products is below the 75 V direct current (DC) limit. Therefore, these fall outside the scope of the Low Voltage Directive.

The EMC Directive (2014/30/EU) does not apply to the products supplied by SCAPS GmbH, as these are not intended for the end user (person without knowledge of EMC) and are not otherwise made available on the market.

Wiring:

For any wiring, data or power cable connecting and disconnecting, make sure that all devices are turned off and the wires do not carry any voltage!

All cables should be only as long as needed to minimize coupling capacitance and coupling inductance. Especially unshielded cables should be as short as possible. Avoid ground loops.

Avoid long drain wires for the connection of the cable shield.

For differential data signals as for the scan head (XY2-100) and encoder (MOTF), also connect the ground wires.

Power supplies, GND, Shield:

All power sources should be grounded, the potential difference of all power supplies should be below ±5 V. Advice: Use the same ground potential for all used devices if not otherwise needed (GND, Opto_GND, Laser_GND).

PE, shield and power supply: Refer to the manual of the power supply, respectively device manufacturer and stick to them exactly. Connect PE, used cable shields and ground potentials to the same potential if the electrical specifications allow this. Relevant country-specific safety regulations regarding installation and operation of the system must be checked.

Recommendation for shielded cables: Connect one side of the shielded cable to GND to avoid ground loops. For the 25-pin scanner or 37-pin Laser, I/O cable, this can be done with the solder jumper on the bottom side on the USC card (PE to GND, SJ 1 – to SJ 4) – please check the respective manual.

If needed for EMC issues it is also possible to connect both sides of the cable shield, this decision has to be taken by the machine builder and cannot be answered in general.

2 Installation and Configuration

This chapter describes the system requirements to run SAMLight, where to download the SCAPS installer and how to install SAMLight.

2.1 System Requirements

SAMLight will run under the following Microsoft Windows operating systems:

- Windows 10 - 32/64 (complete support including Secure Boot starts with SAM version 3_6_5_20161230_004 or 3_6_0_20170116_0001)
- Windows 8 - 32/64
- Windows 7 - 32/64
- Windows Embedded Standard 7 - 32
- Windows Vista - 32/64
- Windows XP - 32/64 (only up to SAM version 3_6_5_0124_20170907 or 3_6_0_0012_20170714, newer SAM versions are not supported)
- Windows XP Embedded - 32 (only up to version 3_6_5_0124_20170907 or 3_6_0_0012_20170714, newer SAM versions are not supported)
- Windows2000 (only up to version 3_6_5_0124_20170907 or 3_6_0_0012_20170714, newer SAM versions are not supported)

NOTE

In order to run SAMLight correctly it is necessary that all current Windows Updates have been installed. For example:

- *On Windows 7 x64, KB3033929 must be installed for it to recognize the Jungo driver signature.*
- *On Windows Vista, 7 and 8, KB3118401 must be installed, because of a bug in the universal C runtime.*

These updates can be installed via automatic Windows updates or manually from the provided links.

WARNING

In order to avoid unwanted laser emissions or malfunctions during the operation procedure it is necessary to deactivate the Windows hibernation or any power saving mode.

The hardware requirements are not strictly defined. Depending on the application and to get a reasonable working speed you need much better components. Minimum requirements for SAM (Light) are very low:

- 500 MHz x86-CPU
- 256 MB RAM
- 512 MB HDD

For big job files the following is recommended:

- 8 GB RAM or more
- CPU with high clock rate (SAMLight mainly runs on a single core)
- 64bit Windows (allows SAMLight to use up to 4 GB RAM)

- SSD or fast HDD

Increased hardware requirements could be necessary for the following applications:

- Many Bitmaps → more RAM required
- Large Jobs with many vector data → more RAM required
- Using mode Job IO Select → more RAM required, because of buffering of jobs
- Using complex jobs or using often calculations on the vector data → more RAM required
- Using option SAM3D → more RAM required (for slices) → because of more calculation operations for slicing; also more hard disc memory is recommended to buffer the slice data which have to be stored during the use of SAMLight; faster graphic is required for View3D
- Using option Optic3D → no additional requirements needed
- Using option MultiHead controlling many heads (up to six) → more RAM and processor power is recommended
- Using Background Camera → more GPU and processor power is required

2.2 Software Download

The newest installer of SAMLight can be downloaded at the following link: https://www.scaps.com/?id=30 or is available on a CD.

In the following <SCAPS> describes the destination of the software on your PC. It is set per default to C:\scaps\sam2d.

2.3 Software Installation

Installing SAMLight:

Run sc_sam_setup_v_3_X_X_YYYYMMDD.exe and follow the instructions.

As default, the SCAPS software will be installed into the folder **C:\scaps\sam2d**. In the following text, **<SCAPS>** is a placeholder of the software installation path.

NOTE

Administrator rights on the PC are necessary for the software installation. If you want to change the scaps installation path by a new installation, please uninstall the old version completely. If necessary, make a backup of your scaps system folder, job and correction files before. It is also recommended to make a backup of these files after successful hard- and software setup.

Files / settings created by the setup:

- environment variable %SCAPS_SAM% as installation folder, by default C:\scaps\sam2d\
- sc_*.dll and sc_*.ocx files in \WINDOWS\system32\ (for 32bit)
- sam_light.exe in <SCAPS>\samlight\
- sc_setup.exe in <SCAPS>\tools\
- sc_usc_server.exe in <SCAPS>\system\
- sc_light_settings.sam in <SCAPS>\system\ , contains all program settings for SAMLight. It is recommended to make a backup of this file after setup.
- sc_corr_table.exe in <SCAPS>\tools\.
- Some of the files created by the setup are important for the reproduction of problems. You can find the name of the files, the description and the location in the following link: Files for Problem Reproduction.

Configuration of SAMLight:

The next step is to open the software, after SAMLight has been installed successfully on the PC. Please connect your USB-Dongle or USC card via USB to your PC. If you are using a RTC card, please connect it to a free PCI slot.

After connecting the hardware to the PC, it may take some time for the drivers to be installed automatically. If you are not sure, if all drivers have been installed correctly you can check it in the windows device manager. If you are using an USC card, there should be a folder *Jungo Connectivity* and *SCAPS USC: USB Scanner Controller.* If you are using an USB-Dongle, then it should be located at *USB controllers → CBUSC 2.0.*

If you are using a RTC# card, then it should be shown as *SCANLAB.*

After that open the tool sc_setup.exe and type in your password.

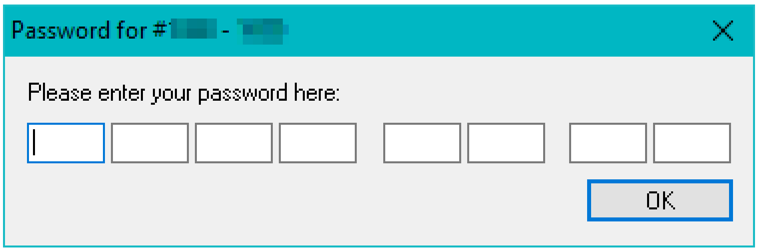


Figure 2: Password dialog.

It is also possible to create a text file named *sc_password.txt* which contains the password for the corresponding Dongle. This file can be saved in *C:\scaps\sam2d\system* .If the software cannot find the file *sc_#_#.scl* it searched for the file *sc_password.txt* and uses the saved password.

If you are using an older password with 16 or 24 letters, keep the fields on the right empty.

There is a short-cut to plug in the password. At first select the password with Ctrl+C. Then select the first box at the password dialog and press Ctrl+V. After that press two times ENTER. Now the password should be inserted correctly.

At the menu bar *Resource* you can select the wanted language.

2.4 Hardware Installation

For installing the SCAPS hardware please connect the hardware with the PC as described in the following steps:

Connection USC with a PC:

The connection between the USC card and the PC can be done by USB or by Ethernet (except USC-1). Please make sure that the USB cable is as short as possible to avoid connection problems. After connecting the hardware to the PC it may take some time until the drivers are automatically installed. If you are not sure, if all drivers have been installed correctly, you can check it in the windows device manager. If you are using an USC card, there should be a folder *Jungo Connectivity* and *SCAPS USC: USB Scanner Controller.* If you are using an USB-Dongle, then it should be located at *USB controllers → CBUSC 2.0.*

Read the information in this chapter very careful. Wrong power supply or inappropriate components could result in hardware damage!

The USC will be delivered without housing, all relevant safety regulations for integration and operation have to be complied. Mounting the scanner controller in an small housing will require an extra air ventilation to avoid a critical increase of temperature.

Power supply:

It's highly recommended to supply the USC board with an external power supply. By default, the jumper for bus powered mode is closed for first tests and evaluation purpose. This jumper (USC-1 and USC-2 only) has to be removed before an external power supply is connected. Please verify to connect the USC board to the external power supply before the voltage is applied.

Installing a SCAPS USB dongle:

Plug the USB-dongle in at the USB connector of the PC (not necessary if there is a USC-1, USC-2 or USC-3 scanner card installed).

Installing scanner driver cards from a third party manufacturer:

Install scanner driver board and the software drivers as described in the manufacturer's manual. Copy the correction file and the driver files (*.dll) in a separate folder of the hard disk.
If using an RTC card there should be an entry RTC... in the entry SCANLAB.

Scan head:

As a general rule, only shielded data cables should be used to connect a scan head to the scanner controller. It's elemental that the cable shield is aligned to the electric shield of the scanner controller. To avoid electrical damage to the scan head or scanner controller, ensure that all components are de-energized during every connection and disconnection. The potential difference between the mass potentials of the scanner controller and scan head should not exceed ±7 V.

The solders jumpers on the bottom can be used to connect the cable shield to the USC GND. It's strongly recommended to connect the USC GND with the cable shield of the scanner.

2.5 Copy or Backup Settings

If you want to **update the SAMLight version on your PC**, please first make a backup of the system (by copying the 8 files mentioned below to a place safe).

If you want to **build up a new system on another PC** with the same parameters as the old one - the SAMLight version may be different - , please first install the new SAMLight version and then copy the following 9 files from the old system folder to the new one:

File	Location	Description
sc_light_settings.sam	<SCAPS>\system	settings file
sc_motion_settings.txt	<SCAPS>\system	motion settings
sc_motion_<MOTION TYPE>_settings.txt	<SCAPS>\system	motion stepper settings (depends on the motion type defined in sc_motion_settings.txt, refer to Motion_Controller)
sc_usc_card_ids.txt	<SCAPS>\system	IDs file (only when using multiple cards)
sc_usc.cfg	<SCAPS>\system	configuration file
hosts.allow	<SCAPS>\system	
hosts.deny	<SCAPS>\system	
sc_resource_sc_<LANGUAGE>.sam	<SCAPS>\system	Only required if the language is NOT English.
sc_resource_settings.sam	<SCAPS>\system	

It is also necessary to copy the correction file - for example "cor_neutral.ucf" - from the old system to the new one. The path of the correction file is defined in sc_light_settings.sam. Please refer, depending on the scanner controller card, to chapter Hardware Settings.

Since version 4_0_5_0005 it is possible to use Help - Generate backup zip to generate a backup zip folder.

3 sc_usc_server.exe

The sc_usc_server.exe is located in the folder <SCAPS>\system\.

Per default it is running in the background in the non visible mode.

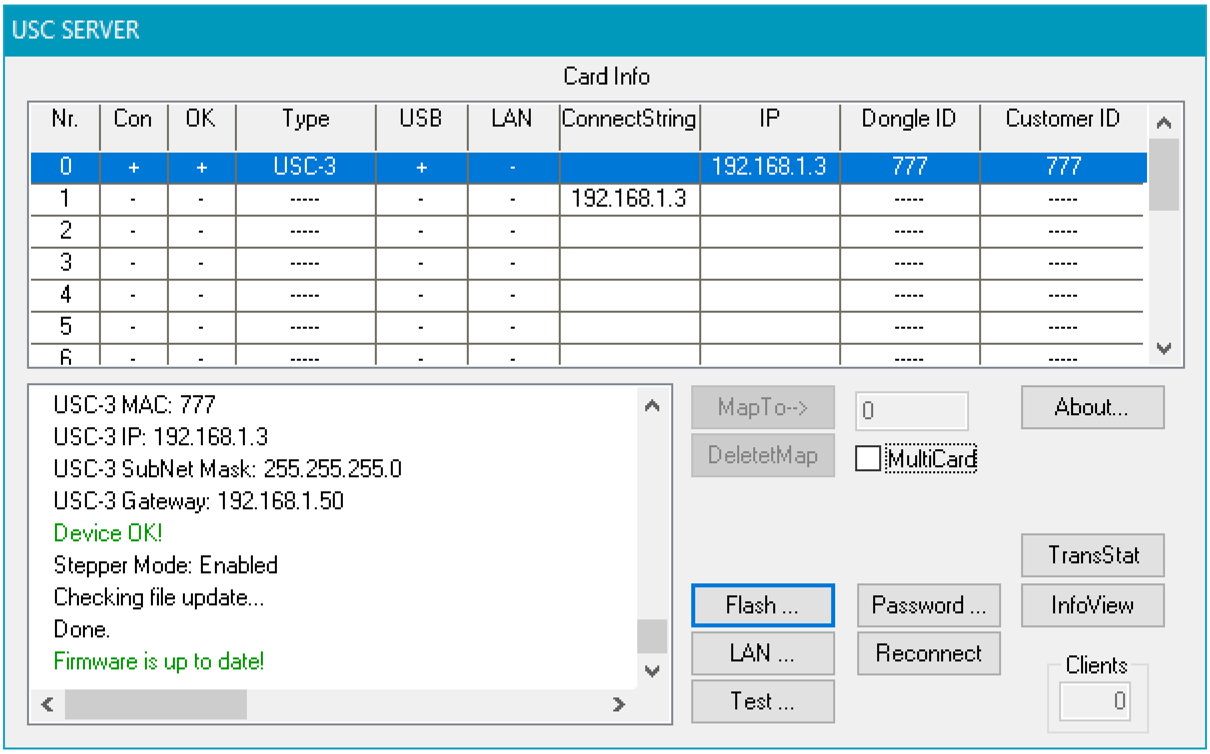


Figure 3: sc_usc_server.exe

The USC Server can be started with specific command line parameters (e.g. for visible mode and for firmware update or flash initialization).

In the following, further information is given for:

- **Visible mode**
- **MultiCard**
- **Flash...**
- **LAN...**
- **Test...**
- **Password...**
- **Reconnect**
- **TransStat**

3.1 Visible Mode

Per default the sc_usc_server runs invisible in the background.
For debug and configuration, it's useful to start the sc_usc_server in the visible mode.
To change the start mode, it's necessary to close SAMLight or any other running SAM program.
Please close a running sc_usc_server with the task manager, too.

There are four options to start sc_usc_server.exe in visible mode:

1. Use the shortcut in the Windows start menu: SCAPS SAMLight --> Make USCServer visible.
2. Press "Start", choose "Run...", type in "C:\scaps\sam2d\system\sc_usc_server.exe /v", hit the enter key.
3. Create a shortcut of <SCAPS>\system\sc_usc_server.exe and expand the target with /v.
4. Set the line "VisibleMode=0" in the file <SCAPS>\system\sc_usc.cfg to "VisibleMode=1".

3.2 MultiCard

The MultiCard flag enables the option to mark with a certain head when more than one card is connected. It also activates the "MapTo-->" and "DeleteMap" buttons. With these buttons the list "Card Info" can be edited. The list will be stored in the file sc_usc_card_ids.txt.

	Single head (normal)	MultiHead	MultiCard
Number of cards	1	multiple	multiple
Number of working areas	1	multiple, overlapping	one for all heads
Splitting	no	depends on the size of the working areas	no
Marking	whole job	each head marks a certain part of the job	the active head can be selected manually; default is head 0
License	normal	+ MultiHead	+ Flash

Table 1: MultiCard vs. MultiHead

MultiCard is not MultiHead!

The MultiCard flag will be ignored, if a MultiHead license is installed.

3.3 Flash

With the Flash option the USC-2 and USC-3 cards can be used as stand alone units.
A PC is only required for job setup.

When clicking on Flash... in the sc_usc_server, the following dialog opens:

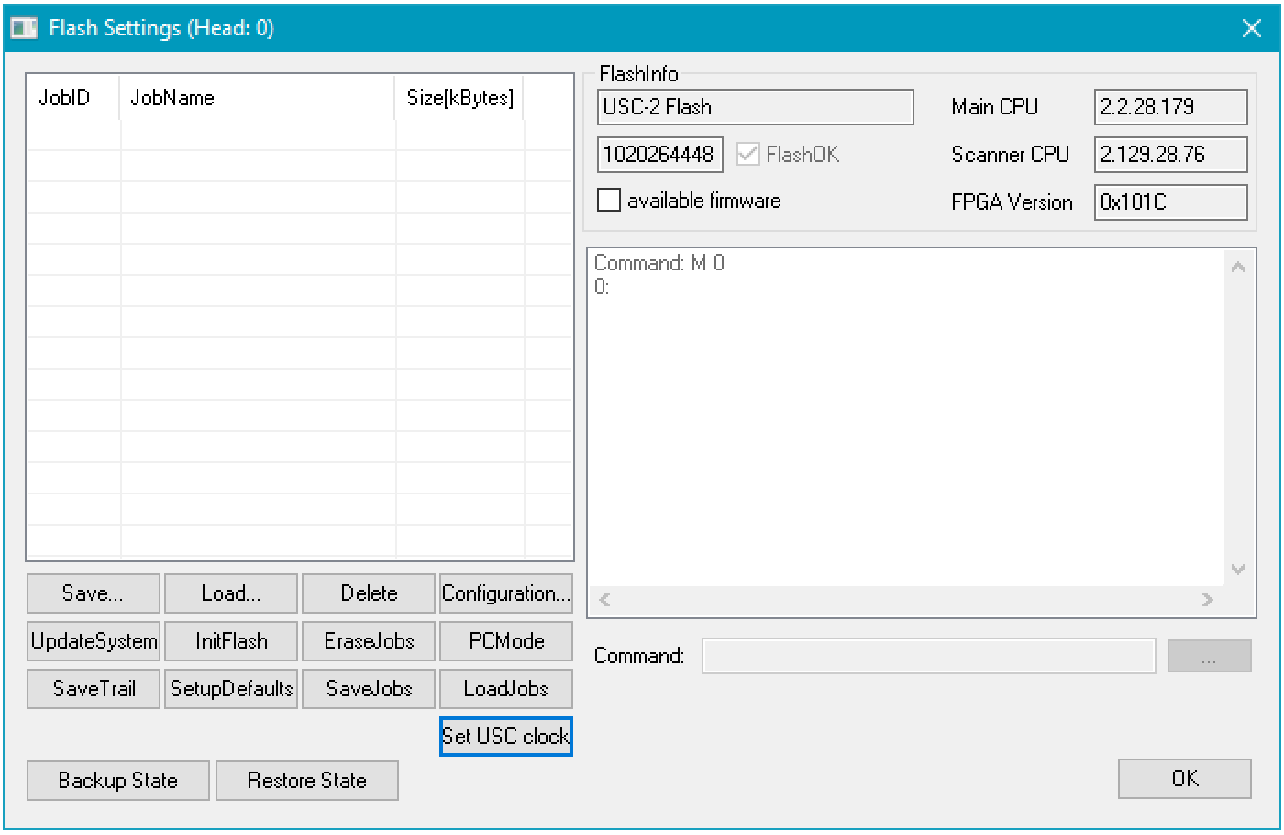


Figure 4: Flash settings

Available buttons are explained in the following:

Save: Opens the dialog to save the selected job from the job list with a desired name at a desired location.

Load: Opens the dialog to load a job into the job list. The new job will be added to the list at the first free slot.

Delete: Deletes the selected job from the flash memory.

Configuration...: Opens the Flash Configuration dialog (see figure 5 and table 2).

UpdateSystem: Initiates an update of the firmware of the USC card. This is recommended in case of the message "New firmware available" or "Firmware newer than available".

InitFlash: Initializes the complete flash memory. All jobs currently stored on the flash will be deleted and default settings will be restored.

EraseJobs: Deletes all jobs from the flash memory.

PCMode: Activates the communication to the USC card via FCI commands (in the "Command" line).

SaveTrail: Saves trail files in <SCAPS>\intermed if the trailing has been activated (USC-2 , USC-3).

SetupDefaults: Clears all customized settings on the card and restores default settings.

SaveJobs: Saves all jobs currently in the job list in a new folder created in <SCAPS>\intermed.

LoadJobs: Loads all previously saved jobs onto the card. Those jobs should follow the convention: jobnumber_jobname.unf, for example *2_infinitecircle.unf*.

Set USC clock: Synchronizes the timer on the USC card with the time of the PC.

Backup State: Takes the folders cor, config and misc on the USC-3 card and puts them together in a zip folder. Saves this niofs.tar file where the user specifies. This feature is only available for USC-3 cards.

Restore State: Asks where the niofs.tar file with backup states is located, takes it and restores it to the USC-3 card. This feature is only available for USC-3 cards.

The button *Configuration...* opens the following dialog:

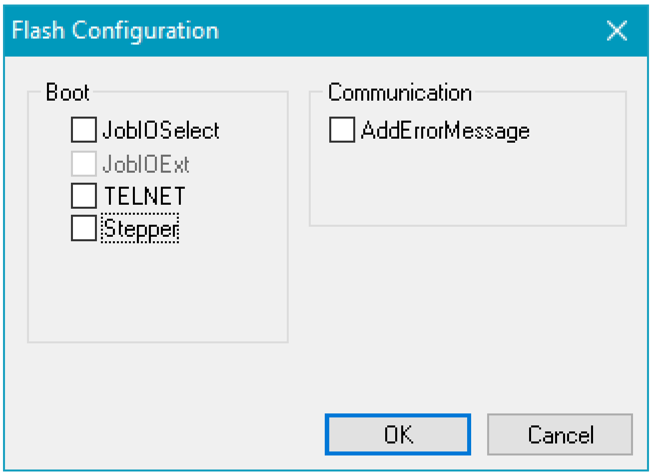


Figure 5: Flash Configuration

Checkbox	Description
JobIOSelect	After booting, the device goes into the Job IO Selection mode.
JobIOExt	In combination with Job IO Selection mode, this mode enables the inputs [0..7] of the Extension connector, so that 255 Jobs can be selected. (only available with USC-2 and USC-3)
TELNET	Activates the TELNET remote control for an USC-2 or an USC-3 card. After the activation the card can be accessed via the TELNET (RJ45) port #23. (only available with USC-2 and USC-3)
Init	Leads the FEB-1 to initialize on start up. (only available with FEB-1)
AddErrorMessage	Shows Error Message also in readable text in RS-232 and PCMode (not just the error code).

Table 2: Flash Configuration

NOTE *JobIOExt and TELNET are only available with USC-2 and USC-3.*

3.4 Password

NOTE *This button will be enabled only if a USC-2/3 or FEB-1 is detected.*

Now the eight blocks password has to be typed in.

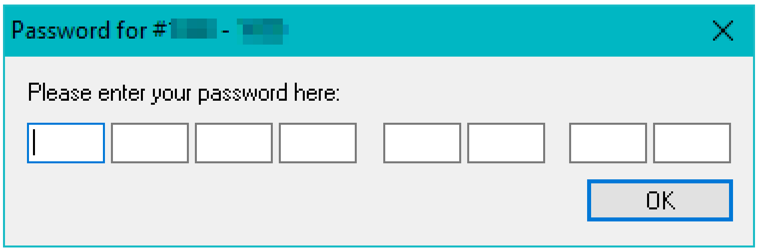
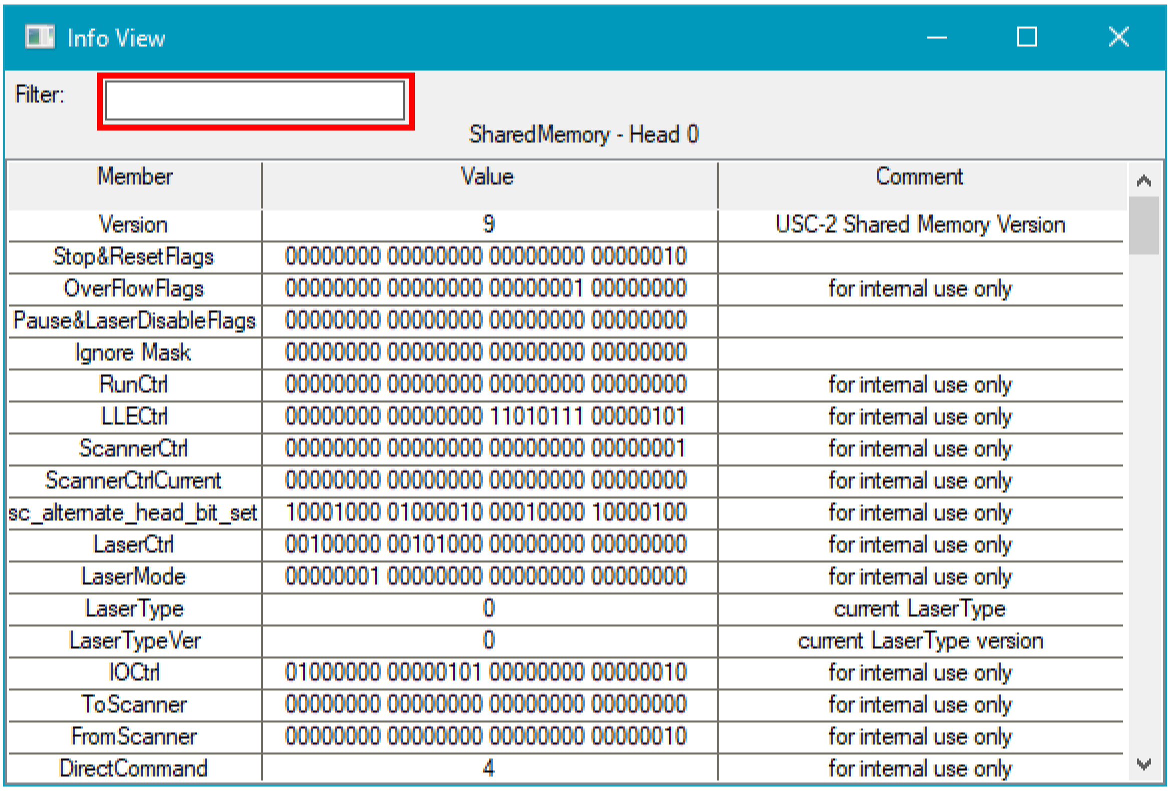


Figure 6: Password Dialog

3.5 InfoView

After selecting the InfoView button (USC-2/3 only), two new windows will open:



Member	Value	Comment
Version	9	USC-2 Shared Memory Version
Stop&ResetFlags	00000000 00000000 00000000 00000010	
OverFlowFlags	00000000 00000000 00000001 00000000	for internal use only
Pause&LaserDisableFlags	00000000 00000000 00000000 00000000	
Ignore Mask	00000000 00000000 00000000 00000000	
RunCtrl	00000000 00000000 00000000 00000000	for internal use only
LLECtrl	00000000 00000000 11010111 00000101	for internal use only
ScannerCtrl	00000000 00000000 00000000 00000001	for internal use only
ScannerCtrlCurrent	00000000 00000000 00000000 00000000	for internal use only
sc_alternate_head_bit_set	10001000 01000010 00010000 10000100	for internal use only
LaserCtrl	00100000 00101000 00000000 00000000	for internal use only
LaserMode	00000001 00000000 00000000 00000000	for internal use only
LaserType	0	current LaserType
LaserTypeVer	0	current LaserType version
IOCtrl	01000000 00000101 00000000 00000010	for internal use only
ToScanner	00000000 00000000 00000000 00000000	for internal use only
FromScanner	00000000 00000000 00000000 00000010	for internal use only
DirectCommand	4	for internal use only

Figure 7: Info View List

NOTE

USC-2/USC-3: StatusHead0 corresponds to pin 19 and 6 of the 25pin scanner connector and Status Head1 corresponds to pin 11 and 12 of the 26 pin Head2 connector.

The value displayed in the InfoView is different since the value for the USC-3 is shifted 12 bits to the left. The content is the same, but the value is just different for USC-3.

In three columns, information about different states are displayed. The first column shows the name of the state, the second the value. In the third column, a short explanation/comment will be given. The value will be updated with 100ms when SAMLight is not running, or 200ms when SAMLight is running, respectively (can be even slower when SAMLight is busy).

With the filter, you can narrow the selection to only display information on a specific topic. See Examples for some topics that might be of interest or could be helpful for e.g. troubleshooting. Multiple topics can be separated by semicolon.

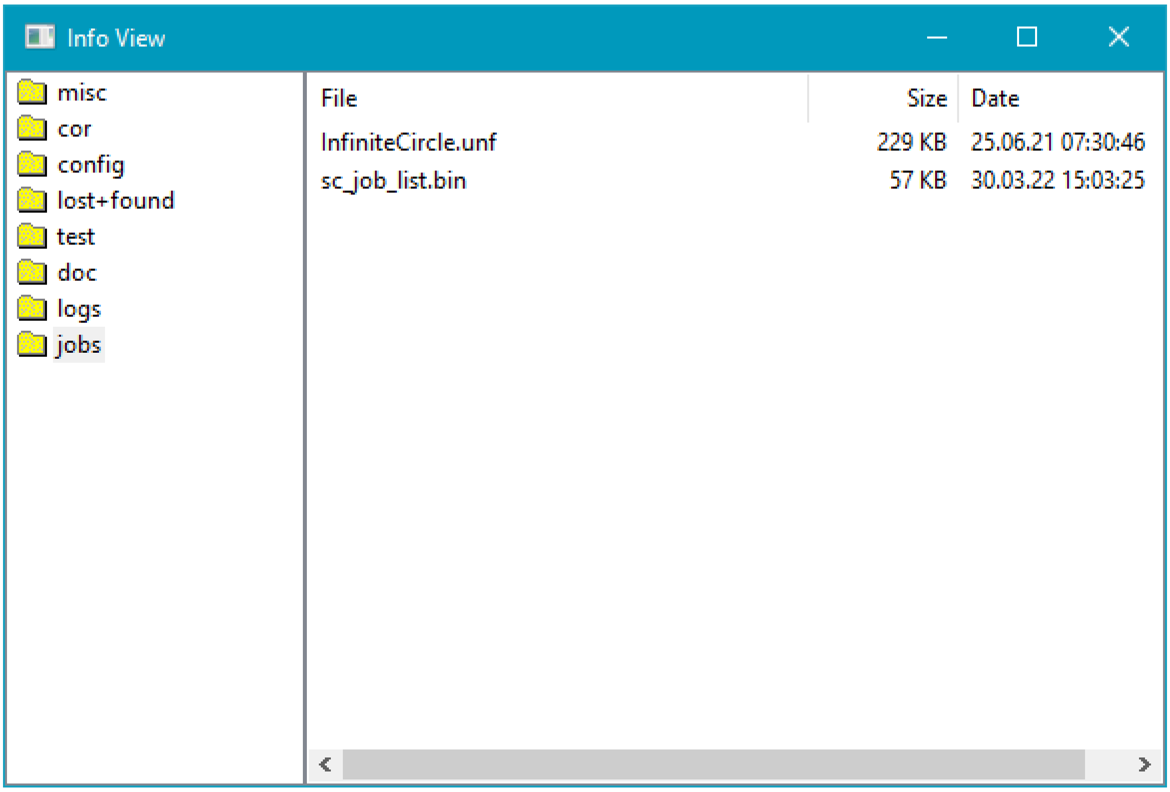


Figure 8: Info View Folder

Here, the file folders saved on the USC-card can be viewed. These contain for example settings files, log files and the FCI jobs in form of .unf files.

3.6 LAN

STOP *Make sure that the card is connected via USB when editing the LAN settings!*

NOTE *If you want to change the IP address the jumper (JP2) next to the USC-2 / USC-3 Ethernet connector has to be connected/set. If the jumper is unconnected the card can be found by the default IP address (USC-2: 192.168.1.2, USC-3: 192.168.1.3). Communication port: 8112.*

This dialog can be used to setup the Ethernet connection to the USC-2 / USC-3. Select a card from the "Card Info" list and click "LAN...". The following dialog appears. To change the IP address, Subnet mask or Gateway of the selected USC-2 / USC_3, type in the new value and click "Connect".

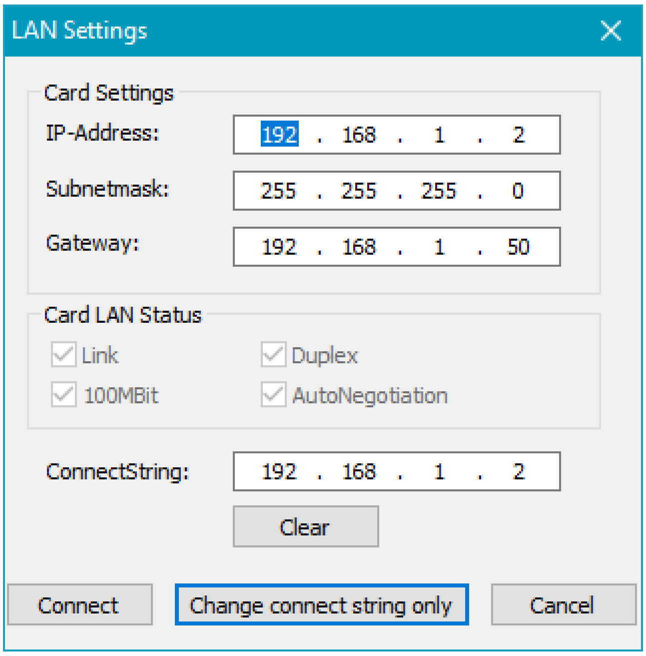


Figure 9: USC-2/3 LAN Settings Dialog

Card Settings: Setup the IP-Address, the Subnetmask and the Gateway as preferred.

Card LAN Status: The check boxes display the current LAN status of the USC-2 / USC-3 (Link, 100Mbit, Duplex and AutoNegotiation).

ConnectString: The *ConnectString* is used to generate the entry in sc_usc.cfg, when clicking *OK* or *Connect*. *Clear* deletes the entry from the file. An empty *ConnectString* will be set to the IP-Address when clicking *OK* or *Connect*.

Connect: sc_usc_server writes the Ethernet settings to the USC-2 / USC-3 and the *ConnectString* to sc_usc1.cfg. After that, sc_usc_server tries to connect to the USC-2 / USC-3 via Ethernet and closes the dialog.

Change connect string only: sc_usc_server writes the *ConnectString* to sc_usc1.cfg and closes the dialog.

Cancel: Discard changes and leave the dialog.

3.7 Reconnect

Reconnect the selected device.

NOTE *Reconnect is only possible when using USB!*

3.8 Test

The Test dialog can be used to set or display the OptoIOs as well as the analog outputs.

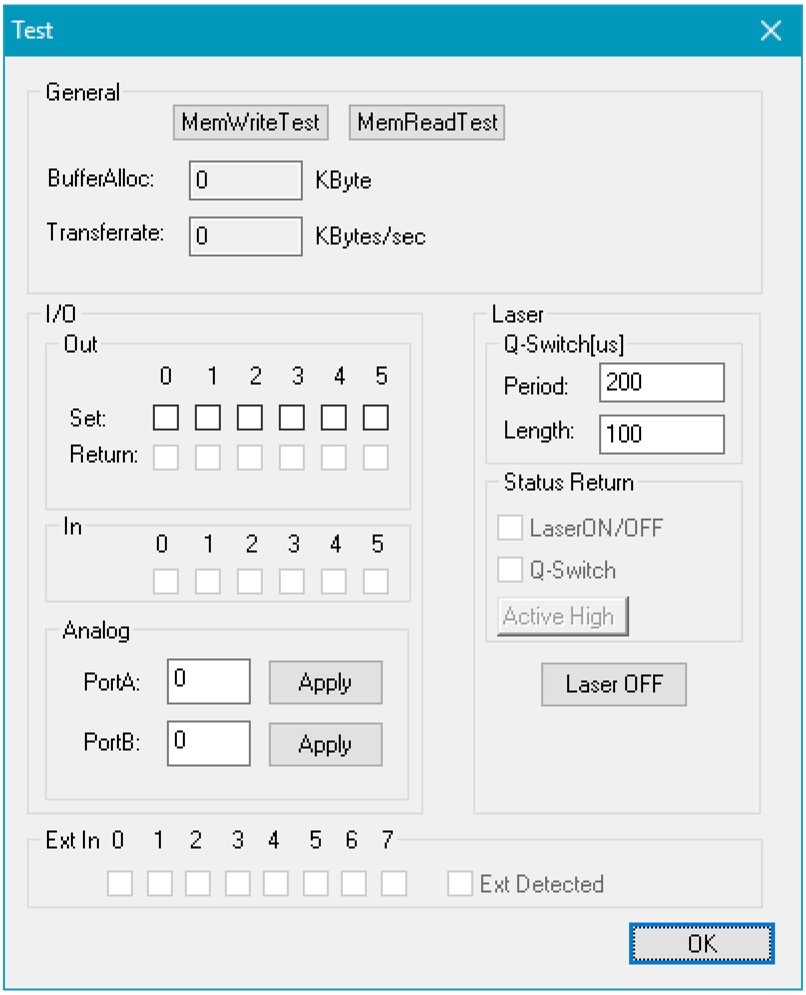


Figure 10: Test dialog

3.9 Update SD-card image USC-3

The USC-3 card contains an SD-card. Usually, this SD-card should not be unmounted from the USC card and does not need to be manipulated separately. If the required SD version differs from the current version (see red frame in figure 11), an SD-card image update is necessary for the USC-3 card. Then, please proceed as follows:

- Download the USC-3 SD card image: https://download.scaps.com/index.php?c=1&f=/downloads/usc3-sd-images/
- Unzip the USC-3 SD card image. The *.img file is needed.
- Unmount the SD card from the USC-3.
- Mount the SD card into the PC (SD card reader).
- Download an image tool for SD cards like this one: https://sourceforge.net/projects/win32diskimager/files/latest/download
- Write the USC-3 image with the image tool on the SD card.
- Safety remove the SD card from the PC. (important)
- Mount the SD card into the USC-3.

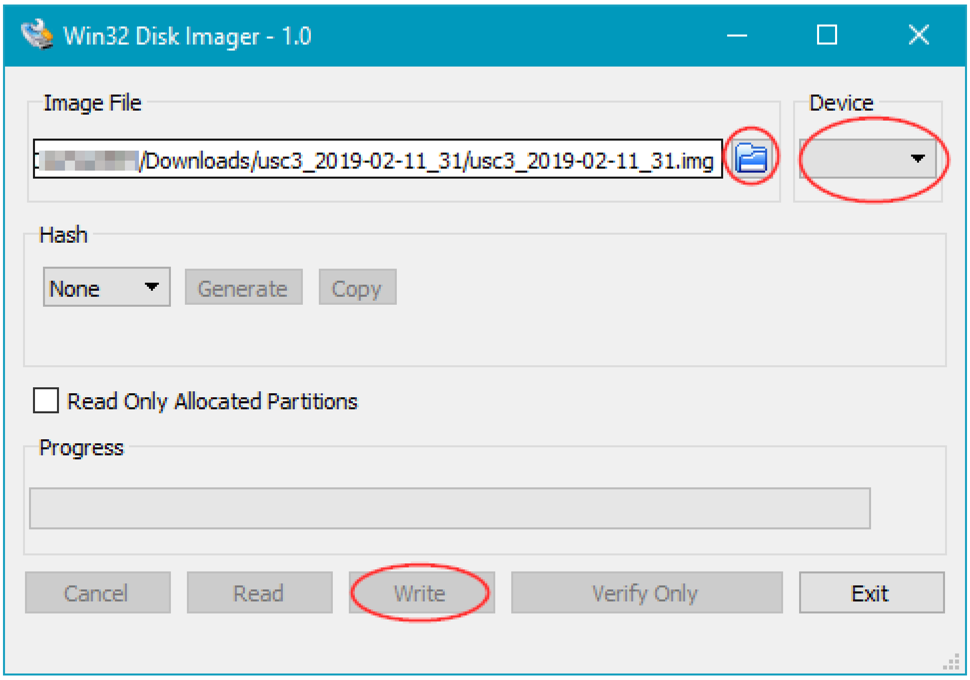


Figure 11: Example program to write image on SD-card

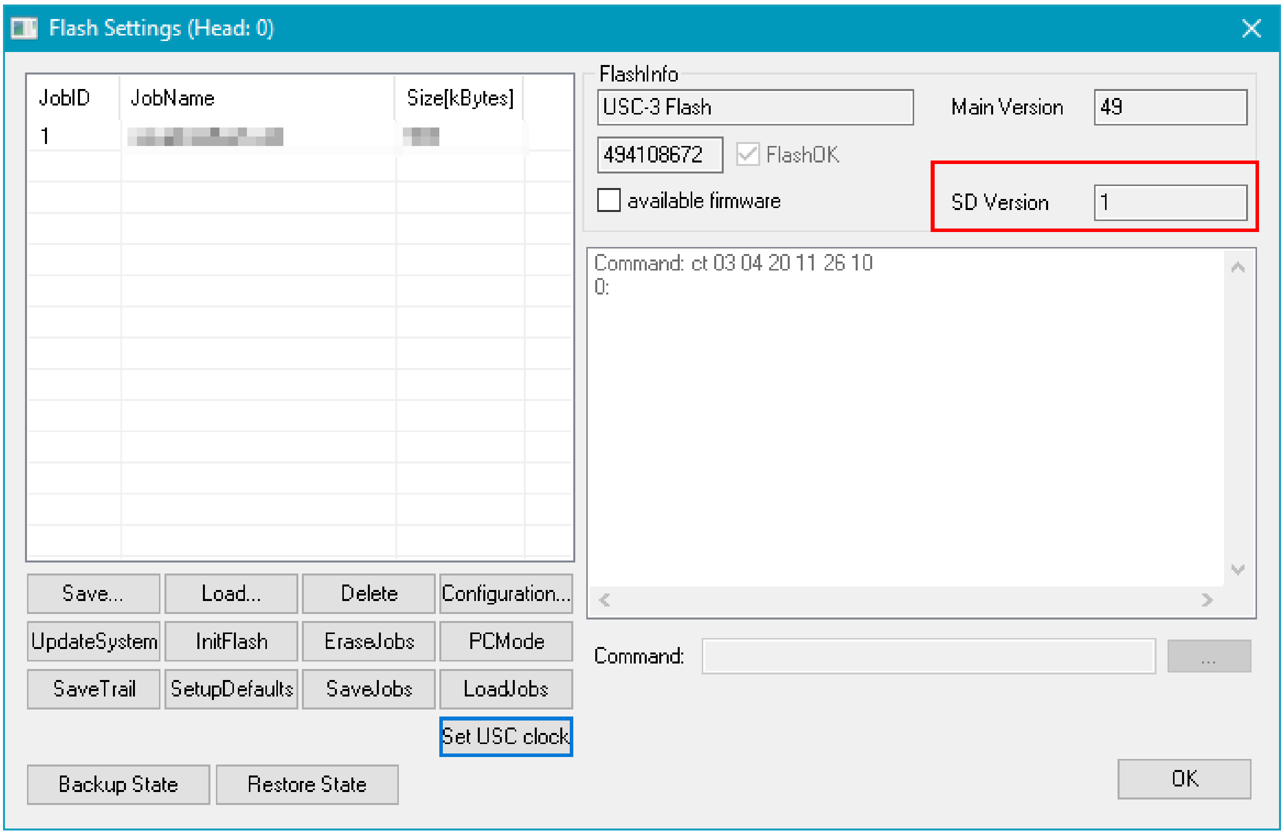


Figure 12: SD-card version

3.10 sc_usc.cfg

The ASCII file <SCAPS>\system\sc_usc.cfg consists of the following entries.

Go to the Windows task manager and make sure the process sc_usc_server.exe is not running before you edit the file.

Changes only take effect after re-powering the USC card.

Entry	Value	Description
InvertOptoOut	**0** | 1	invert the Opto_Outs (only for USC-1)
VisibleMode	**0** | 1	start sc_usc_server.exe in visible mode
ServerRestart	0 | 1	deprecated, will be deleted using the latest sc_usc_server.exe
ForceUSB11	**0** | 1	If set to 1 the sc_usc_server.exe is establishing a USB 1.1 connection to the USC card. Not available for USC-3.
Connect0	IP-Address	IP-Address of the USC-2/3
Connect1	IP-Address	-
Connect2	IP-Address	-
Connect3	IP-Address	-
Connect4	IP-Address	-
Connect5	IP-Address	-
Connect6	IP-Address	-
Connect7	IP-Address	-
Connect8	IP-Address	-
Connect9	IP-Address	-
Connect10	IP-Address	-
Connect11	IP-Address	-
Connect12	IP-Address	-
Connect13	IP-Address	-
Connect14	IP-Address	-
Connect15	IP-Address	-
MultiCard	**0** | 1	set the value to one when using more than one USC-2/3
DontOverrideCfgAndId	**0** | 1	The files sc_usc.cfg and sc_usc_card_ids.txt are being write protected if it is set to 1
FlashTrailDir		Default: "%SCAPS_SAM%\intermed" Defines the location where the trail file sc_flash_trail.zip will be saved.
FlashLoadSaveTrail	**0** | 1 | 2	0: disable 1: enable trailing for UNF uploads 2: enables trailing for UNF uploads without deleting temp files
MaxNumTrailEntries	**default: 128**	defines maximal number of subfolders in sc_flash_trail.zip file in FlashTrailDir

Table 3: sc_usc.cfg

3.11 sc_usc_card_ids.txt

The file sc_usc_card_ids.txt (formerly sc_usc1_card_ids.txt) consists of 16 entries, which define the order to enumerate the cards. The first entry defines Head0 and so on.

This file will be created dynamically the first time more than one card is connected.
Head0 must exist!
Each ID must exist only once!
Unused entries are set to zero.

3.12 Command Line for USC Server

command line parameter	description
/v	Sc_setup is started in visible mode.
/updatesystem=<HeadId>	A firmware update is done silently if the current firmware is not up to date. <HeadId> is the parameter of the head you want to update. For more than one USC card connected, the ID corresponds to the position of the card in the USC Server list. The value -1 is also possible for <HeadId>. In this case, the USC Server looks at every position (of the 16 available positions) one after the other. If a card is connected, the update is initiated. If no card is found with a timeout of 5s, the next position is controlled. This means, the update process will start after 75s if only the last position is connected with a card. You can start (a new instance of) the USC Server using this command line parameter. If another instance is already present, this new instance will be closed after updating.
/initflash=<HeadId>	Initializes the complete flash memory in silent mode. All jobs currently stored on the flash will be deleted and default settings will be restored. <HeadId> is the parameter of the head you want to initialize. For more than one USC card connected, the ID corresponds to the position of the card in the USC Server list. The value -1 is also possible for <HeadId>.

Table 4: Available command line parameters for sc_usc_server

3.13 sc_usc_server_log

While running, the USC server is writing a protocol on connection status, firmware update availability and IP and MAC address, as well as several license-depending features, such as status of Flash Mode, Job IO Selection mode, TELNET Mode, FTP server, HTML server and Stepper Mode.

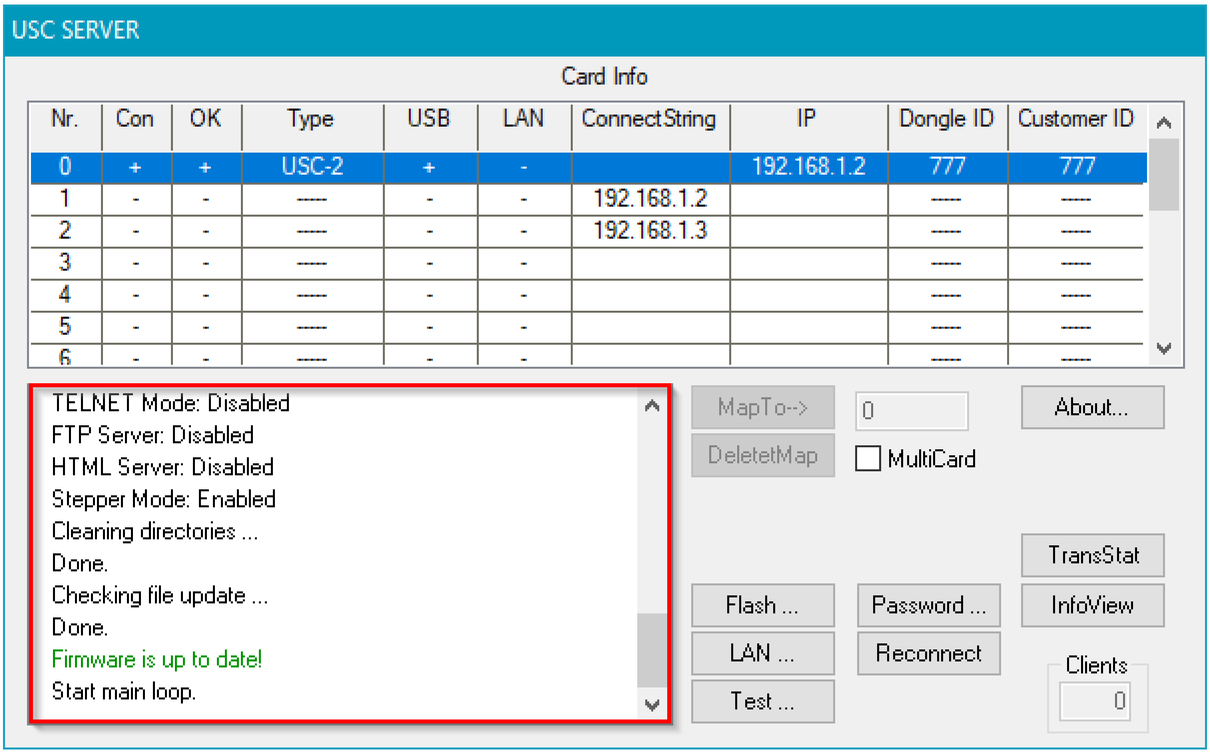


Table 5:USC server

After closing the server, the protocol is saved in the sc_usc_server_log.txt file. This file can be helpful for troubleshooting.

NOTE *For USC-3 cards, FTP is always enabled, although it is possible that this is not displayed in the server log.*

4 sc_setup.exe, Hardware Settings

The *sc_setup* provides all functionalities to setup the hardware and software. It is located in the folder *<SCAPS>\tools*.

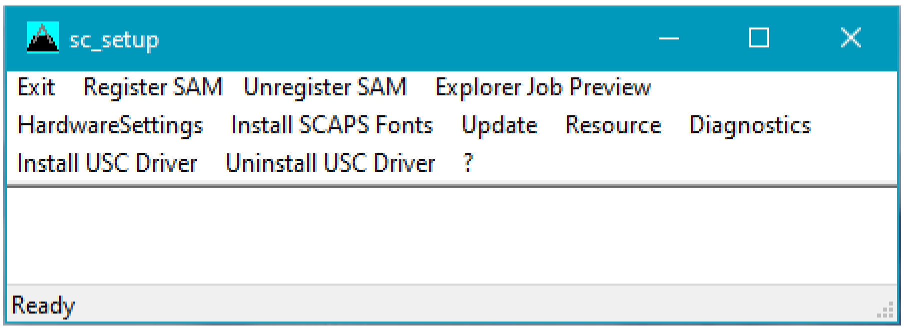


Figure 13: sc_setup.exe dialog

Menu Item	Description
Exit	Exit the software

Menu Item	Description
Register SAM	Register the SCAPS DLLs
Unregister SAM	Unregister the SCAPS DLLs
Explorer Job Preview	Register or unregister the preview of job content by thumbnail for any folder view different than "details"
HardwareSettings	Open the "HardwareSettings" dialog
Install SCAPS Fonts	Install the SCAPS laser fonts
Update	Update from previous versions
Resource	Edit the language resources
Diagnostics	Display information about the plugged in dongle, software version and dll versions
Install USC Driver	Install the USC driver software
Uninstall USC Driver	Uninstall the USC driver software
?	Open the "About" dialog

Table 6: sc_setup.exe menu

4.1 Hardware Settings

All settings which are described in the following chapters, will be stored to the settings file. These are saved in files stored at <SCAPS>\system.

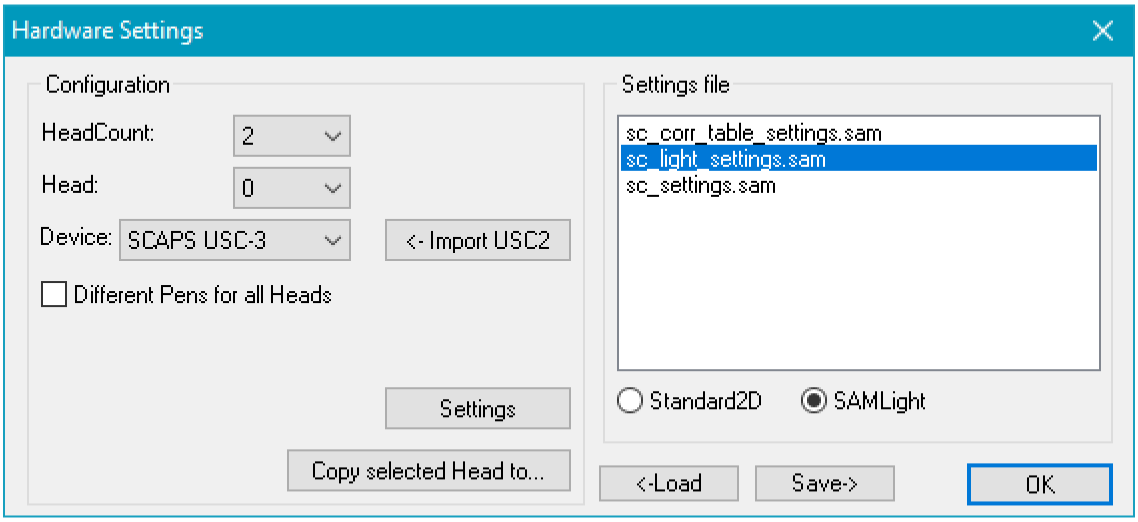


Figure 14: Hardware Settings Dialog

Configuration:

HeadCount: Choose the number of cards.

Head: Select the head which has to be setup.

Device: Select the hardware device (USC-1/-2/-3, RTC-3/-4/-5/-6, etc.) for the previous selected head.

Different Pens for all heads: When selected, the pen settings can be saved for each head individually.

Settings: Opens the Generals Settings Dialog.

Copy selected Head to...: Copy selected head settings to other head.

Settings file:

- *sc_light_settings.sam* is the default settings file for SAMLight.
- *sc_settings.sam* is the default SAM2D settings file.

Choose the settings file from the list on the right side and click *<-Load*. When using the radio buttons, *Standard2D* loads *sc_settings.sam* and *SamLight* loads *sc_light_settings.sam*.

NOTE

When using USC-3 with Head2 license, it's still one card!

4.1.1 General Settings

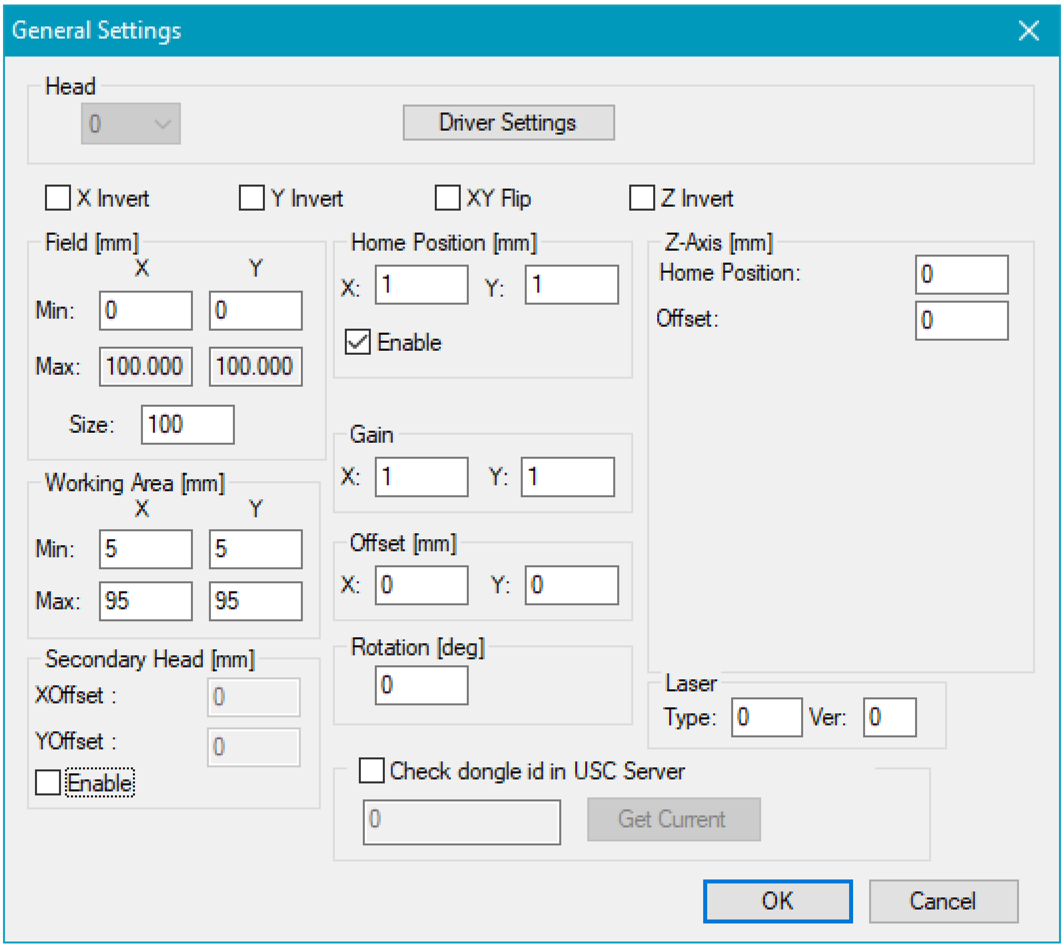


Figure 15: General Settings Dialog

Head: Choose the Head for which the driver settings should be edited. The drop down menu is only enabled if the *HeadCount* is greater than 1.

Driver Settings: Opens the Driver Settings dialog where you can adjust hardware specific settings.

Invert: Each scanner axis can be inverted separately.

Flip: The X and the Y axis can be flipped, so that the X axis gets the Y coordinates and vice versa.

Field [mm]: The size of the *field* has to be typed in mm.

> **Min:** This value can be negative, so that the field can be set up symmetrically to the origin.
>
> **Max:** This value is computed as *Min* + *Size*.
>
> **Size:** The edge length is *Size*. The field is always a square.

Working Area [mm]: The Working Area has a rectangular shape. The area is defined by *Min* and *Max*. The *Working Area* has to be within the *Field*. Consider rotation too.

Home Position [mm]: If *HomeJump* is enabled, the scanner goes back to this position after marking. Please see home jump pen for further information.

Gain: The gain values are there to slightly compensate X/Y gain errors to achieve a quadratic field.

Offset [mm]: With the *offset values* you can slightly compensate X/Y offset errors to achieve the theoretical midpoint of the scanner field. Global offset errors which have the same deviation in X and Y direction should be corrected by changing the *Field* parameters.

Rotation [deg]: The scanner output will be rotated counter clockwise by this angle.

Laser: Type in *Type* and *Ver* as advised by SCAPS.

Z-Axis [mm]: This option is only for Optic3D.

> **Z Home Position [mm]:** If *HomeJump* is enabled, the scanner goes back to this position after marking.
>
> **Z Offset [mm]:** Adds an offset to all Z values.
> USC cards: XY coordinates are not compensated, like Pen Defocus.
> RTC cards: XY coordinates are compensated, unlike Pen Defocus.

4.1.2 Advanced Settings

4.1.2.1 Correction Settings

By clicking on the Correction Button in *Settings → System → Optic → Advanced → Correction → Settings* the correction dialog of the USC-2/-3 card is being opened:

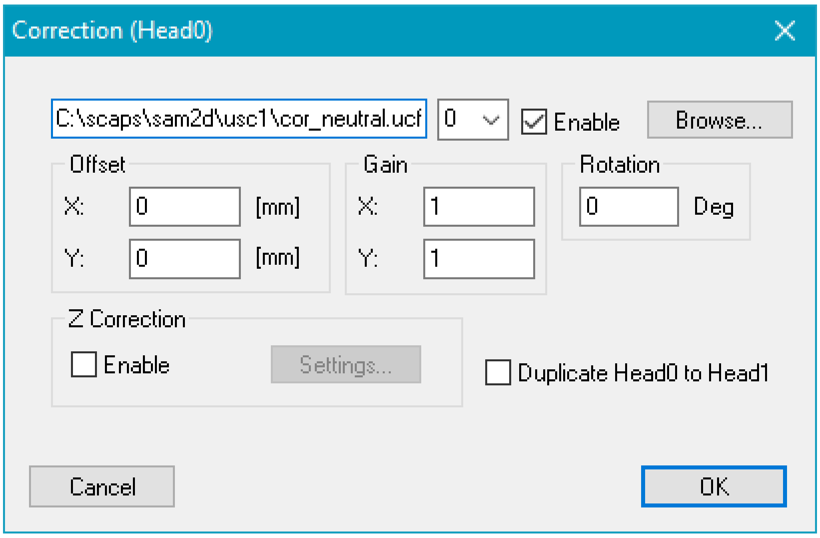


Figure 16: Correction Settings for USC-2/-3 Card

Offset: Define an offset of the marking result that is calculated on the scanner card and thus a "Galvo Coordinates out of range" error will not be shown if the translated entity extends to the outside of the working area.

Gain: Define a Gain for the marking result that is calculated on the scanner card and thus a "Galvo Coordinates out of range" error will not be shown if the scaled entity extends to the outside of the working area.

Rotation: Enter a rotation angle here. Rotation center will be the middle point of the working area. Please remind that this rotation is calculated in the scanner card just one step before marking output and thus a "Galvo Coordinates out of range" error will not be shown if the rotated entity extends to the outside of the working area.

Z-Correction: If enabled and clicking on Settings... the Z Correction Table Dialog will be opened.

Duplicate Head0 to Head1: If enabled, the signals of Head0 are duplicated and send to Head1. Thus, the signals on Head0 and Head1 are identical.

The Offset, Gain and Rotation parameters of this dialog could be interesting if you have several Jobs on the Flash of the USC-2/-3 card and want to change these parameters for all these jobs. Since the calculation is done on the card, the parameters would be applied to all jobs already existing on the Flash.
If you use the parameters from the Optic dialog, you would have to reload each job in SAMLight and restore it to the Flash because the effect of these parameters is calculated by SAMLight and is stored in the bit information of the job.
So, if you want to change Offset, Gain and Rotation, it is recommended to use the parameters in Optic - Advanced - Correction instead of Optic.

4.1.2.2 Analog In

Any input voltage will be live updated and can influence the running marking procedure.

This feature is only supported for USC-2/-3 cards.

Z offset is only available for USC-3.

This dialog offers the possibility to set an offset of scanner position in x, y and z direction by putting a voltage on analog input.

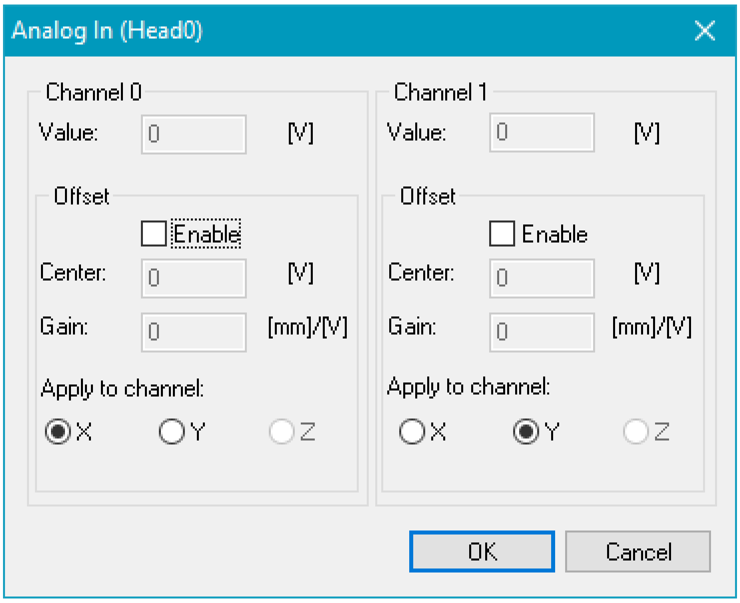


Figure 17: Analog-In Dialog

Value: Value shows the voltage value been placed on pin Analog In.

Center: In this checkbox you can define a voltage value with which the Offset is 0 mm.

Gain: In this checkbox you define the width of offset per Volt. Positive and negative values are valid. Depending on the sign, the direction of the offset is adapted.

Example: set Center = 5 V, Gain = 2 mm/V for Offset X, when you place 6 V on pin AI 0, the scanner gets an offset of 2 mm in x direction. For Center = 5 V, Gain = - 2 mm/V, the scanner gets an offset of - 2 mm in x direction when you place 6 V on pin AI 0.

4.1.3 Pin Assignment

Port	USC-1 / USC-2 / USC-3	RTC3	RTC4
Analog A	37-pin connector DAC_A → pin 10	9-pin SUB-D laser connector (VB3) Analog OUT1 → pin 4, only available if jumper is set accordingly, see RTC3 installation manual	9-pin SUB-D laser connector (VB3) Analog OUT1 → pin 4, only available if jumper is set accordingly, see RTC4 installation manual
Analog B	37-pin connector DAC_B → pin 29	9-pin SUB-D laser connector (VB3) Analog OUT2 → pin 2, only available if jumper is set accordingly, see RTC3 installation manual	9-pin SUB-D laser connector (VB3) Analog OUT2 → pin 2, only available if jumper is set accordingly, see RTC4 installation manual
8 Bit Output	37-pin connector LP 0 → pin 19 LP 1 → pin 37 LP 2 → pin 18 LP 3 → pin 36 LP 4 → pin 17 LP 5 → pin 35 LP 6 → pin 16 LP 7 → pin 34	LASER extension connector L0 → pin 1 L1 → pin 3 L2 → pin 5 L3 → pin 7 L4 → pin 9 L5 → pin 11 L6 → pin 13 L7 → pin 15 / pin 17 only if jumper is set accordingly, see RTC3 installation manual	LASER extension connector L0 → pin 1 L1 → pin 3 L2 → pin 5 L3 → pin 7 L4 → pin 9 L5 → pin 11 L6 → pin 13 L7 → pin 15 / pin 17 only if jumper is set accordingly, see RTC4 installation manual
Q-Switch / laser frequency	37-pin connector LaserA → pin 13	9-pin SUB-D laser connector (VB3) Laser1 → pin 1	9-pin SUB-D laser connector (VB3) Laser1 → pin 1
LaserGate	37-pin connector Laser_Gate → pin 31	9-pin SUB-D laser connector (VB3) LaserOn → pin 2	9-pin SUB-D laser connector (VB3) LaserOn → pin 2
IO	37-pin connector OPTO_IN 0 → pin 1 OPTO_IN 1 → pin 20 OPTO_IN 2 → pin 2 OPTO_IN 3 → pin 21 OPTO_IN 4 → pin 8 OPTO_IN 5 → pin 9 OPTO_OUT 0 → pin 3 OPTO_OUT 1 → pin 22 OPTO_OUT 2 → pin 4 OPTO_OUT 3 → pin 23 OPTO_OUT 4 → pin 27 OPTO_OUT 5 → pin 28	RTC3 I/O Extension Board, 68-pin SCSI Connector DIGITAL_IN0 ... DIGITAL_IN11, oDIGITAL_IN12+- ... oDIGITAL_IN15+- DIGITAL_OUT0 ... DIGITAL_OUT15 For proper I/O handling, use the SCAPS Marking Active signal instead of the RTC busyout signal.	40-pin extension DIGITAL IN0 ... DIGITAL IN15 DIGITAL OUT0 ... DIGITAL OUT15 For proper I/O handling, use the SCAPS Marking Active signal instead of the RTC busyout signal.
Ext. Trigger	37-pin connector	9-pin SUB-D laser connector (VB3)	9-pin SUB-D laser connector (VB3)
Start	OPTO_IN 0 → pin 1	/START → pin 8	/START → pin 8
Stop	OPTO_IN 1 → pin 20	/STOP → pin 9	/STOP → pin 9

Port	RTC5 / RTC6	SCANalone
Analog A	15-pin SUB-D laser connector Analog OUT1 → pin 8	9-pin SUB-D laser connector Analog OUT1 → pin 4, only available if jumper is set accordingly, see RTC SCANalone installation manual
Analog B	15-pin SUB-D laser connector Analog OUT2 → pin 15	9-pin SUB-D laser connector Analog OUT2 → pin 2, only available if jumper is set accordingly, see RTC SCANalone installation manual
8 Bit Output	EXTENSION 2 connector L0 → pin 1 L1 → pin 3 L2 → pin 5 L3 → pin 7 L4 → pin 9 L5 → pin 11 L6 → pin 13 L7 → pin 15 / pin 17 only if jumper is set accordingly, see RTC5 installation manual	LASER extension SUB-D connector L0 → pin 1 L1 → pin 2 L2 → pin 3 L3 → pin 4 L4 → pin 5 L5 → pin 6 L6 → pin 7 L7 → pin 8 / pin 9 only if jumper is set accordingly, see RTC SCANalone installation manual
Q-Switch / laser frequency	15-pin SUB-D laser connector Laser1 → pin 1	
Laser_Gate	15-pin SUB-D laser connector Laser_On → pin 2	
IO	40-pin extension DIGITAL IN0 ... DIGITAL IN15 DIGITAL OUT0 ... DIGITAL OUT15 For proper I/O handling, use the SCAPS Marking Active signal instead of the RTC busyout signal.	37-pin extension1-SUB-D connector DIGITAL IN0 ... DIGITAL IN15 DIGITAL OUT0 ... DIGITAL OUT15 For proper I/O handling, use the SCAPS Marking Active signal instead of the RTC busyout signal.
Ext. Trigger	15-pin SUB-D laser connector	9-pin SUB-D laser connector
Start	/START → pin 3	/START → pin 8
Stop	/STOP → pin 11	/STOP → pin 9
Marking in progress		

4.1.4 USC-1 Settings

Card Settings for USC-1 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Optic → Advanced...:*

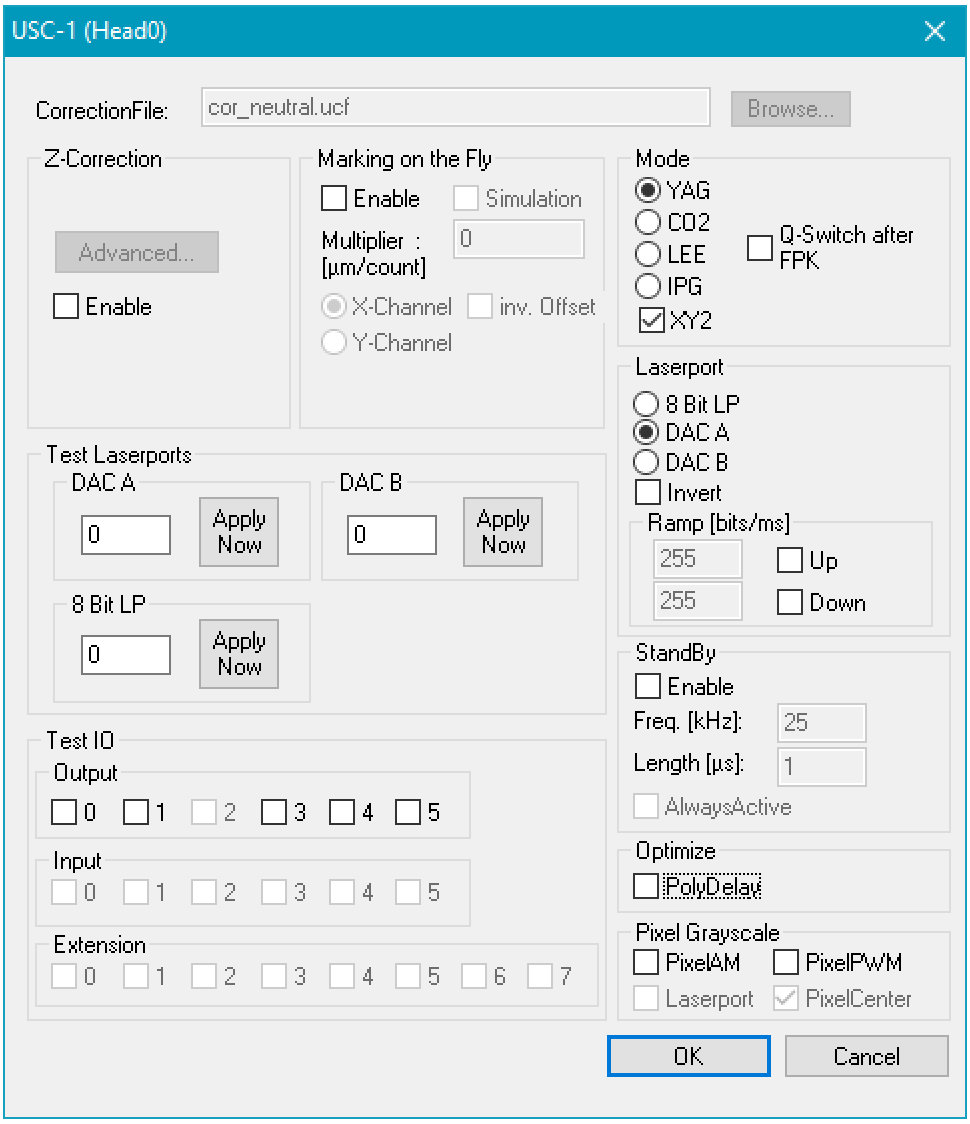


Figure 18: Card Settings for USC-1 Card

CorrectionFile: Click on the Browse button to select the correct correction file for this card.

MarkingOnTheFly: Requires the Marking On The Fly (MOTF) SAMLight option. MOTF related parameters are described in chapter Card Specific: USC-1.

Test Laserports:

DAC A/B: Here you can test the Digital Output A/B. This can be observed with a DB-37 Diagnostic Board attached to the USC-1 card. The minimum value for the digital output is 0, the maximum is 255. Value is also set after reboot (if stored) or with the start of SAMLight.

8 Bit LP: Example: If 5 is entered and *Apply Now* is pressed, the first and the third bit of the digital port are *high*.

Test IO: The IO-Port is a 6-bit port. For the USC-1 scanner card the first output bit is predefined for marking in progress. The first input bit is predefined for external start, the second for stop.

Mode: Here you can choose the type of the laser (YAG, CO2, LEE, IPG). Standard timing diagrams of laser signals are given here.

XY2: Defines that the scanner driver signals are standard XY2-100

ParaDel. [ms]: This field is only available for laser modes 588-1 and 588-2. The value causes the software to wait after a pulse width change had been sent to the interface. The unit of the entered value is [ms].

Sync mode: Only visible for some Laser Types. Restart the Laser_A signal with a defined delay before the opening of the Laser_Gate signal. This option enables synchronized Laser_A signal with rising edge of the Laser_Gate.

WARNING *Make sure that the 'XY2' check box is checked if not mentioned differently.*

Q-Switch after FPK: Sets Q-Switch after the first pulse killer.

Laserport: Defines the port that sends the power signal for the laser (8 Bit LP, DAC A, DAC B).

Invert: Here the laser power value can be inverted. Note: In case of 8 Bit LP this checkbox does not invert the bits.

Ramp: Here you can define a velocity to increase or decrease the power of the laser port.

StandBy:

Enable: Globally enables standby mode.

Freq.: Q-Switch frequency of the laser pulses.

Length: Q-Switch length in µs.

AlwaysActive: Standby mode is also used when the laser signal is on.

The settings are done after leaving the global dialog Settings. Standby Settings for pens overwrite these settings if enabled for a pen as soon as this pen is used.

Optimize:

PolyDelay: If this is selected the length of the polygon delay gets varied depending on the angle between two successive vectors.

Pixel Grayscale:

PixelAM: Enables Amplitude Modulation.

PixelPWM: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

Laserport: If checked the selected Laserport gets used for the output of PixelAM, else port DAC B is taken.

4.1.5 USC-2 Settings

Card Settings for USC-2 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Optic → Advanced...:*

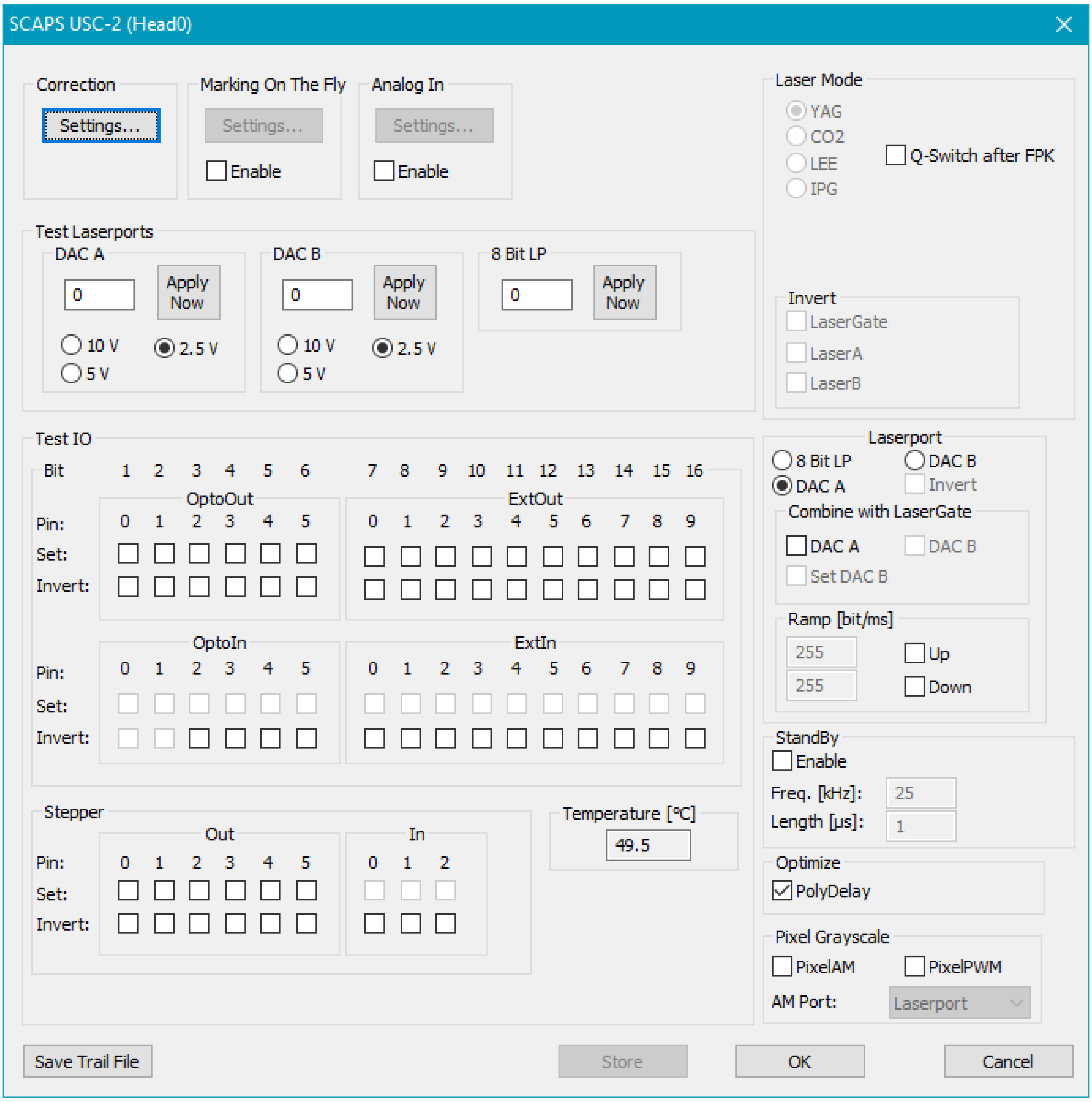


Figure 19: Card Settings for USC-2 Card

Correction:

Settings...: Click this button to open the Correction File Dialog for the USC-2 card. See chapter correction settings.

Marking On The Fly (MOTF): Requires the Marking On The Fly (MOTF) SAMLight option.

Settings...: See chapter Card Specific: USC-2.

Enable: Check this to enable MOTF.

Analog In:

Settings...: There an Analog Input signal can be used that affects the scanner position. See chapter Analog In.

Enable: Check this to enable *Analog In*.

Test Laserports:

DAC A/B: Here you can test the Digital Output A/B. This can be observed with a DB-37 Diagnostic Board attached to the USC-2 card. The minimum value for the digital output is 0, the maximum is 4095. Value is also set after reboot (if stored) or with the start of SAMLight. For 10V, the full 12 bit resolution is used, for 5V, the values are shifted 1 bit, for 2.5V 2 bit to the right (down).

8 Bit LP: Here you can test the 8 bit laser port by entering a value from 0 to 255. This can be observed with a DB-37 Diagnostic Board connected to the USC-2 card.

Test IO:

Opto_Out/In: Here you can test if the bits of Opto_Out and Opto_In can be set correctly.

ExtOut/In: Here you can test the additional IO bits. These IOs can be used for Job IO Selection mode or to control external devices.

Stepper:

Out/In: If connected test the I/Os of a stepper motor.

Mode: Here you can choose the type of the laser (YAG, CO2, LEE, IPG). Standard timing diagrams of laser signals are given here.

Q-Switch after FPK: Sets the Q-Switch after the first pulse killer.

LaserB as FPK: Only visible for CO2 Laser. Enables the First Pulse setting at pen settings main.

Sync mode: Only visible for some Laser Types. Restart the Laser_A signal with a defined delay before the opening of the Laser_Gate signal. This option enables synchronized Laser_A signal with rising edge of the Laser_Gate.

ParaDel. [ms]: This field is only available for laser modes 588-1 and 588-2. The value causes the software to wait after a pulse width change had been sent to the interface. The unit of the entered value is [ms].

Invert: Laser Gate, Laser A, Laser B: Here the laser signals can be set to be *active low* or *active high*.

NOTE

For security reasons the USC-2 Laser signals can only be changed with <SCAPS>\tools\sc_setup.exe → HardwareSettings, "choose settings file" → Settings → Driver Settings, not in SAMLight.

Laserport: Defines the port that sends the power signal for the laser (8 Bit LP, DAC A, DAC B).

Invert: Shows the status of the Laserport. Just for "LEE" Laser Mode the 8 Bit LP is inverted.

Combine with Laser_Gate: If DAC A or DAC B is chosen under Laserport it is possible to turn off the laser power signal with the Laser_Gate. If Laser_Gate is going down DAC A or DAC B will also go down to 0. If the checkbox *Set DAC B* is activated the DAC A will not go to 0 when Laser_Gate is going down but it will be set to the value that is defined under *Test Laser ports → DAC B*.

Ramp: Defines the velocity of the increasing or decreasing of the power of the laser port.

StandBy:

Enable: Globally enables standby mode.

Freq.: Q-Switch frequency of the laser pulses.

Length: Q-Switch length in µs.

AlwaysActive: Standby mode is also used when the laser signal is on.

Optimize:

PolyDelay: If this is selected the length of the polygon delay gets varied depending on the angle between two successive vectors.

Pixel Grayscale:

PixelAM: Enables Amplitude Modulation.

PixelPWM: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

AM Port: Select the output of the bitmap grayscale amplitude modulation. *PixelAM* must be enabled.

- DAC B: The 12 bit analog port DAC_B on the *Laser, I/O - 37 pin* connector.
- Laserport: The laser port selected from above.
- Digital Out: From Digi_Out_0 (LSB) to Digi_Out_9 (MSB) on the *Extension - 40 pin* connector. This only works in Gray Advanced mode and does not work in flash (standalone mode), yet.

Save Trail File: Saves the USC-2 trail file to *<SCAPS>\intermed*. The file name corresponds to *sc_usc2_trail_head_x.txt*, where x stands for the head number.

Store: Stores the settings to the USC-2 EPCS (this is necessary for stand-alone operation). Make sure that the settings fit to the laser and other machinery. The settings will be loaded during powering on the card.

4.1.6 USC-3 Settings

Card Settings for USC-3 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Optic → Advanced...:*

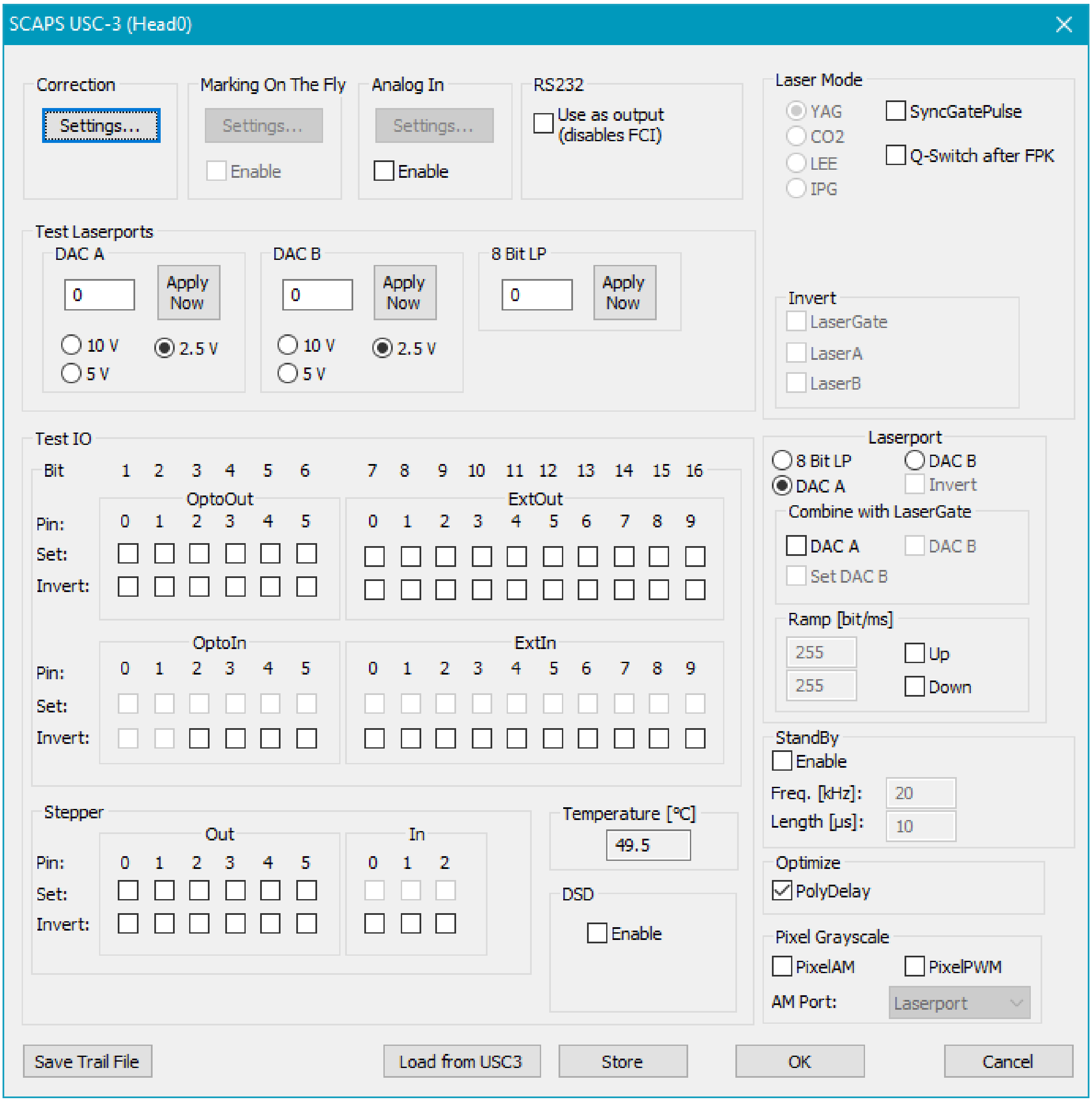


Figure 20: Card Settings for USC-3 Card

Correction:

Settings...: Click this button to open the Correction File Dialog for the USC-3 card. See chapter USC-3 correction settings.

Marking On The Fly (MOTF): Requires the Marking On The Fly (MOTF) SAMLight option.

Settings...: See chapter Card Specific: USC-3.

Enable: Check this to enable MOTF.

Analog In:

Settings...: There an Analog Input signal can be used that affects the scanner position. See chapter Analog In.

Enable: Check this to enable *Analog In*.

RS232:

Use as Output (disables FCI): Activates RS232 as Output for controlling laser and motion controller, therefore Flash Control Interface (FCI) to send Flash commands by RS232 is disabled.

Test Laserports:

DAC A/B: Here you can test the Digital Output A/B. This can be observed with a DB-37 Diagnostic Board attached to the USC-3 card. The minimum value for the digital output is 0, the maximum is 4095. Value is also set after reboot (if stored) or with the start of SAMLight. For 10V, the full 12 bit resolution is used, for 5V, the values are shifted 1 bit, for 2.5V 2 bit to the right (down).

8 Bit LP: Here you can test the 8 bit laserport by entering a value from 0 to 255. This can be observed with a DB-37 Diagnostic Board connected to the USC-3 card.

Test IO:

OptoOut/In: Here you can test if the bits of Opto_Out and Opto_In can be set correctly.

ExtOut/In: Here you can test the additional I/O bits. These I/Os can be used for Job IO Selection mode or to control external devices.

Stepper:

Out/In: If connected test the I/Os of a stepper motor.

Mode: Here you can choose the type of the laser (YAG, CO2, LEE, IPG). Standard timing diagrams of laser signals are given here.

Q-Switch after FPK: Sets the Q-Switch after the first pulse killer.

LaserB as FPK: Only visible for CO2 Laser. Enables the First Pulse setting at pen settings main.

IPG sync mode: Only visible for IPG Laser. Restart the Laser_A signal with a defined delay before the opening of the Laser_Gate signal. This option enables synchronized Laser_A signal with rising edge of the Laser_Gate.

ParaDel. [ms]: This field is only available for laser modes 588-1 and 588-2. The value causes the software to wait after a pulse width change had been sent to the interface. The unit of the entered value is [ms].

Invert: Laser Gate, Laser A, Laser B: Here the laser signals can be set to be *active low* or *active high*.

For security reasons, the USC-3 Laser signals can only be changed with <SCAPS>\tools\sc_setup.exe → HardwareSettings, "choose settings file" → Settings → Driver Settings, not in SAMLight.

Laserport: Defines the port that sends the power signal for the laser (8 Bit LP, DAC A, DAC B).

Invert: Shows the status of the Laserport. Just for "LEE" Laser Mode the 8 Bit LP is inverted.

Combine with Laser_Gate: If DAC A or DAC B is chosen under Laserport it is possible to turn off the laser power signal with the Laser_Gate. If Laser_Gate is going down DAC A or DAC B will also go down to 0. If the checkbox *Set DAC B* is activated the DAC A will not go to 0 when Laser_Gate is going down but it will be set to the value that is defined under *Test Laser ports → DAC B*.

Ramp: Defines the velocity of the increasing or decreasing of the power of the laser port.

StandBy:

Enable: Globally enables standby mode.

Freq.: Q-Switch frequency of the laser pulses.

Length: Q-Switch length in µs.

AlwaysActive: Standby mode is also used when the laser signal is on.

Optimize:

PolyDelay: If this is selected the length of the polygon delay gets varied depending on the angle between two successive vectors.

Pixel Grayscale:

PixelAM: Enables Amplitude Modulation.

PixelPWM: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

AM Port: Select the output of the bitmap grayscale amplitude modulation. *PixelAM* must be enabled.

- DAC B: The 12 bit analog port DAC_B on the *Laser, I/O - 37 pin* connector.
- Laserport: The laser port selected from above.
- Digital Out: From Digi_Out_0 (LSB) to Digi_Out_9 (MSB) on the *Extension - 40 pin* connector. This only works in Gray Advanced mode and does not work in flash (standalone mode), yet.

Save Trail File: Saves the USC-3 trail file to *<SCAPS>\intermed*. The file name corresponds to *sc_usc3_trail_head_x.txt*, where x stands for the head number.

Store: Stores the settings to the USC-3 (this is necessary for stand-alone operation). Make sure that the settings fit to the laser and other machinery. The settings will be loaded during powering on the card.

Load from USC3: Load the settings from USC-3 into SAMLight. Only a subset containing the following information is loaded:

- All settings made in the *sc_setup.exe* at *Hardware Settings* → *Settings*
- *Pen Settings* without *Ramping* and the function *Mark Lines as Dots*

4.1.7 RTC-3 Settings

Card Settings for RTC-3 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Card → Advanced... → Driver Settings*

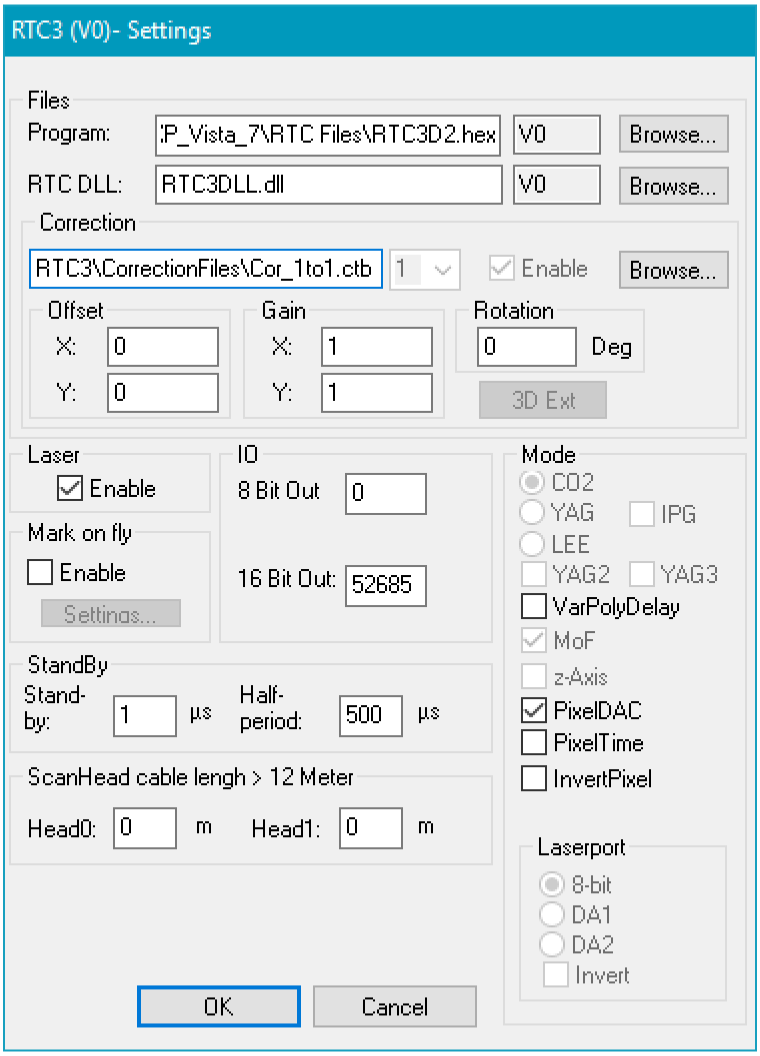


Figure 21: Card Settings for RTC3 Card

Files:

Program: Specifies the program file. This file is delivered together with the scanner card. The standard extension is *.hex. Click on the *Browse* button to make a search on the files location or type in the file name into the edit window.

RTC DLL: Specifies the *.dll file. This file is delivered together with the scanner controller card.

Using a 64-bit operating system you should nevertheless load a 32-bit RTC DLL, not the 64-bit one.

Correction: Specifies the location of the correction file. This file is delivered together with the scanner card. The standard extension is *.ctb. Click on *Browse* button to make a search on the files location or type in the file name into the edit window. It is possible to load two different correction files. This is mainly used in connection with the secondary head feature of the RTC3.

Offset, Gain, Rotation: Allow a global adjustment for each correction file. These features are mainly used to adjust the fields of both heads when a secondary head is used. See also: Chapter Optic Settings Dialog.

3D Ext: Opens a dialog, where a Z-Table can be defined. For detailed information have a look at the RTC3 manual. See chapter Optic3D for RTC cards.

Laser: Globally enables or disables laser output.

Mark on Fly: Requires the Marking On The Fly RTC option. MOTF related parameters are described in chapter Card Specific: RTC cards.

StandBy: Globally enables standby mode.

Stand-by: Q-Switch length in µs for stand-by modus. If this is set to zero the stand-by mode is switched off.

Half-period: Half of the laser pulse period for stand-by modus.

NOTE

Settings are done after leaving the global dialog Settings. Standby Settings for pens will overwrite these settings if enabled for a pen as soon as this pen is used.

ScanHead cable length > 12 Meter: Put in the cable length of each head if one cable is longer than 12m.

IO:

Lamp/8-bit: Sets the 8 bit or one of the digital outputs of the RTC Card during start up (as selected under Laserport). The 8 bit Output corresponds to the write_8bit_port command of the RTC.

16 Bit Out: Sets the 16 bit output of the RTC Card during start up.

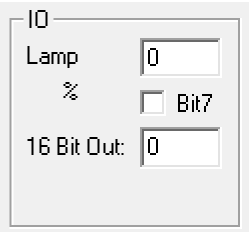


Figure 22: If *LEE Mode* is selected, the eighth bit is selectable separately

Mode: Here you can choose the type of the laser.

VarPolyDelay: If checked the length of the polygon delay gets varied depending on the angle between two successive vectors.

MoF: Shows whether the scanner card is able to do Marking On The Fly or not.

z-Axis: Indicates whether the card is able to do 3D Marking or not.

PixelDAC: Enables Amplitude Modulation.

PixelTime: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

InvertPixel: Inverts bitmap pixels.

Pixel Mode 0: See chapter Pixelmode.

Laserport: Defines the port that sends the power signal for the laser if the laser is not a CO2 Laser. For a CO2 Laser the power signal is done by modulating the Laser A signal.

Invert: Here the laser power value can be inverted. Note: In case of 8-bit this checkbox does not invert the bits.

The RTC PCI Board does not support power saving modes, that switch off power to the PCI bus. This could lead to PC freezes at the start of SAMLight. Accordingly, you must disable standby or sleep modes of the operating system and via the „SleepMode.cmd“ script, which you can find in your RTC tools folder. Please refer to your RTC manual for more information.

4.1.8 RTC-4 Settings

Card Settings for RTC-4 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Card → Advanced...*

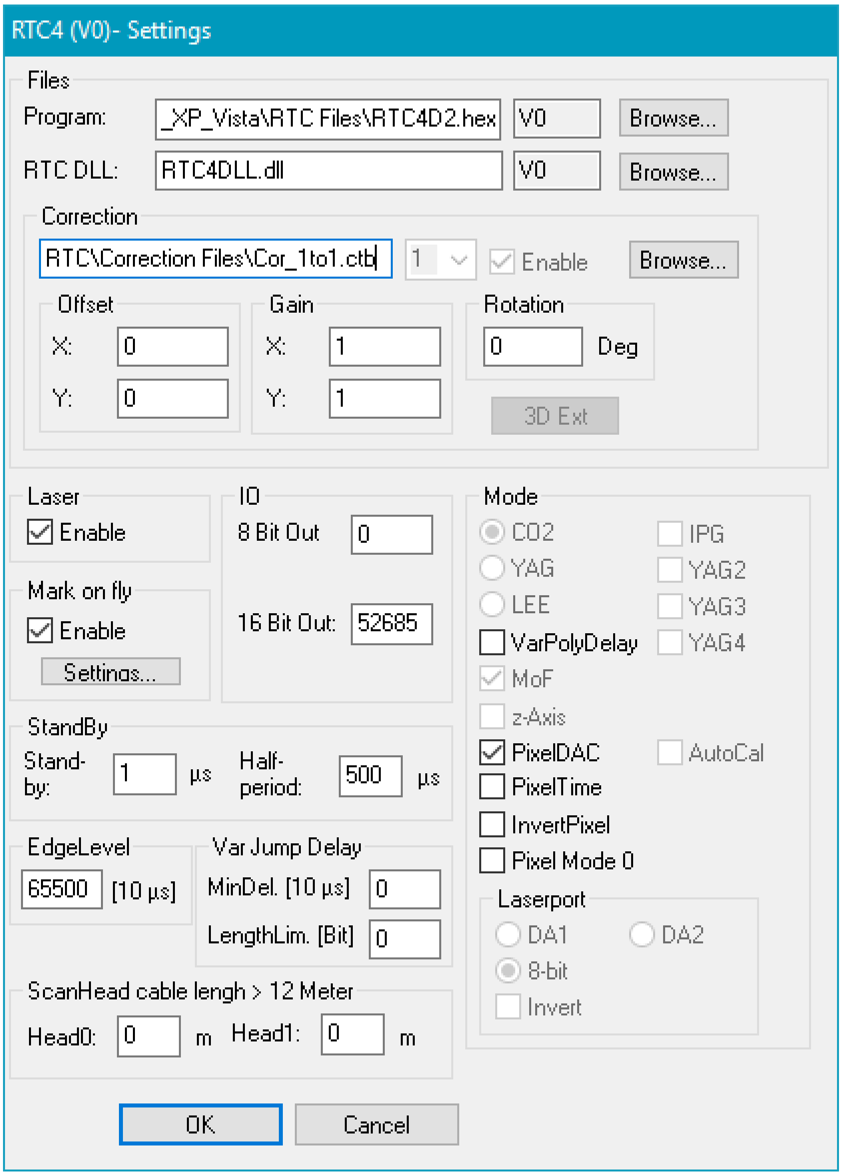


Figure 23: Card Settings for RTC4 Card

Files:

Program: Specifies the program file. This file is delivered together with the scanner card. The standard extension is *.hex. Click on the *Browse* button to make a search on the files location or type in the file name into the edit window.

RTC DLL: Specifies the *.dll file. This file is delivered together with the scanner controller card.

Using a 64-bit operating system you should nevertheless load a 32-bit RTC DLL, not the 64-bit one.

Correction: Specifies the location of the correction file. This file is delivered together with the scanner card. The standard extension is *.ctb. Click on *Browse* button to make a search on the files location or type in the file name into the edit window. It is possible to load two different correction files. This is mainly used in connection with the secondary head feature of the RTC4.

Offset, Gain, Rotation: Allow a global adjustment for each correction file. These features are mainly used to adjust the fields of both heads when a secondary head is used. See also: Chapter Optic Settings Dialog.

3D Ext: Opens a dialog, where a Z-Table can be defined. For detailed information have a look at the RTC4 manual. See chapter Optic3D for RTC cards.

Laser: Globally enables or disables laser output.

Mark on Fly: Requires the Marking On The Fly RTC option. MOTF related parameters are described in chapter Card Specific: RTC cards.

StandBy: Globally enables standby mode.

Stand-by: Q-Switch length in µs for stand-by modus. If this is set to zero the stand-by mode is switched off.

Half-period: Half of the laser pulse period for stand-by modus.

NOTE

Settings are done after leaving the global dialog Settings. Standby Settings for pens will overwrite these settings if enabled for a pen as soon as this pen is used. Also please remind that after marking has finished the settings from the Home Jump Style Pen are used.

EdgeLevel: The variable Polygon Delay gets very high if the angle of two successive vectors is close to 180° . This can lead to burn in effects. To prevent this an *EdgeLevel* value can be defined. If the Polygon Delay between two mark commands is greater than this value the RTC4 card switches off the laser after the first command and after the laser-off delay is over. Then a new polygon marking with the second vector will be started.

Var Jump Delay: Normally after a jump command a constant jump delay is inserted. But for very small jumps it is not necessary to have such a long jump delay. The jump delay can be reduced without loosing marking quality. The minimum delay is the delay for a jump of length 0.

IO:

Lamp/8-bit: Sets the 8 bit or one of the digital outputs of the RTC Card during start up (as selected under Laserport). The 8 bit Output corresponds to the write_8bit_port command of the RTC.

16 BIT Out: Sets the 16 bit output of the RTC Card during start up.

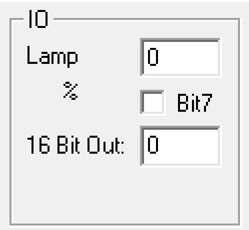


Figure 24: If *LEE Mode* is selected, the eighth bit is selectable separately

Mode: Here you can choose the type of the laser.

YAG2, YAG3 and YAG4: YAG4 corresponds to Laser Mode 4 like described in the SCANLAB RTC manual - YAG2 and YAG3 respectively.

VarPolyDelay: If checked the length of the polygon delay gets varied depending on the angle between two successive vectors.

MoF: Shows whether the scanner card is able to do Marking On The Fly or not.

z-Axis: Indicates whether the card is able to do 3D Marking or not.

PixelDAC: Enables Amplitude Modulation.

PixelTime: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

InvertPixel: Inverts bitmap pixels.

Pixel Mode 0: See chapter Pixelmode.

AutoCal: Displays the ScAutoCalib button within the functionality object toolbar to generate AutoCalib control objects in the entity list, see section AutoCal Control Object.

Laserport: Defines the port that sends the power signal for the laser if the laser is not a CO2 Laser. For a CO2 Laser the power signal is done by modulating the Laser A signal.

Invert: Here the laser power value can be inverted. Note: In case of 8-bit this checkbox does not invert the bits.

WARNING

The RTC PCI Board does not support power saving modes, that switch off power to the PCI bus. This could lead to PC freezes at the start of SAMLight. Accordingly, you must disable standby or sleep modes of the operating system and via the „SleepMode.cmd" script, which you can find in your RTC tools folder. Please refer to your RTC manual for more information.

4.1.9 RTC-5 Settings

Card Settings for RTC-5 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Card → Advanced...*

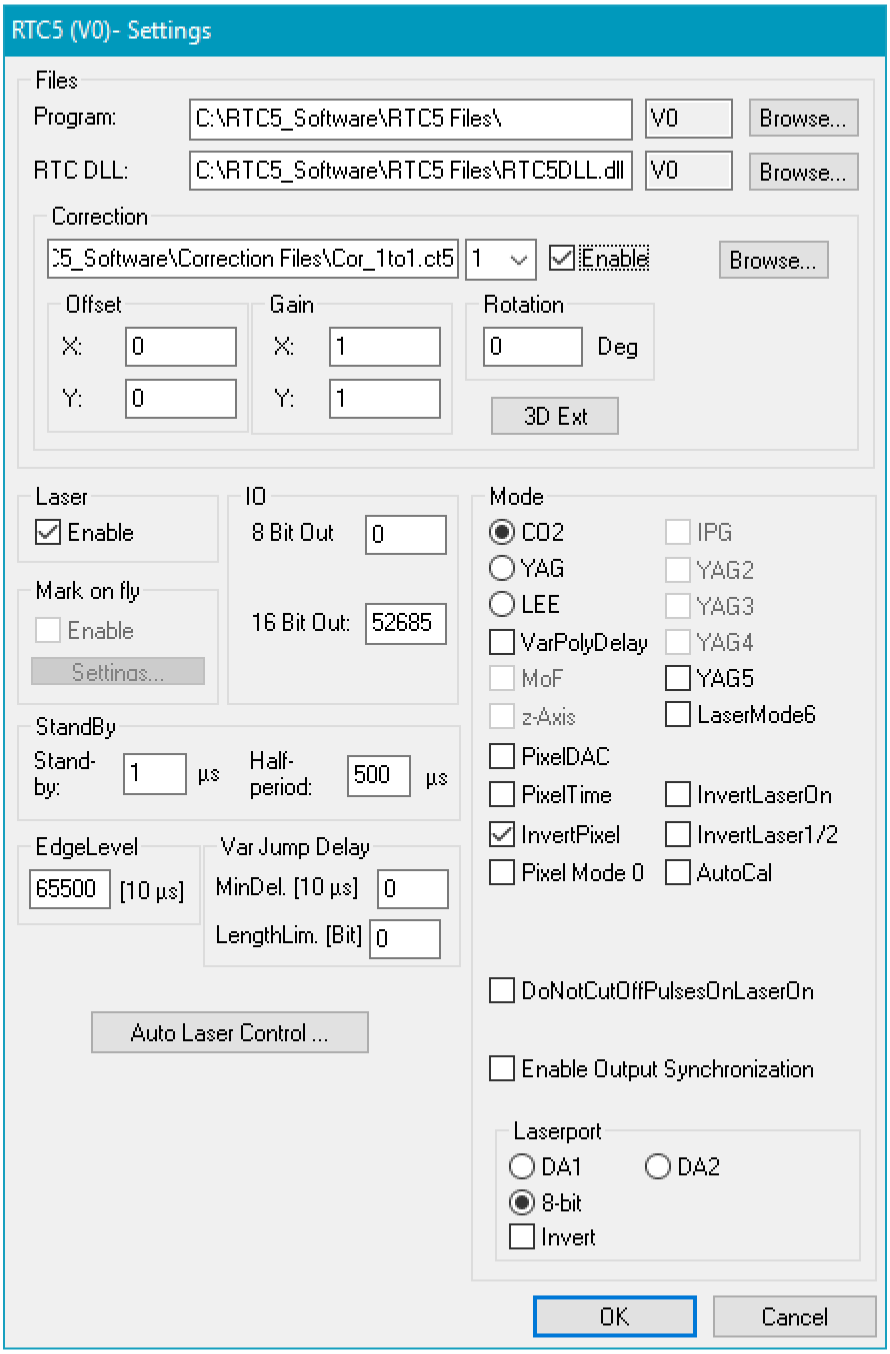


Figure 25: Card Settings for RTC5 Card

Files:

Program: Specifies the path to the *RTC5 Files* folder, delivered together with the scanner controller card driver.

RTC DLL: Specifies the path to the *RTC5DLL.dll* file. This file is in the *RTC5 Files* folder, delivered together with the scanner controller card driver.

Using a 64 bit operating system you should nevertheless load a 32 bit RTC5DLL.dll, not the 64 bit RTC5DLLx64.dll.

Correction: Specifies the location of the correction file. This file is delivered together with the scanner card. The extension is *.ct5. Click on *Browse* button to make a search on the files location or type in the file name into the edit window. It is possible to load two different correction files. This is mainly used in connection with the secondary head feature of the RTC5.

Offset, Gain, Rotation: Allow a global adjustment for each correction file. These features are mainly used to adjust the fields of both heads when a secondary head is used. See also: Chapter Optic Settings Dialog.

For the RTC5 card the X and Y Gain values have to be equal.

3D Ext: Opens a dialog, where a Z-Table can be defined. For detailed information have a look at the RTC5 manual. See chapter Optic3D for RTC cards.

Laser: Globally enables or disables laser output.

Mark on Fly: Requires the Marking On The Fly RTC option. MOTF related parameters are described in chapter Card Specific: RTC cards.

StandBy: Globally enables standby mode.

Stand-by: Q-Switch length in µs for stand-by modus. If this is set to zero the stand-by mode is switched off.

Half-period: Half of the laser pulse period for stand-by modus.

Settings are done after leaving the global dialog Settings. Standby Settings for pens will overwrite these settings if enabled for a pen as soon as this pen is used.

EdgeLevel: The variable Polygon Delay gets very high if the angle of two successive vectors is close to 180° . This can lead to burn in effects. To prevent this an *EdgeLevel* value can be defined. If the Polygon Delay between two mark commands is greater than this value the RTC5 card switches off the laser after the first command and after the laser-off delay is over. Then a new polygon marking with the second vector will be started.

Var Jump Delay: Normally after a jump command a constant jump delay is inserted. But for very small jumps it is not necessary to have such a long jump delay. The jump delay can be reduced without loosing marking quality. The minimum delay is the delay for a jump of length 0.

IO:

Lamp/8-bit: Sets the 8 bit or one of the digital outputs of the RTC Card during start up (as selected under Laserport). The 8 bit Output corresponds to the write_8bit_port command of the RTC.

16 BIT Out: Sets the 16 bit output of the RTC Card during start up.

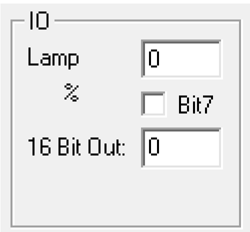


Figure 26: If *LEE Mode* is selected, the eighth bit is selectable separately

Mode: Here you can choose the type of the laser.

YAG2, YAG3 and YAG4: YAG4 corresponds to Laser Mode 4 like described in the SCANLAB RTC manual - YAG2 and YAG3 respectively.

LaserMode6: A synchronization signal will be given at Laser1 output for *laser active* and *laser standby* operation. Please see RTC5 manual for detailed information.

VarPolyDelay: If checked the length of the polygon delay gets varied depending on the angle between two successive vectors.

MoF: Shows whether the scanner card is able to do Marking On The Fly or not.

z-Axis: Indicates whether the card is able to do 3D Marking or not.

PixelDAC: Enables Amplitude Modulation.

PixelTime: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

InvertPixel: Inverts bitmap pixels.

Pixel Mode 0: See chapter Pixelmode.

InvertLaserOn: Inverts the Laser_On status bit (15-pin SUB-D Laser connector).

InvertLaser1/2: Inverts the Laser1 and Laser2 status bits (15-pin SUB-D Laser connector).

AutoCal: Displays the ScAutoCalib button within the functionality object toolbar to generate AutoCalib control objects in the entity list, see section AutoCal Control Object.

DoNotCutOffPulsesOnLaserOn: The final pulse is fully executed despite completion of the LASERON signal. Please consult RTC5 manual for Ctrl Command set_laser_control at Bit #0 Pulse Switch Setting.

Enable Output Synchronization: Synchronize scanner signals with a freely running laser. Please read RTC5 manual section 7.4.10 for detailed information (command set_laser_control).

Laserport: Defines the port that sends the power signal for the laser if the laser is not a CO2 Laser. For a CO2 Laser the power signal is done by modulating the Laser A signal.

Invert: Here the laser power value can be inverted. Note: In case of 8-bit this checkbox does not invert the bits.

Auto Laser Control...: By clicking this button a new window opens where a position or speed control of the laser can be defined. For more details pleas refer to the RTC5 manual. See chapter Auto Laser Control.

The RTC PCI Board does not support power saving modes, that switch off power to the PCI bus. This could lead to PC freezes at the start of SAMLight. Accordingly, you must disable standby or sleep modes of the operating system and via the „SleepMode.cmd“ script, which you can find in your RTC tools folder. Please refer to your RTC manual for more information.

4.1.9.1 Auto Laser Control

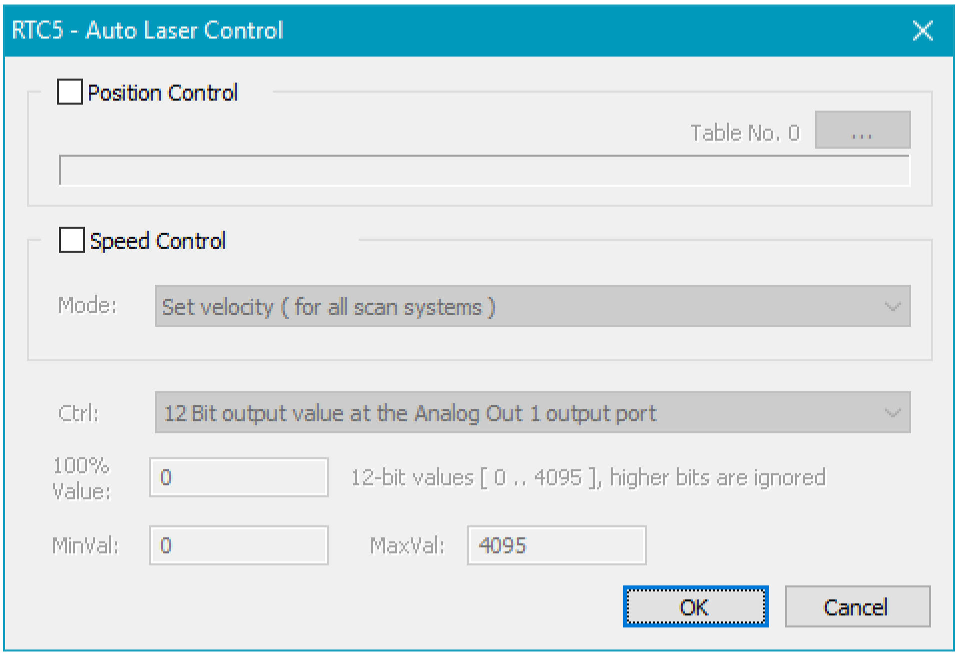


Figure 27: Auto Laser Control Settings for RTC5 Card

Position Control: Activates the position-dependent laser control, which performs an automatic position-dependent correction. For more information, please take a look at RTC5 manual at subsection Position-Dependent Laser Control.

Speed Control: Activates the speed-dependent laser control, which performs an automatic speed-dependent correction. For more information, please take a look at RTC5 manual at subsection Speed-Dependent Laser Control.

4.1.10 RTC-6 Settings

Card Settings for RTC-6 card. This dialog can also be opened when SAMLight is running: *Menu bar → Settings → System → Card → Advanced...*

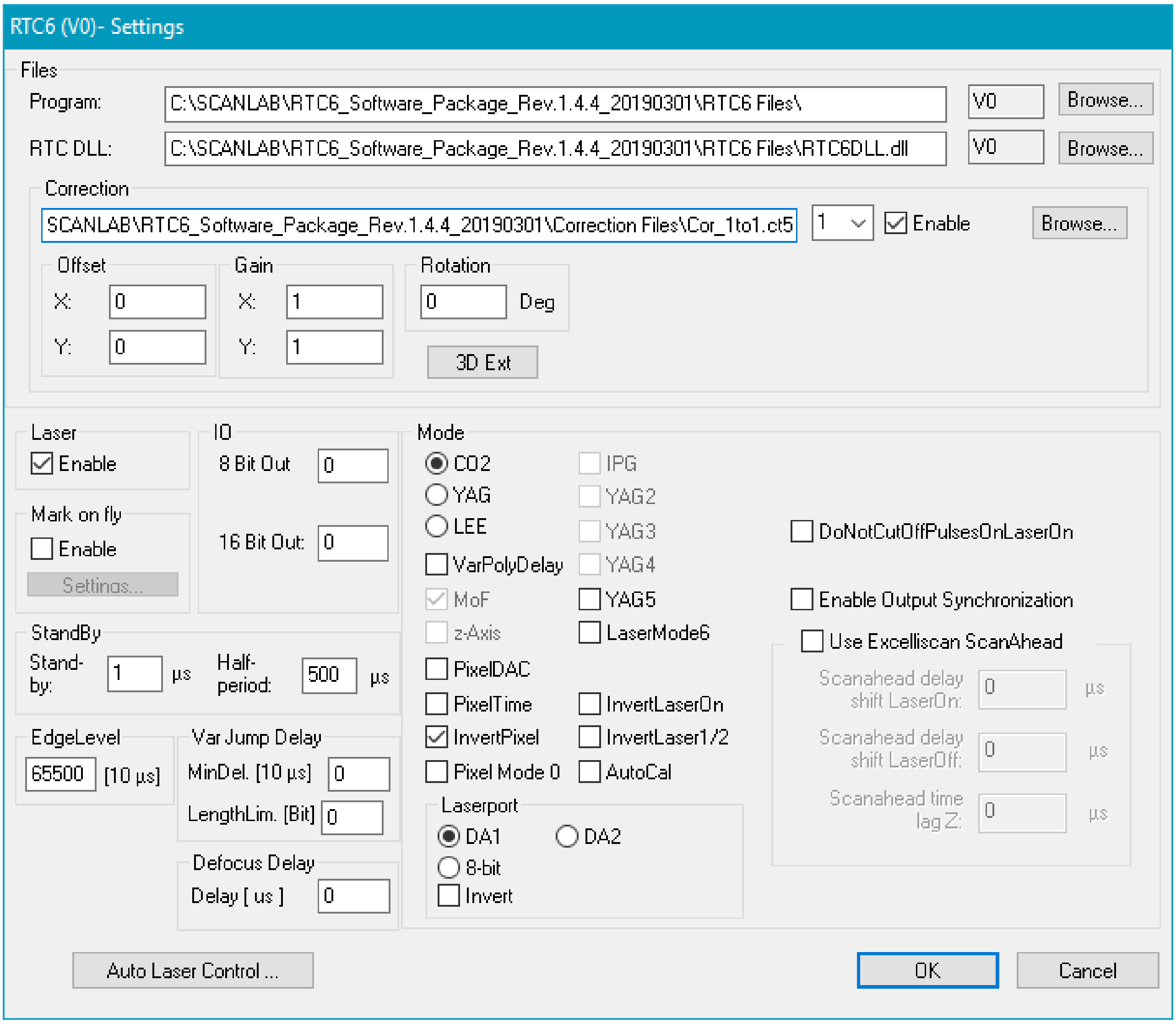


Figure 28: Card Settings for RTC6 Card

Files:

Program: Specifies the path to the *RTC6 Files* folder, delivered together with the scanner controller card driver.

RTC DLL: Specifies the path to the *RTC6DLL.dll* file. This file is in the *RTC6 Files* folder, delivered together with the scanner controller card driver.

NOTE

Using a 64 bit operating system you should nevertheless load a 32 bit RTC6DLL.dll, not the 64 bit RTC6DLLx64.dll.

Correction: Specifies the location of the correction file. This file is delivered together with the scanner card. The extension is *.ct5. Click on *Browse* button to make a search on the files location or type in the file name into the edit window. It is possible to load two different correction files. This is mainly used in connection with the secondary head feature of the RTC6.

Offset, Gain, Rotation: Allow a global adjustment for each correction file. These features are mainly used to adjust the fields of both heads when a secondary head is used. See also: Chapter Optic Settings Dialog.

For the RTC6 card the X and Y Gain values have to be equal.

3D Ext: Opens a dialog, where a Z-Table can be defined. For detailed information have a look at the RTC6 manual. See chapter Optic3D for RTC cards.

Laser: Globally enables or disables laser output.

Mark on Fly: Requires the Marking On The Fly RTC option. MOTF related parameters are described in chapter Card Specific: RTC cards.

StandBy: Globally enables standby mode.

Stand-by: Q-Switch length in µs for stand-by modus. If this is set to zero the stand-by mode is switched off.

Half-period: Half of the laser pulse period for stand-by modus.

Settings are done after leaving the global dialog Settings. Standby Settings for pens will overwrite these settings if enabled for a pen as soon as this pen is used.

EdgeLevel: The variable Polygon Delay gets very high if the angle of two successive vectors is close to 180° . This can lead to burn in effects. To prevent this an *EdgeLevel* value can be defined. If the Polygon Delay between two mark commands is greater than this value the RTC6 card switches off the laser after the first command and after the laser-off delay is over. Then a new polygon marking with the second vector will be started.

Var Jump Delay: Normally after a jump command a constant jump delay is inserted. But for very small jumps it is not necessary to have such a long jump delay. The jump delay can be reduced without loosing marking quality. The minimum delay is the delay for a jump of length 0.

IO:

Lamp/8-bit: Sets the 8 bit or one of the digital outputs of the RTC Card during start up (as selected under Laserport). The 8 bit Output corresponds to the write_8bit_port command of the RTC.

16 BIT Out: Sets the 16 bit output of the RTC Card during start up.

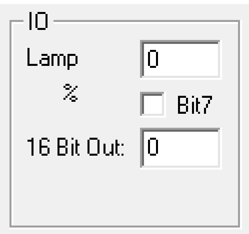


Figure 29: If *LEE Mode* is selected, the eighth bit is selectable separately

Mode: Here you can choose the type of the laser.

YAG2, YAG3 and YAG4: YAG4 corresponds to Laser Mode 4 like described in the SCANLAB RTC manual - YAG2 and YAG3 respectively.

LaserMode6: A synchronization signal will be given at Laser1 output for *laser active* and *laser standby* operation. Please see RTC6 manual for detailed information.

VarPolyDelay: If checked the length of the polygon delay gets varied depending on the angle between two successive vectors.

MoF: Shows whether the scanner card is able to do Marking On The Fly or not.

z-Axis: Indicates whether the card is able to do 3D Marking or not.

PixelDAC: Enables Amplitude Modulation.

PixelTime: Enables Pulse Width Modulation. For more details see chapter Pulse Modulation.

InvertPixel: Inverts bitmap pixels.

Pixel Mode 0: See chapter Pixelmode.

InvertLaserOn: Inverts the Laser_On status bit (15-pin SUB-D Laser connector).

InvertLaser1/2: Inverts the Laser1 and Laser2 status bits (15-pin SUB-D Laser connector).

AutoCal: Displays the ScAutoCalib button within the functionality object toolbar to generate AutoCalib control objects in the entity list, see section AutoCal Control Object.

DoNotCutOffPulsesOnLaserOn: The final pulse is fully executed despite completion of the LASERON signal. Please consult RTC5 manual for Ctrl Command set_laser_control at Bit #0 Pulse Switch Setting.

Enable Output Synchronization: Synchronize scanner signals with a freely running laser. Please consult the RTC6 manual for information on set_laser_control.

Use Excelliscan ScanAhead: Activates the SCANahead control functionality of an excelliSCAN scan head. Important: the excelliSCAN must be powered before starting SAMLight. Additional RTC6 SCANahead parameters can be set at Pen Misc Settings.

Scanahead delay shift Laser_On: Sets the time delay for the LASERON signal in [1/64 µs]. Please consult RTC6+excelliSCAN manual for Ctrl Command set_scanahead_laser_delay_shift for more information.

Scanahead delay shift LaserOff: Sets the time delay for the LASEROFF signal in [1/64 µs]. Please consult RTC6+excelliSCAN manual for Ctrl Command set_scanahead_laser_delay_shift for more information.

Scanahead time lag Z: Sets the time delay for the z time lag compensation in [10 µs]. Please consult RTC6 manual for Ctrl Command set_timelag_compensation for more information.

Laserport: Defines the port that sends the power signal for the laser if the laser is not a CO2 Laser. For a CO2 Laser the power signal is done by modulating the Laser A signal.

Invert: Here the laser power value can be inverted. Note: In case of 8-bit this checkbox does not invert the bits.

Auto Laser Control...: By clicking this button a new window opens where a position or speed control of the laser can be defined. For more details pleas refer to the RTC5 manual. See chapter Auto Laser Control.

WARNING

The RTC PCI Board does not support power saving modes, that switch off power to the PCI bus. This could lead to PC freezes at the start of SAMLight. Accordingly, you must disable standby or sleep modes of the operating system and via the „SleepMode.cmd“ script, which you can find in your RTC tools folder. Please refer to your RTC manual for more information.

4.1.10.1 Auto Laser Control

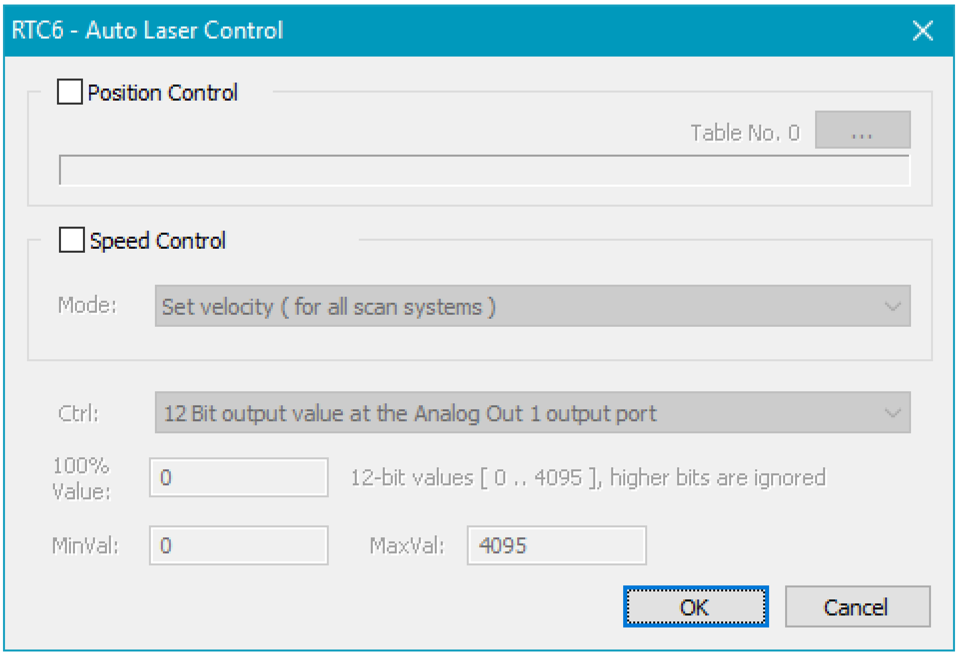


Figure 30: Auto Laser Control Settings for RTC6 Card

Position Control: Activates the position-dependent laser control, which performs an automatic position-dependent correction. For more information, please take a look at RTC6 manual at subsection Position-Dependent Laser Control.

Speed Control: Activates the speed-dependent laser control, which performs an automatic speed-dependent correction. For more information, please take a look at RTC6 manual at subsection Speed-Dependent Laser Control.

Spot Distance Control: Activates the spot distance-dependent laser control, which performs an automatic spot-distance correction in connection with the SCANahead Technology. For more information, please take a look at RTC6 manual at subsection Spot Distance Control. To use Spot Distance Control, activate the "Speed Control" checkbox in connection with Mode "Queried actual velocity (SCANahead)" and Ctrl "Spot distance control". Set the Spot distance in [Bits] below "MinVal:" value and activate the "Use excelliSCAN SCANahead" checkbox at RTC6 settings dialog in connection with the set SCANahead delays. Spot Distance Control is available since installer 3.9.5 Build 42.

4.1.11 Timing Diagram Laser

Timing diagrams to show default settings are given for the following laser types:

- Laser type 0, version 0, CO2
- Laser type 0, version 0, YAG
- Laser type 0, version 0, IPG

4.1.11.1 CO2 laser

0Laser type 0, version 0, CO2

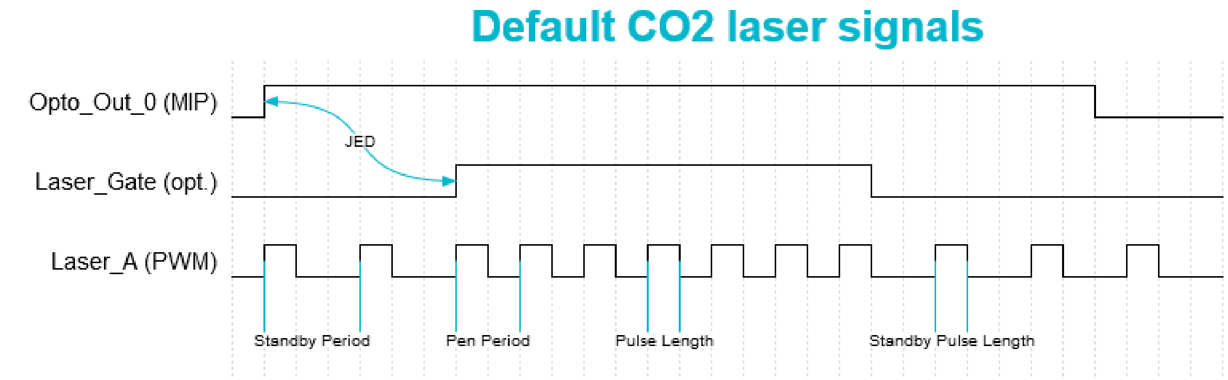


Figure 31: Timing diagram of default CO2 laser signals

- Opto_Out_0 is the marking in progress (MIP) signal which is activated at the start of the job and closed at the end of the job.
- The Laser_Gate signal is activated with a minimum delay of the job execution delay (JED).
- The pen frequency is used for Laser_A. The power (PWM, duty cycle) of the selected pen is set via the pulse length of Laser_A (according to the power in % which is set in the pen, the Power Map and the overwrite parameter).
- For inactive Laser_Gate signal, the standby frequency is taken for Laser_A with the standby pulse length (if enabled).

4.1.11.2 YAG laser

Laser type 0, version 0, YAG

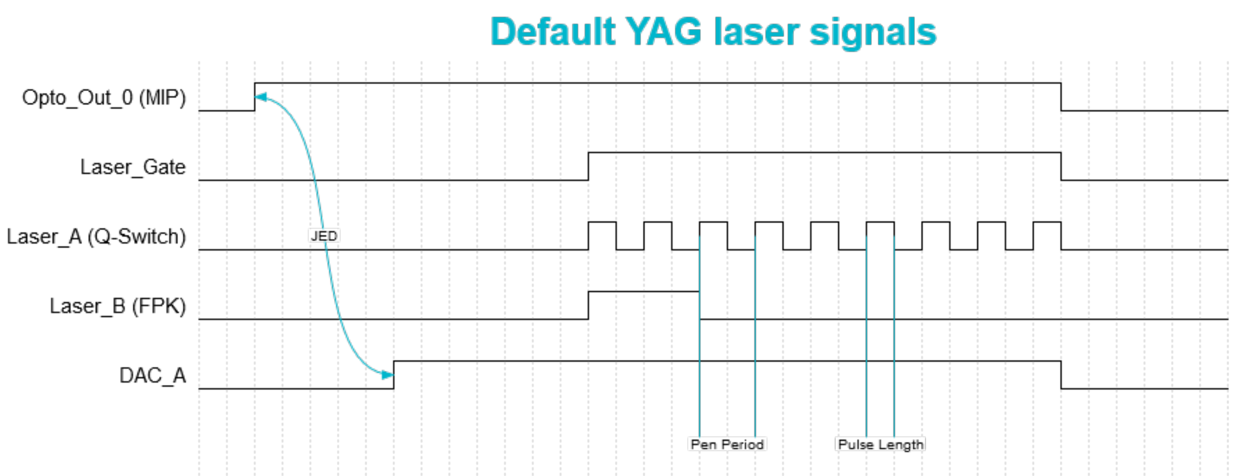


Figure 32: Timing diagram of default YAG laser signals

- Opto_Out_0 is the marking in progress (MIP) signal which is activated at the start of the job and closed at the end of the job.
- The power is set on the chosen laserport (DAC_A in this example; depending on the power of the pen, the Power Map and the overwrite parameter; other available laserports are: DAC_B, 8-bit digital) with a minimum delay of the job execution delay (JED).
- Laser_Gate is activated at the beginning of the vector marking. Starting with Laser_Gate, the frequency of the selected pen is activated for the Laser_A signal. For inactive Laser_Gate signal, Laser_A signal is inactive as well.
- The pulse length of Laser_A can be specified in the pen settings.

- Laser_B if connected gives a first pulse killer (FPK) signal synchronized with Laser_Gate.

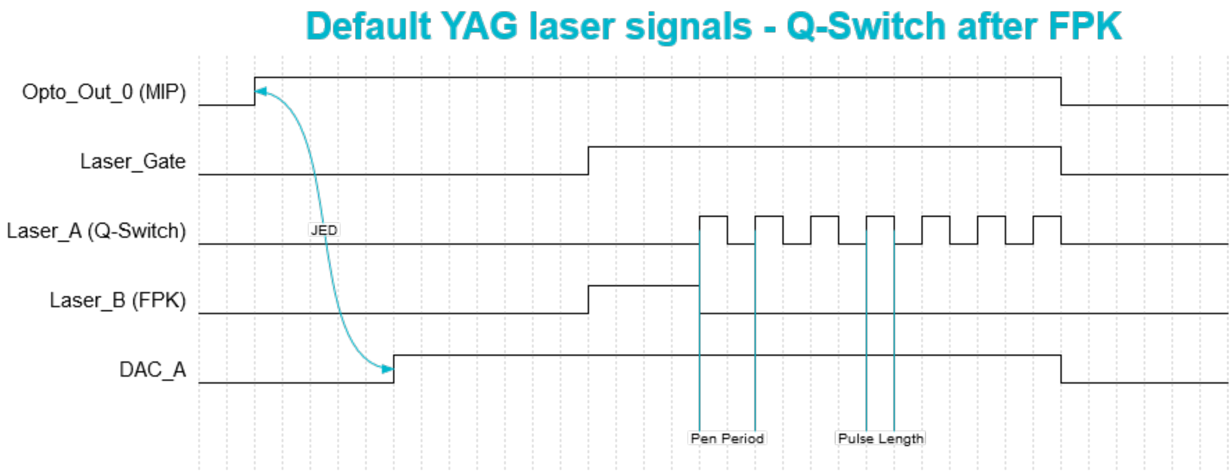


Figure 33: Timing diagram of default YAG laser signals with Q-Switch after FPK

- Opto_Out_0 is the marking in progress (MIP) signal which is activated at the start of the job and closed at the end of the job.
- The power is set on the chosen laserport (DAC_A in this example; depending on the power of the pen, the Power Map and the overwrite parameter; other available laserports are: DAC_B, 8-bit digital) with a minimum delay of the job execution delay (JED).
- Laser_Gate is activated at the beginning of the vector marking.
- If "Q-Switch after FPK" is active, the frequency of the selected pen is activated for the Laser_A signal only after the end of the first pulse killer signal. For inactive Laser_Gate signal, Laser_A signal is inactive as well.
- The pulse length of Laser_A can be specified in the pen settings.
- Laser_B if connected gives a first pulse killer (FPK) signal synchronized with Laser_Gate.

4.1.11.3 IPG laser

Laser type 0, version 0, IPG

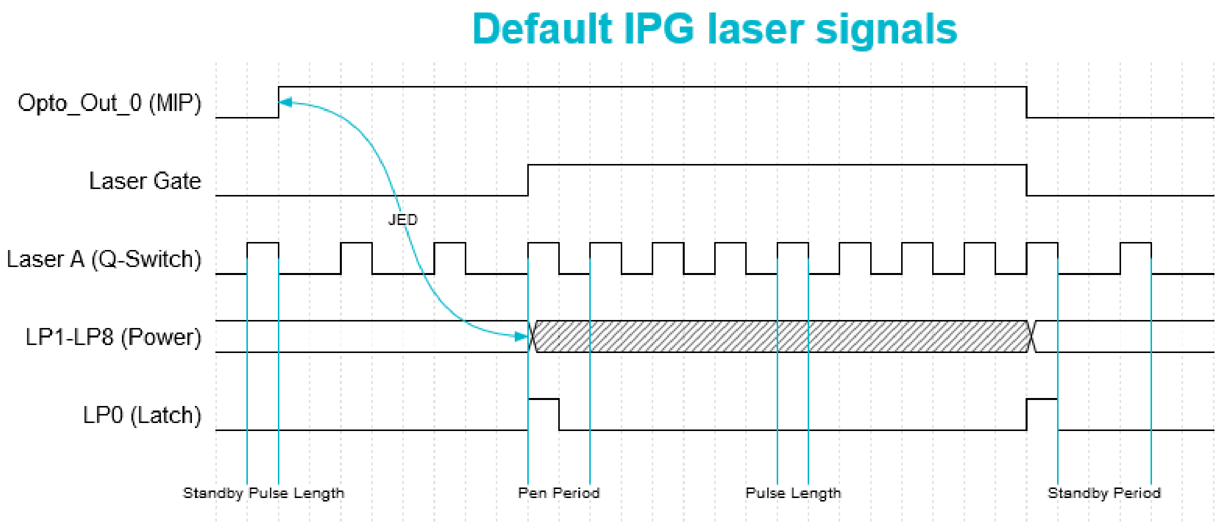


Figure 34: Timing diagram of default IPG laser signals

- Opto_Out_0 is the marking in progress (MIP) signal which is activated at the start of the job and closed at the end of the job.
- Laser_Gate is activated at the beginning of the vector marking.

- The power is set on 8-bit digital laser port (depending on the power of the pen, the Power Map and the overwrite parameter) with a minimum delay of the job execution delay (JED).
- Starting with Laser_Gate, the frequency and pulse length of the selected pen is activated for the Laser_A signal. For inactive Laser_Gate signal, the standby frequency is taken for Laser_A with the standby pulse length (if enabled).
- LP0 gives a latch signal for changes in the power (pen settings).

4.2 Diagnostics

The menu item *Diagnostics → Dongle* displays the following dialog.

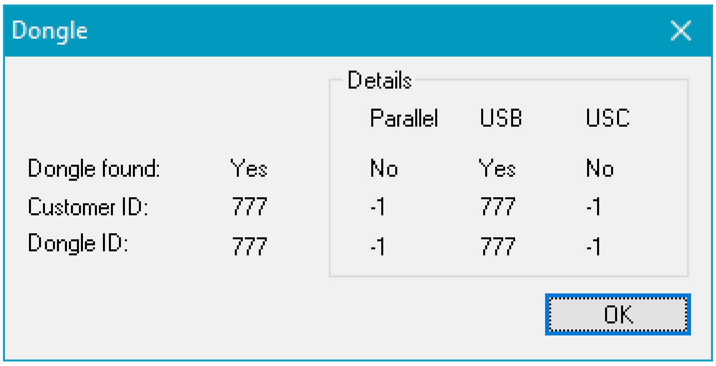


Figure 35: Dongle Dialog

NOTE

Use the sc_usc_server.exe to display the dongle information of multiple USC cards!

Only one USB-dongle displayed. Do not use more than one!

4.3 Resource

All available parameters are explained below.

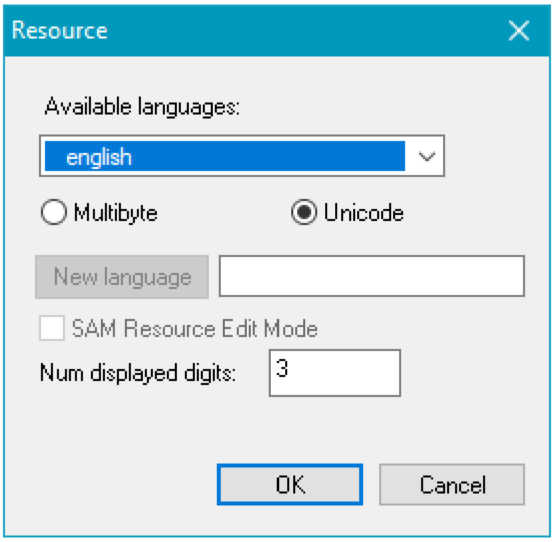


Figure 36: sc_setup.exe dialog

Available languages: Select the desired language. The amount of available languages depends on the checkbox below.

Multibyte/Unicode: Swap between Multibyte or Unicode language resources. Customized languages are located in the *Multibyte* mode.

New language: Enter new language name.

SAM Resource Edit Mode: Activate the Resource Edit Mode.

Num displayed digits: Edit the total number of digits after the comma.

5 Motion Control Settings

The type of motion controller needs to be defined in the text file ***sc_motion_settings.txt***. This file can be found in the folder *<SCAPS>\system*.

Type=#: Type of the motion controller stands for a number that specifies the controller (see table below):

Type=#	Name of motion controller	Corresponding settings file in <SCAPS>\system\
0	Disable motion control	none
1	IMS Stepper Drives	ims_settings.txt
2	Standard (not supported)	not supported any more
3	Microcon Drive (not supported)	not supported any more
4	External custom controller	defined in customized *.dll
5	IMS MDrive	sc_motion_mdrive_settings.txt
6	Faulhaber motion controller	sc_motion_faulmc_settings.txt
7	Isel IT Stepper Controller / DNC	sc_motion_iselit_settings.txt
8	Generic stepper controller	sc_motion_stepper_settings.txt
9	Generic RS-232 interface	sc_motion_generic232_settings.txt
10	SHS 2000 Star	sc_motion_shstar2000_settings.txt
11	Jena Ecostep100	sc_motion_jenaecostep100_settings.txt
12	IO Switcher	sc_motion_ioswitcher_settings.txt
13	Isel CanApi controller	sc_motion_isel_settings.txt
14	USC-2 stepper controller	sc_motion_stepper_settings.txt

Table 7: Available motion types

5.1 Direct Motion Control

Direct Motion Control: In the following all features of the Direct Motion Control are described. Depending on the capabilities of the used motion controller some of the functions may be not available.

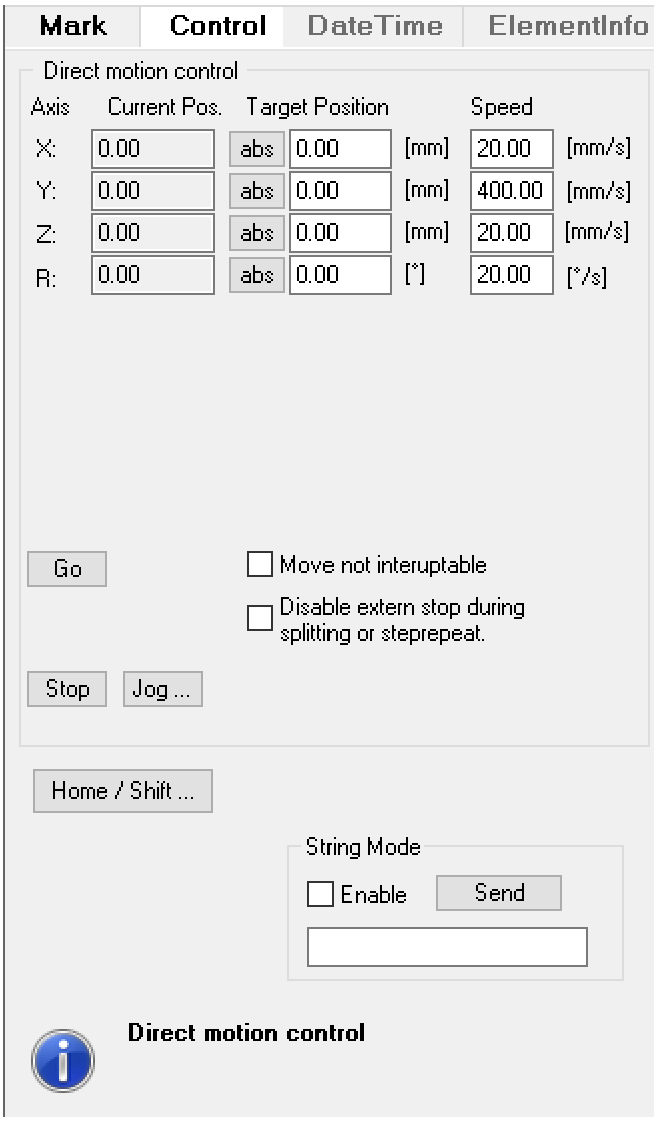


Figure 37: Direct motion control dialog

Direct motion control:

Axis: Axis name like specified in the settings.

Current Pos.: Shows the current position.

abs/rel: The user can switch between absolute or relative movement to the set target position.

Target Position: Target position of the selected axis.

Speed: Define the axis moving speed.

Go: Moves all axes to the defined values above.

Stop: Stops the movement. In some cases you can loose the current position! Then a homing is necessary to re-calibrate your axes.

Update: By pressing this button the actual position is updated. This might be necessary if manual motions are done. The *Update* Button is not available for stepper motor controllers.

Jog...: Opens the Jog Dialog.

Home / Shift: Opens direct motion control Home / Shift dialog. Homing of a single axis is available by right-clicking on this button and then selecting the desired axis.

String Mode: Enable and send RS232 commands.

(There is no further information to both checkboxes.)

5.2 Jog Dialog

NOTE

You can use short keys for the motion control. You can set the short keys via 'Settings → System → Short Keys'. The short keys are assigned to the jog of the axes.

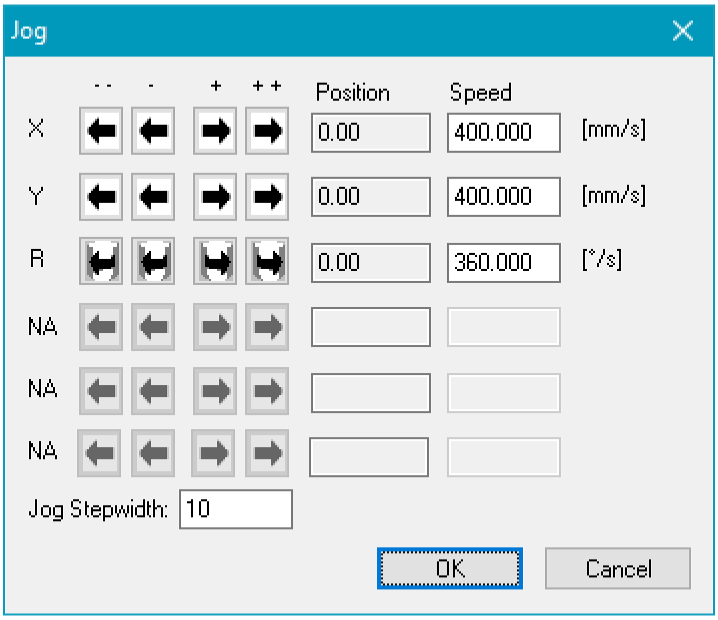


Figure 38: Jog dialog

Jog: Clicking on this button opens a dialog where the motor can be moved in both directions by "Jog Stepwidth" (++) or by "Jog Stepwidth / 10" (+).
The default jog step width value can be set in *Settings→System→Extras*. If the dialog is closed and opened again, this value is restored to the value that has been set in the menu. For each axis an independent speed can be used.

5.3 Home / Shift Dialog

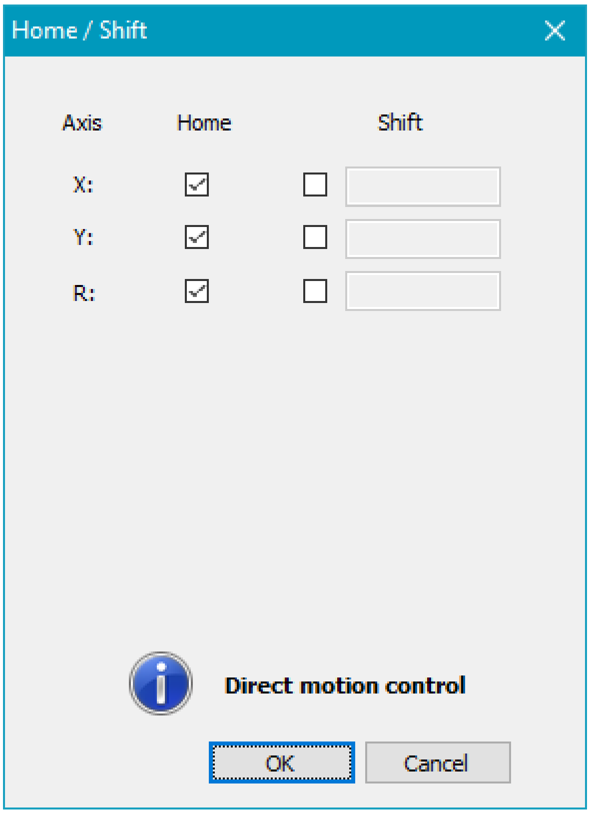


Figure 39: Direct motion control Home / Shift dialog

Home: When leaving the *Home / Shift Dialog* with *OK*, all axes with enabled *Home* checkbox will start the homing procedure.

Shift: When the direct motion control Home / Shift dialog is opened, the current Shift values are shown. When leaving the Home / Shift dialog with *OK*, the shift values will be applied. To reset the current shift of an axis, set the shift value to *0*. The *Home* and *Shift* functionality can be combined, first the homing procedure is performed, then the shifts will be applied.

NOTE

If an axis does not move when homing is triggered, inversion of the polarity of the refIO might help.

A homing procedure consists of 2 parts:

1) Move on the switch with homing speed.

2) Leave the switch with switch leaving speed, which is usually much slower than the homing speed. If the polarity of the reference IO is incorrect, part 2 of the homing process is executed at first. No movement is done because the switch is already left.

5.4 Step & direction motion controller

The following step and direction motion controllers are supported.

5.4.1 Type 8 - Generic stepper controller

Key features

- Available for USC and RTC controller cards
- Up to 7 axes can be controlled
- Axes move one after another
- The IOs of the scanner controller card are used to control the stepper
- Control the motor via step and direction signal
- The maximum possible frequency of the step signals is 10kHz using OPTO_IOs, 25kHz using other possible outputs (Stepper, Digi_IOs, 8bit)
- Homing procedures available

Required settings

- Set 'Type=8' in sc_motion_settings.txt in <*SCAPS*>\system\
- Define sc_motion_stepper_settings.txt in <*SCAPS*>\system\ as described below
- Enable 'Motion Control' in SAMLight (Settings→System→Extras).

How to create the settings file

- The file 'sc_motion_stepper_settings.txt' is a plain text file that contains different configuration parameters.
- A line which begins with a # sign is interpreted as a comment and will be ignored.
- All parameters have to start exactly at the beginning of a line.
- Global parameters have to be specified only once at the head of the settings file.
- Axis-specific parameters have to be defined for all axes separately.
- The axis-specific (and the optional) parameters have to be arranged in one block of parameters per axis. Each block begins with the parameter 'axis'.
- The optional parameters do not have to be specified if you do not want to use them.
- A parameter gets defined by typing <parameter>=<value>, parameter and values are described below,
  e.g. 'axis=0' defines the parameter 'axis' with the value '0'

Global parameters (type 8): These parameters have to be defined once at the head of the settings file.

Debug	Enables the debug log.	
	Value	Function
	0	Disable debug mode
	1	Enable debug mode
	Enable debug mode to log debug information in '<*SCAPS*>\system\sc_motion_stepper_debuglog.txt'. Only enable this option if you need it, because there can be huge amounts of data that are logged into this file.	

DisableHomingDuringStartUp	No homing on startup.	
	Value	Function
	0	Enable homing on startup of SAMLight
	1	Disable homing on startup of SAMLight
	If set to '1', homing is just performed when pressing 'Control→Home' and not when software is started.	

Axis parameters (type 8): All of the following parameters has to be defined for each axis after the global parameters in the settings file. Each of these axes can be configured differently.

axis	Sets the number of axis.	
	Value	Function
	0...6	Index of axis
	Zero based index of up to 7 axes. Defines the order of axis movement and display order in 'SAMLight→Control'. This parameter has to be at the beginning of each axis specific parameter block.	

dname	Sets the name of axis.	
	Value	Function
	a...Z	Name of axis
	Only the first character is used as axis-name in 'SAMLight→Control'.	

mode	Sets the type of the axis.	
	Value	Function
	POSITION	Defines a straight axis [mm].
	ANGLE	Defines a rotational axis [°].
	Depending on the mode of the axis several parameters has to be adjusted (factor, incperrot, sfactor, defspeed, see below).	

incperrot	Converts degrees into steps for ANGLE mode.	Unit: steps/360°
	This value defines how many increments are needed for a whole rotation. 'factor' and 'sfactor' have to be equal to (1/360)*'incperrot'. This parameter is only used in ANGLE mode.	

factor	Converts mm (or °) into steps.	
	POSITION mode, unit: steps/mm	ANGLE mode, unit: steps/°
		Value has to be equal to **(1/360)*'incperrot'**
	This value defines how many increments are needed for 1mm (or 1°). Depending on the factor of the axis several parameters has to be adjusted (incperrot, sfactor, llimit, hlimit, hslimit, accel, decel, refspeed, refspeed2, refpos see below).	

sfactor	Sets speed factor.	
	POSITION mode, unit: steps/mm	ANGLE mode, unit: steps/°
	Value has to be equal to **'factor'** for POSITION and ANGLE mode.	

llimit	Sets the lower limit.	Unit: steps
	No movement below this limit will be possible. Minimum value for llimit is '-1E13'. To get the 'llimit' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

hlimit	Sets the upper limit.	Unit: steps
	No movement above this limit will be possible. Maximum value for hlimit is '1E13'. To get the 'hlimit' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

hslimit	Sets the upper speed limit.	Unit: steps/s
	No speed above this speed limit will be possible. To get the 'hslimit' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

defspeed	Sets default speed displayed in SAMLight user interface.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be smaller or equal than 'hslimit'/'sfactor'.	

accel	Sets increase of speed (in steps per second) per step.	Unit: steps/s per step
	A value of '-1' or '0' disables acceleration. In this case the step frequency corresponds to the applied speed in SAMLight. This is not recommended because of possible hardware damage and miscalibration. Please consider the unit which leads to an exponential acceleration.	

decel	Sets decrease of speed (in steps per second) per step.	Unit: steps/s per step
	A value of '-1' or '0' disables deceleration. In this case the step frequency just stops when the in SAMLight applied position has been reached. This is not recommended because of possible hardware damage and miscalibration. Please use only positive values for this parameter. Please consider the unit which leads to an exponential deceleration.	

accStartSpeed	Sets start speed.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be positive and smaller or equal than 'hslimit'/'factor'. If the in SAMLight demanded speed is smaller than the value for this parameter the axis will move with the lower speed without acceleration.	

decStopSpeed	Sets stop speed.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be positive and smaller or equal than 'hslimit'/'factor'. If the in SAMLight demanded speed is smaller than the value for this parameter the axis will move with the lower speed without deceleration.	

<table>
<tr><td rowspan="2">stepIO</td><td>Sets output bit of the scanner controller card for the step signal.</td></tr>
<tr><td>Value: refer to Stepper I/O parameters</td></tr>
</table>

<table>
<tr><td rowspan="3">dirIO</td><td>Sets output bit of the scanner controller card for the direction signal.</td></tr>
<tr><td>Value: refer to Stepper I/O parameters</td></tr>
<tr><td>Since all axes move successively the 'dirIO' bit can be the same for all axes. Although the parameter 'dirIO' has the same value for multiple axes it still has to be defined for every axis.</td></tr>
</table>

<table>
<tr><td rowspan="4">dirvalue</td><td colspan="2">Sets the polarity of the 'dirIO' signal for positive movement.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>low active</td></tr>
<tr><td>1</td><td>high active</td></tr>
</table>

<table>
<tr><td rowspan="2">DelayAfterDir</td><td>Sets a delay which is executed after change of the direction.</td></tr>
<tr><td>This command is used to specify a delay in us to be waited between a change in direction and the first step in the new direction.</td></tr>
</table>

<table>
<tr><td rowspan="5">refIO</td><td colspan="2">Sets input bit of the scanner controller card for the reference signal.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>Stepper I/O parameters</td><td>Default homing, 'refmode=(1..6)': In type 8 reference movements are software controlled and (due to the limitations of the pc) the maximum speed is much lower and the signal is noisy (jitter).</td></tr>
<tr><td>1
(ext. stop)</td><td>Fast homing (USC only): If this value is used together with 'refvalue=1' and 'refmode=(1 or 4)' the movement to the reference switch is performed with normal speed. This value is recommended only if you use just one axis.</td></tr>
<tr><td>20</td><td>No homing: Use this value and 'refmode=0' if no reference switch is used.</td></tr>
</table>

<table>
<tr><td rowspan="4">refvalue</td><td colspan="2">Sets the polarity of the 'refIO' signal.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>low active</td></tr>
<tr><td>1</td><td>high active</td></tr>
</table>

refmode	Sets the behavior of homing movement.	
	Value	Function
	Common reference modes:	
	0	No homing movement, current position is set to 'refpos'
	1	Go to switch in neg. dir. and leave it in pos. dir.
	4	Go to switch in pos. dir. and leave it in neg. dir.
	Uncommon reference modes:	
	2	Go to switch in neg. dir. and leave it in neg. dir.
	3	Go to switch in neg. dir. and stay there
	5	Go to switch in pos. dir. and leave it in pos. dir.
	6	Go to switch in pos. dir. and stay there

refspeed	Sets the speed of the homing movement.	Unit: steps/s
	This value defines how fast the motor moves in case of a homing movement to find the reference switch. Reference movements are software controlled and have due to limitations of the PC a much lower maximum speed. To get the 'refspeed' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

refspeed2	Sets the speed while leaving the reference switch.	Unit: steps/s
	If refspeed2 is '-1', 'refspeed'/4 is used. To get the 'refspeed2' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

refpos	Sets the home position of the axis.	Unit: steps
	This value will be set after SAMLight startup and after a homing movement. refpos should be outside the position limits 'llimit' and 'hlimit' to avoid that the reference switch is activated during normal movement. To get the 'refpos' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

Optional axis parameters (type 8): All of the following optional parameters can be defined for each axis. Each axis can be configured differently. If no parameter is specified the corresponding feature will not be used.

SignalAxisMoving	Sets an output bit during an axis movement.	
	Value	Function
	0	inactive
	1	active

SignalAxisMovingStateBitPosition	Sets output bit of the scanner controller card for the 'SignalAxisMoving' signal.	
	Value: refer to Stepper I/O parameters	

SignalAxisMovingState	Sets the polarity of the 'SignalAxisMoving' signal.	
	Value	Function
	0	low active
	1	high active

SignalAxisMovingStatePreDelay	Delays the start of the motion.	Unit: ms
	Use SignalAxisMovingStatePreDelay to add a delay before the start of the motion.	

MoveThisAxisFirst	Sets homing priority of this axis.	
	Value	Function
	0	disable homing priority
	1	enable homing priority
	This parameter can only be used for one axis. If it is defined for more than one axis the parameter will be ignored for all axis.	

`corr1 0.0 0.0`
`corr2 500.0 100000.0`
`corr3 1000.0 200000.0`
`corr4 1500.0 300000.0`
`corr5 2000.0 400000.0`
 	
<corr#> <mm#> <steps#>
	Correction table instead of parameter 'factor' for POSITION mode.

	This parameter is defined without '=' sign in the form <corr#> <mm> <steps>.
	This correction table can be used to compensate nonlinearities of a straight axis. If 'factor' is defined, the correction table is ignored. Use for corr1 the same step value as for 'llimit' and for corr5 the same step value as for 'hlimit'.

Example sc_motion_stepper_settings.txt file for 1 straight and 1 rotational axis for 'Type=8':

```
# Global parameters:
Debug=0
DisableHomingDuringStartUp=1

# Z-axis parameters:
axis=0
dname=Z
mode=POSITION

factor=200.0
sfactor=200.0
llimit=-400000
hlimit=400000
hslimit=6000
defspeed=20.0
accel=50
decel=50
accStartSpeed=5.0
decStopSpeed=5.0

stepIO=1
dirIO=3
dirvalue=0

refIO=2
refvalue=1
refmode=0
refspeed=2000.0
refspeed2=500.0
refpos=0.0

# R-axis parameters:
axis=1
dname=R
mode=ANGLE

factor=10.0
incperrot=3600
sfactor=10.0
llimit=-1E13
hlimit=1E13
hslimit=6000
defspeed=400.0
accel=50
decel=50
accStartSpeed=5.0
decStopSpeed=5.0

stepIO=4
dirIO=5
dirvalue=0

refIO=3
refvalue=1
refmode=0
refspeed=2000.0
refspeed2=500.0
refpos=0.0
```

5.4.1.1 Motion Settings Dialog Type 8

In Settings → System → Extras there is a button "Motion settings dialog". If you press it you will access the settings dialog GUI for the stepper motor type 8:

Here you can configure the motor as described in the file <SCAPS>\system\sc_motion_stepper_settings.txt.

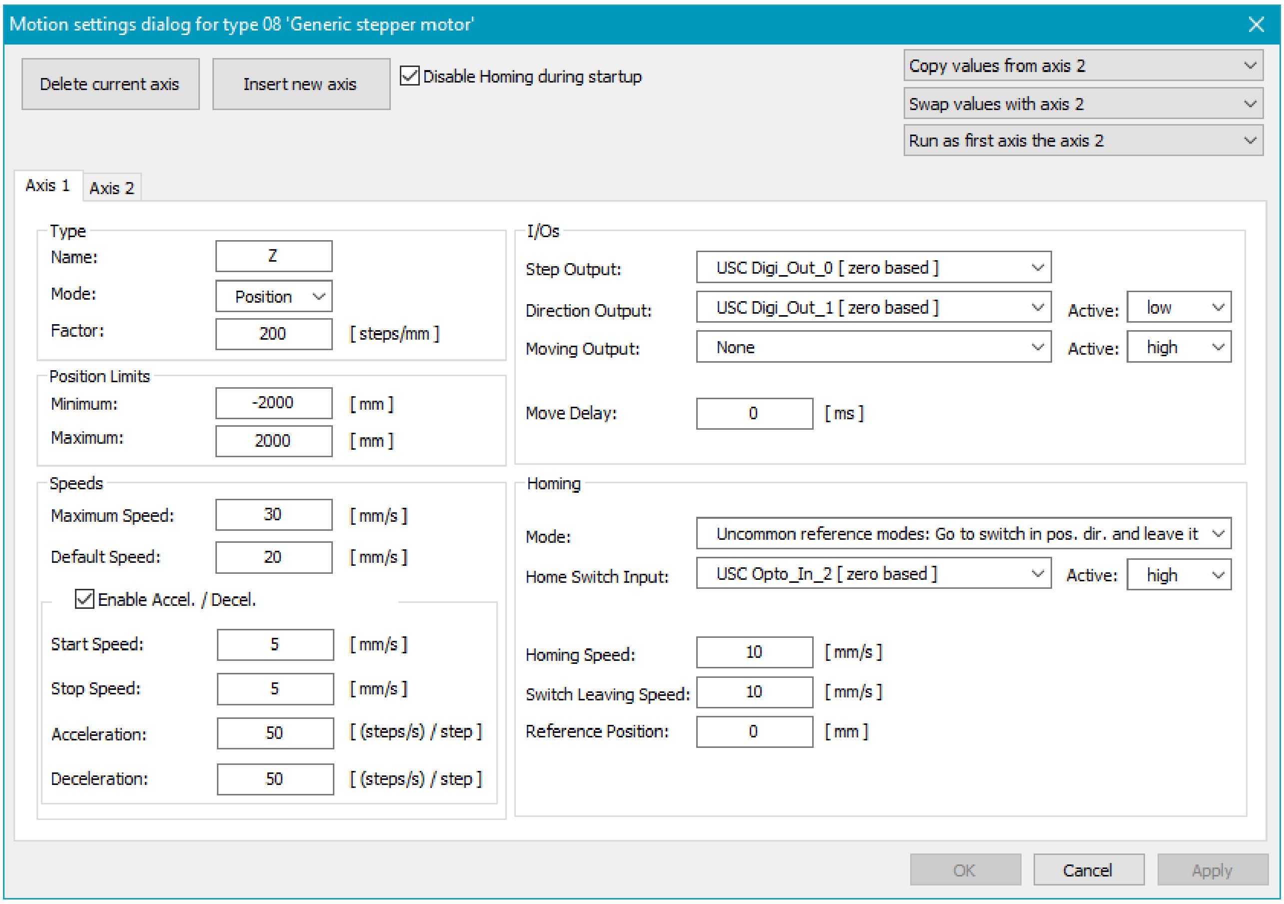


Figure 40: Stepper GUI for motor type 8

Disable Homing during startup: Enable automatic homing after the SAMLight startup.

Delete current axis: Delete the currently opened axis tab.

Insert new axis: Add a new axis (max. 6 for Motion Type 14) at the end of the list.

Copy values from axis: Choose an other axis in the drop down menu and copy all values in the current selected axis tab.

Swap values with axis: Choose an other axis in the drop down menu and swap all values between the selected and the current axis tab.

Run as first axis the axis: Choose an axis to run with the configuration of axis 1.

Type:

Name: Sets name of axis. Has to be an unique single capital letter for each axis.

Mode: Position[mm] as straight axis or Angle[°] as rotational axis can be chosen.

Factor:

For 'mode=POSITION': Unit: [steps/mm]

For 'mode=ANGLE'   : Unit: [steps/ °], value has to be (1/360°)*'incperrot'

Converts mm (or °) used in SAMLight into steps.

Position limits: Both values are read as double to increase the value range [-1E13, 1E13]. This limits should not contain 'refpos'.

Minimum: This parameters defines the lower limit of the axis.

Maximum: This parameters defines the upper limit of the axis.

Speeds: Unit depends on chosen axis. For planar Unit: [mm/sec] and for roation Unit: [°/sec]

Maximum speed: Define the maximum moving speed.

Default speed:

For 'mode=POSITION': Unit: [mm/sec]

For 'mode=ANGLE'   : Unit: [°/sec]

Sets default speed displayed in SAMLight user interface.

Enable Accel./Decel.: The checkbox is always enabled.

Start speed: Specify the speed, where the motor should start to accelerate.

Stop speed: Specify the speed, where the motor should end to decelerate.

Accel. / Decel.:

Unit: increase of [steps/second] per step, def. value: 50. Please consider the unit which leads to an exponentially acceleration.

A value of '0' or '-1' disables acceleration /deceleration.

I/Os:

Step output: Select the output bit for the step signals by the drop down menu.

Direction output: Select the output bit for the direction signals by the drop down menu.

active: Choose if the direction output bit state should be active high or active low.

Moving output: Specify the output bit during the axis is moving.

active: Choose if the moving output bit state should be active high or active low.

Move delay: Unit: [msec]. Use this parameter to delay the start of the motion.

Homing: This mode defines the behavior of homing.

Mode: Choose the required homing mode in the drop down menu. Homing is set off per default.

1. No homing movement, current position is set to 'refpos'.
2. Go to switch in neg. dir. and leave it in pos. dir.
3. Go to switch in pos. dir. and leave it in neg. dir.
4. Uncommon reference modes: Go to switch in neg. dir. and leave it in neg. dir.
5. Uncommon reference modes: Go to switch in neg. dir. and stay there.
6. Uncommon reference modes: Go to switch in pos. dir. and leave it in pos. dir.
7. Uncommon reference modes: Go to switch in pos. dir and stay there.

Home switch input: Select the input bit for the home switch input signals by the drop down menu.

active: Choose if the input bit state should be active high or active low.

Homing speed: Specify the speed during the homing process.

Switch leaving speed: Specify the switch leaving speed.

Reference position: Set a reference position for the homing.

5.4.2 Type 14 - USC-2/-3 stepper controller

Key features

- Only available for USC-2 / USC-3 cards
- Up to 6 axes can be controlled
- All axes can move at the same time if they have the same speed. Otherwise they will move one after another. The X-, Y- and Z-axis can move interpolated so that they reach the final position of the movement at the same time.
- The IOs of the USC-2/-3 card are used to control the stepper
- Control the motor via step and direction signal
- The signals are controlled on the USC-2/-3 card directly
- The maximum possible frequency of the step signals is 16666.7 Hz
- Available in Flash-Mode
- Fast homing procedures available
- Keeps position values of axes after a movement was interrupted
- It can be used via regular motion control in SAM programming, or via G-Code commands (RS-232, Telnet or flash commands)

Required settings

- Set 'Type=14' in sc_motion_settings.txt in <*SCAPS*>\system\
- Define sc_motion_stepper_settings.txt in <*SCAPS*>\system\ as described below
- Enable 'Motion Control' in SAMLight (Settings→System→Extras).
- Use SAMLight Version later than May 2013 and use related firmware on the USC-2/-3 card

How to create the settings file

- The file 'sc_motion_stepper_settings.txt' is a plain text file that contains different configuration parameters.
- A line which begins with a # sign is interpreted as a comment and will be ignored.
- All parameters have to start exactly at the beginning of a line.
- Global parameters have to be specified only once at the head of the settings file.
- Axis-specific parameters have to be defined for all axes separately.
- The axis-specific (and the optional) parameters have to be arranged in one block of parameters per axis. Each block begins with the parameter 'axis'.
- The optional parameters do not have to be specified if you do not want to use them.
- A parameter is defined by typing <parameter>=<value>, parameter and values are described below,
  e.g. 'axis=0' defines the parameter 'axis' with the value '0'
- To save the stepper settings on the USC-2/-3 card (if you want to use it in Flash-mode) you need to click the *Store* button in the System→Settings→Optic→Advanced dialog in SAMLight.

Global parameters (type 14): These parameters have to be defined once at the head of the settings file.

Debug	Enables the debug log.	
	Value	Function
	0	Disable debug mode
	1	Enable debug mode
	Enable debug mode to log debug information in '<*SCAPS*>\system\sc_motion_stepper_debuglog.txt'. Only enable this option if you need it, because there can be huge amounts of data that are logged into this file.	

DisableHomingDuringStartUp	No homing on startup.	
	Value	Function
	0	Enable homing on startup of SAMLight
	1	Disable homing on startup of SAMLight
	If set to '1', homing is just performed when pressing 'Control→Home' and not when software is started.	

Axis parameters (type 14): All of the following parameters have to be defined for each axis after the global parameters in the settings file. Each of these axes can be configured differently.

axis	Sets the number of axis.	
	Value	Function
	0...5	Index of axis
	Zero based index of up to 6 axes. Defines the order of axis movement and display order in 'SAMLight→Control'. This parameter has to be at the beginning of each axis specific parameter block.	

dname	Sets the name of axis.	
	Value	Function
	a...Z	Name of axis
	Only the first character is used as axis-name in 'SAMLight→Control'.	

mode	Sets the type of the axis.	
	Value	Function
	POSITION	Defines a straight axis [mm].
	ANGLE	Defines a rotational axis [°].
	Depending on the mode of the axis several parameters have to be adjusted (factor, incperrot, defspeed, see below).	

incperrot	Converts degrees into steps for ANGLE mode.	Unit: steps/360°
	This value defines how many increments are needed for a whole rotation. 'factor' has to be equal to (1/360)*'incperrot'. This parameter is only used in ANGLE mode.	

factor	Converts mm (or °) into steps.	
	POSITION mode, unit: steps/mm	ANGLE mode, unit: steps/°
		Value has to be equal to **(1/360)*'incperrot'**
	This value defines how many increments are needed for 1mm (or 1°). Depending on the factor of the axis several parameters have to be adjusted (incperrot, llimit, hlimit, hslimit, accel, decel, refspeed, refspeed2, refpos see below).	

llimit	Sets the lower limit.	Unit: steps
	No movement below this limit will be possible. Minimum value for llimit is '-1E13. To get the 'llimit' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

hlimit	Sets the upper limit.	Unit: steps
	No movement above this limit will be possible. Maximum value for hlimit value is '1E13' To get the 'hlimit' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

hslimit	Sets the upper speed limit.	Unit: steps/s
	No speed above this speed limit will be possible. The maximum possible frequency of the direction signals is 16666.7 Hz. To get the 'hslimit' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

defspeed	Sets default speed displayed in SAMLight user interface.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be smaller or equal than 'hslimit'/'factor'.	

accel	Sets increase of speed.	Unit: steps/s^2
	The values '0' and '-1' are not valid. If you want to avoid acceleration in general you can set 'accStartSpeed=hslimit/factor'. To get the 'accel' in steps/s^2 like it is required here the value in mm/s^2 (or °/s^2) has to be multiplied by the 'factor'.	

decel	Sets decrease of speed.	Unit: steps/s^2
	Please use only positive values for this parameter. The values '0' and '-1' are not valid. If you want to avoid deceleration in general you can set 'decStopSpeed=hslimit/factor'. To get the 'decel' in steps/s^2 like it is required here the value in mm/s^2 (or °/s^2) has to be multiplied by the 'factor'.	

accStartSpeed	Sets start speed.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be positive and smaller or equal than 'hslimit'/'factor'. If the in SAMLight demanded speed is smaller than the value for this parameter the axis will move with the lower speed without acceleration.	

decStopSpeed	Sets stop speed.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be positive and smaller or equal than 'hslimit'/'factor'. If the in SAMLight demanded speed is smaller than the value for this parameter the axis will move with the lower speed without deceleration.	

stepIO	Sets output bit of the scanner controller card for the step signal.
	Value: refer to Stepper I/O parameters

dirIO	Sets output bit of the scanner controller card for the direction signal.
	Value: refer to Stepper I/O parameters

dirvalue	Sets the polarity of the 'dirIO' signal for positive movement.	
	Value	Function
	0	low active
	1	high active

DelayAfterDir	Sets a delay which is executed after change of the direction.
	This command is used to specify a delay in us to be waited between a change in direction and the first step in the new direction.

refIO	Sets input bit of the scanner controller card for the reference signal.
	Value: refer to Stepper I/O parameters
	Since all axes move successively the 'refIO' bit can be the same for all axes. Although the parameter 'refIO' has the same value for multiple axes it still has to be defined for every axis.

refvalue	Sets the polarity of the 'refIO' signal.	
	Value	Function
	0	low active
	1	high active

<table>
<tr><td rowspan="12">refmode</td><td colspan="2">Sets the behavior of homing movement.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td colspan="2">Common reference modes:</td></tr>
<tr><td>0</td><td>No homing movement, current position is set to 'refpos'
Use 'refmode=0' together with 'RefOnlyForHome=1'.</td></tr>
<tr><td>1</td><td>Go to switch in neg. dir. and leave it in pos. dir.</td></tr>
<tr><td>4</td><td>Go to switch in pos. dir. and leave it in neg. dir.</td></tr>
<tr><td colspan="2">Uncommon reference modes:</td></tr>
<tr><td>2</td><td>Go to switch in neg. dir. and leave it in neg. dir.</td></tr>
<tr><td>5</td><td>Go to switch in pos. dir. and leave it in pos. dir.</td></tr>
<tr><td>11</td><td>Go to switch in neg. dir. and leave it in pos. dir. (no timeout)</td></tr>
<tr><td>44</td><td>Go to switch in pos. dir. and leave it in neg. dir. (no timeout)</td></tr>
</table>

refspeed	Sets the speed of the homing movement.	Unit: steps/s
	This value defines how fast the motor moves in case of a homing movement to find the reference switch. Reference movements are USC-2 controlled and have a much higher maximum speed than stepper type 8. When no reference signal is detected the homing movement is stopped after 10000 mm (or °). In refmode 11 or 44 (no timeout) the reference movement will never stop if no reference switch could be found. To get the 'refspeed' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

refspeed2	Sets the speed while leaving the reference switch.	Unit: steps/s
	If refspeed2 is '-1', 'refspeed/4' is used. When the reference signal is not released the switch leaving movement is stopped after 100 mm (or °). In refmode 11 or 44 (no timeout) the reference movement will never stop if the reference switch is not released. To get the 'refspeed2' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'factor'.	

refpos	Sets the home position of the axis.	Unit: steps
	This value will be set after SAMLight startup and after a homing movement. refpos should be outside the position limits 'llimit' and 'hlimit' to avoid that the reference switch is activated during normal movement. To get the 'refpos' in steps like it is required here the value in mm (or °) has to be multiplied by the 'factor'.	

Optional axis parameters (type 14): All of the following parameters are optional parameters. If a parameter is not specified the corresponding feature will not be used.

<table>
<tr><td rowspan="5">RefOnlyForHome</td><td colspan="2">Sets movement behavior for this axis when reference switch is active during a normal movement. This parameter can be defined for each axis. Each axis can be configured differently.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>Motion stops if reference switch is active.</td></tr>
<tr><td>1</td><td>Motion does not stop if reference switch is active (for normal movements).</td></tr>
<tr><td colspan="2">The behavior of the homing process will not change due to this parameter. The default value (if not defined) for this parameter is '0'.</td></tr>
</table>

<table>
<tr><td rowspan="5">ForceSingleAxisMovement</td><td colspan="2">Forces all defined axes to always move individually. This parameter is a global parameter for all axes.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>No single axes movement forcing. Axes can move simultaneously under certain conditions.</td></tr>
<tr><td>1</td><td>All defined axes will move individually. No simultaneous movement of axes will occur.</td></tr>
<tr><td colspan="2">Forcing of single axis movement can sometimes be desired for 2D splitting where usually individual axis speed is overwritten by the global splitting speed resulting in simultaneous movement.</td></tr>
</table>

Example sc_motion_stepper_settings.txt file for 1 straight and 1 rotational axis for 'Type=14':

```
# Global parameters:
Debug=0
DisableHomingDuringStartUp=1

# Z-axis parameters:
axis=0
dname=Z
mode=POSITION

factor=200.0
llimit=-400000
hlimit=400000
hslimit=6000
defspeed=20.0
accel=2000
decel=2000
accStartSpeed=5.0
decStopSpeed=5.0

stepIO=216
dirIO=217
dirvalue=0

refIO=16
refvalue=1
refmode=0
refspeed=2000
refspeed2=500
refpos=0.0
RefOnlyForHome=1

# R-axis parameters:
axis=1
dname=R
mode=ANGLE

factor=10.0
incperrot=3600
llimit=-1E13
hlimit=1E13
hslimit=6000
defspeed=400.0
accel=2000
decel=2000
accStartSpeed=5.0
decStopSpeed=5.0

stepIO=218
dirIO=219
dirvalue=0

refIO=17
refvalue=1
refmode=0
refspeed=2000
refspeed2=500
refpos=0.0
RefOnlyForHome=1
```

GCode cmd	Description
GCL M0	Stop execution (aborts moves).
GCL M46	Starts execution. This is required after GCL M0 for further motion commands.
GCL F?	Set movement speed to ? mm/s (e.g. GCL F10.0).
GCL G1??	Move to ??, where ?? is axis identifier (X, Y, Z, A, B, C) followed by position in mm (e.g. G1X1.5000). Multiple axis can be combined (e.g. G1X1Y2.5Z0)

GCode cmd	Description
GCL G90	Switch to absolute mode. After this all move coordinates are interpreted as absolute values (default).
GCL G91	Switch to relative mode. G1 commands are interpreted relative to previous position.
GCL G28?1	Home axis ?, where axis is X, Y, Z, A, B, C. Only one axis at a time.
GCL Q????	Get current axis position. ???? can be 7032-7037, for axis X, Y, Z, A, B, C respectively.

Table 8: Stepper GCode commands for a USC-2 in Flash mode

Necessary steps for changing from stepper type 8 to stepper type 14:

- The parameters 'sfactor', 'SignalAxisMoving', 'SignalAxisMovingStateBitPosition', 'SignalAxisMovingState', 'SignalAxisMovingStatePreDelay', 'MoveThisAxisFirst' and 'corr#' are not supported in the stepper type 14 any more.
- Optionally you can define the parameter 'RefOnlyForHome'. If you define 'RefOnlyForHome=1' you will have the behavior like you are used to from stepper mode 8.
- The units for 'accel' and 'decel' have been changed to steps/s². This leads to a linear acceleration and deceleration of the speed of the axes. You need to adjust the values for these parameters as well. In stepper type 14 the values for these parameters will be much higher. The values '0' and '-1' are not valid in stepper type 14.
- The 'refmode's 3 and 6 are no longer supported in stepper type 14.
- In stepper type 14 the X, Y and Z axes can move in parallel so please do not use the same value for different axes for the parameter 'refIO'. Do not use external stop ('refIO=1') as well because fast homing mode is not necessary in stepper mode 14 any more.

5.4.2.1 Motion Settings Dialog Type 14

This dialog can be found in SAMLight → Settings → Extras → Motion Settings Dialog.

Here you can configure the motor as described in the file <SCAPS>\system\sc_motion_stepper_settings.txt.

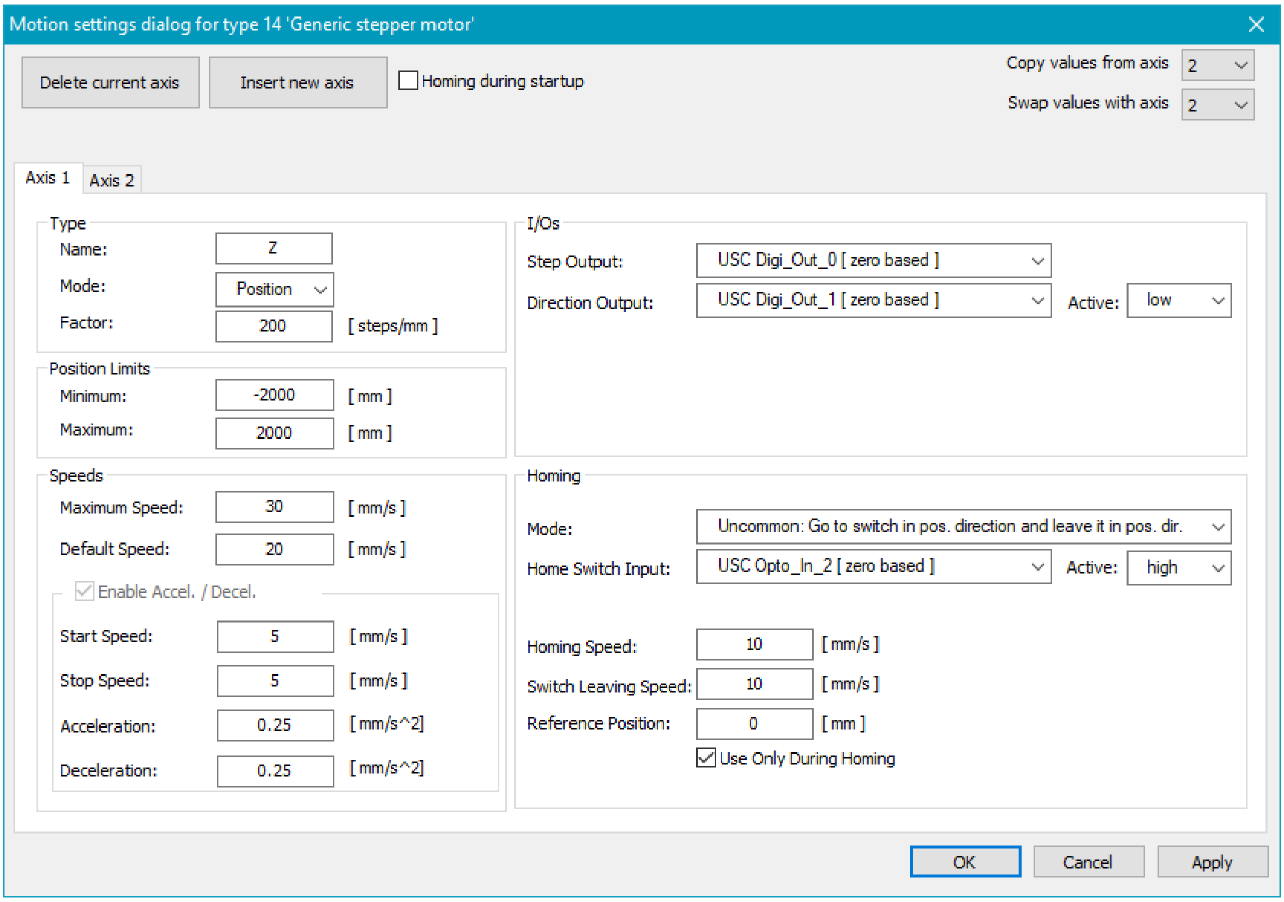


Figure 41: Direct motion control Home / Shift dialog

Homing during startup: Enable automatic homing after the SAMLight startup.

Delete current axis: Delete the currently opened axis tab.

Insert new axis: Add a new axis (max. 6 for Motion Type 14) at the end of the list

Copy values from axis: Choose an other axis in the drop down menu and copy all values in the current selected axis tab.

Swap values with axis: Choose an other axis in the drop down menu and swap all values between the selected and the current axis tab.

Type:

> **Name:** Sets name of axis. Has to be an unique single capital letter for each axis.
>
> **Mode:** Position[mm] as straight axis or Angle[°] as rotational axis can be chosen.
>
> **Factor:** For 'mode=POSITION': Unit: [steps/mm]. For 'mode=ANGLE': Unit: [steps/ °], value has to be (1/360°)*'incperrot' Converts mm (or °) used in SAMLight into steps.

Position limits: Both values are read as double to increase the value range [-1E13, 1E13]. This limits should not contain 'refpos'.

Minimum: This parameters defines the lower limit of the axis.

Maximum: This parameters defines the upper limit of the axis.

Speeds: Unit depends on chosen axis. For planar Unit: [mm/sec] and for roation Unit: [°/sec]

Maximum speed: Define the maximum moving speed.

Default speed: Sets default speed displayed in SAMLight user interface.

Enable Accel./Decel.: The checkbox is always enabled.

Start speed: Specify the speed, where the motor should start to accelerate.

Stop speed: Specify the speed, where the motor should end to decelerate.

Acceleration: This value defines the acceleration at the start of the motor. A value of '0' or '-1' disables acceleration /deceleration.

Deceleration: This value defines the deceleration at the start of the motor. A value of '0' or '-1' disables acceleration /deceleration.

NOTE

When SAMLight Units are set to inch, the units in the motion settings dialog are set to inch as well, save for the acceleration and deceleration. Here, the units are still mm/s^2!

I/Os:

Step output: Select the output bit for the step signals by the drop down menu.

Direction output: Select the output bit for the direction signals by the drop down menu.

active: Choose if the direction output bit state should be active high or active low.

Homing: This mode defines the behavior of homing.

Mode: Choose the required homing mode in the drop down menu. Homing is set off per default.

1. No homing movement, current position is set to 'Reference position'.
2. Go to switch in neg. direction and leave it in pos. dir.
3. Go to switch in pos. direction and leave it in neg. dir.
4. Uncommon: Go to switch in neg. direction and leave it in neg. dir.
5. Uncommon: Go to switch in pos. direction and leave it in pos. dir.
6. Uncommon: Go to switch in neg. sirection and leave it in pos. dir. (no timeout)
7. Uncommon: Go to switch in pos. direction and leave it in neg. dir. (no timeout)

Home switch input: Select the input bit for the home switch input signals by the drop down menu.

active: Choose if the input bit state should be active high or active low.

Homing speed: Specify the speed during the homing process.

Switch leaving speed: Specify the switch leaving speed.

Reference position: Set a reference position for the homing.

Use only during homing: If enabled, the reference position will be only used during the homing.

5.4.3 Stepper I/O parameters

Here you can find all available I/O parameters and the corresponding pin assignment for the motion type 14 (USC-2 stepper controller) and type 8 (generic stepper controller) for the sc_motion_stepper_settings.txt file.

USC cards:

Input pin	Stepper settings parameter	Output pin	Stepper settings parameter
Opto-insulated Inputs		**Opto-insulated Outputs**	
Opto_In_0	Reserved for trigger start	Opto_Out_0	Reserved for marking active
Opto_In_1	Reserved for external stop	Opto_Out_1	1
Opto_In_2	2	Opto_Out_2	2, if not used for red pointer
Opto_In_3	3	Opto_Out_3	3
Opto_In_4	4	Opto_Out_4	4
Opto_In_5	5	Opto_Out_5	5
		Laser Ports, **if not used for laser control**	
		LP_0	100
		LP_1	101
		LP_2	102
		LP_3	103
		LP_4	104
		LP_5	105
		LP_6	106
		LP_7	107
Following pins are not available for USC-1			
Digital Inputs		**Digital Outputs**	
Digi_In_0	6	Digi_Out_0	206
Digi_In_1	7	Digi_Out_1	207
Digi_In_2	8	Digi_Out_2	208
Digi_In_3	9	Digi_Out_3	209
Digi_In_4	10	Digi_Out_4	210
Digi_In_5	11	Digi_Out_5	211
Digi_In_6	12	Digi_Out_6	212
Digi_In_7	13	Digi_Out_7	213
Digi_In_8	14	Digi_Out_8	214
Digi_In_9	15	Digi_Out_9	215
Stepper inputs		**Stepper Outputs**	
Stepper_In_0	16	Stepper_Out_0	216
Stepper_In_1	17	Stepper_Out_1	217
Stepper_In_2	18	Stepper_Out_2	218
		Stepper_Out_3	219
		Stepper_Out_4	220
		Stepper_Out_5	221

Table 9: Stepper I/O parameters for USC cards

RTC cards:

Input pin	Stepper settings parameter	Output pin	Stepper settings parameter
Digital Inputs		Digital Outputs	
Digi_In_0	0	Digi_Out_0	0
Digi_In_1	1	Digi_Out_1	1
Digi_In_2	2	Digi_Out_2	2
Digi_In_3	3	Digi_Out_3	3
Digi_In_4	4	Digi_Out_4	4
Digi_In_5	5	Digi_Out_5	5
Digi_In_6	6	Digi_Out_6	6
Digi_In_7	7	Digi_Out_7	7
Digi_In_8	8	Digi_Out_8	8
Digi_In_9	9	Digi_Out_9	9
Digi_In_10	10	Digi_Out_10	10
Digi_In_11	11	Digi_Out_11	11
Digi_In_12	12	Digi_Out_12	12
Digi_In_13	13	Digi_Out_13	13
Digi_In_14	14	Digi_Out_14	14
Digi_In_15	15	Digi_Out_15	15

Table 10: Stepper I/O parameters for RTC cards

5.5 Other motion controller

The following motion controllers (RS-232, CAN, etc.) are supported.

5.5.1 Type 1 - IMS Stepper Drives

After leaving the program an *ims_settings.txt* file is created in the folder *<SCAPS>\system* where controller settings can be done in case it is configured for the Ims motion controller.

For example:

ComPort = 1	**serial interface number, alternatively: "ComPort = USC1" (CH_O)**
ComSettings = 9600,N,8,1	baud rate, parity, word length, stop bits, flow control
PartyMode = 1	drive controller in party mode (for multiple axis) → With the current version the party mode needs to be used.
EncoderMode = 0	Read back encoder values (0/1)

For each axis the following settings need to be done.

AxisName: Definition of the name of the axis (one letter) where the commands are being sent to.

AxisMode: Either angle or pos. Sets the units in the user interface.

AxisScale: Number of increments per unit. Taken for pos mode.

AxisIncPerRot: Number of increments for one rotation, relevant for angle mode.

EncoderAxis: Defines if the motor has an encoder, values: *0* or *1*.

SpeedScale: Factor for speed. In case the motor drives another wheel this factor is needed for the wheel to achieve the entered speed. The default value is *1*.

MinSpeed: Minimum Speed in steps per second.

MaxSpeed: Maximum Speed in steps per second.

DefaultSpeed: Default Speed in steps per second.

NOTE

Pressing the Home button in the control motion property page the "G 0" command is sent to comport.
If Auto Variable Resolution mode is used the speed will be the entered speed divided with the resolution factor.

5.5.2 Type 4 - External custom controller

SAMLight supports an interface for a customized implementation of a motion controller *.dll. A sample package is available on our homepage (http:\\www.scaps.com) or on request. The name of the external motion controller is sc_motion_control_ext_<DN>.dll where <DN> corresponds to the defined controller name in sc_motion_settings.txt. The *.dll has to be located in the same directory as sam_light.exe or in C:\Windows\System32\

Motion controller type: The external custom controller and the device name needs to be defined in the text file **sc_motion_settings.txt**. This file can be found in the folder <SCAPS>\system\.

Type=4 Sets the type of the motion controller to the external custom controller.

DeviceName=<DN> Specifies the name of the device, which determines the name of the *.dll that is loaded during runtime to access the motion controller functionality (see above).

5.5.3 Type 5 - IMS MDrive

Key features

- Available for USC and RTC controller cards
- Up to 7 axes can be controlled
- Axes operate in party mode and move at the same time (without synchronization)
- The MDrive motor is controlled via a serial port, e.g. COM port of the PC or the RS-232 interface of an USC controller card
- Homing procedures available

Required settings

- Before SAMLight can communicate and control the MDrive motor you need to define some parameters directly at your MDrive motor (e.g. via IMS Terminal or Windows HyperTerminal, refer to 'Hardware settings (type 5)' described below).
- Control all MDrives in IMS Terminal or Windows HyperTerminal before you try to control them by SAMLight to make sure the settings of the MDrives are correctly. Use the same settings for the HyperTerminal as for SAMLight ('PortName', 'PortParity' and 'PortBaudRate').
- Set 'Type=5' in sc_motion_settings.txt in <*SCAPS*>\system\
- Define sc_motion_mdrive_settings.txt in <*SCAPS*>\system\ like described below.
- Enable 'Motion Control' in SAMLight (Settings→System→Extras).

How to create the settings file

- The file 'sc_motion_mdrive_settings.txt' is a plain text file that contains different configuration parameters.
- A line which begins with a # sign is interpreted as a comment and will be ignored.
- All parameters have to start exactly at the beginning of a line.
- Use integers for every parameter value, decimal points will be ignored.
- Global parameters have to be specified only once at the head of the settings file.
- Axis-specific parameters have to be defined for all axes separately.
- The axis-specific and the optional parameter have to be arranged in one block of parameters per axis. Each block begins with the parameter 'axis'.
- The optional parameters do not have to be specified in the case you do not want to use them.
- A parameter gets defined by typing <parameter>=<value>, parameter and values are described below,
  e.g. 'axis=0' defines the parameter 'axis' with the value '0'

Hardware settings (type 5): These settings have to be defined directly at every MDrive motor.

WARNING

Connect only one motor for the hardware settings to the COM port and repeat this steps for each MDrive motor.

Necessary steps for every motor before SAMLight can control MDrive motors.	
Before SAMLight can communicate and control the MDrive motor you need to define the following parameters directly at your MDrive motor (e.g. via IMS Terminal or Windows HyperTerminal). The actual MDrive commands may vary a bit compared to the commands you find here. For further information see the MDrive manual.	
Suggested MDrive command	Function
PR AL	Prints all MDrive values: Echoes all motor parameters. While party mode is disabled no prefix '?' is used.
DN=?	Defines device name: '?' has to be a single capital character. Use a unique value '?' for each MDrive motor. E.g. 'DN=X' (if party mode is disabled)
PY=1	Enables party mode: Party mode is necessary to address the commands from SAMLight to the right MDrive motor, even you want to use only one motor.
If the MDrive is in party mode you need to specify every command with the prefix '?' which corresponds to the device name. This is necessary to address commands to the right device because all motors communicate through a single COM port.	
?IP	Initializes parameters for the '?'-axis: Discards all temporary changes and applies the stored values of the MDrive motor.
?EM=2	Disables Echo mode for the '?'-axis: Suppresses echoing of almost all MDrive commands.
?DE=1	Enable the motor driver for the '?'-axis.
?S	Saves settings for the '?'-axis: E.g. 'XS' (if party mode is enabled) E.g. 'S' (if party mode is disabled)
?PR AL	Prints all MDrive values for the '?'-axis: Echoes all parameters stored directly at the motor.

Global parameters (type 5): These parameters have to be defined once at the head of the settings file.

PortName	Defines the serial port used for the MDrive motors.	
	Value	Function
	COM#	Uses a COM port of the PC where '#' is the port number (e.g. 'COM1').
	USC-1	Uses the RS-232 interface of the USC card at the 37-pin connector (CH_0). For further information please refer to the corresponding USC manual.
	USC-2	

PortParity	Defines the parity mode for the COM port.	
	Value	Function
	0	No parity
	1	Odd parity
	2	Even parity
	3	Mark parity
	4	Space parity
	The default value that is used by most IMS MDrives is '0'.	

PortBaudRate	Defines the data rate for the COM port.	Unit: Bd (bit/s)
	The default value that is used by most IMS MDrives is '9600'.	

TimeOut	Sets the connection timeout.	Unit: s
	Sets how long it shall tries to access the communication between interface and the motion controller. If you have connection problems the debug mode (see below) could be helpful.	

Debug	Enables the debug log.	
	Value	Function
	0	Disable debug mode
	1	Enable debug mode
	Enables debug mode to log debug information in <*SCAPS*>\system\sc_motion_mdrive_debuglog.txt'. Only enable this option if you need it, because there can be huge amounts of data that are logged into this file.	

DisableHomingDuringStartUp	No homing on startup.	
	Value	Function
	0	Enable homing on startup of SAMLight
	1	Disable homing on startup of SAMLight
	If set to '1', homing is just performed when pressing 'Control→Home' and not when software is started.	

RtsControl	Enables or disables RTC (Request to Send) for the RS232 connection for MDrive's RS232-USB-adapter if used.	
	Value	Function
	0	set RTS_CONTROL_DISABLE
	1	set RTS_CONTROL_ENABLE
	If this parameter is not set the internal default state '1' is used. This parameter is available starting with installer 3.7.5 Build 0033.	

CheckForProgramRunning	Enables or disables checking for a MDrive subroutine after homing.	
	Value	Function
	0	Disable CheckForProgramRunning
	1	Enable CheckForProgramRunning
	If this parameter is not set the internal default state is '1'. In case after homing an own subroutine is used that returns a '1' for 'PR BY' (Print Busy), CheckForProgramRunning should be set to '0'. This parameter is available starting with installer 3.7.5 Build 0035.	

Axis parameters (type 5): All of the following parameters has to be defined for each axis after the global parameters in the settings file. Each of these axes can be configured differently.

axis	Sets the number of axis.	
	Value	Function
	0...6	Index of axis
	Zero based index of up to 7 axes. Defines the display order in SAMLight. This parameter has to be at the beginning of each axis specific parameter block.	

dname	Sets the name of axis.	
	Value	Function
	A...Z	Name of axis
	Has to correspond to the device name 'DN' which is stored directly at the MDrive motor. (Refer to 'Hardware settings (type 5)'). Has to be an unique single character for each MDrive motor.	

mode	Sets the type of the axis.	
	Value	Function
	POSITION	Defines a straight axis [mm].
	ANGLE	Defines a rotational axis [°].
	Depending on the mode of the axis several parameters has to be adjusted (corr#, incperrot, sfactor, llimit, hlimit, defspeed, see below).	

corr1 -1000 -20000000 corr2 -500 -10000000 corr3 0 0 corr4 500 10000000 corr5 1000 20000000 **<corr#> <mm#> <steps#>**	Correction table converts mm into steps and defines the range of the axis for POSITION mode.
	These five parameters are defined without '=' sign in the form <corr#> <mm#> <steps#>
	Setup for linear corr table: The range of the axis in mm corresponds to the difference of the max. value <mm5> and min. value <mm1>. The other mm-values have to be linearized. The step-values can be calculated by: **<steps#> = <mm#> * factor** The factor can be determined experimentally if not known. Non-linear corrections: This table can be used to compensate non-linearities of a straight axis by adjusting the values. For ANGLE mode these parameters are ignored, the parameter 'incperrot' is used instead.

incperrot	Converts degrees into steps for ANGLE mode.	Unit: steps/360°
	This value defines how many increments are needed for a whole rotation. The default value is 51200 and depends on the MDrive setup 'MS' (micro steps). It can be calculated by **200 * 'MS'**. For POSITION mode this parameter is ignored, the <corr#> table is used instead.	

sfactor	Sets speed factor.	
	POSITION mode, unit: steps/mm	ANGLE mode, unit: steps/°
	Value has to be equal to the factor of the corr table for a linear corr table.	Value has to be equal to **(1/360)*'incperrot'**.
	Converts mm/s (or °/s) used in SAMLight into steps/s.	

llimit	Sets the lower limit for the axis.	Unit: steps
	No movement below this limit will be possible. This value is used for POSITION and ANGLE mode and has to be equal to the **<steps1>** value of the corr table.	

hlimit	Sets the upper limit for the axis.	Unit: steps
	No movement above this limit will be possible. This value is used for POSITION and ANGLE mode and has to be equal to the **<steps5>** value of the corr table.	

hslimit	Sets the upper speed limit.	Unit: steps/s
	No speed above this speed limit will be possible. To get the 'hslimit' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'sfactor'.	

defspeed	Sets default speed displayed in SAMLight user interface.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be smaller or equal than 'hslimit'/'sfactor'.	

Optional axis parameters (type 5): All of the following optional parameters can be defined for each axis. Each axis can be configured differently. If no parameter is specified the corresponding feature will not be used.

invertDir	Defines the motion direction.	
	Value	Function
	0	Default rotation
	1	Inverted rotation

ival	Sets initialization parameters.
	On startup of SAMLight every axis can be initialized with this ival parameter. This parameter can exist more than once and will be executed in the same order as they appear in the settings file. Each parameter corresponds to any MDrive command the drive supports.
	Recommended initialization parameters (refer to your MDrive manual for further information): ival=IP ival=EM 2 ival=DE

<table>
<tr><td rowspan="4">hval</td><td colspan="2">Sets homing parameters.</td></tr>
<tr><td colspan="2">With hval parameters MDrive commands a homing procedure for each axis can be defined. This parameter can exist more than once and will be executed in the same order as they appear in the settings file. The homing procedure will be executed on startup of SAMLight and by a manual homing. The homing procedure at startup of SAMLight can be disabled with global parameter 'DisableHomingDuringStartUp'.</td></tr>
<tr><td colspan="2">Suggested Mdrive commands for programming a homing procedure (for further information refer to MDrive manual):
'PG', 'HI', 'HM', 'P', 'S1'</td></tr>
<tr><td>Example 1:

No homing movement

Current position is set to <position> in steps.

hval=P <position></td><td>Example 2:

Referencing using a home switch

I/O number one is used for the home switch (low active); the drive is moving into negative direction with <speed> in steps/s until that input goes to low triggered by a home switch. Afterwards the current position is set to <position> in steps.

hval=PG 100
hval=VM <speed>
hval=S1 1,0
hval=HM 1
hval=H
hval=P <position>
hval=E
hval=PG
hval=EX 100</td></tr>
</table>

Example sc_motion_mdrive_settings.txt file for 1 straight and 1 rotational axis for 'Type=5':

```
# Global parameters:
PortName=COM1
PortParity=0
PortBaudRate=9600
TimeOut=10
Debug=0

# X-axis parameters:
axis=0
dname=X
mode=POSITION
corr1 -1000 -20000000
corr2  -500 -10000000
corr3     0         0
corr4   500  10000000
corr5  1000  20000000
sfactor=20000
llimit=-20000000
hlimit=20000000
hslimit=500000
defspeed=25
invertDir=0

ival=IP
ival=EM 2
ival=DE 1
```

```
hval=P 0

# R-axis parameters:
axis=1
dname=R
mode=ANGLE
incperrot=51200
sfactor=142
hslimit=500000
defspeed=250
invertDir=0

ival=IP
ival=EM 2
ival=DE 1

hval=P 0
```

5.5.4 Type 6 - Faulhaber motion controller

The Faulhaber motion controller series are supported directly. To enable it, please configure the Faulhaber type 6 like it was described above.

For configuring the Faulhaber controller interface, a configuration file *sc_motion_faulmc_settings.txt* is required in the same location where the general motion settings file exists. There can be used one motion controller of this type at the same time, it is configured for the Z-axis fixed and can be used to perform positional changes. The Faulhaber configuration file can be used to specify different parameters like the COM-port, the data rate and others more. The file itself is a plain text file that contains different statements. Additionally there can be comments within the file that begin with a # sign directly at the beginning of a line. These comments are ignored completely. Following parameters, that have to start exactly at the beginning of a line, are supported for configuring the interface.

PortName=xxx : Specifies to which port the Faulhaber controller is connected to, here for *"xxx"* e.g. COM1 has to be set.

PortBaudRate=yyyy : Defines the data rate (in bps) the COM port has to work with.

corr1 mm inc **corr2 mm inc** **corr3 mm inc** **corr4 mm inc** **corr5 mm inc**	The *corr* commands define a conversion and correction table from metric positions to incremental positions of the used Z-axis. This table consists of 5 entries where *corr1* defines the smallest possible value and *corr5* the biggest one. Here more than only a factor is given to allow it a user to equalize non-linear variances. The syntax of the *corr* table commands requires a metric value *mm* in millimetres and the appropriate incremental value *inc* that is equal to this metric position.

NOTE *The incremental value of corr1 must be smaller or equal than the parameter llimit and the incremental value of corr5 must be equal or bigger than the value of parameter hvalue that are described below.*

llimit=yyy : This parameter defines the lower limit the motion controller can drive to with the specified axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is smaller than the one set here.

hlimit=yyy : This parameter defines the higher limit the motion controller can drive to using the current axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is bigger than the one set here. So with *llimit* and *hlimit* a range can be defined where the motion controller is allowed to work within.

5.5.5 Type 7 - isel IT Stepper Controller / DNC

Key features

- Available for USC and RTC controller cards
- Up to 3 axes can be controlled
- Axes can move synchronized
- The motion controller is connected via a COM port of the PC
- Homing procedures are available

Required settings

- Set 'Type=7' in sc_motion_settings.txt in <*SCAPS*>\system\
- Define sc_motion_iselit_settings.txt in <*SCAPS*>\system\ like described below.
- Enable 'Motion Control' in SAMLight (Settings→System→Extras).

How to create the settings file

- The file sc_motion_iselit_settings.txt' is a plain text file that contains different configuration parameters.
- A line which begins with a # sign is interpreted as a comment and will be ignored.
- All parameters have to start exactly at the beginning of a line.
- Global parameters have to be specified only once at the head of the settings file.
- Axis-specific parameters have to be defined for all axes separately.
- The axis-specific and the optional parameters have to be arranged in one block of parameters per axis. Each block begins with the parameter 'axis'.
- The optional parameters do not have to be specified in the case you do not want to use them.
- A parameter gets defined by typing <parameter>=<value>, parameter and values are described below,
  e.g. 'axis=0' defines the parameter 'axis' with the value '0'.

Global parameters (type 7): These parameters have to be defined once at the head of the settings file.

PortName	Defines the serial port used for the isel it controller.	
	Value	Function
	COM#	Defines the COM port of the PC where '#' is the port number (e.g. 'COM1'). The USC RS232 port cannot be used for isel it controllers.

PortBaudRate	Defines the data rate for the COM port.	Unit: Bd (bit/s)
	The default value that is used by most isel controllers is '9600'.	

TimeOut	Sets the connection timeout.	Unit: s
	Sets how long it shall tries to access the communication between interface and the motion controller. If you have connection problems the debug mode (see below) could be helpful.	

<table>
<tr><td rowspan="4">DeviceNumber</td><td colspan="2">Defines the serial port used for the isel it controller.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>#</td><td>The device number specifies which specific controller has to be accessed. At the controller hardware that number can be changed with the DNC command @<DN>G<DNnew>. The default value is '0'.</td></tr>
</table>

<table>
<tr><td rowspan="5">Debug</td><td colspan="2">Enables the debug log.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>Disable debug mode</td></tr>
<tr><td>1</td><td>Enable debug mode</td></tr>
<tr><td colspan="2">Enables debug mode to log debug information in <<i>SCAPS</i>>\system\sc_motion_iselit_debuglog.txt'. Only enable this option if you need it, because there can be huge amounts of data that are logged into this file.</td></tr>
</table>

<table>
<tr><td rowspan="5">DisableHomingDuringStartUp</td><td colspan="2">No homing on startup.</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>Enable homing on startup of SAMLight</td></tr>
<tr><td>1</td><td>Disable homing on startup of SAMLight</td></tr>
<tr><td colspan="2">If set to '1', homing is just performed when pressing 'Control→Home' and not when software is started.</td></tr>
</table>

<table>
<tr><td rowspan="5">switch1</td><td colspan="2">Enables the home switch for 'axis=0' for older motion controller</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>Disable home switch</td></tr>
<tr><td>1</td><td>Enable home switch</td></tr>
<tr><td colspan="2">This parameter is only valid for older motion controller, which support the "IE" command. Set the value to '0' for a newer motion controller and define the homing procedure with 'hval' commands.</td></tr>
</table>

<table>
<tr><td rowspan="5">switch2</td><td colspan="2">Enables the home switch for 'axis=1' for older motion controller</td></tr>
<tr><td>Value</td><td>Function</td></tr>
<tr><td>0</td><td>Disable home switch</td></tr>
<tr><td>1</td><td>Enable home switch</td></tr>
<tr><td colspan="2">This parameter is only valid for older motion controller, which support the "IE" command. Set the value to '0' for a newer motion controller and define the homing procedure with 'hval' commands.</td></tr>
</table>

level1	Sets the polarity of the home switch for 'axis=0'	
	Value	Function
	0	low active
	1	high active
	This parameter is only valid for older motion controller, which support the "IE" command.	

level2	Sets the polarity of the home switch for 'axis=1'	
	Value	Function
	0	low active
	1	high active
	This parameter is only valid for older motion controller, which support the "IE" command.	

Axis parameters (type 7): All of the following parameters have to be defined for each axis after the global parameters in the settings file. Each of these axes can be configured differently.

axis	Sets the number of axis.	
	Value	Function
	0...2	Index of axis
	Zero based index of up to 3 axes. Defines the display order in SAMLight. This parameter has to be at the beginning of each axis specific parameter block. The axes are named x, y and z automatically. Start with index '0' for the first axis.	

mode	Sets the type of the axis.	
	Value	Function
	POSITION	Defines a straight axis [mm].
	ANGLE	Defines a rotational axis [°].
	Depending on the mode of the axis several parameters has to be adjusted (factor, corr#, incperrot, sfactor, llimit, hlimit, defspeed, see below).	

factor	Converts mm into steps for POSITION mode.	Unit: steps/mm
	This value defines how many increments are needed for 1mm. Depending on the factor of the axis several parameters has to be adjusted (sfactor, defspeed, see below). This parameter is only valid for SAMLight versions >= 3_3_5_0224. With older SAMLight versions the corr table has to be used.	

incperrot	Converts degrees into steps for ANGLE mode.	Unit: steps/360°
	This value defines how many increments are needed for a whole rotation. Depending on the factor of the axis several parameters has to be adjusted (sfactor, defspeed, see below).	

sfactor	Sets speed factor.	
	POSITION mode, unit: steps/mm	ANGLE mode, unit: steps/°
	Value has to be equal to **'factor'**.	Value has to be equal to **'incperrot/360'**.
	Converts mm/s (or °/s) used in SAMLight into steps/s.	

llimit	Sets the lower position limit.	Unit: steps
	No movement below this limit will be possible. Minimum value is '-8388606'.	

hlimit	Sets the upper position limit .	Unit: steps
	No movement above this limit will be possible. Maximum value is '8388607'.	

hslimit	Sets the upper speed limit.	Unit: steps/s
	No speed above this speed limit will be possible. To get the 'hslimit' in steps/s like it is required here the value in mm/s (or °/s) has to be multiplied by the 'sfactor'.	

defspeed	Sets default speed displayed in SAMLight user interface.	
	POSITION mode, unit: mm/s	ANGLE mode, unit: °/s
	This value has to be smaller or equal than 'hslimit/sfactor'.	

Optional axis parameters (type 7): All of the following optional parameters can be defined for each axis. Each axis can be configured differently. If no parameter is specified the corresponding feature will not be used.

`corr1 -8000.0 -400000.0`
`corr2 -4000.0 -200000.0`
`corr3 0.0 0.0`
`corr4 4000.0 200000.0`
`corr5 8000.0 400000.0`
 	
<corr#> <mm#> <steps#>
	Correction table instead of parameter 'factor' for POSITION mode.

	This parameter is defined without '=' sign in the form <corr#> <mm> <steps>.
	This correction table can be used to compensate non-linearities of a straight axis. If 'factor' is defined, the correction table is ignored. Use for corr1 the same step value as for 'llimit' and for corr5 the same step value as for 'hlimit'.

<table>
<tr><td rowspan="4">hval</td><td colspan="2">Sets homing parameters.</td></tr>
<tr><td colspan="2">With hval parameters a homing procedure for each axis can be defined. This parameter can exist more than once and will be executed in the same order as they appear in the settings file. The homing procedure will be executed on startup of SAMLight and by a manual homing.</td></tr>
<tr><td colspan="2">Refer to the isel manual for the commands.</td></tr>
<tr><td>Example 1, no homing, set current position of 3 axes to 0:

hval=n7</td><td>Example 2, homing of 3 axes:

hval=d<xSpeed>, <ySpeed>, <zSpeed>
hval=R7</td></tr>
</table>

Example sc_motion_iselit_settings.txt file for 1 straight and 1 rotational axis for 'Type=7':

```
# Global parameters:
PortName=COM1
PortBaudRate=9600
TimeOut=30
DeviceNumber=0
Debug=0
DisableHomingDuringStartUp=1
switch1=0
switch2=0
level1=0
level2=0

# x axis parameters:
axis=0
mode=POSITION
factor=50.0
sfactor=50.0
llimit=-8388606.0
hlimit=8388607.0
hslimit=10000
defspeed=100.0
hval=n1

# y axis parameters:
axis=1
mode=ANGLE
incperrot=18000
sfactor=50.0
llimit=-8388606.0
hlimit=8388607.0
hslimit=10000
defspeed=180.0
hval=n2
```

5.5.6 Type 9 - Generic RS-232 interface

This special interface can be used to access devices by sending plain strings to them via RS-232. Controlling the connected device is done exclusively by sending RS-232 strings. There is no axis enabled within the motion control property pane. To configure the generic RS-232 interface:

- Set 'Type=14' in sc_motion_settings.txt in <SCAPS>\system\.
- Define sc_motion_generic232_settings.txt in <SCAPS>\system\ as described below.
  - This file is a plain text file that contains different configuration parameters.
  - A line which begins with a # sign is interpreted as a comment and will be ignored.
  - All parameters have to start exactly at the beginning of a line.
  - A parameter is defined by typing <parameter>=<value>, parameter and values are described below, e.g. 'PortName=COM1' defines the parameter 'PortName' with the value 'COM1'.

PortName	Defines the RS-232 COM port.	
	Value	Function
	COM#	COM port of the PC where # is the port number (e.g. 'COM1'). # must be in the range of [1, 9].
	USC-1	Uses the RS-232 interface of the USC 37-pin connector.

PortBaudRate	Sets the data rate.
	Unit: [Bd] (bit/s)

PortByteSize	Sets the number of data bits.
	Defines how many bits are transmitted per character.

PortParity	Sets the parity mode.	
	Value	Function
	0	No parity
	1	Odd parity
	2	Even parity
	3	Mark parity
	4	Space parity

PortStopbits	Sets the number of stop bits.
	The following values are supported: 1, 1.5, 2.

SendEolValue	Use a custom ASCII end character EolValue.	
	Value	Function
	0	End character is carriage return and line feed.
	1	End character is *EolValue*.

EolValue	Specify custom end of line control character
	The value must be the ASCII character in decimal representation and in the range of [0, 255]. For example '13' is carriage return. *SendEolValue* must be '1' to use this end character.

TimeOut	Sets a timeout for sending the string.
	Unit: [s]. If sending takes longer than the timeout, it is aborted.

Debug	Enables debug mode.	
	Value	Function
	0	Enables debug mode.
	1	Disables debug mode.
	Debug log can be found in *%SCAPS_SAM %/system/sc_motion_generic232_debuglog.txt*. There can be huge amounts of data that are logged into such a file so enable this option only if it is necessary.	

ival	Sends initialization on SAMLight startup.
	On startup of SAMLight the motion controller can be initialized with this parameter. This parameter can exist more than once and will be executed in the same order as they appear in the settings file.

Example sc_motion_generic232_settings.txt file:

```
PortName=COM1
PortBaudRate=9600
PortByteSize=8
PortParity=0
PortStopbits=1
SendEolValue=0
EolValue=13
TimeOut=10
Debug=0
```

5.5.7 Type 10 - SHS 2000 Star

The SHS Star 2000 motion controller series is supported directly. To enable it, please configure the drive type number 10 like it was described above.

For configuring the SHS controller interface, a configuration file *sc_motion_shstar2000_settings.txt* is required in the same location where the general motion settings file exists. The SHS Star 2000 configuration file can be used to specify different parameters like the COM port, the data rate, the initialization and others more. The file itself is a plain text file that contains different statements. Additionally there can be comments within the file that begin with a # sign directly at the beginning of a line. These comments are ignored completely. Following parameters, that have to start exactly at the beginning of a line, are supported for configuring the interface:

PortName=xxx : Specifies to which port the SHS controller is connected to, here for *"xxx"* e.g. COM1 has to be set.

PortBaudRate=yyyy : Defines the data rate (in bps) the COM port has to work with. The default value that is used by most SHS controllers is 19200.

PortParity=y : This parameter changes the parity mode for the port, here for *y* following values are supported:

- 0 - No parity
- 1 - Odd parity
- 2 - Even parity
- 3 - Mark parity
- 4 - Space parity

PortRTS=y : Using this parameter the ready to send behaviour can be configured like it is necessary for some serial protocols. Here following values are possible for *y*:

- 0 - disable RTS fully
- 1 - enable standard RTS mode
- 2 - enable RTS handshake mode
- 3 - enable RTS in control toggle mode

TimeOut=yy : The *TimeOut* value specifies how long the interface shall retry to access the communication interface until it fails with a time out in case such an operation is not successful.

Debug=y : For special debugging purposes it is possible to log several information into a file *sc_motion_shstar2000_debuglog.txt* that will be created at the position where the settings file is located at. With the value *1* the debugging mode is turned on, when it is set to *0* debugging is disabled and no data are put into the log file.

NOTE

There can be huge amounts of data that are logged into such a file so enable this option only if it is necessary.

All these values are global ones and therefore they are valid for everything that is controlled by that motion controller. This is important for the multi-axes-mode where it is possible to handle up to five axes using this controller interface. Each of these axes can be configured in a different way. To do that, the *axis* statement that defines for which *axis* the following parameters are valid has to be use. In the worst case when three axes have to be used with completely different configurations all the following parameters have to exist five times, separated by the different calls of *axis*:

axis=y : Specifies for which axis the following configuration parameters are valid for. If this statement is not used somewhere in a configuration file it is assumed that the third axis is used. Else a value out of the allowed range 0..4 can be set here. The axis number is also used as an address for the controller. It has to be configured with the same number so that the controller is able to access it.

mode=POSITION : The SHS Star 2000 motion controller supports two operational modes for an axis: ANGLE and POSITION that can be set using this statement. Depending on this mode values can be entered in degrees or mm (inch/bits) only. A mixture between angular and planar movements is not possible for the same axis. Depending on that mode some of the following settings are ignored.

corr1 mm inc
corr2 mm inc
corr3 mm inc
corr4 mm inc
corr5 mm inc

The *corr* commands define a conversion and correction table from metric positions to incremental positions of the currently specified (or default) axis. This table consists of 5 entries where *corr1* defines the smallest possible value and *corr5* the biggest one. Here more than only a factor is given to allow it a user to equalize non-linear variances. The syntax of the *corr* table commands requires a metric value in millimeters and the appropriate incremental value that is equal to this metric position.

NOTE

The incremental value of corr1 must be smaller or equal than the parameter llimit and the incremental value of corr5 must be equal or bigger than the value of parameter hvalue that are described below. These values are used for the mode POSITION.

llimit=yyy : This parameter defines the lower limit the motion controller can drive to with the specified axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is smaller than the one set here. This value is used for the mode *POSITION*.

hlimit=yyy : This parameter defines the higher limit the motion controller can drive to using the current axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is bigger than the one set here. So with *llimit* and *hlimit* a range can be defined where the motion controller is allowed to work within. This value is used for the mode *POSITION*.

incperrot=yyy : Comparing to the correction table that defines the conversion factors from increments to positions and back this value has to be used to define a factor that specifies the relation between the number of increments and the appropriate angle for an axis. With *incperrot* it has to be specified how many increments are equal to one complete rotation (360°). This value is used for the operation mode *ANGLE*.

sfactor=yyy : For the speed conversion from mm/sec to increments/sec or from degrees/sec to increments/sec the speed factor parameter "sfactor" is used. Here it has to be specified how many increments/sec are equal to a speed of one mm/sec or one degree/sec. The distinction between the unit mm/sec or degree/sec is made by the current operational mode, the first one is true for mode "POSITION", the second for "ANGLE".

hslimit=yyy : This parameter can be used to set a maximum speed in unit increments/second. If a speed value sent from the application is larger than the value *yyy* after conversion using the *sfactor*, the speed setting sent to the drive is limited to the value given with this parameter. The high speed limit parameter is used for both operation modes.

defspeed=yyy : Using this parameter only the behaviour of the user interface is influenced, but not the functionality of the drive. Here a default speed value *yyy* can be set in unit mm/second that is displayed within the scanner application by default.

dname=c : When the motion controller is operated in party mode, that means with more than only one SHS at the same communication line, a single controller hardware needs to be accessed by using its device name. The parameter *dname* specifies such a name for the current axis. That name has to consist of a single char and it of course has to be configured and saved directly at the SHS motion controller hardware itself. Setting this name is necessary when more than one axis is configured using the *axis* command. Here for every axis a different name has to be specified. In case the *axis* command is not used and the controller operates using the default axis number two, the party mode for the motion controller interface and the hardware has to be set by sending the initialization command *PY 1* explicitly (please see the command *ival* below). Elsewhere the motion controller interfaces does not assume a party mode operation and ignores this name.

ival=sssss : The initialization of a single axis of the controller can be done using every command the drive supports. To specify, which of these controller commands have to be used during the initialization of it, the *ival* parameter can be used. Different to the preceding parameters this one can exist more than once in a configuration file. Here a second *ival* will not overwrite the value of the preceding one. So for every controller command that has to be sent one single line starting with this parameter has to be set. The controller commands are executed in the same order as the *ival* parameters appear in the settings file. The commands *sssss* that are defined with this parameter are used for initialization and are therefore sent as very first to the drive. The commands that can be set here are equal to the ones that are sent to the drive. Since the SHS controllers work with binary data they have to be entered as single byte hexadecimal values separated by a space without the CRC. This checksum is added to every command byte sequence automatically. So for an example such an entry could look like this:

- ival=0xFC 0x20 0x01
- Here 0xFC is the start byte, 0x20 defines the axis number the command is valid for and the length of it (according to the specification of the SHS Star 2000 commands), 0x01 is the command itself. The missing CRC byte does not has to be set, it is added by the controller automatically for every command.

hval=sssss : Similar to the *ival* parameter *hval* can be used to define the operation during a movement to the home position for the current or the default axis. Here the operation that has to be performed to reach the home position has to be programmed using one or more *hval* parameters that are executed in the same order as they appear in the configuration file. The syntax of the byte value commands is the same like described above for the *ival* command.

The original *sc_motion_shstar2000_settings.txt* settings file that is delivered with this scanner software contains some default settings that have to be changed according to the target environment of the motion controller. Additionally there are some example initialization and homing sequences defined within this file for several scenarios. Here the desired parts of these examples simply have to be uncommented. Other default parameters may have to be removed or commented out. As it can be seen there for both, the *ival* and the *hval* parameter, complete programs can be defined that work within the motion controller to perform the initialization or the homing of it.

5.5.8 Type 11 - Jena Ecostep100

The *ECOSTEP 100* motion controller series is supported directly. To enable it, please configure the drive type number *11* like it was described above.

For configuring the *ECOSTEP* controller interface, a configuration file *sc_motion_jenaecostep100_settings.txt* is required in the same location where the general motion settings file exists. The *ECOSTEP* configuration file can be used to specify different parameters like the COM port, the data rate, the initialization and others more. The file itself is a plain text file that contains different statements. Additionally there can be comments within the file that begin with a # sign directly at the beginning of a line. These comments are ignored completely. Following parameters, that have to start exactly at the beginning of a line, are supported for configuring the interface:

PortName=xxx : Specifies to which port the *ECOSTEP* controller is connected to, here for *"xxx"* e.g. COM1 has to be set.

PortBaudRate=yyyy : Defines the data rate (in bps) the COM port has to work with, the default and only value that is used by *ECOSTEP 100* controllers is *9600*.

PortParity=y : This parameter changes the parity mode for the port. Here for *y* the following values are supported:

- 0 - No parity
- 1 - Odd parity
- 2 - Even parity
- 3 - Mark parity
- 4 - Space parity

PortRTS=y : Using this parameter the ready to send behaviour can be configured like it is necessary for some serial protocols. Here the following values are possible for *y*:

- 0 - disable RTS fully
- 1 - enable standard RTS mode
- 2 - enable RTS handshake mode
- 3 - enable RTS in control toggle mode

TimeOut=yy : This value specifies how long the interface shall retry to access the communication interface until it fails with a time out in case such an operation is not successful.

Debug=y : For special debugging purposes it is possible to log several information into a file *sc_motion_jenaecostep100_debuglog.txt* that will be created at the position where the settings file is located at. With the value *1* the debugging mode is turned on, when it is set to *0* debugging is disabled and no data are put into the log file.

NOTE

There can be huge amounts of data that are logged into such a file so enable this option only if it is necessary. With (Debug=1) additional status request commands are send to the controller (and are reported in the log).

All these values are global ones and therefore they are valid for everything that is controlled by that motion controller. This is important for the multi-axes-mode where it is possible to handle up to five axes using this controller interface. Each of these axes can be configured in a different way. To do that, the *axis* statement that defines for which axis the following parameters are valid has to be used. In the worst case when three axes have to be used with completely different configurations all the following parameters have to exist five times, separated by the different calls of *axis*:

axis=y : Specifies for which axis the following configuration parameters are valid for. If this statement is not used somewhere in a configuration file it is assumed that the third axis is used. Else a value out of the allowed range *0..4* can be set here. The axis number is also used as address for the controller. It has to be configured with the same number so that the controller is able to access it.

mode=POSITION : The *ECOSTEP* motion controller supports two operational modes for an axis: *ANGLE* and *POSITION* that can be set using this statement. Depending on this mode values can be entered in degrees or mm (inch/bits) only. A mixture between angular and planar movements is not possible for the same axis. Depending on that mode some of the following settings are ignored.

llimit=yyy : This parameter defines the lower limit the motion controller can drive to with the specified axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is smaller than the one set here. This value is used for the mode *POSITION*.

hlimit=yyy : This parameter defines the higher limit the motion controller can drive to using the current axis (in unit increments). Independent from the values that are sent from the program, the controller will never be driven to a value that is bigger than the one set here. So with *llimit* and *hlimit* a range can be defined where the motion controller is allowed to work within. This value is used for the mode *POSITION*.

sfactor=yyy : For the speed conversion from mm/sec to increments/sec or from degrees/sec to increments/sec the speed factor parameter *sfactor* is used. Here it has to be specified how many increments/sec are equal to a speed of one mm/sec or one degree/sec. The distinction between the unit mm/sec or degree/sec is made by the current operational mode, the first one is true for mode *POSITION*, the second for *ANGLE*.

hslimit=yyy : This parameter can be used to set a maximum speed in unit increments/second. If a speed value sent from the application is larger than the value *yyy* after conversion using the *sfactor*, the speed setting sent to the drive is limited to the value given with this parameter. The high speed limit parameter is used for both operation modes.

defspeed=yyy : Using this parameter only the behaviour of the user interface is influenced, but not the functionality of the drive. Here a default speed value *yyy* can be set in unit mm/second that is displayed within the scanner application by default.

incperrot=yyy : Comparing to the correction table that defines the conversion factors from increments to positions and back this value has to be used to define a factor that specifies the relation between the number of increments and the appropriate angle for an axis. With *incperrot* it has to be specified how many increments are equal to one complete rotation (360°). This value is used for the operation mode *ANGLE*.

DevnoEcoId=c : *Device No. / ECO Id*. represents the device id of the *ECOSTEP* host base station. Allowed values are: *1 .. 15*.

dname=c : Sets the character this device has to be displayed with.

ival=sssss : These parameters can contain sequences of *Jena ECOSTEP 100* commands as hex-numbers that perform operations on the drive during initialization. For a detailed description of these parameters please refer to the specifications of the controller.

NOTE

The checksum and the deviceID do not have to be added to the command sequences. This is done automatically. An optional command identifier text (is logged in Debug mode), separated by a blank character, terminates the line.

E.g.:
The following three commands enable the drive to move (may be axis specific).
After the motion Controller default initialization no moving is possible
due to end position monitoring.

- ival=0x23 0x70 0x21 0x00 0x00 0x00 0x00 0x00 inputPolarityToDefault
- ival=0x23 0x71 0x21 0x02 0x00 0x00 0x00 0x00 pLockConfiguration
- ival=0x23 0x72 0x21 0x02 0x00 0x00 0x00 0x00 nLockConfiguration

hval=sssss : These parameters can contain sequences of *Jena ECOSTEP 100* commands as hex-numbers that perform operations on the drive during a homing sequence. For a detailed description of these parameters please refer to the specifications of the controller.

The checksum and the deviceID do not have to be added to the command sequences. This is done automatically.

E.g.:
homing to zero position (normal absolute positioning operation with default speed to zero position)

- hval=0x23 0x81 0x60 0x00 0x00 0x7E 0x04 0x00 setSpeed
- hval=0x23 0x7A 0x60 0x00 0x00 0x00 0x00 0x00 setPos
- hval=0x23 0x60 0x60 0x00 0x01 0x00 0x00 0x00 modeAbsolutPositioning
- hval=0x23 0x40 0x60 0x00 0x3F 0x00 0x00 0x00 executeRun

eval=sssss : These parameters can contain sequences of Jena EcoStep100 commands as hex-numbers that perform operations on the drive during a program exit. For a detailled description of these parameters please refer to the specifications of the controller.

The checksum and the deviceID do not have to be added to the command sequences. This is done automatically.

E.g.:
store all parameters permanent
eval=

- 0x23 - readWrite
- 0x10 - objLsb
- 0x10 - objMsb
- 0x01 - subIndex
- 0x73 - data0
- 0x61 - data1
- 0x76 - data2
- 0x65 - data3
- storeAllParameters - (for SAMLight Debug purposes)

ECOSTEP 100 / 200 Command format

byte **id;**
// id of ecostep100 station [1 .. 15]

byte **readWrite;**
// *Send from PC to ECO controller:*
// - 23 hex = Send command 4 Byte data (Byte 4..7 contain an 32Bit-value)
// - 2B hex = Send command 2 Byte data (Byte 4, 5 contain an 16Bit-value)
// - 2F hex = Send command 1 Byte data (Byte 4 contains an 8Bit-value)
// - 40 hex = Read data object from slave
//
// *Send from ECO controller to PC:*
// - 43 hex = Command 40 successfully send, Byte 4..7 contain an 32Bit-value
// - 4B hex = Command 40 successfully send, Byte 4, 5 contain an 16Bit-value
// - 4F hex = Command 40 successfully send, Byte 4 contains an 8Bit-value
// - 60 hex = Commands 23 or 2B or 2F successfully send
// - 80 hex = Commands 23 or 2B or 2F or 40 not successfully send, Byte 4..7 contain error

byte **objLsb;**
// obj id: last significant Byte

byte **objMsb;**

// obj id: most significant Byte

byte **subIndex;**
// obj subIndex

byte **data0;**
// if *controlword*:
// Bit0 = Switch On
// Bit1 = Disable Voltage
// Bit2 = Quick Stop
// Bit3 = Enable Operation
// Bit4 = operating specific
// Bit5 = operating specific
// Bit6 = operating specific
// Bit7 = Fault Reset
//
// if *statusword*:
// Bit0 = Ready to Switch on
// Bit1 = Switched On
// Bit2 = Operation Enable
// Bit3 = Fault
// Bit4 = Voltage Disabled
// Bit5 = Quick Stop
// Bit6 = Switch on Disable
// Bit7 = Warning

byte **data1;**
// if *controlword*:
// Bit0 = Halt
// Bit1 = reserved
// Bit2 = reserved
// Bit3 = manufacturer specific
// Bit4 = manufacturer specific
// Bit5 = manufacturer specific
// Bit6 = manufacturer specific
// Bit7 = manufacturer specific
//
// if *statusword*:
// Bit0 = manufacturer specific
// Bit1 = Reserved
// Bit2 = Setpoint reached
// Bit3 = Limit reached
// Bit4 = Setpoint confirmation / halt / found reference
// Bit5 = Contouring error / reserved / reference error
// Bit6 = Commutation found
// Bit7 = Reference found

byte **data2;**
// data byte 2

byte **data3;**
// data byte 3

byte **checksum;**
// inverted sum(Byte 0 .. Byte 8)

string **szPacketName;**

// for debug purposes

5.5.9 Type 12 - IO Switcher

With this controller special drive types can be used that start a movement after they received one single digital signal. That means using this driver it is not possible to stop software-controlled at an exact position. The driver only initiates the movement and the drive itself moves to a (predefined) position or by a (predefined) distance. When the drive has reached that position it gives back a signal to the software. This controller supports up to five motors. To enable this type of controller, please configure the IO switcher type 12 like it was described above.

For configuring the IO switcher motor controller interface itself, a configuration file *sc_motion_ioswitcher_settings.txt* is required in the same location where the general motion settings file exists. The IO switcher configuration file can be used to specify different parameters like the I/Os for the different output and input signals and others. The file itself is a plain text file that contains different configuration parameters. Additionally there can be comments within the file that start with a # sign directly at the beginning of a line. These comments are ignored completely. Following parameters, that have to start exactly at the beginning of a line, are supported for configuring the interface:

axis=y : Specifies for which axis the following configuration parameters are valid. If this statement is not used somewhere in a configuration file it is assumed that the first axis is used. That statement assigns all following parameter to this axis until another *axis* parameter is found within the settings file. That means using the *axis* parameter it is possible to configure more than one axis. Here all parameters have to be repeated for every axis and every section needs to be started by such an *axis* statement. The axis number itself also specifies the position within the motion control panel of the scanner software the axis input fields will appear.

startOutput=y : This parameter specifies which output number has to be used to send the signal for starting the movement to the drive. The value given for *y* is a number that specifies the output port of the scanner card and the output pin that is connected with the input of the drive. If a value in the range *0..15* is entered here then a standard digital output of the scanner card is used. When the value is in the range *100..107* the pins *0..7* of the laser port are used for sending the step signal.

The last method can be used only if the laser port is NOT used for controlling the laser. Else the results can be undefined and may harm your equipment seriously.

doneInput=y : This parameter specifies which input number has to be used for the movement feedback. At this input a signal is expected that is sent from the drive to the controller to tell that the final position was reached. The value given for *y* is a zero-based number that specifies the input pin that switch is connected with.

doneInputValue=y : The preceding parameter an this one are related to each other: *doneInputValue* defines what signal at the input with number *doneInput* is expected to be interpreted as the information: *the movement was completed*. Here for *y* the values *1* and *0* are possible.

dname=c : This parameter specifies a name for the current axis that is used to display the name within the scanner software. That name has to consist of a single character.

5.5.10 Type 13 - Isel CanApi Controller

To use this motion control define type 13 in the file sc_motion_settings.txt. The corresponding settings file is sc_motion_isel_settings.txt. In the following the possible parameters are explained:

Debug=x: The debug mode can be enabled in case of problems, here different important information are put into a debug log named 'sc_motion_ext_isel_debuglog.txt' that is located at the same position like this settings file. To enable debugging set x=1 else set x=0.

Dll=C:\CNCWorkbench\Control\CAN\CanApi.dll: Set the Path to Isel Can Api Dll file. If path contains blank(s), set whole path between double quotes.

Ini=C:\\CNCWorkbench\\Control\\CAN\\CAN_PCI_1_Axis.ini: Set the Path to Isel Can Api Ini file. If path contains blank(s), set whole path between double quotes.

SwitchOffPowerState=x: If set to '1', a call to 'SetPowerState(1)' is done before each movement. After each movement (including homing) 'SetPowerState(0)' is called. 'SetPowerState(0)' disables axis amplifiers (switching them off), 'SetPowerState(1)' enables axis amplifiers (switching them on).

DisableHomingDuringStartUp = x: If set to '1', homing procedure during SAMLight start-up will be switched off.

axis=x: Axis index (zero based)

llimit=x: Axis range - Low Limit ['um' in case POSITION, '1/3600°' in case ANGLE]

hlimit=x: Axis range - High Limit ['um' in case POSITION, '1/3600°' in case ANGLE]

lslimit=x: Low speed limit [mm/second or degree/second]

hslimit=x: High speed limit [mm/second or degree/second]

defspeed=x: Default speed [mm/second or degree/second]

6 Global Settings

In this chapter the global settings of SAMLight are listed.

System: Opens the system settings dialog.

Reset License: In SAMLight it is possible to redefine the license key. You can use it, if you have received an update password from SCAPS and to activate it. Therefore click in the menu on *Settings → Reset license*. A questioning window will appear, which you have to answer with OK. Then the usual password enter dialog will be displayed where you can enter the new password. If you copy the password out of a text editor first, you can easily paste it by clicking in the first enter field and then press Ctrl-V and Enter. After upgrading of a license, a new password is supposed to replace the old password, in this situation, a restart of SAMLight is necessary for resetting license.

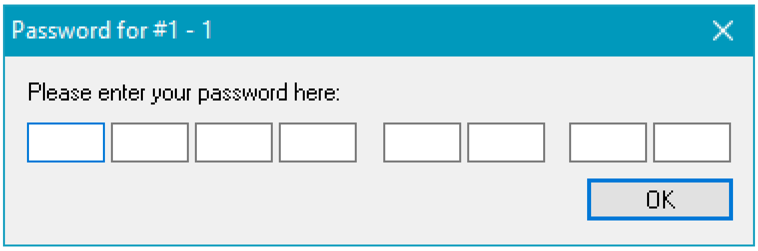


Figure 42: Reset License password Dialog

Units: Here you can choose the displayed unit. Available is mm, inch and bit.

6.1 View

The following dialog can be reached by Menu item *Settings → System → View*.

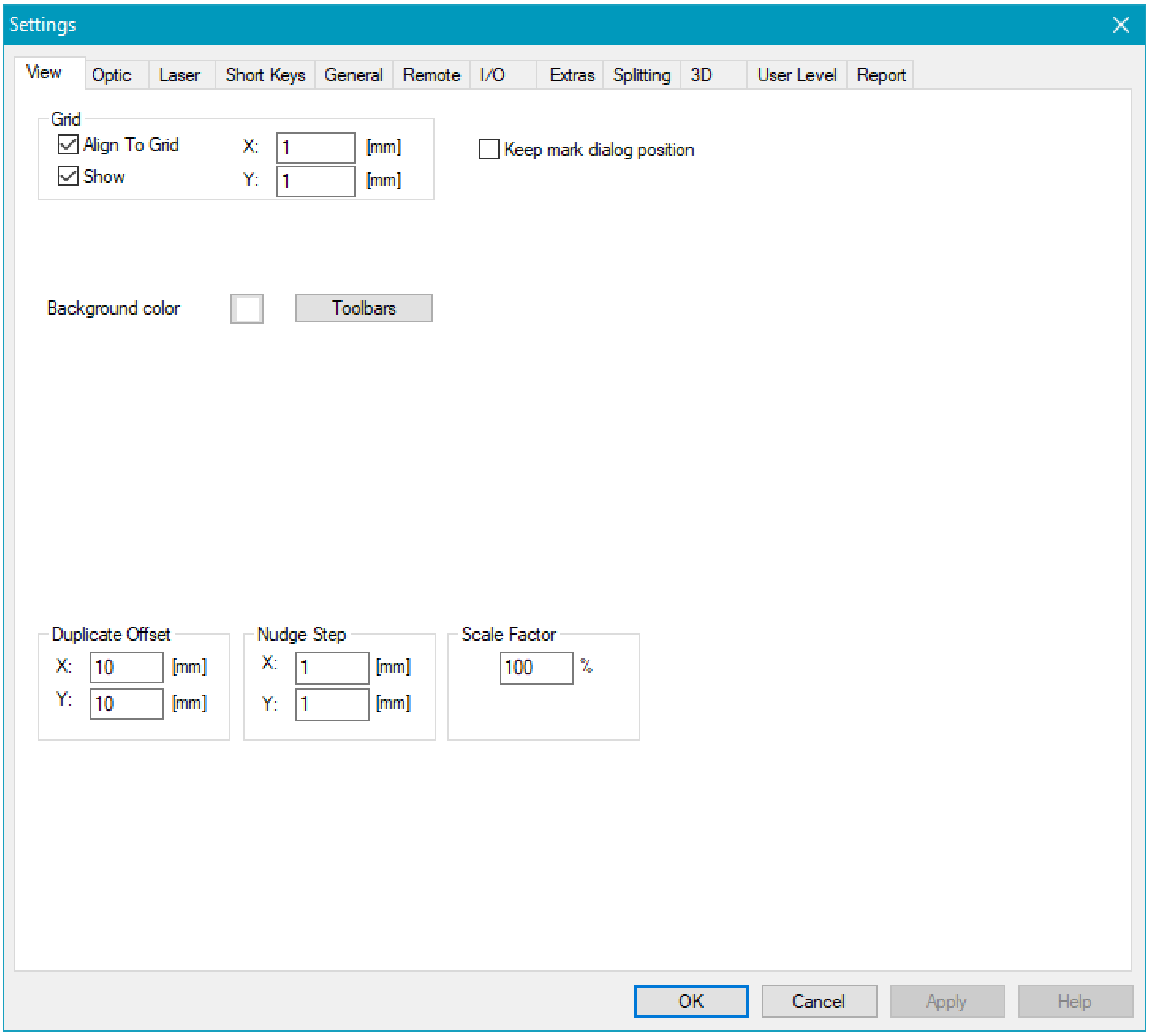


Figure 43: View Settings Dialog

Grid:

> **Align To Grid:** If checked each new object placed in the View 2D will be aligned to the grid.
>
> **Show:** If checked the grid will be displayed in the View 2D.
>
> **X, Y:** These two values define the grid size.

Keep mark dialog position: If checked, the position of the mark dialog is remembered permanently (also after restarting SAMLight).

Background color: Clicking on this button opens a color dialog, where the background color of the View 2D can be defined (also possible in ViewProperties).

Toolbars: Clicking on this button opens a dialog where the user can choose which of the available toolbars is shown. See chapter Toolbars.

Duplicate Offset:

> **X, Y:** These two values define the copy offset in x and y direction. For example, select an object in the View 2D and click *Menu bar → Edit → Copy*. This creates a copy of the selected object which is placed next to the original translated by the copy offset. If the copy offset is zero in both directions the created copy will cover the original.

Nudge Step:

> **X, Y:** These two values define the nudge step in x and y direction. The nudge step is used for the operations *nudge left*, *right*, *up* and *down* described in chapter Edit. Nudge can also be used with hotkeys while using the red pointer (see figure 44).

Scale Factor: This value is taken for the scaling of the job with hotkeys while using the red pointer in the mark dialog (see figure 44).

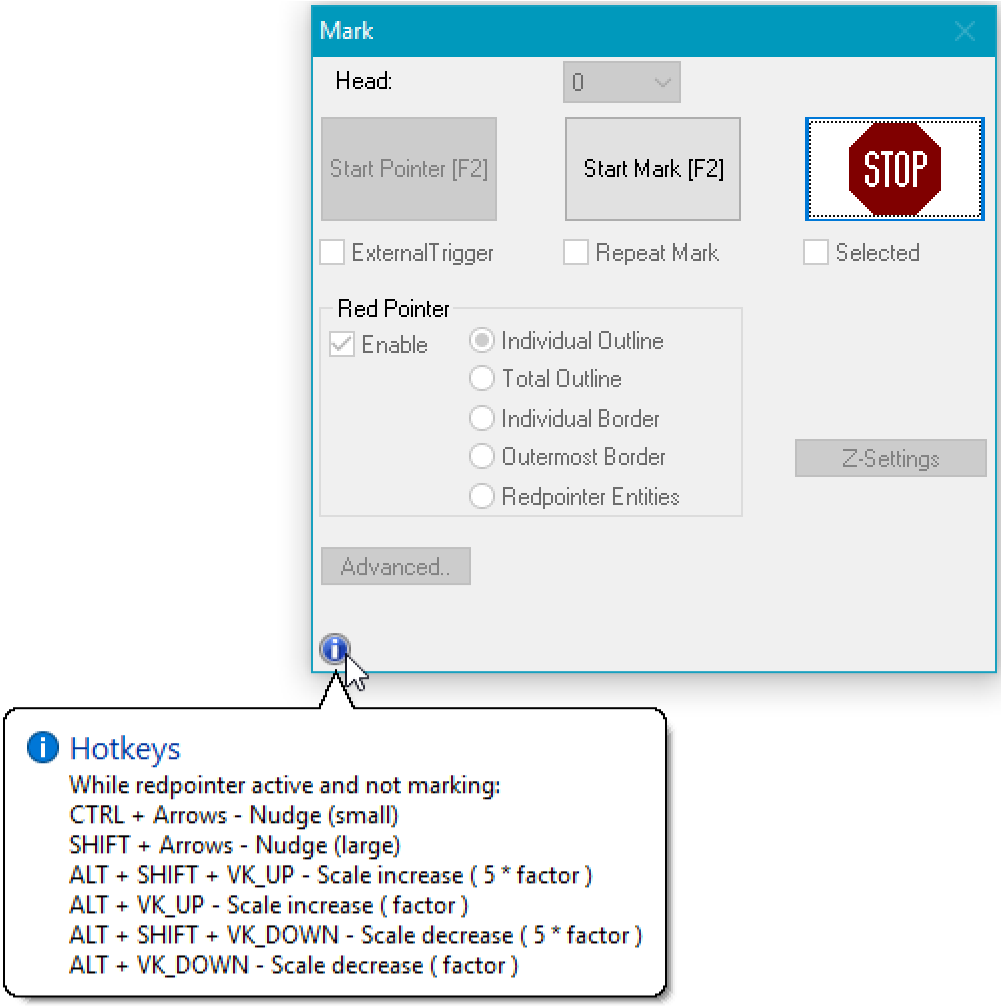


Figure 44: Scale/translate job with hotkeys while using red pointer in the Mark Dialog

6.2 Optic

6.2.1 USC Cards

The following dialog can be reached by Menu item *Settings → System → Optic*. The dialog allows to define the settings of the geometrical dimensions of the scanner field for the selected *head*:

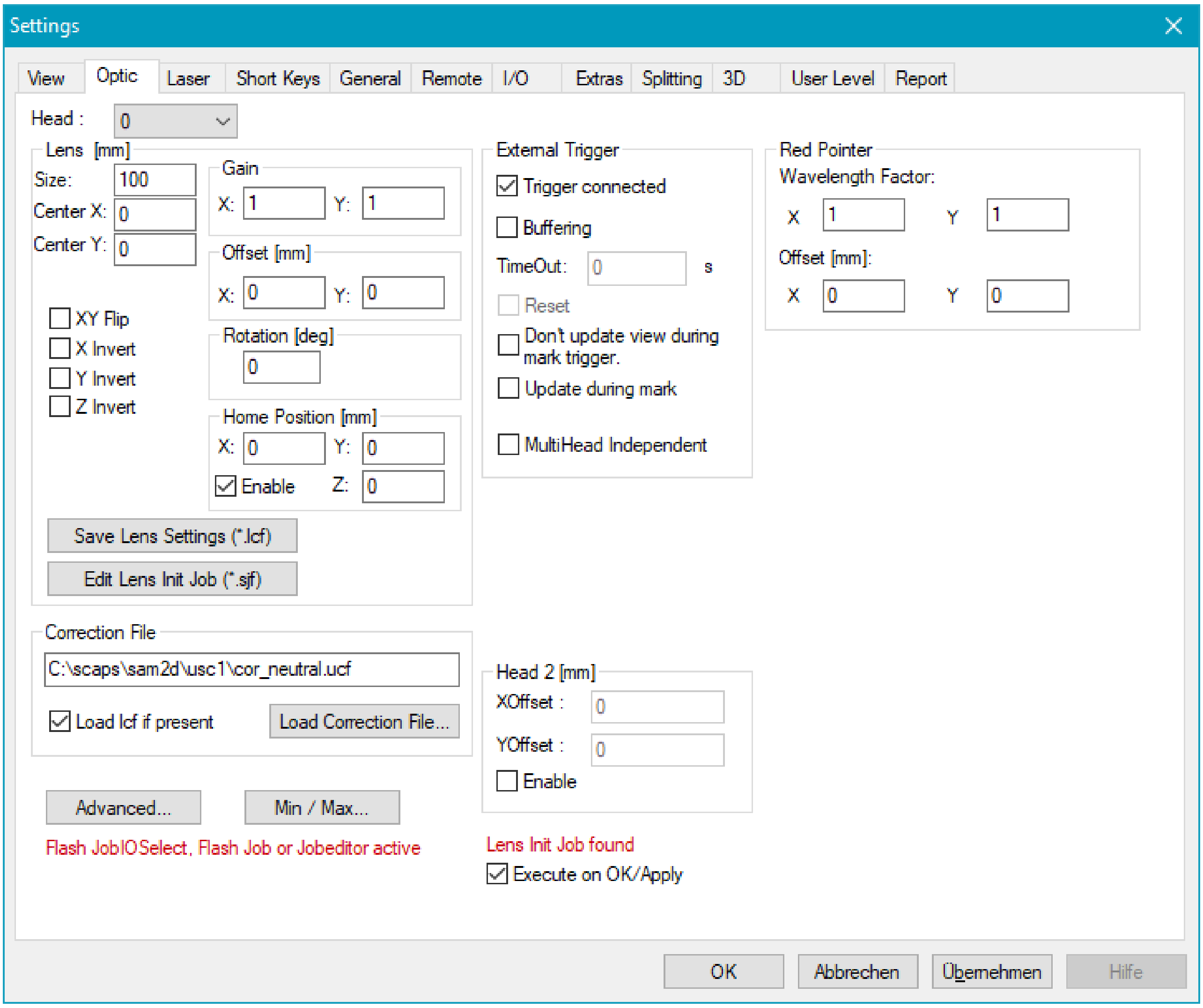


Figure 45: Optic Settings for USC cards

Head: Here the head number can be selected if more than one head is present. For this setting the software option MultiHead is required.

Lens [mm]:

Size: Specifies the maximum scanning field. This is the maximum field the scanner system can drive. In general it depends on the optical system (like lenses, etc.), the scanning distance and the maximum scanning angle of the scanner system. On 16 bit scanning systems the field extent corresponds in general to the bit values -32767 to 32767 respectively.

If the dimensions of output do not correspond to the dimensions of the drawn object, this can be corrected by changing the field size. Thereby the new field size is being calculated out of the current field size in the following way:

corrected field size = current field size * C / O

C is the current output dimension and **O** is the original dimension of the drawn object.

Center X, Y: Defines the center of the scanner field according to the world coordinate system.

XY Flip: Exchanges X and Y axis.

X, Y, Z Invert: Inverts the axis direction. *Z Invert* requires the SAM software options Optic3D or SAM3D.

Gain X, Y: The X and Y gain values are thought to compensate slightly X, Y gain errors to archive a quadratic field. Values less than 1 will reduce the scanner field. Values greater than 1 will expand the scanner field. Global gain errors which have the same deviation in X and Y directions should be corrected by changing the size of the scanner field (see above).

Example for calculating the size of the scanner field:

If for example the correction table has a step size of 550 bits/mm and the scanner has a 16 bit scanning system, the

size of the scanner field = 65535 bits * mm / 550 bits = 119.16 mm

Considering a Gain factor:

scanner field in x direction = 119.16 mm * *Gain X*

scanner field in y direction = 119.16 mm * *Gain Y*

Offset X, Y: The offset values are thought to slightly compensate X, Y offset errors to achieve the theoretical midpoint of the scanner field. Global offset errors which have the same deviation in X and Y directions should be corrected by changing the field X, Y min values in *field*.

Rotation [deg]: The scanner output is rotated by an angle that is entered here.

NOTE

The Gain, Offset and Rotation values entered here will not affect the Optic Matrix. To change the Optic Matrix you have to go to Advanced → Correction → Settings.

Home Position X, Y, Z: If this option is enabled, the home position is the position where the scanner is located when no scanning takes place i.e. during handling activity. The home position is in normal cases outside the working area but it must be inside the scanner field.

Save lens settings (*.lcf): A lens settings file stores all lens specific information. The file will be generated or updated if you click this button. The name of the *.lcf file is bound to the name of the correction file you want to use and needs to be in the same directory. If you do not want to load a *.lcf file together with your correction file you can disable 'Load lcf if present'. Please use the button 'Load Correction File...' to load the correction and *.lcf file. The data stored in the lens settings file include the whole parameters in the Lens field, the 4 parameters under *Settings → System → Laser → Red Pointer* and the Red pointer delay in *Mark Dialog → Advanced*.

Edit Lens Init Job (*.sjf): Opens the Edit_Lens_Init_Job_Dialog. If a job should be executed every time a lens is chosen, a lens initiation job can be created. This job will be executed when switching to the corresponding correction file via this dialog. A typical job could be: waiting for an input, setting an output, executing a motion control or running an executable. The name of the lens init job (*.sjf file) must equal the name of the selected correction file and needs to be in the same directory (automatically done when using the button). When switching correction files, the message 'Lens Init Job found' indicates if an existing lens init job file for this correction file has been found. To suppress the execution of this file please disable 'Execute on OK/Apply' before clicking apply/OK. If the job should not be executed any more it can be deleted in the corresponding directory.

Correction File Settings: Here a correction file can be specified. It is used to compensate optical distortions (barrel / pincushion) of the scan head. Usually the scan head manufacturer can provide a correction file for their products. For USC cards the standard extension is *.ucf*. By clicking on 'Load Correction File...' a browser dialog opens. After selecting a correction file the corresponding lens settings file (*.lcf) and lens init job (*.sjf) will be opened together with the correction file if these files are stored at the same directory. That means using this button different lens- and correction file configurations can be managed.

Advanced...: See dialog USC-1 Card Settings, USC-2 Card Settings or USC-3 Card Settings.

Flash JobIOSelect, Flash Job or Jobeditor active: Appears for USC-2 or USC-3 card if Flash Job IO Selection mode, a Flash Job or a Jobeditor license is used and deactivates the "Advanced..." button. To activate it in case of Flash Job IO Selection mode, please open Flash dialog at *Menu bar → Extras → Flash* an press "STOP IOSelect" button, which then turns to "START" button.

Min:/Max: Opens the Min/Max Dialog to define the range of the values speed, frequency and first pulse killer length of the laser.

External Trigger:

Trigger connected: enables the external trigger and the flags described below. This checkbox activates the "External trigger" in the Mark dialog and "Mark - Trigger".

Buffering: If in trigger mode, the job is buffered (several times) on the scanner card. This allows a faster start of the marking in response to the external trigger signal. The buffer is refilled before it is drained. In buffering trigger mode, the Status Outputs do not work. This function is only recommended if no exe or motion objects are included to the job. It can be marked, but only one job can be buffered.

TimeOut: If waiting for the trigger, the buffer on the scanner card will be cleared after there was no external trigger signal for longer than a given time. While clearing and refilling the buffer, an external trigger signal is ignored. The time out value needs to be bigger than the time for marking.

Reset: If this option is enabled it extends the timeout operation. In this case not only the buffer of the scanner card is cleared but the sequence is reset too. That means when a timeout occurs and this checkbox is selected all serial numbers are reset to their defined starting values automatically.

Don't update view during mark trigger: If this option is enabled the view will not be updated when using Mark → Trigger as long as the Mark Trigger window is active.

Update during mark: If this checkbox is active, the next sequence is set within the current job before it ends. This means that serial numbers are for example incremented or re-split. This checkbox is only valid if "Next sequence before mark" is unchecked.

MultiHead Independent: This allows to trigger the scan heads independently from each other when the Mark→Trigger dialog is used. This might be useful if one of the scan heads has to mark for example 5 objects while the other scan head will only mark one object at the same time. This function will not work together with serial number objects nor with DateTime objects.

Red pointer:

Wavelength Factor: Correction for the red pointer in X and Y direction caused by different wavelength of the laser. The factor is determined experimentally. When marking one reference point with the red pointer and the laser, the vector length R from center to the point marked with the red pointer and the vector length L from center to the point marked with the laser needs to be measured in x and y direction. Wavelength Factor = Red pointer / Laser.

Offset: Static X/Y offset correction only for the red pointer caused by the different wavelengths of the laser. The factor is determined experimentally (see Wavelength Factor).

Z Dimension: This requires the software option Optic3D.

NOTE *Optic settings can be changed in the optic settings dialog which can be reached by Menu item Settings → System → Optic, but there is also an other way:*

1. Close all SAM based applications (Check this with task-manager).
2. Start *sc_setup.exe* from folder *<SCAPS>\tools*
3. Click on Menu *Hardware Settings → Settings*

Head 2 (*only available for USC-2*):

X, Y Offset: Define an offset for the second scan head. If everything is set to 0 then the second scan head will mark on the same position as the first head is marking.

Enable: Activate this checkbox to enable the usage of the second scan head.

NOTE *If the MultiHead option is used, then the Head 2 options are disabled. Instead there is an additional option in the External Trigger options: MultiHead Independent. Therefore see Optic Settings Dialog USC-1.*

Lens Init Job found: Appears if a correction file is loaded and an appropriate Lens Init Job is available.

6.2.1.1 Edit Lens Init Job Dialog

The following dialog can be reached by using menu item *Settings → System → Optic. Here for a specific correction file (*.ucf) an init job can be defined which will be executed automatically if the corresponding correction file with the same name is chosen via the 'Load Correction File...' button.*

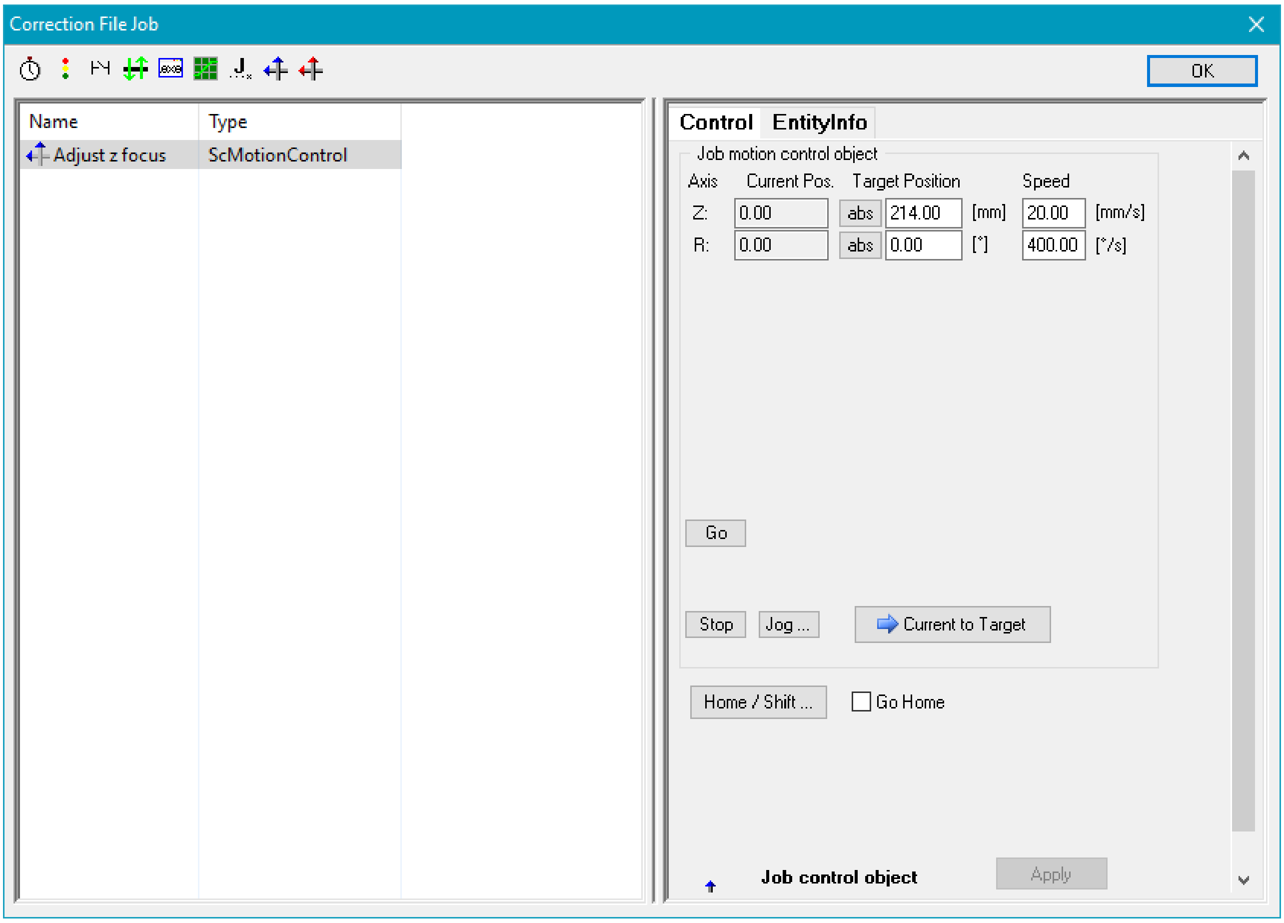


Figure 46: Correction File Job Dialog

6.2.2 RTC Cards

The following dialog can be reached by Menu item *Settings → System → Optic*. The dialog allows to define the settings of the geometrical dimensions of the scanner field for the selected *head*:

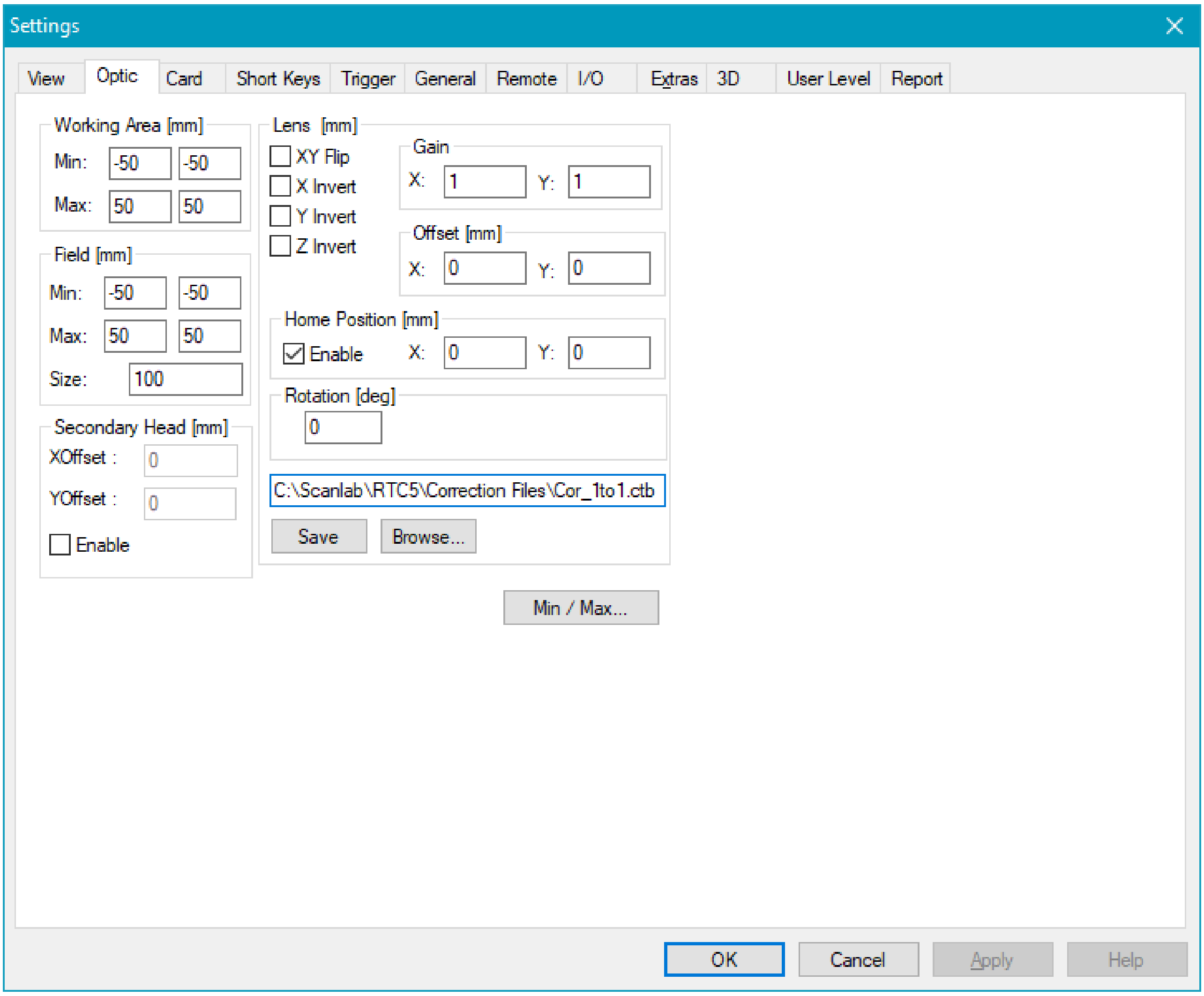


Figure 47: Optic Settings for RTC cards including SCANalone

Working Area [mm]:

Min:/Max: Specifies the Min and Max X, Y scanning field on which the exposure should take place. The working area must be inside the *field* dimensions.

Field [mm]:

Min:/Max: Specifies the Min and Max X, Y scanning field.

Size: Specifies the maximum scanning field.

Secondary Head [mm]: This field is available for RTC3/RTC4 and SCANalone only. It allows to control two heads with one card. After enabling it there are two working areas shown on the job editor.

X, YOffset: Defines the offset of the second head to the first head. For more information about how to use a second head see also the chapter Option MultiHead.

Lens [mm]:

XY Flip: Exchanges X and Y axis.

X, Y, Z Invert: Inverts the axis direction. *Z Invert* requires the SAM software options Optic3D or SAM3D.

Gain X, Y: The X and Y gain values are thought to compensate slightly X, Y gain errors to archive a quadratic field. Values less than 1 will reduce the scanner field. Values greater than 1 will expand the scanner field. Global gain errors which have the same deviation in X and Y directions should be corrected by changing the size of the scanner field (see above).

Example for calculating the size of the scanner field:

If for example the correction table has a step size of 550 bits/mm and the scanner has a 16 bit scanning system, the

size of the scanner field = 65535 bits * mm / 550 bits = 119.16 mm

Considering a Gain factor:

scanner field in x direction = 119.16 mm * *Gain X*

scanner field in y direction = 119.16 mm * *Gain Y*

Offset [mm] X, Y: The offset values are thought to slightly compensate X, Y offset errors to achieve the theoretical midpoint of the scanner field. Global offset errors which have the same deviation in X and Y directions should be corrected by changing the field X, Y min values in *field*.

Rotation [deg]: The scanner output is rotated by an angle that is entered here. The resulting scanner output can be checked with Mark -> Preview. Beside the rotation defined in SAMLight -> Settings -> System -> Card (which is performed by the RTC card during the marking process) this rotation will affect the coordinates which SAMLight is sending to the RTC card. Please note that this rotation can lead into a galvo out of range error message.

Home Position X, Y, Z: If this option is enabled, the home position is the position where the scanner is located when no scanning takes place i.e. during handling activity. The home position is in normal cases outside the working area but it must be inside the scanner field.

Correction File Settings: Here the correction file can be specified. This file normally is delivered by the scan head manufacturer and needs to fit to the scanner. For RTC-cards the standard extension is *.ctb*. When the *Save* button is pressed the Lens settings from this panel are stored related to the currently used correction file. By clicking on *Browse...* such a correction file and its related settings can be loaded afterwards. That means using these buttons different lens- and correction file configurations can be managed.

Min:/Max: Opens the Min/Max Dialog to define the range of the values speed, frequency and first pulse killer length of the laser.

6.2.3 Min/Max

By pressing *Min /Max...* in the Menu *Settings → System → Optic* dialog following dialog appears. The settings define the range of the values *Mark* and *Jump Speed [mm/s]*, *Frequency [kHz]* and *First Pulse [µs]* killer length of the laser.

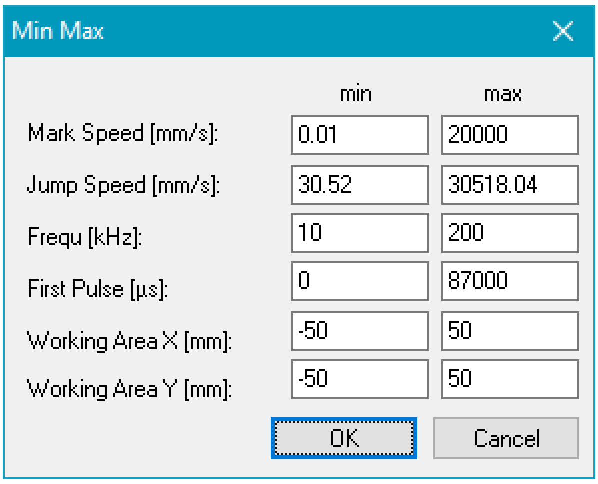


Figure 48: Min Max Dialog

When using the USC-1 or USC-2 card additional min max values, *Working Area X [mm]* and *Working Area Y [mm]* coordinates, can be set to define the working area inside the scanner field.

NOTE

The Min and Max values of Working Area X and Working Area Y are automatically adapted to the field size values (default), if Size / Center X / Center Y in the optic dialog are being changed.

NOTE

The Min and Max value would be saved in pen settings. So, if "save pens on exit" in General is not enabled, don't forget to click the button "Save Pens Now".

Although the Min and Max values are saved in pen settings, after loading a job file with pen settings, the Min and Max values in system would not be overwritten.

6.3 Laser

The following dialog can be reached by using menu item *Settings → System → Laser*, here the laser and the shutter control can be configured.

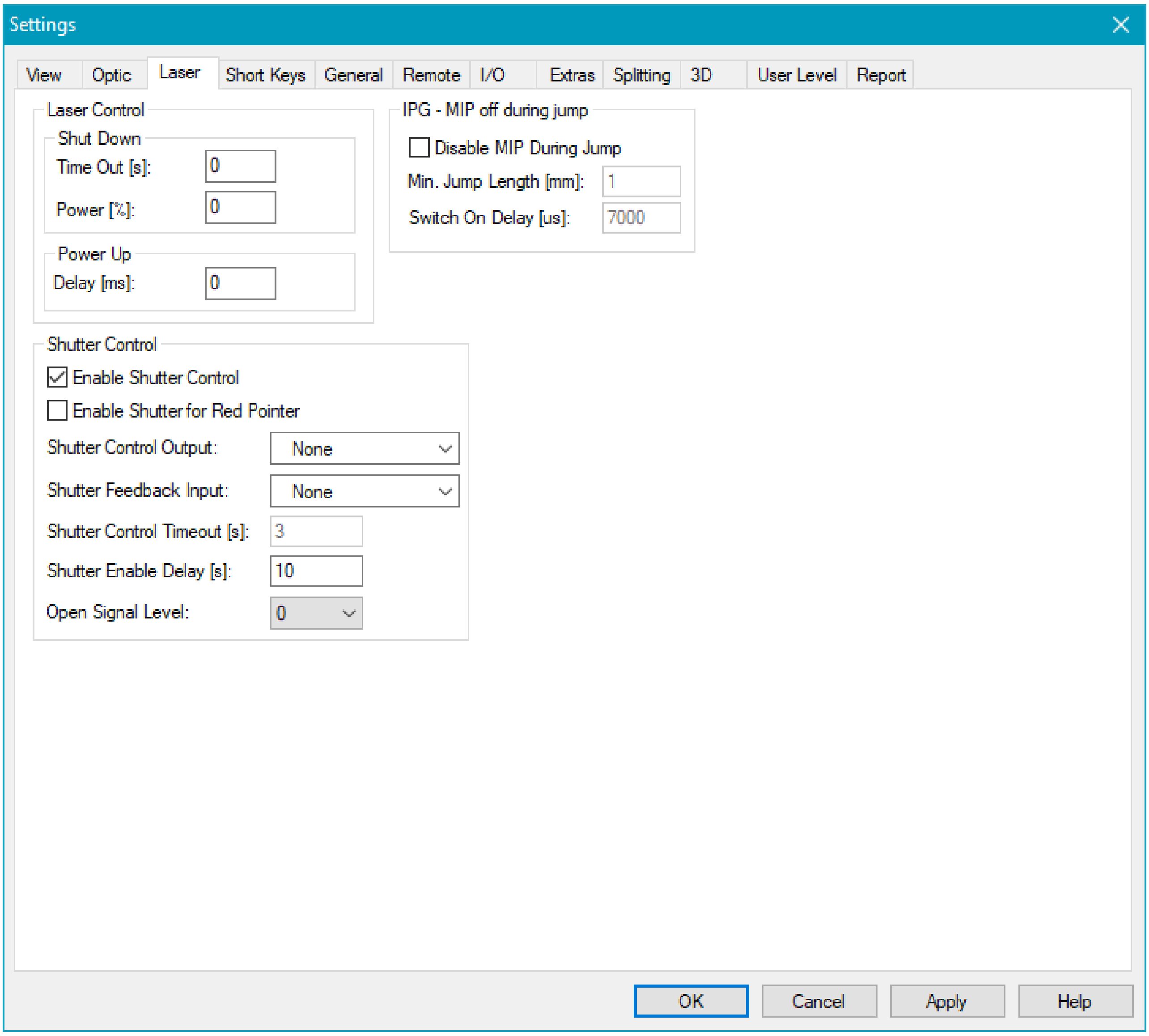


Figure 49: Laser Settings Dialog

Laser Control:

Shut down (power save mode):

Time out [s]: The Laser is going to power save mode after a defined time in which the laser was not active. If a value of 0 is entered here the laser power save mode feature is disabled.

Power [%]: Percentage of laser power for the power save mode. When the power is '0' the laser will shut down.

Power up:

Delay [ms]: When the laser is in power save mode and is going to normal power again a delay can be defined for the power up procedure. In this power up time the laser power will ramp from the power save to full power. The laser gate and the marking in process signal will not be active during the power up procedure. After the power up procedure the laser is in normal mode again and the marking procedure will begin.

IPG - MIP off during jump: This field is available only if using an USC-1 or USC-2 card and if *IPG* is selected under *Settings → System → Optic → Advanced → Mode*. If an IPG laser is connected this mode allows it to switch off Emission Enable (connected to OPTO_OUT_0) during jumps.

Disable MIP during jump: If this checkbox is activated then the IPG laser will be switched off during jumps.

Jump Length [mm]: Jumps that are longer than this value will lead to a switch off.

Delay [µs]: When switching on after a jump, this delay is executed before the scanner continues with the next vector.

Shutter Control:

Enable shutter control: Enable the shutter control functionality. All other parameters can be edited

Enable shutter for red pointer: Enable the shutter functionality for activation of the red pointer.

Shutter control output: Specify the output bit that is used to control the shutter can be defined. It sends opening and closing signals to the shutter depending on current state of the application. If the shutter is closed, an opening signal is sent before the laser is turned on e.g. for marking operations. If marking has finished and the enable delay (please see input field described below) has elapsed a close signal is sent to the shutter.

Shutter feedback input: Select a digital input that can be used for reading back the current opening or closing state of the shutter.

Shutter control timeout [s]: Here a timeout value can be configured that is related to the *Shutter feedback input*. If the shutter is controlled by the application to open or close an error message is displayed and the current operation is cancelled if that operation takes longer than the timeout that is configured here. This timeout requires an input where the current state of the shutter can be read back.

Shutter enable delay [s]: With this delay the shutter can be closed automatically and without any user interaction. If more seconds have been gone after the last time the laser was turned on, the shutter is closed automatically. So this field defines a delay after the last marking operation after that the shutter will be closed.

Open signal level: Change the signal level for the *Shutter control output*. The user can change between level *0 (active high)* and level *1 (active low).* With this option the behaviour of the scanner software can be modified in order to fit to the used shutter hardware.

Level 0: Not recommended to use! The *Shutter control output* signal during the boot process until the first marking procedure might be wrong and the shutter first operates correctly after the first marking.

Level 1: Recommended to use! Depending on the required signal level for the shutter, the selected *Shutter control output* bit can be inverted manually.

6.4 Shortkeys

The following dialog can be reached by Menu item *Settings → System → Short Keys*.

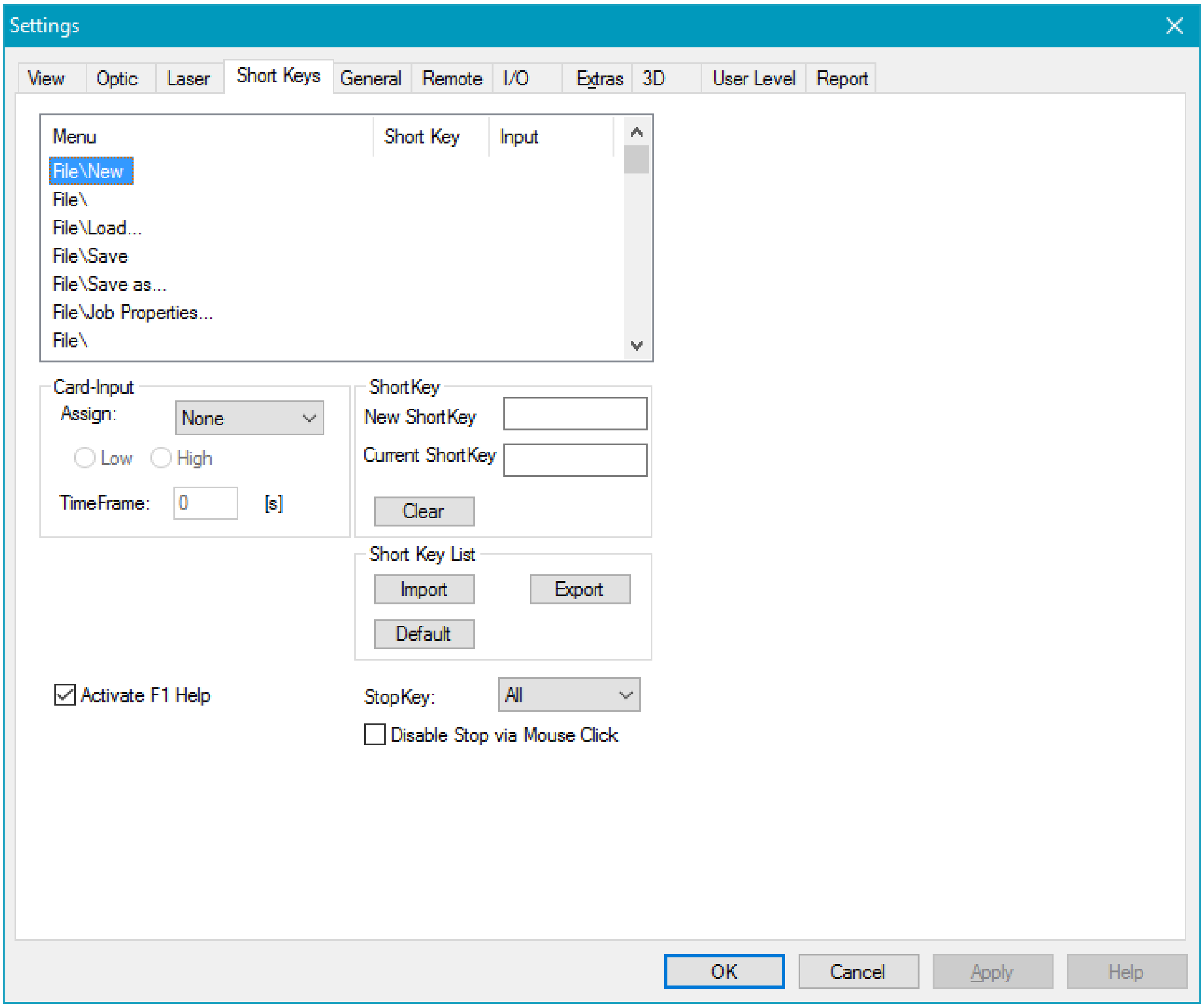


Figure 50: Short Keys Settings Dialog

Card-Input (only available for USC-1 and USC-2): To each short key a card input can be assigned. There are six input bits available (IN1 to IN6) which corresponds to the OPTO_IN0 to OPTO_IN5 pins of the USC card. The default assignment is None. The effect is that a special short key will be triggered if the assigned input is high/low if high/low was selected.

TimeFrame: Defines a repetition rate of the operation the input bit is assigned to. The operation will be executed as long as the input bit is low (or high) with the repetition rate 'TimeFrame' in seconds. The lower limit for this value is about 200 ms.

Short Key: For each menu item a user defined short key can be assigned: Select the requested menu item from the list. Click the empty field *New Short Key* and press the corresponding short key on the keyboard (for example F2) to assign it to the selected menu item.

Activate F1 Help: Activates the context sensitive help. Therefore the F1 key is predefined.

The default short key for the start buttons inside the mark dialog is F1. So if Activate F1 Help is selected the short key defined for the menu item Mark → Start will be taken for those buttons instead.

Short Key List: With *Import* a short key file (*.sam) can be imported. The short key list defined in the dialog can be exported by *Export*. *Default* restores the short key list to its default values.

StopKey: The keys that stop marking can be selected within this combo box.

Disable Stop via Mouse Click: If you enable this function, marking can only stopped with the defined StopKeys.

You can use short keys for the motion control. The short keys are assigned to the jog of the axes.

6.5 General

The following dialog can be reached by the Menu item *Settings → System → General*.

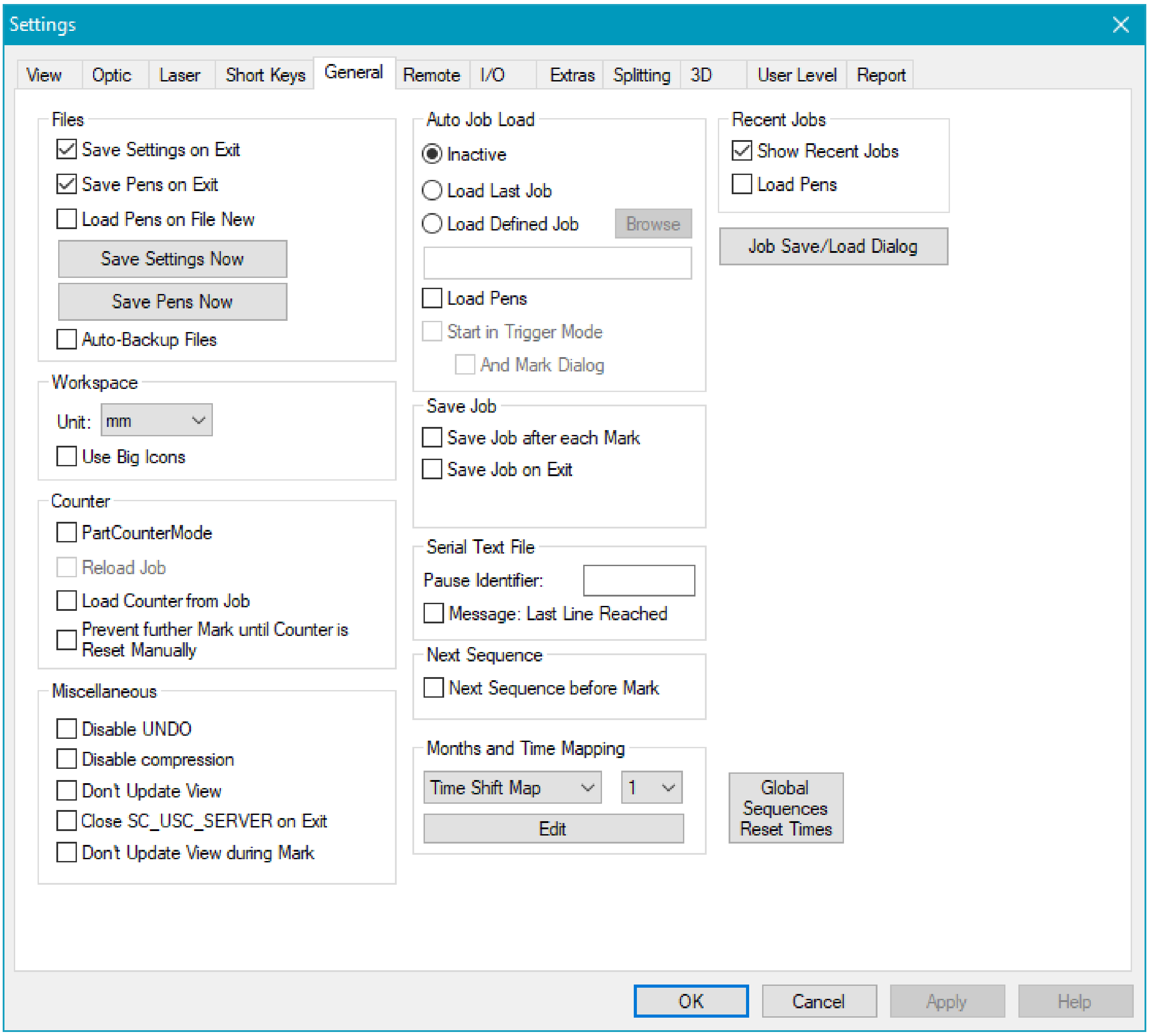


Figure 51: General Settings Dialog

Files:

Save Settings on Exit: If checked, the modified settings will be saved on every program exit.

Save Pens on Exit: If checked, the pens will be saved on every program exit and overwrite the last ones.

Load Pens on File New: If checked, the pens are loaded by clicking on *Menu bar → File → New*.

Save Settings Now: Saves the modified settings now and overwrites the last ones.

Save Pens Now: Saves the pens now and overwrites the last ones.

Auto-Backup Files: Creates automatically backups at the same path, where the job is saved with a continuous number in range 1..99 when this box is checked. To restore such a backup, the desired *.sjb file has to be renamed into *.sjf and can be loaded normally afterwards. The backup file names consist of the original file name plus the backup number and the new extension .sjb. The backup is created after the current job has been saved. The newest backup will be always listed with number 1 and all further possible already existing backups will be incremented. If the backup with number 99 already exists, it will be deleted.

Job Auto Load Box:

Inactive: No job will be loaded at program start up.

Load Last Job: The last edited job will be loaded at start up.

Load Defined Job: If this is activated browse through the file tree (button *browse*) and choose an *.sjf file to be your predefined job or type in the absolute path to this file in the edit line below. The chosen file will be loaded at start up.

Load Pens: If this is checked the pens of the job will be loaded as well, otherwise only the objects. This only makes sense if pens are saved within this job. See Job Format.

Start in Trigger Mode: If checked the application switches into trigger mode after loading a job at start up or after loading a job in Job Select input mode. The effect is the same as loading a job and starting the trigger mode by clicking *Menu bar → Mark → Trigger*.

And Mark Dialog: Defines if you want to start with the trigger dialog (standard) or with the mark dialog. Indeed, in some versions of SAMLight, the trigger dialog forces the user to start marking manually before accepting trigger signals.

Save Job:

Save Job after each Mark: Saves the job after each mark. No new file will be created. The existing file will be overwritten.

Save Job on Exit: The job is saved automatically when closing SAMLight. No new file will be created. The existing file will be overwritten.

Workspace:

Unit: Here on can define the length unit of the workspace. Three settings are possible: Millimetres (mm), Inch (inch), Bits (bits)

Use Big Icons: When this field is checked, bigger icons with double width and height are used after the next program start for all toolbars. This feature is useful e.g. in environments with limited input capabilities like touchscreens. Other settings like the sizes of menus and window title bars are not subject to the application. These properties are managed by the operating system exclusively. To get menu entries and title bars that are big enough to grip them via a touch, please change the appropriate operation system settings.

Counter:

PartCounterMode: Enables counting number of parts per marking process. The counter will be set to the number of marked parts while marking instead of being set to the number of marking sequences. One part is made of the number of objects defined in the job properties. If for example NumObjectsPerPart is set to 2 and there are 10 objects in the job list, the counter gets incremented with 5 after each marking sequence.

NOTE *If quantities is defined and it is not a multiple of the number of parts in the job list, the software offers to delete the overlaying objects, so that they do not get marked within the last sequence. To ensure that the original job will not be changed, the following check box can be selected.*

Reload Job: Reloads the job after quantity got reached.

Load Counter from Job: The counter information can be stored in the job. When loading the job the counter is also set to the old value.

Serial Text File:

Pause Identifier: The string that is defined here is used as a pause identifier for serial numbers read from an ASCII file. When the identifier is found in the current line of the assigned text file it causes a break of the marking and a pop up window which asks for continue is opened.

NOTE

For how to use ASCII for serialization see chapter How To / Automate Serialization.

Message: Last Line Reached: Gives a message if the current line of the serial number text file which needs to be assigned to the serial number is empty. The user has the possibility to reset the serial number sequence with committing the message. Instead or in addition you can also set an I/O bit if the last line of the text file has been reached.

Miscellaneous:

Disable UNDO: If this checkbox is selected the UNDO-and REDO-functionality is disabled completely. That means changes within a job have to be reverted manually, the automated function is disabled. It is recommended to disable the UNDO-functionality in case of speed or memory problems on smaller computer systems (embedded systems which are not used to create and edit jobs but to control the marking process only). Depending on the size and complexity of a job this option is able to save nameable amounts of memory and calculation time.

Disable Compression: If this checkbox is selected the zip compression of a saved *.sjf file is deactivated. This can help to save big job files on computer systems with low main memory.

Don't Update View: If this checkbox is selected the View2D is not updated any more. This can help to save processing time, when entities are updates (date/time and serial number entities, reimport of bitmaps, ...)

Close SC_USC_SERVER on Exit: If this checkbox is selected the sc_usc_server is close when closing SAMLight.

Don't Update View during Mark: If this checkbox is selected the View2D is not updated during marking and during *Mark → Trigger* Dialog is open.

Next Sequence (not for Splitting):

Next Sequence before Mark: If you start the marking from the mark dialog at first the serial numbers in the job will be increased and then marking will start.

Months and Time Mapping:

Time Shift Map: Opens a dialog where the working shift placeholder of date time can be mapped to working shifts times, see chapter Shift Map.

Year Map, Months Map, Day Map: The naming of the years, months and days can also be fully customized.

Combo Box: Up to 4 different maps can be defined at the same time. To address a certain shift map with the % placeholder it is necessary to put the corresponding number before the letter. E.g if you want to address Time Shift Map number 2 you would have to write %2T. If no number is specified then number 1 is assumed.

NOTE

In Flash jobs (USC-2/3) only the first shift/year/month/day map is available and will be used instead of the applied map number. The map has to be stored to the USC-2 card by clicking the STORE button in Optic settings→advanced.

Global Sequence Reset Times: When activated, serial numbers which are assigned to a global sequence can be reset. Chose the sequence in the drop down menu and set the desired reset date, time and period. See chapter Global Sequences Reset Times Dialog.

Job Save/Load Dialog: Opens a dialog where the defaults for saving and loading of jobs can be edited. See chapter Job Save/Load Dialog.

6.5.1 Time Shift Map

6.5.1.1 Shift Map

The following dialog allows to define a string mapping for working shifts. The according placeholder of Date Time is automatically replaced with these string definitions regarding the time of day. For how to define this placeholder, see the format definitions of Date Time object. The dialog can be reached by *Menu bar → Settings → System → General → Shift Map*.

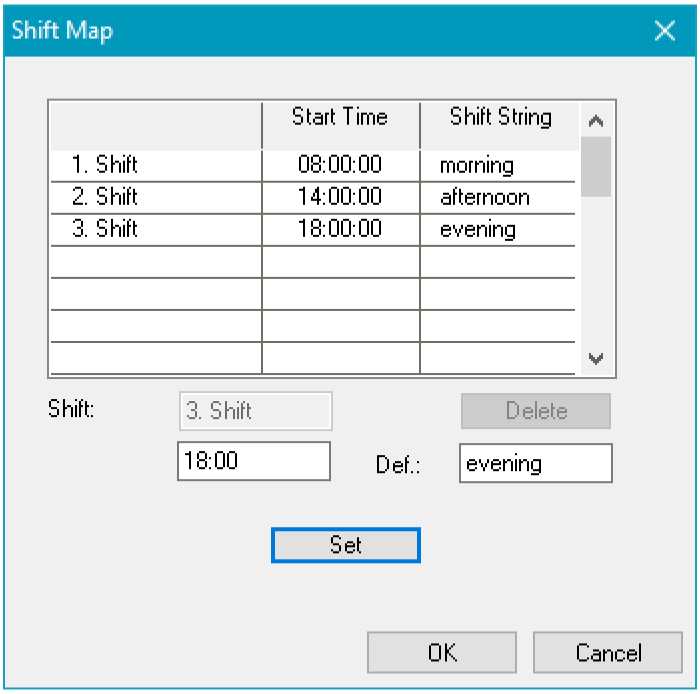


Figure 52: Shift Map Dialog

Shift: Shows the working shift which is currently selected.

Delete: Deletes the currently selected working shift.

Shift Start: Defines a start time for the selected working shift. The shift ends with the definition of a following shift.

Time format: hour:minute:second, Range: hour 0-23, minute 0-59, second 0-59.

Def.: Defines a string for the shift placeholder of date time to the given shift time.

Set: Creates a new shift if it is not defined yet, otherwise overwrites the selected working shift. The newly defined working shift gets numbered and sorted according to its start time.

NOTE *In Flash jobs (USC-2/3) only the first shift map is available and will be used instead of the applied map number. The map has to be stored to the USC-2 card by clicking the STORE button in Optic settings→advanced.*

NOTE *The latest time shift ends at 24:00:00. Per default a new shift is added at 00:00:00 so that the shift string is used until the first user defined shift (1.Shift will be incremented to 2.Shift).*

6.5.1.2 Months Map

The following dialog allows the user to enter a customized naming for each month of the year. The dialog can be reached by *Menu bar → Settings → System → General → Month Map*.

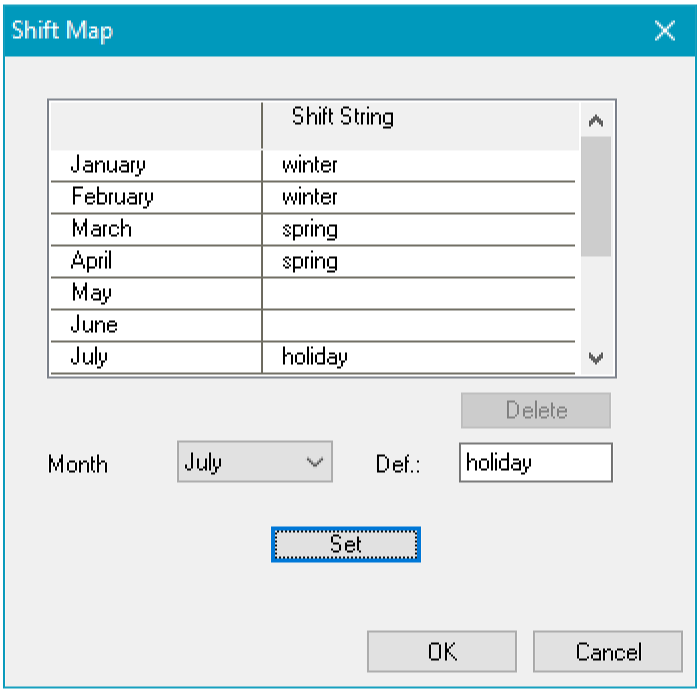


Figure 53: Months Map Dialog

Month: Tells which month has to be renamed.

Delete: Deletes the current selected working shift.

Def.: Set desired shifting text here.

Set: Enters the modification in the shift list.

In Flash jobs (USC-2/3) only the first month map is available and will be used instead of the applied map number. The map has to be stored to the USC-2 card by clicking the STORE button in Optic settings→advanced.

6.5.1.3 Day Map

The following dialog allows the user to enter a customized naming for day of a month. The dialog can be reached by *Menu bar → Settings → System → General → Day Map*.

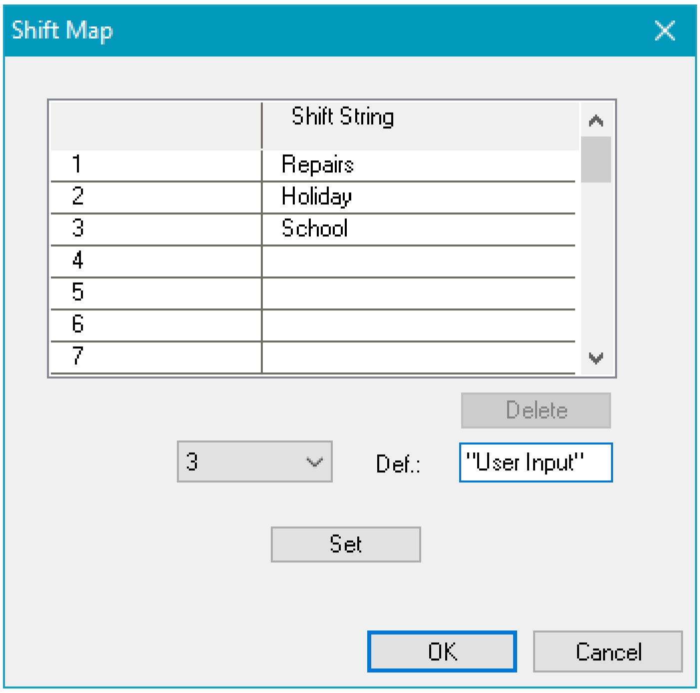


Figure 54: Day Map Dialog

Day: Tells which day has to be renamed.

Delete: Deletes the current selected working shift.

Def.: Set desired shifting text here.

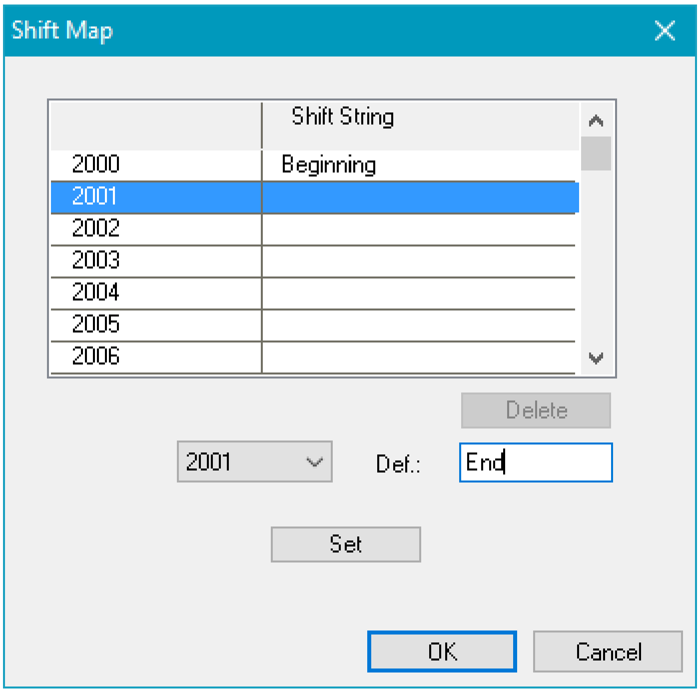
Set: Enters the modification in the shift list.

NOTE

In Flash jobs (USC-2/3) only the first day map is available and will be used instead of the applied map number. The map has to be stored to the USC-2 card by clicking the STORE button in *Optic settings→advanced*.

6.5.1.4 Year Map

The following dialog allows the user to enter a customized naming for a year. The dialog can be reached by *Menu bar → Settings → System → General → Year Map*.



Year: Tells which year has to be edited.

Def.: Define the edit string.

Set: Enters the modification in the shift list.

Figure 55: Year Map Dialog

NOTE *In Flash jobs (USC-2/3) only the first year map is available and will be used instead of the applied map number. The map has to be stored to the USC-2 card by clicking the STORE button in Optic settings→advanced.*

6.5.2 Global Sequences Reset Times Dialog

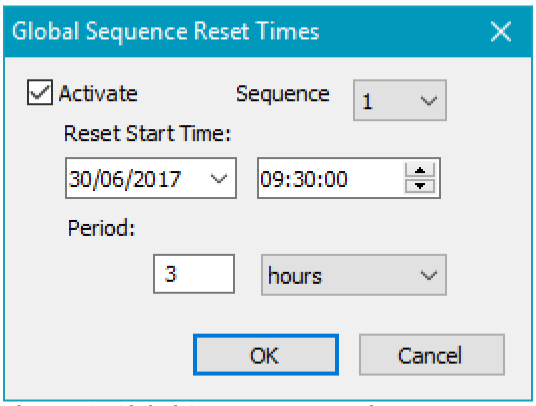


Figure 56: Global Sequence Reset Times

With this option, serial numbers that were defined as global sequences can be reset with this option. The first reset will be at the specified Reset Start Time. The following resets will be repeated after the defined Period. This reset does affect all serial numbers with the same global sequence number, which can be set in Serial Number -> Global Sequence.

6.5.3 Job Save/Load Dialog

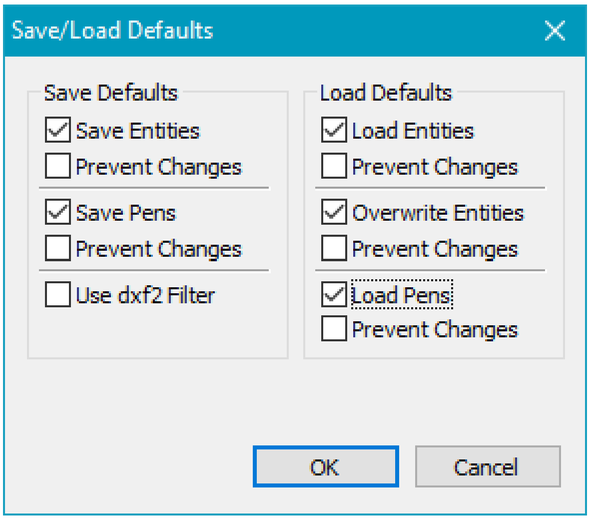


Figure 57: Save/Load Defaults Dialog

This dialog can be used to set the default state of entities and pens of a job during saving and loading. The checkboxes (Entities, Overwrite Entities, Pens) which are manipulated here can be found in the Open and Save As dialog.

In the Save/Load Defaults dialog, each field has two checkboxes:

- the first checkbox for the state of the according checkbox in the Open or Save As dialog. It has 3 states: On, Off, Gray. If Gray, the behaviour of the dialog is as usual: The last used state is used. This means there is no effect of this dialog.
- the second checkbox *Prevent Changes* to secure the setting of the according checkbox above such that it can not be changed by the user any more.

6.6 Remote

The settings for an external control are available in *Menu bar → Settings → System → Remote*:

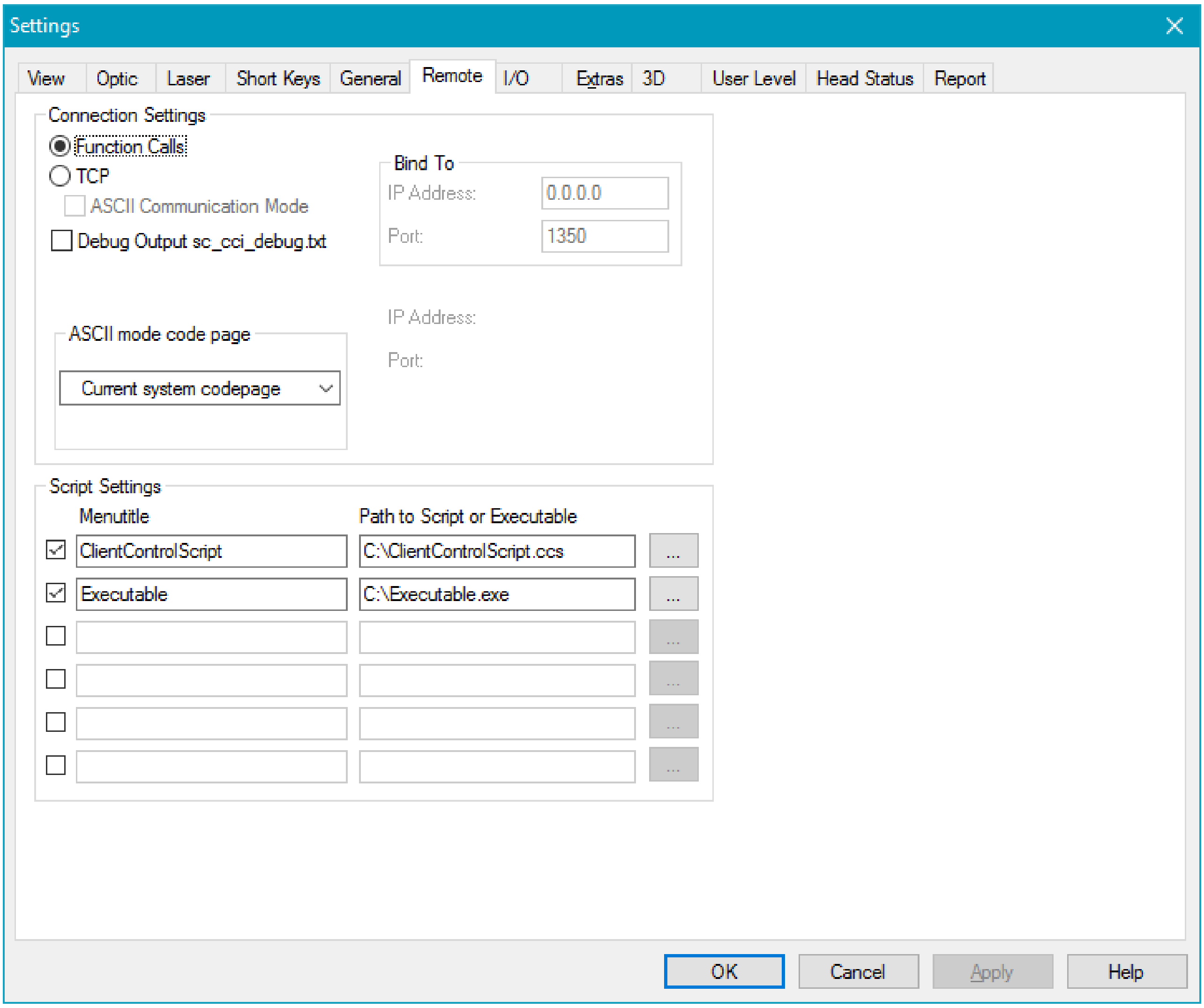


Figure 58: Remote Settings Dialog

Connection Settings: Here it is possible to define the connection method that an external CCI application uses to access the scanner application.

Function Calls: This option is only achievable inside one PC. That option has to be chosen if the controlled software and the application that controls it via the client control interface are running at the same host.

TCP: If the program should be controlled over the network using the client control interface, this option has to be selected. Normally the TCP protocol should be used. Only one TCP connection can be established at the same time.

ASCII Communication Mode: If checkbox is activated it is not necessary to send the initialization string "SAM CCI Plain\n" to start SAMLight Client Control in ASCII communication mode. This is particularly helpful if SAMLight Client Control is always used in ASCII communication mode and also for testing and debugging your own SAMLight Client Control ASCII communication application.

Debug Output sc_cci_debug.txt: See further details at CCI Debug Log.

ASCII mode code page: For TCP ASCII Communication Mode normally the current used system code page of Windows is used to send the complete command string to SAMLight. If a different code page must be used to set or get text contents (especially relevant to set a text of a text entity in another language) you can select and set the required code page from the drop-down list. After a change of the code page the *Apply* button has to clicked.

Bind To:

IP Address: This is the IP address of the PC on which SAMLight is running. Here following values are possible:

- *0.0.0.0* if the software has to be accessible from everywhere
- One of the host systems IPs if it uses more than one and if the software has to be accessible by only one specific IP (Depending on the setup of your local network, this should be something like *192.168.1.100*. Please ask your system administrator for details.)
- *127.0.0.1* if only local connections have to be accepted. This case is a more theoretical one and normally should be used for testing purposes only because for plain local connections the *Function Calls* option can be used.

Port: Number defines where the software has to be accessible. Here any value in the range of *1...65535* is possible, but it has to be noticed that a port can be used only once per IP. So for an example if there is already a web server running at a system, port number *80* cannot be used. Therefore it is recommended to use port numbers *>1024*. They are reserved for custom usage.

On the other hand the application that controls the software using the *SCAPS.ScSamlightClientCtrl* client control interface has to open its connection to the server socket configured here using ScOpenTCPConnection(). If the plain ASCII communication has to be used, opening of a TCP/IP connection depends on the operating system and the programming language. In C the appropriate function calls to do that mainly are *socket()* and *connect()*, for sending data *send()* is used and *recv()* for the reception of an answer.

The network access to the scanner application can be restricted to some specific IPs to avoid that connection attempts are successful from hosts that are not allowed to control the application. These IPs can be defined using the two files *hosts.allow* and *hosts.deny* that are located within *<SCAPS>\system* of the installation directory of the scanner application. Both files expect IP numbers in the format "*WWWW.XXXX.YYYY.ZZZ*".

Within *hosts.allow* all the IPs that are allowed to connect to the scanner application can be listed. If there are some IP numbers define here, connections are allowed only from these ones exclusively. If there are no IPs defined, connections are allowed from all IPs except the ones that are listed within *hosts.deny*. The file *hosts.deny* can contain a list of IP numbers with the format "*WWWW.XXXX.YYYY.ZZZ"* and defines which hosts are not allowed to control the application. If there a no IPs listed within this file, no hosts are forbidden explicitly to connect to the scanner application.

Script Settings: Besides the possibility to have an external application that is under the control of the user it is also possible to start such an application from within the scanner application. To do that the following settings are used:

Menutitle: Specify the name which is shown in the *Special* entry of the Menu bar. This name is used to address the application that has to be started out of SAMLight. The script can also be executed via the special menu bar.

Path to Script or Executable: This script or *.exe is executed whenever the related name in the *Special* entry in the menu bar is selected. The following programs are supported:

- *Executables* (*.exe) that are started and that should access the *Client Control Interface* of the application
- *Client Control Scripts* (*.ccs) that contain ASCII CCI commands that are interpreted and executed. These scripts may contain comments using # as a delimiter.

NOTE

After these scripts are executed within the context of the scanner application there is no need to perform any initialization or opening of a connection. The commands to control the application can be used directly.

6.6.1 CCI Debug Log

If this checkbox is active, a debug text file is written after each Client Control Interface (CCI) command. When SAMLight starts the file *<SCAPS>\system\sc_cci_debug.txt* will be deleted. The first CCI command after SAMLight start creates the text file and writes the debug information into it and saves the file. All further CCI commands (in the same SAMLight session) will append the file with their debug information and save the file. The timing resolution in sc_cci_debug.txt is limited to the resolution of the system timer, which is typically in the range of 10 ms to 16 ms.

The following information is written in the file:

- Date
- Time
- ProcessTime: Time after the command has run.
- CCI Command: The sent command including all set parameters.
- CCI Return: 0: Command failed.
  1: Command run successful
- CCI Return Value: Possible return value from get functions.

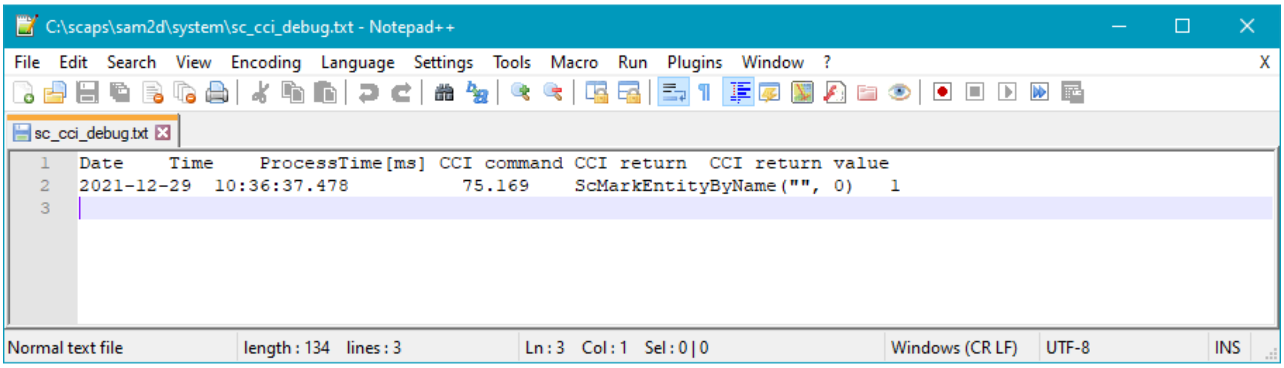


Figure 59: CCI Debug Log

6.7 I/O

The settings dialog described here can be reached by selecting the menu item *Settings → System → I/O*.

Use this window to set special inputs and outputs or to make settings for the SAMLight Job IO Selection mode.

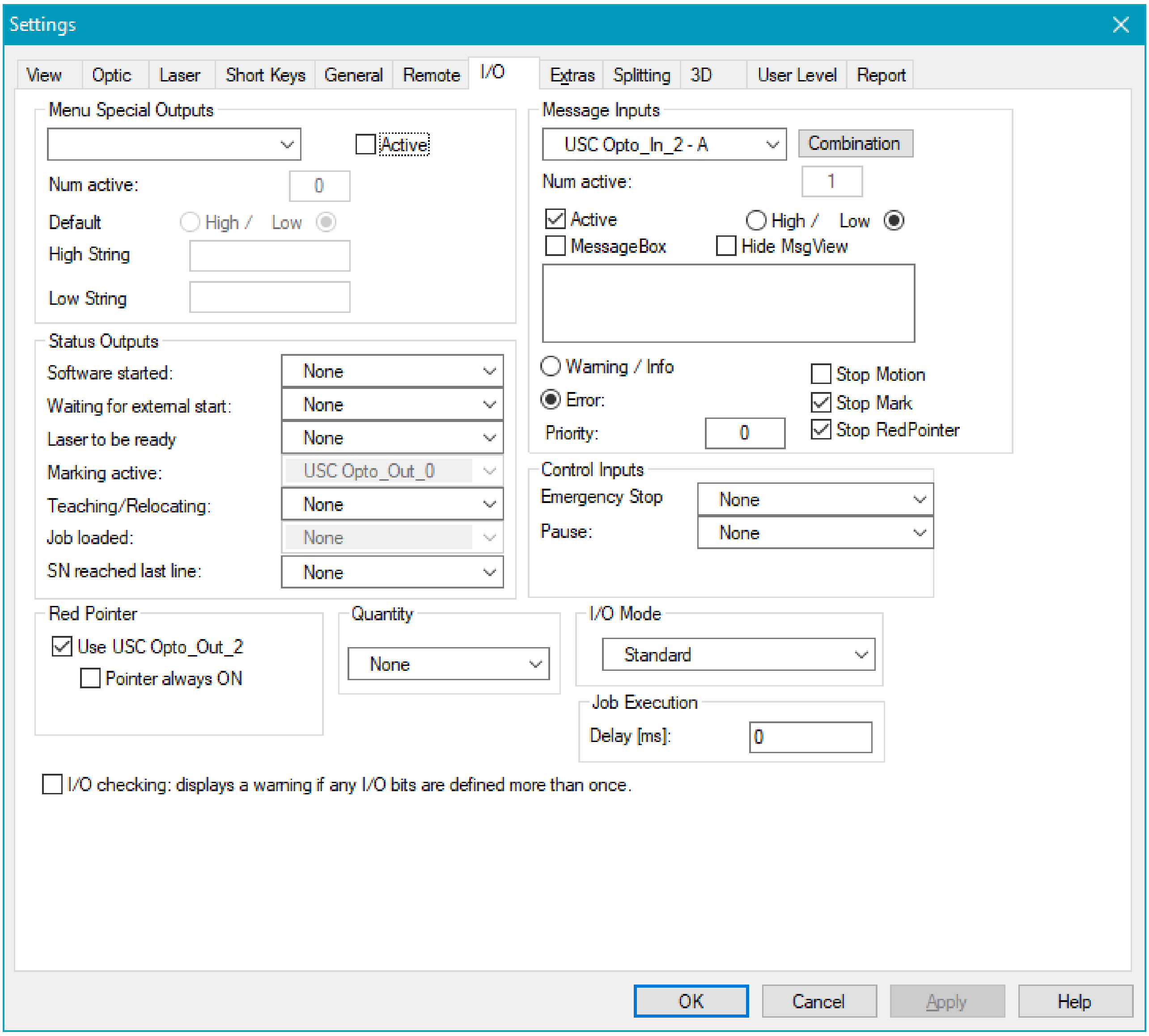


Figure 60: IO Settings Dialog

Menu Special Outputs: It is possible to insert new menu items for switching I/Os on and off. The bits of the I/O port which will be controlled from special menu points can be defined in the dialog. If one bit is selected and the *Active* checkbox is enabled a string (*Num Active*) which indicates the current state of the bit can be defined. Under the following menu the items for switching the I/Os on and off will be inserted.

Figure 61: Mark Special Outputs Menu

Default: Defines the default state of the special output that is shown in the menu.

High String: Defines the String that is shown when the special output is set to Low.

Low String: Defines the String that is shown when the special output is set to High.

Status Outputs: This block defines state I/Os that can be switched on an off according to specific program and usage actions. Using the combo boxes it is possible to assign a special output pin for such an action. Using this functionality an integration of external equipment can be done.

Software started: The selected bit is set to *high* as long as the software is running.

Waiting for external start: Set to *high* if in trigger mode. This changes to *low* during marking.

Laser to be ready: Set to *high* if trigger or mark dialog is open. This stays *high* during marking.

Marking active: Set to *high* during marking of a job. Only for USC cards this signal is hardware controlled. For RTC cards this signal is software polled with the result of jitter. For proper I/O handling, this signal has to be used, not the RTC busyout.

Teaching / Relocating: If an output pin is defined here it is set to *high* on every time the teaching or relocating dialog is active.

Job loaded: The selected bit is set to *high* if a non empty job is loaded in the View.

SN reached last line: If a file is assigned to a serial id and the last line of this file has been reached this bit is set. To change the state afterwards you have to reset the serial id or load a new job. Instead or in addition you can pop up a message box if the last line of the file has been marked.

Red Pointer:

Use Bit 3: Bit 3 of the I/O port is used to indicate that the red pointer is active.

Pointer always ON: The Red Pointer stays always on (while SAMLight is running) if this flag is activated.

Invert: This option is only available for RTC cards and USC-1. It inverts bit 3 for controlling the red pointer.

Quantity: Here an output can be defined that goes high when a predefined number of mark quantities has been reached. The predefined number of mark quantities can be set up in Mark→Counter→Set Quantities.

Message Inputs: Input bits can be used to cause a message output. The selected bit must be activated to send the defined message by the *Active* check box. The message appears if the selected bit is either *high (H)* or *low (L)* according to the selected radio button. An error or a warning message is displayed in the message view dialog as well as in the status bar. If *Hide MsgView* is activated, the message view is invisible.

Stop Motion: If Message Input is executed the Motion is stopped. Does not influence any homing during startup. (only available for USC-3)

Stop Mark: If Message Input is executed the Marking is stopped.

Stop RedPointer: If Message Input is executed the red pointer is stopped.

Priority: For each activated input (Warning/Info and Error), a priority can be set. If more than one input is active at the same time, the one with the highest priority will be shown in the status bar. Error always wins over Warning/Info.

- RTC input bits:
  - 0 .. 15 (Extension 1 connector, **Digital_In 0 .. 15**)
- USC input bits:
  - 3 .. 6 (37-pin connector, **Opto_In2 .. 5**)
- USC-2/3 additions:
  - 7 .. 16 (Extension connector, **Digi_In0 .. 9**)
  - **Status #0 .. #19**: These status bits refer to the 20-bit word of the XY2-100 feedback from the scan head. The user may define a unique message / error for any of these bits.
  - Error warning info: These warnings can be reset in the status bar.
    - **Out of data with laser on (OD)** is set when a buffer underrun occurred.

- **Out of field (OF)** is set when a position overflow (head specific optic matrix, MOTF, Analog In, USC-3 Wobble) occurred.
- **Overflow MOTF offset (OM)** is set when a MOTF offset is reached too early (e.g. still marking).

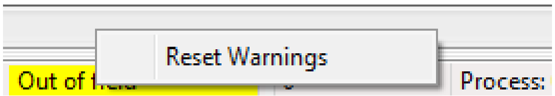


Figure 62: Reset OF, OD and OM by right-click followed by a left-click on Reset Warning

Num active: Counts the number of activated Message Inputs.

Combination: This button opens a dialog *Message Input Combination*, which can be used to combine two or three input bits.

Control Inputs:

Emergency Stop: If an input is selected here it is handled as a watch for an emergency stop condition. That means if the selected input goes to low all marking operations are stopped and a special emergency stop dialog is displayed. This dialog blocks all other operations and stays in front of the screen until normal operation is resumed by pressing the "Resume Operation" button. This button becomes active and can be pressed only after the selected emergency stop input goes to high. When the resume button is pressed the application is brought back into its initial state. That means the connected motion controllers are driven to their home position automatically before the emergency stop dialog disappears and before a user can continue with normal operation.

NOTE *It is recommended to connect the appropriate input pin before this option is enabled. An open input normally is recognized as a low-signal so that leaving the I/O settings dialog would put the application in the emergency stop state immediately.*

Figure 63: Emergency Stop

Pause: This functionality is only available with an USC-2 or an USC-3 card. Here you can select an input bit to pause the job. The laser is switched off immediately. After the restart, the job continues at the same position, the laser will turn on again after the next mark command, after a PolyEnd or after a jump command.

I/O Mode: There are three SAMLight I/O modes (which differ from the Flash JobIOSelection mode) that can be chosen:

- **Standard**: the settings described above can be made including the freely definable Message Inputs.
- **SAMLight JobIOSelect**: SAMLight Job IO Selection mode. Opto_In_2 (LSB) .. Opto_In_5 (MSB) are used to load jobs (up to 15 jobs).
- **SAMLight JobIOSelect Ext**: SAMLight Job IO Selection mode. Digi_In_0 (LSB) .. Digi_In_7 (MSB) are used to load jobs (up to 255 jobs) (USC-2/-3 only).

For more information see SAMLight Job IO Selection.

Message inputs can also be used in Job IO Selection mode. These must be set and activated in standard mode.

In Job IO Selection mode it can happen that the input loads a job as well as being defined as a message input.

Job Execution:

Delay [ms]: Defines an execution delay which is the time between a mark output signal is given and the execution. Only for USC cards the marking active signal is hardware controlled. For RTC cards the marking active signal is software polled with the result of jitter.

Input and Output bit values:

USC cards:

Input pin	Output pin	Bit value
Opto-insulated Inputs and Outputs		
Opto_In_0 [a]	Opto_Out_0 [c]	1
Opto_In_1 [b]	Opto_Out_1	2
Opto_In_2	Opto_Out_2 [d]	3
Opto_In_3	Opto_Out_3	4
Opto_In_4	Opto_Out_4	5
Opto_In_5	Opto_Out_5	6
Following pins are only available for USC-2/3		
Digital Inputs and Outputs		
Digi_In_0	Digi_Out_0	7
Digi_In_1	Digi_Out_1	8
Digi_In_2	Digi_Out_2	9
Digi_In_3	Digi_Out_3	10
Digi_In_4	Digi_Out_4	11
Digi_In_5	Digi_Out_5	12
Digi_In_6	Digi_Out_6	13
Digi_In_7	Digi_Out_7	14
Digi_In_8	Digi_Out_8	15
Digi_In_9	Digi_Out_9	16
Stepper Inputs and Outputs		
Stepper_In_0	Stepper_Out_0	17
Stepper_In_1	Stepper_Out_1	18
Stepper_In_2	Stepper_Out_2	19
	Stepper_Out_3	20
	Stepper_Out_4	21
	Stepper_Out_5	22

Table 11: I/O bit values for USC cards

RTC cards:

Input pin	Output pin	Bit value
Digital Inputs and Outputs		
Digi_In_0	Digi_Out_0	1
Digi_In_1	Digi_Out_1	2
Digi_In_2	Digi_Out_2	3
Digi_In_3	Digi_Out_3	4
Digi_In_4	Digi_Out_4	5
Digi_In_5	Digi_Out_5	6
Digi_In_6	Digi_Out_6	7
Digi_In_7	Digi_Out_7	8
Digi_In_8	Digi_Out_8	9
Digi_In_9	Digi_Out_9	10
Digi_In_10	Digi_Out_10	11
Digi_In_11	Digi_Out_11	12
Digi_In_12	Digi_Out_12	13
Digi_In_13	Digi_Out_13	14
Digi_In_14	Digi_Out_14	15
Digi_In_15	Digi_Out_15	16

Table 12: I/O bit value for RTC cards

[a]: Reserved for trigger start

[b]: Reserved for external stop

[c]: Reserved for marking active

[d]: Only reserved for the red pointer, if the red pointer is active

6.7.1 Message Input Combination

Combine Inputs: The input bits intended to be combined are selected via the drop down menus (Input bit A, Input bit B, Input bit C, Input bit D). For a combination of only two bits via the first two drop down menus, chose -1 for Input bit C. After clicking on *Combine Inputs*, a combination of the selected bits will appear in the lower drop down menu and in the List of all Message Input bits.

We are using the following shortcut: OI = *Opto_In* and DI = *Digi_In*.

The combination of two bits creates four options (*HH, HL, LH, LL*), the combination of three bits creates eight options and the combination of four bits creates 16 options, respectively. Each bit can only be used once, either in one single combination or directly without a combination.

Remove Combination: In order to cancel an existing combination, select the combination in the drop down menu and click on *Remove Combination*. Please note that this action is irreversible!

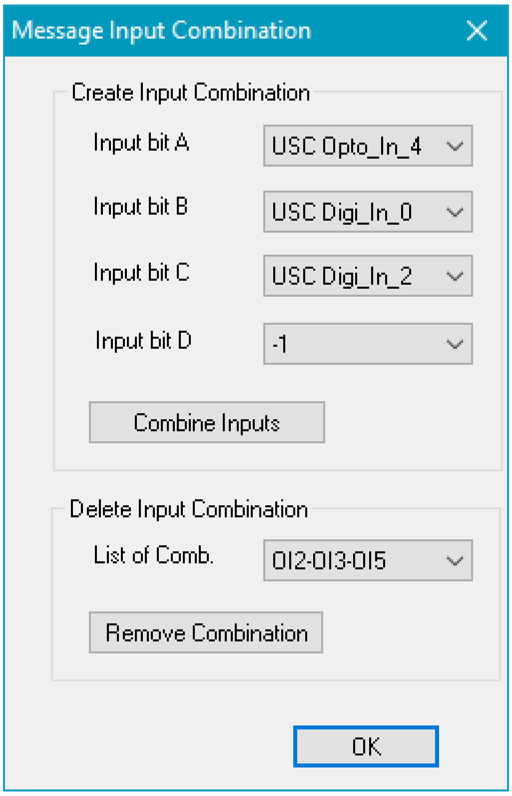


Figure 64: Message Input Combinations Dialog

6.8 Extras

The settings dialog described here can be reached by selecting the menu item *Settings* → *System* → *Extras*.

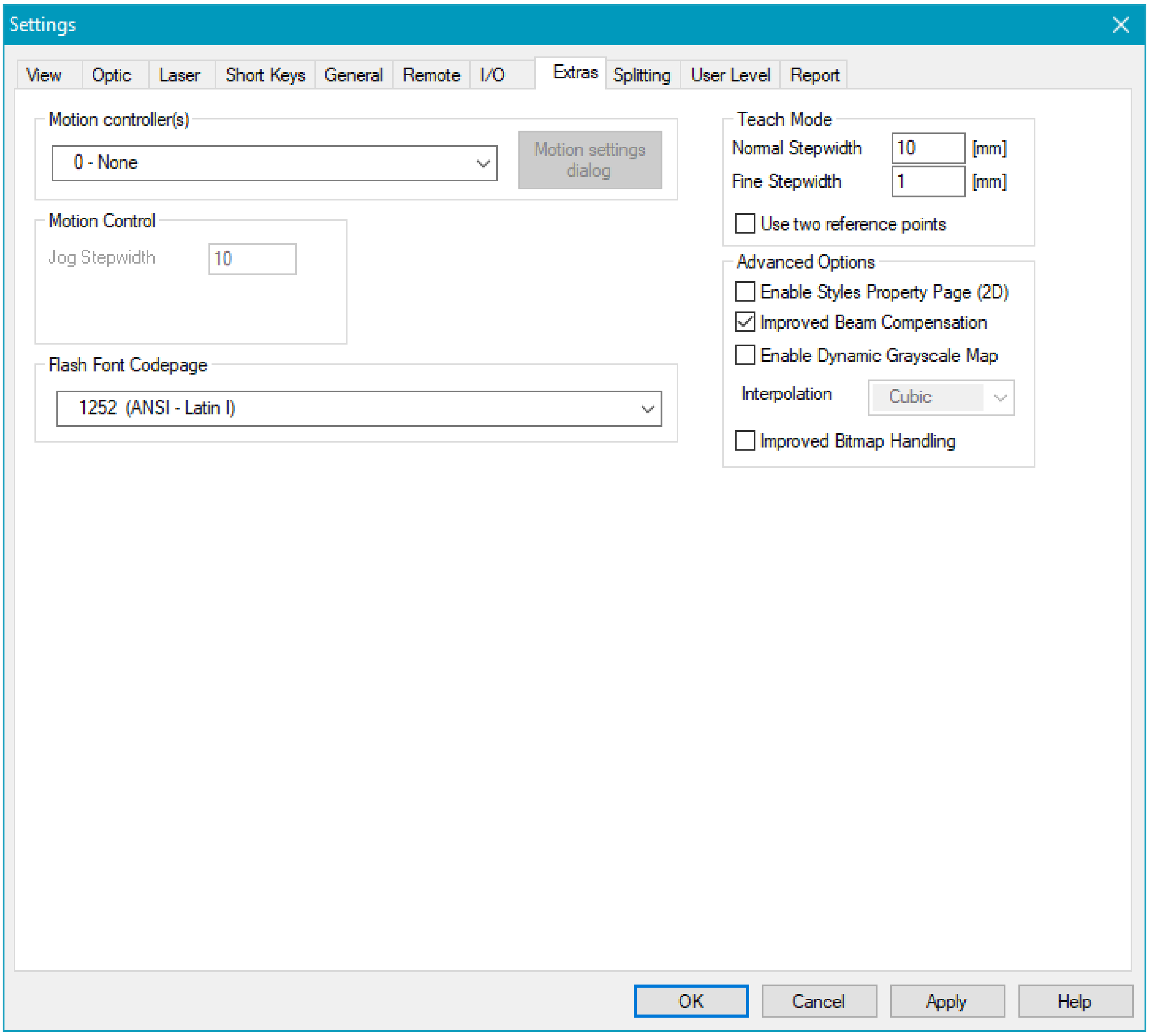


Figure 65: Extras Settings Dialog

Motion Control: This area is related to motion control elements.

Motion Type settings: Here you can choose the motion type which is corresponding to the value in the file ***sc_motion_settings.txt***. Please refer to the chapter Motion Controller for further information. The motion settings dialog is coming soon. If you choose entry 0 here the motion control will be deactivated. For motion type 8 = Stepper motor there is a motion settings GUI which can be accessed by the button "Motion settings dialog".

Jog Stepwidth: Default value for the step width that is used for the big Jog movements (-- or ++).

Flash Font Codepage: Here you can choose which font code page should be saved to the flash of a USC-2 card. This is necessary if you want to change the dynamic text of a serial number (or barcode, DateTime etc.) on the flash.

Teach Mode: This block is related to the teaching / relocating mode. Using this mode it is possible to teach reference positions for a job that are related to a specific work piece. When this work piece was exchanged and the new one has a different position and / or rotation angle comparing to the preceding one, it is possible to modify the job so that it fits to the new position. To do that, the relocating function can be used. The parameters that can be defined here influence the teaching / relocation behaviour in following ways.

Normal Stepwidth: Defines the normal step width that is used in the teaching / relocating dialog to move the laser pointer

Fine Stepwidth: Defines the smaller, more exact step width that is used in the teaching / relocating dialog to move the laser pointer

Use two reference points: When this box is checked, the relocation can be done using two reference points. With one point it is possible to equalize a work pieces translation in parallel to the preceding position only. When two reference points are used, a rotation can be calculated too. The new position of the work piece doesn't need to be exactly parallel to the preceding one.

NOTE *An output pin that is toggled every time the teaching / relocating dialog is opened and closed can be defined in IO Settings (e.g. switch a camera on and off).*

Advanced Options:

Enable Styles property page (2D): Enables the 2D styles property page.

Use improved BC: Activates improved Beam Compensation Algorithm for Hatching and Beam Comped Copy.

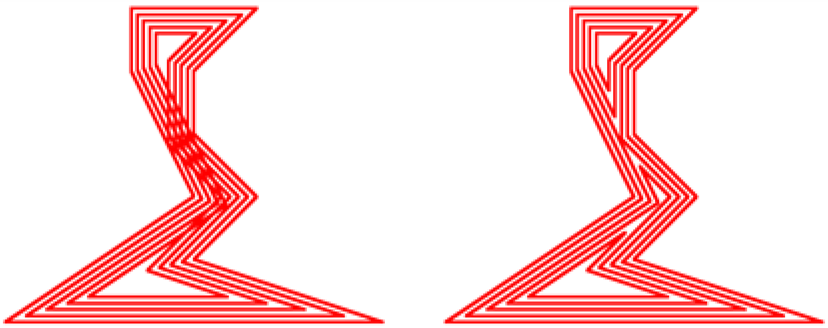


Figure 66: Left: old BC Algorithm, right: improved BC Algorithm

Enable Dynamic Grayscale Map: Activates a dynamic System PixelMap. Three different modes of interpolation are available in the drop down menu: Linear, Cubic or Hermite.

Improved Bitmap Handling: Enables the bitmap advanced mode. Further settings can be made at *Pen Settings → Bitmap*.When activated, several settings necessary for bitmap marking will automatically be made which formerly had to be made manually. In addition, the bitmap tab in your pen will be adapted to present clear parameters for bitmap optimization.

6.8.1 Flash Font Codepage

The generation of a font in Flash depends on 2 parts:
1. Flash Font Codepage applied in SAMLight
2. font of the entity itself

In more detail, each single ASCII position of the Flash Font Codepage (0-255 positions) is taken and the Unicode number of its character is looked for in the applied font of the entity (0-65535 positions). This means that from the entities point of view, each character's Unicode number needs to be present in the selected Flash Font Codepage for a valid entry on flash.

Three cases can occur:

1. Identical case - good case

If a Unicode number of the Flash Font Codepage is found in the applied font of the entity, it is put at the corresponding position in the resulting font on flash (see example in figure 67: ASCII position 0xA3 in the Flash Font Codepage (Windows-1250) is occupied by Unicode character 0x141 (L), this Unicode character is present in the applied font and is thus mapped to ASCII position 0xA3 in flash.).

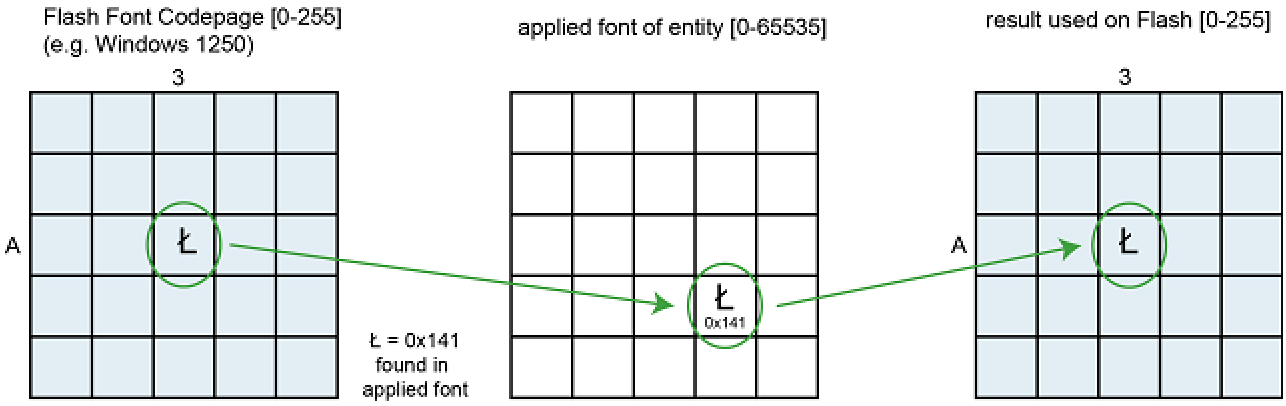


Figure 67: Example for successfully mapping Unicode character 0x141 from Windows-1250 Flash Font Codepage

2. Mixed case - good if on purpose

If a Unicode number of the Flash Font Codepage is found in the applied font of the entity, it is put at the corresponding position in the resulting font on flash. This does not mean that the vectors of the characters need to be the one of the Unicode number. You can easily modify the character in your applied font of the entity using for example the font converter (see example in figure 68: ASCII position 0xA3 in the Flash Font Codepage (Windows-1250) is occupied by Unicode character 0x141 (L). This Unicode character 0x141 is found in the applied font. But in the applied font, the content of position 0x141 was exchanged with a smiley in the font of the entity. Because the smiley is located at 0x141, it is recognized as the correct character and is put at position 0xA3 in the resulting font in flash.).

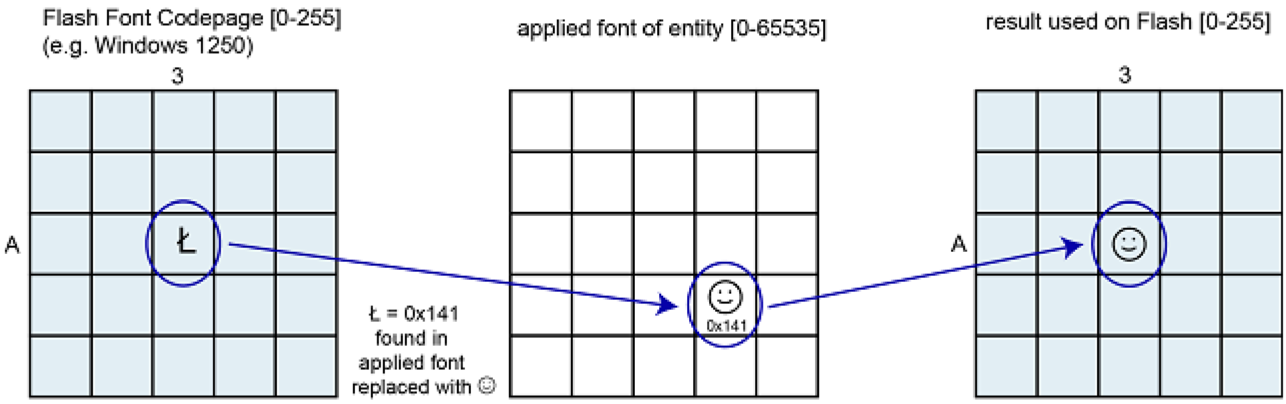


Figure 68: Example for successfully mapping Unicode character 0x141 from Windows-1250 Flash Font Codepage to a custom designed character

3. Missing case - bad case

a. If a Unicode number of the Flash Font Codepage is not found in the applied font of the entity, the corresponding ASCII position stays empty in the resulting font on flash (see example in figure 69: ASCII position 0xA3 in the Flash Font Codepage (Windows-1250) is occupied by Unicode character 0x141 (L) which is not present in the applied font. Thus, ASCII position 0xA3 stays empty in flash.).

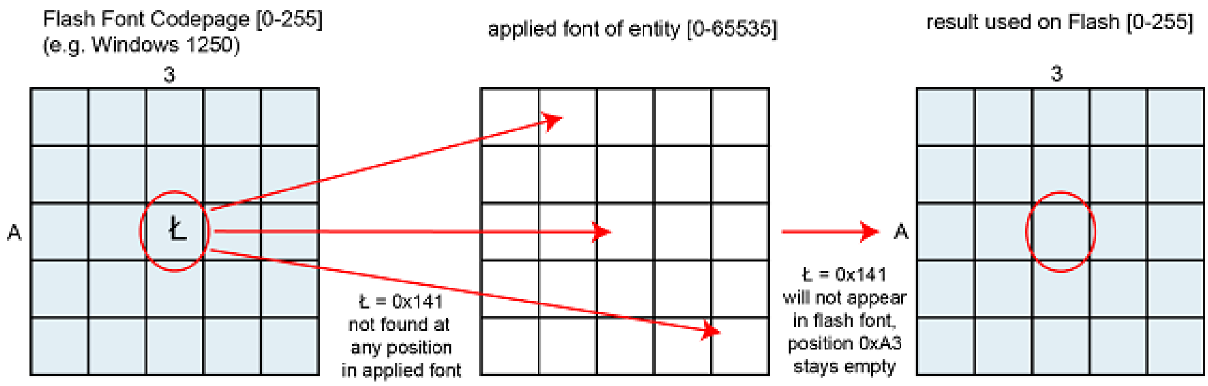


Figure 69: Example for not mapping Unicode character 0x141 from Windows-1250 Flash Font Codepage

b. If a Unicode number of the applied font of the entity is not found in the Flash Font Codepage, it will not be present in the result on flash (see example in figure 70: Unicode character 0x192 (ƒ) of the applied font of the entity is not present anywhere in the Flash Font Codepage (Windows-1250).).

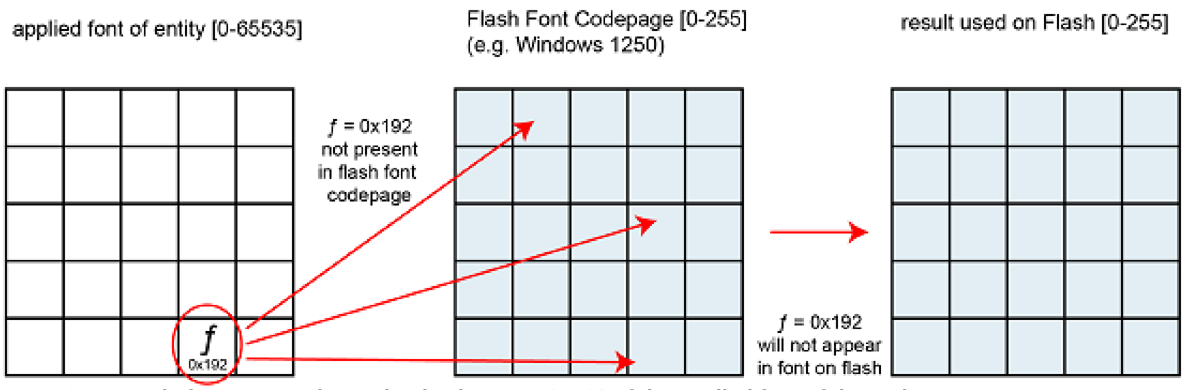


Figure 70: Example for not mapping Unicode character 0x192 of the applied font of the entity

NOTE

The Flash Font Codepage is only taken for the generation of characters in Flash. For the generation of characters in SAMLight, the font codepage of the windows system (applied for non-Unicode programs) is taken. For central Europe, this is usually the most common one: Windows-1252.

6.9 Splitting

The settings dialog described here can be reached by selecting the menu item *Settings → System → Splitting*.

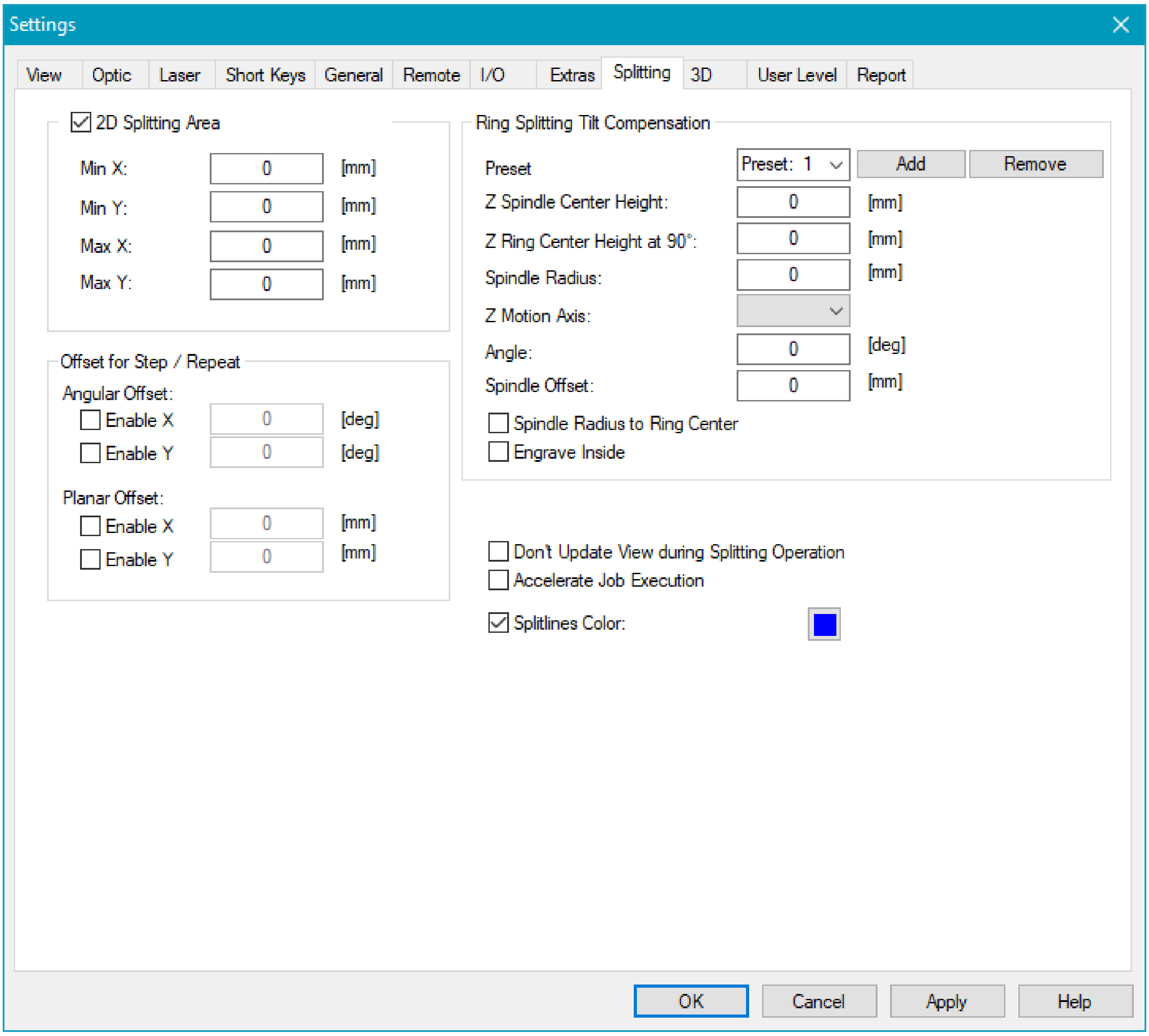


Figure 71: Splitting Settings Dialog

2D Splitting Area: All split entities have to be located within this defined splitting area.

Min X: Minimum X coordinate of the splitting area.

Min Y: Minimum Y coordinate of the splitting area.

Max X: Maximum X coordinate of the splitting area.

Max Y: Maximum Y coordinate of the splitting area.

Offset for Step / Repeat:

Angular Offset: Define a global angular offset in X-/Y- direction.

Planar Offset: Define a global planar offset in X-/Y- direction.

Ring Splitting Tilt Compensation: This feature is to be used with Ring Splitting. Here you can type in the key value of your equipment so that SAMLight can calculate the correct compensation.

Preset: You can save different preset for different equipment. You can enter a name, save the settings with *Add* and delete the preset with *Remove*.

Z Spindle Center Height: See in figure below. (orange - Z ring center height)

Z Ring Center Height at 90°: See in figure below. (light blue - Z spindle center height)

Spindle Radius: See in figure below. (blue - Spindle radius)

Z Motion Axis: Choose the axis that should be used for the z dimension. Only position and no angular axis can be used.

Angle: See in figure below. (red - Z tilt compensation angle)

Spindle Offset: Can be used to set an offset along the spindle direction.

Spindle Radius to Ring Center: If this flag is this enabled, the value of the spindle radius is measured to the ring center instead of the edge of the ring.

Engrave Inside: If this flag is enabled, it is possible to engrave inside of the ring.

Don't update View during splitting operation: If this checkbox is selected the View2D is not updated during marking of all split parts.

Accelerate Job Execution: Improves the total marking time if multiple split entities are included in the job. With this checkbox activated, there will not be any internal start execution triggers for each split entity. There will be one global internal start trigger. There will be no information in the mark dialog telling moving or which split part is currently executed.

Splitlines Color: Choose the color of the split lines. If the checkbox is unchecked, the color of pen #1 will be used for the split lines.

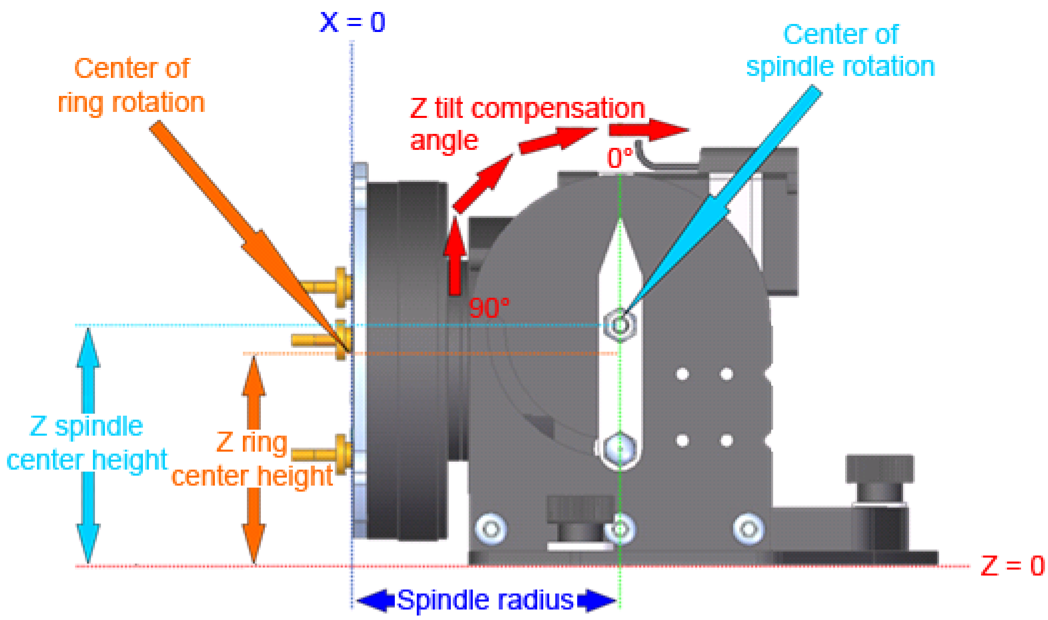


Figure 72: Ring Splitting Tilt Compensation

6.10 3D

NOTE

This option is only available for SAM software with the SAM3D option.

The following dialog can be reached by Menu item *Settings → System → 3D* and covers several 3D marking functionalities like they are useful e.g. for rapid prototyping.

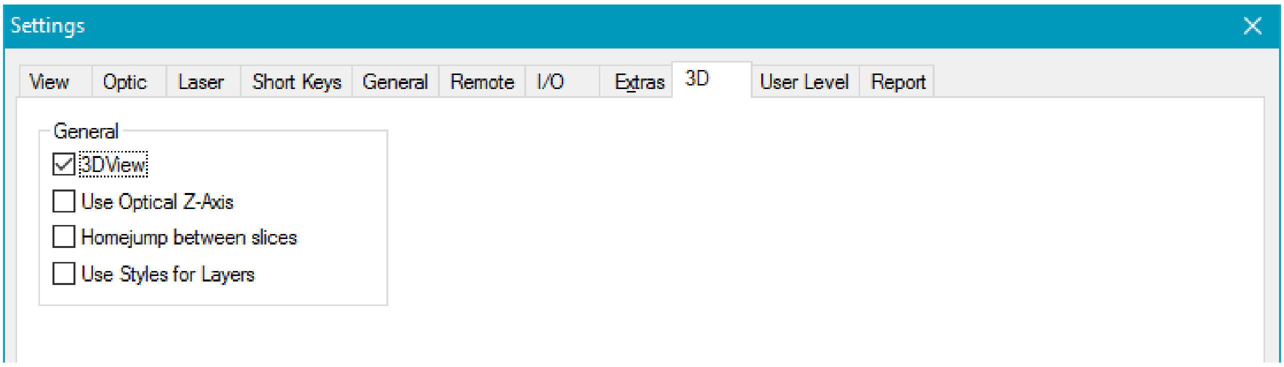


Figure 73: 3D Settings Dialog

General:

3DView: Enables / Disables SAM3D mode. The software must be restarted for the change to take effect.

Use Optical Z-Axis: Check this option, when you want to shift the focus optically with a 3D scan head.

Home jump between slices: Performs a home jump after each marked slice. If it is checked and *Settings → System → Optic → Home Position* is disabled, there will not be a home jump, but the laser power of the HomeJumpStyle is set after each marked slice.

Use Styles for Layers: Activates the Styles Property Page for SAM3D.

6.11 User Level

The following dialog can be reached by Menu item *Settings → System → User Level*.

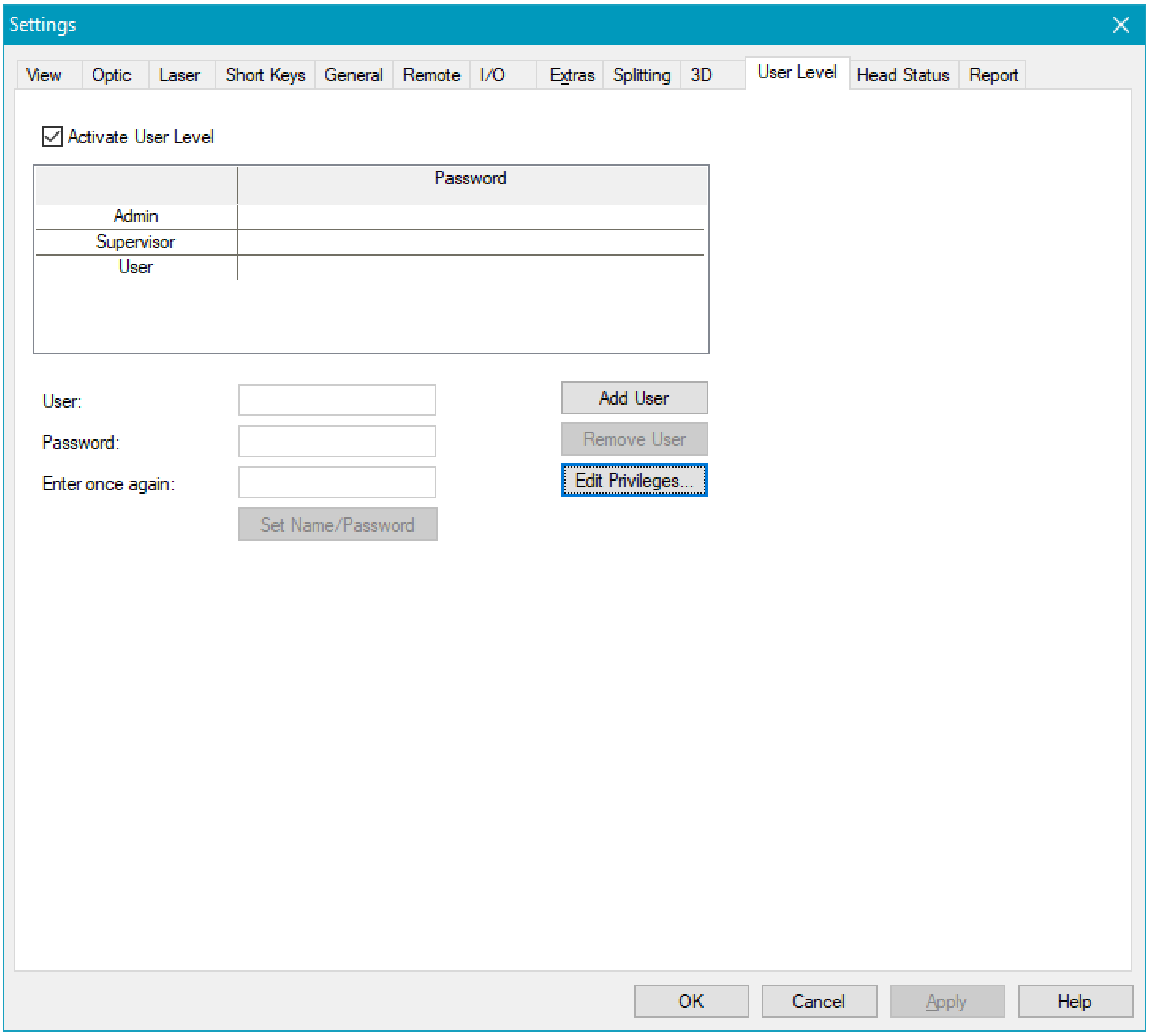


Figure 74: User Level Settings Dialog

NOTE *The following entries are possible only if the right for password assignment is given, see* *Access Rights*.

Ask for User Password: If checked the user is asked for the user name and the password before the software starts. If it is not checked the software starts with full functionality.

User: Displays the selected user.

Password / Enter once again: The password needs to be entered twice.

Set Name/Password: Applies the password that is entered in the edit field to the selected user and changes the name of the user if wanted.

Add User: Adds a new User to the list. Up to 30 different users can be defined.

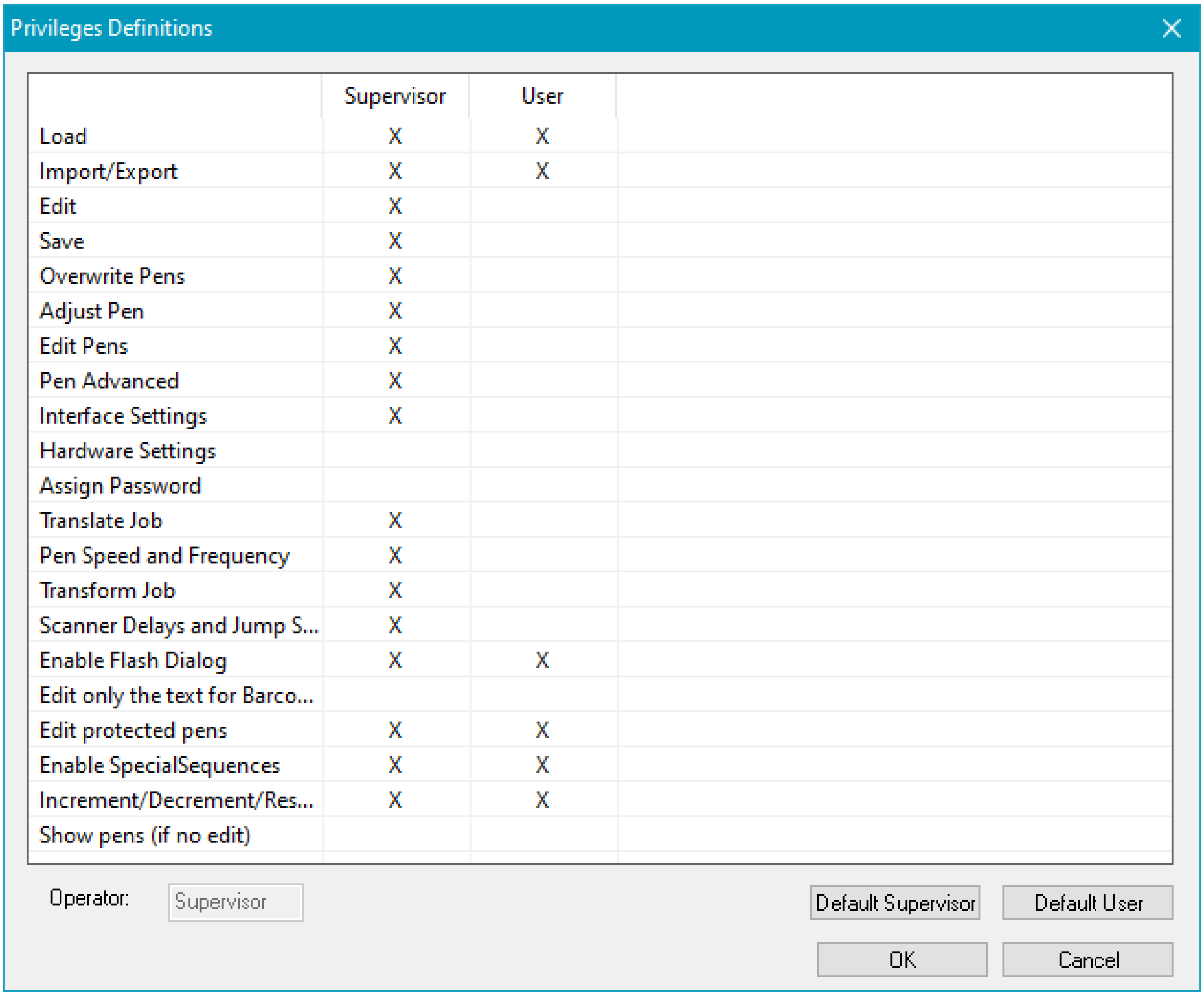
Remove User: Removes User from the list.

Edit Privileges: Opens a dialog where it is possible to for define access user rights.

If a user has no password this user will be a default user. For a login with an invalid password the default user is taken. If more than one default user exists, the first one from the list is taken.

6.11.1 Access Rights

The following dialog opens by clicking on the Edit Privileges button on the User Level page. Up to 30 Users can be specified here. The column size will be stored.


Privileges Definitions

	Supervisor	User
Load	X	X
Import/Export	X	X
Edit	X	
Save	X	
Overwrite Pens	X	
Adjust Pen	X	
Edit Pens	X	
Pen Advanced	X	
Interface Settings	X	
Hardware Settings		
Assign Password		
Translate Job	X	
Pen Speed and Frequency	X	
Transform Job	X	
Scanner Delays and Jump S...	X	
Enable Flash Dialog	X	X
Edit only the text for Barco...		
Edit protected pens	X	X
Enable SpecialSequences	X	X
Increment/Decrement/Res...	X	X
Show pens (if no edit)		

Figure 75: Access Rights Dialog

The features can be enabled or disabled by clicking on the table fields.

Default Supervisor: Sets the supervisors default settings to the selected operator.

Default User: Sets the users default settings to the selected operator.

Access Rights:

Load: Allows to load job files

Import/Export: Allows import and export.

Edit: Allows to setup and edit a job.

Save: Allows to save a job.

Overwrite Pens: Allows to overwrite pens.

Adjust Pen: Allows to set a pen to an object.

Edit Pens: Allows to edit pens.

Pen Advanced: Enables the Advanced... button on the mark property page.

Interface Settings: Enables following property pages of *Settings → System*: View, ShortKeys, General, Extras

Hardware Settings: Enables following property pages of *Settings → System*: Optic, Laser, IO-Settings, Remote

Assign Password: Enables property page User Level of *Settings → System*.

Translate Job: Allows to translate entities even if the Edit privilege is deactivated.

Pen Speed/Freq: Allows to modify the Speed and the Frequency of the current pen.

Transform Job: Allows to Translate, Scale, Mirror and Rotate entities even if the Edit privilege is deactivated.

ScannerDelaysAndJumpSpeed: Allows to change Scanner Delays and Jump Speed if Pen Speed/Freq. is activated also.

Enable Flash Dialog: Enables the Flash Dialog

Edit only the text for Barcodes, SerialNums and Text: Allows to edit the text in the property page Text2D but not the rest of the parameters on this page.

Edit protected pens: Allows to change the status of Protected Pens.

Enable SpecialSequences: Allows to select Special Sequences.

Increment/Decrement/Reset Sequence/Serialnumber: Allows to Increment, decrement and reset a serial number.

Show pens (if no edit): Allows to see the Pens on the mark property page.

6.12 Card

These settings for RTC cards are described in section Card Settings.

6.13 Trigger

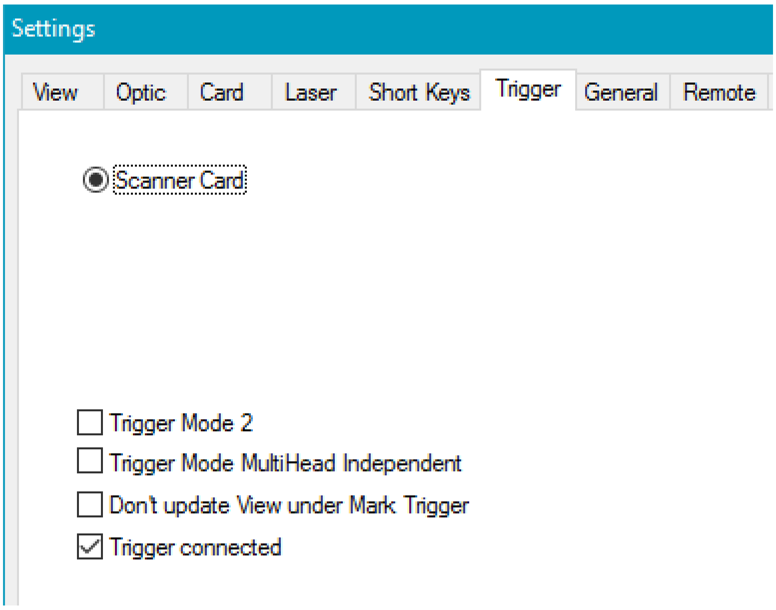


Figure 76: Trigger Dialog

Trigger Mode 2: Serial numbers and Date Time objects are not being update between marking by trigger. This allows shorter delay between triggering. Trigger Mode 2 cannot be used together with endless repetitions (like mark loop count -1).

Trigger Mode MultiHead Independent: Only available for MultiHead systems. This function allows to trigger the cards independently from each other when Mark→Trigger dialog is used. This might be useful if one of the scan heads has to mark for example 5 objects while the other scan head will only mark one object at the same time. This function will not work together with serial number objects nor with DateTime objects.

Don't update View under Mark Trigger: The view area will not be updated while the mark trigger dialog is open.

Trigger connected: This flag allows to use the trigger mode.

NOTE *This special trigger mode is only available for RTC cards.*

6.14 Report

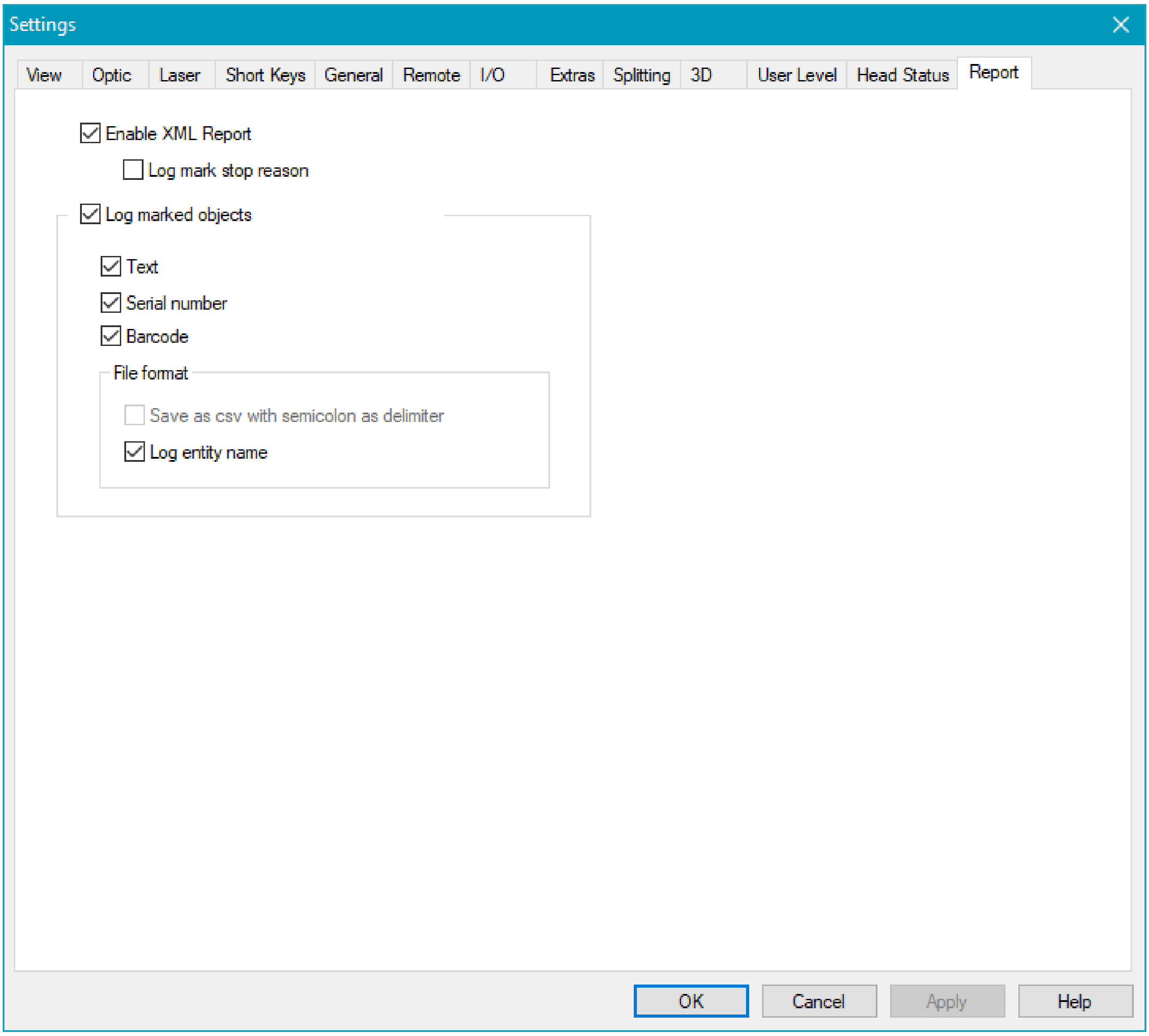


Figure 77: Report Dialog

Enable XML Report: Enable the report file output. If active each marking produces an output to a file named <Date>_<JobPath>.txt or .csv in which the marked objects are listed. The file will be located at *<SCAPS>\reportfiles*.

> **Log mark stop reason:** Extends the XML report file to record possible outer influences, which may stop the marking procedure.

Log Marked Objects: This feature can be used in combination with the XMP Report or without. If *Enable XML Report* is checked then the report file will be extended. If *Enable XML Report* is unchecked then a new file is created at the directory *<SCAPS>\reportfiles* in **.txt* format, containing the time and marked object.

> **Text, Serial Number, Barcode:** Choose which entities should be included in the log.
>
> **File Format:**
>
> > **Save as csv:** If checked the output file will be in csv-format if not it will be a simple txt-file. Can not be used in combination with *Enable XML Report.*
> >
> > **Log entity name:** If checked the entity name will be additionally included in the report file.

7 Calibration

A well calibrated system is the base for a precisely working marking system. This chapter describes the 2D and 3D calibration of scanner marking system.

Depending on the scan head, there are several options:

	2D scanhead	3D scanhead	
2 D	F-Theta lense + 2D UCF*	F-Theta lense + 2D UCF*	FlatLense = z + 3D UCF**
3 D Optic3D		F-Theta lense + FlatLense = z + 2D UCF* + z-lookup table	FlatLense = z + 2D UCF* + z-lookup table

Figure 78: Calibration Options, Red = necessary licenses

- With a 2D scanhead, only 2D applications are possible. For calibration, an F-Theta lense and a 2D UFC* (USC Calibration File) are needed.
- With a 3D scanhead, the type of calibration depends on the required application and if there is an F-Theta lense integrated into the system:
  - For a 2D application with F-Theta lense, a 2D UCF* is needed.
  - For a 2D application without F-Theta lense, the FlatLense license (in order to access the z-axis for calibration) is required and a 3D UCF**.
- For 3D applications, the Optic3D license is required. Again, there are two types of calibration:
  - With an F-Theta lense, the FlatLense license, a 2D UFC* and additionally a z-lookup table is needed for calibration.
  - Without F-Theta lense, the FlatLense license, a 2D UFC* and a z-lookup table is needed for calibration.

* The 2D UFC can be acquired from the scan head manufacturer or created / edited with the correction file tool <SCAPS>\tools\sc_corr_table.

** The 3D UFC can be acquired from the scan head manufacturer. It is also possible to generate a 3D correction file with the correction file tool, but for this the 3D calibration is necessary

7.1 2D Calibration

Before starting with any marking application, the optic system needs to be well-calibrated. A precisely calibrated static system (correction file and optic settings) is absolutely required and really critical for a good marking results.

It is recommended to start with neutral optic settings:

SAMLight → Settings → System → Optic:

- Disable X invert, disable Y invert and disable XY flip.
- Set X gain to 1.0, Y gain to 1.0 and rotation to 0.0°.

SAMLight → Settings → System → Optic → Advanced → Correction, Settings:

- Set X gain to 1.0, Y gain to 1.0 and rotation to 0.0°.
- If necessary, inversion and flip of axes can be included in the correction file *.ucf using <SCAPS>\tools\sc_corr_table.

Start to check the calibration:

- Mark a job like ABC-circle.sjf: Is the orientation (X and Y axes) of the marking result correct - is the text "ABC" readable?
  - o If not, the correction file must be manipulated. This can be done using <SCAPS>\tools\sc_corr_table (correct the orientation).

- Mark a job like Grid.sjf: Are the X and Y aspect ratios correct? In other words: is 1 mm in the drawing very precisely 1 mm on the target in vertical and horizontal direction throughout the whole working area (not only in the middle of the field)? This is very important for Mark On The Fly and splitting applications.
  - o If not, adapt the field size until the dimension in the direction which you want to use as MOTF axis is correct. Mark again: are the X and Y aspect ratios correct?
    - If not, adapt the gain value of the axis, which is not used as MOTF axis. Mark again: are the X and Y aspect ratios correct?
      - If not, improve the correction file with sc_corr_table. Please find further information on how to work with sc_corr_table online.

7.2 3D Calibration

This chapter explains how to calibrate a 3D optic system. There are two different possibilities depending on your scan head:

- Z Lookup Table (2D scan head with a moveable z-axis):
Using lookup control points in SAMLight. The process consists of an iteration of measurements, of the read out of the marking result and of the calculation of new software settings using the measured values. The most important and time consuming part is the calibration of the Z lookup table (mapping between 'Focus distance [mm]' and 'Z - value [bit]' of the Z shifting device).

- F-Theta Factor (3D scan head with F-Theta lens)

7.2.1 Z Lookup Table Calibration

WARNING

Before starting with the calibration of the 3D system, the optic system needs to be well-calibrated in 2D. Then, the calibration of the 3D optic system (lookup table in SAMLight) is done with the following steps (see figure 79 for an overview):

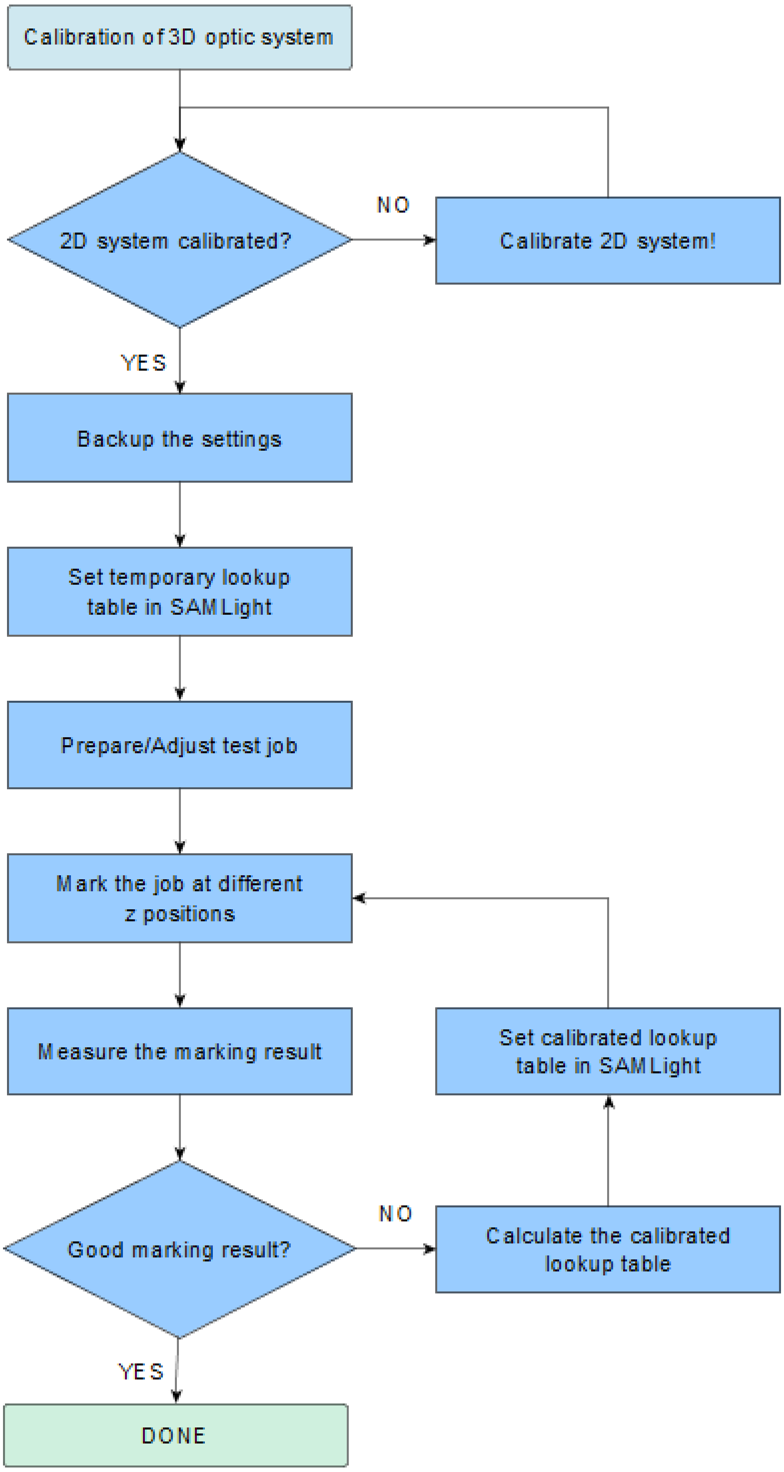


Figure 79: Overview of calibration process.

1. Set up of the temporary lookup table

The lookup table is situated in Settings → System → Optic → Advanced → Correction, Settings → Z-Correction, Settings (see figure 80).

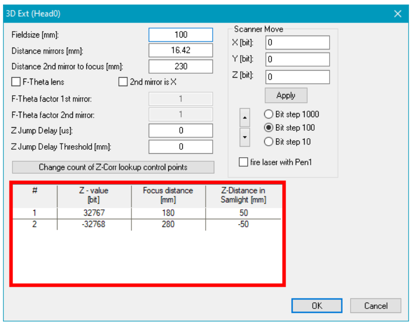


Figure 80: Z-Correction Settings

- Set *field size [mm]*: Use the same value as in Settings → System → Optic → Lens → Size. (Set X gain to 1 and Y gain to 1).
- Set *Distance mirrors [mm]*: This value of the distance between the x and y mirror is given by the scan head manufacturer or can be estimated.
- Set *Distance 2nd mirror to focus [mm]*: This value defines the focus distance for SAMLight Z=0mm. This value has to be inside the *Focus distance [mm]* range of the Z lookup table (see figure 81). Please note that this distance is NOT the distance between the focus and the case of the scan head but between the focus and the position of the 2nd mirror inside the box of the scan head.
- Enable *F-Theta lens* if an F-Theta lens is used.
- Set up a temporary linear Z lookup table. Make sure, the value of *Distance 2nd mirror to focus* is within the lookup table. In the example given in figure 80 and below, the lookup table is set symmetrically around the *Distance 2nd mirror to focus*.
  i. Example for typical Z shifting devices:
     - o Distance 2nd mirror to focus = 230mm
     - o -32767bit at 190mm
     - o 0 bit at 230mm
     - o 32767bit at 270mm
  ii. Example for Optotune lens:
     - o Distance 2nd mirror to focus = 230mm
     - o 20000bit at 190mm
     - o 0 bit at 230mm
     - o -20000bit at 270mm

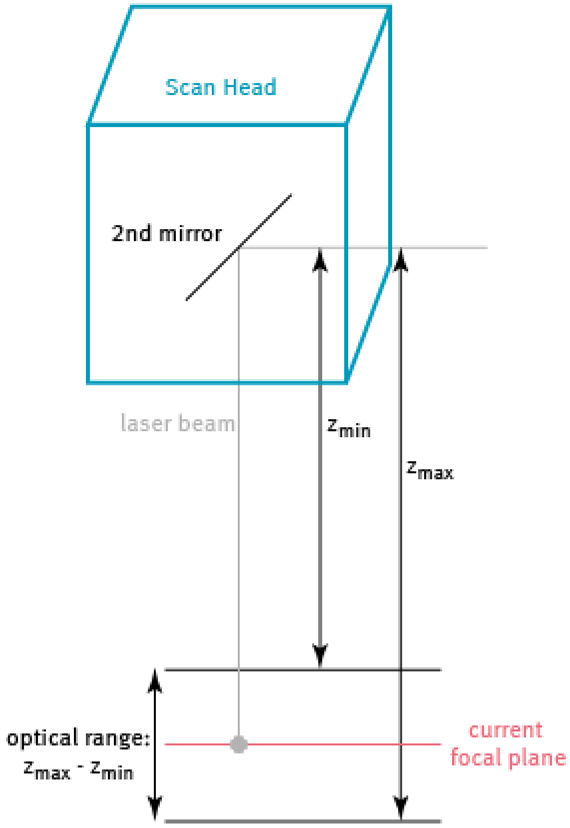


Figure 81: Illustration of optical range and distance 2nd mirror to focus.

2. Preparation and marking of the test job:

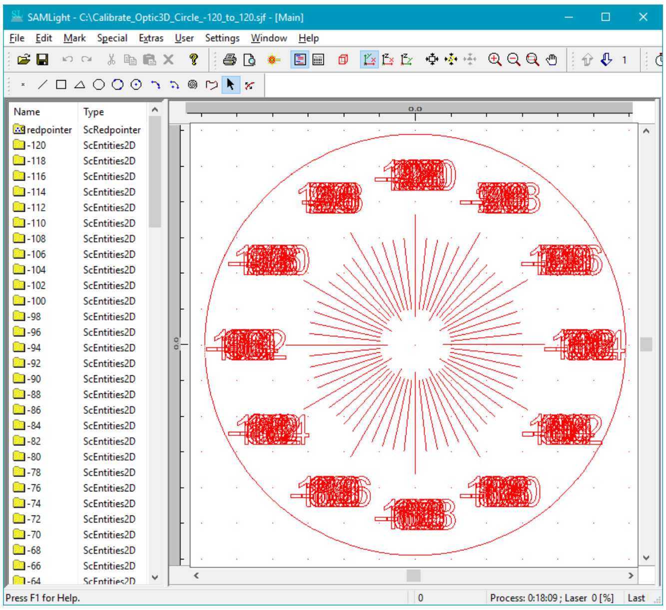


Figure 82: job file Calibrate_Optic3D_Circle_-

120_to_120.sjf

a. Load the job file *Calibrate_Optic3D_Circle_-120_to_120.sjf* (see figure 83).
b. Remove unneeded Z values of the job:
   i. Delete all Z values below *Distance 2nd mirror to focus [mm]* minus *minimum of Z lookup table Focus distance [mm]*.
      Example: 230mm - 270mm = -40mm.
      Delete entities from *-120* to *-42*.
   ii. Delete all Z values above 'Distance 2nd mirror to focus [mm]' minus 'maximum of Z lookup table Focus distance [mm]'.
      Example: 230mm - 190mm = +40mm.
      Delete entities from *+42* to *+120*.
   iii. Save the reduced job with a new proper name.
      Example: Calibrate_Optic3D_Circle_-40_to_40.sjf
c. Start marking at *Distance 2nd mirror to focus [mm]* (for Z =0) and optimize the marking:
   i. Use a slow mark speed.
      Example: Mark speed = 500mm/s
   ii. Use a slow mark speed and a high jump delay.
      Example: Jump delay = 3000µs
   iii. Adjust the pen power (eventually use SAMLight Parameter Finder) that you can easily define the *Best marked line* in the marking result. Usually, you find the *Best marked line* at the middle of marked scale, see figure 84.
d. Mark the same job file at different *Target heights* which you reach for example with the mechanical z-axis (see figure 83 for illustration):
   i. Decrease the distance between scan head and marking target step by step.
      1. Mark the job on each *Target height* until you are completely out of focus (no more marking result).
      2. Write next to each marking result the *Target height*.
         Target height = Distance 2nd mirror to focus [mm] (for Z =0) minus *distance between scan head and marking target*
         Note: Mind the sign! A distance between scan head and marking target smaller than *Distance 2nd mirror to focus [mm]* is related to a positive Target height (SAMLight Z value).
   ii. Now increase the distance between scan head and marking target step by step (start marking at *Distance 2nd mirror to focus [mm]* (for Z =0)).
      1. Mark the job on each *Target height* until you are completely out of focus (no more marking result).
      2. Write next to each marking result the 'Target height'.
         Target height = Distance 2nd mirror to focus [mm] (for Z =0) minus *distance between scan head and marking target*

         Note: Mind the sign! A distance between scan head and marking target greater than *Distance 2nd mirror to focus [mm] can be related to*

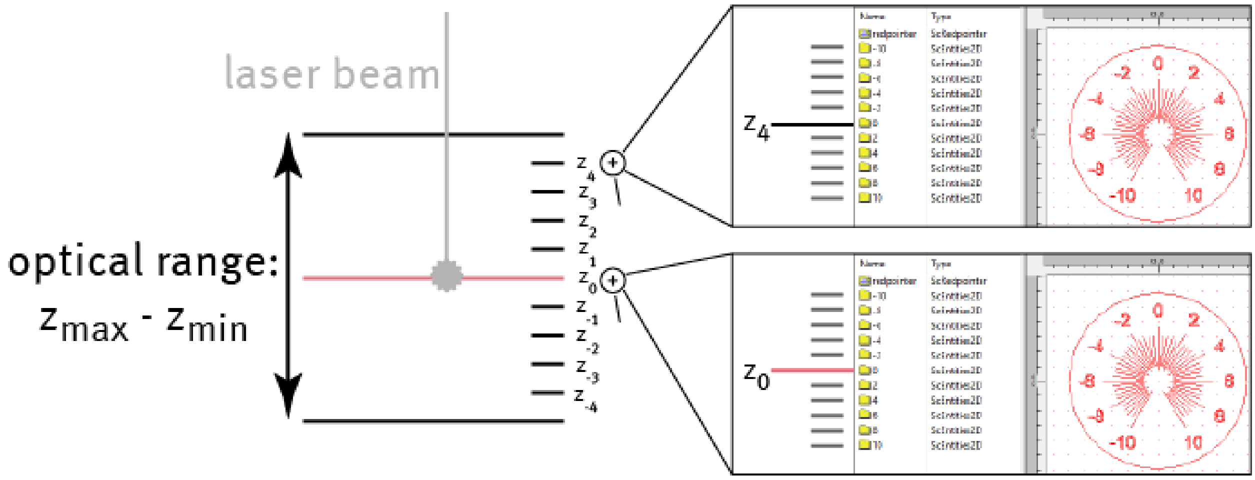


Figure 83: Illustration of the process of marking of the job at several different target heights: in this example, the z-axis is driven to the z-positions between z_4 and z_-4. At each position, the job is marked which consists of several entities at different z-positions. Examples are given for z_4 and z_0.

After step 2d) you should have marking results together with target heights similar to figure 84:

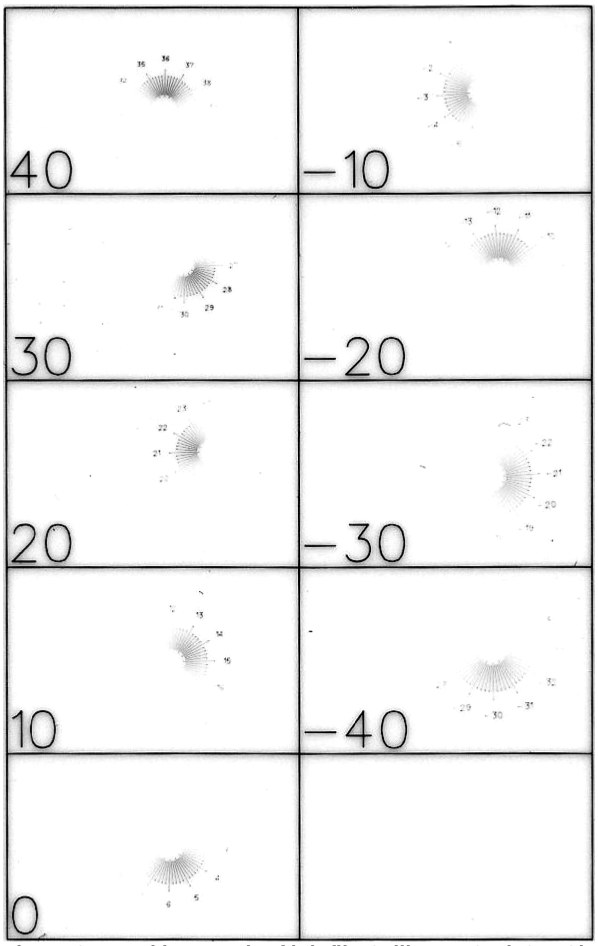


Figure 84: Marking result of job file Calibrate_Optic3D_Circle_-50_to_50.sjf

3. Analysis of the marking result with calculation of the new (calibrated) lookup table

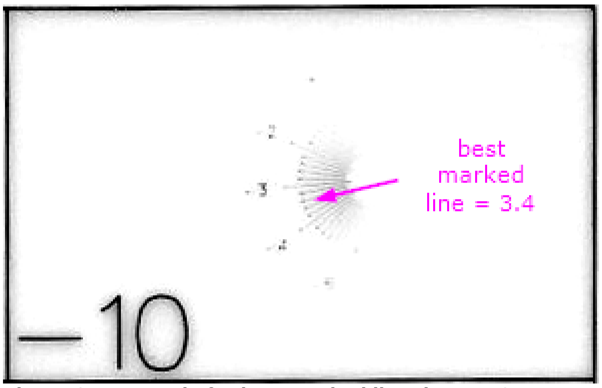
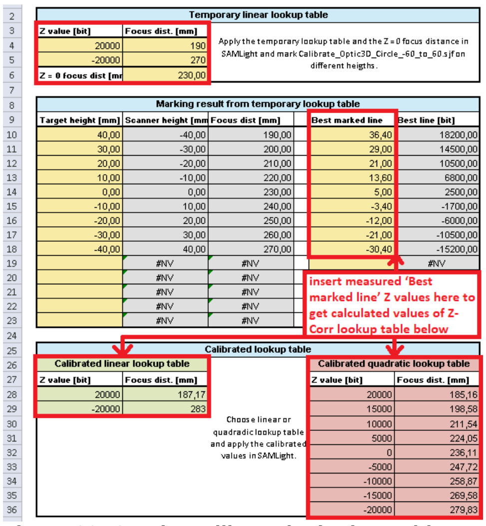


Figure 85: Example for best marked line: here at 3.4.



Temporary linear lookup table		
Z value [bit]	Focus dist. [mm]	Apply the temporary lookup table and the Z = 0 focus distance in SAMLight and mark Calibrate_Optic3D_Circle_-60_to_60.sjf on different heigths.
20000	190	
-20000	270	
Z = 0 focus dist [mm	230,00	

Marking result from temporary lookup table				
Target height [mm]	Scanner height [mm	Focus dist [mm]	Best marked line	Best line [bit]
40,00	-40,00	190,00	36,40	18200,00
30,00	-30,00	200,00	29,00	14500,00
20,00	-20,00	210,00	21,00	10500,00
10,00	-10,00	220,00	13,60	6800,00
0,00	0,00	230,00	5,00	2500,00
-10,00	10,00	240,00	-3,40	-1700,00
-20,00	20,00	250,00	-12,00	-6000,00
-30,00	30,00	260,00	-21,00	-10500,00
-40,00	40,00	270,00	-30,40	-15200,00
	#NV	#NV		#NV
	#NV	#NV		
	#NV	#NV		
	#NV	#NV		
	#NV	#NV		

insert measured 'Best marked line' Z values here to get calculated values of Z-Corr lookup table below

Calibrated lookup table			
Calibrated linear lookup table		Calibrated quadratic lookup table	
Z value [bit]	Focus dist. [mm]	Z value [bit]	Focus dist. [mm]
20000	187,17	20000	185,16
-20000	283	15000	198,58
		10000	211,54
		5000	224,05
		0	236,11
		-5000	247,72
		-10000	258,87
		-15000	269,58
		-20000	279,83

Choose linear or quadradic lookup table and apply the calibrated values in SAMLight.

Figure 86: Get the calibrated Z lookup table with the marking results from 2. and the excel file Calibrate_Optic3D_v0.9.3.xlsx

You can use the excel file Calibrate_Optic3D_v0.9.3.xlsx which you need to adjust according to your temporary lookup table and your measurement. Delete all lines that you did not use in your measurement.

a. Temporary linear lookup table: Insert the values of your temporary linear Z lookup table and *Distance 2nd mirror to focus* (Z = 0 focus dist) to the excel file (see figure 88)

b. Marking result from temporary lookup table: Insert the values of *Scanner height* and of the *Best marked line* for all valid marked target heights. The *Best marked line* must be read out from the marking result. The marking result consists of several lines corresponding to different z-values. Look for the line which is best in focus, meaning sharply marked and in the middle of the marked lines. Towards the edges, the lines should become less sharp and lighter. See figure 87 for illustration. Skip *Best marked line* values, which are very close to the limits of the Z range of the temporary linear lookup table.

c. Calibrated lookup table with linear and quadratic fit details. In the excel file, there is a graphic representation of the data. In this plot, you can check your data for inconsistencies like measurement errors. Decide if you want to use the linear fit or the quadratic fit.

4. Adaption of the lookup table in SAMLight

Insert the values of the calibrated lookup table (linear or quadratic, from figure 86 Calibrated lookup table) to SAMLight. To do so, click on the button *Change count of Z-Corr lookup control points* and define the total number of control points. Then, fill the lookup table with the values from your excel file. Leave the dialog with OK.

WARNING

It could be that your system does not need the full defined bit range from the temporary linear Z lookup table to reach the limit in z-range. Remember that you can only set points in the lookup table that are shifted maximum +/- FS/2 relative to the defined zero point in SAMLight. If you try to set the full calibrated linear/quadratic lookup table in SAMLight you will get the error message telling that you are out of z limit.

Use the linear/quadratic fit function and use the theoretical focus distance limits from the temporary linear Z lookup table to calculate the bit value for the calibrated lookup table. Now you have the limits. If you use the quadratic fit, then remove all calculated points that are located outside your calculated limits.

5. Marking of the test job using the calibrated lookup table

a. Use your job file (modification of *Calibrate_Optic3D_Circle_-120_to_120.sjf*) and mark this job again at different *Target heights* (in reality) just as you have been doing in 2.

b. The Z lookup table is optimal, when the *Target height* and the *Best marked line* are identical.
Target height = Distance 2nd mirror to focus [mm] (for Z =0) minus *distance between scan head and marking target*
Note: Mind the sign! A distance between scan head and marking target greater than *Distance 2nd mirror to focus [mm]* can be related to a negative Target height depending on the Z-axis orientation in SAMLight.

6. After the Z lookup table calibration process, it is possible to redefine the distance between scan head and marking target for SAMLight Z=0 by adjusting *Distance 2nd mirror to focus [mm]*. The Z lookup table remains calibrated.

NOTE *Please note that the z-range is limited by the value of the field size (FS): starting with Z=0 (position of 'Distance 2nd mirror to focus [mm]', the reachable range is +/- FS/2.*

7.2.2 F-Theta Factor Calibration

If you are using an F-Theta lens for marking on different z-heights, the x- and y-coordinates have to be corrected if you are not marking on z=0.
This is done by "F-Theta Factors". Please follow the following steps to calibrate the F-Theta correction.

Calibrate F-Theta Factors:

For the calculation of the F-Theta factors, the small program 'F-Theta Factor Calculator' is necessary.

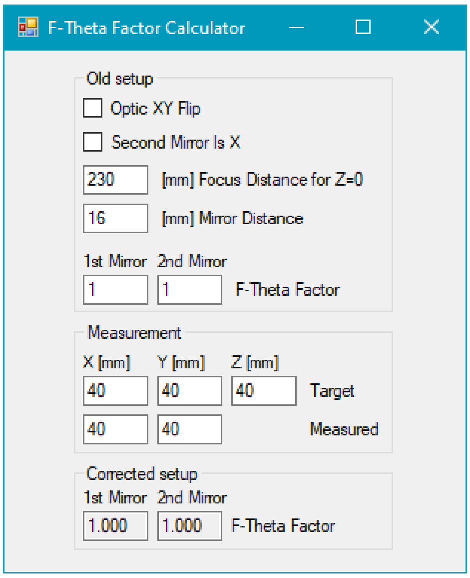


Figure 87: F-Theta Factor Calculator.

F-Theta Factor 1st mirror:

Create a job with two crosses.
A cross in the middle of the working area (X = center, Y = center and Z = 0) and another cross close to the X limit of the wo

The distance between the crosses must be 0.40*field size. If not, probably the field size is incorrect.
Mark the two crosses like above, but both at the same positive Z height, close to the Z limit. The positive Z limit is "Distanc
Measure the distance between the crosses.
Use the marking result Z=0 as Target value and the marking result with the positive Z-height as measured value for X [mm

F-Theta Factor 2nd mirror:

Repeat the marking with these two crosses along the Y-axis (X = center + 0.40 * field size, Y = center).
Measure the distance between the crosses.
Use the marking result Z=0 as Target value and the marking result with the positive Z-height as measured value for Y [mm

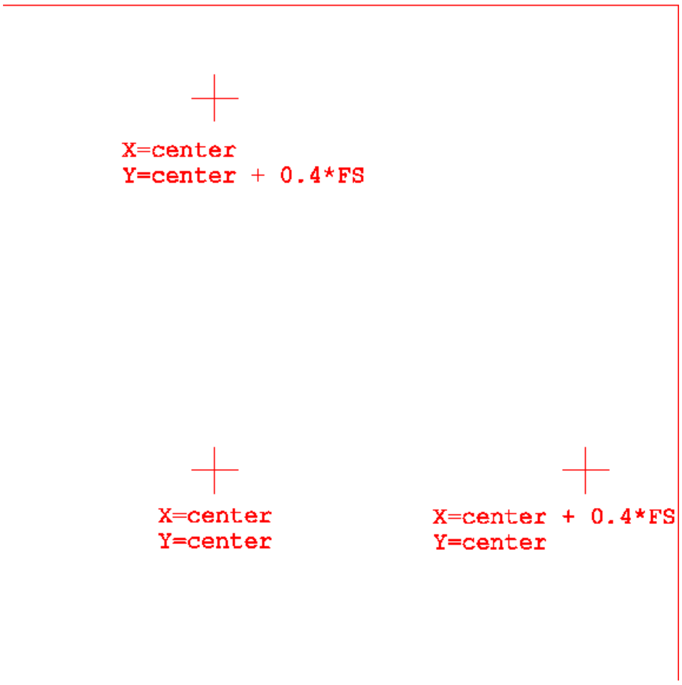


Figure 88: Job in SAMLight for Measurement to calculate F-Theta factor values.

Use the measured values to calculate the F-Theta factor 1st mirror and second mirror.
For that, the values from SAMLight are required.

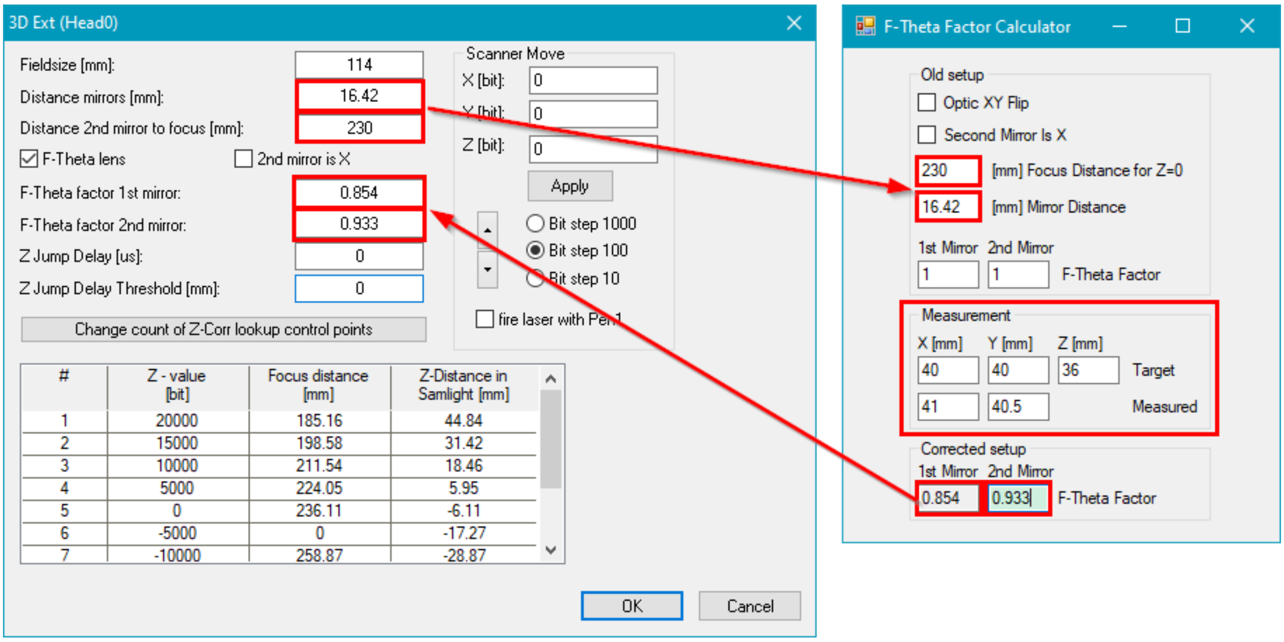


Figure 89: transfer calculated F-Theta factor values to SAMLight

Apply the corrected F-Theta Factor to SAMLight.
To test the new calibrated F-Theta Factor repeat the test. The F-Theta Factor is optimal, when the 'Target Position' and the 'Measured Position' are identical.

7.3 Marking On The Fly

The feature "Marking On The Fly" (MOTF) is for marking moving targets on a product line. General information on a (MOTF) setup is given in the SAMLight Manual in the Chapter Option MOTF.

7.4 Motion Axis

With the feature Motion Control its possible to control motion axes with SAMLight and USC cards. General information on a motion setup is given in the SAMLight Manual in the Chapter Motion Control Settings.

7.5 Background Camera

Background Camera is a feature for the integration of the picture of an off-axis webcam to allow a comfortable job handling.

The tool is able to compensate optical distortions/rotations.

Please find further information in the chapter Background Camera.

7.6 sc_corr_table

The program sc_corr_table is used to edit USC Correction Files (UCF). These files compensate for scan-head-specific optical distortions. Since build version 3_6_5_20170411_0052 of SAMLight, sc_corr_table is included in the installer (<SCAPS>\tools\). Please find further information on how to work with sc_corr_table online.

7.7 Parameter Finder

To reach a good marking result its necessary to find the best pen parameters. The Parameter Finder is a tool which helps you to find this parameters in an easy and fast way.

8 Pen Settings

Each entity is assigned to a pen. This is by default pen 1. To be visualized on the screen each pen can have a different color. Where to set the color, see in chapter View. The mark property page seen below can be used to assign a pen to the selected entity. The parameters are different for YAG and CO2 lasers.

NOTE

Pen 255 is used for the red pointer. In this pen, any defined power will be disabled with the use of the red pointer.

Pen 256 is used for the home jump. When the scanner moves to the home position (usually after executing) pen 256 is used. For example this feature can be used to switch off the lamp current after marking. See power control of pen. Pen 256 can be edited by clicking Advanced... → HomeJumpStyle (Pen #256).
Please note that Pen 256 is ALWAYS applied at the end of a job even if the home position itself is not activated. In this case, the current position is taken as "home position".

WARNING

In general, the Pen Settings are saved in <SCAPS>\system\sc_light_settings.sam if [Save Pens On Exit] is enabled or "Save Pens Now" is used.
But, the Pen Settings can also be [saved within a job] (.sjf). Be careful not to load any Pen Settings with a job and then accidentally overwrite your global Pen Settings in sc_light_settings.sam via "Save Pens On Exit".*

It is highly recommended to backup sc_light_settings.sam every time after having done any important or time consuming configuration of the Pen Settings.

There are several possibilities to save and load the pen settings:

Save pens into SAMLight settings file (*.sam)	**Load pens from SAMLight settings file (*.sam)**
• Save pens on SAMLight exit. Define in *Settings → System → General → Files →* [*Save Pens on exit*] if the pens should be saved when closing SAMLight. • Save pens directly via *Settings → System → General → Files →* [*Save Pens Now*].	• Load pens on SAMLight start. • Load pens on *File → New*. Define in *Settings → System → General → Files →* [*load pens on file new*] if the current pens should be reverted.
Save pens into SAMLight job file (*.sjf)	**Load pens from SAMLight job file (*.sjf)**
• Save a job directly via *File →* [*Save (as)*]. Define in *Settings → System → General →* [*Job Save/Load Dialog*] the default state of the *Pens* checkbox. • Save a job via *save job after each mark* if the current job already contains the pen settings. • Save a job via *auto save on exit* if the current job already contains the pen settings.	• Load a job directly via *File →* [*Load*]. Define in *Settings → System → General →* [*Job Save/Load Dialog*] the default state of the *Pens* checkbox. • Load a job via *File → RecentJobs*. Define in *Settings → System → General → Recent Jobs → Load pens* if the job pens should be loaded as well. • Load a job via *Job Auto Load*. Define in *Settings → System → General → Job Auto Load → Load pens* if the job pens should be loaded as well.

Table 13: Different possibilities to save and load pens

Apply pens to hatches, LineArrays or PolyLines:

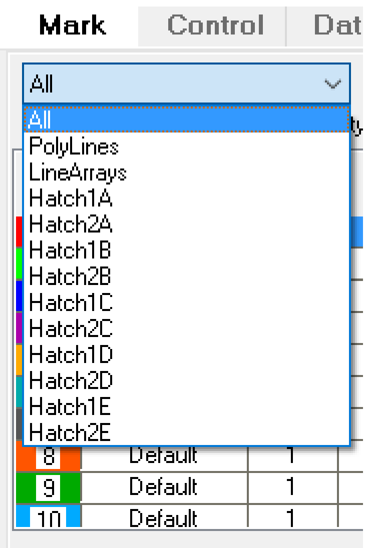


Figure 90: Drop-down menu to apply a pen for hatches, LineArrays and PolyLines

For selected entities a pen can be applied to each of 10 available [hatches], LineArrays and PolyLines. The currently applied pen for an entry of the drop-down menu will be highlighted when selected. To reset this settings select 'All' and apply a pen. For complex configurations we recommend the [styles] feature.

YAG laser:

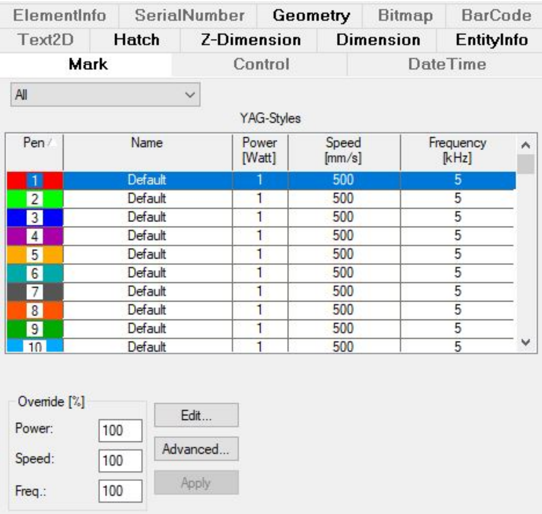


Figure 91: Mark property page with Pen Settings for YAG Laser

Pen: pens are listed by number

Name: Name of the pen

Speed: Galvo mark speed in mm/s.

Power: Laser power in Watt. For calibration see chapter Power Map.

Frequency: Frequency in kHz.

The grid is sortable through clicks on the respective column.

Override [%]: The override factors can be used to increase or decrease all values for all pens during mark process. The pen itself will not be changed.

Edit...: Press the edit button to define the settings of the currently selected pen. Alternatively, double-click on any field in the line of the selected pen (blue in Fig. **91**).

Advanced...: Within the advanced dialog settings for the Power Map, the HomeJumpStyle and the System PixelMap can be done.

Apply: Applies the selected pen to the current selected object.

CO2 laser:

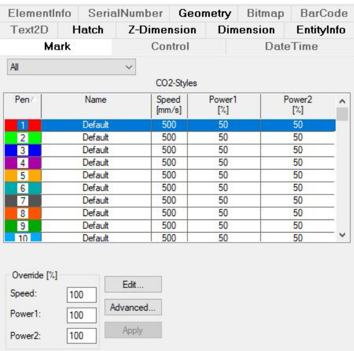


Figure 92: Mark property page with Pen Settings for CO2 Laser

Only the specific parameters are explained below, the others can be looked up above at YAG. Second power has to be enabled in *sc_setup.exe → Hardware Settings → Enable CO2 Power2.*

Power1: LaserPower1 signal in percentage of period.

Power2: LaserPower2 signal in percentage of period.

8.1 Edit Pens

Each single pen can be edited by pressing the *Edit...* button in the *mark* property page (Alternatively, double-click on any field in the line of the selected pen).
The pen that has to be set is selectable in the combo box. Also some variables provide *All* buttons to apply the setting to all pens.

Here the following tabs are available:

- Main
- Scanner
- Misc
- Drill
- Ramping
- Bitmap
- Paths

The following settings are global for each pen, so they can be seen for each pen independent of the chosen tab.

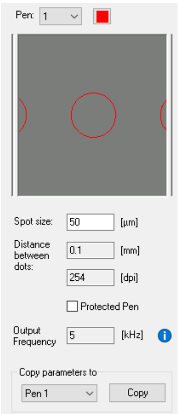


Figure 93: Pen Settings Global

Pen: The drop down menu can be used to switch between pens.

Color: This value can is used to change the color of the selected pen.

Spot size: This value is used for the display of the spots.

Distance between dots: Gives information on the distance between pixels in bitmap mode.

Protected Pen: When activated, the pen is protected. Thus, the parameters cannot be changed any more (also not by the application of an 'All' button in any other pen). It will be possible by user level to deny editing pens for specified user groups. This protected flag is only stored in the SAMLight settings and not in the pen settings. That means it won't be exported via pens included in a job file. If a job with included pens is loaded, a protected pen will be overwritten anyway. It is possible to overwrite a protected pen by CCI.

Output Frequency: It is only available for the USC-3. It shows the actual output frequency.

Copy parameter to: It is possible to copy all settings from one pen to another pen or to all pens.

8.1.1 Main Settings

The following page can be found under *Mark → Edit... → Main*.

The dialog looks different depending on the set laser mode:

- YAG
- CO2

For YAG lasers, the following dialog appears:

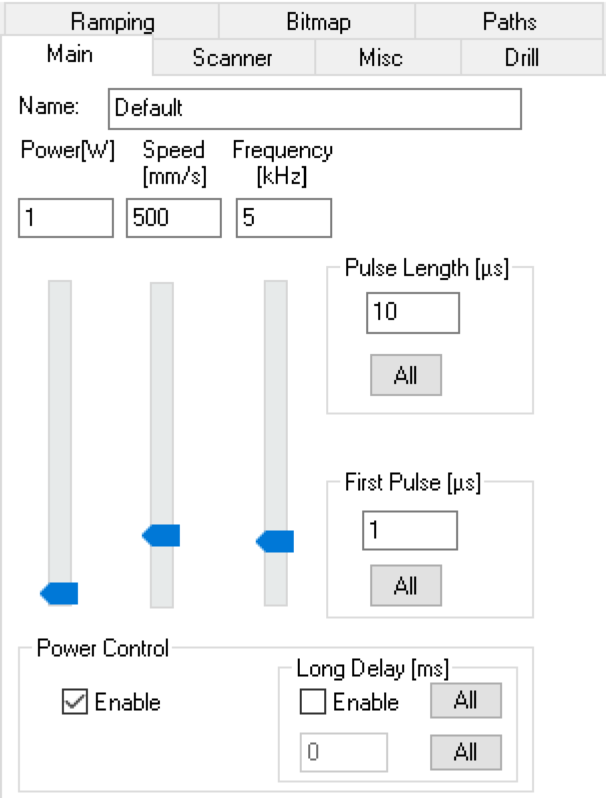


Figure 94: Main Pen Settings for YAG Laser

Power: Power of the laser lamp for selected pen. To redefine the power Power Control has to be enabled.

Speed: Marking speed of the selected pen. Min and max values are defined in the optic settings.

Frequency: Q-Switch frequency of the laser pulses. Min and max values are defined in the optic settings.

Pulse Length: Q-Switch length in µs.

If pulse length ≥ 1/frequency, a red notice 'cw' will appear to indicate continuous mode.

First Pulse: First pulse killer length in µs. Min and max values are defined in the optic settings.

Power Control: Enable or disable laser power and power control for this style.

> **Long Delay:** If enabled and the power is changing, the system will wait the indicated period before it will continue.

CO2 laser Pen → Main settings:

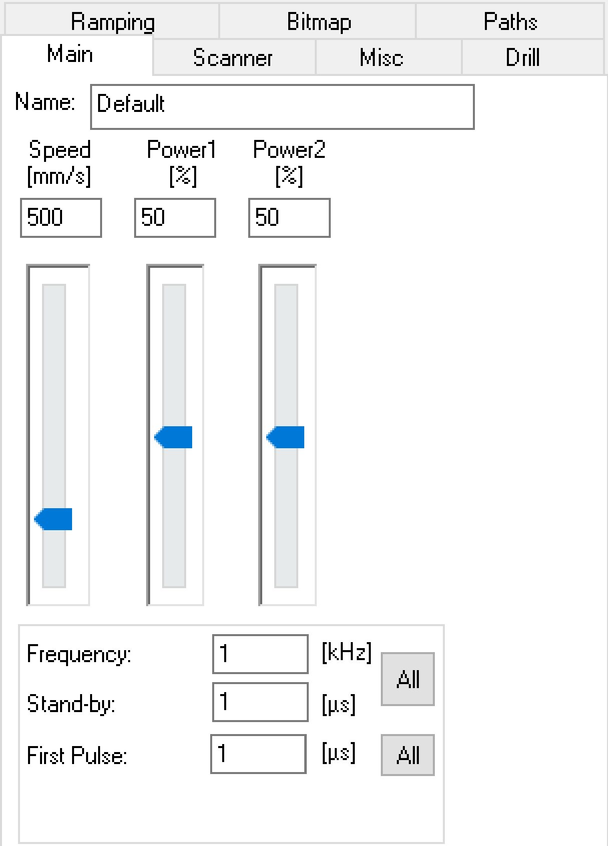


Figure 95: Main Pen Settings for CO2 Laser

Speed: Marking speed of selected pen.

Power1: Pulse length of laser signal 1 in %.

Power2: Pulse length of laser signal 2 in %. Second power is only shown if the checkbox *Show CO2 Power2* is enabled in *sc_setup → Hardware Settings → Settings.*

Frequency: Frequency of the laser pulses.

Stand-by: Stand-by pulse length in µs for stand-by mode (for both signals equal). The stand-by mode can globally be set in optic settings for scanner card.

All: Pressing this button applies Frequency and Stand-by to all pens.

Standby:

> **Half-period:** Half of the laser pulse period for standby mode.
>
> **Stand-by:** Q-Switch length in µs for standby mode. The stand-by mode can be globally set in the card settings.
>
> **First Pulse:** Set the length of the first pulse killer. *Laser_B as FPK* has to be enabled in *sc_setup.exe → Driver Settings* before and *Show CO2 Power2* has to be disabled in *sc_Setup.exe → Hardware Settings → Settings*

8.1.2 Scanner Settings

The following page can be found under *Mark → Edit... → Scanner*.

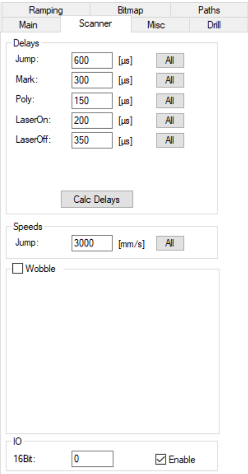


Figure 96: Scanner Delay Settings

Delays:

Jump: Delay after jump.

Mark: Delay at the end of the mark line.

Poly: Delay in polyline corners. Optimize PolyDelay reduces the duration depending on the angle.

LaserOn: Delay between the start of movement of the scan head and the switching on of the laser.

LaserOff: Delay between the end of movement of the scan head and the switching off the laser.

All: The *All* buttons apply the related value to all 255 pens.

Further details can be found in the *Delays* subchapter.

Speeds:

Jump: *Jump speed* of the selected pen. Min and max values are defined in the optic settings. Pen change is usually applied before jump. With change of the field size, the *jump speed* is changed accordingly (the speed in unit bit/s will remain the same).

Wobble: The wobble feature is described in the sub chapter Wobble Settings.

IO: Each pen can send an 8 bit signal entered to the 8 bit port.

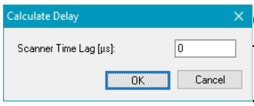


Figure 97: Calculate Delay Dialog

Calc Delays: If this button is pressed a dialog opens where the *time lag value* [µs] of the used scanner has to be entered. Based on this value the five delay values are recalculated. The resulting delays then can be used as a base for own optimizations. The time lag is a scanner-dependent parameter and should be found within the scanner specification.

NOTE *The jump speed is saved in the unit bit/s. This means that the displayed value in mm/s will change if the Field Size is changed.*

8.1.2.1 Delays

Delays: The scanner delays influence the time of the scanning process.

Example: Delays for a vector with jumps from and to the home position

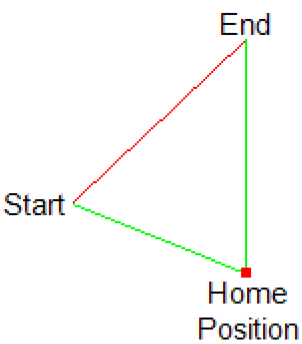


Figure 98 shows a simple job with:

- A jump vector (green) from Home Position to Start
- A mark vector (red) from Start to End
- A jump vector (green) form End to Home Position

Figure 99 shows the corresponding timing diagram:

- The job begins with switching on the Marking in Progress (MIP) signal, while the scanner is at the Home Position.
- After the Job Execution Delay (JED), the scanner jumps to Start, where it waits for Jump Delay.
- Then, the scanner begins with the mark move and the laser is switched on after LaserOn delay.

Delays: The scanner delays influence the time of the scanning process.

Example: Delays for a vector with jumps from and to the home position

Figure 98: Single vector (red) with jumps (green)

- Once, End is reached, the laser is switched off after LaserOff Delay, while the scanner waits for Mark Delay.
- The scanner jumps to the Home Position, where it waits for Jump Delay.
- Finally the MIP signal is switched off.

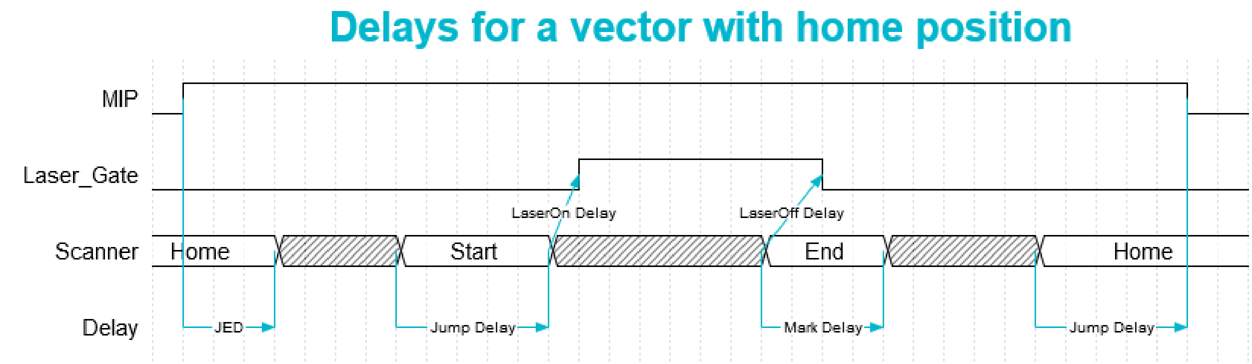


Figure 99: Timing diagram of delays for a single vector with jumps from and to the home position. MIP is Marking in progress.

Example: Delays for two subsequent vectors

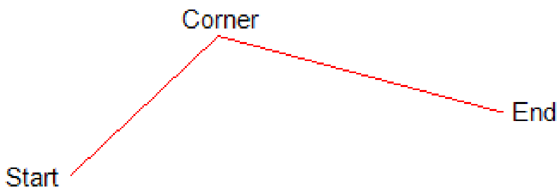


Figure 100: Two subsequent vectors meeting at the corner.

Figure 100 shows a simple job with:

- Two successive mark vectors (red) from Start to End, with a Corner in between.

Figure 101 shows the corresponding timing diagram:

- The job begins with switching on the Marking in Progress (MIP) signal, while the scanner begins with the mark move from Start to Corner and the laser is switched on after LaserOn Delay.
- When the scanner reaches the Corner, it waits for Poly Delay.
- After that, the scanner continues with the mark move from Corner to End.
- Once, End is reached, the laser is switched off after LaserOff Delay, while the scanner waits for Mark Delay.
- Finally the MIP signal is switched off.

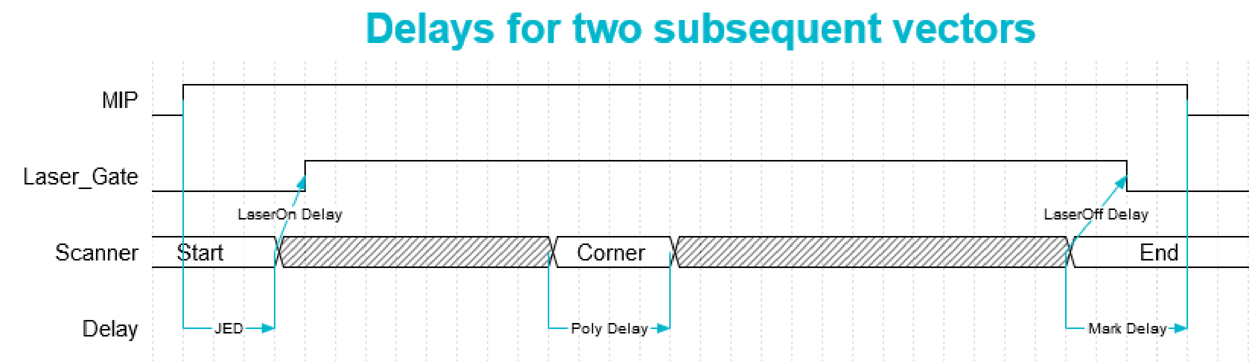


Figure 101: Timing diagram of delays for two subsequent vectors. MIP is Marking in progress.

Some delay rule conditions must be considered:

Delay rule	Description
LaserOff > LaserOn	This is the first delay rule. In case of very short mark commands the mark command may end before LaserOn command is issued and to guarantee that the LaserOff command is not issued before LaserOn command is issued, the LaserOff delay must be greater than the LaserOn delay
Mark + LaserOn > LaserOff	This is the second delay rule. In case there are two marking commands in succession, the LaserOff command after mark command 1 should be issued before the LaserOn command for mark command 2 gets issued. Therefore the mark delay plus the LaserOn delay must be longer than the LaserOff delay.

Table 14: First and second delay rule

To optimize the delays it is recommended to set the scanner delays on high values and define the laser delays first. Then the scanner delays can be optimized.

Delay	Effect if delay is too short	Effect if delay is too long
Jump	Oscillations could occur at the start of a vector.	No problem, just increasing mark time.
Mark	The last part of a vector is turned towards the direction of the jump vector.	No problem, just increasing mark time.
Poly	The corners of the polyline are rounded off.	Burn-in effects at the corners.
LaserOn	Burn-in effects at the beginning of a vector.	The first part of a vector is not marked.
LaserOff	The last part of a vector is not marked.	Burn-in effects at the end of a vector.

Table 15: Marking effects of scanner and laser delays

More detailed explanations can be found at Scanner and Laser delays.

8.1.2.2 Wobble Settings

Marking is possible with a wobble on top of the scanner path. This wobble can be activated in the pen settings on the Scanner tab of the pen property page.

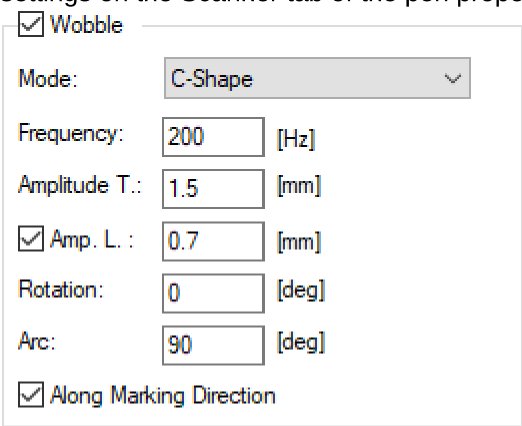


Figure 102: Wobble options

Mode: The desired shape of the wobble is selected with the mode. Depending on the scanner control card, there are different wobble modes available, see table 16. Depending on the wobble mode, there are different wobble parameters available, see table 17.

Frequency: The selected shape is repeated with this frequency. The frequency range is [1, 5000] Hz.

Amplitude T.(transversal): This amplitude modifies the elongation of the shape. For some modes, this is the only parameter specifying the size. Then, the parameter is called *Amplitude*. For some modes, two different amplitudes can be specified. In this case, this amplitude is *transversal*. The range for the amplitude is [1, 10000] bit.

Amp. L. (longitudinal): For some modes, two different amplitudes can be specified. In this case, this amplitude is the *longitudinal* one. It needs to be activated with the checkbox.

Rotation: Most modes can be rotated in respect to the default direction. The range for rotation is [-180, 180] deg.

Arc: This parameter defines the opening angle for C-Shape mode. The range for this arc is [1, 720] deg.

Along Marking Direction: Most wobble shapes can be aligned in respect to the marking direction. If unchecked, the alignment of the wobble shape is along the X-axis.

NOTE

The wobble amplitude is saved in the unit bit. This means that the displayed value in mm will change if the Field Size is changed.

The following modes are available for different scanner control cards:

	Circle	Sine	Ellipse	8-Shape	Double-8	Zig-Zag	C-Shape
USC-1/2, RTC3/4	✔	–	–	–	–	–	–
RTC5/6	✔	✔	✔	✔	–	–	–
USC-3	✔	✔	✔	✔	✔	✔	✔

Table 16: Available wobble modes for different cards. The RTC5/6 wobble modes have some restrictions.

The following wobble parameters are available for different wobble modes:

	Circle	Sine	Ellipse	8-Shape	Double-8	Zig-Zag	C-Shape
Frequency	✔	✔	✔	✔	✔	✔	✔
Amplitude T.	✔	✔	✔	✔	✔	✔	✔
Amp. L.	–	–	✔	✔	✔	–	✔
Rotation	–	✔	✔	✔	✔	✔	✔
Arc	–	–	–	–	–	–	✔
Along Mark. Dir.	–	✔	✔	✔	✔	✔	✔

Table 17: Available wobble parameters for different wobble modes.

For each mode of the wobble, an example of the scanner path is given in the following.

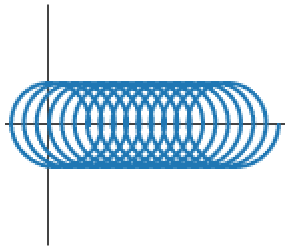


Figure 103: Circle wobble

Circle: The circle wobble can be used with all USC cards and all RTC cards. If the circle wobble is selected, all vectors and jumps are executed with this circular movement of the scanner. Frequency and Amplitude can be changed. The amplitude defines the radius of the circle.

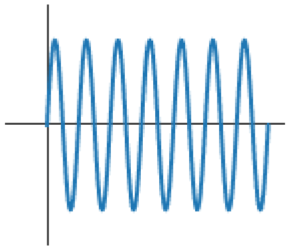


Figure 104: Sine wobble

USC-3 Sine: The sine wobble is available for the USC-3 card only.
Frequency, Amplitude and Rotation can be changed. The amplitude defines the height of the sinus. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).

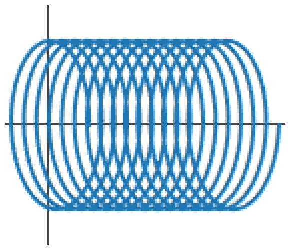
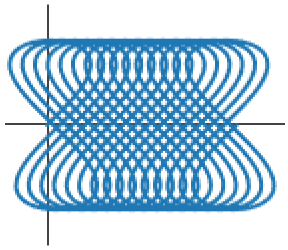


Figure 105: Ellipse wobble

USC-3 Ellipse: The ellipse wobble is available for the USC-3 card only.
Frequency, both amplitudes and Rotation can be changed. The amplitudes define the major and the minor axis of the ellipse. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).



USC-3 8-Shape: The 8-shape wobble is available for the USC-3 card only.
Frequency, both amplitudes and Rotation can be changed. The *transversal* amplitude defines the height of the shape, the *longitudinal* amplitude the width. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).

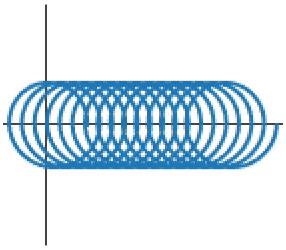


Figure 103: Circle wobble

Circle: The circle wobble can be used with all USC cards and all RTC cards. If the circle wobble is selected, all vectors and jumps are executed with this circular movement of the scanner. Frequency and Amplitude can be changed. The amplitude defines the radius of the circle.

Figure 106: 8-Shape wobble

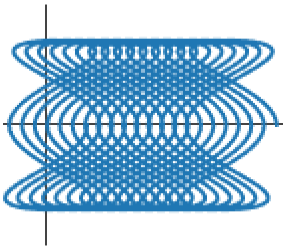


Figure 107: Double-8 wobble

USC-3 Double-8: The double-8 wobble is available for the USC-3 card only.
Frequency, both amplitudes and Rotation can be changed. The *transversal* amplitude defines the height of the shape, the *longitudinal* amplitude the width. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).

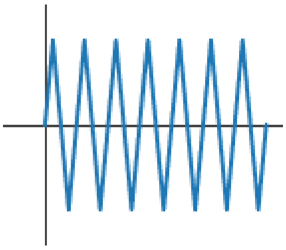


Figure 108: Zig-Zag wobble

USC-3 Zig-Zag: The zig-zag wobble is available for the USC-3 card only.
Frequency, Amplitude and Rotation can be changed. The Amplitude defines height of the shape. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).

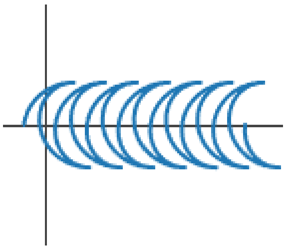


Figure 109: C-Shape wobble

USC-3 C-Shape: The C-shape wobble is available for the USC-3 card only.
Frequency, both amplitudes, Rotation and Arc can be changed. The *transversal* amplitude defines the height of the shape, the *longitudinal* amplitude the width. The shape can be rotated in respect to the default or marking direction (depending on the state of the flag *Along Marking Direction*).

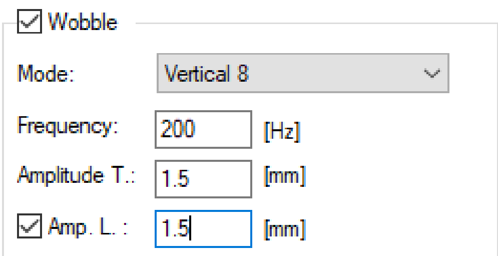


Figure 110: RTC5/6 wobble

RTC5/6 wobble modes: Ellipse, Horizontal 8 and Vertical 8.

With the *Amp. L.* checkbox (global, not pen specific) it is possible to set a different value for *transversal* and *longitudinal* amplitude.

- If the amplitudes are equal, the wobble shape is aligned in respect to the marking direction.
- If the amplitudes are unequal, the alignment of the wobble shape is along the X-axis.

Special Ellipse modes:

- Circle: Use Ellipse and disable *Amp. L.* checkbox.
- Sine: Use Ellipse and set *Amp. L.* to 0 mm.

Negative wobble frequencies are allowed and will result in a clockwise wobble motion. RTC5 wobble frequency range is [-6000 Hz, 6000 Hz].

NOTE *Make sure that the object size + wobble amplitude is smaller than the working area!*

8.1.3 Miscellaneous Settings

On this page SkyWriting, defocus and mark flags can be defined for a pen. The property page can be found under *Mark → Edit... → Misc*.

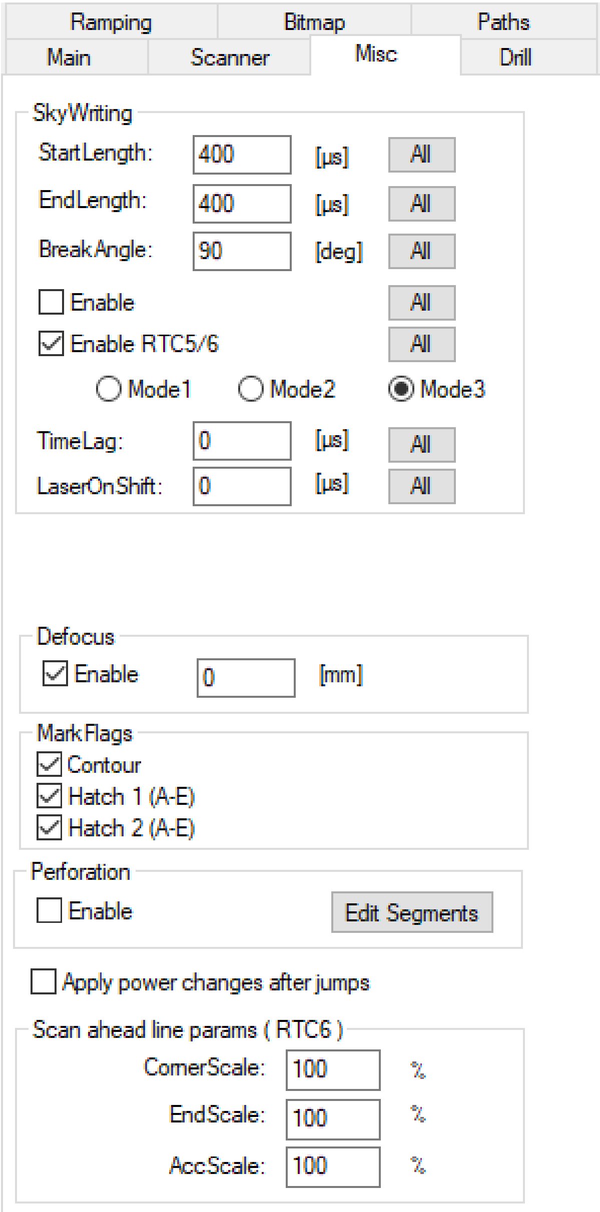


Figure 111: Misc Pen Settings

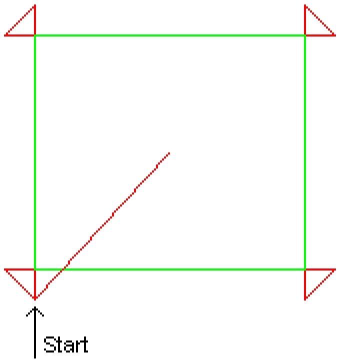


Figure 112: Example of SkyWriting

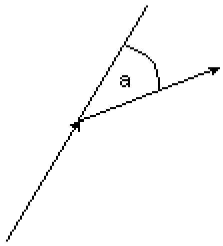


Figure 113: BreakAngle

SkyWriting: SkyWriting is used to avoid marking while acceleration and deceleration of the mirrors from and to marking speed to achieve very exact marked vertices. Therefore the feature adds a acceleration (StartLength) and a deceleration length (EndLength) to vectors. It is required to find new values for the laser and the scanner delays. The SkyWriting parameters can also be used for black and white bitmaps.

StartLength: Adds an acceleration length to the start of vectors.

EndLength: Adds a deceleration length to the end of vectors.

BreakAngle: Only relevant for PolyLines: SkyWriting is only used if angle *a* between two successive mark vectors is greater than or equal to the BreakAngle value, see figure 113.

Enable: Enables SkyWriting for marking.

The following settings are only available with a RTC card:

Enable RTC5/6: Enables the hardware-based SkyWriting. You can choose between the Modes. For more information, please consult your RTC manual.

Mode: Choose between the three different modi.

TimeLag: Set up a time lag value.

LaserOnShift: Set a laser on shift.

All: The All buttons apply the value to all 255 pens.

NOTE *The defocus is saved in the unit bit. This means that the displayed value in mm will change if the Field Size is changed.*

Defocus: This activates a focus shift on the Z-Axis. The unit of pen defocus is:
- o USC cards with FlatLense and Optic3D license: [mm]
- o USC-1 cards with FlatLense but without Optic3D license: *Defocus not available*
- o USC-2/3 cards with FlatLense but without Optic3D license: [2^{16} bit / field size]
- o RTC cards with SCANLAB 3D option: [2^{16} bit / field size]
- o RTC cards without SCANLAB 3D option: *Defocus not available*

MarkFlags: Contour or Hatch 1 & 2 can be activated for every pen.

Perforation: Open the perforation dialog.

Apply power changes after jumps: Delays power changes of the laser after the jump.

SCANahead line params (RTC6): If a RTC6 card is set in SAMLight settings following additional group box appears and gets activated if "Use excelliSCAN SCANahead" checkbox is enabled at RTC6 Settings dialog.

CornerScale: Corner sharpness in percent. 100% = sharp corners.

EndScale: Marking accuracy at mark/jump and jump/mark transitions. 100% = straight line ends.

AccScale: Determines the portion of the acceleration time (not: distance traversed) in which the laser is active, in percent. 100% = entire acceleration time. For more information about the 3 RTC6 SCANahead parameters above please consult chapter 3 and Ctrl Command set_scanahead_line_params of RTC6 and excelliSCAN manual.

8.1.3.1 Perforation Dialog

Perforation: if enabled vectors can be marked as dashed lines. At “Edit Segments” dialog four different On/Off segment lengths can be set, which defines the dashed line. The marking result can be displayed in advance by Preview Window in combination with Mark Preview. The Perforation feature does not work together with Mark Lines as Dots.

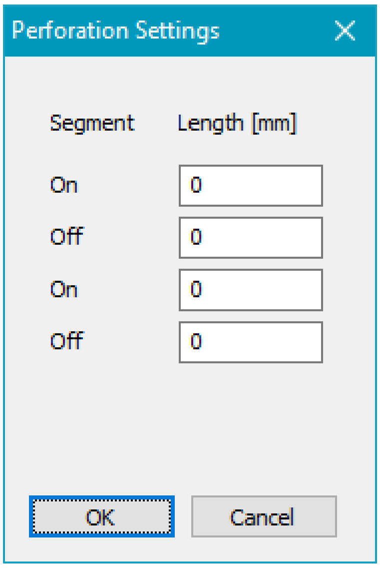


Figure 114: Perforation Dialog

8.1.3.2 SkyWriting Parameters

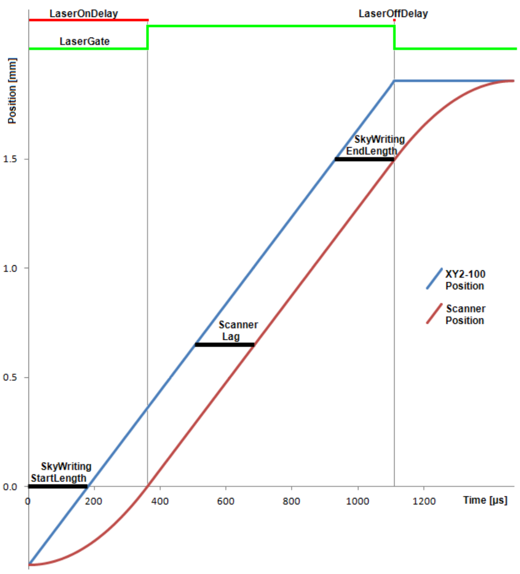


Figure 115: SkyWriting diagram

Delay rule for SkyWriting
LaserOff + StartLength + EndLength > LaserOn
Mark + LaserOn > LaserOff

Table 18: First and second delay rule for SkyWriting

Parameters	Suggested values	Effect if value is too low	Effect if value is too high
StartLength	= Scanner Lag	Laser shots are too narrow at the start of a vector.	No problem, just increasing mark time.
EndLength	= Scanner Lag	Laser shots are too narrow at the last part of a vector	The last part of a vector is marked too long.
Jump Delay	normal	Oscillations could occur at the start of a vector.	No problem, just increasing mark time.
Mark Delay	= 0µs	No problem.	No problem, just increasing mark time.

Parameters	Suggested values	Effect if value is too low	Effect if value is too high
Poly Delay	normal	The corners of the polyline are rounded off.	Burn-in effects at the corners.
LaserOn Delay	= StartLength + Scanner Lag	The first part of a vector is marked too soon.	The first part of a vector is not marked.
LaserOff Delay	= 1µs	First delay rule for SkyWriting must be true.	The last part of a vector is marked too long.

Table 19: Suggested starting values and marking effects of SkyWriting parameters

8.1.4 Drill Settings

The following page can be found at *Mark → Edit... → Drill*. If enabled, single points are scanned with drill mode.

YAG laser Pen → Drill settings:

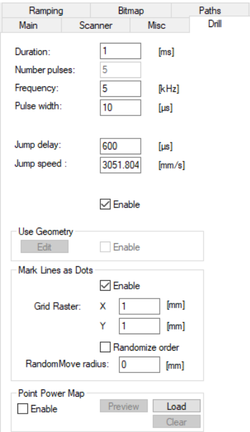


Figure 116: Drill Settings for YAG Laser

Duration: Time for scanning one point.

Number pulses: Displays the resulting number of pulses.

Frequency: Frequency of the laser pulses.

Pulse width: Q-Switch length in [µs].

Jump delay: Delay between the jump to the point and start marking this point.

Jump speed: Speed to jump to a point.

Use Geometry: If enabled, clicking on *Edit* opens a window where a simple geometry can be created. This geometry will be marked at every point instead of just marking a single point. This is an alternative to the drill function, which only generates dots.

Mark Lines as Dots: If the *enable* checkbox is set, lines will be marked as dots that lie on the defined *Grid Raster*. The amount of used points depend on the defined *Grid Raster* values. If those are chosen smaller, a higher amount of points is marked. *The Mark Lines as Dots* feature does not work together with *Perforation*.

Point Power Map: Load a bitmap in the background that defines the gray value for each pixel. Then if a dot is defined somewhere in the working area the power for the marking will be set by the Point Power bitmap. If *Preview* is grayed out, there is no bitmap assigned and the power will be taken from the pen power setting. If a bitmap was loaded with *Load* the bitmap is scaled to fit inside the field size. The bitmap power values are affected by the Power Map also.

CO2 laser Pen → Drill settings:

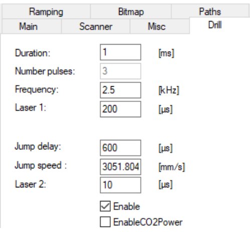


Figure 117: Drill Settings for CO2 Laser

Only the specific parameters are explained below, the others can be looked up above at YAG.

Laser 1: Pulse length of laser signal1 in [µs].

Laser 2: Pulse length of laser signal2 in [µs].

EnableCO2Power: If enabled Frequency, Laser1 and Laser2 are disabled, the values of these parameters are taken from the main page of pen settings instead.

The jump speed is saved in the unit bit/s. This means that the displayed value in mm/s will change if the Field Size is changed.

8.1.5 Ramping Settings

The following dialog can be reached at *Mark → Edit → Ramping*. Ramping allows to smoothly increase or decrease the speed or the power of the pen at the beginning or the end of the marking. It is also possible to add marking vectors at the beginning or the end of a marking in order to slowly increase the amount of energy delivered to the sample object.

If laser mode is IPG and the laser type 0 the power is changed by a manual latch sequence. This takes about 4 ms for each step. For power ramping the number of steps is therefore significantly reduced when using high marking speeds.

Power Ramping and Speed Ramping is not compatible with the MultiHead functionality.

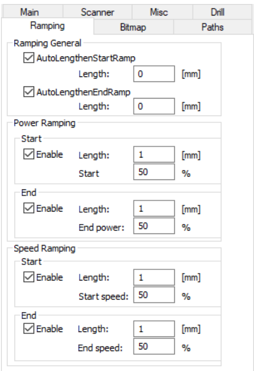


Figure 118: Ramping Settings Dialog

Ramping General:

AutoLengthenStartRamp: Add marking vectors before the beginning of the actual marking.

AutoLengthenEndRamp: Add marking vectors behind the end of the actual marking.

This function has to be used together with Power Ramping or Speed Ramping.

Power Ramping:

Start: Enable power ramping at the beginning of the marking. This smoothly increases the power of the laser from a given start value Start power to 100 % within the Length in mm. The checkbox Activate must be checked to enable this functionality.

End: Enable power ramping at the end of the marking. This smoothly decreases the power of the laser from 100 % to a given End power within the Length in mm. The checkbox Activate must be checked to enable this functionality.

This function can be used without any other Ramping or together with Ramping General/Speed Ramping.

Speed Ramping:

Start: Enable speed ramping at the beginning of the marking. This smoothly increases the speed of the scanner from a given start value Start speed to 100 % within the Length in mm. The checkbox Activate must be checked to enable this functionality.

End: Enable speed ramping at the end of the marking. This smoothly decreases the speed of the scanner from 100 % to a given End speed within the Length in mm. The checkbox Activate must be checked to enable this functionality.

This function can be used without any other Ramping or together with Ramping General/Power Ramping. Speed changes with increasing marking length.

8.1.6 Bitmap Settings

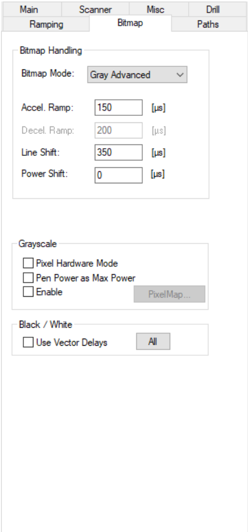


Figure 119: Bitmap Settings Dialog

Bitmap Handling:

This options are only available for the improved bitmap handling which is activated in *Settings → System → Extras → Improved Bitmap Handling*.

Bitmap Mode: This drop down menu allows to select between grayscale and b&w bitmap modes.

Acceleration Ramp [us]: This parameter adds a start vector for acceleration of the scanner at the beginning of each bitmap line. The length of the vector can be calculated with the marking speed.

Line Shift [us]: This parameter shifts each bitmap line in marking direction. The length of the shift can be calculated with the marking speed.

Power Shift [us]: This parameter shifts the setting of the power modulation (PixelAM mode) in respect to Laser_Gate. The power shift is rounded to integers of 1/frequency. This parameter can be used to compensate delays of the laser to apply a power shift (if the laser takes some time after getting the new power value to actually adapt to the new value).

Grayscale:

Pixel Hardware Mode: This checkbox should be enabled if using a gray scale bitmap.*[a]* If the improved bitmap handling is activated, the checkbox is set automatically for grayscale bitmaps and will not be visible any more.

Pen Power as Max Power: Uses the current pen power as maximum power for gray scale bitmap marking. As in all gray scale modes the power map is ignored. The factor is calculated by the current power value divided by the maximum power value.

PixelMap: This should be used to correct non-linearities of the material. For example for some materials there is a minimum laser power for which a marking result can be observed. For general laser dependent non-linearities you can also use the System PixelMap which will adjust the grayscale values after they have been mapped by the Pen PixelMap. For further information please refer to adjusting bitmap grayscale values.

Black / White: If the checkbox *Use Vector Delays* is activated, the vector delays set in Scanner will be applied. This option will not be visible if the improved bitmap handling is enabled.

[a] Please note that Pixel Hardware Mode must be disabled for Black&White bitmap marking!

8.1.6.1 Pen PixelMap

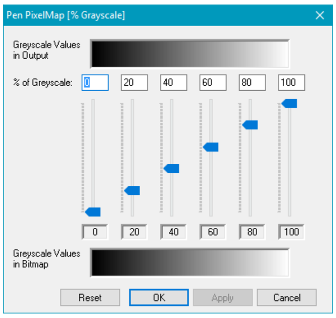


Figure 120: Pen PixelMap Dialog

The Pen PixelMap is a function which adjusts grayscale values. This means each grayscale value of each bitmap is mapped to an output value. The grayscale values of bitmaps are adjusted by the Pen PixelMap before they are mapped again by global System PixelMap.

- The Pen PixelMap can be used to optimize the bitmap output for different materials.
- The System PixelMap can be used to compensate general laser non-linearities. In addition, many laser sources should not be switched off completely for white pixels but needs a minimum laser power for fast switching between power levels.

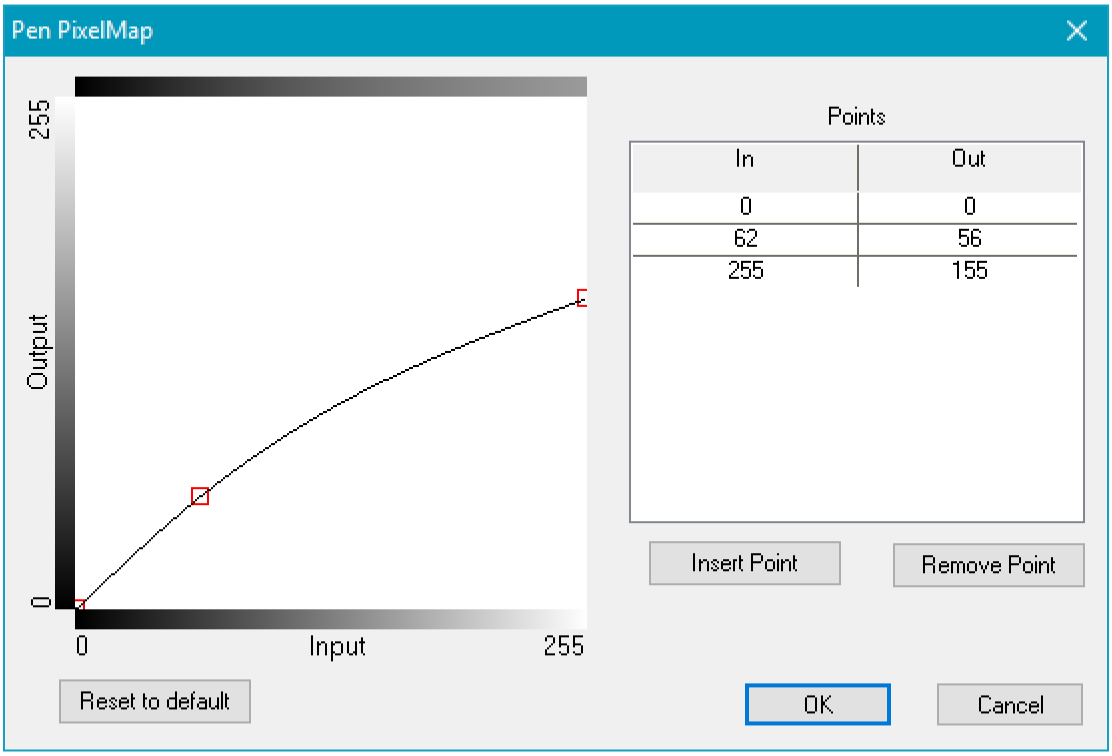


Figure 121: Dynamic Pen PixelMap Dialog

The dynamic Pen PixelMap and the mode of interpolation is activated via Settings → System → Extras → Enable Dynamic Grayscale Map. Up to 255 individual interpolation points can be set by clicking on the line in the input-output view or by 'Insert Point'. Points can be changed by editing the values in the table or by dragging the red boxes in the input-output view.

This Pen PixelMap applies a new grayscale value to grayscale bitmaps according to the linear, cubic or Hermite interpolated input-output values in the table of Points.

8.1.7 Paths Settings

The Pen Path property sheet can be opened via the entity property sheet *Mark* by double-clicking on a pen or by clicking the *Edit* button. The appearance is as follows:

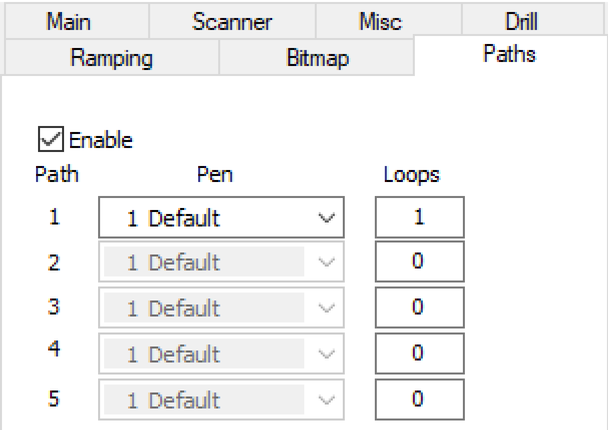


Figure 122: Pen Paths Dialog

Figure 123: Icon Pen Path active

Enable: If checked, the pen path function is active.

Path: A maximum of 5 different pens can be assigned to the path.

Pen: Define which of the 256 pens should be assigned to the position in the pen path.

Loops: Define how often a position in the pen path is repeated.

Activate: To activate a new pen path go to the edit field below *Loops* and enter a number greater than 0. Then click on *Apply*. Now the pen field of the defined path becomes active. Choose the desired pen by clicking on the drop down box in the pen field. It is not possible to create Pen Path self-recursions. This means, if a pen is included as a pen path of another pen, then the pen paths of the included pen will not be executed. So the included pen will be treated as a normal pen.

Execution of pen paths: Groups have an assigned pen as well. The pen path of the group is executed when the following two points are true:

- When the group-pen-path is enabled (Group > EntityInfo > PenPaths).
- And when the pen-pen-path is enabled (Group assigned pen number > Paths > Activate) as well.

If the pen path of the group is executed all pens within this group will be ignored! This includes pen paths of sublevel objects as well.

WARNING *All pen settings made in the master pen used for pen paths will be ignored. The configuration for the chosen pens will be used.*

NOTE *You can not use a pen, where pen paths are defined, for another pen path. So no pen paths of pen paths.*

8.2 Pen Advanced

The following dialog appears when pressing the *Advanced...* button on the M*ark* property page.

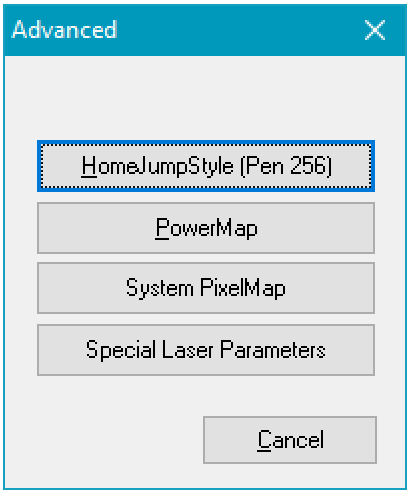


Figure 124: Pen Advanced Dialog

Head Selection: For Multiple Head applications HomeJumpStyle, Power Map and System PixelMap can be set up head independent.

HomeJumpStyle (Pen 256): Opens the Edit dialog for Pen #256 which is used for the home jump.

Power Map: The power map can be used to calibrate the laser in case the signals given to the laser and the resulting laser power do not behave linearly.

System PixelMap: The pixel map can be used to calibrate the laser in case the signals given to the laser and the resulting pixel occurrence do not behave linearly.

Special Laser Parameters: For some laser types (not visible if not available), pre-definition of values for special parameters like pulse length or APD index is possible.

8.2.1 Power Map

The Power Map can be edited by clicking Advanced... on the mark property page and then clicking on Power Map. The power of the laser is controlled by 8 bit values coming out of the controller board. The behaviour is not always linear. The power map can be used to calibrate the laser, means to find out the 8 bit values for the exact power value. This map is helpful to transfer jobs between systems using laser sources with different power.

Dialog for YAG laser:

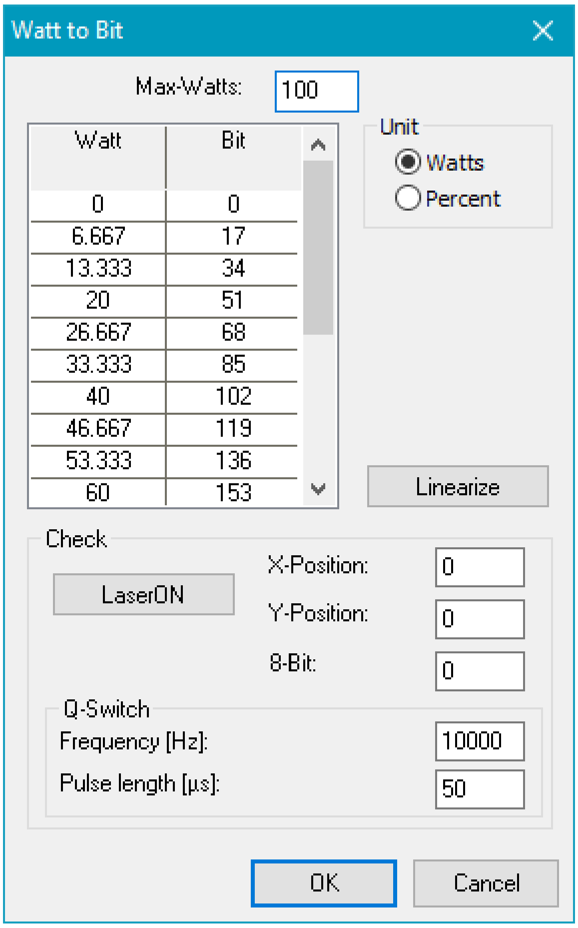


Figure 125: Power Map for YAG Laser

Max-Watts: Maximum power which can be emitted.

Unit: Watt or Percent can be used for the value

List: Edit this list to map a resulting power of laser pulse to a 8 bit value. The Watt column is divided into sixteen equidistant values from 0 to Max-Watts whereas the Bit column is editable.

Linearize: Calculating power values so that the Bits increase linearly from 0 Watt to Max Watts.

Check: The Check field is for measuring the laser power in watt on a specific X-/Y-Position working with the edited Q-switch settings under a 8-bit value signal.

> **LaserOn:** With this button the laser can be switched on and off again.

For calibration the following procedure is suggested:

1. Measure the maximum power by putting out a 8-bit value of 255.
2. Type in this max value in the Max-Watts field and press RETURN.
3. Now the Watt values in the list are updated.
4. Find out the corresponding 8-bit value for each of the given Watt values.

Dialog for CO2 laser:

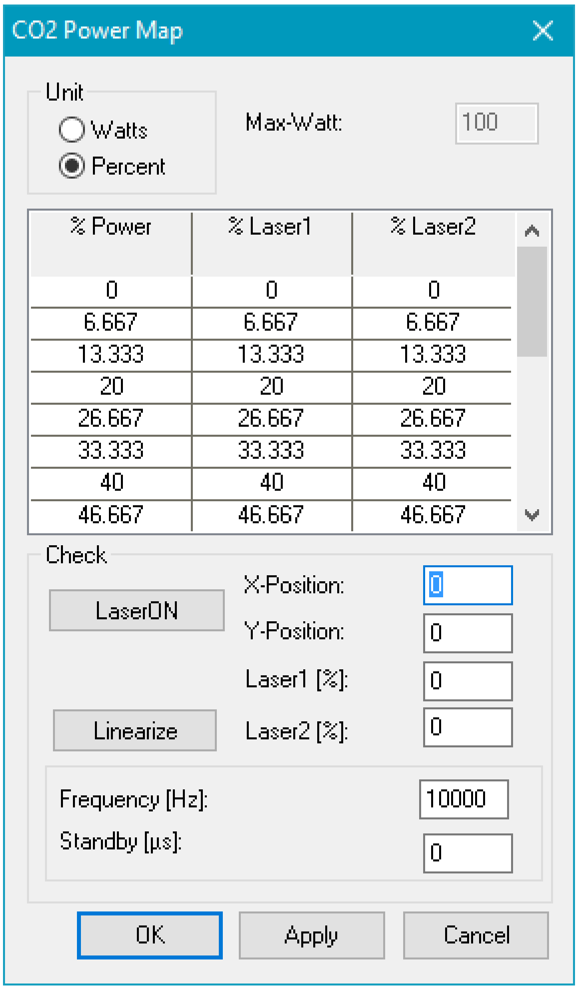


Figure 126: Power Map for CO2 Laser

Percent: The power is given in percentage.

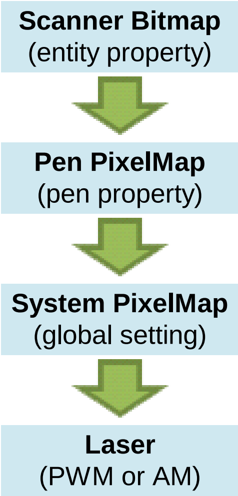
List: Edit this list to map a resulting power of laser pulse to the laser signal 1 and 2. The % Power column is divided into sixteen equidistant values from 0% to 100% whereas the %Laser1 and %Laser2 columns are editable.

Check: The Check field is for measuring the laser power in % on a specific X-/Y-Position working with the edited Frequency under defined Laser1 and Laser2 signals.

> **LaserOn:** With this button the laser can be switched on and off again.

8.2.2 System PixelMap

At the marking process, the grayscale value of the bitmap corresponds to the laser power. Due to non-linearities of the laser source and of material behaviour, the laser power needs to be adjusted for each grayscale value to get the desired marking result of the bitmap. The grayscale values of the scanner bitmap can be adjusted via two different pixel maps: the Pen PixelMap and the System PixelMap.



The bitmap grayscale values will first be adjusted by the Pen PixelMap and afterwards by the System PixelMap:

- Pen PixelMap (useful to consider material behaviour)
  Before marking, every grayscale bitmap which is assigned to this pen will be changed by this pixel map.
- System PixelMap (useful to consider laser properties like minimum power and non-linearities)
  Before marking, every grayscale bitmap will be changed by this pixel map.
- Power Map is not used for grayscale marking at all.

Table 20: Order of grayscale adjustment

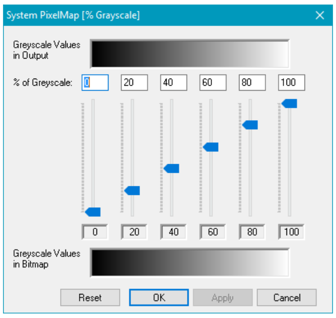


Figure 127: System PixelMap Dialog

The global System PixelMap is a function which adjusts grayscale values. This means each grayscale value of each bitmap is mapped to an output value. The grayscale values of bitmaps are already adjusted by the Pen PixelMap before they are mapped again by this System PixelMap.

- The Pen PixelMap can be used to optimize the bitmap output for different materials.
- The System PixelMap can be used to compensate general laser non-linearities. In addition, many laser sources should not be switched off completely for white pixels but needs a minimum laser power for fast switching between power levels. For further information, refer to Table 20.

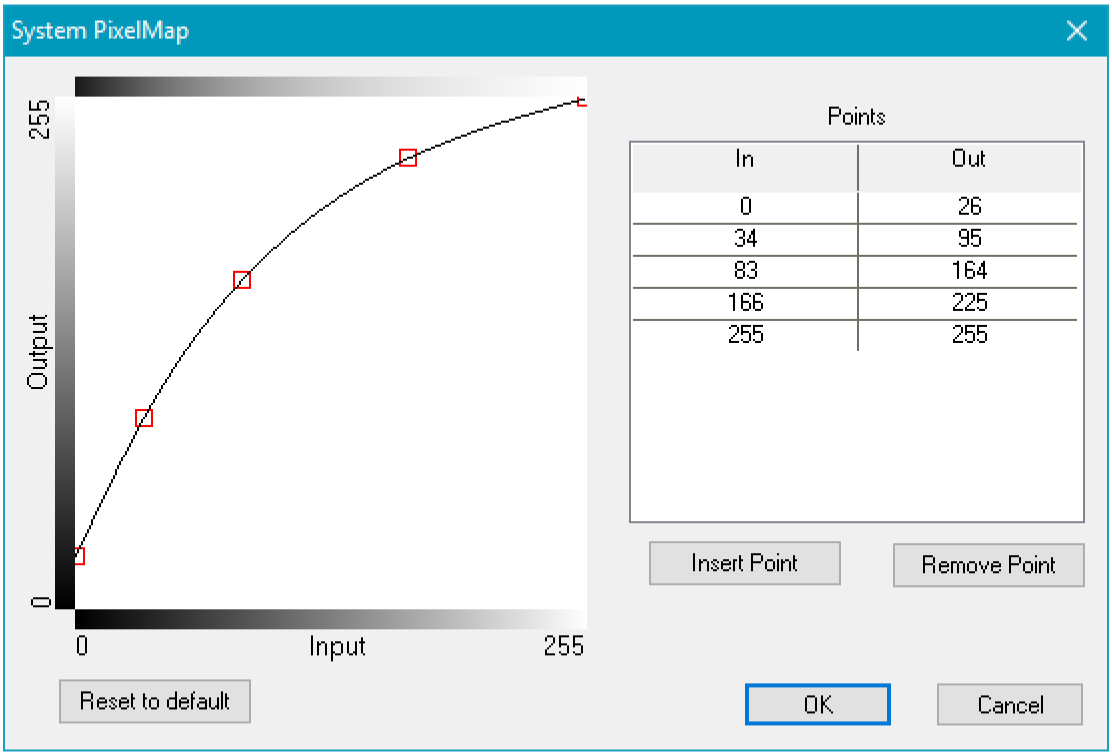


Figure 128: Dynamic System PixelMap Dialog

The new System Pixel Map and the mode of interpolation is activated via Settings → System → Extras → Enable Dynamic Grayscale Map. Up to 255 individual interpolation points can be set by clicking on the line in the input-output view or by *Insert Point*. Points can be changed by editing the values in the table or by dragging the red boxes in the input-output view.

This system pixel map applies a new gray value to gray scale bitmaps according to the linear, cubic or Hermite interpolated input-output values in the table of Points.

8.2.3 Special Laser Parameters

Special Laser Parameters: For some laser types (not visible if not available), pre-definition of values for special parameters like pulse length or APD index is possible. Use this dialog to pre-define the values for the drop down list.

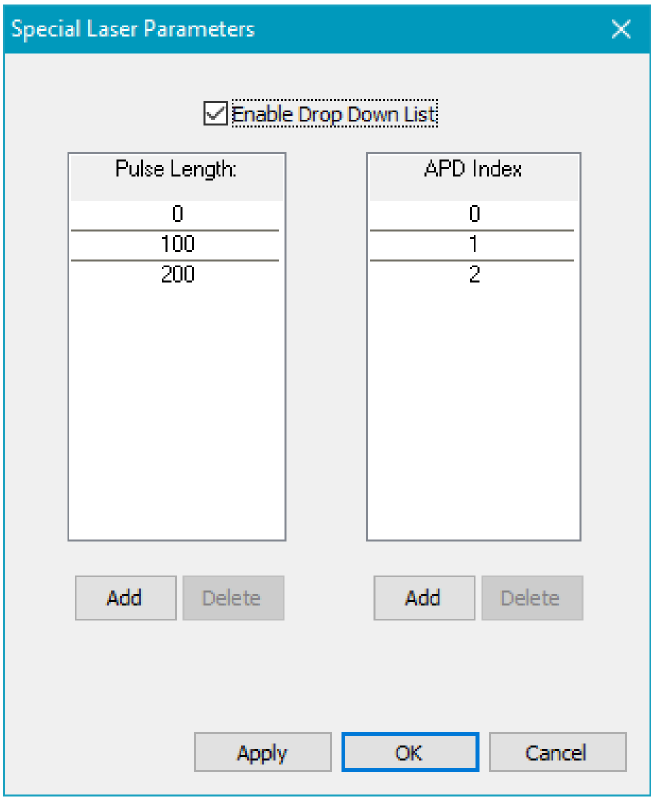


Figure 129: Special Laser Parameters

Enable Drop Down List: Activate the drop down a drop down list on the pen property page.

Add: Click on this button to add a value to the list. These values will then be available in the drop down list on the pen property page.

Delete: Select a value in the list and click on this button to delete it.

9 User Interface

At start up the application displays the user interface like it is shown below.

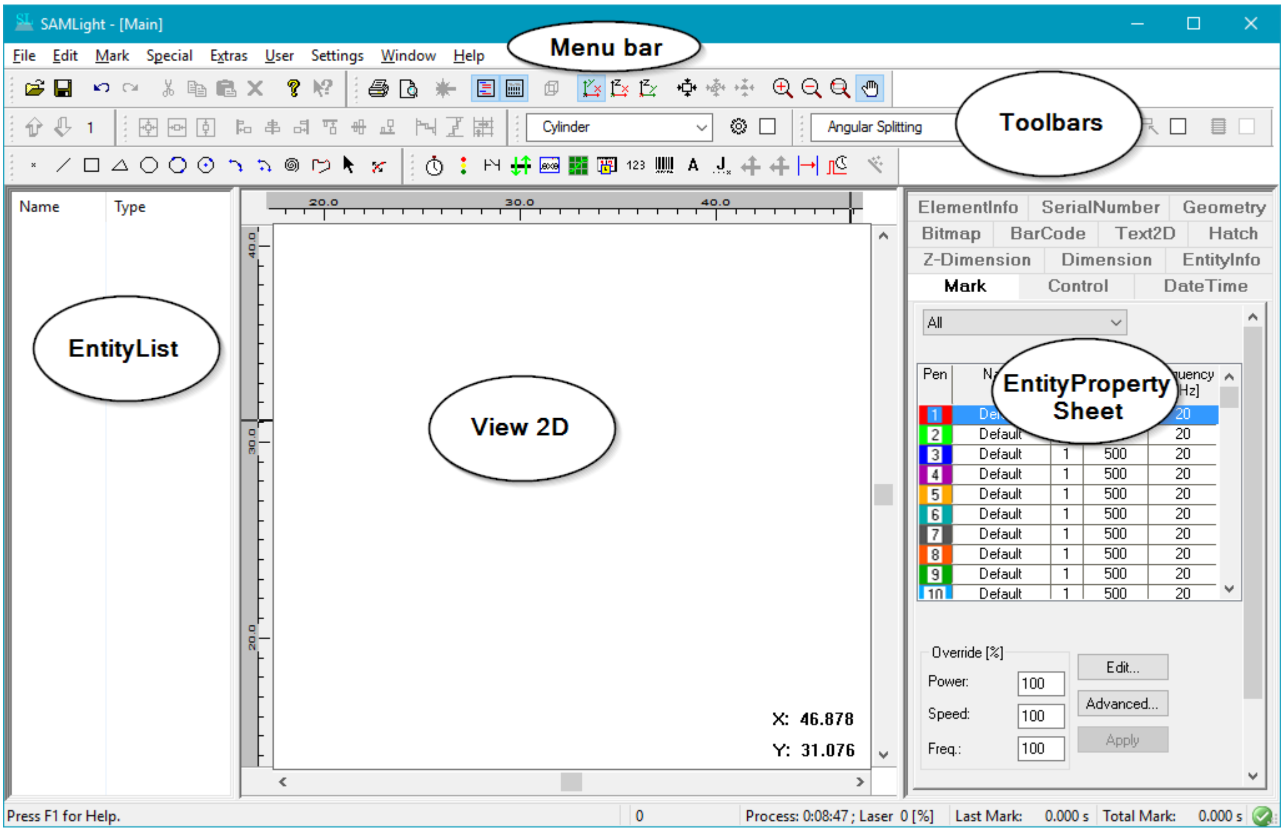


Figure 130: Main Window

Topics of User Interface:

- Menu bar
- Toolbars
- Entity List
- View 2D
- Entity Property Sheet
- Status Bar

9.1 Menu Bar

File Edit Mark Special Extras User Settings Window Help

Figure 131: Menu bar

Topics of the Menu bar:

- File
- Edit
- Mark
- Extras
- User
- Settings
- Window
- Help
- Special: The menu item Special is shown, if there are user defined Script settings.

9.1.1 File

New: Prepares for a new job and it deletes all current entities.

Load...: Opens a dialog to read jobfiles in SCAPSJobFile-Format (*.sjf) format. See chapter Job Format.

Save...: Saves the current job in sjf format.

Save as...: Opens a dialog to save the current job under a new file name (*.sjf). See chapter Job Format.

Job Properties...: See chapter Job Properties.

Import...: Import of data in format, for more details see chapter Import.

Extension	Description
*.ai	Adobe Illustrator (AI) is a vector graphics file format.
*.gif	Graphics Interchange Format (GIF) is a raster graphics file format.
*.job	GSI PC-Mark job file format containing vector graphics.
*.tif	Tagged Image File (TIF) is a raster graphics file format.
*.txt	Point Cloud Data is a ASCII format containing 3D vertices.
*.bmp	Bitmap (BMP) is a raster graphics file format.
*.cmx	Corel Metafile Exchange (CMX) is a vector graphics file format.
*.cnc	CNC G-Code is a language to control CNC (Computer Numerical Control) machines.
*.dst	Tajima DST is an embroidery vector graphics file format.
*.dxf	Drawing Exchange Format (DXF) is a CAD file format.
*.emf	Enhanced Metafile (EMF) is a raster graphics file format.
*.jpg	Joint Photographic Experts Group (JPEG) is a compressed image format.
*.mcl	Marker Control Language (MCL) is a GSI PC-Mark vector graphics file format.
*.pcx	Personal Computer Exchange (PCX) is a raster graphics file format.
*.plt	Hewlett-Packard Graphics Language (HPGL) Plotter File (PLT) is a language format for printing line drawings, specifically designed for 2D plotters.
*.png	Portable Network Graphics (PNG) is a raster graphics file format.
*.saf	SAF is a SCAPS archive.
*.svg	Scalable Vector Graphics (SVG) is a 2D vector graphics file format.
*.tga	TGA or TARGA is a raster graphics file format.
*.twain	TWAIN is a software protocol and applications programming interface between software and scanner.

Table 21: Available import formats

Export...: Export of selected entities in HPGL (PLT) or SCAPS Archive (SAF) format. See chapter Export.

Print...: Prints the current View2D.This function works only if a printer is installed. See chapter View 2D.

PrintPreview: Shows a print preview of the current View2D. This function works only if a printer is installed. See chapter View 2D.

PrinterSettings: Shows the printer settings dialog.

Exit: Closes the SAMLight software.

9.1.1.1 Job Format

Menu bar → File → Load opens a dialog to load a new job from an SJF file (SCAPS Job Format). On the right hand side, there is a preview window. Directly below, there is a display box of all available entries inside the currently selected job file.

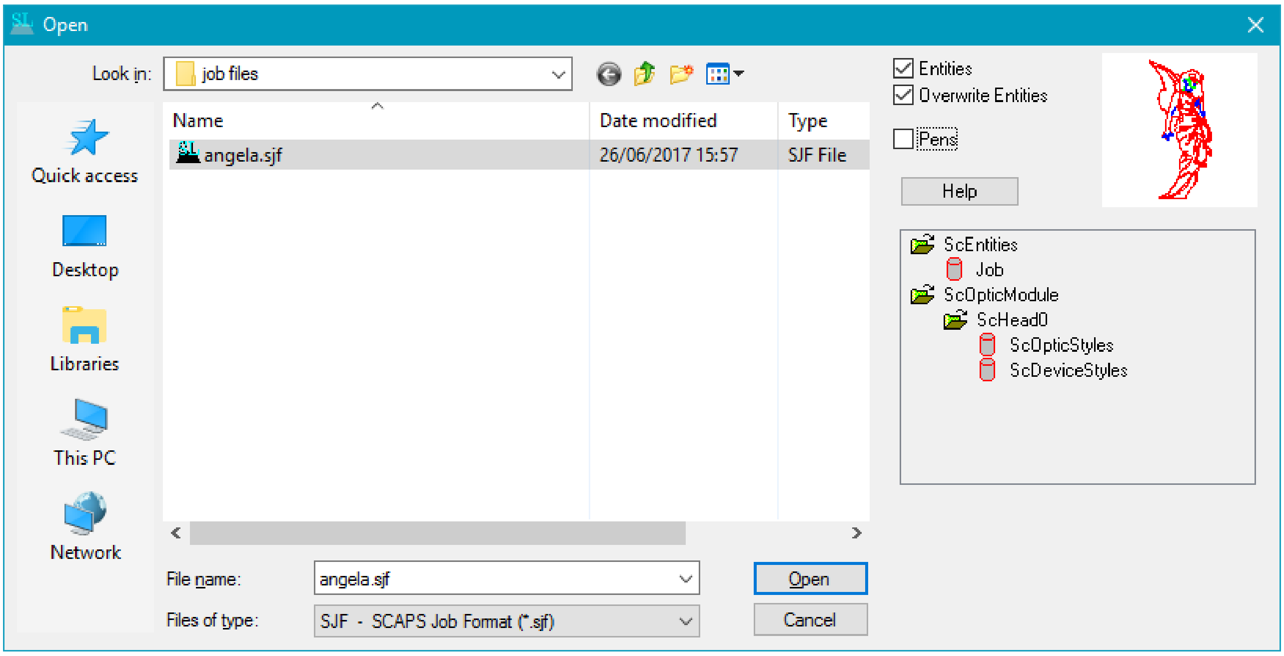


Figure 132: Open File Dialog

Menu bar → File → Save saves the current job. If there is no job name defined it is called *Save as*. *Menu bar → File → Save as* opens a dialog to save the current job under a new name.

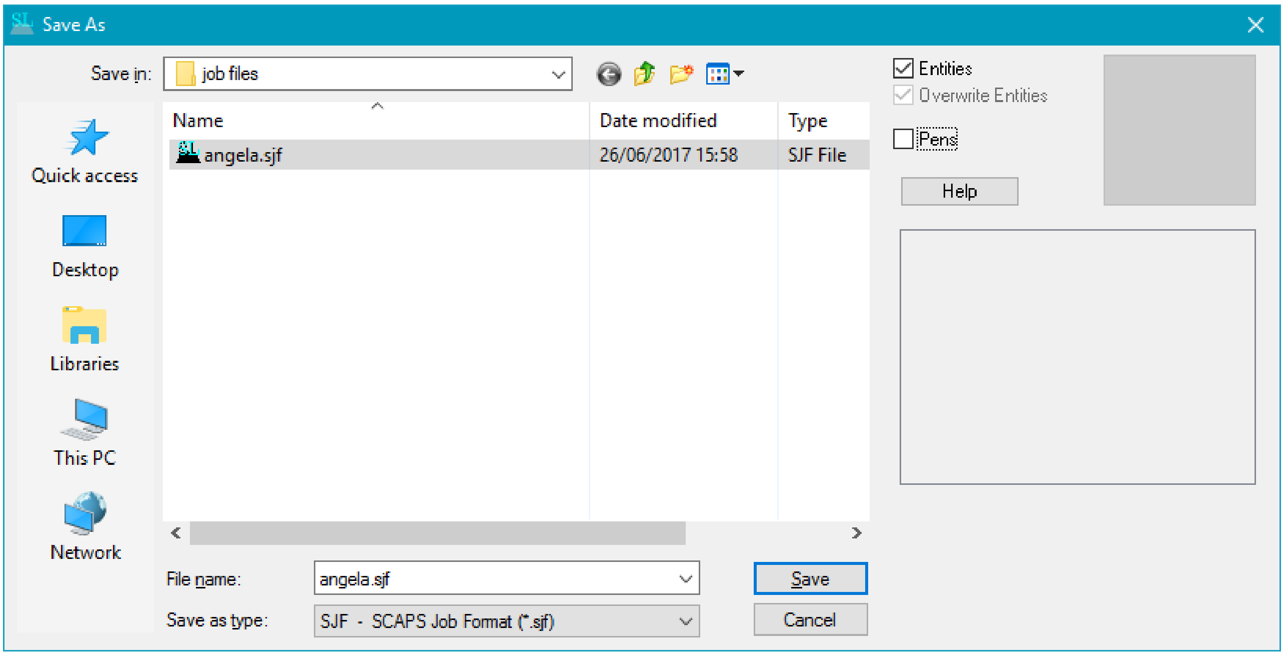


Figure 133: Save File As Dialog

Entities: If selected the entities of the selected file are loaded / saved.

Overwrite Entities: This Check button is only active for dialog Load. If activated the entities of the current job are deleted when the job is loaded. If not the job entities are added to the current job.

Pens: If selected, the pens of the job are loaded / saved. Greyed out if no pens are saved in the job file.

In general, the Pen Settings are saved in <SCAPS>\system\sc_light_settings.sam if "Save Pens On Exit" is enabled or "Save Pens Now" is used.
It is possible to save and load the Pen Settings within a job (.sjf). Be careful not to load any Pen Settings with a job and then accidentally overwrite other Pen Settings when leaving SAMLight while "Save Pens On Exit" is activated.*

It is highly recommended to backup sc_light_settings.sam every time after having done any important or time consuming configuration of the Pen Settings.

The default state of the checkboxes Entities, Overwrite Entities and Pens can be set in the Job Save/Load Dialog.

9.1.1.2 Job Properties

Save job properties in the *.sjf job file

To set the job properties, which have to be saved within the *.sjf job file, go to *File --> Job Properties*... The following dialog window appears:

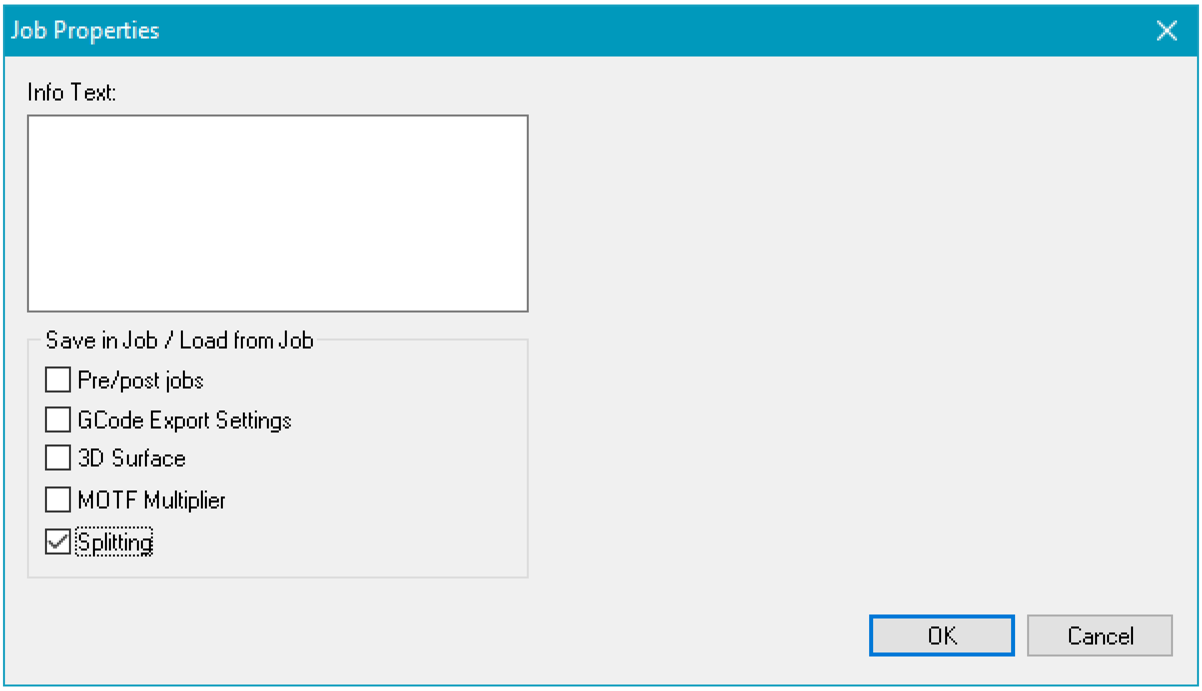


Figure 134: Job Properties Dialog

Info Text: here you can enter any text that will be visible while loading the job (see figure 136).

Pre/post jobs: allows to save Special Job Sequences within the job file. Please note that "Startup Job / Program Start" and "Exit Job / Program End" are not saved in the job specific special sequences to prevent unwanted marking while loading the job or closing SAMLight. Note that loading a job with pre/post jobs stored doesn't change the status of Jobs Toolbar. So, if the Jobs Toolbar is not activated, the pre/post jobs will not be active as well.

GCode Export Settings: the GCode Advanced Styles will be saved with the Job.

3D Surface: 3D Surfaces settings will be saved within the job. It is not necessary to activate the *Enable 3D Surfaces* checkbox in this case. Note that if *STL Projection* is on, only the path to the STL file will be saved within the job, but not the STL file itself. So, if you move the STL file to another directory, the STL file will not be loaded and you will get an error message.

Note as well that the 3D Surface settings will be loaded while loading the job even if the 3D Surfaces Toolbar is deactivated.

MOTF Multiplier: the multiplier value for MOTF functionality will be saved within the job.

Splitting (activated per default): the splitting settings will be saved within the job. Note that the *Enable Splitting* checkbox has to be activated as well (see figure 135)**.** If the checkbox is not activated, the splitting settings won't be saved. Splitting settings for the redpointer will be saved as well. You will find this in the Mark Dialog under Advanced... --> Splitted Redpointer repetitions.

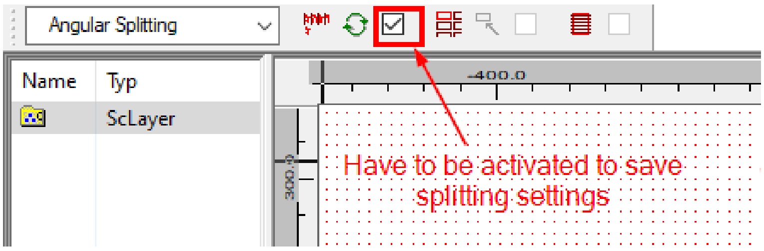


Figure 135: Save splitting settings

Load the Job with stored properties

You can see the job properties saved within the job in the window on the right side of the loading dialog (see figure 136).

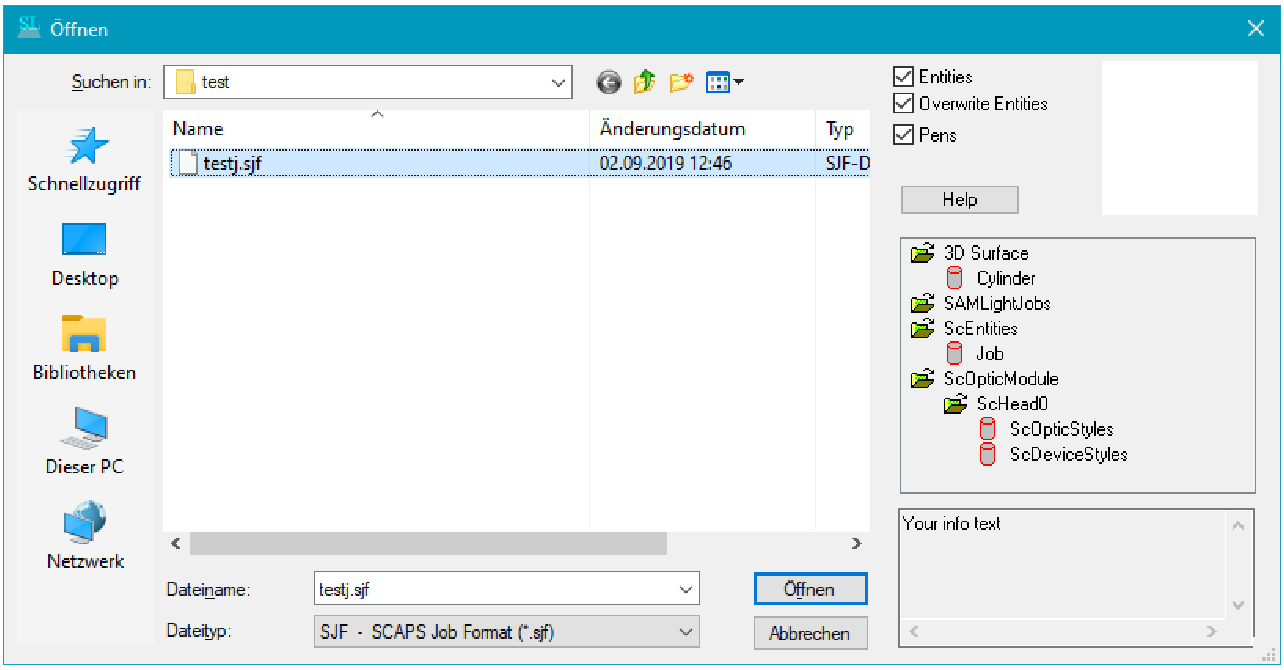


Figure 136: Job loading dialog

9.1.2 Edit

Undo: Undo of the last operation. Not all operations support Undo. Undo is a command that erases the last change done to the current job reverting it back to the preceding state. The opposite of undo is redo, please see above for details.

Redo: Redo of last Undo. The operation that has been reverted by a Undo-operation is re-done.

Move: moves the position of the selected entity or entities in the entity list. The sequence in the entity list determines the sequence of marking.

Delete: Deletes the selected entities.

Duplicate: Copies the selected entities and brings them to view level 1 of the Entity List. See also chapter Entity List.

ArrayCopy: This creates an array of copies of the selected entities (objects). See chapter ArrayCopy.

ArrayPolarCopy: This creates the specified number of copies of the selected entities (objects) arranged in a circular way. See chapter ArrayPolarCopy.

Select: Provides functions for selecting the enteties one by one, etc. Works in the first view level. You can select all, first, previous, next and last entity of your entity list in the job.

Group: Groups the selected entities and puts them into an entities group. See also chapter Object Hierarchy.

UnGroup: Ungroups the selected entities group. The view level of all entities inside the group will be decreased by one. See also chapter Object Hierarchy.

Align...: If at least two objects are selected they can be aligned with the border or center of their outlines. See also chapter Align and Spacing Toolbar.

Spacing...: If at least three objects are selected they can get evenly distributed inside their common outline which is possible in horizontal or vertical direction. To do more specific spacing see dialog Spacing Advanced.

Nudge...: Translates a selected object by a small step. The nudge step is user defined in *Menu bar → Settings → System → View*. The Nudge 10 functions use 10 times of the Nudge step for the translation.

Center...: Translates a selected object so that the center of the working area becomes the center of the object. You can center along horizontal, vertical or both axes.

Rehatch All: Rehatches all entities of the job using the hatch values that are specified for them. When an entity is hatched and afterwards e.g. scaled the hatch lines are influenced by this scaling operation and are no longer conform to the hatch parameters that have been set before. By calling this operation such modifications are removed and the original hatch values are restored.

Set Pen Number: Assigns a pen (1..10) to the current job.

9.1.2.1 Spacing Advanced

The following dialog can be found under *Edit → Spacing → Advanced...*

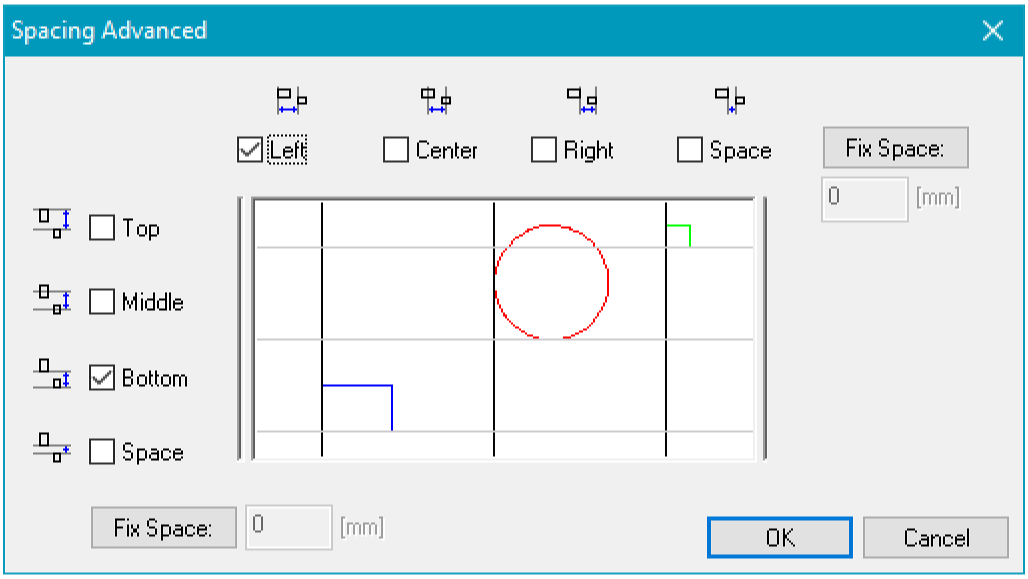


Figure 137: Spacing Advanced Dialog

The view in the middle of the dialog gives an example of the selected spacing.

Left: The spaces between the left outlines of successive elements are set equidistant inside the common outline.

Center: The spaces between the center of successive elements are set equidistant inside the common outline.

Right: The spaces between the right outline borders of each element are set equidistant inside the common outline.

Space: The spaces between the right outline border and the left outline border of the following object are set equidistant inside the common outline.

Fix Space: If one of the described attributes is defined an according fix space can be defined as well.

9.1.2.2 ArrayCopy

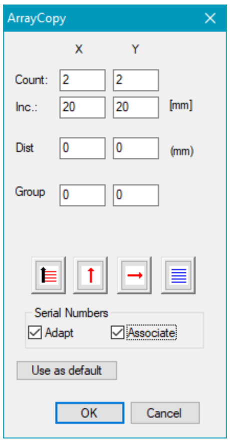


Figure 138: ArrayCopy Dialog

Count: defines how many copies to make for x and y direction.

Inc.: defines the distance between the copies in x and y direction.

The buttons in the middle define where and how to put the copies (this will be interesting for adapting serial numbers). In the case shown in the dialog screenshot above, the copies will be placed (and enumerated in the case of serial numbers).

1. columns first (rows first is possible by clicking on the first button)
2. above (below is possible by clicking on the second button) and right (left is possible by clicking on the third button) of the original one
3. all columns / rows due to 1. are ordered the same way first to last (alternating first to last / last to first is possible by clicking on the fourth button)

Adapt: If checked the ArrayCopy of a serial number would result in an array of serial numbers where the copies are enumerated from the number of the original one up to n. If not checked all copies will get the same number as the original one.

Associate: This function is useful for creating multiple serial numbers with a complex sequence. If checked the ArrayCopy of two and more different serial numbers results in one serial so that no serial number gets repeated. This check button is only enabled if Adapt is selected and association is possible. Creating one serial after an array copy is only possible if the actual values of the current serial differ in a constant step. To get more information, tale a look at the example below.

Example of creating an Associated Array Copy of two different serial numbers:

1. We create two serial numbers and give them two different starting values. We take 1 and 2.

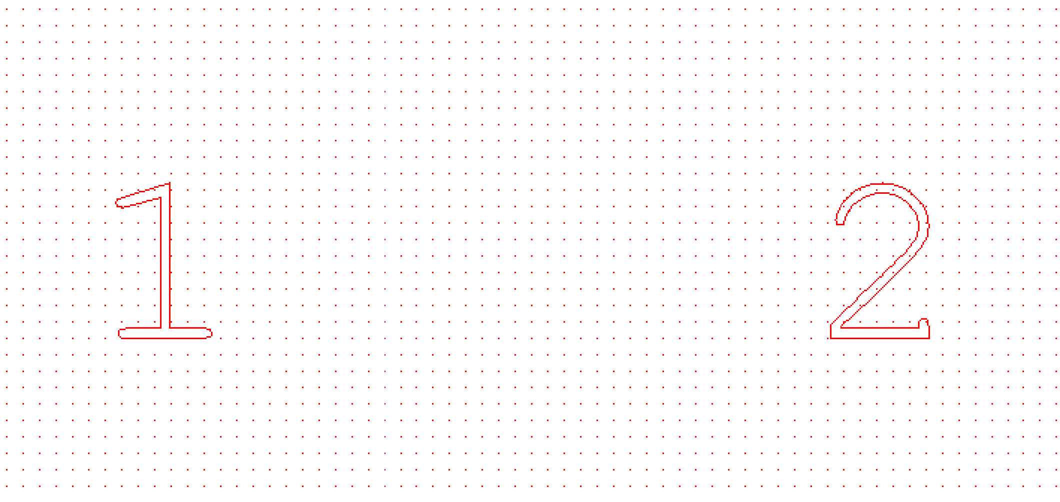


Figure 139: ArrayCopy Creating Serial Numbers

2. Select both serial numbers and choose Array Copy with the right click or going to Edit.

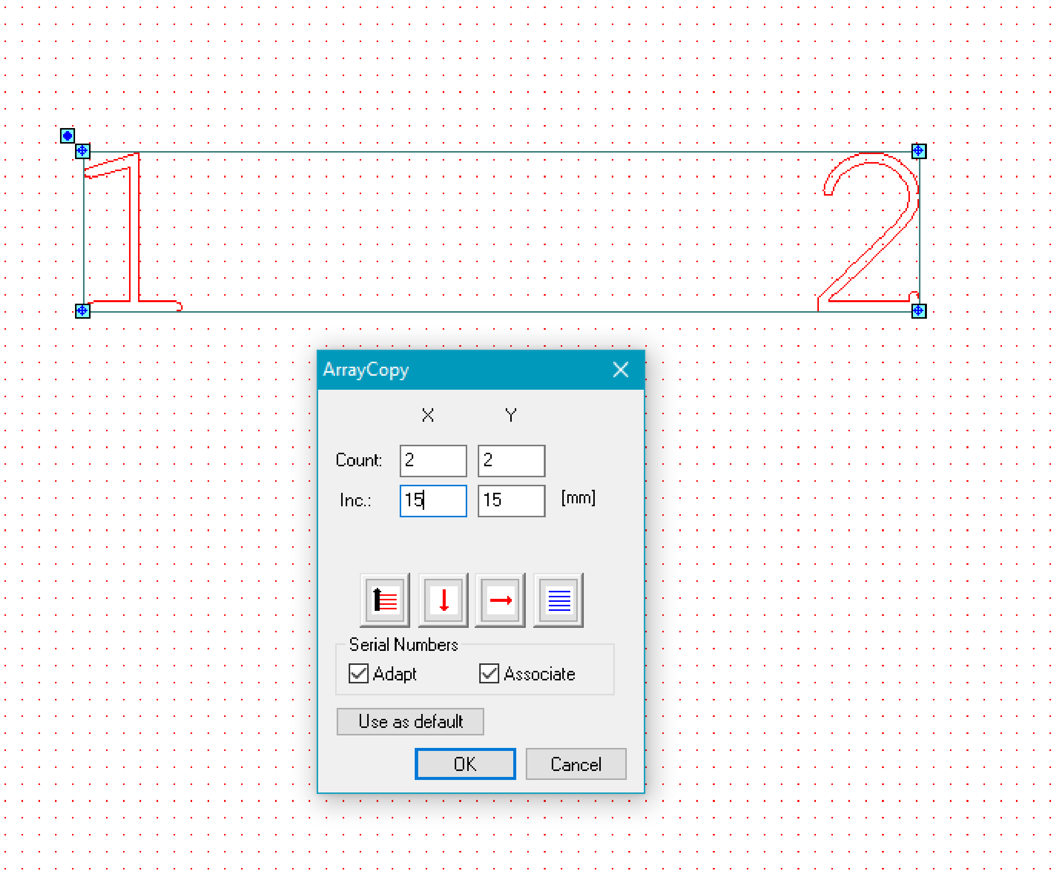


Figure 140: ArrayCopy Serial Numbers

3. Define distances and number of copies you need. Note, that SAMLight will consider two different serial numbers separately and will make N copies of each of them. Choose the sequence. Press OK. SAMLight creates further serial numbers, but now the two serial numbers we started from will not be considered separately and will get a define sequence, so that no serials will be repeated.

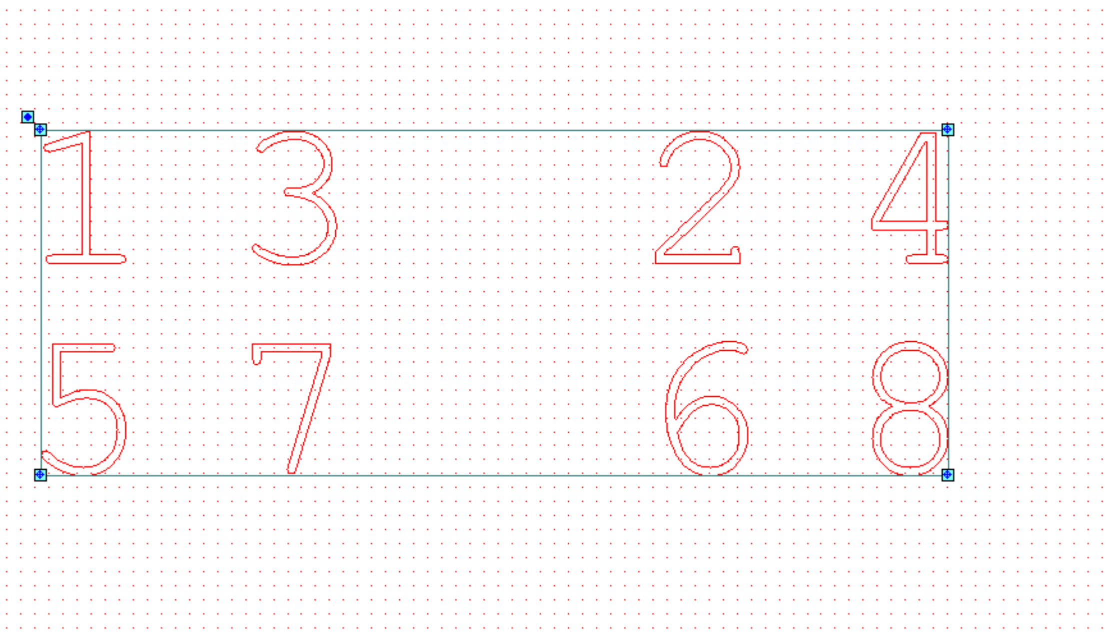


Figure 141: ArrayCopy Serial Numbers Copied

4. After incrementing each serial number will be increased not by 1, but by number of copies you've made. In our case it's 8.

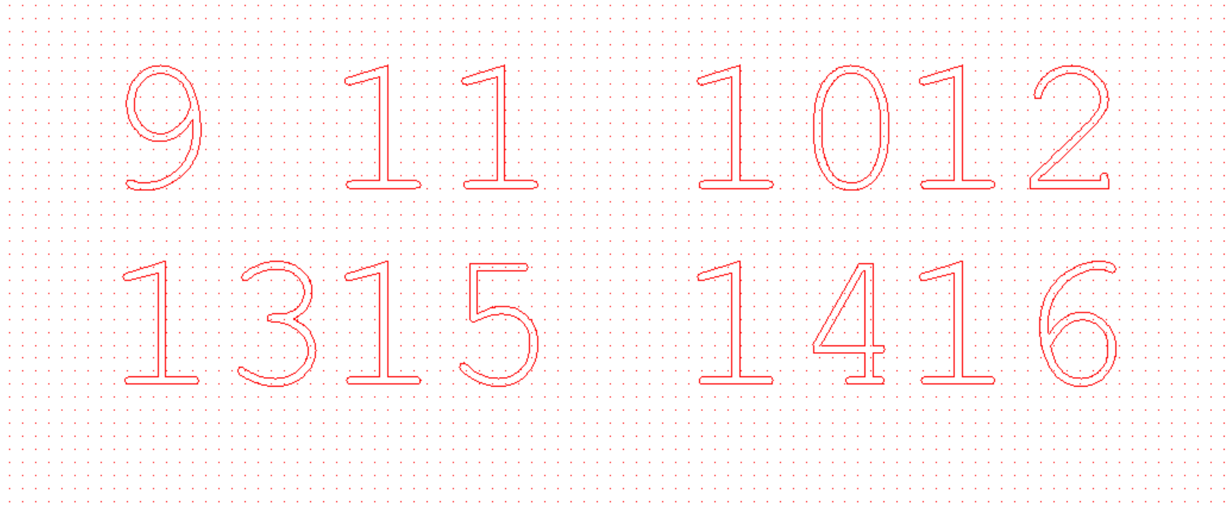


Figure 142: ArrayCopy Serial Numbers Incremented

9.1.2.3 ArrayPolarCopy

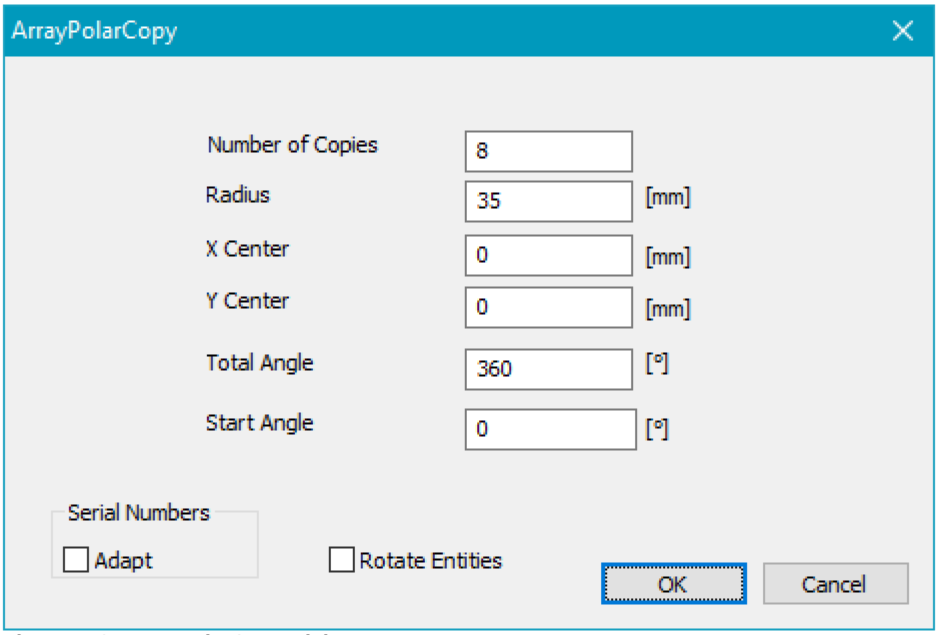


Figure 143: ArrayPolarCopy Dialog

Number of Copies: defines how many copies to make (inclusive original).

Radius: defines the radius of the circle on which the entities will be placed. The absolute position of the circle is specified by **X Center** and **Y Center**.

Total Angle: defines the angle over which the entities will be spread evenly.

Start Angle: defines the angle for the first entity to be placed.

Adapt: If checked, the ArrayPolarCopy copies of a serial number will be enumerated from the number of the original one up to n. If not checked, all copies will get the same number as the original one.

Rotate Entities: If checked, the copies will be aligned radially. If not checked, all copies will be aligned identically to the original entity.

9.1.3 Extras

Topics of Extras:

- Teach Reference - Teach new reference position(s)
- Relocate Reference - Recall and use reference position(s)
- Flash - opens the Flash dialog for standalone operation
- Splitting - split a job into pieces to mark the parts separately on a ring or with a working area that is smaller than the job itself
- Step / Repeat - repeat marking of the same job for a defined time including a movement to modify the position where that job is marked
- Bitmap Marking - adapt and mark a bitmap line by line on a ring
- Motion Jog - Move the axes

9.1.3.1 Teach / Relocate Reference

This feature allows you to define a reference position and relocate entities to this position. The reference position can be chosen with the 'Teach Reference' dialog. The translation can be applied to the job with the 'Relocate Reference' dialog. The red pointer (output bit 3) is active while these dialogs are open to show the position on the working field. This feature can be useful when it is difficult to move the workpiece.

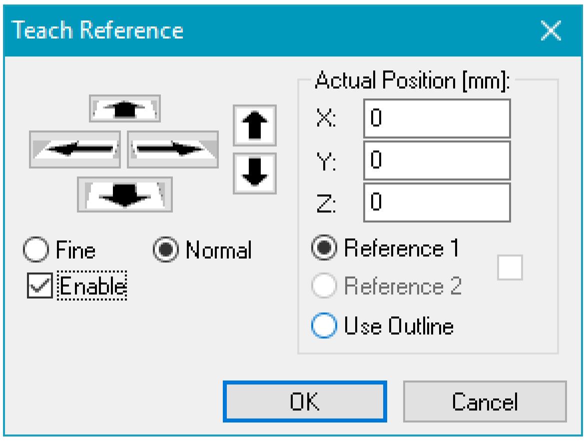


Figure 144: Teach Reference dialog

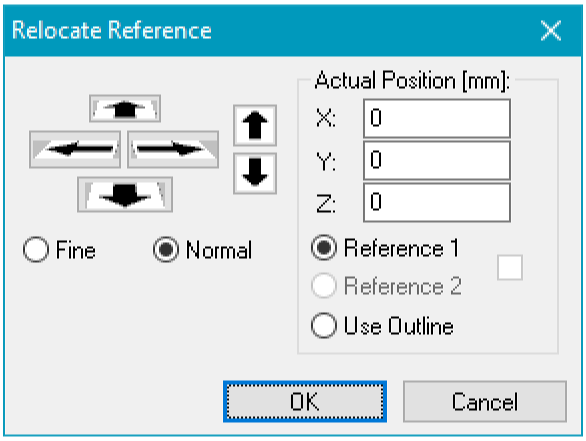


Figure 145: Relocate Reference dialog

A typical process with one reference position contains following steps:

1. Open 'Extras → Teach Reference' and choose a reference position.
2. Leave this dialog with 'OK' to store the reference position into the job (it is recommended to save that job afterwards).
3. Place a workpiece below the scan head.
4. Open 'Extras → Relocate Reference' and move the red pointer to the desired place on the target to get the marking result on the right place of the workpiece.
5. Leave this dialog with 'OK' to shift the entities in the View2D.
6. Mark the workpiece.

The teaching will be done only once, steps 3) - 6) need to be repeated for every workpiece.

NOTE *Relocate uses the speed defined in Settings → System → Optic → Min/Max → max Jump Speed*

For teaching and relocating reference points the related dialogs offer the following functions. These functions can be controlled by the keys described in brackets:

ArrowKeys (cursor up, down, left, right): Move the currently selected reference point in X- and Y-axis. The width of such a movement depends on the stepwidth.

DepthKeys (page up, down): Move the actual reference point in Z-axis (depth coordinate). The width of such a movement depends on the stepwidth.

Fine/Normal (toggle with Shift): Switch between the normal and the fine stepwidth, these values can be configured.

Enable: This checkbox is only available in the teach dialog. It can be used to enable or disable teaching and relocating for a job.

Actual Position: X, Y and Z are absolute coordinates of the currently selected reference points

Reference 1 / 2 (toggle with CTRL): Check *'Settings → System → Extras → Teach Mode, Use two reference points'* to enable a second reference position. With two reference positions it is possible to shift and **rotate** the entities in the View2D.
The one that is currently selected is changed using the cursor keys. If the checkbox between these both radio buttons is selected, the behavior is slightly different. In this case the reference point two is moved relatively to reference point one. This can be used for a raw location of the working piece in a first step to avoid moving wide distances for both reference points separately. If this box is unchecked, both reference points are changed completely independent from each other.

Use Outline: Instead of a single reference point the outline of a job can also be used to select a position. If this option is checked it behaves same as the reference position 1 and can be used to teach or relocate position changes. Because an outline is always a rectangle with its sides parallel to X and Y axis this option can be used only in positioning mode with one reference point.

9.1.3.2 Step / Repeat

The step and repeat marking offers the possibility to repeat the same job for a defined number of times with some specific movements between every marking step. That is useful e.g. in cases when the job has to be marked at different positions on a working piece or to different working pieces that are located on the same working area. The movement can be:

- a rotation of the working piece performed by an external drive
- a planar movement of the working piece performed by external drives
- a planar movement performed by the translation of the current job while the working piece stays on its position

To use this feature following steps are necessary:

- configuration and enabling of the external motion control to perform the automatic movement of the object that has to be marked in case an external drive has to be used
- definition of the total number of repeats that have to be done for the same job during one marking cycle
- definition of the speed the motion controller has to drive the external hardware with between two marking steps in case an external drive is used
- selection of a Step/Repeat mode (Angular or Planar mode) and configuration of its specific settings

Configuring the *Motion Control* has to be done at *Menu bar → Settings → System → Extras*. Here in the field *Motion Control* the checkbox *Active* has to be selected. Additionally the motion control configuration file for the used controller has to be configured. Depending on the *Step/Repeat* mode that has to be used, the motion controller needs to be set up for angular or planar movements. The configuration dialog will only offer these modes and axes for configuration that are enabled using the correct *Motion Control* mode. If the motion controller was configured in a way that no axes are available for rotating or that the wrong axes (not X and Y) are configured for moving, there will be no configuration possibilities available in the *Step/Repeat* settings dialog beside the planar mode that translates the whole job to simulate some kind of movement.
The following dialog can be opened via *Menu bar → Extras → Step / Repeat → Settings.*

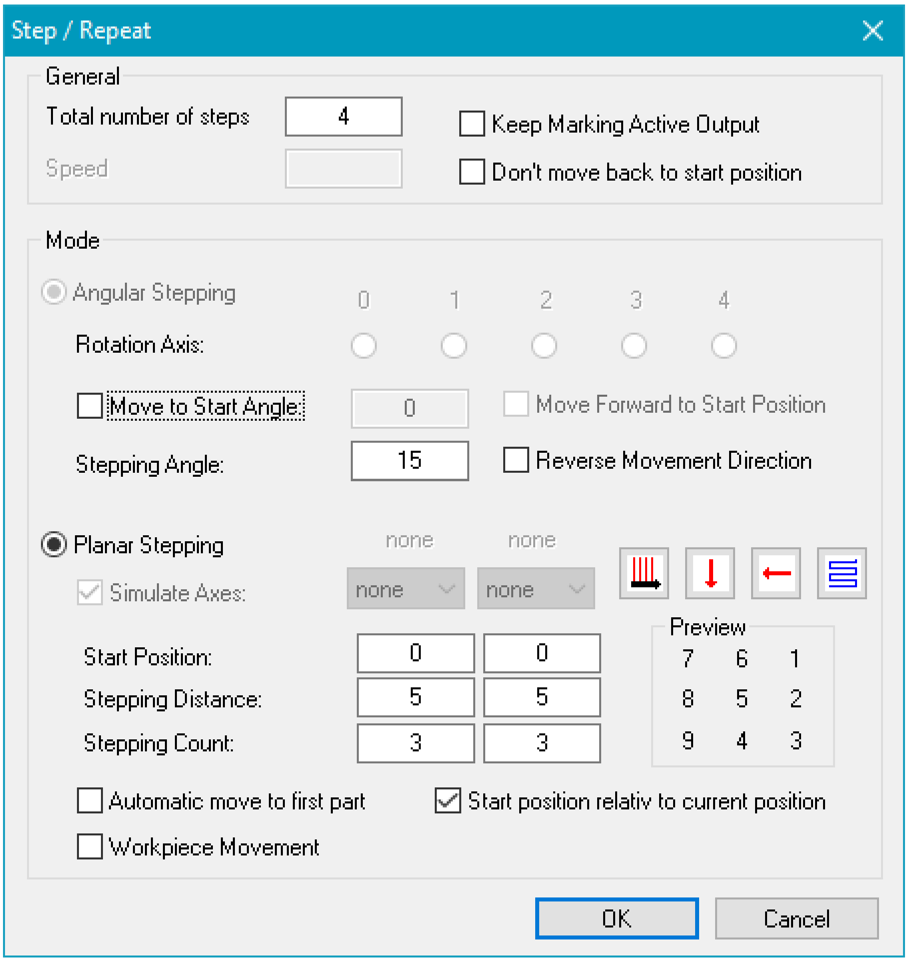


Figure 146: Step / Repeat Dialog

General: At the top of the dialog window the common settings that are valid for all Step/Repeat modes can be made. The *total number of steps* and the motion *speed* can be defined here.

> **Keep Marking Active Output:** In normal operation mode the *digital_output_0* is set to 1 every time a split is marked and it is set back to 0 during the movement operation. This behavior can be changed by enabling the checkbox *Keep Marking Active Output*. If it is active then the marking signal via *digital_output_0* would stay at 1 as long as the complete job including all splits is marked.

> **Workpiece Movement:** If the scan head is not moving but the workpiece is moving instead, then all relative movements have to be inverted.

Mode:

> **Angular Stepping:** The angular mode has to be used for environments where the working pieces can be moved by performing a rotation. Here the following has to be defined:

- which axis has to be used for rotating (X, Y or Z)
- the starting angle (*Move to Start Angle*) and where it has to be moved back after performing a full marking Cycle
- the angle the working piece has to be moved during a step (Stepping Angle)

>> **Move to Start Angle**: This checkbox allows you to enable or disable the movement to the starting angle.

>> **Forward to Start Position:** Here you can force a movement that completes a full rotation instead of going back to the origin.

Planar Stepping: The lower part of the configuration window has to be used for setting up the planar "step and repeat" operation. Here movements within one level and in two directions X and Y can be defined. Accordingly the motion controller needs to be configured in a way that the two axes X and Y are available for planar movements. Else this option can work in movement simulation mode only when the complete job is translated instead. This option can be enabled by setting the checkbox Simulate Axes. In this case no start position can be used. Here it is defined by the current position of the jobs entities within the working area.

For both axes the following values can be set:

Start Position: The starting position to which the motion controller moves to at the beginning of a full Step/Repeat marking cycle

Stepping Distance: The distance the axes have to be moved in a defined direction for every repeated marking

Stepping Count: The stepping count defines how often the job is marked.

Additionally there are four buttons that define the order and direction of the movements that are performed after every mark. Here the following things can be defined:

- the main stepping direction (mark rows or columns first)
- the direction for the Y axis (move in positive or negative direction using the defined distance)
- the direction for the X axis (move in positive or negative direction using the defined distance)
- the movement type (unidirectional or bidirectional)

According to the direction and movement types that are selected here the preview below of these buttons shows the order and positions of the marks like they will be executed.

Other like it is known from the rotary mode here no option exists that allows it to force a movement back to the starting position. Instead of this it is possible to use the option "Move Forward and Back" that performs a special optimized movement. Here the first one goes forward and works exactly like defined by the movement direction buttons but after finishing it, it stays at the reached position. The next movement is reverse to the one defined by the buttons so that it goes back to the starting position.

Automatic move to first part: If the center of the field is at the same position as the entity that has to be marked, this option enables the software to find the starting point automatically.

Start position relative to current position: For moving workpieces on a conveyor or by a motor this option sets the starting position of the following split relative to the actual position.

Don't move back to start position: If this option is unchecked, the scan head moves back to the defined starting angle or position, after the last marking cycle has been done.

Enable Step/Repeat: When all settings are made conform to the used working environment the Step/Repeat marking mode has to be enabled by using M*enu bar → Extras → Step/Repeat → Enable Mode*. Now starting a marking operation no longer results in one single marking cycle. Instead the marking is repeated until the configured number of steps is reached.

9.1.3.3 Bitmap Marking

Bitmap Marking (or Bitmap Rotary) allows you to mark bitmaps on rings continuously without the need to split the bitmap in parts manually. Comparing to the standard splitting functionality this one:

- requires a connected motor, which is controlled via step and direction signals. SAMLight motion type must be "**Type 8 - Generic stepper controller**" or "**Type 14 - USC-2 stepper controller**".
- requires a bitmap, which is not rotated in SAMLight
- modifies the bitmap automatically to fit to the resolution of the drive and the desired bitmap marking parameters; here the dither step value is an important parameter

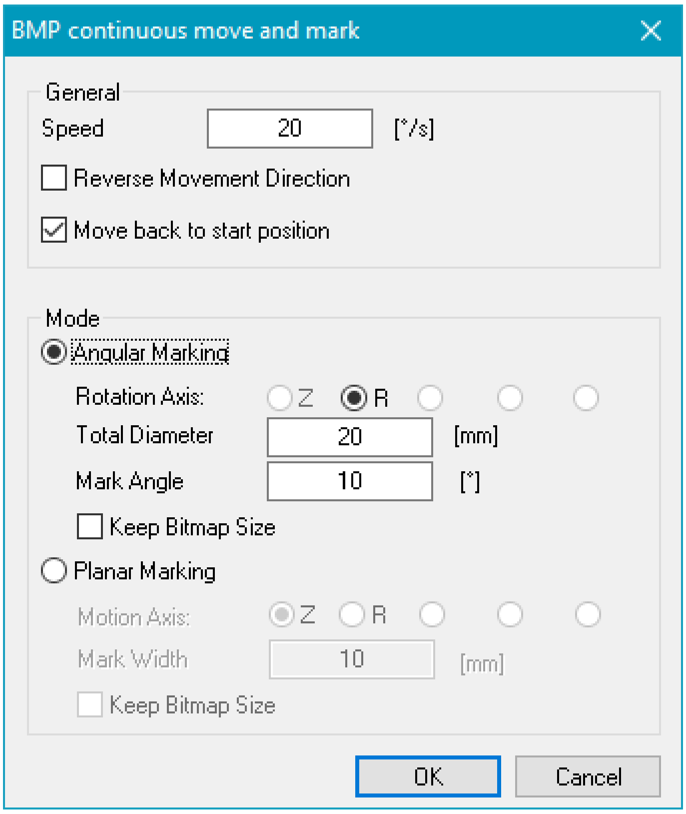


Figure 147: Bitmap Marking Dialog

The following steps are necessary to mark a bitmap along one of its axes:

1. The motion driver has to be configured for rotary or planar marking mode
2. A bitmap has to be imported. Its scanner bitmap has to be created and configured according to the desired results. If the option "Scan XDir" is set for that bitmap the bitmap splitting functionality will recognize this automatically and will perform all checks and calculations using the correct bitmap direction.
3. The Continuous Bitmap Marking setup dialog that can be found in submenu "Settings" has to be called to set up the rotation or motion axis, the diameter of the ring, the marking angle for angular marking as well as the rotation direction and the mark width or planar marking.
4. The Continuous Bitmap Marking feature has to be enabled for the next marking process by *Menu bar → Extras → Bitmap Marking → Enable Mode*.

General:

Speed: The speed the connected drive has to be moved in between two lines of the bitmap

Reverse Movement Direction: Reverses the direction the drive moves e.g. to mark the inner part of a ring

Mode:

Angular Marking:

Rotation Axis: Selects the axis of the drive that has to be used for the movement. This option is important when there are more than one angular axes configured for a motion drive, but it does not influence the splitting direction of the bitmap. If there is no angular axis configured the complete angular marking mode is not available.

Total Diameter: The diameter of the ring that has to be marked. This value is used to check if the laser is able to create a result without gaps. Is the dither step of the bitmap much smaller than the drives resolution the operation would fail.

Mark Angle: Specifies which part of the ring has to be marked with the bitmap. Depending on this value the size of the bitmap is modified to fit. In case the option *Keep Bitmap Size* is not chosen.

Keep Bitmap Size: If that option is set the bitmap is not scaled in order to get a size that results in the specified marking angle, here the source bitmap is left untouched.

Planar Marking:

Motion Axis: Selects the axis of the drive that has to be used for the movement. If there is no planar axis configured the complete planar marking mode is not available.

Mark Width: Indicates the width of the marking.

Keep Bitmap Size: If that option is set the bitmap is not scaled in order to get the size that is specified by the mark width, here the source bitmap is left untouched.

9.1.3.4 Motion Jog (Menu Bar)

All well-defined axes can be moved from the menu bar. By clicking on Extras - Motion Jog a list of the available axes will open. For each axis, you can choose between a small and a big jog step in forward and backward direction depicted with --, -, + and ++. Next to the sign for the stepwidth, the currently defined value is given.

The stepwidth for the big step is defined in Settings → System → Extras → Motion Control.

9.1.4 User

You can login as a defined user at *Menu bar → User → Login*. It is enabled with user password mode, see section User Level.

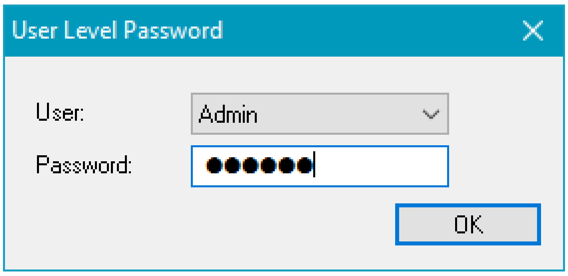


Figure 148: User Level Password Dialog

9.1.5 Window

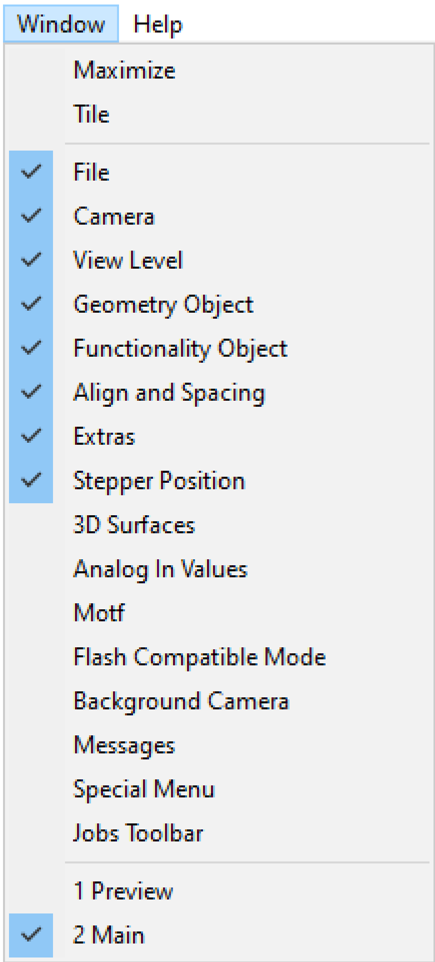


Figure 149: Window

Maximize: Maximizes the actually displayed window (Main Window or Preview Window).

Tile: Shows the Main Window and the Preview Window next to each other.

Toolbars: Choose the needed toolbar to be enabled. This is an additional way to display the toolbars analog to *Settings → System → View → Toolbars.* The configuration for the *Motf* and *Analog In Values* toolbar has to be done there.

Preview: Shows the Preview Window.

Main: Shows the Main Window.

9.1.6 Help

Contents: Opens the help window.

Generate backup zip: Generates a backup zip folder. This folder is saved in <SCAPS>\intermed\ with the name sc_support_YYYY-MM-DD_HH-MM-SS.zip. It is available since version 4_0_5_0004.

The following files are saved in the backup zip folder:

- <SCAPS>\system folder without sc_usc_server.exe

- used correction file

- the job currently loaded in SAMLight

- all jobs stored on the card (for USC-2/3)

- correction file stored on the card

NOTE *The backup zip folder can be used if you would like to update the SAMLight version or setup a new application with the same settings. In additional it is useful for any support requests.*

About...: Shows an information window with the version number.

9.2 Toolbars

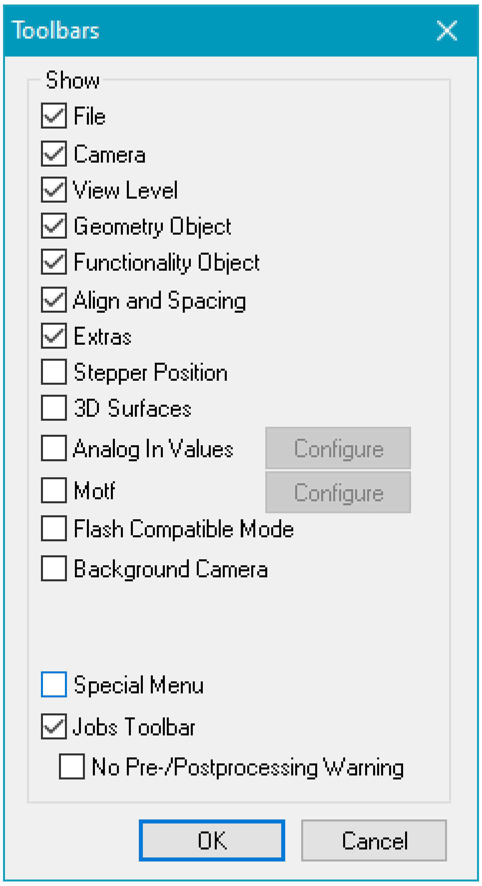


Figure 150: Select Toolbars Dialog

The following toolbars can be activated/deactivated in *Menu bar → System → Settings → View:*

- File Toolbar
- Camera Toolbar
- View Level Toolbar
- Geometry Object Toolbar
- Functionality Object Toolbar
- Align and Spacing Toolbar
- Extras Toolbar
- Stepper Position Toolbar
- 3D Surfaces Toolbar
- Analog In Values
- Marking on the Fly
- Special Sequences Toolbar
- Flash Compatible Mode
- Background Camera Toolbar
- Special Menu
- Jobs Toolbar

9.2.1 File Toolbar

Figure 151: File Toolbar

 **Load:** *Load* an existing jobfile (*.sjf). See chapter Job Format.

Save: *Save* the actual job. See chapter Job Format.

Undo: *Undo* the last operation.

Redo: *Redo* the previously undo operation

Cut: *Cut* the selected entity.

Copy: *Copy* the selected entity.

Paste: *Paste* the copied entity.

Delete: *Delete* an item from the Entity List.

Help: Get context related help for several dialogs and controls.

9.2.2 Camera Toolbar

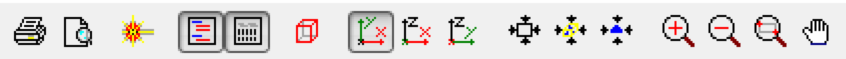


Figure 152: Camera Toolbar

Print: Prints the current View2D.This function works only if a printer is installed. See chapter View 2D.

Print preview: Shows a print preview of the current View2D. This function works only if a printer is installed. See chapter View 2D.

Mark: Opens the Mark Dialog.

Show Entity List: Shows and hides the Entity List on the left side of the View2D. See chapter Entity List.

Show PropSheet: Shows and hides the Entity Property Sheet on the right side of the View 2D. See chapter Entity Property Sheet.

3D View: Select at least one entity and click this button to open the 3D view. This feature is available only with Optic 3D.

Plan View XY: Clicking on this button changes the perspective of the View2D to plain view. This feature is available only with Optic 3D.

Side View XZ: Clicking on this button changes the perspective of the View2D to side view. This feature is available only with Optic 3D.

Side View YZ: Clicking on this button changes the perspective of the View2D to side view. This feature is available only with Optic 3D.

Fit All: Clicking on this button fits the view to the scanner field.

Fit All Entities: Clicking on this button fits the view to all entities in the view.

Fit Selected: Clicking on this button fits the view to the current selected entities.

Zoom Plus: Clicking on this button zooms in by factor 2.

Zoom Minus: Clicking on this button zooms out by factor 2.

Zoom Window: Clicking on this button allows the user to do a user defined zoom window. To define a zoom window follow these steps:

- After clicking this button move the mouse to the first corner of the window.
- Click the left mouse button and keep it pressed.
- Drag the mouse to the second window corner and release the left mouse button.

Hand Tool: Clicking on this button activates the following functions:

- Drag: With the mouse the working space of view 2D can be dragged.
- Free zoom: With the mouse wheel you can zoom in or out relative to the mouse pointer.
- Centered zoom: By holding CTRL and drag with the left mouse button the user can zoom in or out.

9.2.3 View Level Toolbar

Figure 153: View Level Toolbar

View Level Toolbar: This toolbar provides two arrow buttons for increasing or decreasing the View Level of the Entity List. The third button of the View Level Toolbar displays the level number of the actual View Level. For more detailed information see the chapter Entity List.

Upper layer: Go in upper layer.

Lower Layer: Go in lower layer.

9.2.4 Geometry Object Toolbar

Figure 154: Geometry Object Toolbar

Point: Creates a point by clicking the left mouse button at the desired position in the View 2D.

Line: Creates a straight line defined by a start and end point: Click the left mouse button at the desired start position in the View 2D. Move the mouse to the desired end position and click the left mouse button again.

Rectangle: Creates a rectangle in two steps: Click the left mouse button at the desired position of the left upper corner in the View 2D. Move the mouse while keeping the left mouse button pressed to the desired position of the right lower corner and release the left mouse button. To change the geometry of the rectangle after its creation see the chapter Rectangle. To change the position and/or the orientation of the rectangle see the chapter View 2D → Manipulation of objects.

Triangle: Creates a triangle that is defined by its three corners: Click the left mouse button at the desired position of first corner in the View 2D. Click the left mouse button at the desired position of second corner. Click the left mouse button at the desired position of third corner. To change the position and/or the orientation of the triangle see the chapter View 2D → Manipulation of objects.

Ellipse: Creates an ellipse: Click the left mouse button at the desired position in the View 2D. Move the mouse while keeping the left mouse button pressed until the ellipse has the requested size. To change the geometry of the ellipse's after its creation see the chapter Ellipse. To change the position and/or the orientation of the ellipse see the chapter View 2D → Manipulation of objects.

Circle - 3 Points: Creates a circle out of 3 Points. Click the left mouse button at three desired positions one after another in the View 2D. Change the circle by moving those three contact points. Change the segment count within the Geometry Property Page.

Circle - Center Radius: Creates a circle out of two Points: the center of the circle and one point which is an element of circle. Click left mouse button at the two desired positions one after another in the View 2D. Change the circle by moving those two contact points. Change the segment count within the Geometry Property Page.

Arc - 3 Points: Creates an arc out of 3 Points which are elements of the arc. Click the left mouse button at the three desired positions one after another in the View 2D. Change the arc by moving those three contact points. Change the segment count within Geometry Property Page.

Arc - Center Angle: Creates an arc out of 3 Points: the center of the arc and two points which are element of the arc and define the beginning and the end of the arc. Click the left mouse button at the three desired positions one after another in the View 2D. Change the arc by moving those three contact points. Change the segment count within Geometry Property Page.

Spiral: Creates a Spiral by clicking the left mouse button at the desired position in the View 2D.

Polyline: Creates a PolyLine defined by a sequence of points: Click successively the left mouse button at the desired positions in the View 2D to generate a sequence of points. To finish the sequence click the right mouse button and choose one operation of the provided two operations "Finish" or "Close&Finish". To change the position and/or the orientation of the Polyline see the chapter View 2D → Manipulation of objects.

Select: Switch to entity select mode. See chapter View 2D → Selection.

Point Editing: Switch to point editing mode. See chapter Entity List → Point Editor.

9.2.5 Functionality Object Toolbar

Timer: Creates a default timer entity. Please refer to section IO Control Objects.

Wait For Input: Creates a default WaitForInput entity. Please refer to section IO Control Objects.

Set Output: Creates a default SetOutput entity. Please refer to section IO Control Objects.

Set Override: Creates a ScOverride entity. By clicking on it a ScOverride Icon is shown in the entity list. For more information see chapter SetOverride Control Objects.

Set Executable: Creates a ScExecutable entity. By clicking on it a ScExecutable Icon is shown in the entity list. For more information see chapter Executable Control Object.

Set Analog Output: Creates a default SetAnalogOutput entity. Sets a value in percent of the output ports A or B.

Motion Control: Creates a default MotionControl entity. This button is only available if Motion Control is activated in the menu *Settings → System → Extras*. Please refer to section Motion Control.

Motion Control Go: Creates a special MotionControl entity. SAMLight will continue with the next entities without waiting for the motion to be ended. Please remind that the motion command will continue even if the marking in progress signal is off at the end of the entity list.

Date Time: Creates a default DateTime entity. Please refer to section Date Time.

Serial Number: Creates a default Serial Number object. Please refer to section Serial number.

Barcode: Creates a default Barcode entity. Please refer to section Barcode.

Text2D: Creates a default Text2D entity. Please refer to section Text2D.

Jump: Creates a default jump entity. Please refer to section ScJump Control Object.

Motf Offset: Defines an offset for a marking on-the-fly application. When this offset has elapsed a trigger event will be released. For more details on how this feature can used to set up advanced MOTF-jobs, please refer to the section Trigger Control Objects.

Wait For Trigger: The execution of a job is stopped until a trigger event is detected. This can be an external hardware trigger or a trigger signal released by a preceding Motf Offset object. For more details on how this feature can used to set up advanced MOTF-jobs, please refer to section Trigger Control Objects.

Data Wizard: Allows to do different data manipulations on the selected objects. Please refer to section Data Wizard.

Parameter Finder: Allows to optimize pen parameters. Please refer to section Parameter Finder.

9.2.5.1 Data Wizard

The following dialog appears after pressing the magic wand in the object toolbar. All operations are done on the selected polylines.

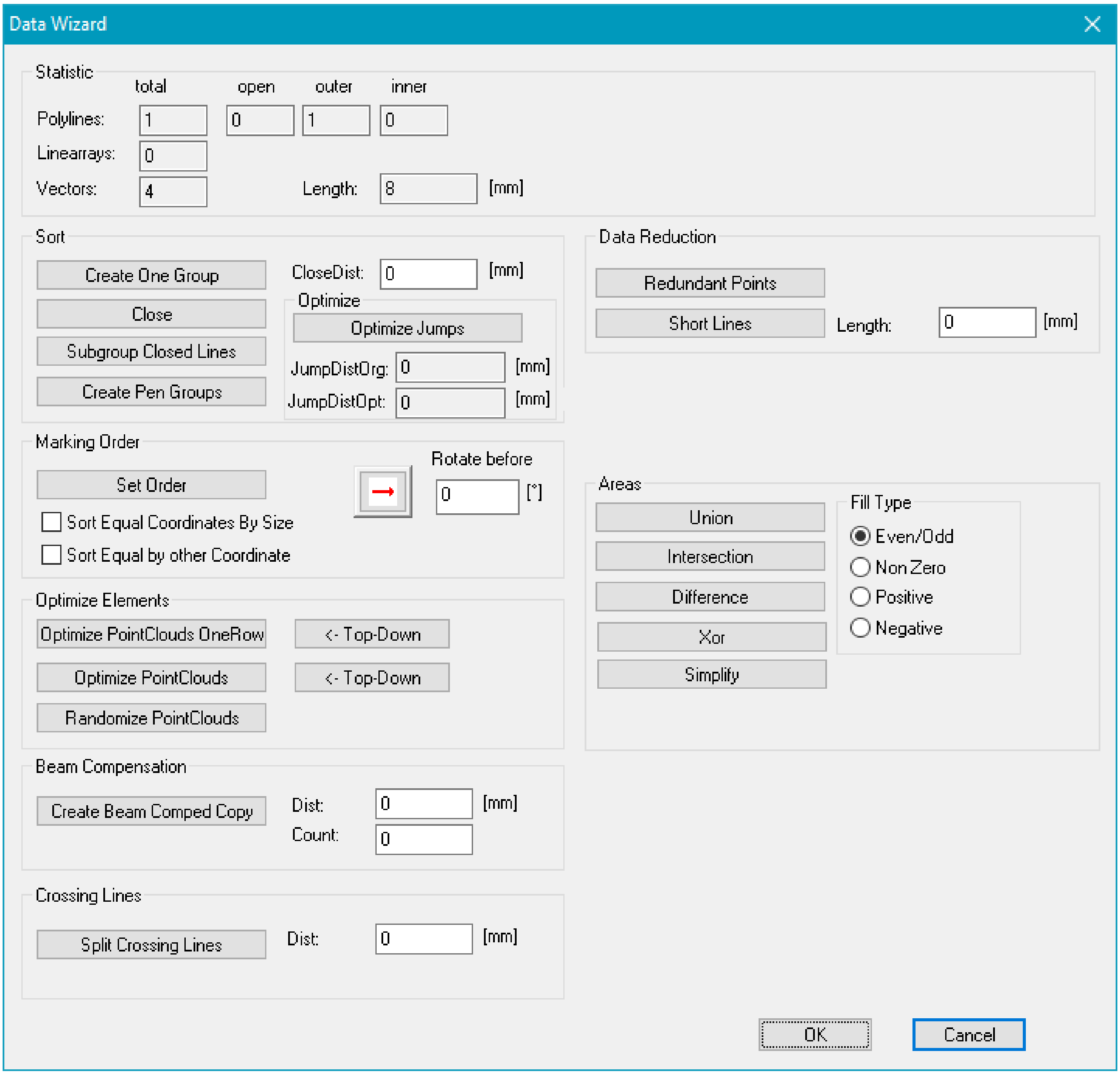


Figure 155: Data Wizard Dialog

Statistic:

total: Number of all objects

outer: Objects which are orientated counterclockwise.

inner: Objects which are orientated clockwise.

Length: Total Length in mm

Sort:

Create One Group: Can be used to optimize the order of the vectors to minimize the marking-time. All selected ScPolyLine2D objects will move into a new ScPolyLines2D folder. Then two polylines will be closed to a new polyline if the distance between them is smaller than CloseDist (in mm). If CloseDist is set to '0' no polylines will be closed. Afterwards all polylines will be sorted in order to minimize jumps. If necessary and possible polylines will be flipped to optimize the marking order of the polylines and thus the marking-time as well.

Close: Sorts and closes open polylines if the distance of open points is smaller than given CloseDist.

Subgroup Closed Lines: Groups selected Subgroups in one Layer.

Create Pen Groups: Puts and sorts all selected objects in one new main group consisting of different subgroups - one for each pen. Depending on the laser type, this can reduce marking time when more than one pen is used for marking because the number of switches between the different pens is reduced.

CloseDist: Distance below open polylines are closed.

Optimize Jumps: the total jump distance between poly lines would be optimized. This feature is only designed for vector graphics. Before clicking the button, a group of entities must be chosen first. After clicking OK, a new group of entities with optimized distance would be created on the position of the original entities. At the end, the original entities are to be deleted. This step is suggested to be made at last, because the optimized group could not be edited as the original entities. The home jump distance is not calculated in the algorithm. If the home jump is not enabled, the default start position is (-10,000, -10,000, -10,000). That means, enabling home position or not may differ the optimized result.

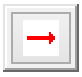
JumpDistOrg: shows the total jump distance of the original group entities.

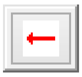
JumpDistOpt: shows the total jump distance of the optimized group entities.

Marking Order:

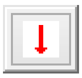
The field Marking Order provides the functionality for automatically arranging the marking order of a chosen group of entities.

- Click the button containing a red arrow to change the primary sorting rule.

 sorts entities in ascending order according to the x-coordinate of their most left points.

 sorts entities in descending order according to the x-coordinate of their most right points.

 sorts entities in ascending order according to the y-coordinate of their lowest points.

 sorts entities in descending order according to the y-coordinate of their highest points.

- activate the check box to enable the secondary sorting rule.

Sort Equal Coordinates By Size: the entities with the same coordinate would be sorted by their size from small to large.

Sort Equal Coordinates By Other Coordinate: the entities with the same coordinate in the primary sorting rule would be sorted by the other coordinate from most left to most right or from lowest to highest.

- **Rotate before:** rotates the chosen entities counterclockwise around the source point by the number defined in the text box in degree before sorting.

Performs the sorting accordingly to the settings above.

Optimize Elements: The following options work for ScPointCloud2D entities.

Optimize PointClouds OneRow: Sort point clouds in stripes with x direction. The width of a stripe is the side length of a square with the area of the point cloud entity divided by the number of points. This kind of sort should improve the marking speed of point clouds.

Optimize PointClouds: Sort point clouds in stripes with x direction. The width of a stripe is 3 times the side length of a square with the area of the point cloud entity divided by the number of points. This kind of sort should improve the marking speed of point clouds.

Randomize PointClouds: Resort point clouds randomly distributed.

Beam Compensation:

Create Beam Comped Copy: This option creates scaled down inner copies or rather scaled up outer copies of each selected object depending on the orientation of the object.

Dist: Distance between the created copies.

Count: Number of copies.

Crossing Lines:

Split Crossing Lines: This feature is designed for polyline, number and text to avoid double marking on the point of two crossing lines. The split will be made on a copy in the same position of the original entities.

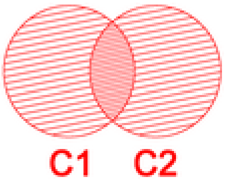
Dist: Distance of the split into two parts at the breaking point.

Data Reduction:

Redundant Points: Removes points of a straight line which do not define the straight line.

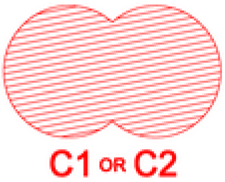
Short Lines: Removes points which are located on a polyline and which have a small distance to the neighboring points of the polyline.

Length: describes the distance of the points.



Areas: There are four boolean operations to combine two areas, for example C1 and C2:

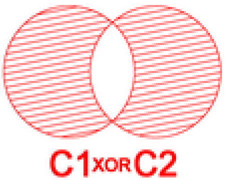
Do not hatch the objects before using the data wizard. Afterwards a new entity is created and this can be hatched depending on the chosen type. Those are explained below.



Union (OR): New area where either C1 or C2 or both are filled

Intersection (AND): New area where only the intersection of C1 and C2 is filled

Difference (NOT): New area where everything that is not C2 is filled



Exclusive Or (XOR): New area where only the non-intersecting areas are filled

Simplify: Self-intersections from the supplied polygon will be removed by performing a boolean union operation.

Fill Type: The filling rules define the handling of areas and holes. There are four filling rules for the four boolean operations:

Even/Odd: Only odd numbered sub areas are filled.

Non Zero: All non zero sub areas are filled.

Positive: All sub areas with winding counts bigger than 0 are filled.

Negative: All sub areas with winding counts smaller than 0 are filled.

Manual Split:

Splits the selected Polyline with respect to the X,Y or Z axis at the desired co-ordinate.

Round:

Each X or Y coordinate of the selected entity will be rounded to the next full value given by the input field in [mm].

NOTE *Some of the functions change the organization of the selected object. For example: If using "Create One Group" on a serial number this will change it into a plain text entity.*

9.2.5.2 ParameterFinder

The Parameter Finder is a tool that helps to optimize pen parameters and the hatch parameters. It is structured as a wizard which guides the user through the different steps of:

1. Define the pen or the hatch which should be optimized, the grid and the entity to be used for testing
2. Choose the "X Parameter" to be varied in horizontal direction
3. Choose the "Y Parameter" to be varied in vertical direction
4. Mark, selection of the best result, fine tuning.

When clicking on the icon of the Parameter Finder, the following dialog appears:

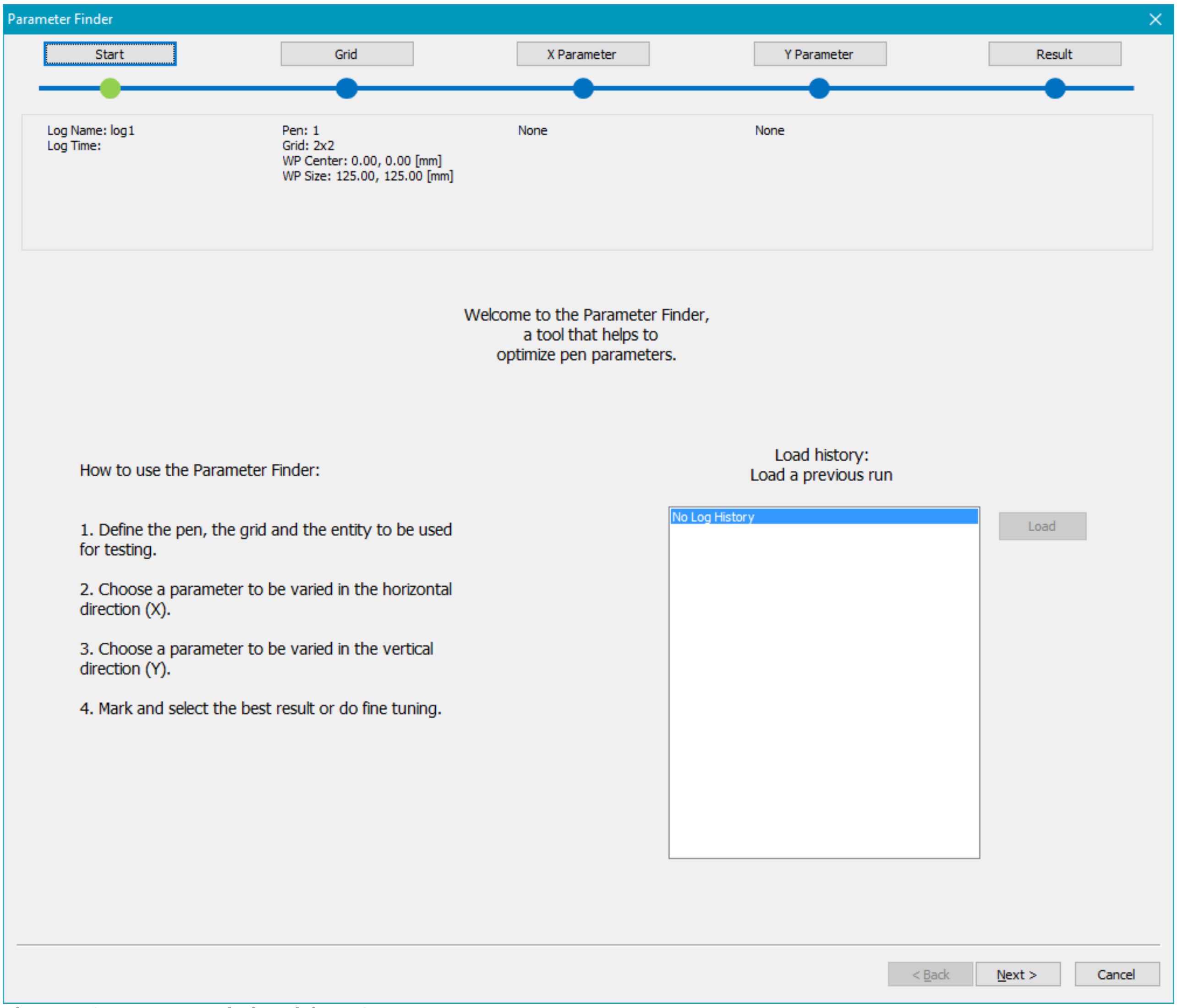


Figure 156: Parameter Finder Dialog - Start

Navigation is possible either with the buttons "Next" and "Back" or by clicking directly on the button representing the desired page in the navigation line at the top.
On the blue line, the current page is highlighted in green.
For each button of the navigation line, a summary of the currently applied values is given in the box underneath.
For each marking, a log is created which can be used to restore the parameters used for that particular marking by clicking on the load button.

<u>1. Define the pen or the hatch which should be optimized, the grid and the entity to be used for testing</u>

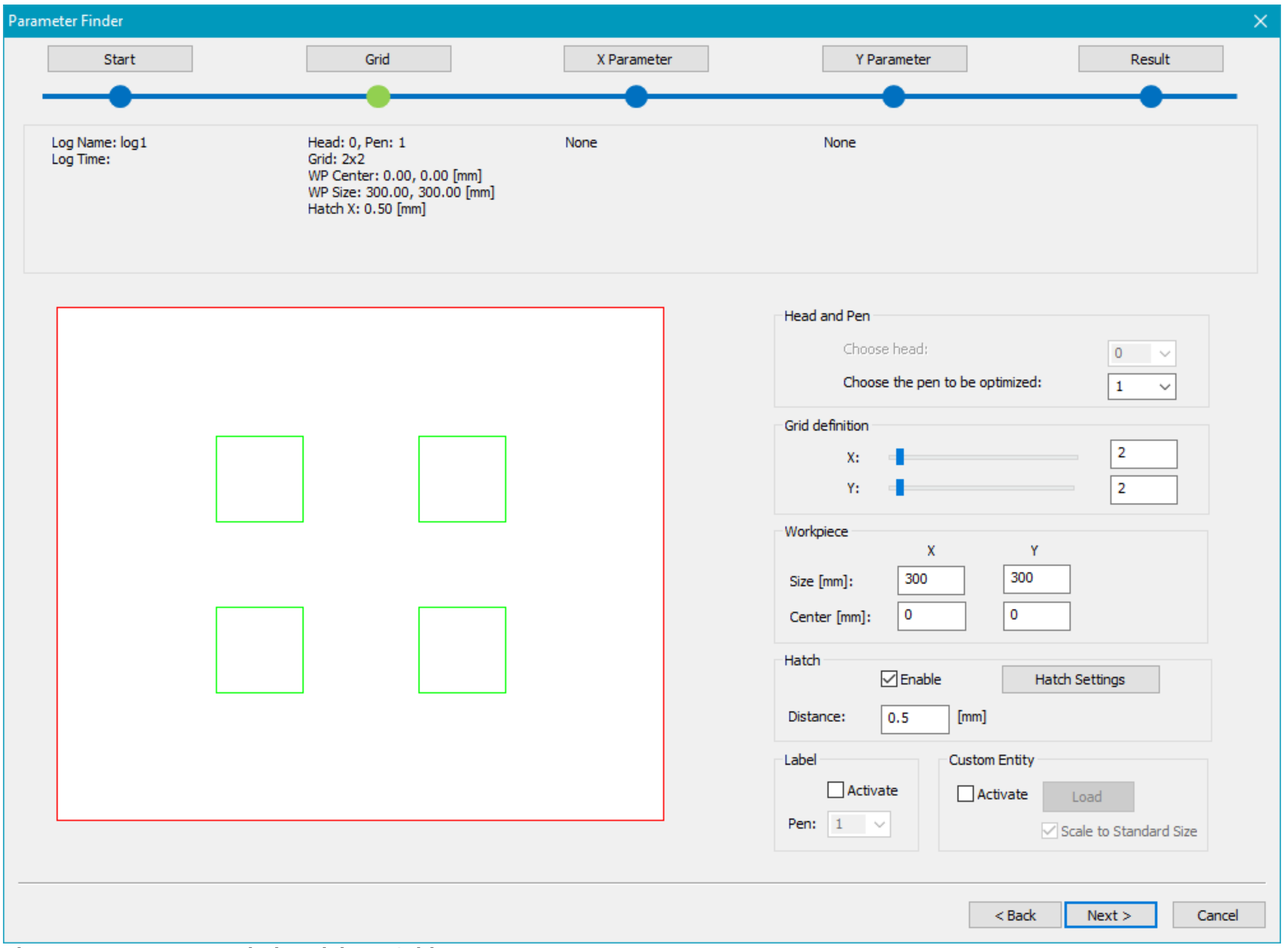


Figure 157: Parameter Finder Dialog - Grid

Preview: A preview is shown on the left half of the dialog. Changes made in the parameter definition will be automatically displayed in the preview.

Pen: Use the drop down menu to choose the pen which should be optimized. The parameter values set in this pen will be taken as start values for the parameters to be chosen on the pages X Parameter and Y Parameter of the Parameter Finder.

Grid definition: Define the total amount of entities in X and in Y direction.

Workpiece: Define the size and center of the workpiece. This is the area on which the entities of the grid will be equally distributed automatically.

Hatch: Activate a hatch. If activated the distance between individual hatch lines can be adjusted.

Hatch Settings: In this window more hatch settings can be adjusted like the hatch settings on the property page.

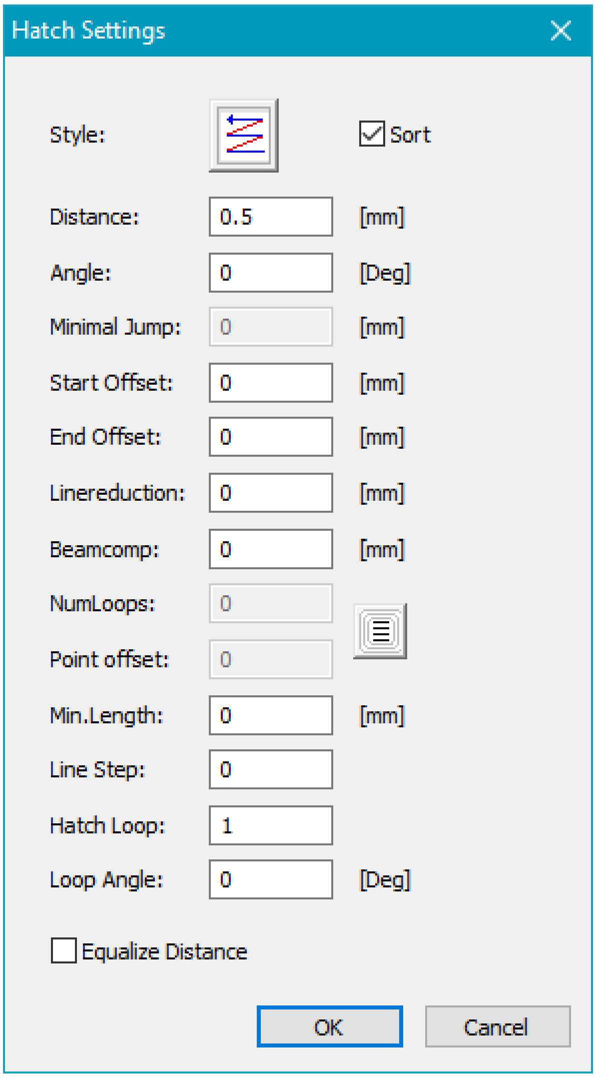


Figure 158: Parameter Finder Dialog - Hatch Settings

Label: Activate a label to be marked for each single entity. This label can be marked with a pen different than the pen which was chosen for optimization before.

Custom Entity: The default entity is a square which is scaled automatically with the grid size. To use an entity different than a simple square, the functionality "Custom Entity" can be used. When activated, a job can be loaded. The entities in this job will then be taken at each entry of the grid. Scaling of the job is possible to make sure it fits within the workpiece. The preview always shows rectangles as a representation of each entity or job.

<u>2. Choose the "X Parameter" to be varied in horizontal direction</u>

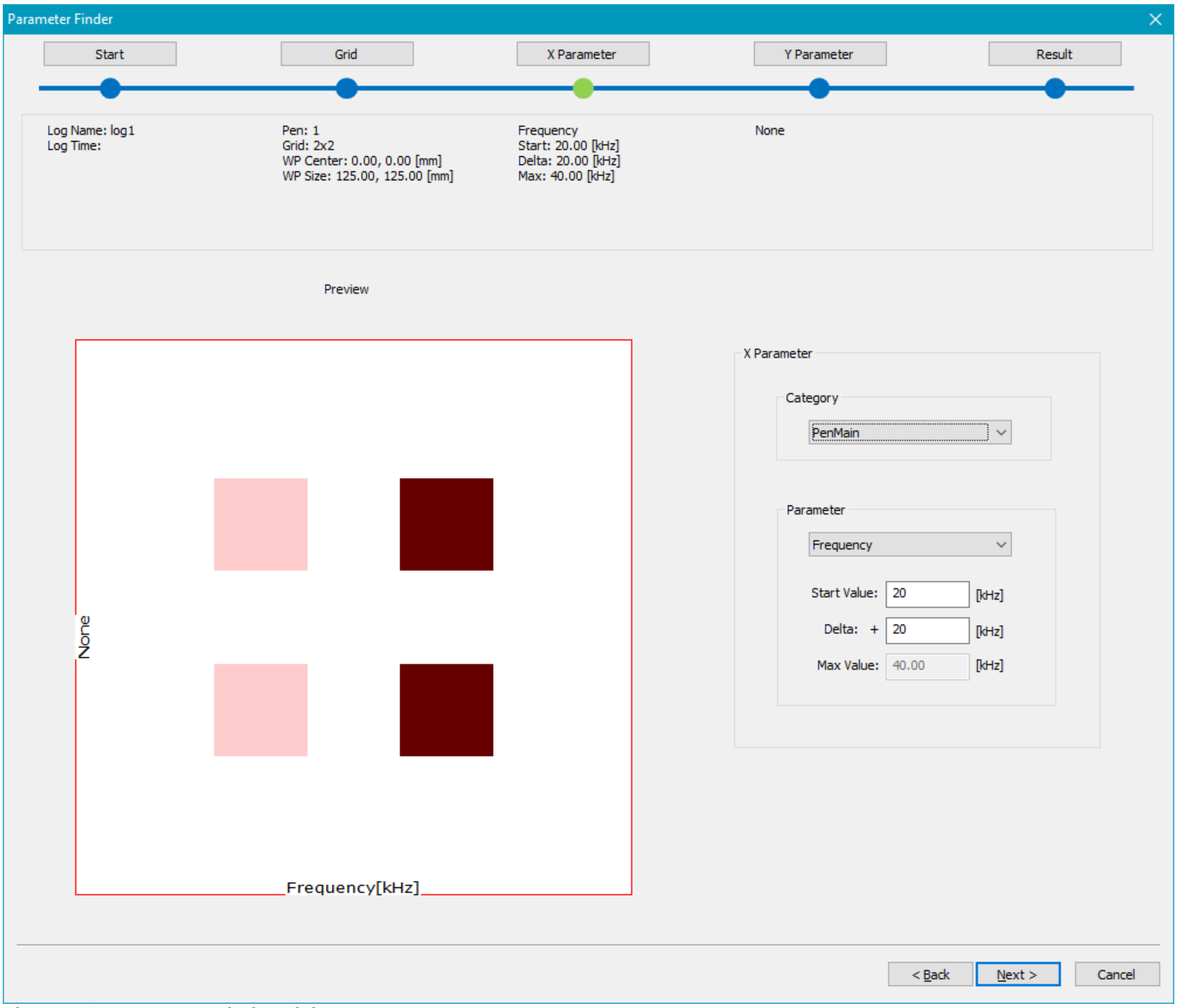


Figure 159: Parameter Finder Dialog - X Parameter

Preview: The color gradient indicates the variation of the parameter chosen in X direction. A label is given on the bottom according to the parameter chosen.

X Parameter: Specify the parameter which should be varied in horizontal direction.

Category: Choose a category from the drop down menu. Parameters are categorized according to the pen property pages and the hatch property page.

Parameter: Choose a parameter from the drop down menu. Per default, the current value of this parameter given in the pen is taken as start value. The start value and the delta (difference between two entities in X direction) can be adjusted. The resulting maximal value is calculated automatically.

3. Choose the "Y Parameter" to be varied in vertical direction

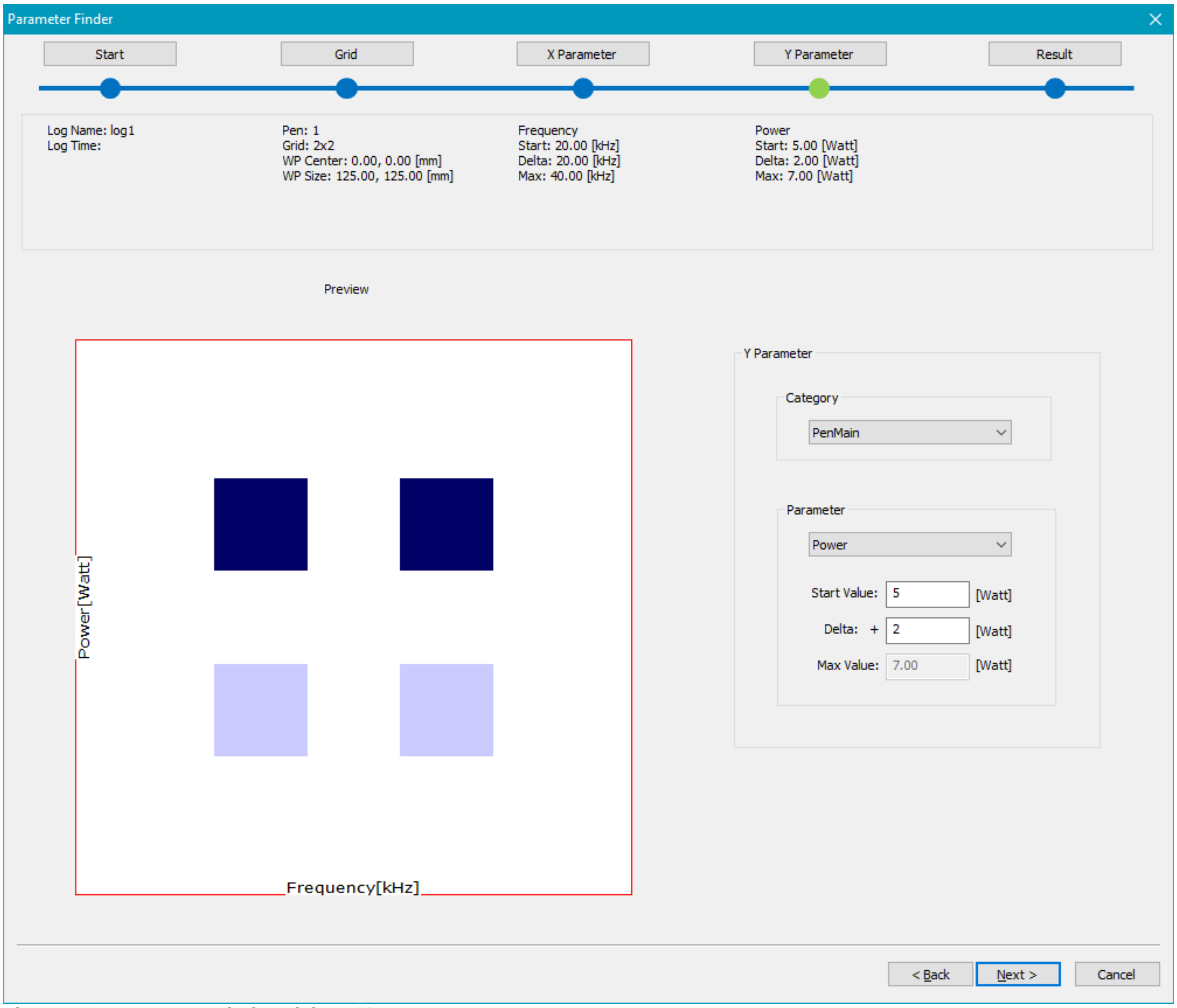


Figure 160: Parameter Finder Dialog - Y Parameter

Preview: The color gradient indicates the variation of the parameter chosen in Y direction. A label is given on the bottom according to the parameter chosen.

Y Parameter: Specify the parameter which should be varied in vertical direction.

Category: Choose a category from the drop down menu. Parameters are categorized according to the pen property pages and the hatch property page.

Parameter: Choose a parameter from the drop down menu. Per default, the current value of this parameter given in the pen is taken as start value. The start value and the delta (difference between two entities in Y direction) can be adjusted. The resulting maximal value is calculated automatically.

4. Mark, selection of the best result, fine tuning

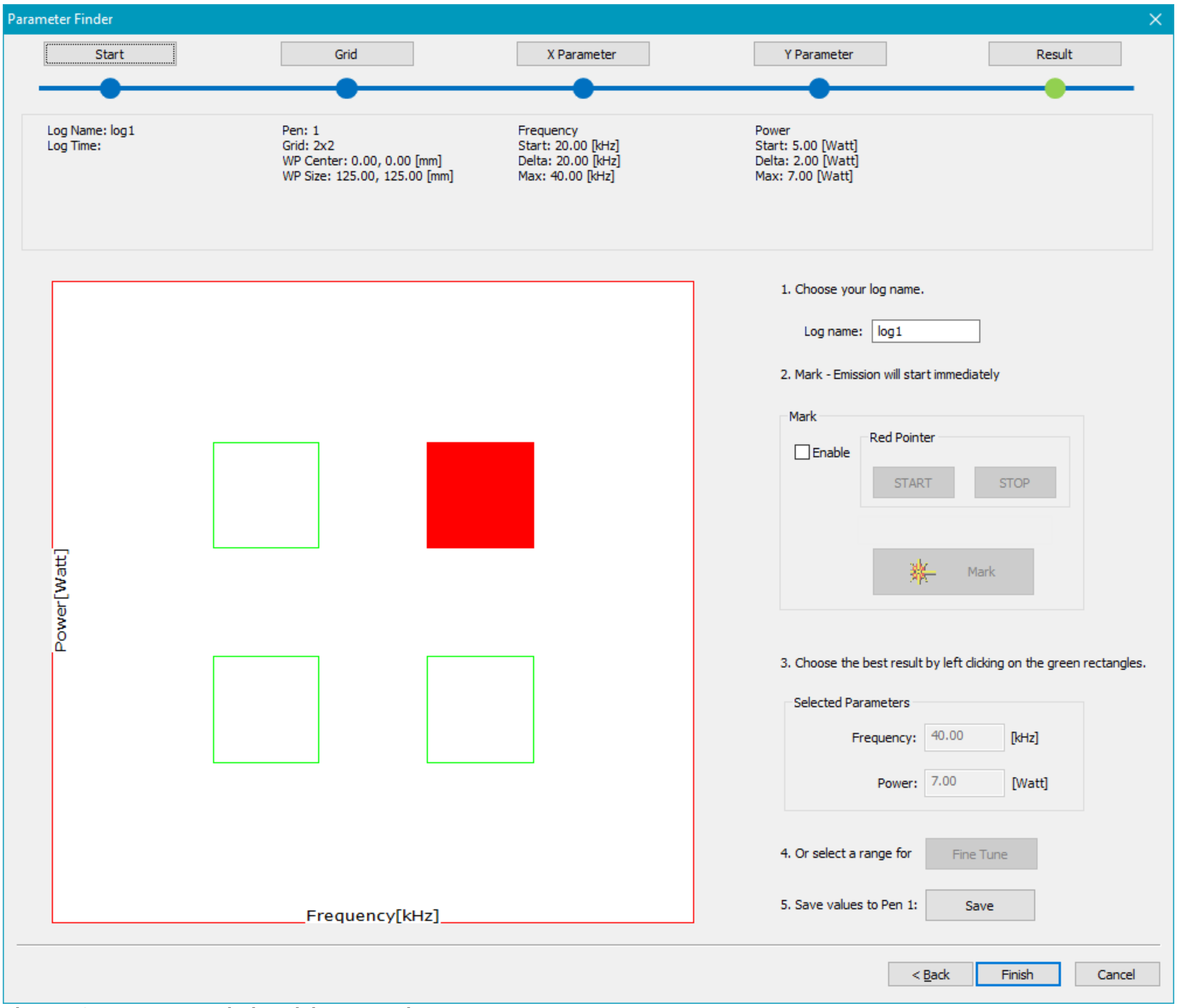


Figure 161: Parameter Finder Dialog - Result

Preview: On this page, the preview is used for selection of the best result (left click, red) or selection of an area for fine tuning (right clicking, blue).

1. Choose a log name: For each mark, a log is created which can be loaded on the Start page of the Parameter Finder. Specify the name of this log here.

2. Mark - Emission will start immediately: When enabled, the red pointer can be used with the buttons Start and Stop. Clicking on the mark button will start the marking.

3. Choose the best result by left clicking on the green rectangle: After having marked, the best result in reality can be detected and chosen by left clicking on the respective rectangle in the preview (will be highlighted in red). The exact parameter values will be given in the fields of "Selected Parameters". To save the chosen values to the pen, click on the Save button in point 5.

NOTE

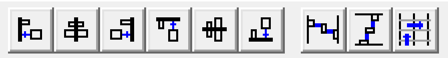
It is not possible to save the chosen parameters for the hatch. The selected parameters must be entered manually in the hatch property page.

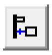
4. Or select a range for: If no best result can be identified, an area can be chosen for fine tuning. The area can either be chosen by right clicking on several rectangles or by dragging with pressed left mouse button (area will be highlighted in blue).

5. Save values to Pen x: If a best result has been chosen, the parameter values can be stored to the pen chosen on the Grid page. It is not possible to save the hatch parameters to the hatch property page.

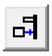
Mark Pre-/Postprocessing feature is available for Parameter Finder.

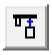
9.2.6 Alignment and Spacing Toolbar

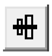


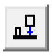
 **Align Left:** Active if at least two objects are selected. The objects are aligned left to the left outside object.

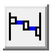
 **Align Center:** Active if at least two objects are selected. The objects are aligned horizontally to the center of their common outline.

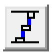
 **Align Right:** Active if at least two objects are selected. The objects are aligned right to the right outside object.

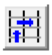
 **Align Top:** Active if at least two objects are selected. The objects are aligned top to the top outside object.

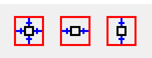
 **Align Middle:** Active if at least two objects are selected. The objects are aligned vertically to the center of their common outline.

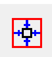
 **Align Bottom:** Active if at least two objects are selected. The objects are aligned bottom to the bottom outside object.

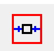
 **Spacing Horizontal:** Active if at least three objects are selected. The objects are distributed evenly inside their common outline.

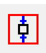
 **Spacing Vertical:** Active if at least three objects are selected. The objects are distributed evenly inside their common outline.

 **Spacing Advanced:** Opens a dialog where more specific spacing can be done. See dialog Spacing Advanced.



 **Center Both:** The selected entities would be centered both horizontally and vertically.

 **Center Horizontally:** The selected entities would be centered horizontally.

 **Center Vertically:** The selected entities would be centered vertically.

9.2.7 Extras Toolbar

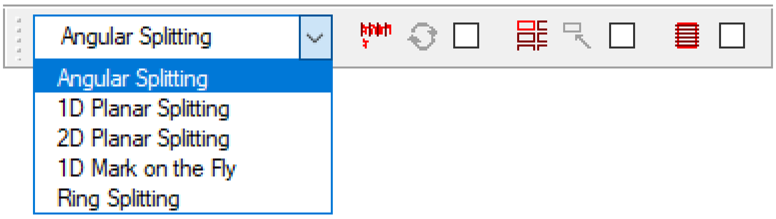


Figure 162: Extras toolbar

This toolbar can be used to directly access the functions for Splitting a job and for Step/Repeat marking. After the same functionality is available via the related Extras menu items it is disabled by default. This toolbar can be activated within the settings before it can be used. It offers following functionalities:

Drop down menu: This menu allows you to select a certain splitting mode. Then this mode can be edited using the button *Splitting Settings*.

Splitting Settings: This button offers direct access to the splitting settings dialog where several splitting parameters can be configured.

Re-split Job: This button is enabled when the splitting mode is activated for a job. This is the case when the checkbox to the right of this button in the toolbar is enabled. It can be used to split the current job again to update the split data.

Step/Repeat Settings: This button offers direct access to the settings dialog for the Step/Repeat parameters.

Reset Position: In case the Step/Repeat mode is activated (the check box right beside that button is selected) it can be used to reset the position of the current object. That causes - depending on the Step/Repeat mode - either a repositioning of the used geometry to its original position or a movement of the used drives so that the starting position is reached.

Bitmap Splitting: If a scanner bitmap is present in the entity list then this bitmap can be split in order to mark on a round surface.

9.2.8 Stepper Position Toolbar

This toolbar is available only if USC-2 stepper type 14 is used. If activated you can see the axis position of all defined axis. The values corresponds to the values displayed in the control property page. The units are mm for straight axis and ° for angular axis.

Figure 163: Stepper position toolbar

NOTE

The stepper position toolbar might turn red for the following reasons:

1. External stop is active.(All axes show red)
2. Home switch is pressed.

If an axis does not move when homing is triggered, inversion of the polarity of the ref IOs might help.

A homing procedure consists of 2 parts: 1 move on the switch with refspeed and 2 leave the switch with refspeed2, which is usually much slower than refspeed. If the polarity of the reference IO is incorrect, part 2 of the homing process is executed at first. No movement is done because the switch is already left. To invert polarity of ref IOs, please see refvalue.

9.2.9 3D Surfaces Toolbar

This toolbar is available only if the option Optic3D license is present. The toolbar is showing mode drop-down list , settings button and enable checkbox. How to use the 3D Surface feature is explained here.

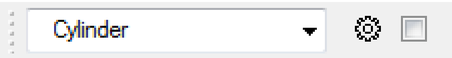


Figure 164: 3D surfaces toolbar

Settings: Open the Optic3D Surfaces Settings (Depending on chosen object.)

9.2.10 Analog In Toolbar

Figure 165: Analog In toolbar

The Analog In toolbar is available only if USC-2/-3 is used. It can be configured for the two 10 bit analog inputs (values from 0-1023) of USC-2/-3 card. The input signals must be connected to Ana_In_Ch_0 or Ana_In_Ch_1 in respect to Ana_In_GND. The current bit value of the analog input bit can be checked at sc_usc_server.exe in visible mode at InfoView.

The input voltage should not exceed 10.3 V.

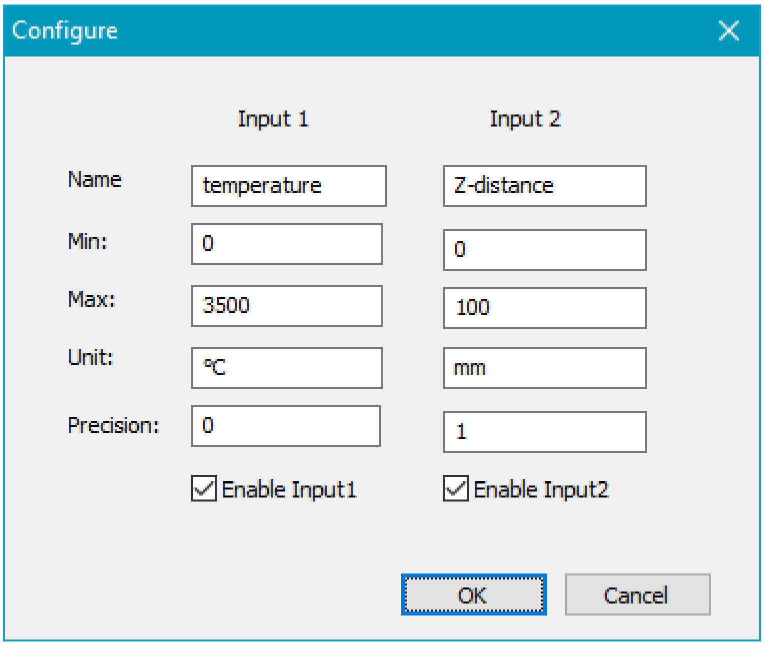


Figure 166: Analog In dialog

Input 1: Settings for analog input ANA_CH_0.

Input 2: Settings for analog input ANA_CH_1.

Name: Here a name can be set for the Analog Input bit that can be used to decide what its purpose.

Min: Here the minimum of the unit range have to be set for the Analog Input bit, which should corresponds to a digital value of 0 or 0 V.

Max: Here the maximum of the unit range have to be set for the Analog Input bit, which should corresponds to a digital value of 1023 or about 10.3 V.

Unit: Here the physical unit of the Analog Input bit can be set.

Precision: Here the number of decimal places can be set. Please keep in mind that the displayed numerical resolution is limited to (Max value - Min value) / 1024.

Enable Input1: Activates the display of analog input ANA_CH_0.

Enable Input2: Activates the display of analog input ANA_CH_1.

If activated the Analog In toolbar displays the converted voltage according to settings of Analog In dialog in accordance with the set names and units there.

9.2.11 Marking on the Fly

Figure 167: Marking on the Fly Toolbar

The Marking on the Fly toolbar has to be configured once before it can be displayed. The configuration can ma done via Global Settings → View → Toolbars → Configuration (Next to Motf). After that the following dialog opens:

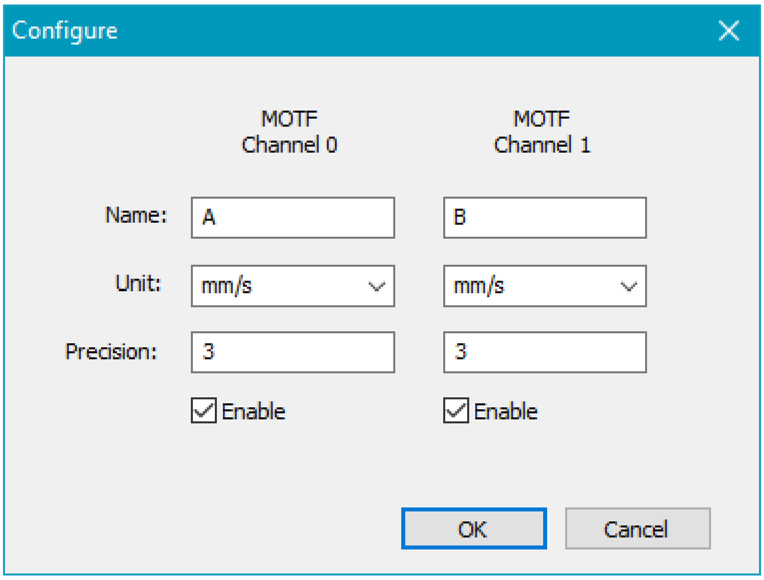


Figure 168: Motf Toolbar Configuration

Both channels can be enabled or disabled. If only one channel gets enabled the toolbar will be smaller in size. The user can apply the corresponding channel name, set the displayed movement unit and choose the amount of the precision.

Name: Enter the Channel name.

Unit: Choose between mm/s and m/min

Precision: Enter the number of decimals.

Enable: Enable or disable the Channels.

9.2.12 Flash Compatible Mode

Figure 169: Flash Compatible Mode Toolbar

The flash compatible mode (FCM) is meant to:
- help creating jobs for the flash
- check the compatibility of existing jobs with flash
- check the compatibility of existing settings.

: Not active.
: Not OK.
: OK.

Please find further information in Chapter Flash compatible mode in SAMLight.

9.2.13 Background Camera

This chapters shows how to integrate the picture of an off-axis Background Camera to allow a comfortable job handling.

The tool is able to compensate optical distortions/rotations.

Windows Camera Access

To be able to use the Background Camera of SAMLight, it is necessary to allow the camera access in windows.

Example for Windows 10:

Please open Windows → Settings → Privacy → App Permission → Camera:

Set "Camera access for this device is on" and set "Allow apps to access your camera" to on.

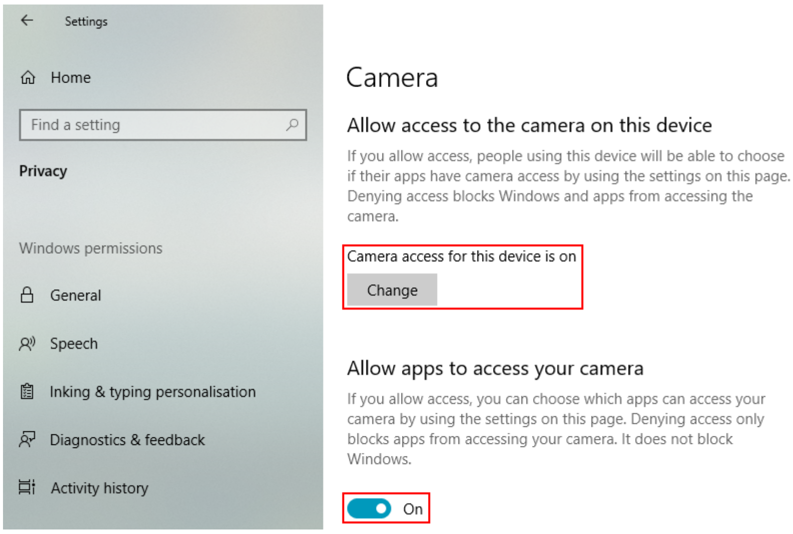


Figure 170: Camera access in Windows 10

System requirements:

For using the Backgound Camera in SAMLight a decent GPU and CPU are required.

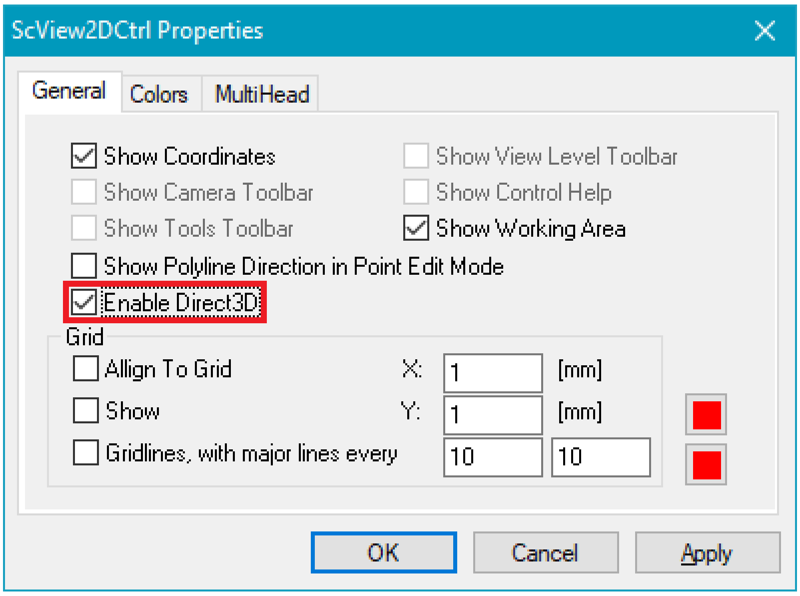
Camera devices are supported through the UVC class driver and must be compatible with UVC 1.1.

Besides this, SamLight can also access cameras that provide a directshow capture device. Network cameras that do not provide this can be accessed via the virtual webcam feature of the Open Broadcaster Studio (https://obsproject.com/). For example, if your camera provides a rtsp stream: install and start the OBS program and then select "create new video vlc video source" to access your camera, then click on "start virtual camera". In SamLight you then select the "OBS Virtual Camera" as your camera device.

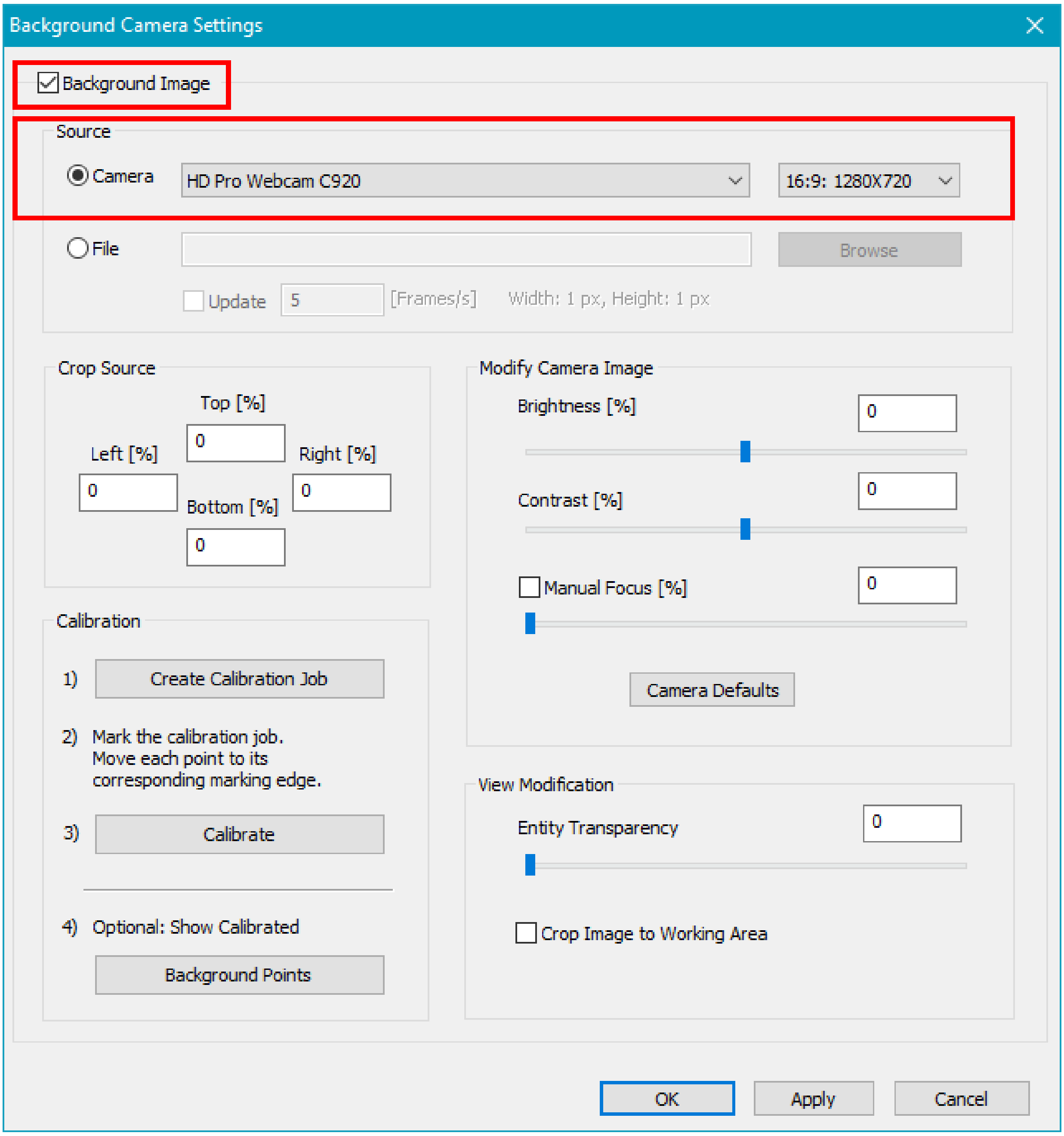
If the OBS internal virtual camera causes issues, the 'OBS Virtualcam' plugin can be used. Start VirtualCam in OBS and select 'directshow OBS-Camera' from the background camera dialog.

Activate the background camera in SAMLight:

- To use the Background Camera tool, it is necessary to activate Direct3D. Activate Direct3D by right clicking the View2D to get the View Properties:



- To activate the camera itself, activate the checkbox in the Background Camera toolbar and open the "Background Camera Settings" dialog by clicking the camera symbol :
  (If the background camera toolbar is not visible, please activate the toolbar in Settings - System - View - Toolbars.)



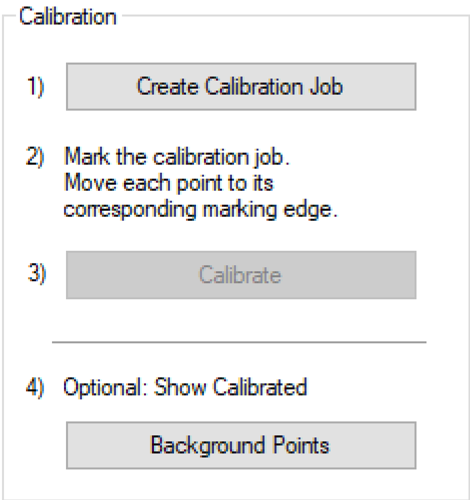
- The radio button „Camera“ activates the livestream of the camera

- The radio button “File” activates the “old” mode where you can select a saved picture from the hard disk for the background.
  It is important to choose an aspect ratio (e.g. 4:3) and a resolution before doing the calibration. When changing aspect ratio or resolution, the calibration process needs to be repeated. With the resolution, the

frame rate of the camera image changes. If you need „Crop Source“, please also choose this parameter before doing the calibration.

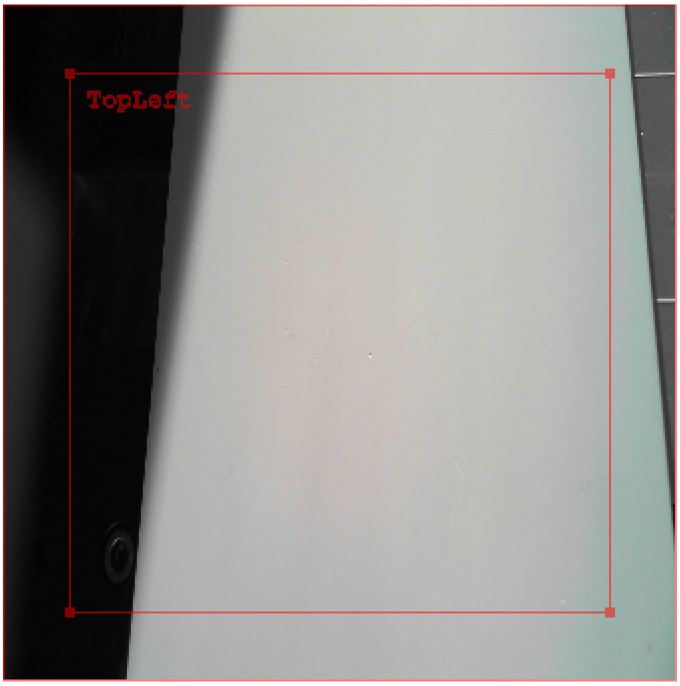
Calibration:

- The calibration of the Background Camera is done with the buttons under „Calibration“:



1) “Create Calibration Job”:

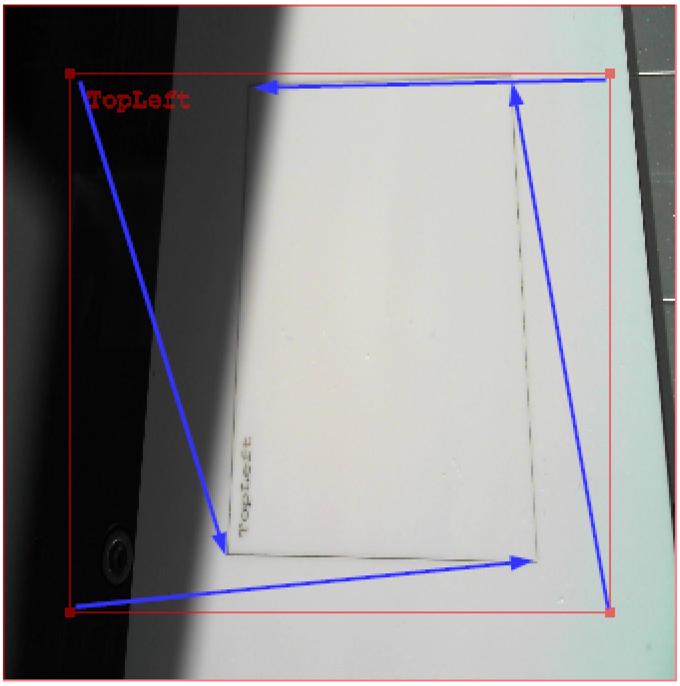
This creates a job in SAMLight which is for easy calibration of the camera image:

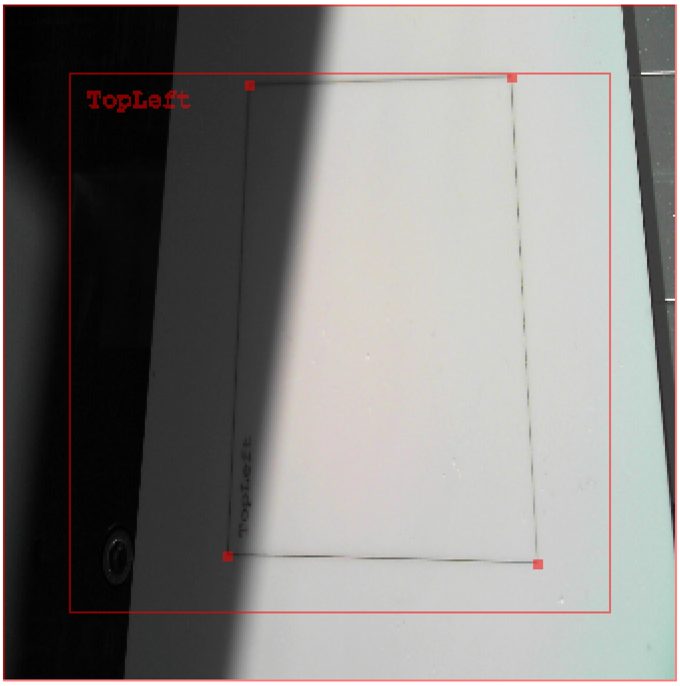


2) Marking of the calibration job:
The next step is to mark the job which was created before. To mark the job, please leave the current dialog and use the mark dialog.

Please make sure to use an appropriate marking speed to get an optimal marking result.
It is important not to move the work piece after having done the marking.

3) Calibrate:
Move the four points in the job to the edges of the marking result to correct the camera image.
It is important to mind the orientation (TopLeft) of the camera image.
By this procedure, it is possible to correct distortions such as tilts, expansion or rotation.
In the following picture, the marking result is rotated about 90° in respect to the camera image.





To guarantee the optimal calibration, use the zoom function of SAMLight to place the points carefully.
After having placed the points, calibration can be done.

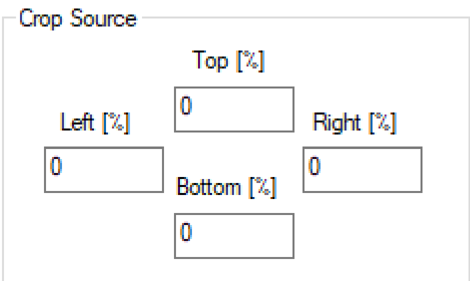
Press the button „Calibrate“ which will correct the camera image. Then, the marking result and the test job should superimpose:

In the picture below, the „TopLeft“ as well as the shadow in the background are now rotated about 90 degrees.

4) Show Calibrated Background Points:
This dialog shows the Calibrated Background Points.

Crop Source:

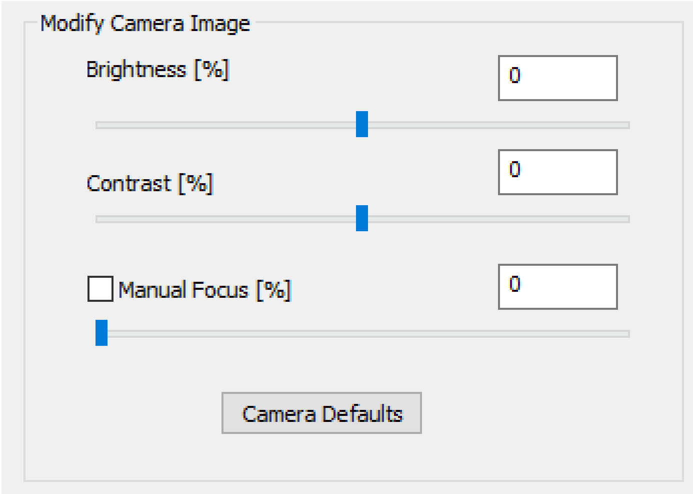
Here, you can modify the image section.
It is necessary to do this selection before doing the calibration.



Modify Camera Image:

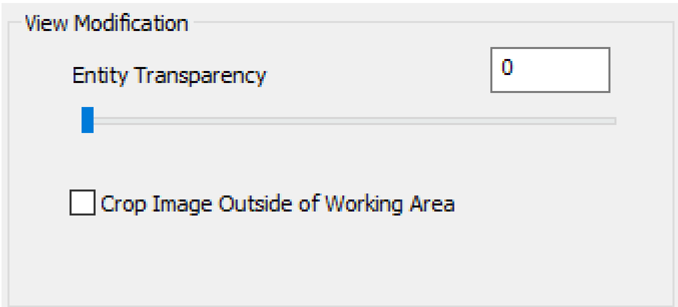
In this area, brightness and contrast can be changed, if default camera values are not satisfying.

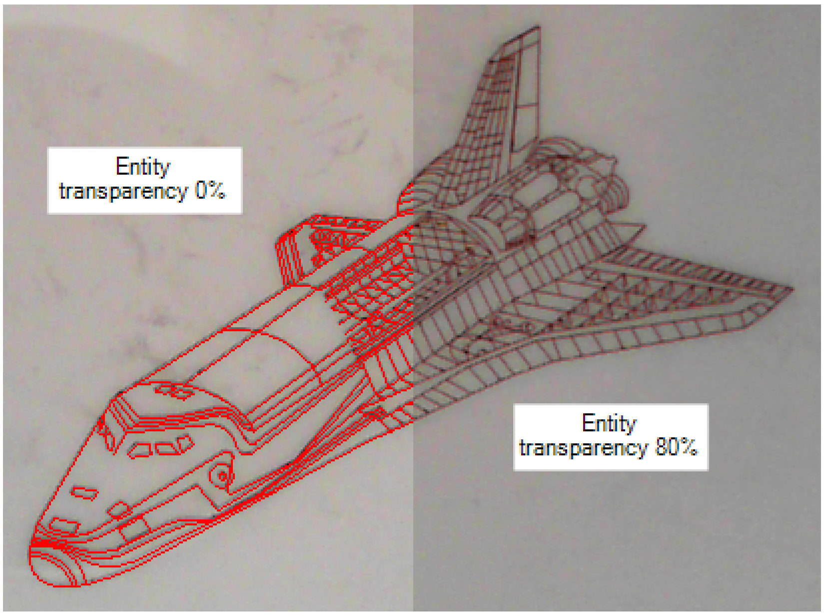
The automatic focus of the camera could cause disturbing behavior during operation. By activating the manual focus its possible to set the focus of the camera to a fixed value.



Entity View Modifikation:

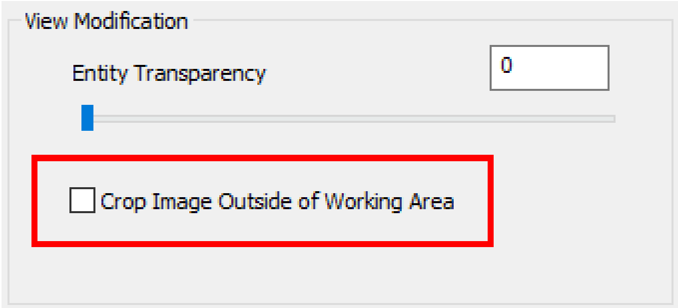
If the calibration was successful, the entity in the View2D superimposes the marking result.
To increase the visibility of the marking result, the transparency of the entity can be adapted in SAMLight:





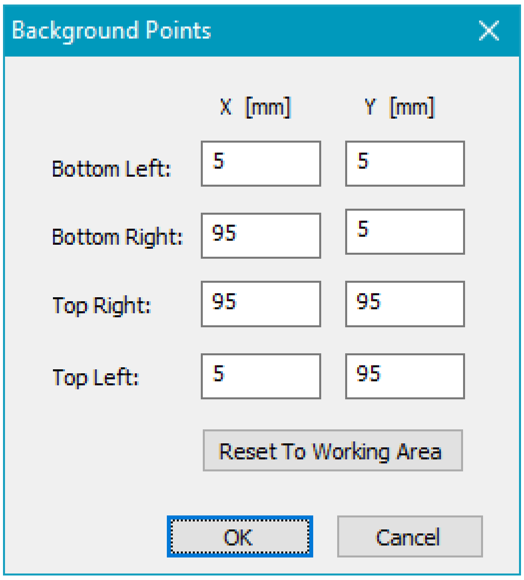
Crop Image Outside of the Working Area:

If the checkbox Crop "Image Outside of Working Area" is activated the View2D will only show the part of the camera image which is inside the working area.



9.2.13.1 Calibrated Background Points

This point is optional. The button opens a new dialog which shows the coordinates of the corner points.



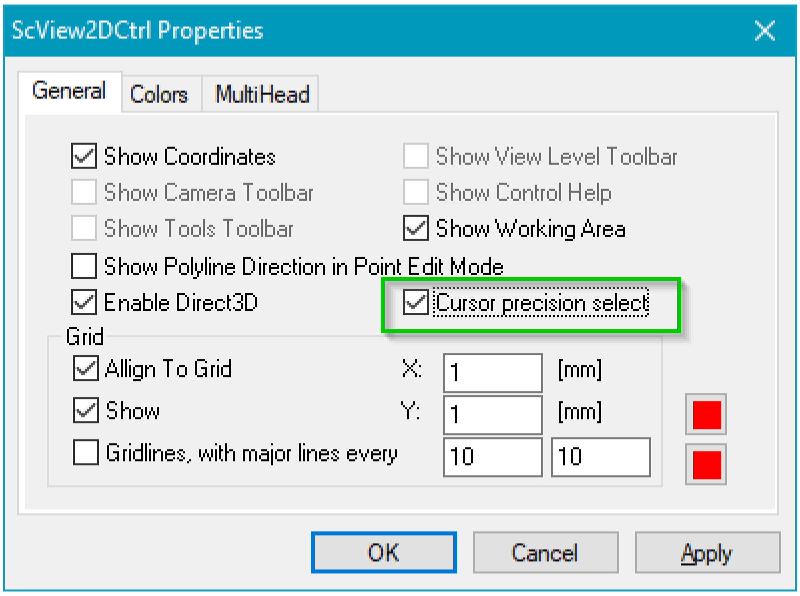
9.2.13.2 Mouse Cursor Settings

In some cases, the desired background camera image/stream might be quite dark, hence generating an insufficient contrast to the black cursor, included in SAMLight.

In order to solve this issue, please follow these steps:

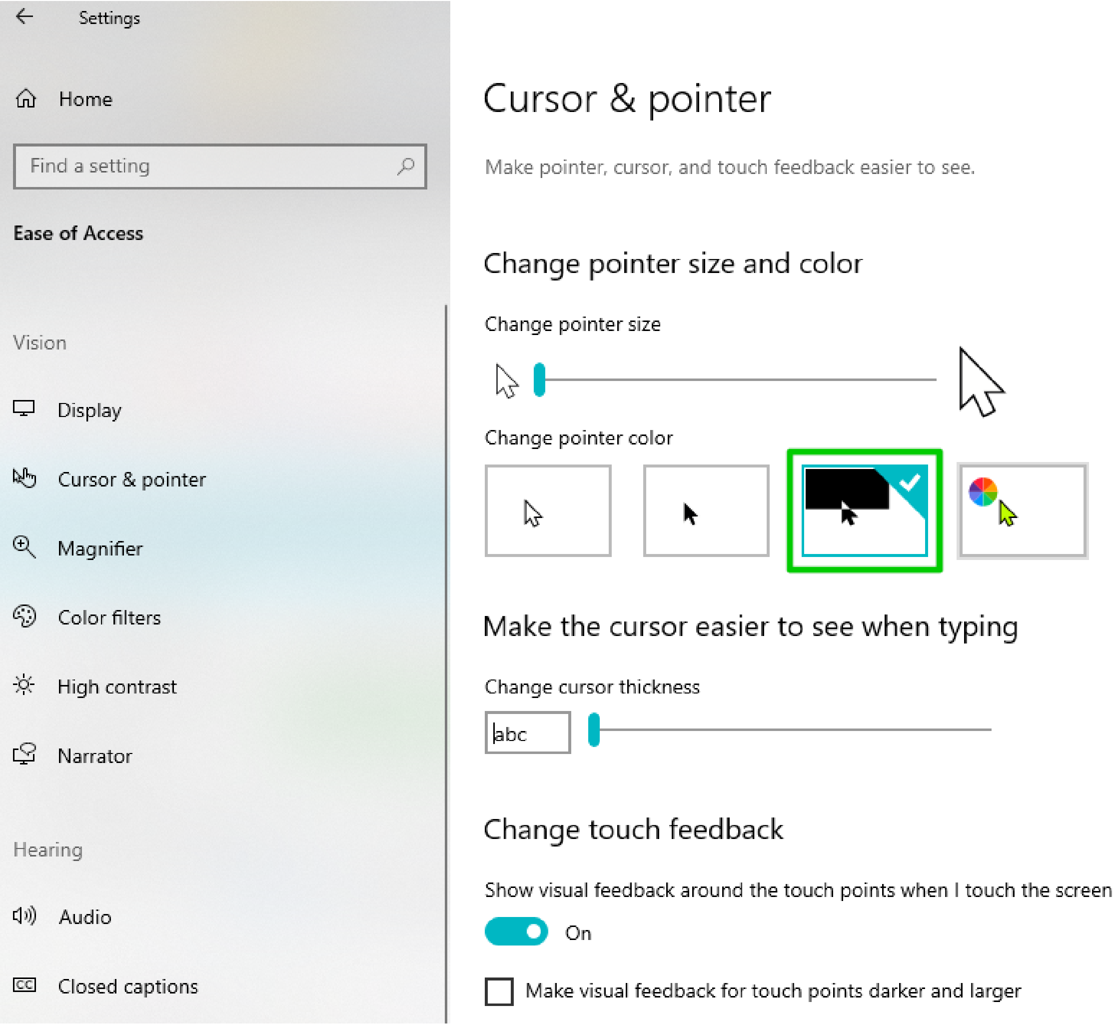
Adjustment within SAMLight:

1. Right click on the (2D-) working area in SAMLight;
2. Pick "View Properties" on the appearing drop-down menu;
3. Activate the "Cursor precision select" radio-button in the "General"-tab of the displayed configuration window (as shown in the picture below);



Adjustment of the Windows Settings:

1. Open the Settings-menu via the Windows-button (alternatively, open the "Control Panel");
2. Go to the "Ease of Access";
3. Click on "Cursor & pointer" on the left-hand side menu;
4. Select the "Inverted mode"-option at "Change pointer color", as marked in the picture below;



This option activates an inverted mode, that dynamically changes the mouse cursor color based on the background color. Now, SAMLight is able to access this feature from Windows.

9.2.14 Special Menu

Figure 171: Special Menu Toolbar

This toolbar corresponds to the Script Settings in Global Settings → Remote. The defined script or executable can here directly be executed. The number in the toolbar corresponds to the position in the script settings.

9.2.15 Jobs Toolbar

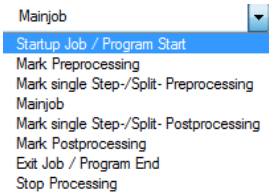


Figure 172: Special Sequences Toolbar

The Special Sequences Toolbar can be made visible in *Settings → System → View → Toolbars → Jobs Toolbar*. This Toolbar allows you to switch between different jobs that have a specific purpose and are executed at a specific moment in program flow. When such a special job beside the main job is selected within the toolbar a dialog opens where the elements of that sequence can be edited. These special sequences are not available in flash or trigger-mode.

Special Sequences: This part of the settings panel handles the special sequences that can be executed during program startup and exit. To avoid collisions between an job externally selected this option is available only in I/O-Mode *Standard*. By default, these special jobs and together with them the Special Jobs Toolbar are disabled.

No Pre-/Postprocessing Warning: If this box is selected, this special security warning is disabled and all special jobs (except the mainjob) are executed immediately and with no separate user interaction when the program is started or exited. Please handle with care! If this option is used it has to be secured by the user that nobody can be injured by potentially dangerous pre- or postprocessing jobs.

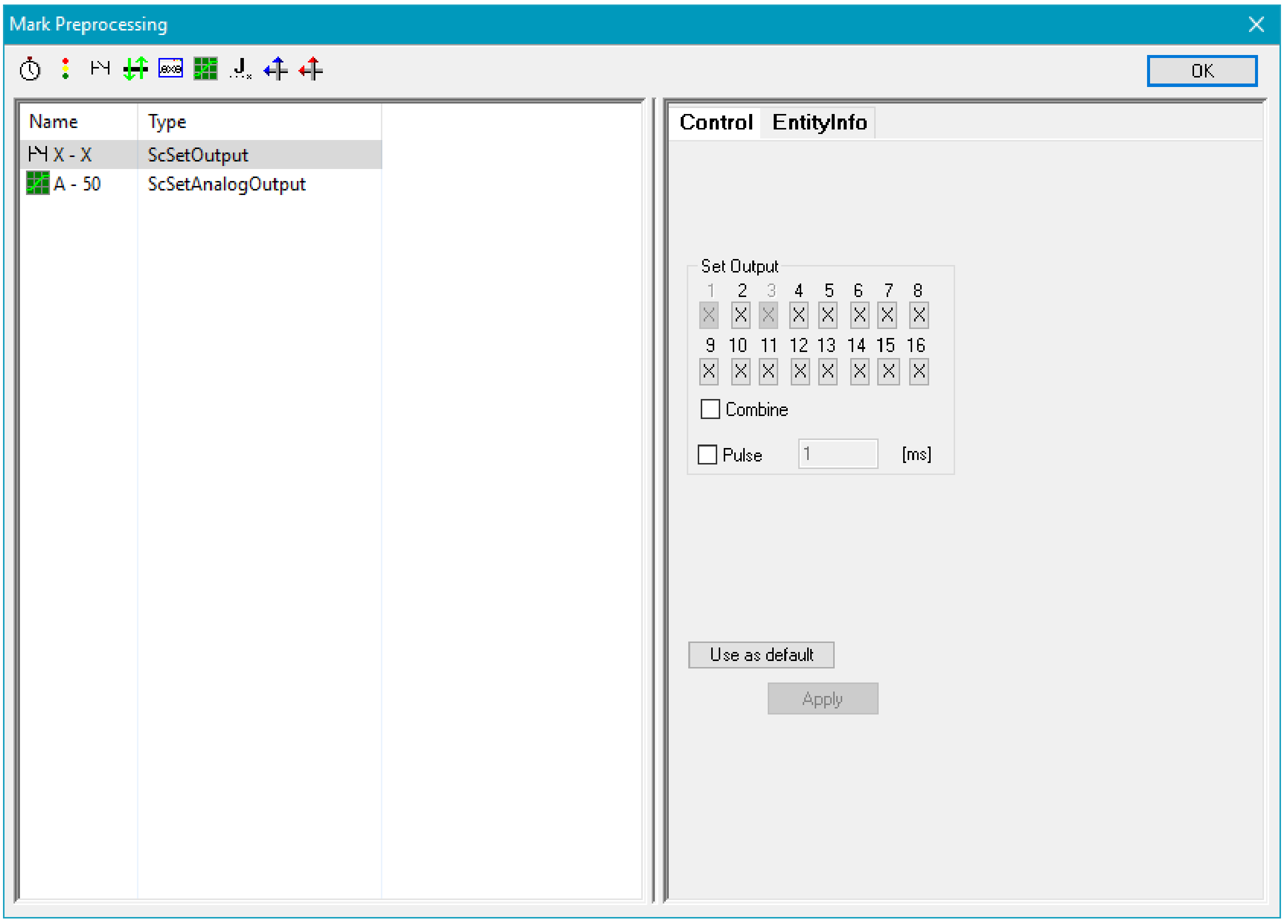


Figure 173: Mark Preprocessing Dialog

Here the [toolbar](#) elements that are known out of the main window can be used. Their [settings and parameters](#) can be edited in a similar way. The following special sequences (special jobs) can be chosen:

Startup Job / Program Start: If a job is defined here it is executed directly after program startup to e.g. initialize special external equipment. A marking operation performed by this preprocessing job is not ranked as a normal operation like the mainjob and therefore not counted as one quantity. Due to security reasons it is recommended to avoid potentially dangerous operations like laser marking operations or heavy movements within this job.

Mainjob: This job is the default job. The items defined here are processed during a marking operation and are counted as one quantity each. This job is the same like the only one that is executed in case the special job functionality is turned off.

Exit Job / Program End: This job is the counterpart of the Startup Job. It is executed when the program is shutting down. Such a Postprocessing Job can be used e.g. to deinitialize external equipment. If a marking operation is performed here it is recommended to avoid potentially dangerous operations within this job.

Mark Preprocessing: This Job is a specific one that is executed directly before the main marking job is executed. If a Splitting or Step/Repeat operation is performed, this job is executed once before the full operation starts. In SAM3D the mark preprocess will only be performed before the first slice of the job. If you start the build with another slice but the first one no mark preprocessing job will be executed.

Mark Postprocessing: If the marking of the main job is finished or when the user has pressed the stop-button during marking then the job that is defined here is executed.

Here no dangerous operations like additional marking operations should be executed. There would be the high risk that if somebody presses stop but instead of stopping an other marking process is started. This special sequence should be used only for deinitialization operations that are necessary after marking, e.g. to set some outputs to defined values.

Mark Pre-/Postprocessing is available for ParameterFinder.

Mark single Step- / Split- Preprocessing: If Splitting or Step/Repeat is used a marking in progress sequence can be defined which will be done e.g. after an axis movement has finished. This sequence will be executed right before the laser starts to mark the next mark job.

Mark single Step- / Split- Postprocessing: If Splitting or Step/Repeat is used this sequence will be executed right after the making procedure has finished and before an axis movement starts.

Slice Preprocessing: This Job is available in 3D-mode only and is executed directly before a single slice of a 3D object is marked. Here several control elements can be added e.g. to move a Z-table that modifies the vertical position or to set specific output pins that perform that task.

Slice Postprocessing: This Job is available in 3D-mode only and is executed directly after a single slice of a 3D object was marked. Here several control elements can be added e.g. to move a Z-table that modifies the vertical position or to wait for specific input pins.

Stop Processing: This task will be performed in case of external stop (via OPTO_IN_1) is recognized if the stop button in the mark dialog is clicked or if the control command client ScStopMarking () is recognized. Here, several control elements can be added for example to move a Z-table that changes the vertical position or to wait for dedicated input pins.

Using the toolbar shown above it is possible to switch between these jobs and then to perform all normal operations for the actually selected job. Because these special jobs are no common operation they are disabled by default so that only the mainjob is visible and useable. The pre- and postprocessing jobs can be enabled using the Special Sequences Settings.

The Special Sequence settings can be saved in each Job file with the checkbox "Save pre/ post jobs" in the Job Properties page.

9.3 Entity List

The Entity List appears in two different modes:

- Entity List
- Point Editor

The *Entity List* shows all the entities in the current job and visualizes their logical structure / hierarchy in a ListView. It provides operations for moving, copying and sorting entities as well as exploring entities that contain sub-entities. The *Point Editor* is a tool to visualize and modify the point description of an entity.

9.3.1 Entity List

The Entity List shows all the entities in the current job and visualizes their logical structure/hierarchy in a ListView. It provides operations for moving, copying and sorting entities as well as exploring entities that contain sub-entities.

Definition entity: An entity is either an element or a container or a group of entities. An element keeps real geometric data like lines, points and pixels and a so-called container contains elements.

Example: For example a simple description of a desk: The desk is a group of the sub-entities *legs* and *desk_top*, where the *desk_top* is a container that contains the geometry of the desk top. The group *legs* contains the sub-entities *leg1*,...,*leg4* which are containers that contain the geometry of a leg.

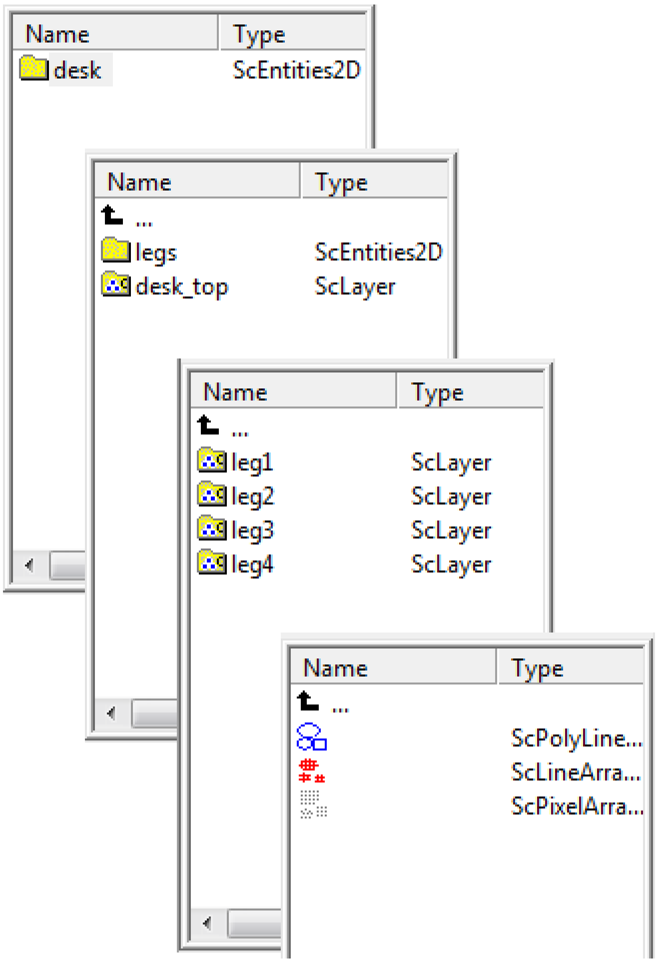


Figure 174: Entity List

The last picture shows the elements of the container *leg1*.

The levels of the Entity List: To understand the levels think again of the desk example above. The entity *desk* is on level 1 and its contents are on level 2. The entity *legs* is a sub-entity of *desk* (level 2), its contents are on level 3 and so on. For a detailed description of the object hierarchy see chapter Object Hierarchy.

Operations:

Exploring: By double clicking an entity which contains other entities (for example a Group) the ListView will step inside and show the entities inside the selected one.

Move: This is the standard operation of a drag and drop process: It moves the selected entity or entities. Here following possibilities are available:

- Move an entity within the job to change its position
- Move a grouped entity out of the group by dragging it onto the level up arrow
- Move an entity into an existing group by dragging it onto the target ScEntities2D while the *Shift key* is hold down
- Move an entity/ entities in an group by options to previous, to next, to top and to bottom.

Copy: This works in the same way as Move, pressing the CONTROL-key additionally during the drag and drop process puts a copy of the selected entity in the drop folder.

Remark to Move/Copy: Not every move or copy operation is allowed. So, for example the move of a line array into a PolyLines group or the move of a element inside a container out to an other container can not be executed since this would corrupt the data consistency.

Group: Several entities of the same level can be grouped by clicking on them while holding the *Shift key* down and choosing *Group* from menu *Edit → Group*. This procedure will create a new group on the actual level that gets the selected entities as sub-entities.

The following operations are provided by pressing the right mouse button. Here the entity list context menu appears that offers the following functionalities:

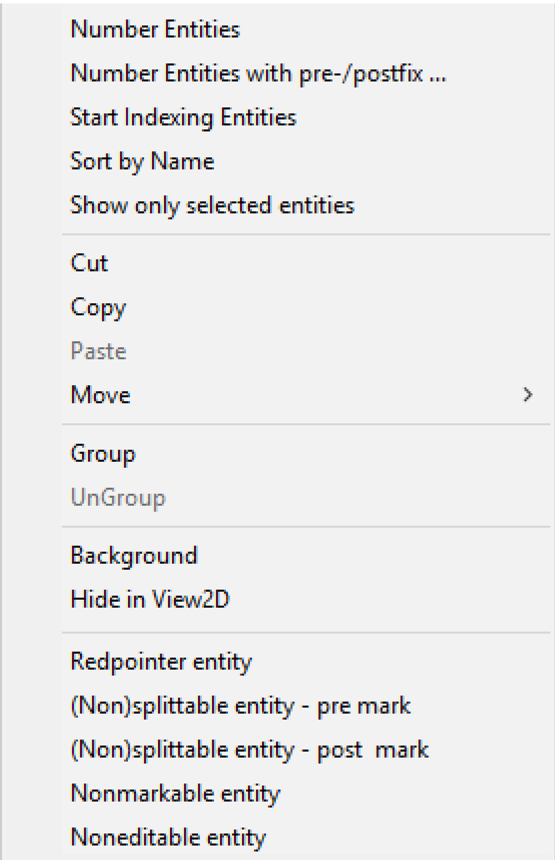


Figure 175: Entity List Context Menu

Number Entities: All Entities, regardless if they are selected or not, are being named with a number starting from 1.

Number Entities with pre-/postfix: Opens the corresponding dialog. All Entities are being named starting from a freely selectable number with a freely selectable incremental value and / or a defined pre or postfix.

Start / Stop Indexing Entities: Opens the corresponding dialog If selecting this the first time all Entity Names are being deleted. Now the user can choose between indexing options. If this is done the user may index any entity by clicking on it with the left mouse button. By default the indexing starts with 1 and is incremented by 1 after each mouse click. If the user wishes to return to normal mode again, he selects *Stop Indexing Entities*.

Sort By Name: Selecting this function sorts the entities on the actual level by their name. The name comparison is case sensitive and starts at the first letter of the name. Entities with the names *1*, *2*, *12* will be sorted to *1*, *12*, *2*, because the first letter of entity *12* is 1 and of entity *2* is 2. So the entity *2* gets the position after entity *12*. Name the entities with leading zeros to solve that issue.

Remark to changing the order of the entities: The entities of a group will always be marked from the top first to the bottom last in the ListView. So changing the order in the ListView can be used to change the order of exposure during the mark process.

Cut: The currently selected entities are cut out of the list and they are put into the internal entity clipboard for use in further operations.

Copy: Comparing to *Cut* using this operation the selected entities aren't removed from the entities list but a copy of them is put into the internal entity clipboard too.

Paste: If there are some entities held in the internal clipboard they can be copied into the entity list using the current position.

NOTE

For this operations the same restrictions are valid than described for the Drag-and-drop-copy operation described above.

Move: Moves the position of the selected entity or entities in the entity list. The sequence in the entity list determines the sequence of marking.

Group: Groups the selected entities and puts them into an entities group. See also chapter Object Hierarchy.

UnGroup: Ungroups the selected entities group. The view level of all entities inside the group will be decreased by one. See also chapter Object Hierarchy.

Back-/Foreground: This option can be used to use a selected bitmap entity as background object or to use a background object as normal entity. When a bitmap-object is put to the background, it isn't selectable in the View 2D any longer but only within the object list. Additionally it won't be marked. Such images can be used as a template for creating new objects, here e.g. a technical drawing can be used. After importing a bitmap to the current job it is possible to scale and position it so that the resulting dimensions are correct and that it fits to the desired working area. After that was done it can be put to the Background so that it doesn't influence the current job any longer but can be used as template. If that background drawing isn't required any longer, it can be selected within the Entity List and then put to the Foreground using this menu item again. Now it is possible to remove it from the job. In the list entities that are used as background are marked by a white folder symbol instead of a yellow one.

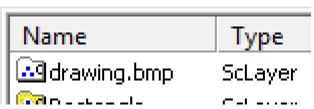


Figure 176: Hidden Entity in View2D

Hide-/Show in View2D: Compared to the previously described background option this one works different. When an entity is set to mode "Hide in View2D" it is no longer visible in View 2D but it still can be marked. It can be selected within the Entity List and it can be moved, scaled and translated when it is selected. This function can be used to hide elements in a job that overlap each other. That is useful in cases where some of these overlapping objects can't be seen because the other ones cover them completely. Using that *Hide* option some of the covering entities can be made invisible without removing them from the marking result. In the entity list entities that are hidden from the View2D are indicated by a dimmed folder symbol instead of a solid yellow one.

Red pointer Entity: When an entity is marked as Red pointer Entity this entity will only show when using the Red Pointer if the checkbox "Red pointer Entity" is activated in the Mark Dialog.

(Non)splittable Entity: When an entity is marked as non-splittable its appearance within the View2D doesn't change and it is highlighted by red brackets within the Entity List. An entity that is marked in this way will behave different for split jobs: Independent from their position within the original job they will be marked before the splits are processed (if set to pre mark) or after the splits are processed (if set to post mark). If an entity state is changed to splittable or non-splittable when the splitting mode is active, that job has to be re-split. That way, this functionality can be used to exclude some parts of the job from being split. These excluded objects will be marked as one piece when a split job is processed. If the splitting mode isn't active this selection doesn't influence the job.

(Non)markable Entity: This can be used to exclude the selected entities from the marking process. By selecting again the selected entities are being included again.

(Non)editable Entity: This can be used to define an entity as non editable. But the entity is still included in the marking process.

9.3.1.1 Index Entities with pro-/postfix Dialogs

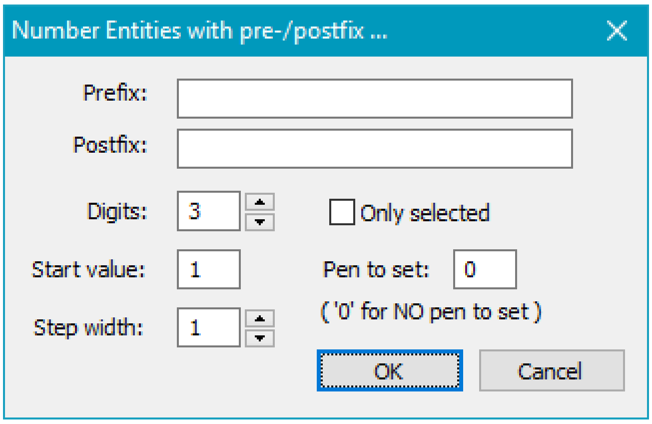


Figure 177: Number Entities with pre-/postfix Dialog

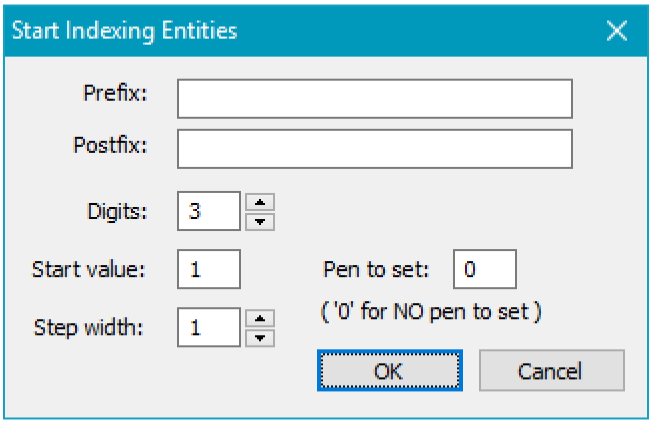


Figure 178: Start Indexing Entities Dialog

9.3.2 Point Editor

Purpose: The Point Editor is a tool that visualizes and modifies the point description of an entity.
Point Editor View of the Entity List: Before doing a point editing operation please switch to the point editing mode (use the Object Toolbar, button). Then select one or more objects in the same way as described in View 2D → Selection. If more than one object is selected the objects must be grouped (use *Edit → Group*) to perform point editing. One can also perform these two steps the other way around.

After selecting objects in point editing mode the Entity List will switch to the Point Editor View as shown in the screenshot below.

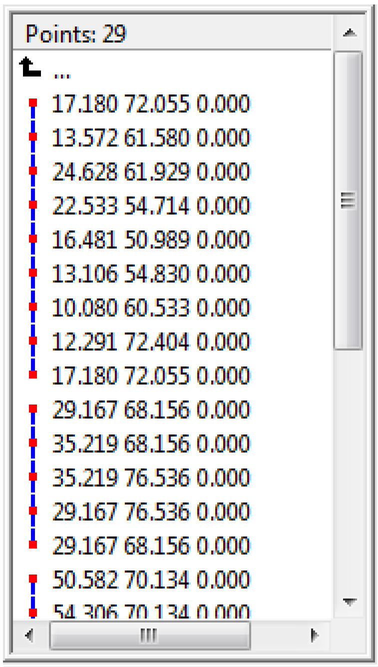


Figure 179: Point Editor View

The Point Editor View shows all points of the selected objects in a single list.

All the points which belong to one PolyLine are connected by a blue line in that list.

The red dots mark single points and the 2D/3D coordinates of a point are displayed next to the red dot. The coordinates are 3D only with Optic3D.

The caption of the Point Editor View displays the number of points.

Context Menu: To modify the points in the list or the list itself the context menu provides five operations. They are described next to the screenshot below. Select one or more items from the list by mouse click while keeping the *Ctrl* pressed. After the selection keep the *Ctrl* pressed and click the right mouse button. Then the context menu will be displayed. The points of the selected objects are marked in the View 2D by black dots. Clicking with the right mouse button on such a black dot will display the context menu too.

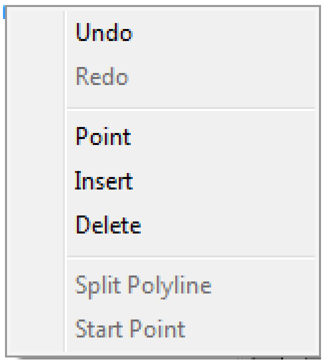


Figure 180: Context Menu

The context menu provides:

Undo: Undo the last operation.

Redo: Redo the last Undo.

Point: Opens the Edit Items dialog like it is shown below.

Insert: Inserts a new point in between the two adjacent points.

Delete: Deletes the selected points.

Point Editing with the Edit Items dialog: To edit one or more points use the Edit Items dialog shown below. One can get this dialog via the context menu (described above) or by double clicking on a point item in the list.

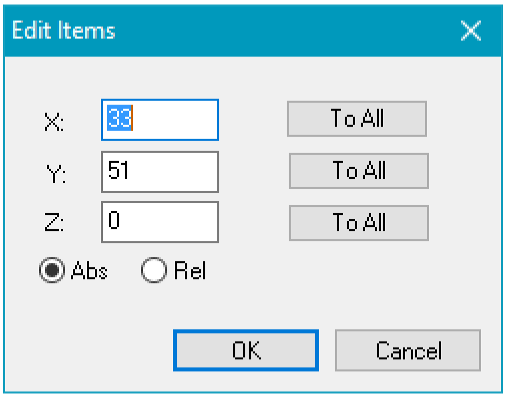


Figure 181: Edit Point Coordinates Dialog

X,Y,Z: Edit these values to change the coordinates of a single point. Remark: Z coordinate is only available with Optic3D.

NOTE *The Z-Coordinate is only available with the Optic3D Option.*

Radio buttons: There are two update modes for updating point coordinates which can be switched by the radio buttons. Abs is the default mode.

Abs mode: The point coordinates are updated by a substitution of the old coordinates by the new ones.

Rel mode: The point coordinates are updated by adding the new values to the actual coordinates.

To All: These buttons are only active if more than one point is selected. To update the X coordinates of all selected points change the value of the X field and click the *To All* button next to it. Updating of the Y or Z coordinate for all points is done analogously.

Examples: To project all selected points on the plane Z = 1 select update mode Abs, change the value in the Z field to 1 and click the ToAll - Button next to the Z field. To move all selected points in Y direction by 3 units select update mode Rel, change the value in the Y field to 3 and click the ToAll - Button next to Y input field.

9.4 View 2D

Purpose: The View2D displays the geometry of all the entities in the current job. It provides basic operations for the manipulation of objects.

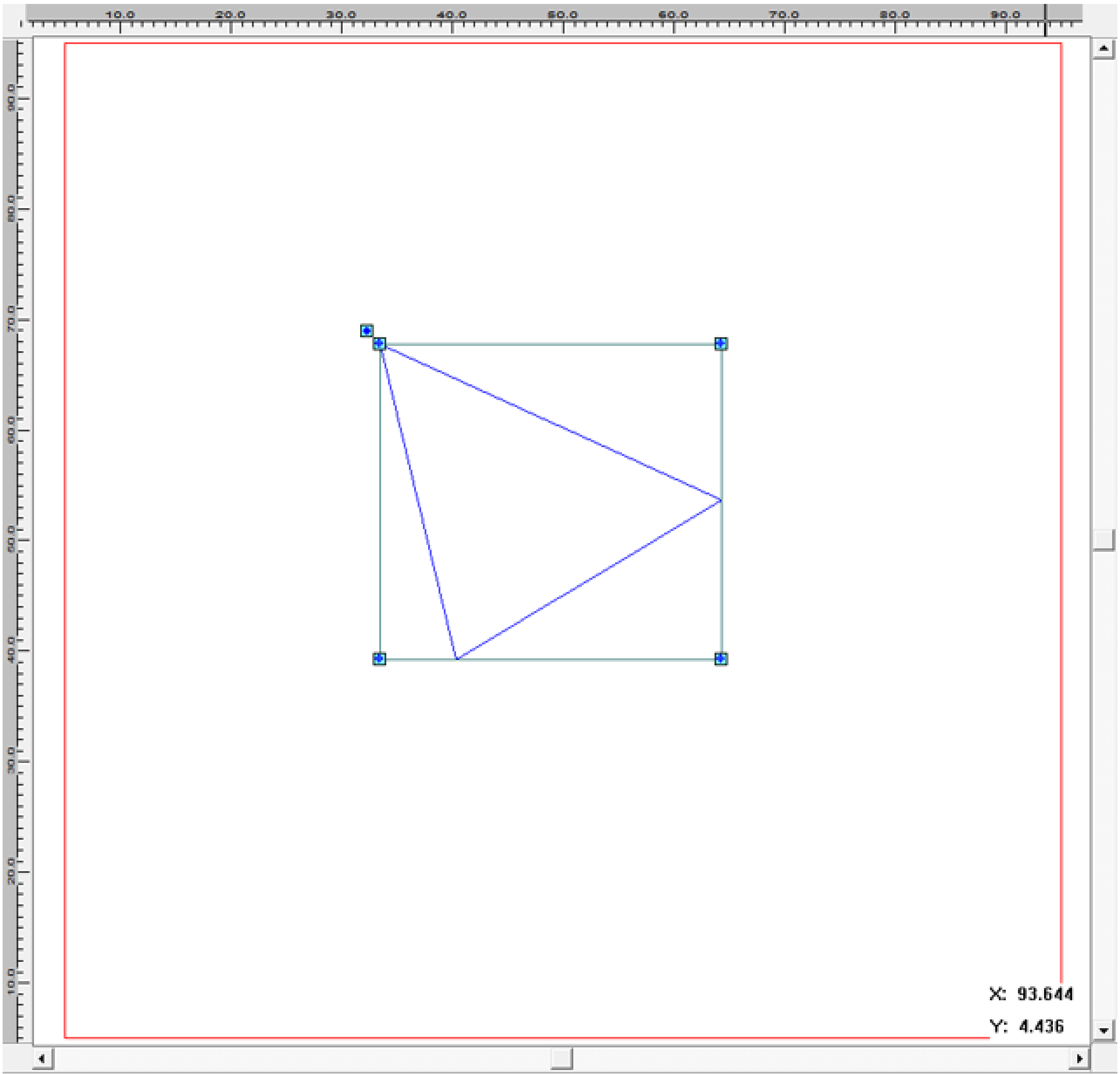


Figure 182: View 2D

Overview: The picture above is a screenshot of the View2D. The red bounding box marks the boundary of the workspace/working area. The values in the lower right corner are the coordinates of the mouse pointer (if the mouse pointer is inside the View2D). The axes of the coordinate system are marked on the left and the upper boundary of the View2D and its origin is in the lower left corner of the workspace/working area. For more details see chapter Operations and Print Preview.

9.4.1 Operations

Creation of objects: To create objects in the View2D use the Object Toolbar.

Selection: Before doing a selection operation switch to the select mode (use the Object Toolbar, button), unless you want to do a point editing operation (see Entity List → Point Editor).

First way: Selection via the Entity List: Select an entity, i.e. an object or a group of objects, by clicking on it in the Entity List. This will show a modify box for the selection, like it is shown for the triangle in the screenshot. To select more than one entity hold down Ctrl and select the entities by clicking on them in the Entity List.

Second way: Selection via the View2D by mouse interaction: To select one or more objects draw a bounding box that covers all objects to select. To draw this bounding box click the left mouse button at the chosen position of the left upper corner and move the mouse while keeping the left mouse button pressed to the chosen position of the right lower corner of the bounding box. Then release the left mouse button. This will show a modify box that contains all the objects in the bounding box.

Third way: Selection via the View2D by mouse clicks: To select only one object click on it in the View2D. To select more than one object, hold down Ctrl and select the entities by clicking on them it in the View2D.

Manipulation of objects: Before an object can be modified it must have been selected. To do this follow the instructions described above (*Selection*). The modify box, shown after the selection, provides three operation modes for transformations which can be switched through by the button.

Translation mode: In this mode the corners of the modify box show the symbol . To translate the selected object click on this symbol, keep the left mouse button pressed and drag the mouse to the new position.

Scaling mode: In this mode the corners of the modify box show the symbol and the edges show the symbol . To scale the object in x and y direction simultaneously click on a corner symbol, keep the left mouse button pressed and drag the mouse. To scale only in x or y direction click on an edge symbol and do the same.

Rotating and slanting mode: In this mode the corners of the modify box show the symbol and the edges show the symbol . To rotate the object around the rotation centre click on a corner symbol, keep the left mouse button pressed and drag the mouse. To slant the object click on an edge symbol and do the same.

The manipulations described above are not object specific. To modify object specific properties use the Entity Property Sheet.

View2D Settings: To modify the settings of the View2D use the dialog *Menu bar → Settings → View*.

Context Menu: When right click somewhere in the View2D the context menu appears:

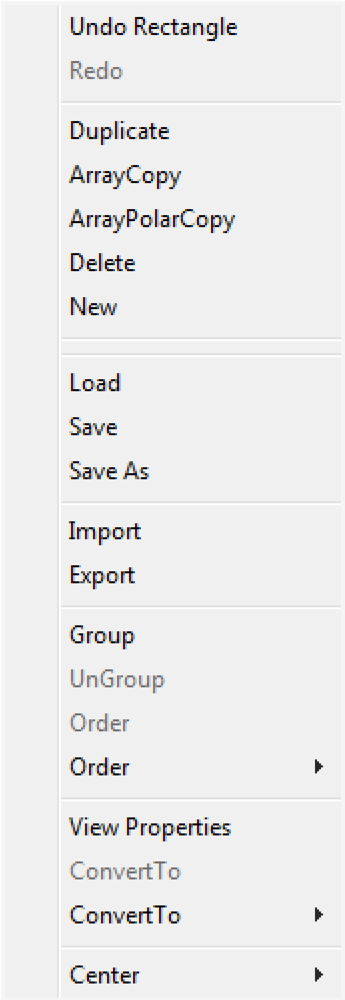


Figure 183: Context Menu

Duplicate: Make a copy of the object

ArrayCopy: See in ArrayCopy chapter.

ArrayPolarCopy: See in ArrayPolarCopy chapter.

Import / Export: Same function as in Menu Bar

Flip: Reverse marking order for Polylines.

9.4.1.1 View Properties

When right click somewhere in the View2D the context menu appears with the entry View Properties. In the following, the four tabs dialog are explained:

- General
- Colors
- MultiHead
- Colors > 20

9.4.1.1.1 General

With these check boxes in the area above, the user can decide which information should be displayed.

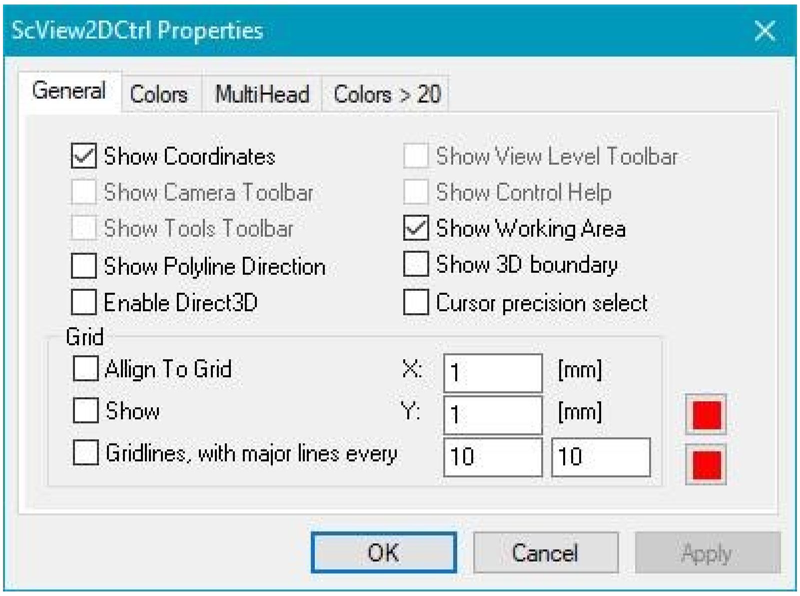


Figure 184: ScView2DCtrl_General

Show Coordinates: Show the coordinates of the mouse cursor in the lower right corner of the View2D.

Show Camera Toolbar: Grayed out since this is edited at *User Interface* → *Toolbars*.

Show Tools Toolbar: Grayed out since this is edited at *User Interface* → *Toolbars*.

Show Polyline Direction: Show the direction of poly lines in the point edit mode.

Enable Direct3D: Change View2D drawing and accelerate the process on the graphic card. Has to be enabled for background camera.

Show View Level Toolbar: Grayed out since this is edited at *User Interface* → *Toolbars*.

Show Control Help: Can not be used.

Show Working Area: Show the working area inside the View2D.

Show 3D boundary: Show the border of the execution if z-correction is enabled.

Cursor precision select: Change the optic of the mouse cursor and use the windows setting from „recision select".

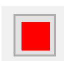
Grid:

Align To Grid: If checked each new object placed in the View 2D will be aligned to the grid.

Show: If checked the grid will be displayed in the View 2D.

X, Y: These two values define the grid size.

Gridlines, with major lines every: Choose distance in x- and y-direction to display grid lines in the View2D.

**:** Choose color of grid background points. If major lines are activated single lines based on the grid size are displayed.

: Choose color of major grid lines.

9.4.1.1.2 Colors

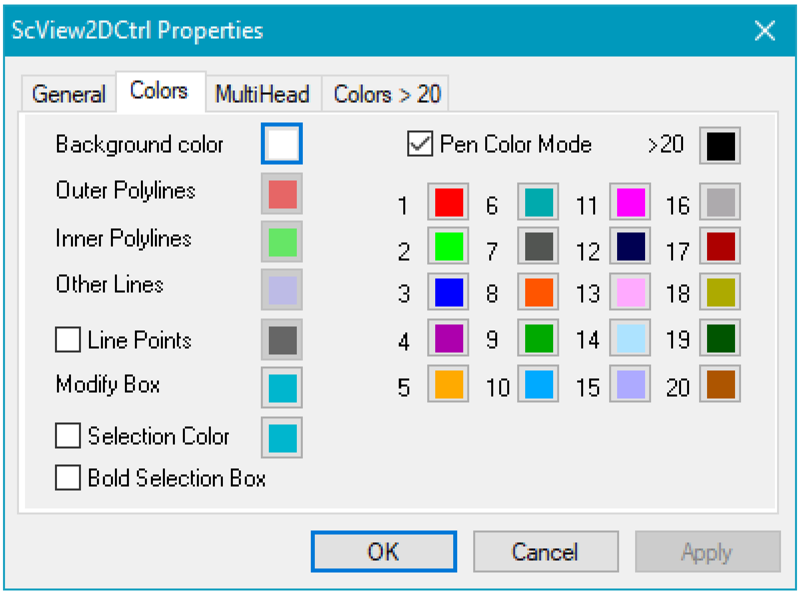


Figure 185: ScView2DCtrl_Colors

With these check boxes the user can define the colors of background, point of crossing lines and the selected entities. Here, the user can also define the colors of the different pens (also possible in the pen settings themselves).

Default View2D object colors:

View2D object	Color	RGB HEX	RGB DEC
Background color		ffffff	255 255 255
Outer PolyLines [a]		ff0000	255 000 000
Inner PolyLines [a]		00ff00	000 255 000
Other Lines [a]		aaaaff	170 170 255
Line Points [b]		000000	000 000 000
Modify Box		00b6cc	000 182 204
Selection Color [c]		00b6cc	000 182 204

Table 22: Default View2D object colors

[a]: Colors for Outer PolyLines, Inner PolyLines and Other Lines are used, when Pen Color Mode is disabled.
[b]: Color for Line Points is used, when Line Points is enabled.
[c]: Selection Color is used, when both Selection Color and Pen Color Mode are enabled.

Background Color: Choose a background color of the View2D.

Outer PolyLines: Color for Outer PolyLines is used, when Pen Color Mode is disabled.

Inner PolyLines: Color for Inner PolyLines is used, when Pen Color Mode is disabled.

Other Lines: Color for Other Lines is used, when Pen Color Mode is disabled.

Line Points: Color for Line Points is used, when Line Points is enabled.

Modify Box:

Selection Color: Selection Color is used, when both Selection Color and Pen Color Mode are enabled.

Bold Selection box: The user can select entities while drawing a box with the mouse. If enabled, this box will be displayed bold.

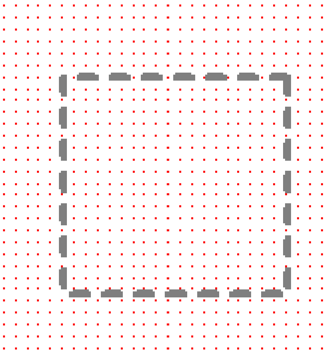


Figure 186: Bold selection box

Pen Color Mode: Define pen colors for pen 1-20. Further pens can be edited at Colors > 20.

Default pen colors:

Pen number	Color	RGB HEX	RGB DEC
1		ff0000	255 000 000
2		00ff00	000 255 000
3		0000ff	000 000 255
4		aa00aa	170 000 170
5		ffaa00	255 170 000
6		00aaaa	000 170 170
7		555555	085 085 085
8		ff5500	255 085 000
9		00aa00	000 170 000
10		00aaff	000 170 255
11		ff00ff	255 000 255
12		000055	000 000 085
13		ffaaff	255 170 255
14		aaffff	170 255 255
15		aaaaff	170 170 255
16		aaaaaa	170 170 170
17		aa0000	170 000 000
18		aaaa00	170 170 000
19		005500	000 085 000
20		aa5500	170 085 000
>20		000000	000 000 000

Table 23: Default pen colors

9.4.1.1.3 MultiHead

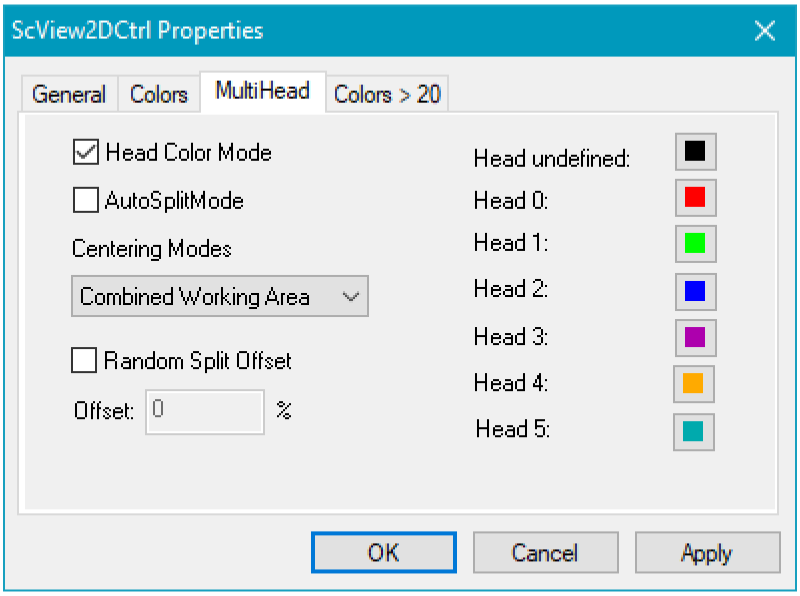
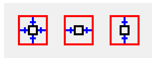


Figure 187: ScView2DCtrl_multihead

Head Color Mode: If enabled one can define a color for each head. All entities within one head will be shown with the defined color here. If disabled, the colors defined in the pens are used.

Auto Split Mode: Each entity will be automatically assigned a head if it is in its working area. If not checked a manual MultiHead Split is required.

Centering Modes: Define the reference of the possible centering modes: 

Combined Working Area: Use the combined working area for centering.

Automatic: If a part of an entity is located in a head, it will be centered on this head.

Ask User: A window is opened where the user can choose on which head the centering should be executed. Choose the wanted head in the drop down menu.

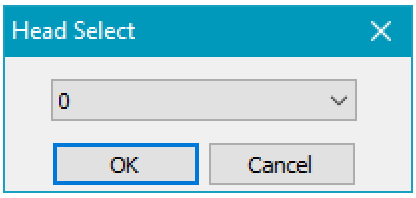


Figure 188: MultiHead centering reference head

Random Split Offset: Function is similar to *Cog Vectors in Overlap Area*. If enabled a percent value can be defined. The overlapped area of 2 heads will then be splitted randomly.

Default MultiHead colors:

Head number	Color	RGB HEX	RGB DEC
Head undefined		000000	000 000 000
Head 0		ff0000	255 000 000
Head 1		00ff00	000 255 000
Head 2		0000ff	000 000 255
Head 3		aa00aa	170 000 170
Head 4		ffaa00	255 170 000

Head number	Color	RGB HEX	RGB DEC
Head 5		00aaaa	000 170 170

Table 24: Default MultiHead colors

9.4.1.1.4 Colors >20

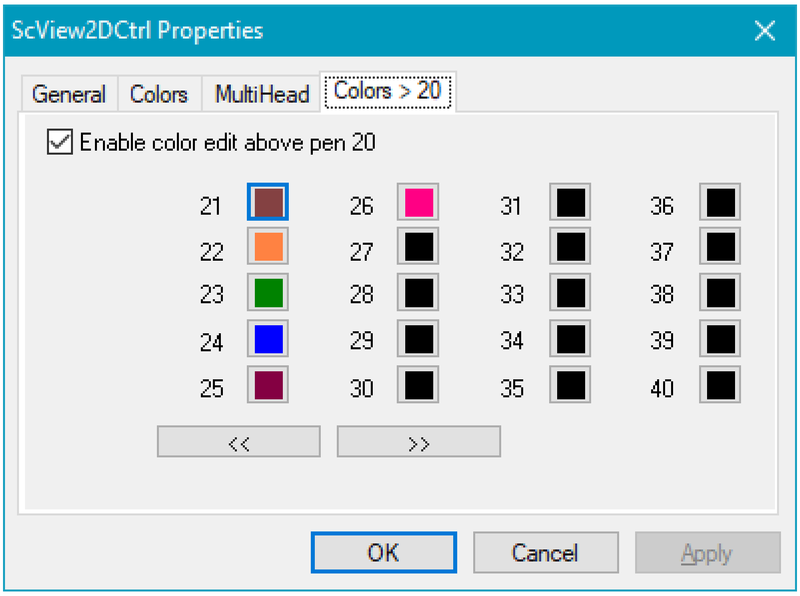


Figure 189: ScView2DCtrl_Colors>20

Here you can change the color of the pens above 20 until 256. You can also change the color in the Pen Settings → Edit Pen.

Enable color edit above pen 20: If enable, colors can also be assigned to pens above 20

9.4.2 Print Preview

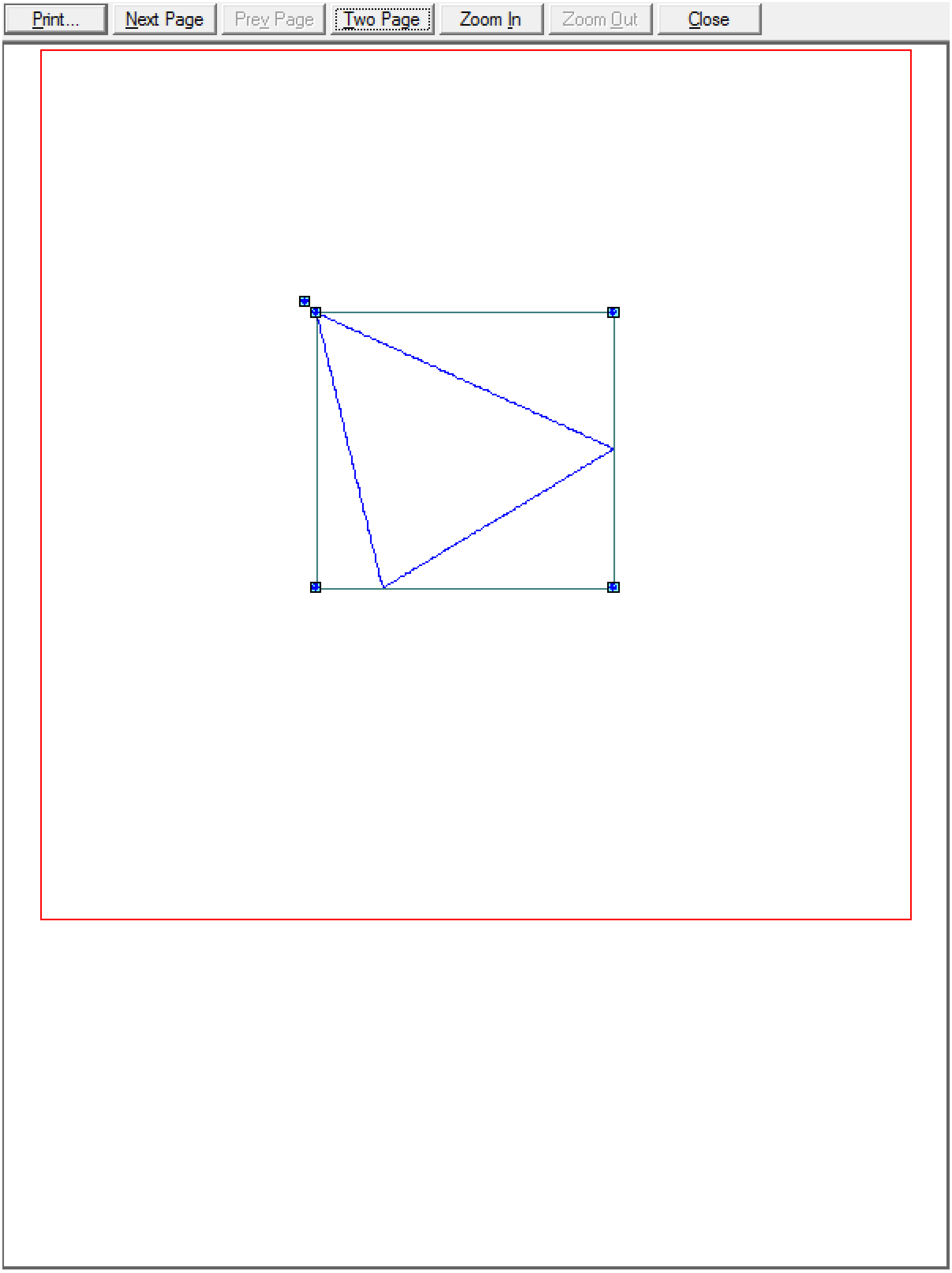


Figure 190: Print Preview Window

The print preview can be reached by the menu *File* → *Print Preview* or by the Camera Toolbar. This is a standard windows print preview, for further information see the windows help.

9.5 Entity Property Sheet

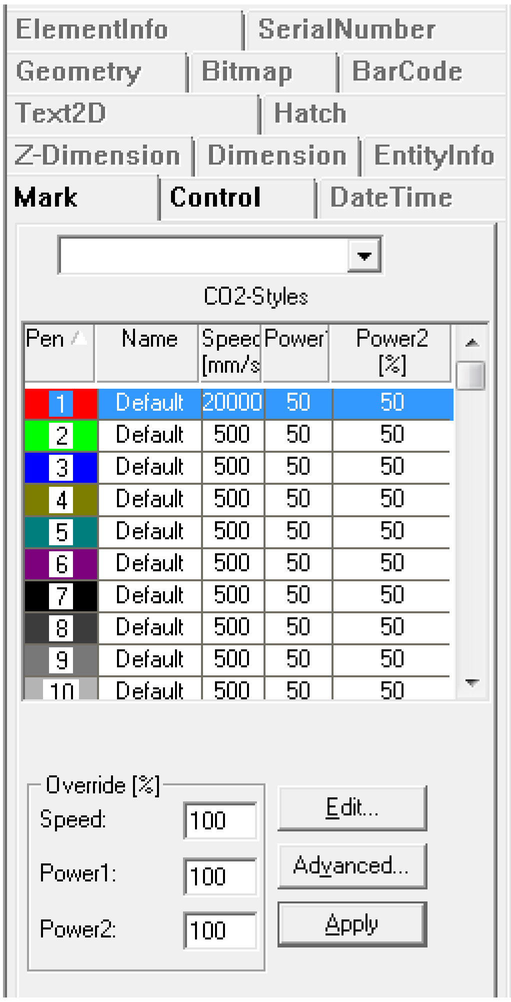


Figure 191: Entity Property Sheet

The Entity Property Sheet contains one Property Page for each possible property.

Only those Property Pages are active which correspond with a property available for the selected object.

For a detailed description of a special property page follow these links:

- Barcode
- Bitmap
- Control
- DateTime
- Dimension
- ElementInfo
- EntityInfo
- Geometry
- Hatch
- Mark
- SerialNumber
- Styles
- Text2D
- Z-Dimension

NOTE

The property page Z-Dimension is only available with Optic3D.

9.6 Status Bar

The mark status bar displays some information about the runtime of SAMLight and the executed marking processes:

Figure 192: Mark Status Bar

From left to right: the number of jobs, that has been start marking; Process [h:mm:ss]: the process time of SAMLight, means the time SAMLight has been running; Laser [%]: Total Mark / Process x 100; Last Mark [s]: the duration of the last completed marking process; Total Mark [s]: the total duration of all completed marking processes.

Last Mark is only a rough estimate for USC-1.

10 Entities (Objects)

This chapter describes the various kind of objects that can be created within SAMLight or that can be loaded as a job file.

10.1 Entity Hierarchy

Entities: All objects in the application are entities by definition. The three main categories of entities are elements, containers and groups and in general an entity is either an element or a container or a group of entities. The elements keep the real geometric data like lines, points and pixels, and the so called containers contain elements.

Elements:

Primitive Elements: Primitive elements are single points and single straight lines.

LineArray, PolyLine and PixelArray: *LineArray*, *PolyLine* and *PixelArray* are the next level of elements. *LineArray* and *PolyLine* represent sets of straight lines and *PixelArray* represents a set of pixels (points with gray values).

The difference between LineArray and PolyLine is as follows: The purpose of the element *LineArray* is to represent a set of lines. So the items of a *LineArray* are single straight lines described by their start and end point. In contrary to this a *PolyLine* is itself a line consisting of straight lines. So the description of a *PolyLine* is a sequence of points and the straight lines connect two subsequential points. They keep items of the corresponding kind in sequential order. For a *PolyLine* the item is a point p. Rectangles, triangles and ellipses are special closed *PolyLines*. So they are derived from *PolyLine*. A hatch is a special *LineArray* and therefore is derived from *LineArray*.

LineArrays,PolyLines and PixelArrays: *LineArrays*, *PolyLines* and *PixelArrays* are sets of special elements. A *LineArrays set* contains one or more *LineArray* elements. A *PolyLines set* contains one or more *PolyLine* elements. A *PixelArrays set* contains one or more *PixelArray* elements.

Containers: Container entities always consist of specific predefined numbers and types of subentities.

Layer: The Layer contains exactly three elements: one *PolyLines*, one *LineArrays* and one *PixelArrays*. This entity type is used for geometrical objects with a heterogeneous object description which are especially hatchable objects. For example if one creates a rectangle (see Object Toolbar) an entity of type Layer will be created in the Entity List (see Entity List) because a rectangle is a hatchable object. The Polylines of the Layer keep the PolyLine that bounds the rectangle and the *LineArrays* will keep the hatches (see Hatch).

10.2 Geometry Objects

Geometric objects and their settings are explained here: Rectangle, Ellipse, Spiral

1. Rectangle:

Click on the Rectangle icon to create a rectangle:

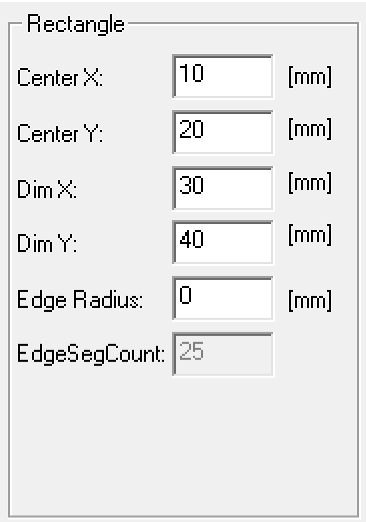


Figure 193: Rectangle properties

Rectangle:

Center, Dimension: It is possible to define the center and the dimension of the rectangle.

Edge Radius: One can generate a rectangle with round edges by defining an edge radius and the EdgeSegCount (Number of edges) to generate for each edge.

2. Ellipse:

Click on the Ellipse icon to create an ellipse:

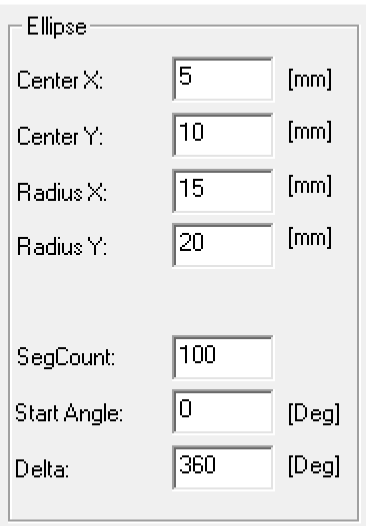


Figure 194: Ellipse properties

Ellipse:

Center X, Center Y: These two values define the center coordinates of the ellipse.

Radius X, Radius Y: With these values the radius in horizontal and vertical direction can be changed. If the values are equal the resulting object is a circle.

Segment count: The ellipse consists of a number of segments. This number is calculated to the complete 360° ellipse.

Start Angle, Delta: With these parameters it is possible to define arcs. Here the start angle defines the angle at which the elliptic or circular arc has to start relative to the x-axis and delta defines the angle of the arc relative to the start angle.

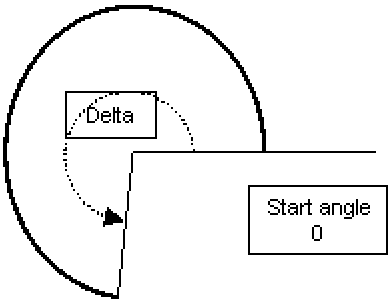
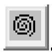


Figure 195: Example of an arc

3. Spiral:

A Standard Spiral can be created with the object toolbar button .

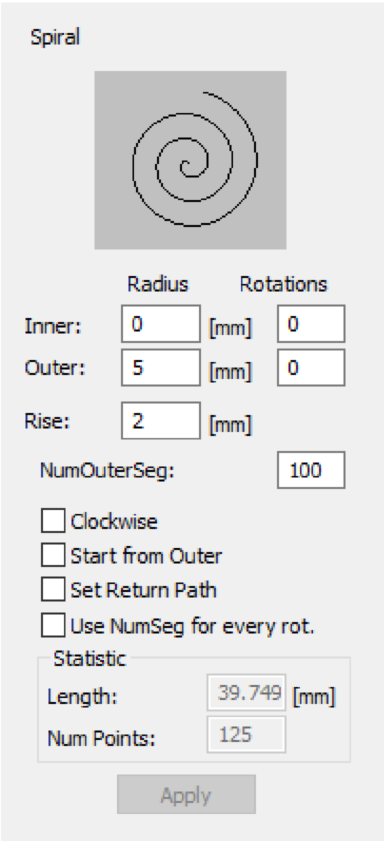


Figure 196: Spiral properties

Radius: Defines the inner and outer radius of the spiral.

Rotations: Defines the number of rotations on the inner or outer circle of the spiral.

Rise: Defines the distance between succeeding circles inside the spiral.

NumOuterSeg: Defines the number of segments of the outer circle. This proportion is taken as for the whole spiral.

Clockwise: Defines the orientation of the convolution.

Start from Outer: Defines the start point of the polyline.

Set Return Path: If checked a return path is added and the spiral ends at its start point.

Use NumSeg for every rot.: If checked the value for the NumSeg is used for every rotation.

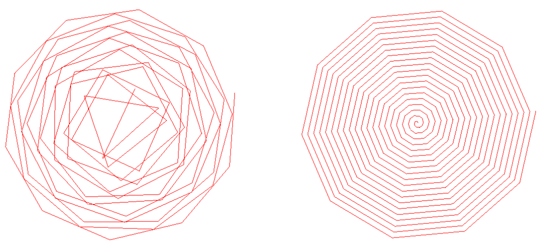


Figure 197: Left without "Use NumSeg for every rot" - Right with "Use NumSeg for every rot"

Statistic:

Length: Length of the polyline.

Num Points: Number of points on the polyline.

You can change the view to Point Editing to see the order the single points will be marked.

10.3 Barcode

Generate a barcode by pressing the Barcode2D button in the object toolbar. Move the mouse to the desired position and press the left mouse button. The barcode property page appears.

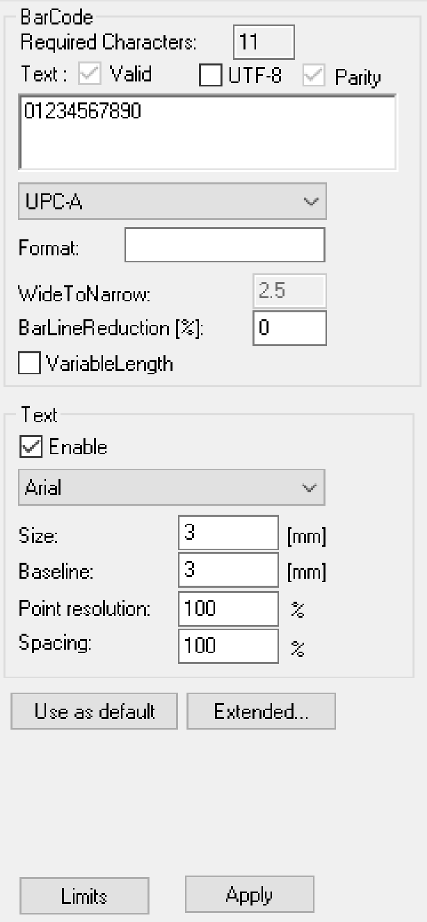


Figure 198: Barcode Dialog

BarCode:

Text: Enter any number, a date or some text into the text field.

Required Characters: Some barcodes require a specific number of characters which is displayed in this field. If 1.. is displayed no specific number of characters is required.

Valid: This is checked when the characters in the text field are valid for the selected barcode type.

Format: See subchapter Format.

Choose the desired BarCode type in the drop down menu above 'Format'.

WideToNarrow: Relation between wide and narrow barcode lines and accordingly between the wide and narrow spaces among the barcode lines. The selectable number ranges between 2 and 3.

BarLineReduction: Reduces the size of the single bars.This is not available for DataMatrixEx, Dotcode, Code39Ex, 2of5Ex and Code128(2).

VariableLength: If this is activated, the size of the barcode changes with the length of characters encoded. E.g. the length of a generated EAN-128 barcode changes almost linear with the length of the text. If this is deactivated the size of the barcode outline doesn't change with the text size.

Text:

Size: Specifies the size of the text.

Baseline: Distance between the baseline of the barcode to the baseline of the text, see example below.

Point resolution: See chapter Text Properties.

Spacing: See chapter Text Properties.

Use as default: Uses the properties of the currently selected barcode object for the generation of new barcodes. The program saves these settings also for a new program start in case save settings on exit is checked in the general settings.

Extended...: See subchapter Extended

Figure 199: Example of a barcode

Limits: Opens the 'Size Limits' dialog. Length and Height limits of the barcode can be specified here.

Supported Barcode Types:

1D: 2 of 5, 2 of 5 EX, 3 of 9, Codabar, Codablock-F, Code 128 Subtype A, Code 128 Subtype B, Code-128, Code-128(2), Code-16K, Code-39, Code-39 EX, Code-49, Code-93, EAN, EAN-128, EAN-13, EAN-13+2, EAN-13+5, EAN-14, EAN-8, EAN-8+2, EAN-8+5, Ex Code39, Ex Code93, Expanded, Expanded-Stacked, GS1-128, GS1-128 CC-A/B, GS1-128 CC-C, I-2/5, ITF-14, ITF-6, Limited, MicroPDF417, Omnidir, PDF417, PostNet-A, PostNet-C, PostNet-Cp, RSS, SSCC, Stacked, Stacked-Omnidir, Truncated, UPC-A, UPC-A+2, UPC-A+5, UPC-E

2D: Aztec, DataMatrix[1], DataMatrixEx[1], DataMatrixRect[1], Dotcode, GS1-DataMatrix[1], Micro QR Code, QR Code, QR Code EX

[1] These Barcodes comply with ECC200 and ISO/IEC 16022:2000.

10.3.1 Barcode Format

It is possible to differentiate between text encoded by the barcode and text shown below the barcode. Therefore a control code is given. To enable following control code, enter *%H* into the format field of the barcode property page. Note: It is not supported for all types of barcodes.

Control Code	Meaning
%b	Start to be only in barcode
%h	Start to be in human readable text only
%e	End of control code
%%	Insert a % sign

Figure 200: Available barcode control codes

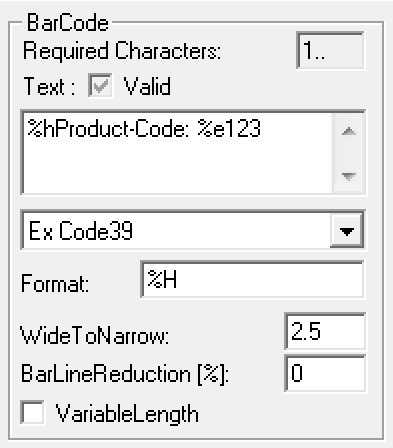


Figure 201: Example of a control code

Figure 202: Example output

For the 2D Barcode DataMatrixEx, it is also possible to enter 4 values separated by a comma. Then the text of the barcode is wrapped around it.

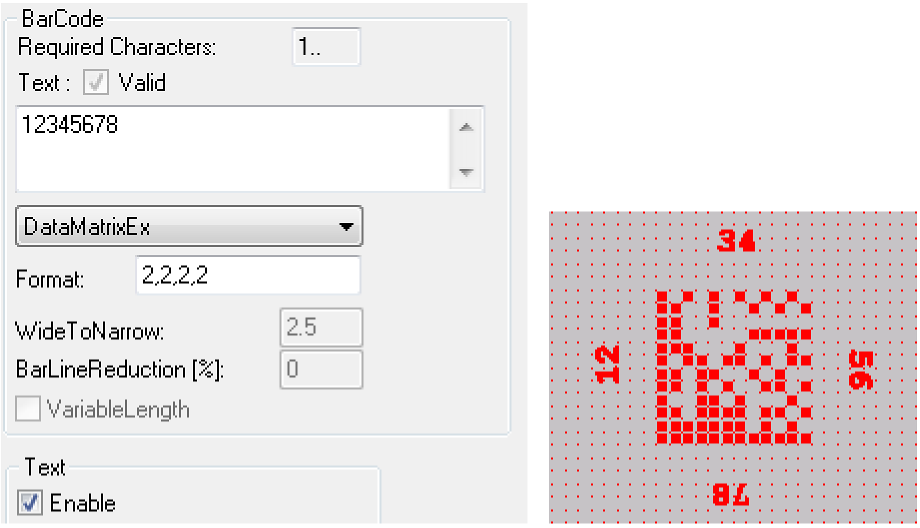


Figure 203: Example of a special formatted barcode

10.3.1.1 2 of 5 Ex

If a **2 of 5 EX** object is selected, the following dialog opens after pressing the Extended... button in the barcode property page.

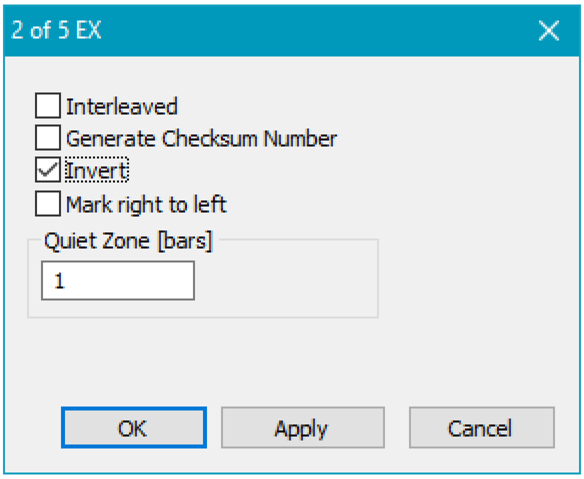


Figure 204: 2 of 5 EX Extended Dialog

Interleaved: In order to use this feature the barcode has to have an even number of ciphers. Then the barcode will be made smaller while increasing the information density.

Invert: The barcode will appear inverted.

Quiet Zone: Generates a border of defined thickness at the left and right end of the barcode.

10.3.1.2 Code-39 Ex

If a **Code-39 Ex** object is selected, the following dialog opens after pressing the Extended... button in the barcode property page.

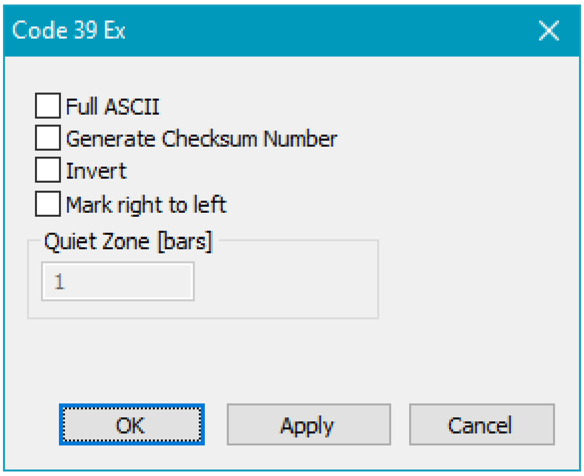


Figure 205: Barcode Extended for Code-39 Ex

BarCode Extended dialog for Code-39 Ex:

Full ASCII: When using ASCII encoding, the range of characters can be extended. But make sure your barcode reader is compatible with ASCII.

Generate Checksum Number: Adds a checksum which is compatible with ASCII encoding.

Invert: When enabled, inverted barcodes can be used in standalone mode. To improve readability, the quiet zone could be increased.

Mark right to left: Changes the direction of marking.

Quiet Zone [bars]: Is only relevant for inverted barcodes. Adds the specified number of bars on the right and left hand side.

10.3.1.3 Code 128

In SAMLight, the Code 128 barcode is also available.

This barcode has three subtypes: A,B and C.
Subtypes A and B can contain letters while subtype C only holds numerical values. Subtype A and B can be selected separately. Subtype C will be used automatically if you use Code 128 and the data only contains numerical values.

10.3.1.4 Code 128(2)

If a **Code 128 (2)** object is selected, the following dialog opens after pressing the Extended... button in the barcode property page.

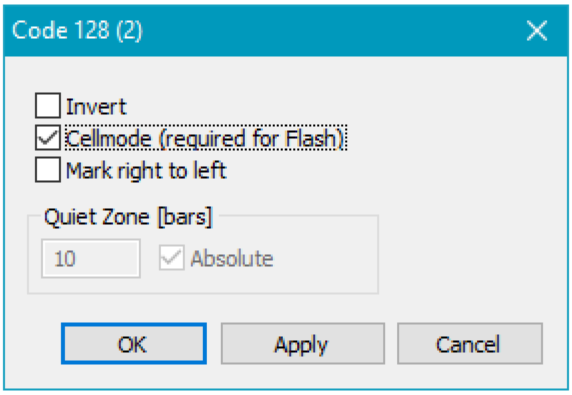


Figure 206: Code 128 (2) Extended Dialog

Invert: The barcode will appear inverted.

Cellmode: The barcdoe will consist of cells.

Quiet Zone: Generates a border of defined thickness around the barcode.

10.3.1.5 DataMatrixEx

If a **DataMatrixEx** object is selected, the following dialog opens after pressing the Extended... button in the barcode property page.

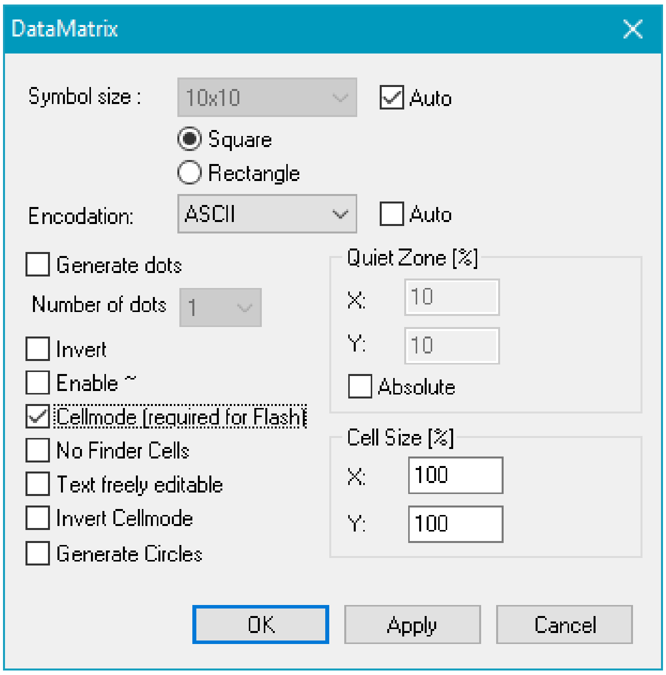


Figure 207: Data Matrix Extended Dialog

Symbol size: Number of cell rows and columns of the DataMatrix.

Auto: Chooses the smallest possible size inside the combobox for the selected barcode object.

Encodation: Choose the type of the encodation. Available encodations are:

- ASCII
- Base256
- C40
- Text
- ANSI X12
- EDIFACT
- GS1 - Application Identifiers must be written with round brackets, like (01). The **FNC1** characters must not be written, they are added automatically.

Auto: Chooses the encodation for which the selected barcode text is being optimally compressed.

Example GS1 DataMatrixEx:

- Barcode type: DataMatrixEx
- Encodation: GS1
- Text: (01)03453120000011(17)191125(10)ABCD1234
- Encoded string: **FNC1**01034531200000111719112510ABCD1234

Figure 208: GS1 DataMatrixEx example

Generate dots: If Generate dots is checked the barcode consists of single points. (For other 2D barcodes, please refer to Generate Dots.)

Invert: If Generate dots is not selected then the barcode can be inverted. If inverted the barcode must be bordered to prevent the outer barcode structures to disappear. Therefore a *Quiet Zone* is defined. The width of the zone can be given in absolute units or in the percentage of the width. The selectable percentage ranges between 1 and 50.

Enable~: Allows to encode 3-digit decimal values. The format is *~d* followed by 3 digits. This leads to a compression of the data because ~d will not be used and the following 3 digits as decimal number will be encoded by the according character in the ASCII-table. *For example: ~d090 will be changed into 90 of the ASCII-table, e.g. the character Z.*

Cellmode (required for Flash): If Cellmode is checked, the barcode consists of a closed polygon for each cell. Cellmode is required for Flash applications. See below for examples.

No Finder Cells: If the barcode is generated by cells then the finder zone is drawn as one closed polygon.

Text freely editable: If checked the user may define an arbitrary text independent of the content of the DataMatrix.

Invert Cellmode: If this checkbox is activated then the DataMatrix will be inverted. So it is possible to define a Quiet Zone. The width of the Quiet Zone can be defined in unit Cells. This works within SAMLight and with the USC-2 Flash but not with the FEB-1 board.

Generate Circles: If this checkbox is activated, each cell in cellmode is a circle, not a square.Generate Circles is only available if Cellmode is activated.

Cell Size: Active if Generate Cells is checked. The size of a single cell can be defined. If it is 100% the cells contact each other.

Quiet Zone: Generates a border of defined thickness around the barcode.

standard barcode

dots generated barcode

cells generated barcode, cell size 75%

Figure 209: DataMatrixEx Examples

10.3.1.6 GS1 Barcodes

If selecting a GS1 barcode, for example GS1-DataMatrix, the application identifiers defined by GS1 have to be in brackets followed by its value. Application Identifiers are for example:

Identifier	Description
(01)	GTIN (Global Traded Item Number)
(10)	Batch or Lot Number
(11)	Product Date (as YYMMDD)
(15)	Best Before Date (as YYMMDD)
(21)	Serial Number
(400)	Purchase Order Number
(422)	Country of Origin (as ISO code)

Table 25: GS1-DataMatrix Application Identifiers

10.3.1.7 QR Code

The QR-Code is a special 2D barcode format. In addition to numerics, it can also contain the letters from a to z and special characters like +, -, /, % ... . To enable this option, click on *Extended...* in the BarCode property page. Then in the window that pops up select *Byte* in the field *Barcode Mode*.

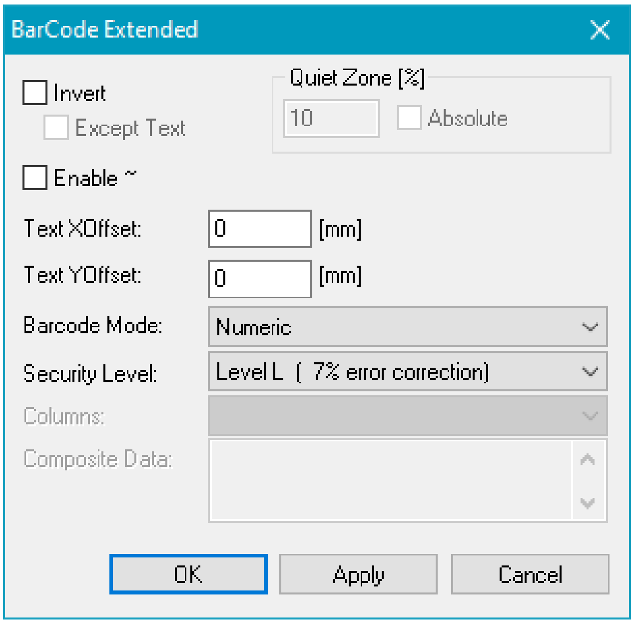


Figure 210: Barcode Extended for QR Code

BarCode Extended for QR Code:

Enable ~: Allows to encode 3-digit decimal values. The format is *~d* followed by 3 digits. This leads to a compression of the data because ~d will not be used and the following 3 digits as decimal number will be encoded by the according character in the ASCII-table. *For example: ~d090 will be changed into 90 of the ASCII-table, e.g. the character Z*

Barcode Mode: Choose the desired mode in the drop down menu. Options are 'Numeric', 'Alpha', 'Byte', and 'Kanji'.

Note that, depending on the selected mode, different types of characters are allowed. You will get direct feedback, via the 'Valid' checkbox (see Barcode).

In addition, the selected mode also changes the number of valid characters!

For further information on Figure 210 see Barcode Extended.

10.3.1.8 QR Code EX

A special barcode type QR Code EX is available. This barcode is like a QR Code barcode with the additional feature that an Entity can be embedded in the center of the Barcode. Therefore select QR Code EX in the barcode drop down box and click on *Extended.* The following dialog will be shown:

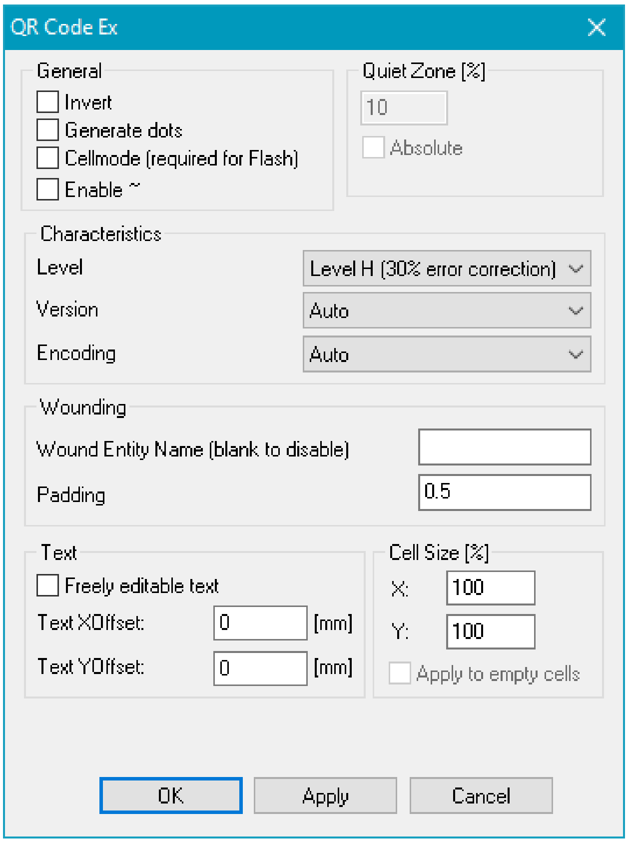


Figure 211: QR Code EX extended dialog

Figure 212: QR Code with "USC" wounding

General:

Invert: Inverts the barcode.

Generate dots: The barcode will consist of dots. If this checkbox is activated, the parameters of the drill property page will be taken (if activated) instead of the parameters of the scanner property page.

Cellmode (required for Flash): The barcode will consist of cells. Cellmode is required for Flash applications.

Enable~: Allows to encode 3-digit decimal values. The format is *~d* followed by 3 digits. This leads to a compression of the data because ~d will not be used and the following 3 digits as decimal number will be encoded by the according character in the ASCII-table. *For example: ~d090 will be changed into 90 of the ASCII-table, e.g. the character Z.*

Quiet Zone [%]: Apply a border to the barcode.

Absolute: Switches from % to absolute unit according to the current SAMLight unit.

Characteristics: Different level of error correction and barcode size can be chosen via drop down menu.

Level: Set the error correction level.

Version: Force the size (x-y).

Encoding: Set the encoding mode. Avaiable is: Automatic, Numeric, Alpha, Byte and Kanji

Wounding: Embed an Entity inside the QR Code.

Wound Entity Name: Name of the entity that has to be embedded inside of the QR code. If the edit field is blank no entity will be embedded.

Padding: Define a border around the embedded entity.

Text: Specify an offset in x and y for a human readable text which can be displayed in addition to the barcode.

Freely editable text: The position of the human readable text can be placed freely and will be kept at this position also if the content of the barcode is changed.

Text XOffset: Shifts the human readable text in x direction. This offset is kept also if the content of the barcode is changed.

Text YOffset: Shifts the human readable text in y direction. This offset is kept also if the content of the barcode is changed.

Cell Size [%]: The cell size can be reduced in x and y. The reduction is specified in %.

Apply to empty cells: If this checkbox is active, the cell size reduction is applied to the empty (white) cells instead of the filled (black) cells.

NOTE

QR Code mode is automatically adjusted to the type of characters in the text field (see Barcode).
(e.g. '12345' = numeric mode, '12345A' = alphanumeric mode, '12345Aa' = binary mode)

With this change of mode, the number of allowed characters often changes as well.

10.3.2 Scaling

When scaling barcodes that have the text feature enabled, the text is normally not scaled with the barcode. Only for those barcodes that can have the checkbox "Extended → Text freely editable" enabled, like DataMatrixEx or QRCodeEX can scale the text with the barcode when this checkbox is activated. Mirroring can lead to unwanted results for those barcodes that have no "Extended → Text freely editable" checkbox.

10.3.3 Barcode Extended

Press the Extended... button in the barcode property page to get dialog in Fig. 213 .

There exist some special dialogs listet in:

- 2 of 5 EX
- Code-39 Ex
- Code 128 (2)
- DataMatrixEx
- QR Code
- QR Code EX

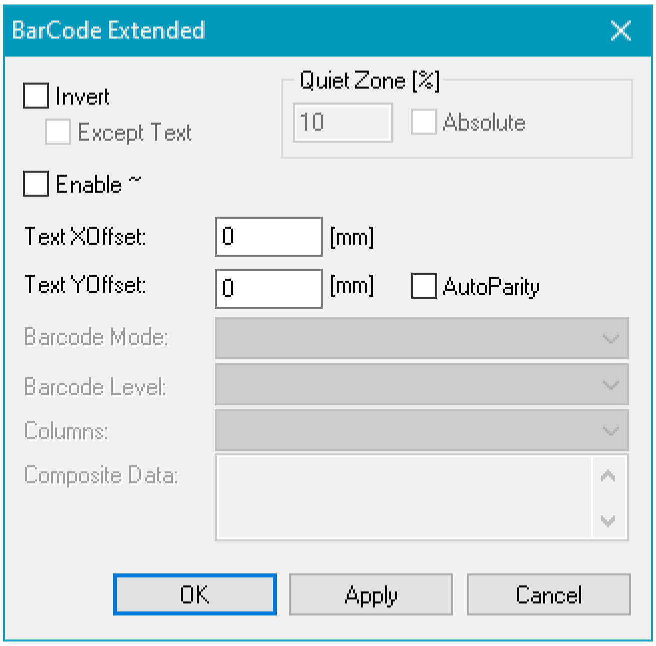


Figure 213: Barcode Extended Dialog

Invert: Inverts the dark and bright parts of the barcode. The barcode must be bordered to prevent the outer barcode lines from disappearing. Therefore a QuietZone is defined, see example below. The width of the zone can be given absolute units or in percentage of the width. The selectable percentage ranges from 1 to 50.

AA-23456-789

Figure 214: Inverse Barcode Example

Except Text: This suboption of Invert excludes inverting the text when Invert is selected and is implemented at the moment just for the EAN-13-Barcodes.

Enable ~: Allows to encode 3-digit decimal values. The format is *~d* followed by 3 digits. This leads to a compression of the data because ~d will not be used and the following 3 digits as decimal number will be encoded by the according character in the ASCII-table. *For example: ~d090 will be changed into 90 of the ASCII-table, e.g. the character Z.* This feature should be supported for each barcode mode. Only the *ex* modes could not have this function.

Text XOffset: This value can be used to move the position of the text in horizontal direction that is activated for the barcode. For positive values the offset point of the barcode text is moved to the left, for negative values it is moved to the right.

Text YOffset: This value can be used to move the position of the text in vertical direction that is activated for the barcode. For positive values the offset point of the barcode text is moved down, for negative values it is moved up.

Barcode Mode / Security Level / Code Columns: These parameters are optional and depend on the barcode type that was chosen. So the number of options, the meaning of the possibly available options and the parameters that can be selected here are defined by the related barcode specification.

The following Barcodes have a Mode option: Aztec, Code 16k, Maxicode, MicroPDF417, PDF417, QR Code, RSS

The following Barcodes have a Security Level: Aztec, MicroPDF, PDF417, QR Code

The following Barcodes have Columns: Composite, MicroPDF417, PDF417

Security Level:

For the QR-Code the following Security Levels for error correction are available:

- Level 0: 7%
- Level 1: 15%
- Level 2: 25%
- Level 3: 30%

A Security Level of 30% means, that the error correction can read the Barcode if up to 30% of the Barcode is disrupted. For different Barcodes there exist a different number of Security Levels with different error correction capabilities.

10.3.4 Barcode Reader

It is possible to connect a barcode reader to the PC and to read barcodes in SAMLight. Therefore create a serial number entity in the Job. Then in the Serial Number Property Sheet click on the *Advanced* button. In the window that is popping up activate the checkbox *Barcode reader mode* in the field *Popup edit box*. Now if you do a mark a window is popping up after the mark has finished, see the picture below:

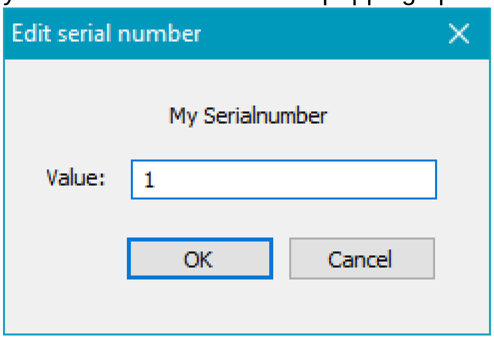


Figure 215: Edit serial number dialog

Here you can enter an arbitrary number or, if a barcode reader device is connected to the PC, you can read any barcode with this device and the value of the barcode will be assigned to the serial number in the job.

10.3.5 Limits

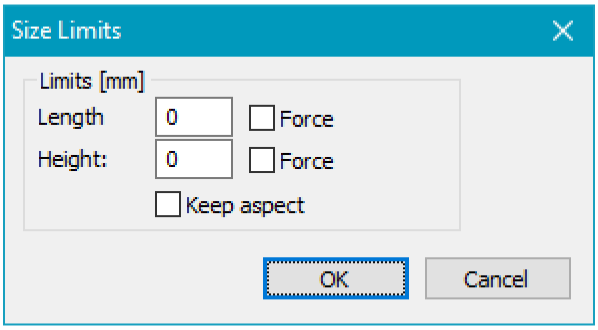


Figure 216: Size Limits Dialog

Length: Set the maximum length of the barcode. Set 0 to disable.

Height: Set the maximum height of the barcode. Set 0 to disable.

Force: Will force the barcode size to the given values.

Keep aspect: If only one parameter is changes, the second will be applied to that the aspect ratio is kept.

Changes will be applied after clicking *OK* and the entity is deselected.

10.4 Bitmap

After importing a Windows Bitmap, it will be converted and displayed as a 8bit gray image. To bring it to the scanner output, it is necessary to calculate a Scanner Bitmap. The generation can be done in the property page of the bitmap. After selecting the bitmap, the bitmap property page becomes active:

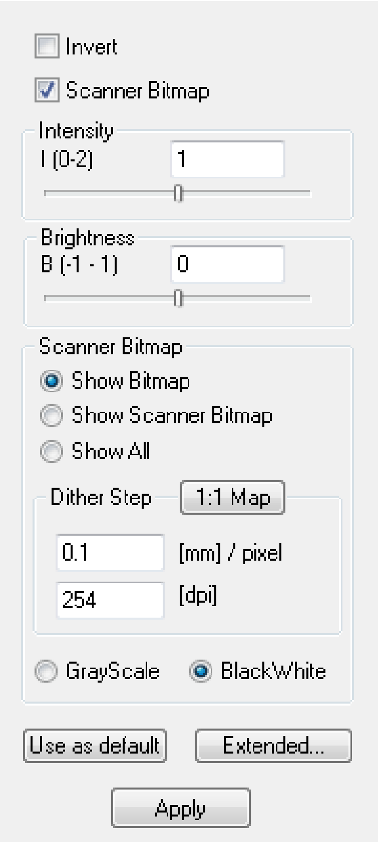


Figure 217: bitmap property page

Invert, Intensity, Brightness: Work on the original bitmap

Scanner Bitmap: Select this check button to generate the Scanner Bitmap. Press Apply. If this is not selected, no scanner bitmap will be created or if it already exists it will be deleted after Apply is pressed.

Scanner Bitmap:

Show Bitmap: Shows the original bitmap.

Show Scanner Bitmap: Shows the scanner bitmap.

Dither Step: Defines the resolution for each pixel inside the scanner bitmap. If for example Dither Step = 0.1, the distance between two neighbouring pixels is 0.1, means one mm on the scanner bitmap contains 10 pixels in x-direction * 10 pixels in y-direction. The resolution should be in the range of the laser focus. You can also define the Dither Step via dpi (dots per inch).

GrayScale: If enabled, the grayscale values will be kept when transforming the bitmap to the scanner bitmap.

BlackWhite: If enabled, the grayscale values will be approximated by specific placements of black and white pixels to give the impression of gray values. This is a similar method as with a Black/White LaserJet. There are different algorithms available in Bitmap Extended for this mode.

NOTE *Scanner Bitmap needs to be enabled when Improved Bitmap Handling is active.*

NOTE *The grayscale mode is only available for scanner cards with an appropriate mode. To scan bitmaps generated in grayscale mode, the hardware mode has to be enabled.*

For the dithered mode, *LaserOn* delay is set to *1* µs and *LaserOff* delay to 10 µs by software. *Jump Speed* is set to *Mark Speed*, except for the jump between two lines in one direction mode where the *Jump Speed* and the *Jump Delay* (defined within pen settings) is used. In all other cases, the scanner delays are set to *0*. In addition, the dithered mode uses the skywriting parameters of the pen if enabled.

Extended...: For how to set up the parameters for marking gray scaled bitmaps in bi-directional mode, see sub-chapter Marking Bidirectional. For the other *Extended...* features see Bitmap Extended.

Apply: Starts the generation of the Scanner Image if the check button Scanner Bitmap is checked. Else deletes an existing Scanner Image.

Example:

Figure 218: original picture (left) and scanner image in error diffuse mode (right)

NOTE

Hardware Mode: For drawing bitmaps in grayscaled mode the hardware mode has to be set which means that scanner movement and laser burning are done at the same time. To enable the hardware mode it is necessary to select the pulse width modulation mode PixelPWM or the amplitude modulation mode PixelAM inside the optic settings advanced dialog. Also the selection of both is possible. In addition the hardware flag inside the scanner settings for pens has to be checked.

If hardware mode is checked, it is recommended to set the LaserOn delay within this page to 1 µs and the LaserOff delay to 2 µs.

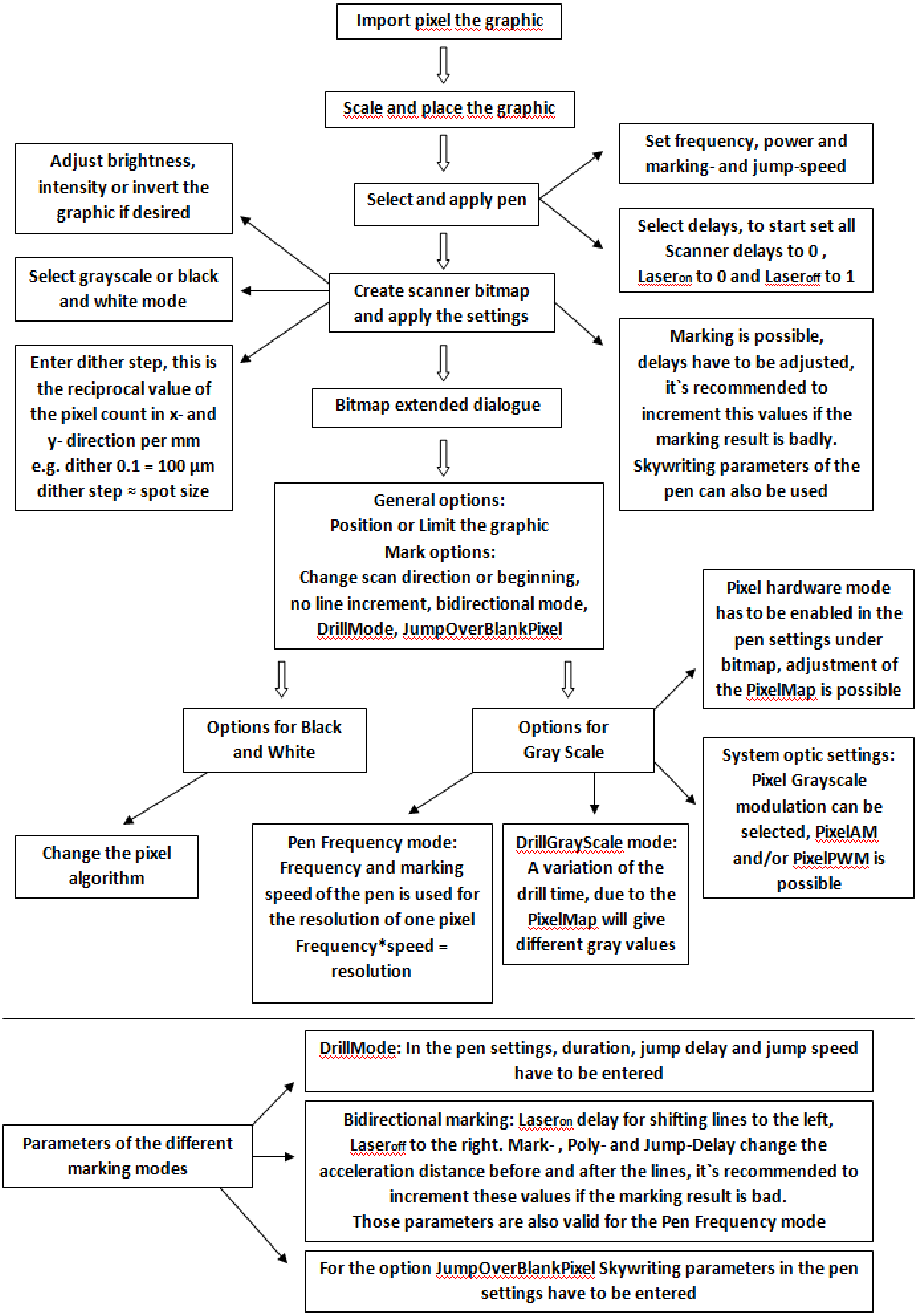


Figure 219: HowTo flowchart for Bitmap Marking

For details about pixelmode see chapter Backgrounds.

10.4.1 Bitmap Extended

Press the Extended... button in the bitmap property page to get the dialog shown below.

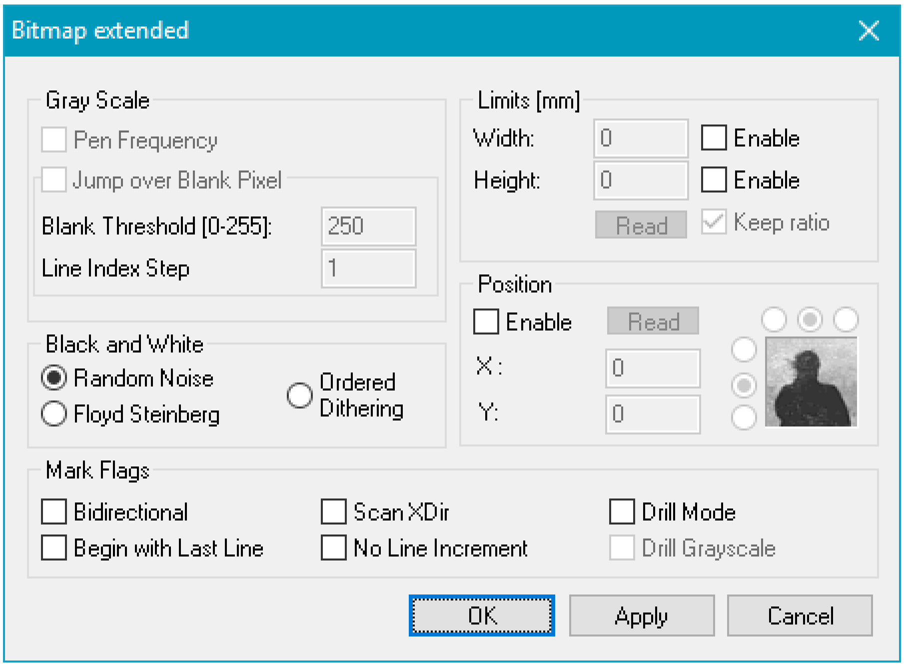


Figure 220: Bitmap Marking Extended Dialog

NOTE *The Following settings are valid for Reimport which is useable for a Client Interface application.*

Gray Scale:

Pen Frequency: Check this box to take the pen frequency for marking bitmaps. This will adapt the resolution of one pixel line to the speed and the frequency of pen.

Jump over Blank Pixel: To optimize marking time blank pixels will be skipped. SAMLight will skip all first and last pixels of a bitmap line with a gray value higher or equal than the threshold value 'Blank Threshold'. Blank pixels in between nonblank pixels will be jumped with MarkSpeed, not with JumpSpeed.

Blank Threshold: This value has a range from 255 (white) to 0 (black). The default value is 250.

Line Index Step: This feature can be used to reduce local heating of the target material at marking gray scale bitmaps. If 'Line Index Step' = x > 1 the bitmap will not be marked line by line in one run but in x runs where in each run every xth line will be marked. For example a Line Index Step of '3' results in this marking order: 1, 4, 7, 10, .., 2, 5, 8, 11, .., 3, 6, 9, 12, ..

Black and White:

Random Noise: Creates a rougher scanner bitmap than Floyd Steinberg, but doesn't tend to produce moiré patterns. As the name suggests, if used several times Random Noise generates different scanner bitmap patterns for the same bitmap.

Floyd Steinberg: Creates a smoother scanner bitmap than Random Noise, but tends to produce moiré patterns. Always generates identical scanner bitmaps for the same bitmap if applied several times.

Limits: Define Width and/or Height as a placeholder property for reimport. With keep ratio the aspect ratio is kept.

Position: Define a position for the bitmap placeholder. The coordinates of the reference point are defined with the X and Y edit fields. The radiobuttons define the point of attack of the bitmap.

Mark Flags:

> **Bidirectional:** If checked the scanner does not jump to the beginning but to the end of the next line if it reaches the end of a line. For more details please refer to Marking Bidirectional.
>
> **Begin with Last Line:** If checked the scanner begins drawing from the last line instead of the first line of the bitmap.
>
> **ScanXDir:** The default scanning direction is y, so the scanner moves from bottom to top while scanning. To choose x as scanning direction activate this checkbox.
>
> **No Line Increment:** If checked the scanner draws all bitmap lines into one line. This is necessary if for example the workpiece itself is rotated during marking.
>
> **DrillMode:** The pixels of a bitmap will be handled as single points. If the drill mode in the Pen is active, the points will be marked with the drill mode property page for pens. Each pixel will be handled if its gray value > 0. So as default it is a black / white mode.
>
> **DrillGrayscale:** Only available in combination with the DrillMode. The drill time on each pixel depends on its gray value. If the pixel is white, the drill time that is adjusted in the pen is executed. If it is black, the drill time is zero. In between the drill time of a pixel grows linearly with its gray value.

10.4.2 Marking Bidirectional

Usually the bitmaps are marked in one-directional mode. In the following it is described how to set up the parameters for marking in bi-directional mode. Press the *Extended...* button in the bitmap property page to show the dialog below and check the bidirectional box.

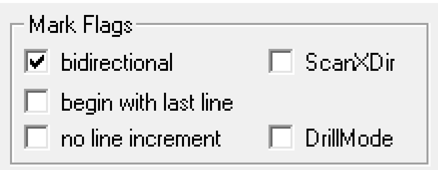


Figure 221: bitmap property page

This picture shows the way the laser should go to mark the bitmap properly.

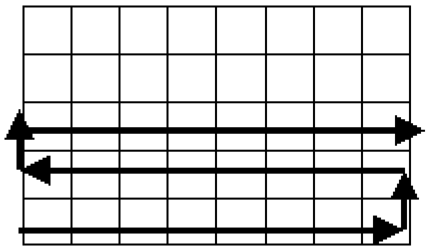


Figure 222: not corrected marking path

But this way of marking will cause picture mistakes. One reason is that a scanner needs a short startup time to reach constant speed and a constant signal frequency. The other reason is the delay of the scanner. To solve these problems an acceleration length was introduced:

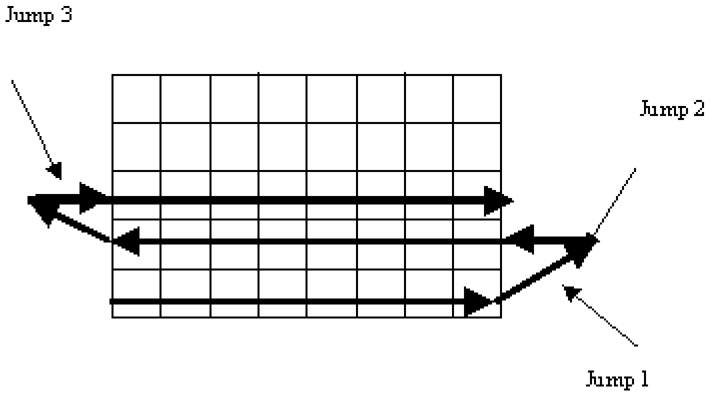


Figure 223: corrected marking path

The used jump speed is set equal to the speed of the pixelmarking. To enable the PixelShifting delays of the scanner the settings that are defined within scanner settings page for pens are used. *LaserOn* and *LaserOff* delays will automatically be set to *1*. The parameters stored in *LaserOn* and *LaserOff* will be used for calculating the shift of every second line: *LaserOn* for shifting left, *LaserOff* for shifting right.

10.4.3 Black and White

This is how Black and White (B&W) bitmap marking works:

Within a line:

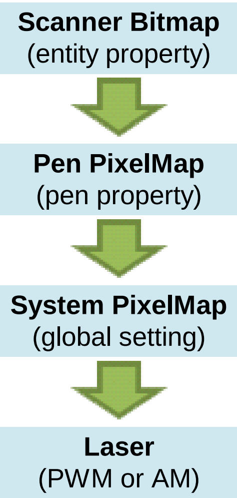
- Automatically used parameters:
  - LaserOn Delay = 1 µs
  - LaserOff Delay = 10 µs
  - Mark Delay = 0 µs
  - Poly Delay = 0 µs
  - Jump Delay = 0 µs
  - Jump Speed [mm/s] = Mark Speed [mm/s]
- Gate Frequency [kHz] = Mark Speed [mm/s] / Dither Step [mm] * 10-3
- Laser pulses / Pixel = Laser Frequency [kHz] / Gate Frequency [kHz]
- Skywriting:
  - StartDistance [mm] = StartLength [µs] * Mark Speed [mm/s] * 10-6
  - EndDistance [mm] = EndLength [µs] * Mark Speed [mm/s] * 10-6

Between lines:

- Jump Speed [mm/s] = Jump Speed [mm/s]
- Jump Delay [µs] = Jump Delay [µs]
- Bidirectional mode:
  - Shift of odd lines [mm] = (LaserOff - LaserOn Delay [µs]) * Mark Speed [mm/s] * 10-6

10.4.4 Grayscale

At the marking process, the greyscale value of the scanner bitmap will correspond to the laser power. That is why the greyscale adjusting can be used for fine-tuning your mark result and consider non linear behavior of the laser and the material. The brightness and the intensity can be changed for the original bitmap in the bitmap property page. After that the greyscales of the scanner bitmap can be adjusted via two different pixel maps:



The original scanner bitmap greyscales will be adjusted by the pen pixel map first and afterwards by the system pixel map again like you can see in the following flow chart:

- Pen PixelMap (useful to consider material behavior)
  Before marking every greyscale bitmap which is assigned to this pen will be changed by this pixel map.
- System PixelMap (useful to consider general laser settings like minimum power)
  Before marking every greyscale bitmap will be changed by this pixel map.
- Power Map is not used for gray scale marking at all.

Table 26: Order of greyscale adjustment

10.4.5 Improved Bitmap Mode

The improved bitmap handling offers a simplified handling of bitmaps with optimized and precise manipulation of laser signals.

The improved bitmap handling needs to be activated by the checkbox "Improved Bitmap Handling" in *Settings → System → Extras* (see Figure Figure 224). When activated, several settings necessary for bitmap marking will automatically be made which formerly had to be made manually.

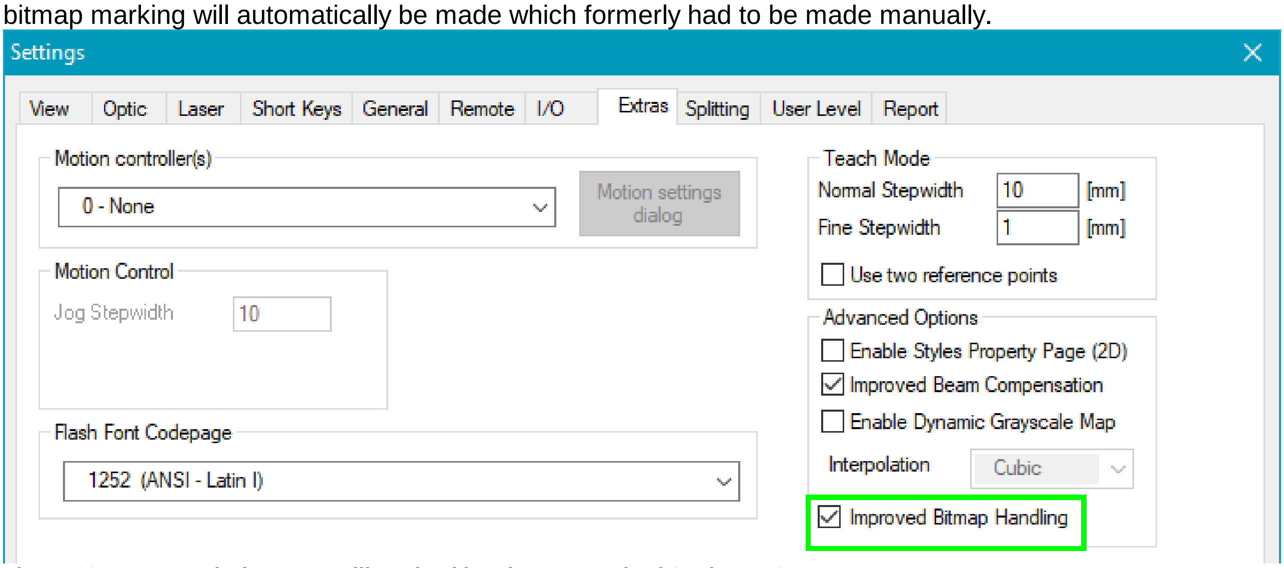


Figure 224: Improved Bitmap Handling checkbox in Extras tab of Settings - System

In addition, the bitmap tab in your pen will be adapted: a drop down menu wiill be available for selection of the exact bitmap mode with parameters for bitmap optimization.

The following improved bitmap modes are available:

- Gray Advanced: this is the optimized mode for grayscale bitmaps. The laser power will be modulated according to the grayscale value of each pixel and in synchronization with the scanner movement along each bitmap line. High quality bitmaps can only be achieved if an appropriate laser is used capable of fast enough power switching. This mode is available for USC-2 and USC-3 cards.

- B&W Normal: this is the standard mode for black&white bitmaps. During the scanner movement along each bitmap line, the laser output will be switched on and off according to the value of each pixel. Most scanner delays will be deactivated or set to small pre-defined values. The scanner can move with the mark speed and with the jump speed. White pixels at the beginning and at the end of each bitmap line are used for jumping, within black pixels they are part of the marking with constant marking speed.

- B&W Scanner Delays: this is the advanced black&white bitmap mode. During the scanner movement along each bitmap line, the laser output will be switched on and off according to the value of each pixel. The scanner delays from the applied pen will be respected. White pixels at the beginning and at the end of each bitmap line are used for jumping, within black pixels they are part of the marking with constant marking speed.

10.4.5.1 Gray Advanced

The *Gray Advanced* mode is the optimized mode for marking grayscale bitmaps. The laser power will be modulated according to the grayscale value of each pixel and in synchronization with the scanner movement along each bitmap line.

NOTE

USC-2 or USC-3 card required - not available for USC-1 or RTC cards.

High quality bitmaps can only be achieved if an appropriate laser is used capable of fast enough power switching.

The following configuration steps in the software are necessary to mark a grayscale bitmap in *Gray Advanced* mode.

- **Settings > System > Optic > Advanced (USC-2, USC-3):**
  - o **PixelAM**, **AM Port** and **PixelPWM**: Select the proper option according to your laser to define which laser signal is modulated by the gray levels of the bitmap.

- **Settings > System > Extras:**
  - o **Improved Bitmap Handling**: Activate this checkbox. The improved bitmap modes offer a simplified handling of bitmaps with optimized and precise manipulation of laser signals. Only the state with activated checkbox is described in this manual.
  - o **Dynamic Grayscale Map** and **Interpolation**: Activate this checkbox which modifies the behavior of the System PixelMap and the Pen PixelMaps. For further information about the order of pixel maps power manipulation, see Table 27.

- **Mark property page > Advanced:**
  - o **System PixelMap**: The global System PixelMap is a function which adjusts grayscale values. This means each grayscale value of each bitmap is mapped to an output value. The System PixelMap can be used to compensate general laser non-linearities. In addition, many laser sources should not be switched off completely for white pixels but needs a minimum laser power for fast switching between

power levels.

- **Pen Settings > Main:**
  - o **Distance between dots**: This displayed value is the ratio between the Speed and the Frequency. It defines the laser shot distance in the bitmap line.
  - o **Speed**: This is the speed of the scanner while marking the bitmap lines.
  - o **Frequency**: This is the Laser_A frequency while marking.

- **Pen Settings > Scanner:**
  - o **Jump Delay**: The scanner waits this time after each jump to the next bitmap line.
  - o **Jump Speed**: The scanner moves with this speed during jumps between the bitmap lines.
  - o Note that all other delay parameters (Mark, Poly, Laser On, Laser Off) are not used in *Improved Bitmap Handling* mode *Gray Advanced*.

- **Pen Settings > Bitmap:**
  - o **Bitmap Mode:** This drop down menu allows to select between grayscale and B&W bitmap modes. Select *Gray Advanced*.
  - o **Accel. Ramp:** This adds a start vector for acceleration of the scanner at the beginning of each bitmap line. The length can be calculated by using the mark speed of the pen. The minimum acceleration set by the software is 2 pixel (even if Accel. Ramp = 0).
  - o **Line Shift:** This parameter shifts each bitmap line in marking direction (especially relevant for bidirectional marking). The corresponding length to this time is calculated with the mark speed.
  - o **Power Shift** (PixelAM only): The power modulation is shifted by this time in respect to Laser_Gate. This is useful if the cannot switch power levels fast enough.
  - o **Pen Power as Max Power**:
  - o **Pen PixelMap**: The Pen PixelMap is a function which adjusts grayscale values. This means each grayscale value of each bitmap is mapped to an output value. The Pen PixelMap can be used to optimize the bitmap output for different materials.

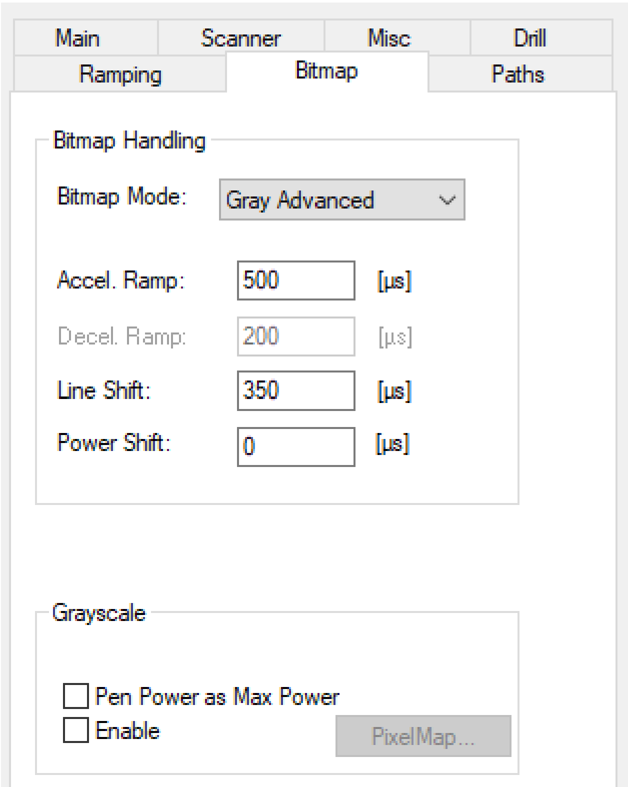


Figure 225: Bitmap tab in Pen

- **File > Import:**
  - o Set 'Files of type' to the desired bitmap format and import a bitmap.

- **Select bitmap:**

o [Mark property page]: Apply the desired pen.

o [Dimension property page]: Scale and position the bitmap.

o [Bitmap property page]: :

- Enable the checkbox **Scanner Bitmap**. Without a scanner bitmap, no marking is possible.
- Set the desired **Dither Step** which defines the spacing of the bitmap lines.
- Select the radio button **Grayscale**.
- Extended > **Bidirectional**: The order of marking the bitmap lines can be unidirectional or bidirectional. Bidirectional is the faster option to mark a bitmap due to minimized jump distances but needs more adjustments.

10.4.5.2 Timing Diagram

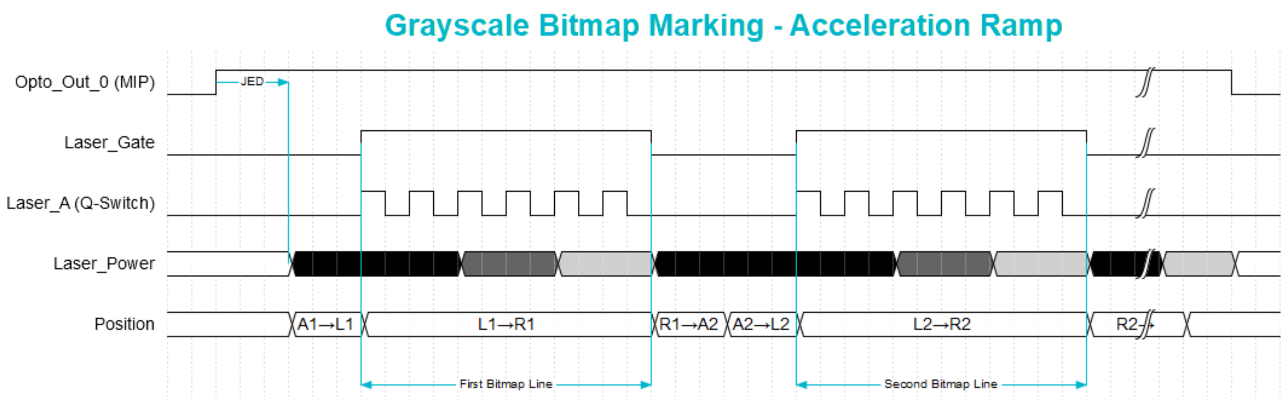


Figure 226: Timing Diagram Grayscale Bitmap

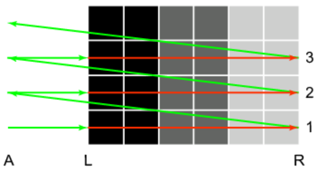


Figure 227: Test bitmap with 6 pixels per line for unidirectional operation from left (L) to right (R) including a ramp starting at A (green - jump, red - mark)

The timing diagram above (Figure 226) shows the laser signals provided by the USC card for grayscale bitmap marking with Gray Advanced mode. This diagram should be seen as a schematic overview for the different signals and parameters which are involved in bitmap marking.
The test bitmap itself consists of lines of six pixels (2 x black, 2x dark gray, 2x light gray, see). Only the first two lines are shown in the timing diagram. Figure 227 shows the relevant positions for the scanner movement: A is the start of the acceleration, L is the left border pixel of the bitmap, R is the right border pixel of the bitmap. The numbers 1, 2 and 3 count the bitmap lines and are used together with A, R and L to connect the timing diagram and the test bitmap. Jumps between positions are displayed in green, a marking (movement of the scanner with simultaneous laser emission) is displayed in red.

The following signals are shown in the timing diagram (Figure 226):

- **Opto_Out_0 (MIP):** The *Mark in Progress* signal stays high during the entire marking process. The *Job Execution Delay* (JED) is the time which is waited after the raising edge of the *MIP* before the first laser power value is set.

- **Laser_Gate:** The *Laser_Gate* signal is activated at the marking start of each bitmap line and inactivated at its end.

- **Laser_A (Q-Switch):** This signal usually triggers single laser shots with the pen frequency.

- **Laser_Power:** Here, the power is shown in the same color as the pixel in the test bitmap. After finishing a bitmap line, the *Laser_Gate* is inactivated and the power of the first pixel of the next line is applied before the jump.

- **Position:** This line in the timing diagram represents the position of the scan head. The scanner movement starts at the beginning of the acceleration ramp A1. From here, the scanner accelerates to the first pixel of the first line L1. Then, the bitmap line is marked and the scanner moves to R1 with constant speed. After finishing the bitmap line, the scanner jumps to the start position of the second line A2. This procedure is repeated until all lines are marked.

The following parameters of the applied pen used to modify the bitmap marking are illustrated in the timing diagrams (Figure 226 and Figure 228):

Acceleration Ramp: In the example timing diagram above (Figure 226), the acceleration ramp is active. The scanner starts at position A1 to accelerate up to the marking speed which is reached at the position of the first pixel L1. With increase of the acceleration ramp parameter, the duration of the ramping and thus the whole bitmap marking is increased. The acceleration ramp will be added for each bitmap line.

Power Shift: The effect of the power shift is shown in the timing diagram below (Figure 228). With increase of the power shift parameter, the change of the power values will be set earlier in respect to the activation of Laser_Gate. This gives the laser more time to switch to the desired power level. The power shift will be added for each bitmap line.

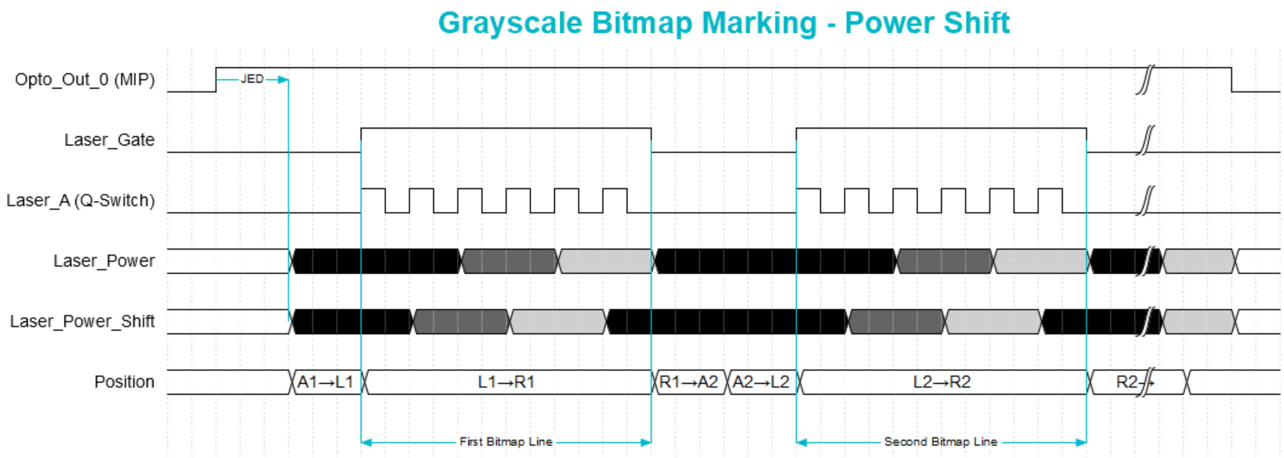


Figure 228: Timing Diagram Grayscale Bitmap - Power Shift

10.5 Serial Number

Generate a Serial Number object by clicking the  button in the toolbar. Then move the mouse to the desired position and press the left mouse button. The serial number property page appears:

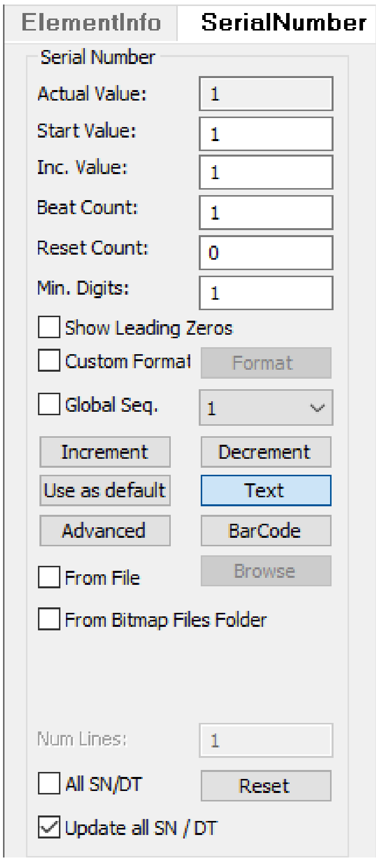


Figure 229: SerialNumber page

Actual Value: Shows the actual value. Can not be edited. It is calculated by:

*ActualValue = StartValue + IncrementValue * RoundDown((SequenceNumber* `Modulo` *ResetCount) / BeatCount)*

Start Value: Value to start with.

Inc. Value: Increment step after each beat.

Beat Count: After number of markings is equal to the beat count the serial number will be incremented. (Even valid if Mark → Selected is used.)

Reset Count: After reset count exposures the serial number will be reset. That means it is set to the start value.

Min. Digits: Minimum number of displayed digits.

Show Leading Zeros: If activated leading zeros are displayed.

Custom Format: If activated an encoded format for serial number can be defined when pressing the Format button.

Format: If the button Text is selected this switches to the Text property sheet. The format of the text can be defined here. If the button BarCode is selected this switches to the BarCode property sheet.

Global Seq.: Global Sequences are serial numbers which go across all jobs. So it is possible to use the same serial number in different jobs. Setting of a global reset time is available in Settings → System → General.

Increment: Manually increments the selected serial number.

Decrement: Manually decrements the selected serial number.

Text: The serial number will be displayed as text.

BarCode: The serial number will be displayed as barcode. Automatisation in flash mode only works for a barcode created as a serial number via this option.

Use as default: Uses the properties of the currently selected serial number object for the generation of new serial numbers. The program saves these settings also for a new program start in case save settings on exit is checked in the general settings.

Advanced: Opens the Serialnumber and Date Time dialog.

From File: When activated, you can use a text or excel file to readout strings for serialization by clicking on the Browse button. See also: Automate Serialization.

From Bitmap File Folder: When activated, you can select a folder with the Browse button. In this folder you can place bitmap files for automatized import. Instead of a serial number, the number of the bitmap file is increased (naming convention is 0001_name.file). The files need to have the same size and resolution, all importable file types are possible. See also: Automate Serialization.

All SN/DT: If enabled, the reset button updates all date time objects and resets all serial numbers in the job. If disabled, only the selected serial number will be reset with click on the reset button.

Reset: Sets the selected serial number to the start value.

Update all SN / DT: This checkbox is global and updates all serial numbers and date time objects in the job. If disabled, serial numbers will not be incremented, date time objects will not be updated.

10.5.1 Serial Number Formats

For the serial numbers the format description is similar to that used in the C-language:

%[flags] [width] [.precision] [optional parameter] type

flags: 0 shows leading zeros

width: Defines the total width of the number including the decimal point. This has only an effect if leading zeros is defined and the width is defined bigger than the width of serial number plus decimal point plus precision, so that leading zeros appear.

precision: Digits after the decimal point.

optional parameter: L or l. L will restrict the serial number to show only 'width' digits and will show leading zeros. l will do the same but will not show leading zeros.

type: f double values

Format examples:

Example	Format	Description
10.000	%6.3f	3 digits after the decimal point
10	%6.0f	0 digits after the decimal point
000010	%06.0f	show leading zeros
012	%03.0Lf	restrict to 3 digits and show leading zeros
12	%03.0lf	restrict to 3 digits and do not show leading zeros

Figure 230: Formating examples

NOTE

Format code and text can be entered simultaneously.

The encoding will only work if Custom Format is selected inside the SerialNumber property page.

10.5.2 Serial Number as Barcode

If the Serial Number is displayed as a Barcode it is possible to reference other text elements of the same job and include their contents into the current serial number. This feature is useful especially when the serial number barcode has to contain encoded information that are available as human-readable text within the same job too. To include the data of another text entity this object must first have an entity name. As a second step that name needs to be referenced in format "<$entity_name>" within the custom format field of the serial number barcode. Here "<$" and ">" are delimiters for that part of the serial number format that has to be replaced dynamically. If the name between these delimiters is not defined within the job no replacement is done and the full placeholder is displayed.

As an example: There is a text object named "TText" within the job that contains the text "My Information". Within the serial number barcode the following custom Format is defined: "%1.0f / <$TText>". Resulting from that the serial number barcode object would display the current serial count plus the text " / My Information":

- "1 / MyInformation"
- "2 / MyInformation"

- "3 / MyInformation"
- "4 / MyInformation"
- "5 / MyInformation"
- ...

The Main Window will show the Entity List, the View2D and the Entity Property Sheet like in the following pictures:

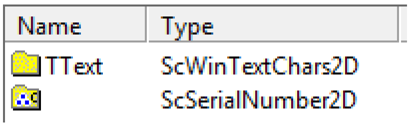


Figure 231: Entity List with Text object and Serial Number Barcode

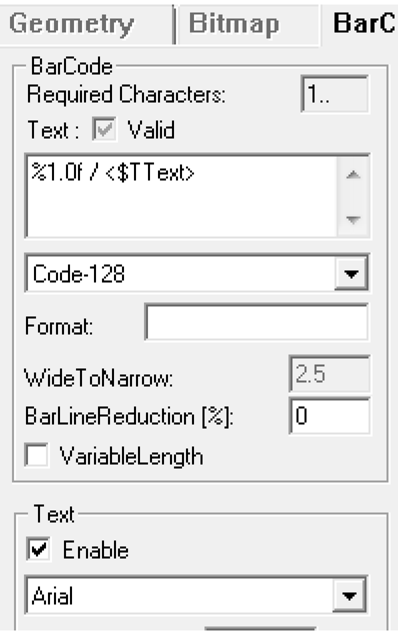


Figure 232: Property Sheet with Serial Number Barcode

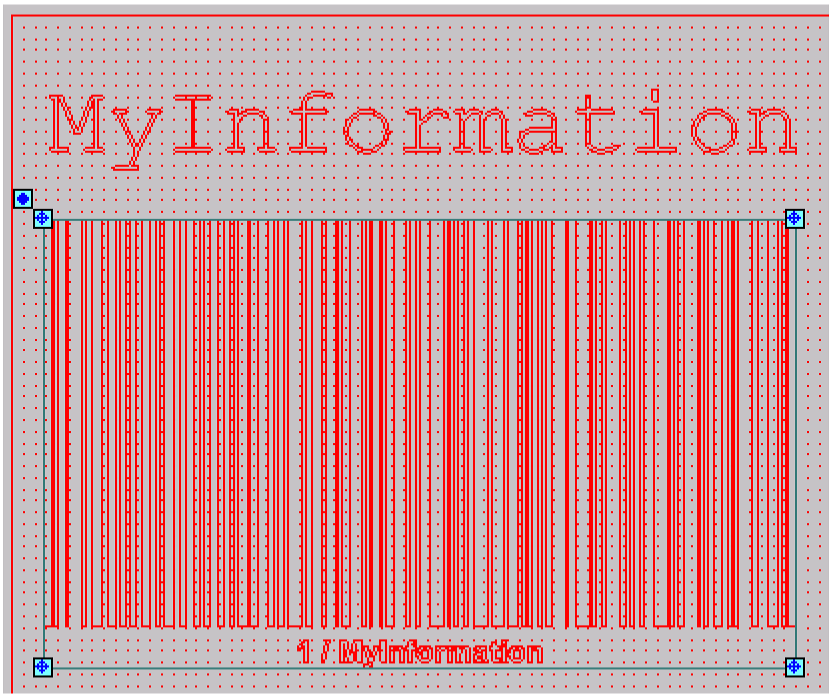


Figure 233: View2D with Text to be included and Serial Number Barcode

NOTE

Serial numbers are updated once per marking process. If such a serial number barcode references to text entities of the job using the <$entity_name> syntax it will contain the information of the element named "entity_name" that is visible at that specific moment when the update is performed. If the entity with that name is changed after the serial number was updated, the contents of the serial number barcode are not updated automatically. This means the that the serial number barcodes will stay with the old value until the next marking cycle was finished or until the sequence was updated from the menu. The same is true if such a serial number barcode object is newly created: it will not show the contents of any referenced object until the sequence was updated.

10.5.3 Serial Number Advanced

The following dialog can be reached by clicking on the Advanced button in the entity property sheet of the serial number.

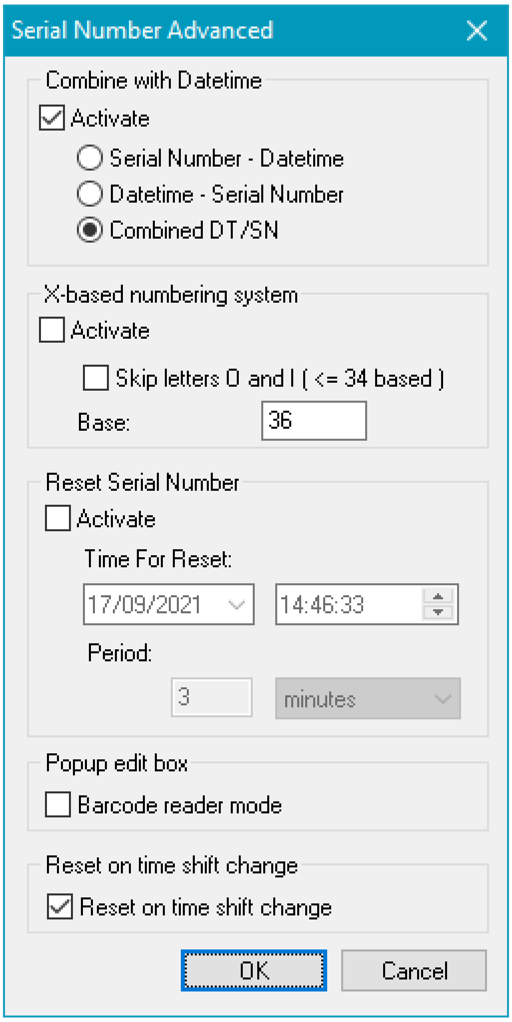


Figure 234: Serial Number Advanced Dialog

Combine with Datetime:

Activate: Adds a DateTime element to the Serialnumber. (See Serial Number and Date Time)

Check: The combined string exists of Serialnumber + Date Time or vice versa.

X-based numbering system:

Activate: Allows to define a user defined base for the display of the Serialnumber.

Base: Base of numbering system, accepts values from 2 up to 36. Base 2 means binary, base 10 means decimal system and for a hexadecimal system the base is 16.

Reset Date Time: If activated the serial number will be reset after a defined amount of time. The first reset will be at the specified Reset Start Time. The following resets will be repeated after the defined Period. This reset does only affect the serial number for which it is defined.

Popup edit box: The Barcode reader mode can be activated. Please refer to "Barcode Reader".

Reset on time shift change: The serial number is reset whenever a shift defined in the Shift Map changes. This reset does only affect the serial number for which it is defined.

NOTE

The format of each element, the Serialnumber and Date Time, can be edited afterwards. Therefore Custom Format in the Serialnumber property page needs to be checked. A combined Serialnumber and Date Time element can also be displayed as a barcode. However, not each Date Time format can be converted to a adequate barcode. For example the signs ".", ":" and "/" are not taken!

The digit definitions of Customer Format are not taken for a x-based numbering system.

10.5.4 Serial Number and Date Time

Combined customizable Serial Number and Date Time are only available beginning from the 3_7_5_20180824 SAMLight version.

If the checkbox *Activate Combine with Datetime* under Serial Number Advanced is active, you have a possibility to mark the serial number and the date time simultaneously. In the same window choose the option *Combined DT/SN* and in the Date Time property page select Edit Format. In the appearing *DateTime Advanced* window write the code for the Date Time and Serial Number you would like to get, but writing the serial number put @ instead of % (Example: %06.0f --> @06.0f). It may for example look like (see the screenshot below):

@06.0f %y\%m\%d %H:%M

- where @06.0f denotes the serial number format
- and %y\%m\%d %H:%M denotes the date time format

Format codes both for Date Time and Serial Number you will find under corresponding pages.

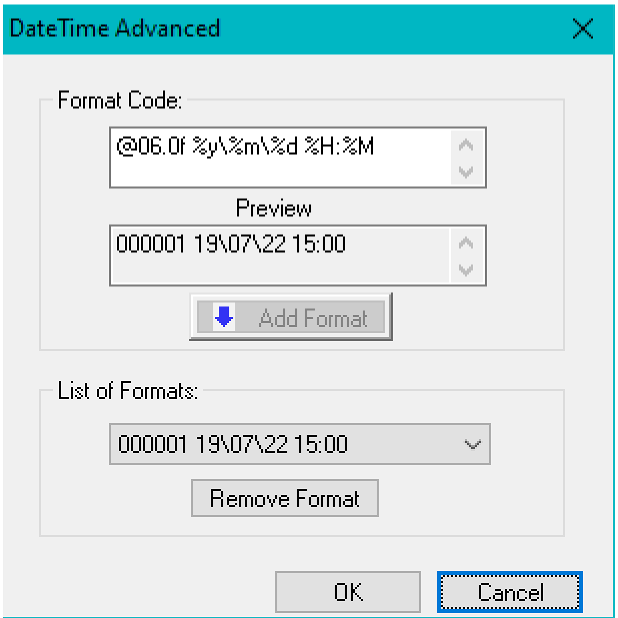


Figure 235: Date Time Advanced Format Code

Example: How to create a barcode consisting of a variable text (serial number object) and a date time object (as <text> Month\Day\Hour):

1. A text object with the variable text, <name> it in the EntityInfo and set it as Nonmarkable Entity in the entity list.

2. A serial number and activate the date time format under Advanced (Combined DT/SN).

3. In the DateTime tab, go to Edit Format and enter <$name>%m\%d\%H. ($ is referencing the text entity, %m is month as decimal number 01-12, %d is day as decimal number 01-31 and %H is hour in 24 hour format 00-23)

4. In the serial number tab, click on barcode.

The barcode can be adjusted according to the needs. When the text in the barcode needs changing, simply change the text entity.

10.5.5 Automate Serialization

The content of a serial number can be assigned from an external file or via a bitmap file folder. The supported file types are: *.txt, *.csv, *.xls and *.xlsx. Some files will be read line by line. In some file types the column can be chosen, so that not the complete line is read. The user can choose either to load the content from a file or reimport bitmaps. If one flag is checked the *Browse* button gets enabled.

If the serial number is incremented, the last number is not incremented classically, since the next line or bitmap is used as new serial number. If the user uses bitmaps, the naming convention *0001_name.file* has to be taken into account. The bitmaps have to be equal in size and resolution.

NOTE

One file can get assigned to more than one serial number.

Chapter Serial Number for the general properties of serial numbers which are also available when using the ASCII Serialization.

NOTE

The complete file content will be loaded when the file has been selected. After each finished marking the time stamp of the file will be checked and in the case of any update of the given file the new content is available. The load procedure can take up to one minute.

In the case that the file is not longer stored at the defined path SAMLight will mark each line like loaded at the beginning. There will be no error message. If the file is back at the defined path, SAMLight will recognize this.

SAMLight will show an error message if a job with a set path is opened, where the file can not be found at the set path.

The following dialog is opened if the *Browse* button is clicked:

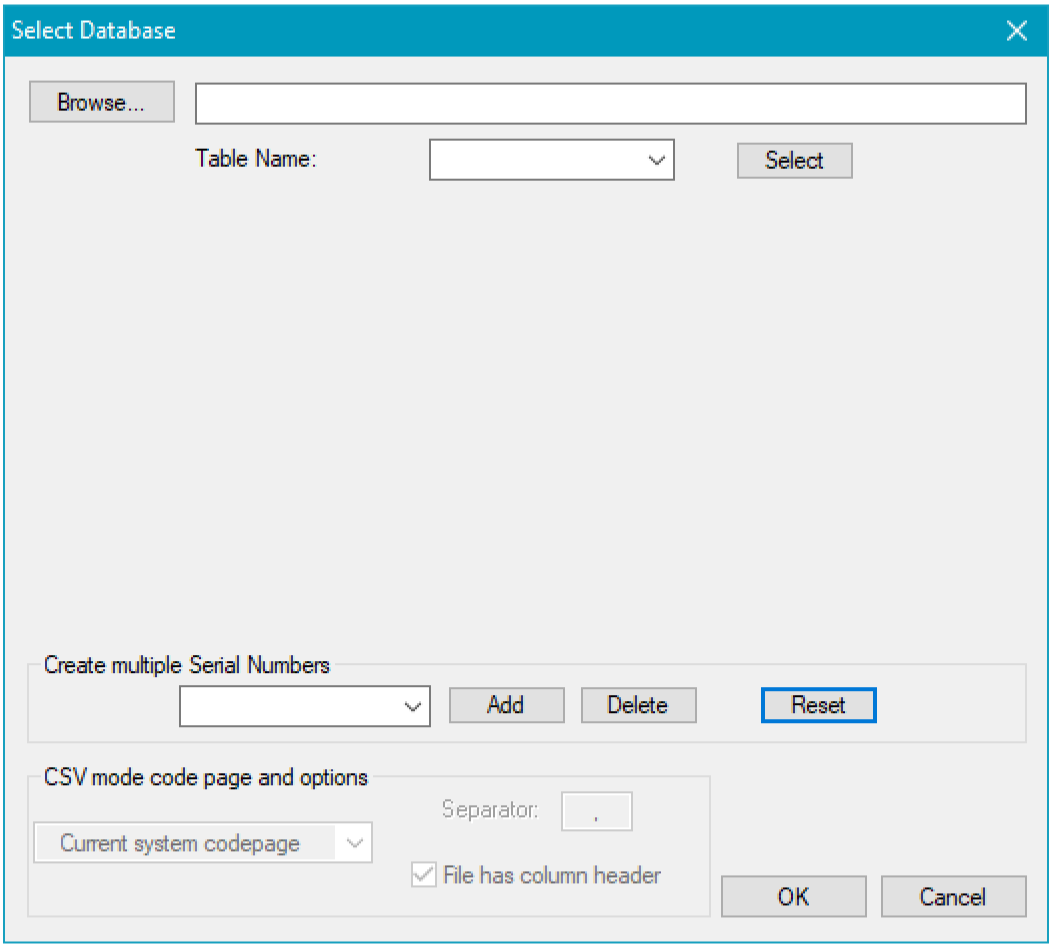


Figure 236: Automate serialization settings

10.5.5.1 ASCII File

A serial can be defined in an ASCII file. Each line in the file matches to one serial number naming. The entries are indexed from 1 on. After an increment the next line of the ASCII File gets set to the assigned serial number. If a Pause Identifier is defined in Settings General the Pause Identifier string will not be assigned to a serial number object but cause a break of marking.

See chapter Automate Serialization for how to assign a file to a serial number.

10.5.5.2 Excel Table

This chapter explains how to assign an excel file (*.xls or *.xlsx) to a serial number. After pressing the select button in the serial number property page a following dialog appears.

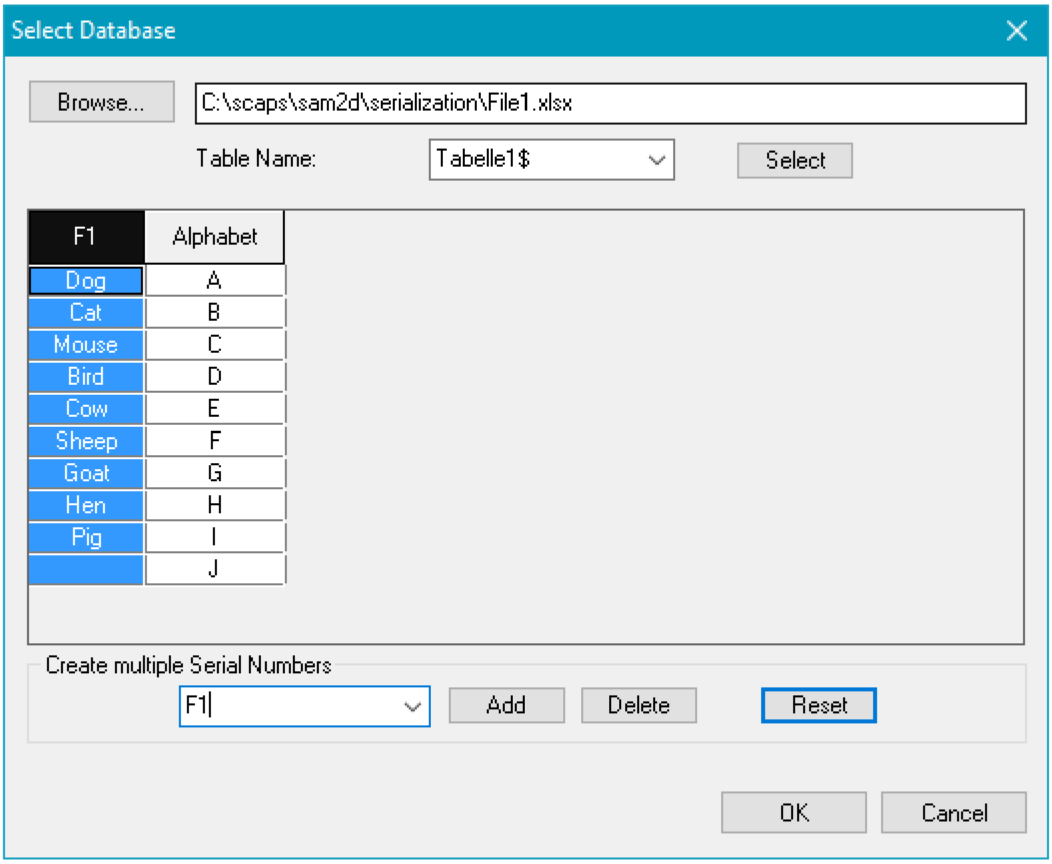


Figure 237: Select File for Serial Number Dialog

Browse: A dialog appears to select a file.

After selecting an excel file the first ten rows of table are being shown. The first row of the table file is taken as a caption of the serial and is not assigned to the serial number. If there is no entry in the first row an automatic naming is taken instead like "F1". The column which is being marked will get assigned to the selected serial number after pressing OK.

NOTE

If the excel file is being changed, the file needs to be reassigned to the serial number.

NOTE

There are several cases which cause an error message:
no Microsoft Excel 2007 or later installed, Microsoft Excel 64 bit 2007 or later installed or Microsoft Excel 32 bit 2007 or later installed and in additional other 64 bit programs from Microsoft installed
*Please download the **32 bit** version of the patch: Microsoft access Database Engine Redistributable and run it with the **commandline parameter /quiet**.*

Create multiple Serial Numbers: To create additional serial numbers add the according table head strings into the combo list below. Therefore select a column and press Add. After confirming with OK the serial numbers is assigned to the terms of the combo list. If the combo box is empty the current selected column item will be assigned to the serial number.

10.5.5.3 CSV Files

This chapter explains how to assign an Comma Separated Values File (*.csv) to a serial number. After pressing the select button in the serial number property page a following dialog appears.

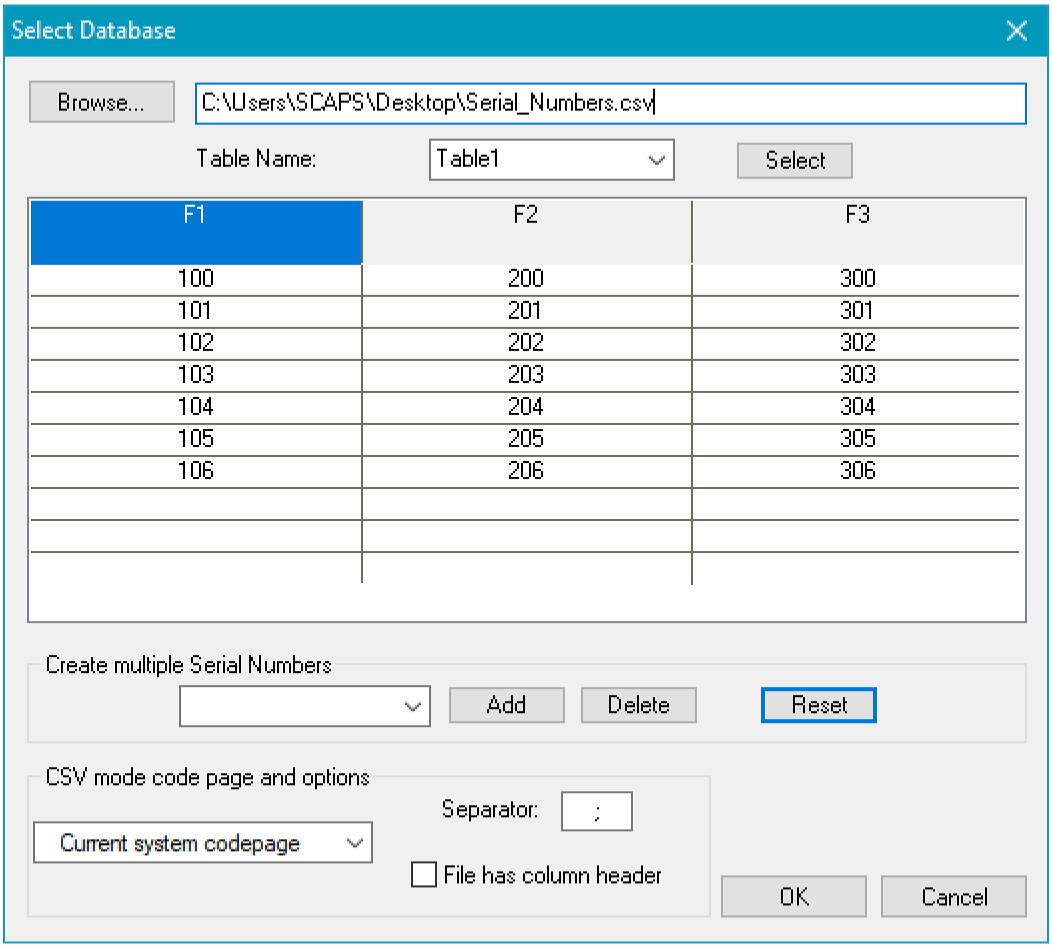


Figure 238: Select File for Serial Number Dialog

Browse: A dialog appears to select a file.

After selecting an CSV file the first seven rows of table are being shown. The first row of the table file is taken as a caption of the serial and is not assigned to the serial number. If there is no entry in the first row an automatic naming is taken instead like "F1". The column which is being marked will get assigned to the selected serial number after pressing OK.

Separator: you can choose the separator for your CSV file. Here we choose the semicolon.

File has Column Header: if checked, the serial number will start from the second line and the first line will be taken as a header.

CSV mode code page and options: choose the code page which is corresponding to the language you are using in the CSV file. There is no need to specify the code page, if you only use numbers.

NOTE

If the CSV file is being changed, the file needs to be reassigned to the serial number.

Create multiple Serial Numbers: To create additional serial numbers add the according table head strings into the combo list below. Therefore select a column and press Add. After confirming with OK the serial numbers is assigned to the terms of the combo list. If the combo box is empty the current selected column item will be assigned to the serial number.

10.5.5.4 Example

The following explains how to control serialization with the help of an ASCII file.

Assumption: There are 3 pens to be marked at the time.

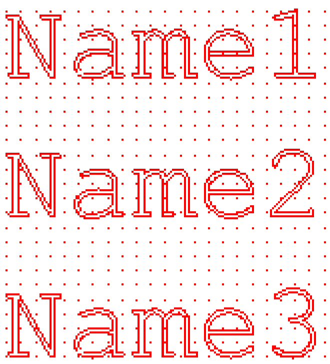
The names that are assigned to the pens needs to be saved in a text or excel file, for example:

- Name1
- Name2
- Name3
- Name4
- Name5
- Name6
- Name7
- Name8
- Name9
- Name10
- ...

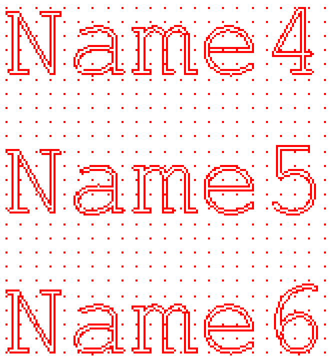
Now 3 serial numbers need to be created and the ASCII file has to be assigned to each of them. For the correct mapping of the serial string to the serial entity the following setup is used. See also: Chapter Serial Number.

	Start Value	Inc Value
SerialNUM1 :	1	3
SerialNUM2 :	2	3
SerialNUM3 :	3	3

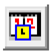
Result:



That means the serial number itself is used as an index into the ASCII file. SerialNUM1 starts at index 1 and will be incremented by 3 after every mark. The corresponding settings for the another 2 serial numbers lead to the next mark as shown next:



10.6 Date Time

Generate a date and time object by pressing the DateTime button  in the object toolbar. Move the mouse to the desired position and press the left mouse button.

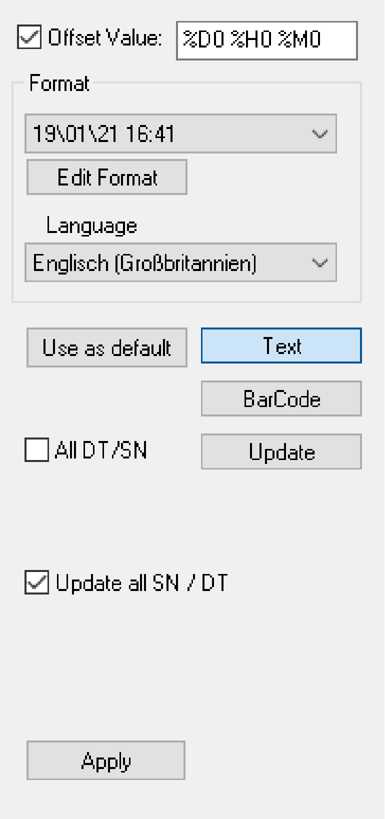


Figure 239: Date Time Dialog

Offset Value: Offset values for day, hour and minutes. The values can be negative or positive.

Format:

Drop down list with the currently available date time formats.

Edit Format: Allows to edit the date time format list. See subchapter Format.

Language: Drop down list with all available languages. Some format specifier depend on the selected language. Per default, the Windows system language will be taken.

Use as default: Uses the properties of the currently selected date time object for the generation of new date time objects. The program saves these settings also for a new program start in case save settings on exit is selected in the general settings.

Text: Generates the date object as text.

BarCode: Generates the date object as barcode and opens the BarCode property page.

All DT/SN: If enabled, the update button refreshes all date time objects and resets all serial numbers in the job. If disabled, only the selected date time object will be updated with click on the update button.

Update: Refreshes the selected date time object to the current Windows time.

Update all SN / DT: This checkbox is global and updates all serial numbers and date time objects in the job. If disabled, serial numbers will not be incremented, date time objects will not be updated.

10.6.1 Date Time Format

Press the *Edit Format*... button in the *DateTime* Property page to show this dialog.

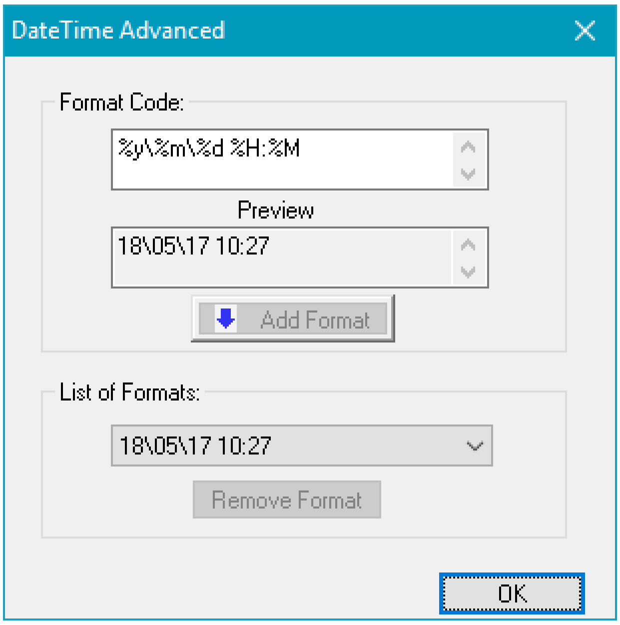


Figure 240: Date Time Advanced Dialog

Code: Enter an encoded format definition for date time. The standard formats are not editable.

Convert: Pressing the arrow button converts the format definition given in Code and extends the date time format list.

Format: List of defined date time formats.

New: Creates an empty format. It is editable in the Code field.

Delete: Deletes current format from format list. The standard formats are not deletable.

Cancel: Leaves *DateTime* Advanced without taking changes.

OK: Changed list is valid. It will be shown in the *DateTime* property page also after a new start of the program.

Format definitions:

Format	Description
%1d	Day of the month as digits without leading zeros for single-digit days.
%2d	Day of the month as digits with leading zeros for single-digit days.
%3d	Abbreviated day of the week as specified by a LOCALE_SABBREVDAYNAME* value, for example, "Mon" in English (United States). Windows Vista and later: If a short version of the day of the week is required, your application should use the LOCALE_SSHORTESTDAYNAME* constants.
%4d	Day of the week as specified by a LOCALE_SDAYNAME* value.
%1M	Month as digits without leading zeros for single-digit months.
%2M	Month as digits with leading zeros for single-digit months.
%3M	Abbreviated month as specified by a LOCALE_SABBREVMONTHNAME* value, for example, "Nov" in English (United States).
%4M	Month as specified by a LOCALE_SMONTHNAME* value, for example, "November" for English (United States), and "Noviembre" for Spanish (Spain).

Format	Description
%1y	Year represented only by the last digit.
%2y	Year represented only by the last two digits. A leading zero is added for single-digit years.
%4y	Year represented by a full four or five digits, depending on the calendar used. Thai Buddhist and Korean calendars have five-digit years. The %4y pattern shows five digits for these two calendars, and four digits for all other supported calendars. Calendars that have single-digit or two-digit years, such as for the Japanese Emperor era, are represented differently. A single-digit year is represented with a leading zero, for example, "03". A two-digit year is represented with two digits, for example, "13". No additional leading zeros are displayed.
%1g %2g	Period/era string formatted as specified by the CAL_SERASTRING value. The %1g and %2g format pictures in a date string are ignored if there is no associated era or period string.
%a	Abbreviated weekday name [a]
%A	Full weekday name [a]
%b	Abbreviated month name [a]
%B	Full month name [a]
%c	Date and time representation appropriate for locale [a]
%C	Month as character digit (A-L)
%d	Day of month as decimal number (01 – 31)
%H	Hour in 24-hour format (00 – 23)
%I	Hour in 12-hour format (01 – 12)
%j	Day of year as decimal number (001 – 366)
%k	Weekday as decimal number (1 - 7; Sunday is 7)
%K	Weekday as decimal number (1 - 7; Sunday is 1)
%L %2L %3L %4L	Month mapping Placeholder, see Months Map. There are 4 month maps available. To address the first, use %L, to address the second use %2L, to address the third and forth use %3L and %4L, respectively.
%l %2l %3l %4l	Day mapping Placeholder, see Day Map. There are 4 day maps available. To address the first, use %l, to address the second use %2l, to address the third and forth use %3l and %4l, respectively.
%m	Month as decimal number (01 – 12)
%M	Minute as decimal number (00 – 59)
%o %2o %3o %4o	Year mapping placeholder, see Year Map. There are 4 year maps available. To address the first, use %o, to address the second use %2o, to address the third and forth use %3o and %4o, respectively.
%O	Year as a single character representation, ASCII character 'H' represents Year 2000. After 'Z', the next year starts with the character 'A' again (starting with SAMLight Version 3_8_5_006_20190117).
%p	Current locale's A.M./P.M. indicator for 12-hour clock [a]
%q	Week of year as decimal number, with Monday as first day of week (01 – 53) (also % W)
%Q	Week of year as decimal number, with Sunday as first day of week (01 – 53) (also % U)
%r, %R	Year as a single decimal number representation (0 - 9; Eg. year 2008 is 8)

Format	Description
%S	Second as decimal number (00 – 59)
%T %2T %3T %4T	Working Shift Placeholder, see chapter Shift Map. There are 4 shift maps available. To address the first, use %T, to address the second use %2T, to address the third and forth use %3T and %4T, respectively.
%v	Hour to letter representation ('A' - 'Z'; 0:00 h is 'A')´
%w	Weekday as decimal number (0 – 6; Sunday is 0)
%x	Date representation for current locale [a]
%X	Time representation for current locale [a]
%*	Year without century, as decimal number (00 – 99) and corresponding week of year as decimal number
%y	Year without century, as decimal number (00 – 99)
%Y	Year with century, as decimal number
%z, %Z	Time zone name or abbreviation; no characters if time zone is unknown
%%	Percent sign

Table 28: Format definitions for Date Time objects

[a]: If If no language is selected in the drop down list "Language" of the Date Time property page, the language of the Windows system is taken. For specifier which do not use any Windows language, this drop down menu is not available.

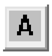
Format	Description
%#a, %#A, %#b, %#B, %#p, %#X, %#z, %#Z, %#%	# flag is ignored.
%#c	Long date and time representation, appropriate for current locale. For example: "Tuesday, March 14, 1995, 12:41:29".
%#x	Long date representation, appropriate for current locale. For example: "Tuesday, March 14, 1995".
%#d, %#H, %#I, %#j, %#m, %#M, %#S, %#U, %#w, %#W, %#y, %#Y	Remove leading zeros (if any).

Table 29: Special format definitions for Date Time objects

Figure 241: Example of Persian DateTime object

The Persian date 1394/07/19 is created with identifiers %4y/%2M/%2d. *"Text2D → Extended... → Decimal number substitution → Persian digit number substitution"* must be enabled to change the figures.

10.7 Text2D

Generate a text object by pressing the  Text button in the Object-Toolbar. Then move the mouse to the desired position and press the left mouse button. The text property page appears:

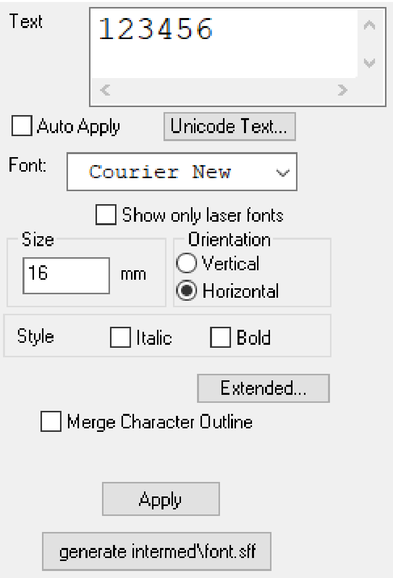


Figure 242: Text2D Dialog

Text: Input field for the text that is generated.

Auto Apply: If activated, a selected font will directly be applied to the text in the text window and in the View2D without the need to push Apply.

Unicode Text...: This will open a dialog box where special characters can be entered. For some languages this is needed when it is not possible to enter the character in the Text edit box.

Font: List with all available True Type fonts.

NOTE

Fonts that begin with @ support a vertical writing for the Asian languages Chinese, Japanese and Korean, in which the characters are rotated 90° counterclockwise (the checkbox “DirectWrite character generation” and for Korean also “Generate single characters” must be deactivated in the Text2D properties dialog). The same font without @ only supports a horizontal writing.

Show only laser fonts: Only true type fonts generated with the sc_font_convert tool are shown in the Font List. These fonts are special true type fonts and the text generator will generate simple line characters for a fast marking processes. For more detailed information and how to generate your own simple fonts see Generate Simple Fonts.

Size: SAMLight uses the *Cell Height (Ascent + Descent)* to scale the font size. The *Em Height (Ascent + Descent - Internal Leading)* or also called character height is often used in other text programs to scale the font size.

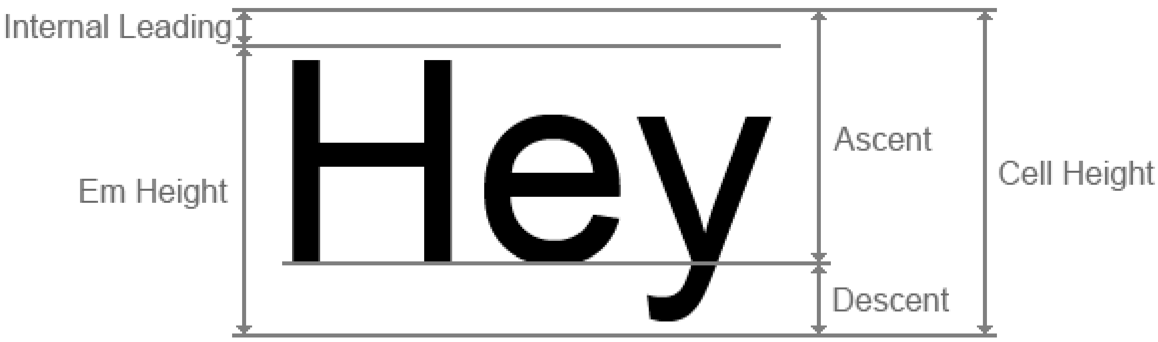


Figure 243: SAMLight uses the Cell Height to scale the font size.

Orientation: Horizontal or vertical text.

Style: Italic or Bold characters. Bold characters are not available for Simple Fonts.

Extended...: Opens the Text2D Properties Dialog for more features.

Merge Character Outline: If text characters within one ScWinTextChars2D entity intersect, the outlines of these characters will be merged. See Fig. 244 for an example: on the left hand side, Merge Character Outline is enabled, on the left hand side it is not. Merging of character outlines also works automatically for characters in serial numbers.

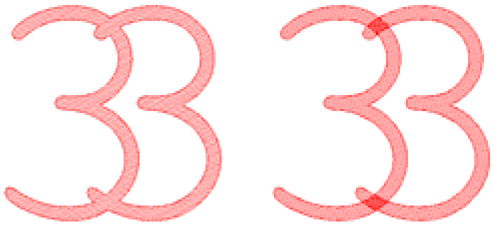


Figure 244: Example for Merge Character Outline.

10.7.1 Text2D Properties

Clicking the *Extended...* button in the Text2D property page the following dialog appears:

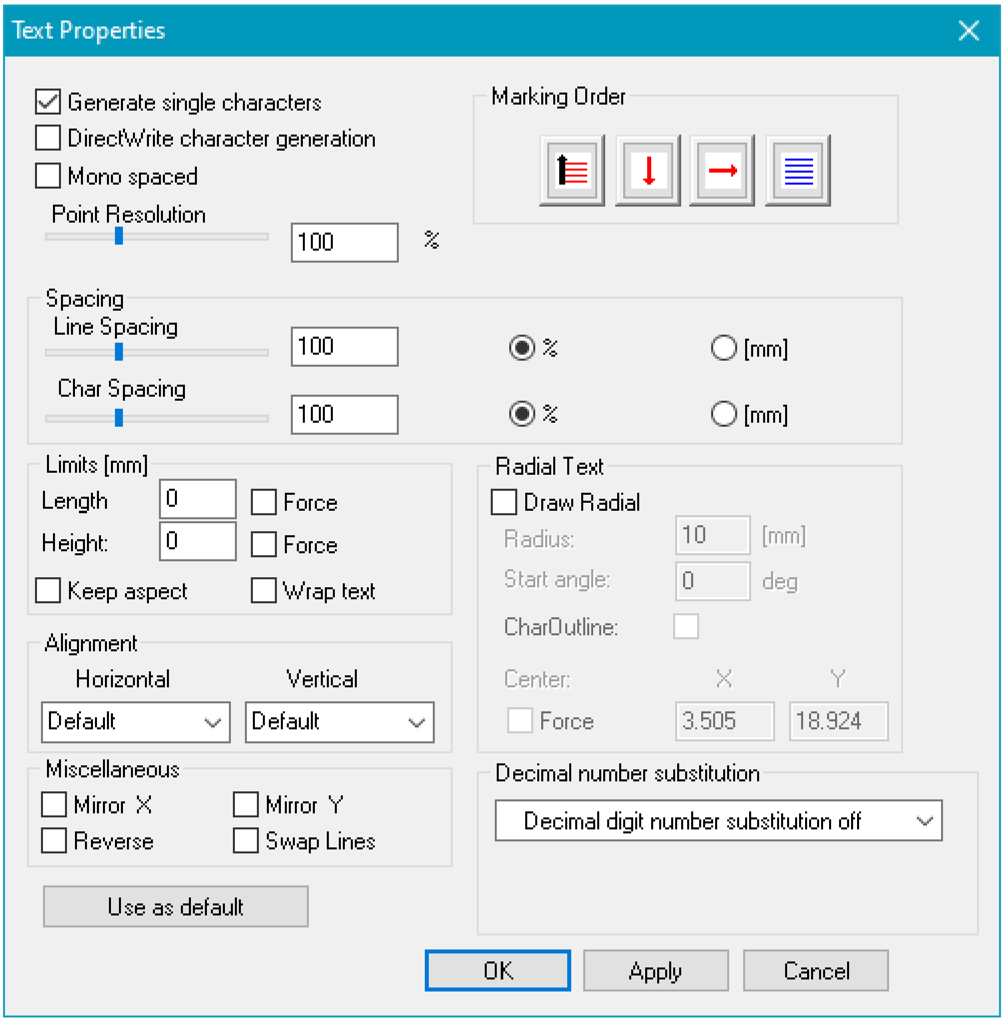


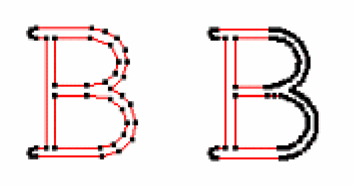
Figure 245: Text Properties Dialog

Generate single characters: When this option is selected all characters in the text string are generated separately, which means that each character is stored in one separate ScWinText2D object. This has the advantage that each character can be accessed and by this way each character can be hatched with a different hatch style for example. If this is not necessary the option should be switched off to fasten operations like transformation and rendering.

DirectWrite character generation: Use the later DirectWrite Windows API instead of the older Uniscribe Windows API for text layout and glyph rendering. It is used, e.g. to display an international language correctly in SAMLight. Alternatively, the command line parameter /UseDirectWrite can be used.

Mono spaced: If selected, the distance between two characters is constant. The center of each letter is spaced equally to its neighbor's center.

Point Resolution: The resolution of the lines in percent. Not available for Simple Fonts.



30 % **100 %**

Line Spacing: Spacing between the lines in percent or in SAMLight unit. The cursor can only be used for % and has a predefined range of 1 to 300. Higher values can be given in the parameter box. Use the radio button to change between % or SAMlight unit.

Char Spacing: Spacing between the single characters in percent or in SAMLight unit. The cursor can only be used for % and has a predefined range of 1 to 300. Higher values can be given in the parameter box. Use the radio button to change between % or SAMlight unit.

Limits [mm]:

Length: Defines maximum Length. If Force is checked the text is adapted to the entered Length. If Radial Text is selected, those set values will be set to 0 and ignored.

Height: Defines maximum Height. If Force is checked the text is adapted to the entered Height. If Radial Text is selected, those set values will be set to 0 and ignored.

Angle: Only available for radial text with enabled "CharOutline". Defines a maximum range of angle to limit the radial text. Text bigger than this angle will be scaled in x direction (width) while the height is untouched.

Keep aspect: Keep aspect limits meaning that the height of the text is reduced together with the width (x and y direction).

Wrap text: Enables text in multiple lines.

Alignment Horizontal: Here you can choose different modes for the alignment setup in horizontal direction. In the following examples vertical is set to default. Each image below contains a blue reference line and a text string showing the selected mode. After changing the mode the positioning of the entity could change.

- Default: Text will be aligned on the left side. There is a small gap between the reference line (blue) and the first char. The reason for this is that the LSB of the first character is located here. This will be ignored for the mode *Left.*
- Right: Text will be aligned on the left side. The string will always end at the reference line. If the string length is increased, the dimension will be increased on the left side.
- Left: Text will be aligned on the left side. This mode is equal to *Default*, with the change that the text will be begin at the reference line.
- Center: The string will be centered, but each new line will be aligned on the left. If the string is larger, the dimension will be equally sized to the right and to the left.
- LineRight: All lines will be aligned on the right side.
- LineLeft: Mode similar to *Left.*
- LineCenter: All lines will be centered. Alignment of each line is now all centered.

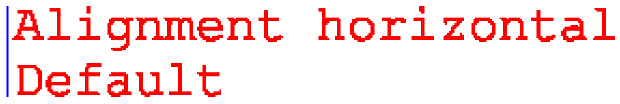


Figure 246: Alignment horizontal Default

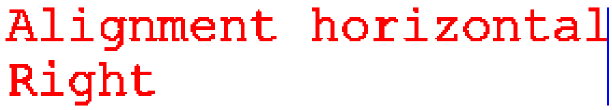


Figure 247: Alignment horizontal Right

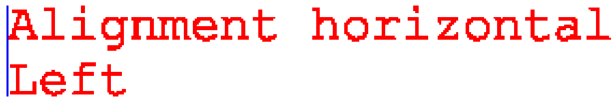


Figure 248: Alignment horizontal Left

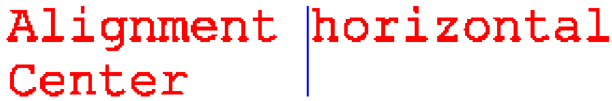


Figure 249: Alignment horizontal Center

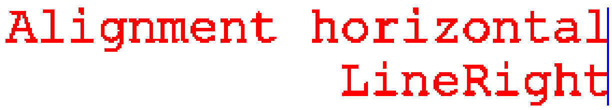


Figure 250: Alignment horizontal LineRight

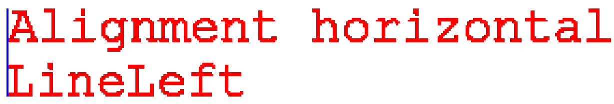


Figure 251: Alignment horizontal LineLeft

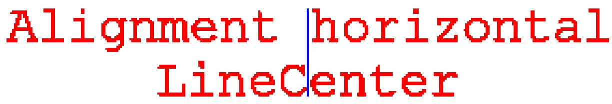


Figure 252: Alignment horizontal LineCenter

Alignment Vertical: Here you can choose different modes for the alignment setup in vertical direction. In the following examples horizontal is set to default. Each image below contains a blue reference line and a text string showing the selected mode. After changing the mode the positioning of the entity could change.

- Default: New text will be added at the lower side. This mode contains a small gap between the reference line and the first line. This distance comes form the font definition and will be ignored for *Top*.
- Bottom: The latest line is on the bottom and new lines will be added on top.
- Top: New text will be added at the lower side. The text begins at the reference line.
- Center: The text will be spaced in y-direction equally around the center.

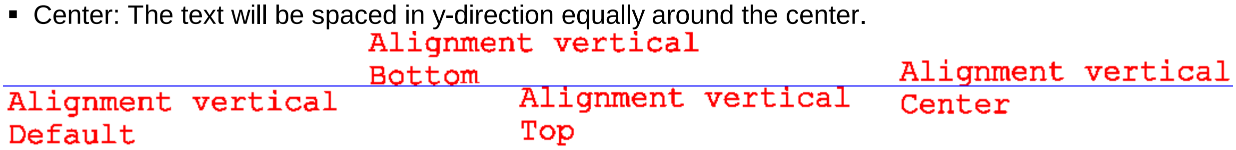


Figure 253: Alignment vertical Default, Bottom, Top and Center

Miscellaneous: Mirror X, Mirror Y, Reverse and Swap Lines can be used for the text formation.

Use as default: Uses the properties of the currently selected text object for the generation of new text. The program saves these settings also for a new program start in case save settings on exit is checked in the general settings.

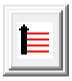
Radial Text:

Draw Radial: Activating radial mode. Radius [mm] and Start angle [deg] define the position of the characters.

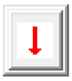
CharOutline: Takes the outlines of character into account when they get disposed radial. Active if radial mode is selected.

Center: Here the initial center coordinates of a radial text can be specified. If this option is selected the center of the radial text is positioned once at the coordinates specified with X and Y.

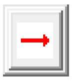
Marking order: The buttons are only active if generation of single characters is checked. The state of the buttons is changeable by clicking.



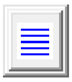
Sets the main direction of the marking order of characters. If case y is selected as main direction the characters get sorted line by line, if case x is selected as main direction they get sorted column by column.



Sets the orientation of the y direction.



Sets the orientation of the x direction.



With this state an unidirectional sort is defined. Also bidirectional is selectable, which results in a zigzag sorting order.

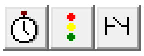
10.8 Control objects

A control object can be defined in between the chronological process of the job. Provided objects are:

- ScTimer
- ScWaitForInput
- ScSetOutput
- ScSetAnalogOutput
- ScExecutable
- ScMotionControl
- ScMotfOffset
- ScWaitForTrigger
- ScAutoCalib
- ScOverride
- ScJump

10.8.1 I/O Control Objects

There are three different control objects available.

 The control objects *ScTimer*, *ScWaitForInput* and *ScSetOutput* can be accessed in the toolbar. When a control object is created it appears in the entity list.

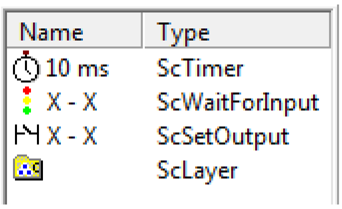


Figure 254: Entity list with I/O Control Obejects

To change the properties of a control object, it has to be selected by clicking on it in the entity list. Then the Control property page can be activated and the properties can be changed.

10.8.1.1 ScTimer

ScTimer:

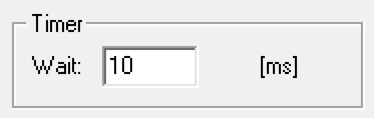


Figure 255: ScTimer

Wait: Interrupts the marking process for n ms, in this example 10 ms.

ScTimer will be conducted according to its position in Entity List. Therefore in multi-head system, ScTimer cannot be assigned to a specific head.

10.8.1.2 ScWaitForInput

ScWaitForInput:

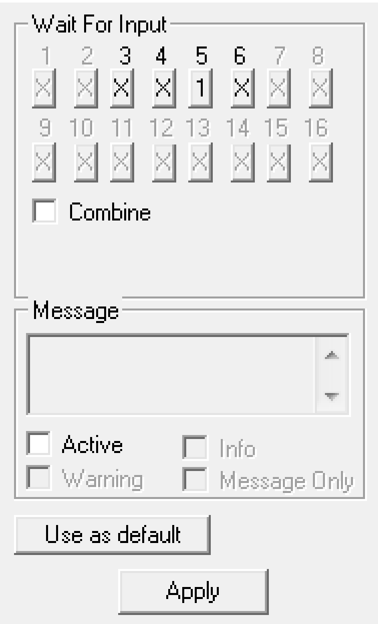


Figure 256: Wait For Input

Wait For Input: Stops the marking process until the specified input bit(s) of the IO port is set to high or low.

X: Do not care / Ignore input bit.
0: Wait for corresponding *bit state 0*.
1: Wait for corresponding *bit state 1*.

Combine: Allows to select multiple input bits. All need to be in the selected state to proceed.

Bit position count starts with "1", but corresponds to bit "0" at the hardware! Find Bit to Pin assignment in [Global Settings → IO]

Message: If the check button *Active* is selected a message box appears containing the text defined in *Message* when the specified input bit(s) is(are) *high*/*low*. The marking process continues after the message box has been replied to.

It is also possible to set no Input, but a message appears. In this case *Active* and *Message Only* has to be selected.

Additionally an info or warning sign can be added to the message box.

The message box only appears, if any text is typed in.

Default name of the control is for current shown state: '5 - H' (masked bit 5, wait for bit high state (H))

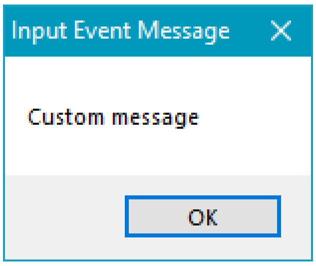


Figure 257: Message Active

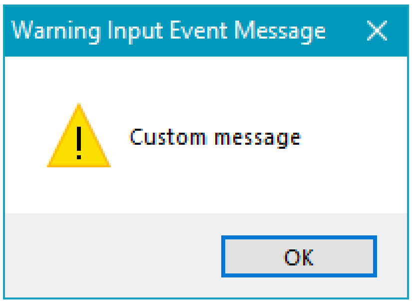


Figure 258: Message Active + Warning

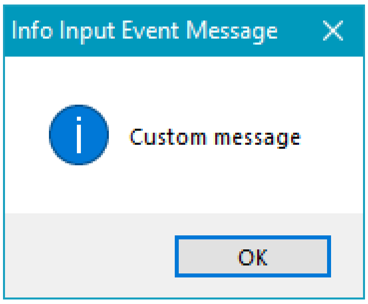


Figure 259: Message Active + Info

10.8.1.3 ScSetOutput

ScSetOutput:

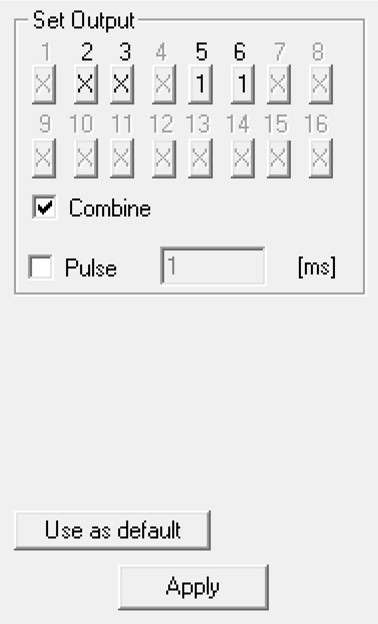


Figure 260: Set Output

Set Output: Sets the specified output bit of the IO port to high/low.

X: Do not care / Ignore input bit.
0: Wait for corresponding *bit state 0*.
1: Wait for corresponding *bit state 1*.

Combine: Allows to set multiple output bits at the same time.

NOTE *Bit position count starts with "1", but corresponds to bit "0" at the hardware! Find Bit to Pin assignment in Global Settings → IO*

Pulse: If selected the output bit gets set to its previous state again, n ms after the bit was set.

Default name of the control is for current shown state: '0036 - 0034' (mask is hexadecimal 0036, output state for this mask is hexadecimal 0034)

10.8.2 Analog Output Control Object

With this control object you can set the Analog Output Value of the DAC A or DAC B to the desired value:

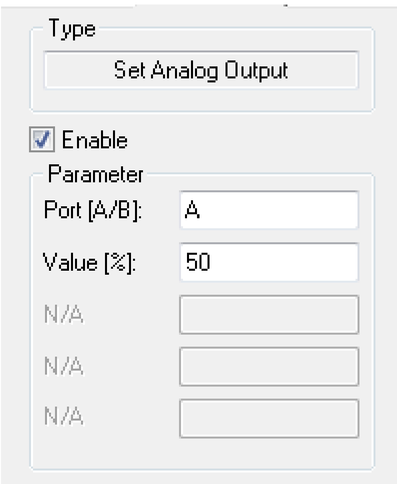


Figure 261: Set Analog Output

Enable: Enables the control.

Parameter:

Port [A/B]: Choose port A or B for control

Value [%]: Define output value in %

10.8.3 Executable Control Object

An executable control object  can be used to start a program at a defined position within the job. Therefore click on the icon in the toolbar. Then a new entry will be generated in the entity list. In the control property page at the right hand several parameters can be entered.

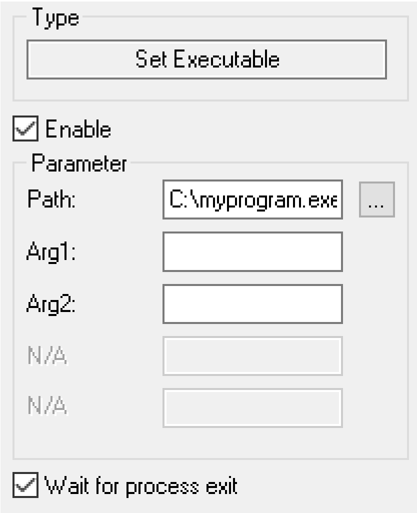


Figure 262: Executable Control Object

Enable: If activated the control object is activated and will be performed within the next marking process.

Parameter:

> **Path:** Defines the full path of the program that should be executed.
>
> **Arg1/Arg2:** Enter command line arguments here.

Wait for process exit: Pause the job process until the exe is closed. Since version 4_0_5_20210611_0029 it is possible to stop the job process depending on the exit code of the program. If the exit code is 4242 the job process will be stopped, else the job will continue.

C# Example: Stop or continue job process after closing the program:

```
private void exit_program( bool stop_job_process )
{
    if (stop_job_process)
        Environment.ExitCode = 4242;
    Application.Exit();
}
```

NOTE

Do not use quotation marks in definition of the Path.

10.8.4 Motion Control Object

Before the ScMotionControl object can be used the motion controller must be set up.

The ScMotionControl object can be accesed in the *Object Toolbar*. Click on the motion control object to modify it within the *Control* property page. In this page the parameters can be assigned to the object. These can be motion control parameters as well as string parameters.

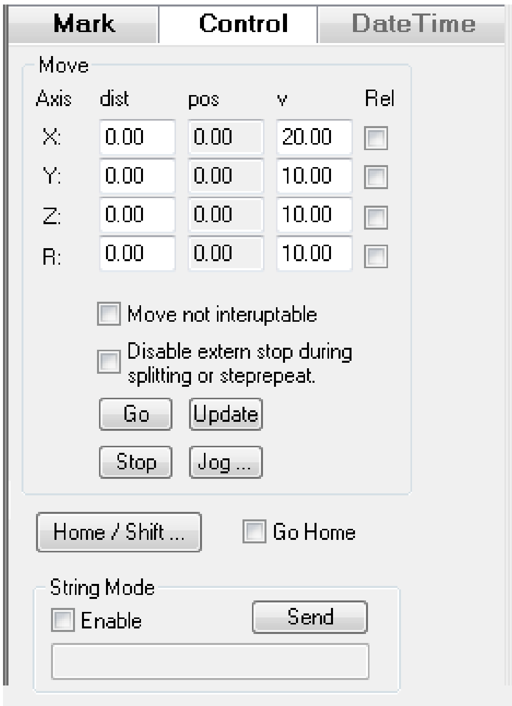


Figure 263: Job control object motion dialog

Move:

Axis: Definition of distance (dist) and speed (v) for each axis.

Rel: If checked, a relative movement is performed instead of an absolute one.

Go: Moves all axes to the defined values above.

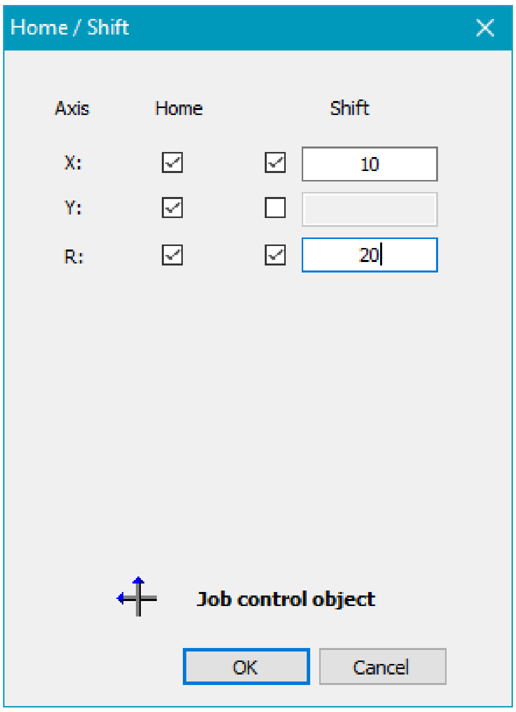
Stop: Stops the movement. In some cases you can loose the current position! Then a homing is necessary to re-calibrate your axes.

Update: Updates actual positions. This might be necessary if manual motions are done. The *Update* Button is not available for stepper motor controllers.

Jog...: Opens the Jog Dialog.

Go Home: Enable this checkbox for setting up a homing procedure in a ScMotionControl object. In the Home / Shift dialog the related axes must be chosen.

String Mode: A RS-232 string command can be send to the motion controller.



Home: When the ScMotionControl object is executed, all axes with enabled *Home* checkbox will start the homing procedure.

Shift: When the ScMotionControl object is executed, the shift values will be applied. To reset the current shift of an axis, enable the checkbox and set the value to *0*. The *Home* and *Shift* functionalities can be combined, first the axis will do the homing procedure, then the shift will be applied.

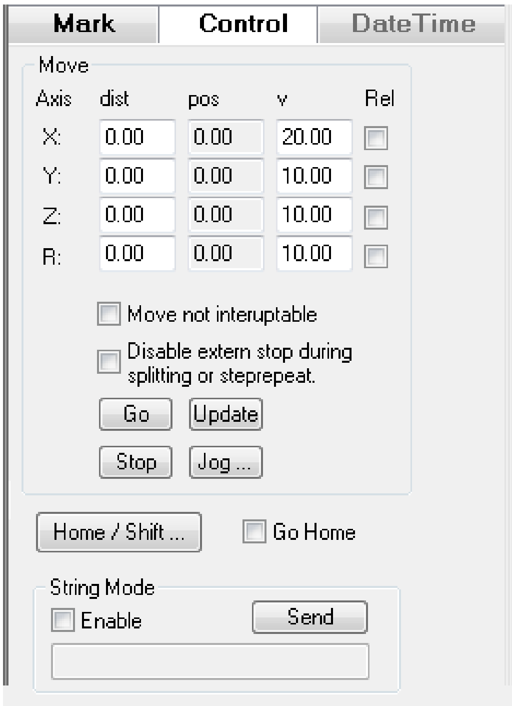


Figure 263: Job control object motion dialog

Move:

Axis: Definition of distance (dist) and speed (v) for each axis.

Rel: If checked, a relative movement is performed instead of an absolute one.

Go: Moves all axes to the defined values above.

Stop: Stops the movement. In some cases you can loose the current position! Then a homing is necessary to re-calibrate your axes.

Update: Updates actual positions. This might be necessary if manual motions are done. The *Update* Button is not available for stepper motor controllers.

Jog...: Opens the Jog Dialog.

Go Home: Enable this checkbox for setting up a homing procedure in a ScMotionControl object. In the Home / Shift dialog the related axes must be chosen.

String Mode: A RS-232 string command can be send to the motion controller.

Figure 264: Job control object motion Home / Shift dialog

 The ScMotionControlGo object (not for motion type 8) has the same functionality as the MotionControl object described above. The difference is that SAMLight will continue with the following entities immediately, so SAMLight will not wait until the motion is finished. Please mind that the motion command will continue even if the marking in progress signal is off at the end of the entity list.

If a motor movement is desired, that is independent of the splitting, it has to be defined as a nonsplittable entity, see chapter Entity List.

The speed of the motion controller can drop if the workload of the job is high. The highest priority is given to the signals to scanner and laser. The remaining time is used to generate stepper signals. The stepper signals of motion type 14 are given every 30us usually, depending on the capacity of the USC card this can be reduced.

10.8.5 Trigger (USC and RTC5 only)

For USC and RTC5/6 scanner cards, the software offers some extended functionality to handle trigger events, to react on them delayed and to generate internal trigger signals. This functionality is useful e.g. to set up jobs that are larger than the available working area or to have defined distances between elements of a job. To set up such an advanced job, there are Trigger Control Objects available within the functional object toolbar.

10.8.5.1 ScMotfOffset

ScMotfOffset (option MOTF required) defines a distance in the entities property page Control.

Figure 265: Wait For Trigger

MOTF Offset:

Enable: If the check box is enabled, a distance in mm can be defined. This distance will be measured in MOTF encoder counts like it is defined in the card-specific MOTF settings. After the distance has been reached, this entity will generate an internal trigger pulse. This internal pulse can be recognized by a ScWaitForTrigger entity.

React to external trigger: By default a ScWaitForTrigger entity will wait for an external or an internal trigger. That means, the pausing of an ScWaitForTrigger entity can be interrupted by an external trigger which will lead to a shorter distance than defined. This can be avoided by deactivating this check box. In this case, the first external trigger will start the pausing defined in the ScWatiForTrigger entity and every other external trigger will be ignored until the internal trigger of the ScMotfOffset entity has been sent.

NOTE

- *React to external trigger only available for USC-2/USC-3. The USC-1 card always reacts to each external trigger, RTC cards ignore external trigger signals while another marking is in progress.*
- *MOTF Offset works on the first activated MOTF channel (USC: first enabled channel (0 or 1), RTC: first channel (X or Y) with multiplier unequal 0).*

When the MOTF distance (calculated by encoder counts) has increased by the ScMotfOffset distance, an internal trigger signal is generated. The internal trigger signal behaves like an external trigger, so the external trigger signal must be low. The ScMotfOffset control object itself does not cause any delay or waiting operation.ScMotfOffset and ScWaitForTrigger together can be used in MOTF applications to separate two marking objects with the distance of ScMotfOffset. This allows you to set up a job where the marking result is bigger than the working area. The order of the entities in the MOTF job should look like this:

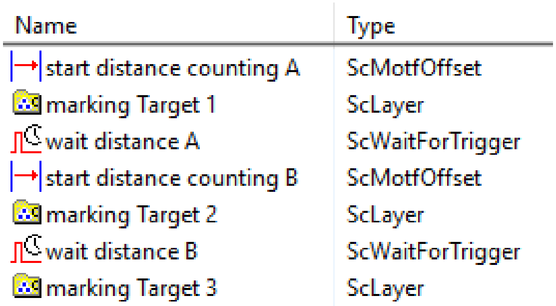


Figure 266: Entity list with Trigger Control Objects

Assume we use the Trigger Mode. Job process would be:

- One external trigger signal starts the job process
- Counting for distance A starts
- Target 1 is marked
  (marking must be done before distance A is traversed)
- Waiting until distance A has passed by
  (external trigger must be low)
- Counting for distance B starts
- Target 2 is marked
  (marking must be done before distance B is traversed)
- Waits until distance B is passed by
- Target 3 is marked
- Job done

ScMotfOffset and ScWaitForTrigger together can also be used to define an initial distance offset after that the first marking part has to be started.

10.8.5.2 ScWaitForTrigger

ScWaitForTrigger pauses the job execution until a trigger signal is received.

- USC-1: Trigger Control objects do not work in MOTF simulation mode.
- USC-1/2/3: The trigger signal can come from an external input or from a preceding ScMotfOffset control object.
- RTC5/6: The ScWaitForTrigger control object does only work with a preceding ScMotfOffset entity. In order to wait for an external signal, a ScWaitForInput control object has to be used.

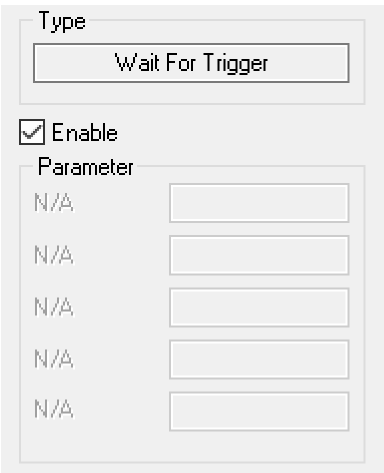


Figure 267: Wait For Trigger

Wait For Trigger:

Enable: If the check check box is enabled, this entity will pause the marking procedure until an internal or external trigger is recognized.

For USC cards, the external start is equal to the signal of the OPTO_IN_0 bit. Therefore, this entity can be used instead of a ScWaitForInput entity which is listening to bit number 1.

An internal trigger can be defined with a ScMotfOffset entity which is described below.

NOTE

ScWaitForTrigger control objects do not work in MOTF simulation mode for USC-1 cards.

10.8.6 AutoCalib (RTC only)

For RTC4 and RTC5 scanner cards, an AutoCalib control object can be inserted into the entity list to perform an automatic self-calibration of an appropriate scan head, which supports this functionality. The corresponding ScAutoCalib symbol within the functionality object toolbar has to be activated by checkbox *AutoCal* within *sc_setup.exe* → *Hardware Settings* → *Settings* → *Driver Settings* → *Mode* before. To change the properties of an AutoCalib control object it has to be selected by clicking on it. Then the *Control* property page can be activated and the properties can be changed:

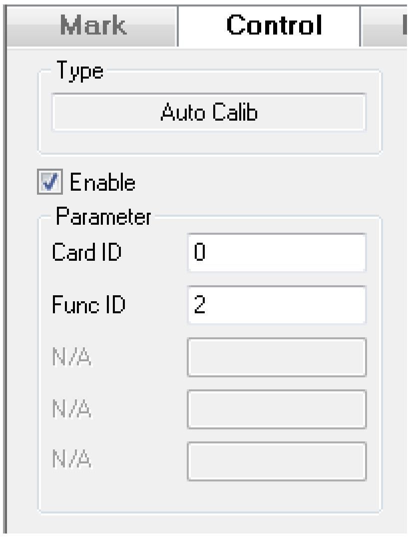


Figure 268: AutoCal Control Object

Enable: This checkbox activates the AutoCal control object.

Parameter:

Card ID: Primary card: 0, secondary card: 1, default = 0

Func ID:

1. Turn auto calibration mode on and go to the reference position for an initial calibration
2. Recalibrate
3. Turn off auto calibration mode

WARNING *The Func ID parameters are different from those of the RTC manuals.*

The output value can be checked by the Command View command *AutoCal* (here it corresponds to the RTC manuals).

10.8.7 SetOverride Control Object

Inserting a SetOverride control object into the entity list, the marking speed, the power and the frequency of the applied pen can be changed using the Override functionality of the Mark property page. This is useful for determining the optimal marking speed, power and frequency parameters by marking an array of identical entities with different speed, power and frequency. The test array can be created by copying a row of entities, in which a SetOverride control entity changes one parameter of the following array entity. The value which is set by SetOverride will stay valid even after the marking as long as no other override value is set.

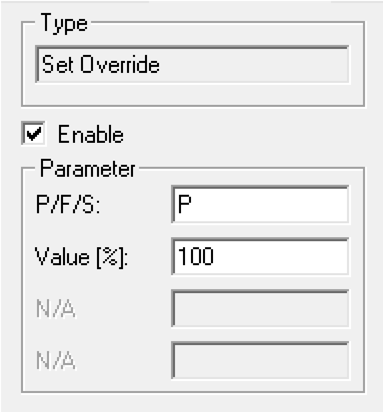


Figure 269: SetOverride Control Object

Enable: This check box activates the *SetOverride* control object.

Parameter:

P/F/S:
- P = Power
- F = Frequency
- S = Speed

Value[%]: The override factor is expressed in percent and can be used to increase or decrease the P/S/F value of the pen during the mark process for the marking of all the following entities in the entity list.

The object SetOverride works also in Flash (it is included in the UNF file). However, the effect is different than in SAMLight: the value does not stay after the marking. This override is not to be confused with the FCI commands OP, OF, OS.

10.8.8 ScJump Control Object

Inserting a ScJump control object into the entity list, creates a jump with the possibility to define specific X, Y, and Z coordinates. Alternatively, the home position can be addressed.

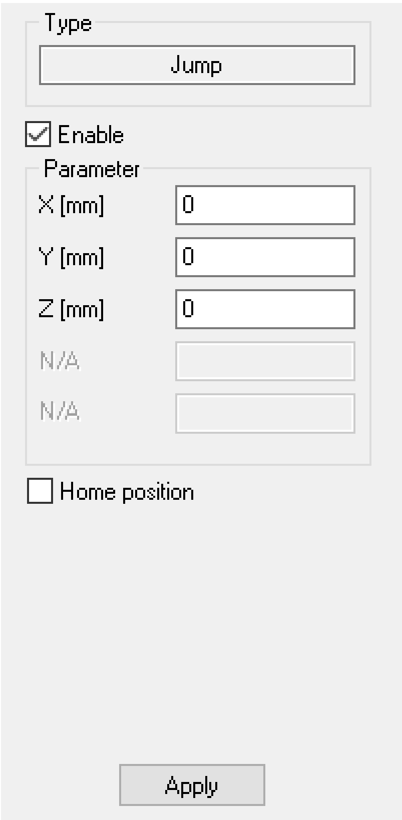


Figure 270: ScJump Control Object

Enable: This check box activates the *ScJump* control object.

Parameter:

X/Y/Z:
- X = X coordinate in mm
- Y = Y coordinate in mm
- Z = Z coordinate in mm

Home position: When activated, the jump parameters from the home position are applied.

A pen can be assigned to the ScJump Control object.
The jump speed set in the pen is then used for the ScJump Control object.
The home jump is set as the default pen.

11 Entities Properties

This chapter describes how to change the properties of the entities.

11.1 Transformations

Translation, scaling, rotation and slant operations can be done on all geometric objects. If the optic dimension 3D tool is available, also 3D Transformations are possible.

11.1.1 2D Transformations

For the transformations by mouse, see chapter View2D, Manipulation of Objects.

Transformations by keyboard: The dimension property page can be used to change the dimension of the selected entity. This page has no Apply button. The Translate, Scale, Rotate and Outline operations can be executed separately by typing in the requested values and clicking the corresponding Go button.

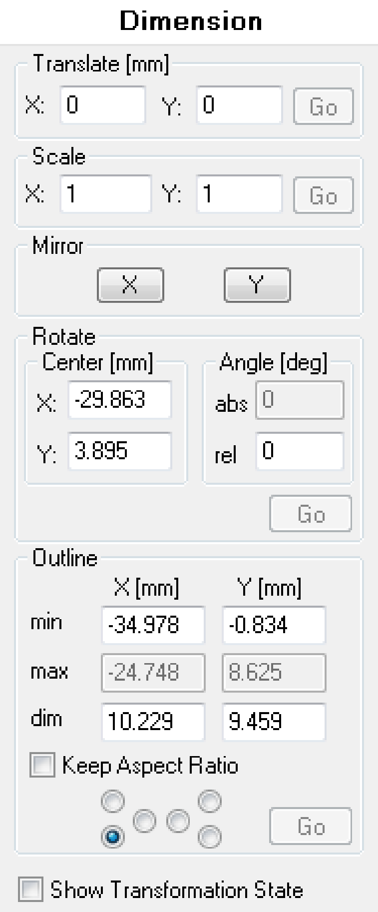


Figure 271: 2D Transformation Dialog

Translate: Translation values X and Y are relative values to the actual position.

Scale: Scale values X and Y are relative values to the actual size of the entity.

Rotate:

Center: The default coordinates of the rotation center are the center coordinates of the selected object. After each transformation the center of the object is recalculated.

Outline:

min:

X: Positions the left border of the outline

Y: positions the lower border of the outline.

max:

X: Positions the right border of the outline

Y: positions the upper border of the outline.

dim: Defines the width X and the height Y of the outline. If Keep Aspect Ratio is selected, the relation between X and Y value keeps constant after an outline transformation.

center: Active if the right middle radio button is selected. Positions the middle point of outline.
Radio buttons: Shows according min/max x/y edit fields, e.g. with selecting the upper left check box the upper left outline point can be repositioned.

Show Transformation State: When this is selected a dialog appears which shows the absolute translation, scale and rotate values of the selected object. See image below:

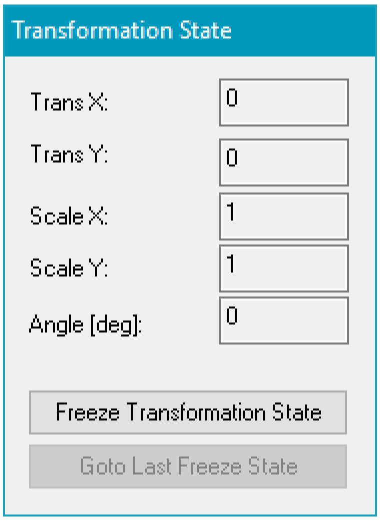


Figure 272: 2D Transformation State

Trans X, Trans Y: Shows the translation in X and Y direction.

Scale X, Scale Y: Shows the scaling in X and Y direction.

Angle: Shows the rotation angle of the entity.

Freeze Transformation State: Saves the current state, so that it can be reproduced later if some changes have been made and you wish to make them undo.

Goto Last Freeze State: Reproduce the state that was previously saved with "Freeze Transformation State".

11.1.2 3D Transformations

The Z-Dimension property page can be used to change the z-dimension of the selected entity. This page has no Apply button. The Translate, Scale and Outline operations can be executed separately by typing in the requested values and clicking the corresponding Go button.

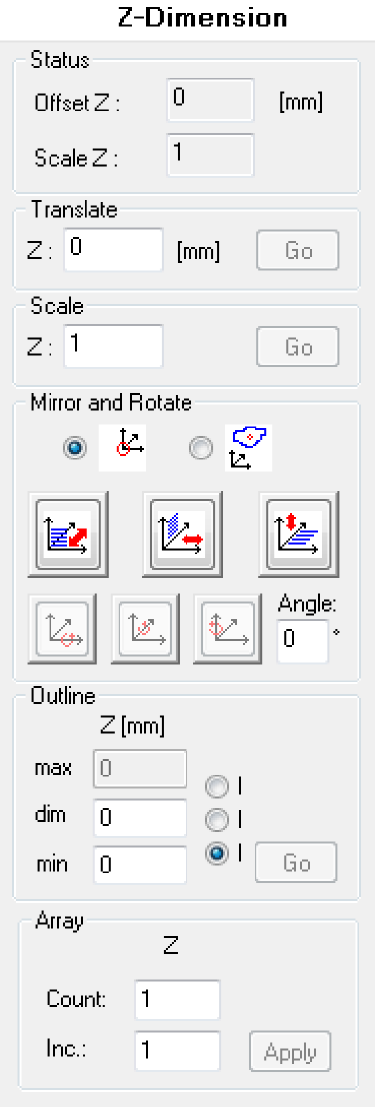


Figure 273: 3D Transformation Dialog

Mirror and Rotate: With the radio button the center of mirroring and rotation is selected.

 The center is the origin point of the world coordinate system.

 The center conforms with the center of the selected object.

Pressing the first mirror button the selected objects gets mirrored to the XZ-plane. The painting plane is the XY-plane.

The rotation buttons are becoming activate when an angle is entered. The angle range is between -360 and 360 degrees.

Outline:

max, min: Positions the outline in the z-coordinate.

dim: Scales the object in z direction to the width given in dim.

Array: The array functionality allows to create reference copies of the selected entity in Z dimension. Reference copies in X and Y dimension can be found in the Entity Info property page.

Count: Define the number of copies in Z dimension.

Inc.: Define the distance between the copies in Z dimension.

The selectable radio buttons aside allow to enable any of the outline value fields, but disable each field which is not editable for the same translation to avoid conflicts.

NOTE

The Z-Dimension property page requires the Optic3D option.

11.2 Hatch

This page can be used to generate hatching for entities which have closed PolyLines. Single objects as well as clustered groups can be hatched. A pen for a specific hatch can be selected on the mark property page.

For a Hatch to be actually marked, the according checkboxes 'MarkFlags' have to be enabled in the Pen Settings of the selected Pen (*Mark* → *Edit...* → *Misc*).

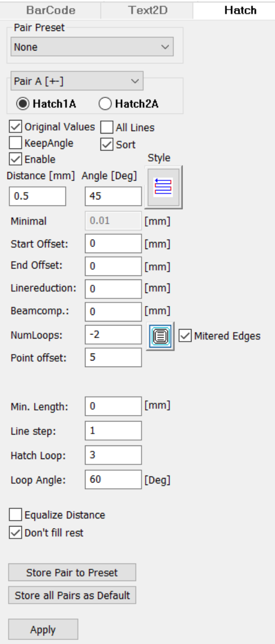


Figure 274: Hatch Dialog

Pair Preset: Access up to 10 different preset parameters.

Pair A [--]: Access to 5 hatch pairs (A, B, C, D and E) with a total of 10 hatches. "Pair A [--]" indicates that Hatch1A and Hatch2A are disabled ." Pair A [+-]" indicates that the Hatch1A is enabled and Hatch2A is disabled.

Hatch1A, Hatch2A: Chose a hatch within a hatch pair. Order of hatching: 1A, 1B, 1C, 1D, 1E, 2A, 2B, 2C, 2D, 2E

Original Values: This switch is useful if the entities have been scaled or rotated after hatching and allows to rehatch the entity with the original values.

KeepAngle: If checked, the angle is relative to the entities rotation when rehatching the object. So if rotating a hatched entity and rehatching the entity with this flag set, the hatch lines are relative to the entity rotation angle.

Enable: If this option is being chosen the current hatch definition is activated.

All Lines: If this is not checked, ScPolyLine2D structures are used for calculating hatches. If checked, also ScLineArray2D objects are taken into account.

Sort: Changes the order of the hatch lines to decrease the total jump length. It also can flip hatch lines (change the marking direction). Below you can find an example, the marking order of the hatch is 1, 2, 3, 4, 5. If *Sort* is not enabled the hatch order is simple from bottom to top. At the crossing points the sort can cause problems depending on the material / application.

Distance: Distance between two hatch lines in mm.

Angle: The angle to the x-axis in degrees.

Style: The movement direction can be defined by clicking on the button with the bitmap. The blue lines on the bitmap show the mark lines and their direction. The red lines show the scanner jumps. See Hatch Styles for further information.

Minimal Jump: For continuous hatch styles and a minimal jump distance can be set. If the minimal jump distance is smaller than the distance between the end and beginning of two hatch lines, then the hatch lines are not connected. See the example below.

Start Offset: Defines the start distance between the outer line of the object and the first hatch line. The default value is 0 which means that the value entered at "Distance" is taken as Start Offset.

End Offset: Defines the end distance between the outer line of the object and the last hatch line.

Linereduction: Shortens the hatch lines.

Beamcompensation: Shortens the hatching lines so that they do not overlap with the contour. See also the examples below.

NumLoops: The number of inner loops can be defined. See the example of blue concentric circles below which was created by a circle with NumLoops. By clicking on the button, the inner loops are activated. Then, you can choose between marking order from outside to inside or from inside to outside. The contour will always be marked first. The NumLoops are disabled with .

Negative NumLoops are possible when Improved Beam Compensation is activated. With negative NumLoops, there is an additional option, mitered edges. When activated, the outer loops will not be rounded at the corners, as per default, but will repeat the corners instead, or cut them, depending on the angle.
Marking from the outside in is beginning next to the contour is this case, and marking from inside to outside is choosing the marking order towards the contour.

Point Offset: Only active when NumLoops is enabled. This will make the hatching lines start at different points from the hatch. For example if you have a circle hatched with NumLoops so that you have some concentric circles every circle marking will start at a different angle.

Min. Length: If this value is bigger than 0, all hatch lines shorter as this value are not

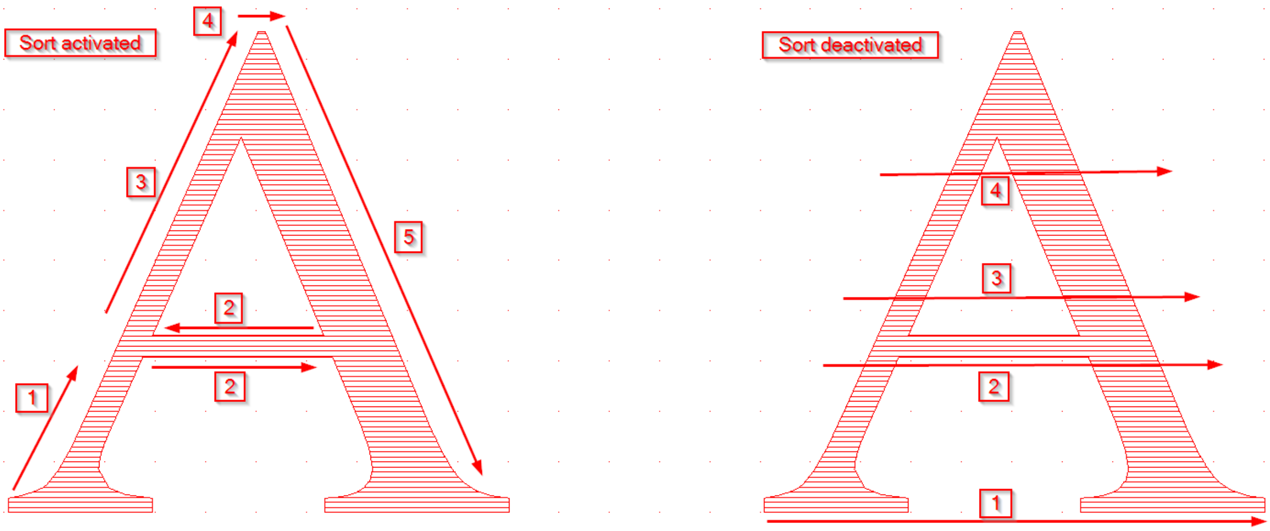


Figure 275: Hatching order with Sort Flag activated (left) and sort flag deactivated (right)

NOTE *If several different hatch angles are used it could be helpful to use the following four special hatch angles to avoid potential unwanted hatch patterns: 0.0°, 18.4349°, 45.0° and 71.5651°.*

NOTE *This distance gets approximated from the hatcher as the conditions Start Offset and Distance do not allow an exact definition of End Offset (for an exact definition of End Offset see: Equalize Distance).*

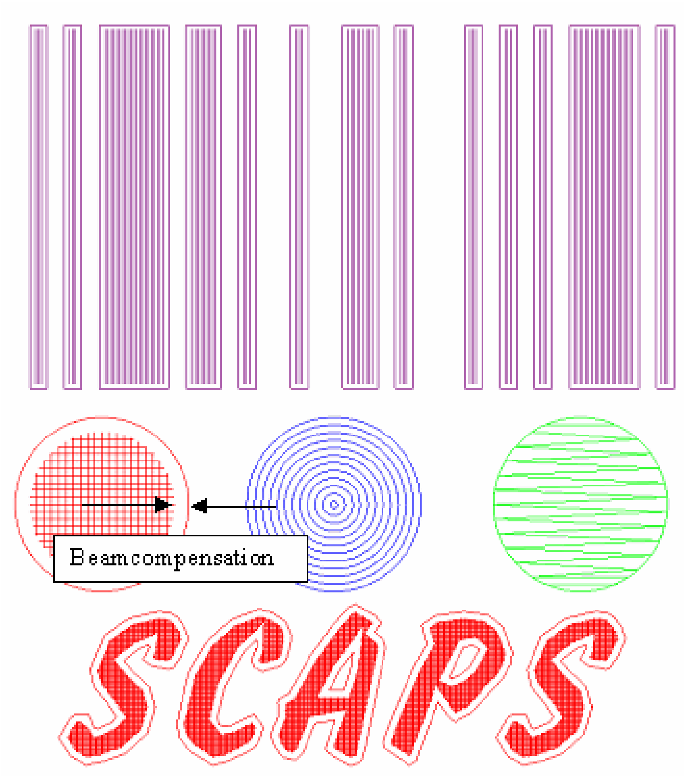


Figure 276: Hatch examples

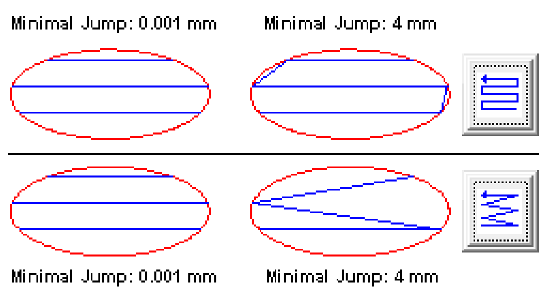
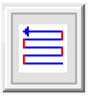


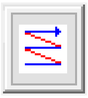
Figure 277: Minimal Jump examples

11.2.1 Hatch Style

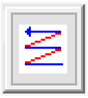
The following hatch styles can be chosen:



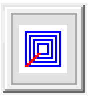
The marking direction is alternating. Jumps are executed between the hatch lines.



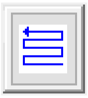
The marking direction is conserved from left to right. Jumps are executed between the hatch lines.



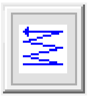
The marking direction is conserved from right to left. Jumps are executed between the hatch lines.



Hatch lines are closed and follow the outline of the entity. Jumps are executed between the hatch lines.



For this continuous hatch style, the marking direction is alternating. Hatch lines are connected (laser remains on) if the distance between end and start point of lines is less than or equal the minimal jump distance.



For this continuous hatch style, the marking direction is alternating. Hatch lines are connected (laser remains on) if the distance between end and start point of lines is less than or equal the minimal jump distance.

Please note: in this case the style leads to non-parallel hatch lines.

Switching between the different hatch styles is done by clicking on the hatch style icon. The blue lines on the bitmap show mark lines and their direction. The red lines show the scanner jumps.

11.3 Entity Info

The Entity Info property page shows the type and the name of the selected entity.

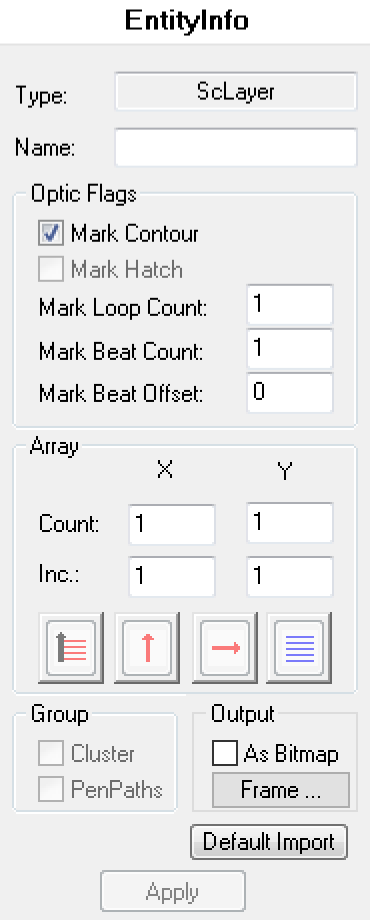


Figure 278: Entity Info Dialog

Name: Here a name can be set for an entity that can be used to decide what its purpose is within the job, to access it via the Client Control Interface or to reference it from within a serial number.

Optic Flags: The markable flags can be changed within this page separately for the contour and for the hatches generated with the hatcher. This flags define whether to mark the entity or not.

Mark Loop Count: Defines how often an entity is marked when a mark command is issued. For example, a circle can be marked 100 times with every mark command to form a hole inside a material. If -1 is entered here the entity will be marked infinitely often.

Mark Beat Count: Defines after how many subsequent mark commands an entity should be part of the mark.

Mark Beat Offset: Allows a shift of the Mark Beat Count for every entity.

Array: The array functionality allows to create reference copies of the selected entity in X and Y dimension. Reference copies in Z dimension can be found in the Z-Dimension property page.

Count: Define the number of copies in X and Y dimension.

Inc.: Define the distance between the copies in X and Y dimension.

Group:

Cluster: If the entity is a group, it can be clustered. A clustered group will be hatched as one single object while in a unclustered group each entity of groups is hatched separately.

PenPaths: Groups have an assigned pen as well. The pen path of the group is executed when the following two points are true:

- When the group-pen-path is enabled (Group > EntityInfo > PenPaths).
- And when the pen-pen-path is enabled (Group assigned pen number > Paths > Activate) as well.

If the pen path of the group is executed all pens within this group will be ignored! This includes pen paths of sublevel objects as well.

Output As Bitmap: If an entity is selected then by activating this checkbox the entity is transformed to a Bitmap. After that the Bitmap property page becomes available and the bitmap can be modified like a normal bitmap.

Frame...: This button opens an extra dialog to specify a frame of blank pixel (in mm) which are added to the bitmap.

Marking examples:

Assume there are 3 entities inside a job with following settings:

Entity	Mark Loop Count	Mark Beat Count	Mark Beat Offset
1	1	1	0
2	10	2	0
3	5	3	1

Table 30: Example for three entities with different mark loop count, mark beat count and mark beat offset

With these settings the following mark sequence can be generated:

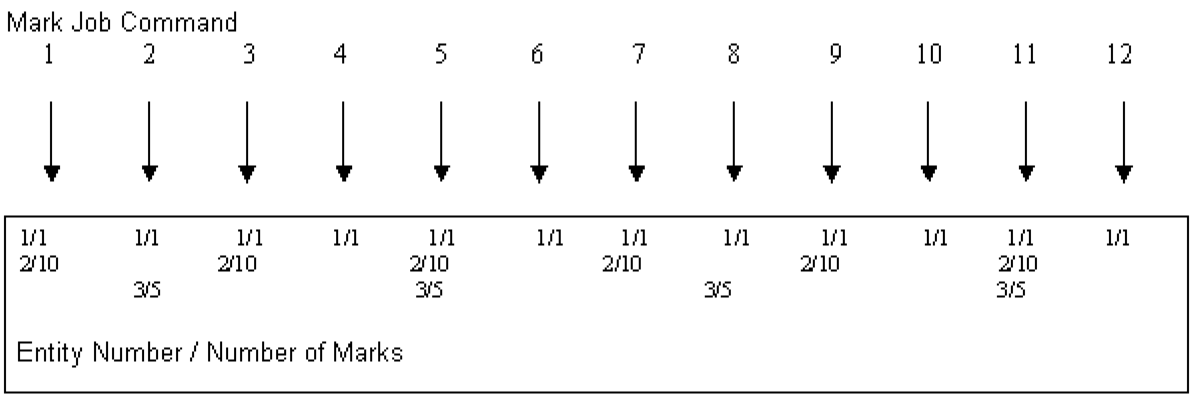


Figure 279: Example for marking order with mark loop count, mark beat count and mark beat offset

- Entity 1 is marked every mark one time (default).
- Entity 2 is marked every second mark and then 10 times.
- Entity 3 is marked every third mark and then 5 times. The Beat Count is offset by 1. So the first time the entity will be marked is on Mark Command 2.

11.3.1 Bitmap Frame

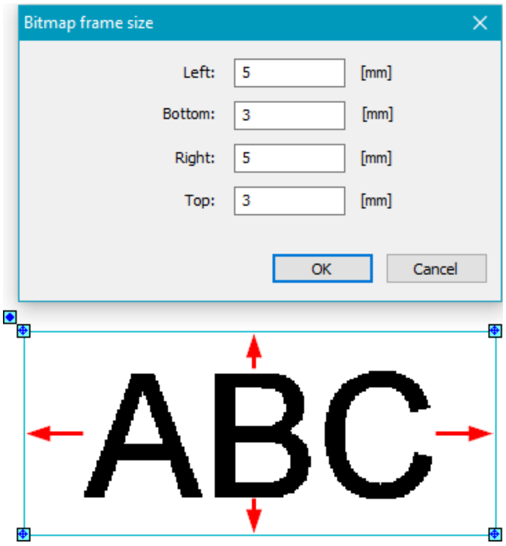


Figure 280: Bitmap Frame Size Dialog

If the "As Bitmap" option is checked, the Bitmap Frame function will be available. This function works similar to a quiet zone for inverted barcodes. It adds blank pixels to each side of your entity. You can specify the size of every side separately.

11.4 Element Info

Gives information about a single polyline structure.

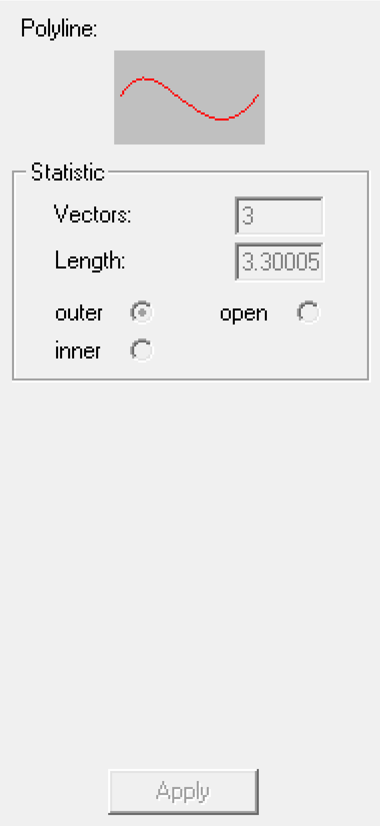


Figure 281: Element Info

Vectors: Number of lines of the selected polyline.

Length: Length of the polyline.

Outer / Inner: Orientation of the polyline

outer: counterclockwise

inner: clockwise.

Open: The polyline is open or the structure of the polyline to which it belongs is open.

11.5 Styles

The style property page has to be enabled in the *Settings* → *System* → *Extra* dialog.

A style consists of several properties and can be assigned to an entity or a group. It is possible to store these information from a single entity to a style as well. You can create a style library for different scenarios, for example different materials, deep engrave or cleaning. The styles will be stored in your SAMLight settings file (default: *<SCAPS>*\system\sc_light_settings.sam).

Here is a list of the properties a style can handle:

- Pen (A pen with a pen-path can be used as well)
- Hatch
- Mark Loop Count
- Mark Contour
- Mark Hatch

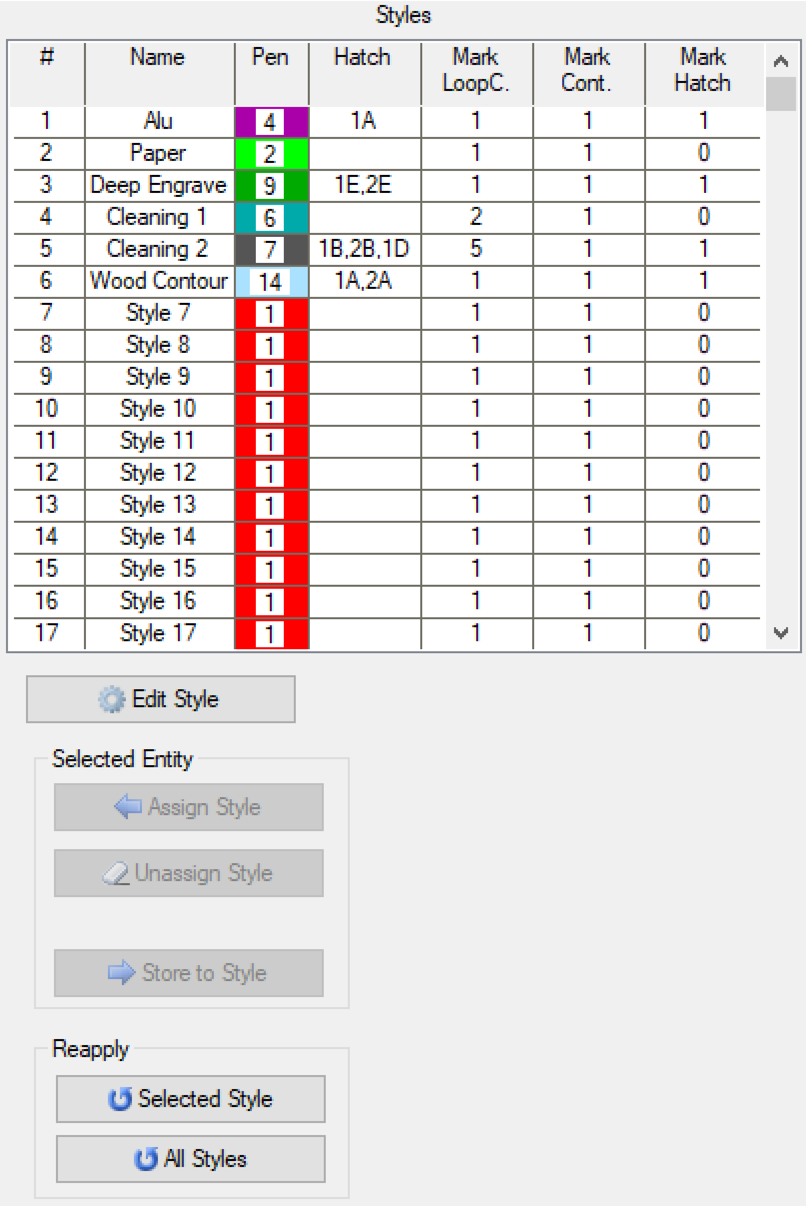


Figure 282: Style Property Page

Edit Style: Opens the Edit Style Dialog

Selected Entity: These three buttons only affects the entities which are selected in the entity list.

Assign Style: Assigns the selected style to the selected entities. All properties of the style will be copied to the entities.

Unassign Style: Unassigns the style of the selected entities. The properties of the entities will not be changed.

Store to Style: All properties of a single selected entity will be stored in the selected style.

Reapply: If a style has been changed after it has been assigned to entities the parameters of these entities will not be changed automatically due to the changes of the style. If you want to change the parameters of the entities you have to reapply the styles.

Selected Style: Copies the current defined properties of the selected style to all entities assigned to this style.

All Styles: Copies the current defined properties of all styles to all entities assigned to each style.

The following can be opened while doing a right mouse click on the styles table:

Import styles from different settings-file: Choose a *.sam (settings file) to load its saved style settings.

Import styles from a *.ssf settings-file: Import a *.ssf file with all the saved styles settings. This feature works also in SAM3D style.

Export styles to a *.ssf settings-file: Save all the edited styles in a *.ssf settings-file. This feature works also in SAM3D style.

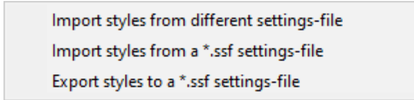


Figure 283: Style Property Page

Coloring the style list: A style assigned to a selected entity can be seen in the color of the current row in the style property page.

- o **blue** [1]: The selected entity has not been assigned to any style.
- o **green** [2]: The entity has been assigned to the highlighted style and the style has not been changed since the assignment.
- o **brown** [3]: The selected entity has been assigned to the highlighted style but the style has been changed afterwards. If you want to update the parameters of the selected entity corresponding to the changes of the assigned style you have to reapply the style.

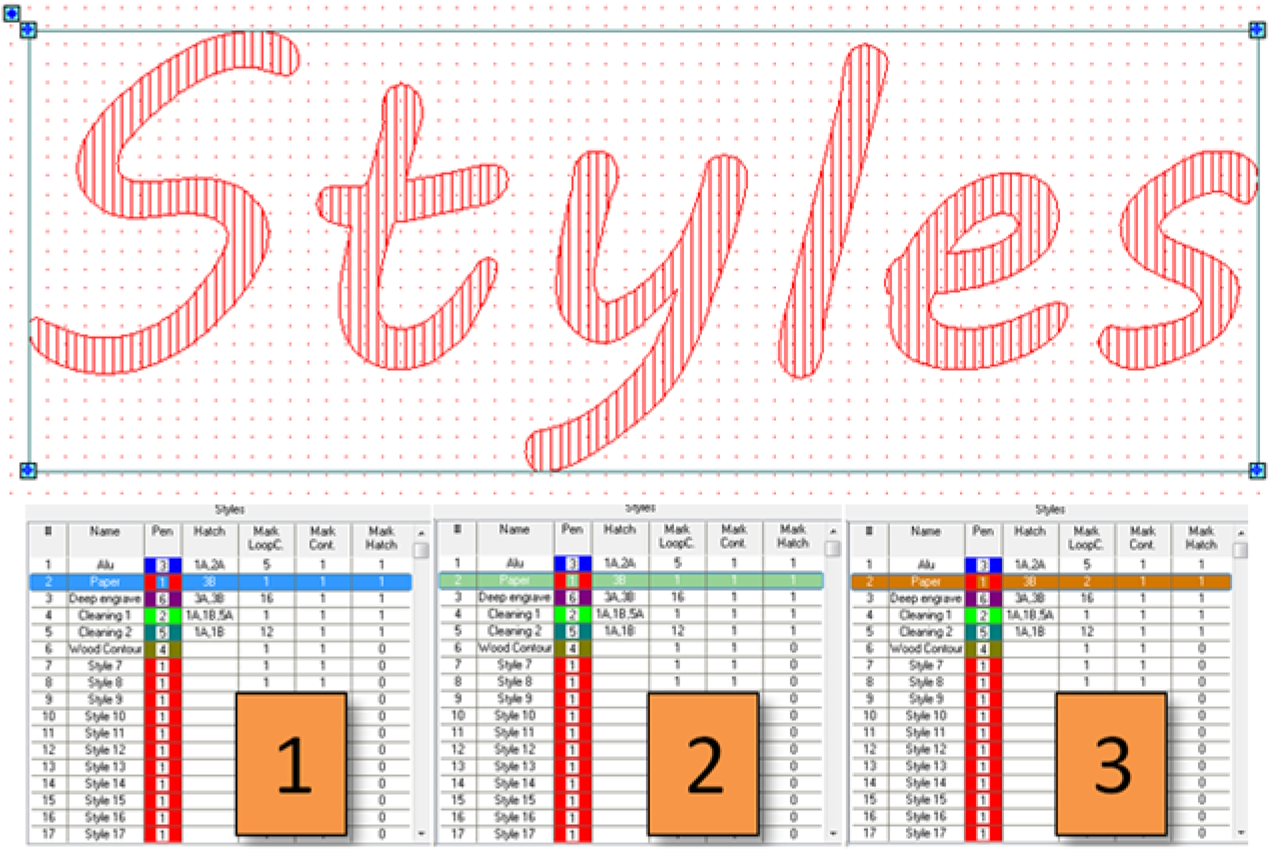


Figure 284: Applying a style to a selected entity

NOTE

There are two different ways to activate/deactivate the marking behaviour of the contour and the hatch. The first one is in EntityInfo and the second possibility is in the pen-settings. The style property page shows only the information of the EntityInfo. If you have deactivated mark contour or mark hatch in the pen-settings you cannot see it in the style property page but the deactivation will be recognized in the marking process.

11.5.1 Edit Styles

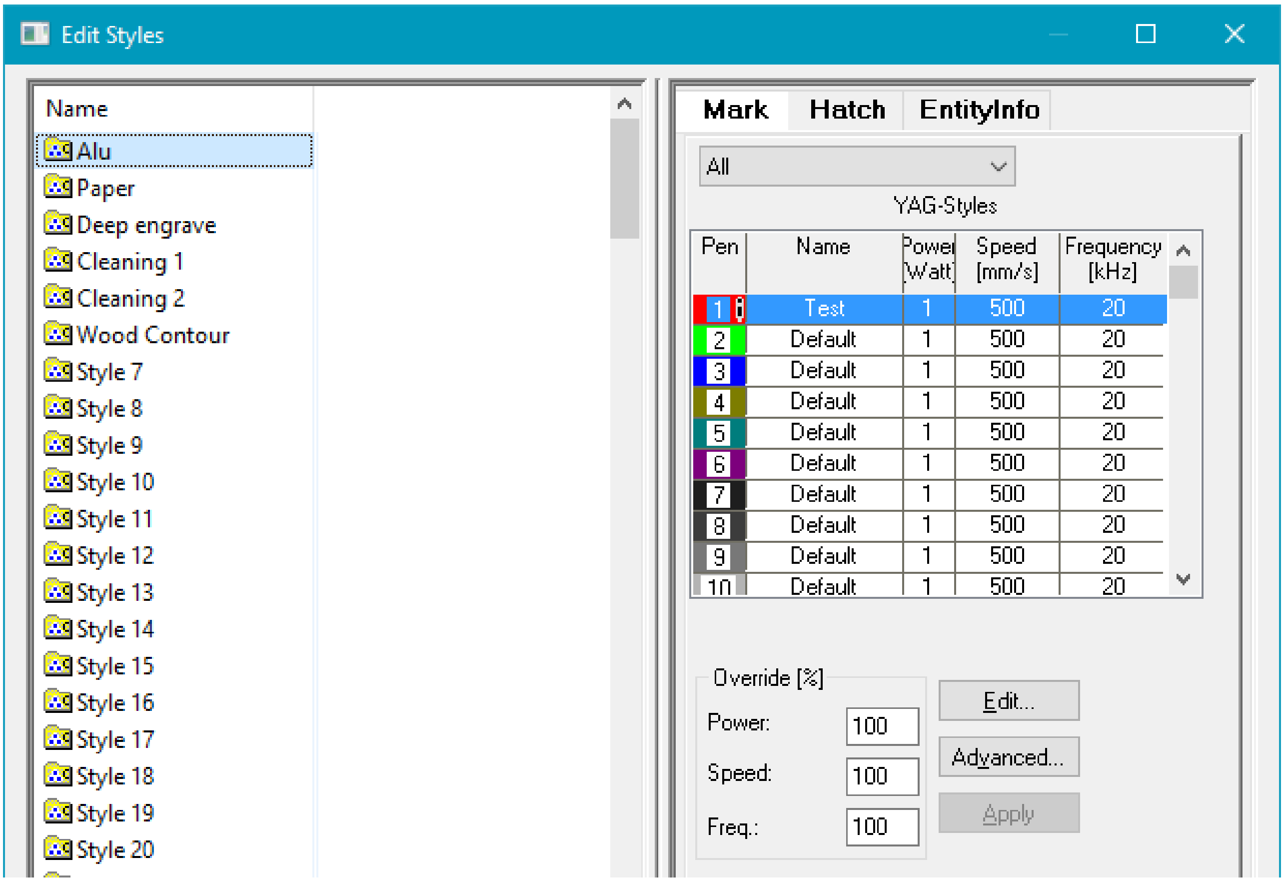


Figure 285: Edit Styles Dialog

Edit Style Dialog: The tabs of the Edit Style Dialog corresponds to the known property pages Mark, Hatch and EntityInfo.

12 Import-Export

This chapter describes how to import and export different file formats into SAMLight.

12.1 Import

Menu bar → File → Import opens a Dialog Box to import Graphic files and puts them into a new layer.

Available import formats are:

Extension	Description
*.ai	Adobe Illustrator (AI) is a vector graphics file format.
*.gif	Graphics Interchange Format (GIF) is a raster graphics file format.
*.tif	Tagged Image File (TIF) is a raster graphics file format.
*.txt	Point Cloud Data is a ASCII format containing 3D vertices.
*.bmp	Bitmap (BMP) is a raster graphics file format.
*.cmx	Corel Metafile Exchange (CMX) is a vector graphics file format.

Extension	Description
*.cnc	CNC G-Code is a language to control CNC (Computer Numerical Control) machines. [a]
*.dxf	Drawing Exchange Format (DXF) is a CAD file format. (Also DXF2)
*.emf	Enhanced Metafile (EMF) is a raster graphics file format.
*.jpg	Joint Photographic Experts Group (JPEG) is a compressed image format.
*.mcl	Marker Control Language (MCL) is a GSI PC-Mark vector graphics file format.
*.pcx	Personal Computer Exchange (PCX) is a raster graphics file format.
*.plt	Hewlett-Packard Graphics Language (HPGL) Plotter File (PLT) is a language format for printing line drawings, specifically designed for 2D plotters.
*.png	Portable Network Graphics (PNG) is a raster graphics file format.
*.saf	SAF is a SCAPS archive.
*.svg	Scalable Vector Graphics (SVG) is a 2D vector graphics file format.
*.tga	TGA or TARGA is a raster graphics file format.
*.twain	TWAIN is a software protocol and applications programming interface between software and scanner.

Table 31: Available import formats

[a]: Please note that not all available G-Code commands are interpreted into entities by SAMLight. For example, G0 commands are ignored during import to SAMLight.

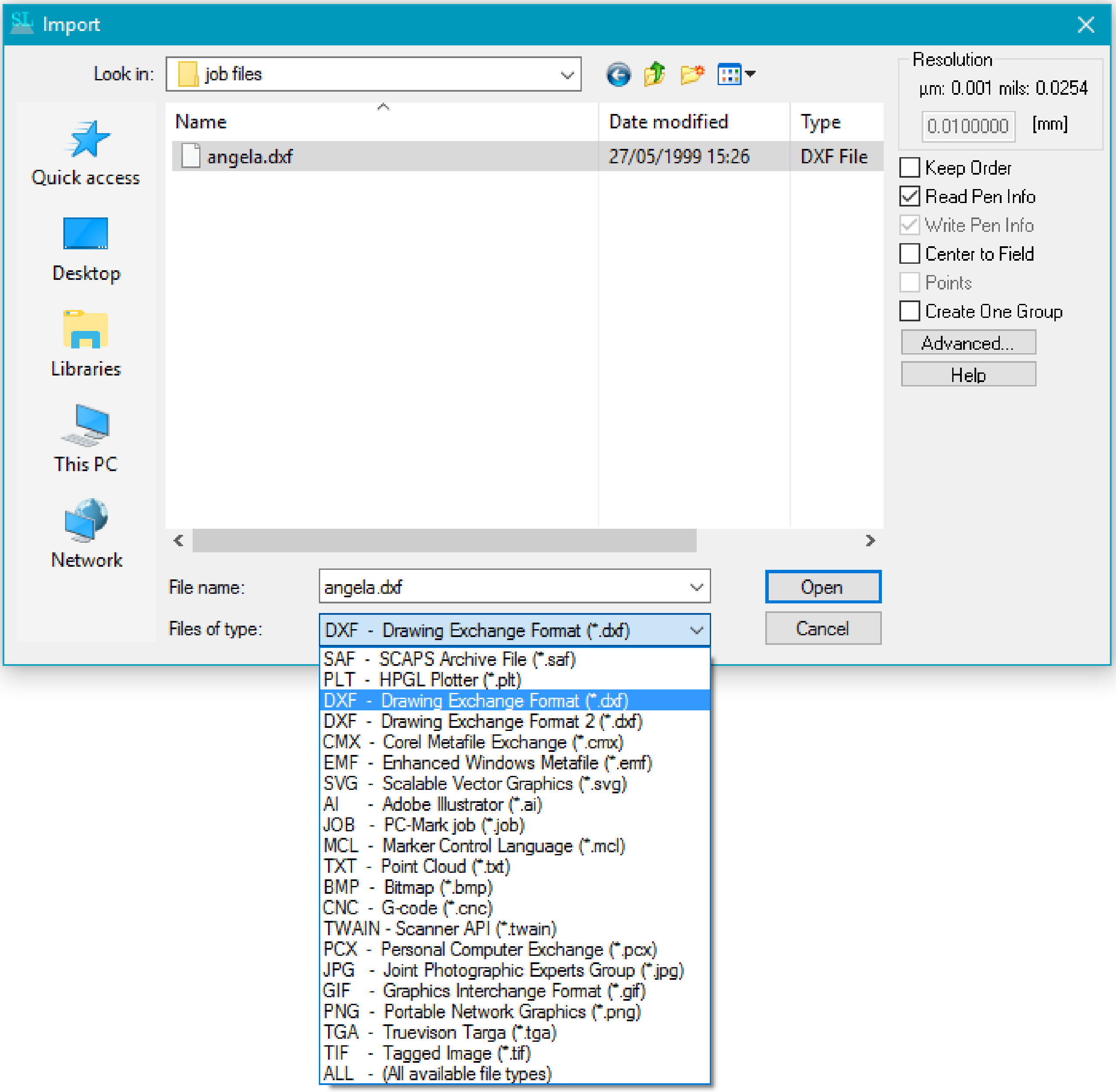


Figure 286: Import Dialog

Resolution: Defines how to transform the vectors from the file in case no units are given. The resulting unit is mm.

Keep Order: If this is selected all data is written in LineArrays to make sure that the exposure order is the same as the order inside the file. Less memory is needed for loading and saving a file. The disadvantage is that closed PolyLines can not be hatched after the import. The example below shows this behaviour.

Read Pen Info: Enables the interpretation of the Pen information given in the File.

Center to Field: Centers the entity to the given field size.

Create One Group: Can be used to optimize the order of the vectors to minimize the marking-time. All selected ScPolyLine2D objects will move into a new ScPolyLines2D folder. Then two polylines will be closed to a new polyline if the distance between them is smaller than CloseDist (in mm). If CloseDist is set to '0' no polylines will be closed. Afterwards all polylines will be sorted in order to minimize jumps. If necessary and possible polylines will be flipped to optimize the marking order of the polylines and thus the marking-time as well.

Advanced...: Allows to influence the import of the drawing more detailed, see chapter Import Advanced.

Example "Keep Order":

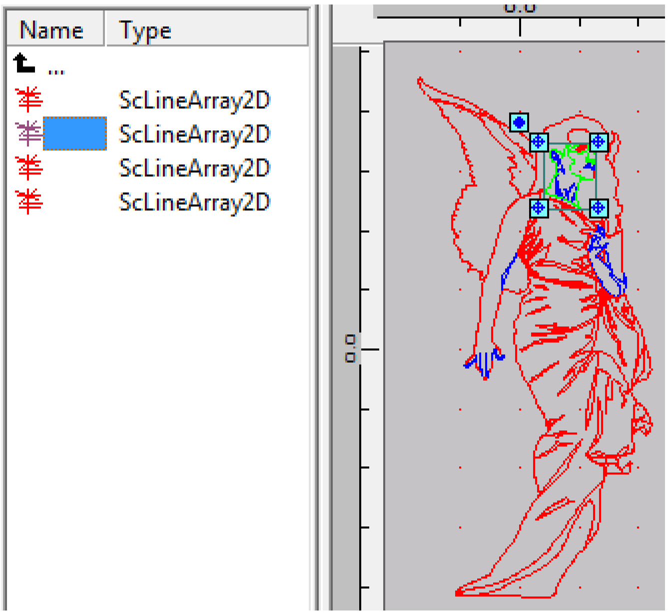


Figure 287: Example for Keep Order enabled

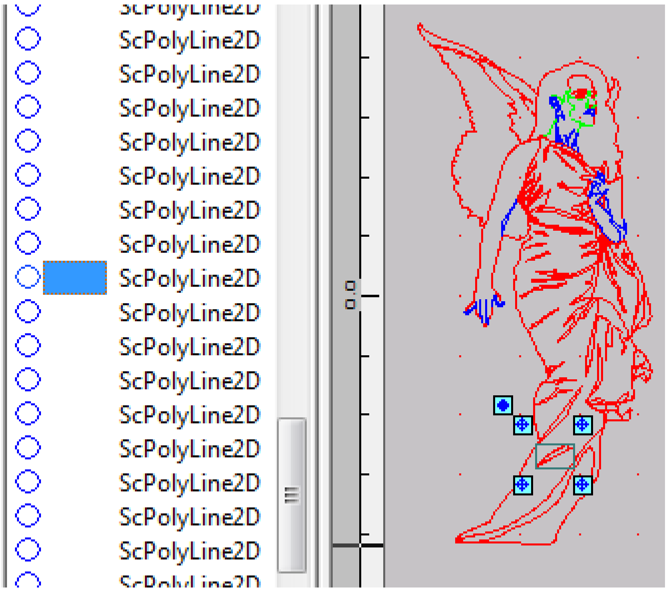


Figure 288: Example for Keep Order disabled

Activated options for available file formats are:

	Resolution	KeepOrder	ReadPenInfo	Points	Preview	PenColors
SAF	–	–	✔	–	✔	–
PLT	✔	✔	✔	✔	✔	–
DXF	–	✔	✔	–	✔	–
DXF v2	–	–	✔	–	✔	✔
CMX	–	–	✔	–	✔	✔
EMF	–	–	✔	–	✔	✔
SVG	–	–	✔	–	✔	✔

	Resolution	KeepOrder	ReadPenInfo	Points	Preview	PenColors
AI	–	✔	✔	–	✔	–
JOB	–	–	✔	–	✔	–
MCL	–	–	✔	–	✔	–
BMP	–	–	–	–	✔	–
PCX	–	–	–	–	✔	–
TXT	–	–	–	–	✔	–
TWAIN	–	–	–	–	✔	–
JPG	–	–	–	–	✔	–
GIF	–	–	–	–	✔	–
PNG	–	–	–	–	✔	–
TGA	–	–	–	–	✔	–
TIF	–	–	–	–	✔	–

Table 289: ✔: available, - : not available

For Import of a SVG file, the Pen Colors are mapped as follows:
- Each RGB channel value of the import file and of each SAMLight pen is divided by 64 and truncated. Thus, all values are mapped into one of the following blocks (0-63, 64-127, 128-191, 192-255).
- Then, the first SAMLight pen which matches the SVG color blocks in R, G and B channel, is taken. If no corresponding pen is found, the next undefined pen is taken, starting with 21.

12.1.1 Point Cloud Files

A point cloud file needs to have following format.

```
x y
10.1 15.13
2 6
```

Importing this creates two points with coordinates x = 10.1, y = 15.13 and x = 2, y = 6.

12.1.2 Import Advanced

For the following file formats, the button *Advanced* (see figure 290) is active to influence the display of the drawing.

- AI
- CMX
- CNC
- DXF Version1
- DXF Version2
- EMF
- MCL
- PLT
- SVG

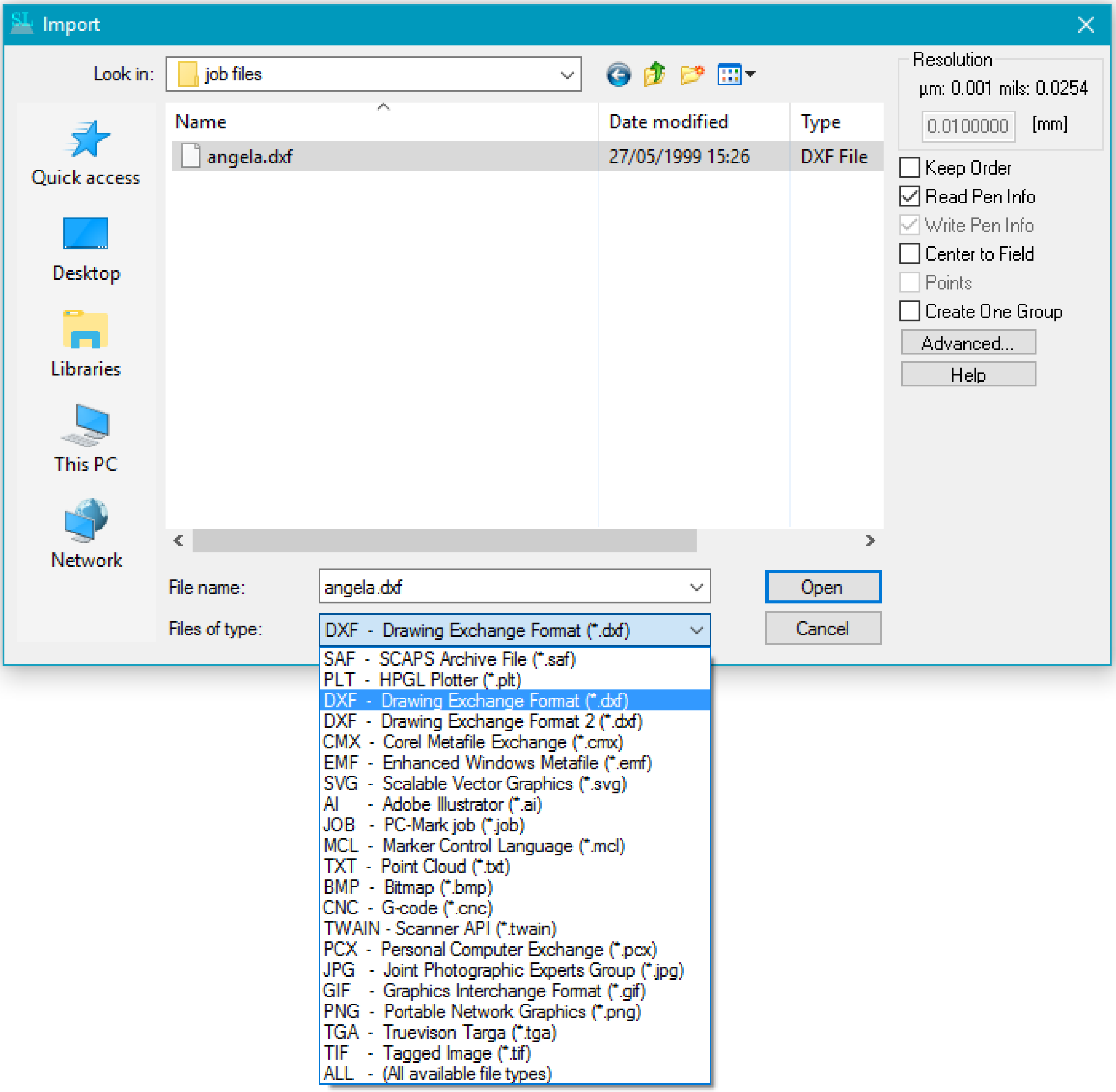


Figure 290: Import dialog

12.1.2.1 Advanced for several formats

For files of format CMX, DXF Version2, EMF and SVG, the following *Advanced Styles* dialog is given:

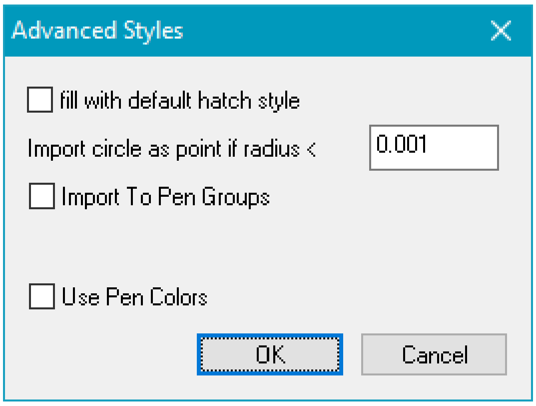


Figure 291: Advanced Styles for other files dialog

fill with default hatch style: Can be selected to hatch filled objects.

Import circle as point if radius < ...: A limit can be given up to which size a circle should be imported as a point.

Import to Pen Groups: If activated, all entities are sorted by pen and are imported to pen entities.

Use Pen Colors: Read the colors of pens from a SVG file. A prerequisite is, however, that the corresponding colors must also exist in the pen list of the Mark property page: The user can add or change any color via ViewProperties or in the View Settings Dialog. Please take off similar colors in the pen list to avoid confusion. By using this check box, the check box of "read pen info" in the Import Dialog should also be activated. This function is now only available for import from a SVG file. If the check box is not checked, the colors of SVG file would be ignored and the pens will be used in the order of the pen list on Mark property page.

12.1.2.2 Advanced for AI

For AI formats, the following *Advanced Styles* dialog is given:

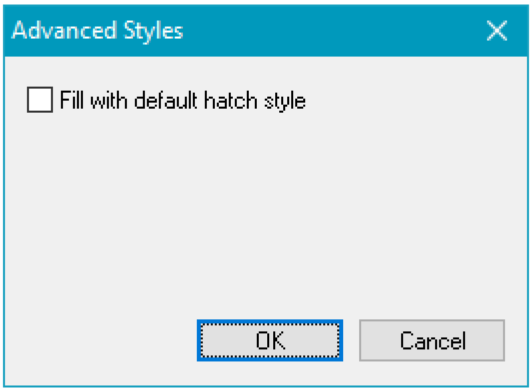


Figure 292: Advanced Styles for AI dialog

Fill with default hatch style: Is unchecked by default. When unchecked, no hatch is taken.

When checked, the default hatch pair is taken to fill all closed polylines. To set a default hatch pair, enable a hatch A and the corresponding hatch B, click on "Store all Pairs as default" and confirm the dialog "Save the state of all hatch 'Enable' checkboxes as well?" with yes.

12.1.2.3 Advanced for CNC

For CNC formats, the following *GCode Advanced Styles* dialog is given:

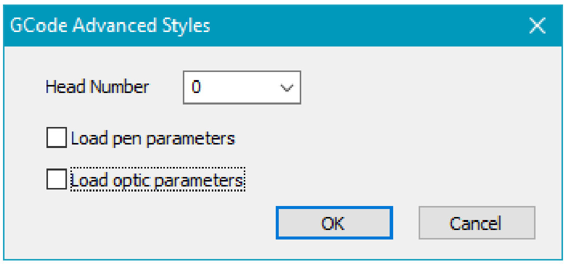


Figure 293: Advanced Styles for CNC dialog

Head Number: For a MultiCard system, specify the head to which the pen and optic parameters should be loaded.

Load pen parameters: Is unchecked by default. If activated, pen parameters (Pen settings) are imported.

Load optic parameters: Is unchecked by default. If activated, optic parameters (Field settings) and MOTF parameters are imported.

Load pen parameters: the values of the following G-Code commands can be loaded to the pen parameters:
$SC_LASERPOWER
$SC_FREQUENCY
$SC_JUMPSPEED
$SC_MARKSPEED
$SC_JUMPDELAY
$SC_MARKDELAY
$SC_POLYDELAY
$SC_LASERONDELAY
$SC_LASEROFFDELAY
$SC_STDBYPERIODE
$SC_STDBYPULSEWIDTH

Load optic parameters:
- the values of the following G-Code commands can be loaded to the optic parameters:

$SC_FIELDXMIN
$SC_FIELDYMAX
$SC_FIELDYMIN
$SC_FIELDXMAX
$SC_FIELDXGAIN
$SC_FIELDYGAIN
$SC_FIELDXOFFSET
$SC_FIELDYOFFSET
$SC_FIELDAXISSTATE
- the values of the following G-Code commands can be loaded to the MOTF parameters:

$SC_LASERPORT
$SC_MOFCHAN
$SC_MOFFACTOR0
$SC_MOFFACTOR1

12.1.2.4 Advanced for DXF Version1

For files of format DXF Version1 (DXF Files *.dxf), the following *Advanced Styles* dialog is given:

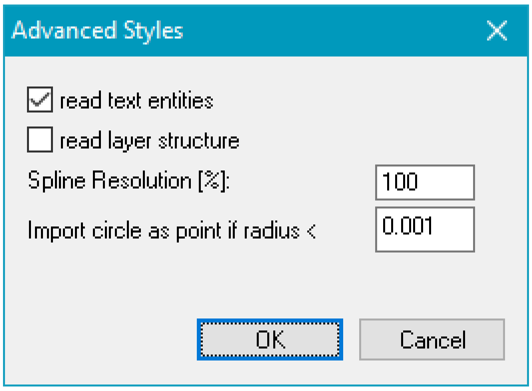


Figure 294: Advanced Styles for DXF dialog

read text entities: Is checked by default. If activated, text entities are imported.

read layer structure: Is checked by default. If different layers are existing within the imported dxf file, these layers are imported to different layers entities in View Level 2 of the entity list.

Spline Resolution [%]: Sets the resolution of splines, if imported. If the resolution is low, imported splines will look angular.

Import circle as point if radius < ...: A limit can be given up to which size a circle should be imported as a point.

During import, splines are split into upt to 200 polylines if the resulting polylines are bigger than a lower limit which depends on the field size.

12.1.2.5 Advanced for MCL

For files of format MCL, the following *Advanced Styles* dialog is given:

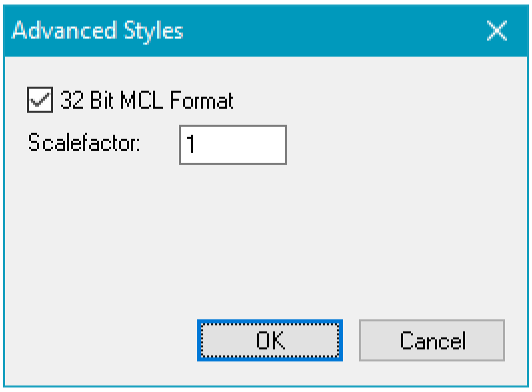


Figure 295: Advanced Styles for MCL dialog

32 Bit MCL Format: Is checked by default. If not activated, there is a limitation to a resolution of 16 bit ([-32768...32767]).

Scalefactor: Defines a factor for scaling of the MCL file. The default factor is 1.

12.1.2.6 Advanced for PLT

For PLT formats, the following *HPGL Advanced* dialog is given:

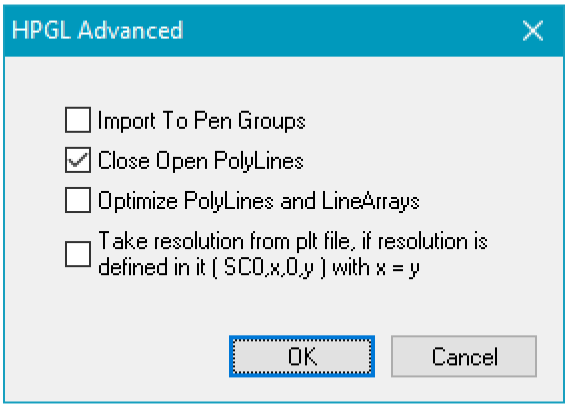


Figure 296: HPGL Advanced dialog

Import To Pen Groups: All objects painted with one pen are merged into one superior entity.

Close Open PolyLines: Is checked by default. Means that PolyLines are closed if there is a distance between start and endpoint which is smaller than the selected resolution.

12.1.3 Vector File Formats

DXF Files (Standard):

- No Text import. (To activate text import, use "read text entities".)
- Supports versions 10,12,13 and 14.

DXF Files Version 2:

- Supports all current variations of the DXF format (versions 2.6, 6, 9, 10, 11, 12, 13, 14, 15(2000), 18(2004)).
- Imports: Layer, line, arc, circle, ellipse, PolyLine, text (no reliable text position), vertex

Corel Presentation Exchange Files (CMX):

- Supported versions are 5, 6, 7, and 8.

Scalable Vector Graphics files:

- Imports: group, circle, line, polyline, Polygon, Polydraw, arc, (text).

AI Files:

- Supported Adobe versions are:
  - o Adobe Illustrator: 5, 7, 8, 9, 11, 12
  - o AI8_CreatorVersion: 9.0, 10.0, 11.0.0, 12.0.1, 13.0.0

12.1.4 SVG Import Supported Objects

Object	Load	Save
Animation	–	–
Circle	✔	✔
Class Styles	✔	–
Clipping, Masking, and Compositing	–	–
Color	✔	✔
Ellipse	✔	✔
Elliptical Paths	✔	–
Filling, Stroking, and Marker	✔	✔
Filter Effects	–	–
Fonts	✔	✔
Gradients and Patterns	–	–
Group Object	✔	–
HPolyBezier	–	✔
Interactivity	–	–
Lines	✔	✔
Linking	–	–
Paths	✔	–
PolyBezier	–	✔
PolyDraw	✔	✔
Polygons	✔	✔
Polyline	✔	✔
Raster	✔	✔
Rectangle	✔	✔
Spline	–	✔
Text	✔	✔
Transformation Matrices	✔	–
Vertex	–	–

Table 32: SVG Supported Objects

12.2 Export

Menu bar → File → Export opens a Dialog Box to export selected entities into graphic files.

Available export formats are:

Extension	Description
*.saf	SAF is a SCAPS archive.
*.plt	Hewlett-Packard Graphics Language (HPGL) Plotter File (PLT) is a language format for printing line drawings, specifically designed for 2D plotters.
*.cnc	CNC G-Code is a language to control CNC (Computer Numerical Control) machines. Option Flash license is required for cnc export.

Table 33: Available export formats

13 Mark

Start: Opens the Mark Dialog to control the mark process.

Trigger: If the trigger is connected (activate in Settings - System - Optic), the device output window (trigger dialog) will open. *

MarkHatchesFirst: If this item is checked the hatches will be marked first.

MarkOnlySelected: If this item is checked, a mark process started by *Menu bar → Mark → Trigger* will only mark the selected entities.

Reset...:

ResetSequence: Resets/Restarts a mark sequence. Please refer to section Entity Info to get details on how one can define a mark sequence.

ResetSerialNumber: Resets all Serial Numbers of the job to their defined start values.

ResetCounter: Resets all Mark Statusbar information.

Sequence/SerialNumber: Executes ResetSequence and ResetSerialNumber.

Counter: The Counter menu handles all counter related functionality:

SetQuantities...: Opens a dialog where the quantities can be defined that is how many times a mark process is repeated.

Edit Counter...: Here a starting value for the counter can be set.

Increment:

Sequence: Increments the current sequence of the job. Please refer to section Entity Info to get details on how one can define a mark sequence.

SerialNumber: Increments all SerialNumbers of the job. If global sequences are used, the ones from

SerialNumber by...: Increments all SerialNumbers of the job by an arbitrary number.

Sequence/SerialNumber: Increments the sequence and serial numbers of the current job. Please refer to section Entity Info to get details on how one can define a mark sequence.

Decrement:

Sequence: Decrements the current sequence of the job. Please refer to section Entity Info to get details on how one can define a mark sequence.

SerialNumber: Decrements all SerialNumbers of the job.

SerialNumber by...: Decrements all SerialNumbers of the job by an arbitrary number.

Sequence/SerialNumber: Decrements the sequence and serial numbers of the current job. Please refer to section Entity Info to get details on how one can define a mark sequence.

Sequence/SerialNumber: Updates the sequence and serial numbers of the current job. Please refer to section Entity Info to get details on how one can define a mark sequence.

TimeInfo...: Shows the TimeInfo dialog below.

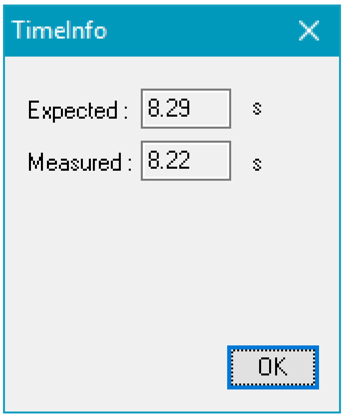


Figure 297: TimeInfo Dialog

Expected: Calculates the expected duration of the mark process in seconds depending on the job and settings information.

Measured: Displays the measured duration of the last mark process.

Preview: See chapter Mark Preview.

NOTE

** If Splitting is active with a defined start position, the Mark Trigger window will not be available. An external trigger can be used in the [Mark Dialog](#).*

13.1 Mark Dialog

To open the Mark Dialog go to *Menu bar → Mark → Start*.

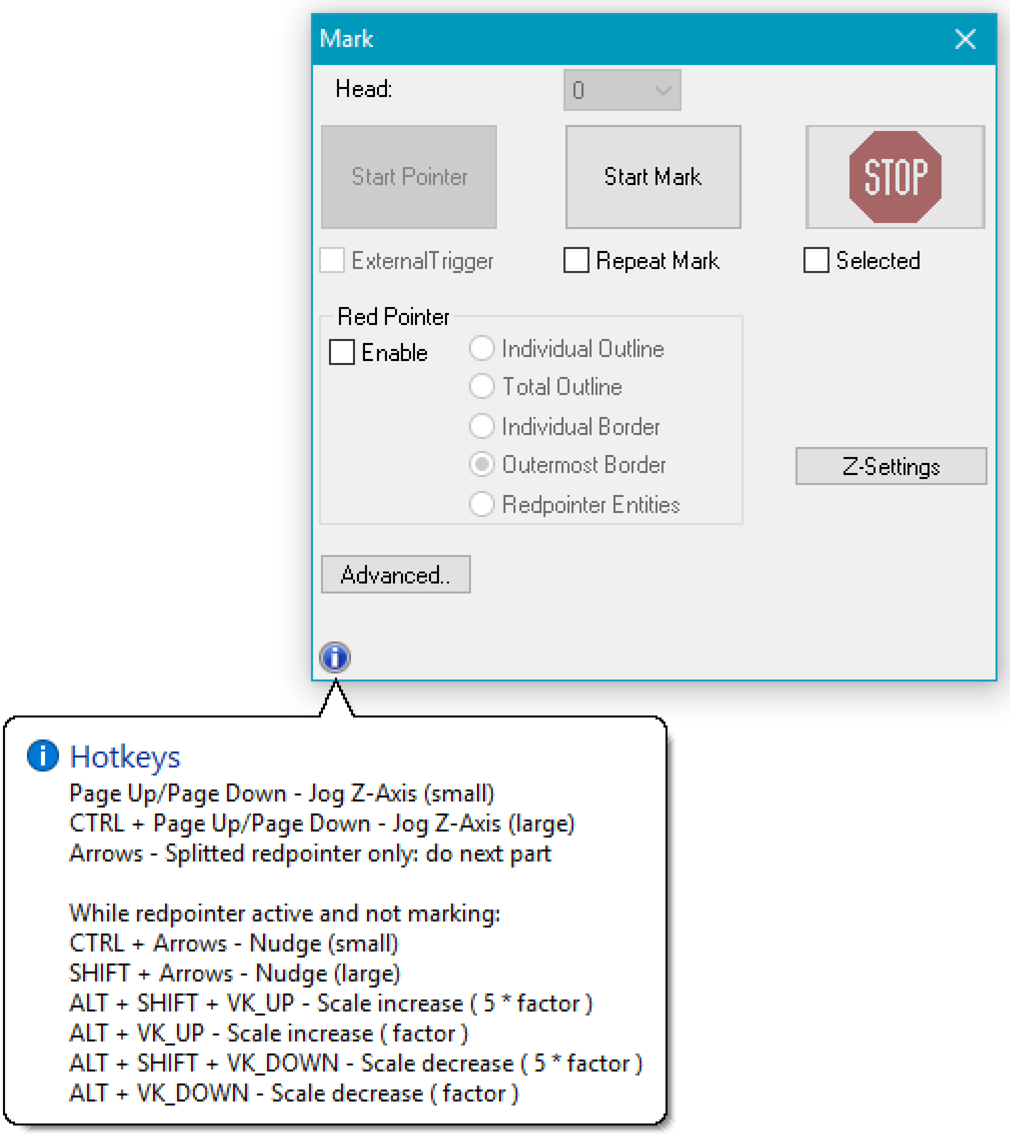


Figure 298: Mark Dialog

Each time a marking operation is triggered (by clicking Start Mark, by pressing F1 or by an external trigger), all objects to be marked are sent to the scanner control card and marked.

Head: This field is active if there is more than one scan head. In this case one of the available heads can be chosen to be used for the marking operation.

Start Mark (the yellow laser warning sign): Starts the mark process.

Stop (the red stop sign): Stops the mark process.

External Trigger: If checked the mark process can be started by an external trigger signal received by the optic device. If MOTF is enabled, this checkbox is not available (due to software jitter). Please use Mark - Trigger instead.

Repeat Mark: If checked the mark process will be repeated.

Selected: If checked only selected entities are considered in the mark process.

Red pointer: See chapter Red pointer.

Advanced: See chapter Red pointer.

Z-Settings: See chapter Deep Engraving.

Hotkeys: Some extra functionality to jog the z-axis or to nudge or scale entities. Controlling the Jog Z-Axis via (CTRL +) Page Up/ Page Down is only available for motion controller type 14.

If you use short keys for motion control make sure, that you do not use the hot keys. If short keys are assigned hot keys, they are overwritten in the mark dialog.

13.1.1 Red pointer

Start Pointer: Starts the red pointer. This button is active only if the red pointer is enabled. For the red pointer *pen #255* is used (see chapter Mark Settings). While the red pointer is outlining, scale and nudge can be done with the short keys listed below.

- **Ctrl + arrow key:** Nudge
- **Ctrl + arrow key:** 5 * Nudge
- **Alt + up/down arrow key:** Scale
- **Shift + Alt+ up/down arrow key:** 5 * Scale

Nudge Step and Scale Factor can be defined within Settings → System → View.

The StopKey in Settings → System → Short Keys should be set to ESC/Space.

Enable: Enables the red pointer.

Individual Outline: If checked each individual outline of the objects is drawn.

Total Outline: If checked the complete outline of all objects will be drawn by the red pointer, otherwise only the outline of one object will be drawn.

Individual Border: If checked, the red pointer draws the real geometry of the object.

Outermost Border: If checked, the red pointer draws the outermost individual borders of objects.

Red pointer Entities: If checked, only red pointer entity checked object(s) will be drawn. An object can be checked as red pointer entity in the entity list window.

Advanced: The following dialog opens after clicking on the Advanced button.

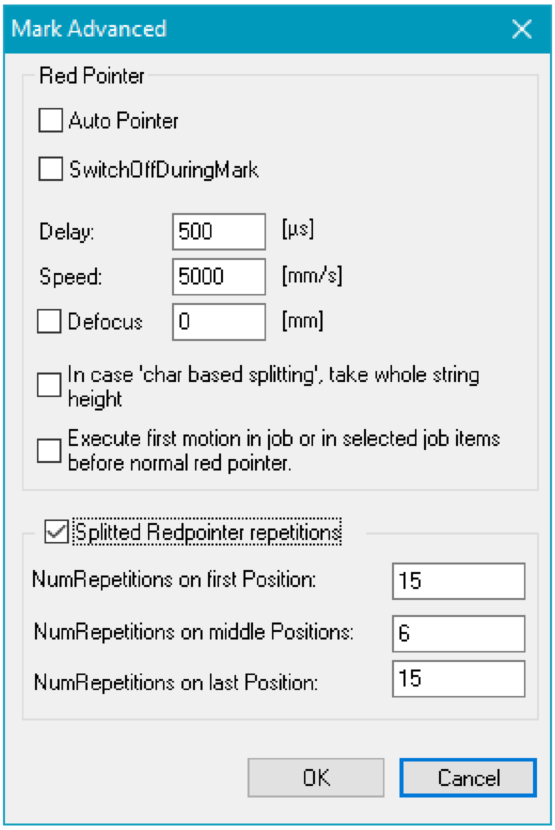


Figure 299: Mark Advanced Dialog

Auto Pointer: If enabled, the red pointer starts automatically after each marking showing the red pointer shape of the next marking sequence.

SwitchOffDuringMark: If enabled, the red pointer is switched off during mark.

Delay: Value determines delay of the red pointer. This value will overwrite the delay values (jump, mark and poly delay) in pen 255 (red pointer pen).

Speed: Value determines speed of the red pointer. This value will overwrite the speed values (mark and jump speed) in pen 255 (red pointer pen).

Defocus: This activates a focus shift on the Z-Axis.

In case 'char based splitting' : If enabled, the height of the whole string is taken for outlining with the Red Pointer. See example in Fig.300:

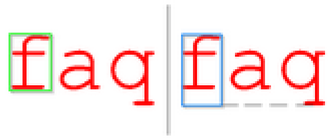


Figure 300: Green for flag disabled, blue for flag activated.

Execute first motion in job or in selected job items before normal red pointer: If enabled, the first ScMotionControl entity of the job will be executed when the Red Pointer start button is pressed.

Split Red Pointer repetitions: Can be checked only if entity is a split entity. You can set for first, middle and last split parts in between a red pointer repetition count. If unchecked (default), the red pointer runs endless on first split part. If the entity is not a split entity, this option is deactivated. This settings are necessary for the correct Red pointer function int he splitting mode.

NOTE

Split Red Pointer repetitions settings are necessary for the correct Red pointer functionality in the splitting mode.

13.2 Trigger Dialog

Trigger: Sends all selected (the selected definition is set in Mark Dialog) entities to the optic device. The controller board lights up each time it receives a trigger impulse. If this menu item was selected a dialog window with a Stop-button appears as long as this trigger state is active and as long as the program waits for an external trigger to start marking of the current job. Depending on the type of scanner card beside the mark start trigger a mark stop signal can be sent to stop the execution of a marking process. In this case the window will be closed. The same happens when the Stop-button of that window is pressed. Here it is assumed that something happened that required immediate stopping of the marking process (e.g. because some kind of special maintenance was necessary). Such events have to be acknowledged by the user explicitly by selecting this menu item again when the marking process has to be restarted using the external trigger.

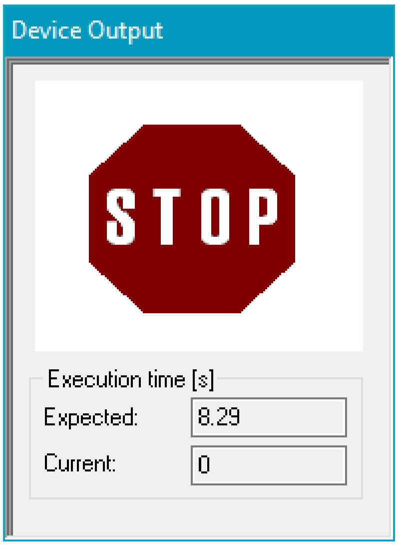


Figure 301: Mark Trigger Dialog

NOTE *The trigger needs to be activated in Settings - System - Optic.*

13.3 Mark Preview

The mark preview generates a preview of the actual job. Generate the preview by clicking *Menu bar → Mark → Preview*. Then, to switch to the Preview Window to see your marking, click *Menu bar → Window → Preview*. The preview window looks like the following:

13.3.1 Preview Window

The Preview Window shows the line outputs of one list execute. A line in this context is a straight line connecting two points. Depending on the line type the lines are drawn in different colours as shown in the screenshot below. The color of the selected line is white, the others lines are colored according to the OpticModule Properties.

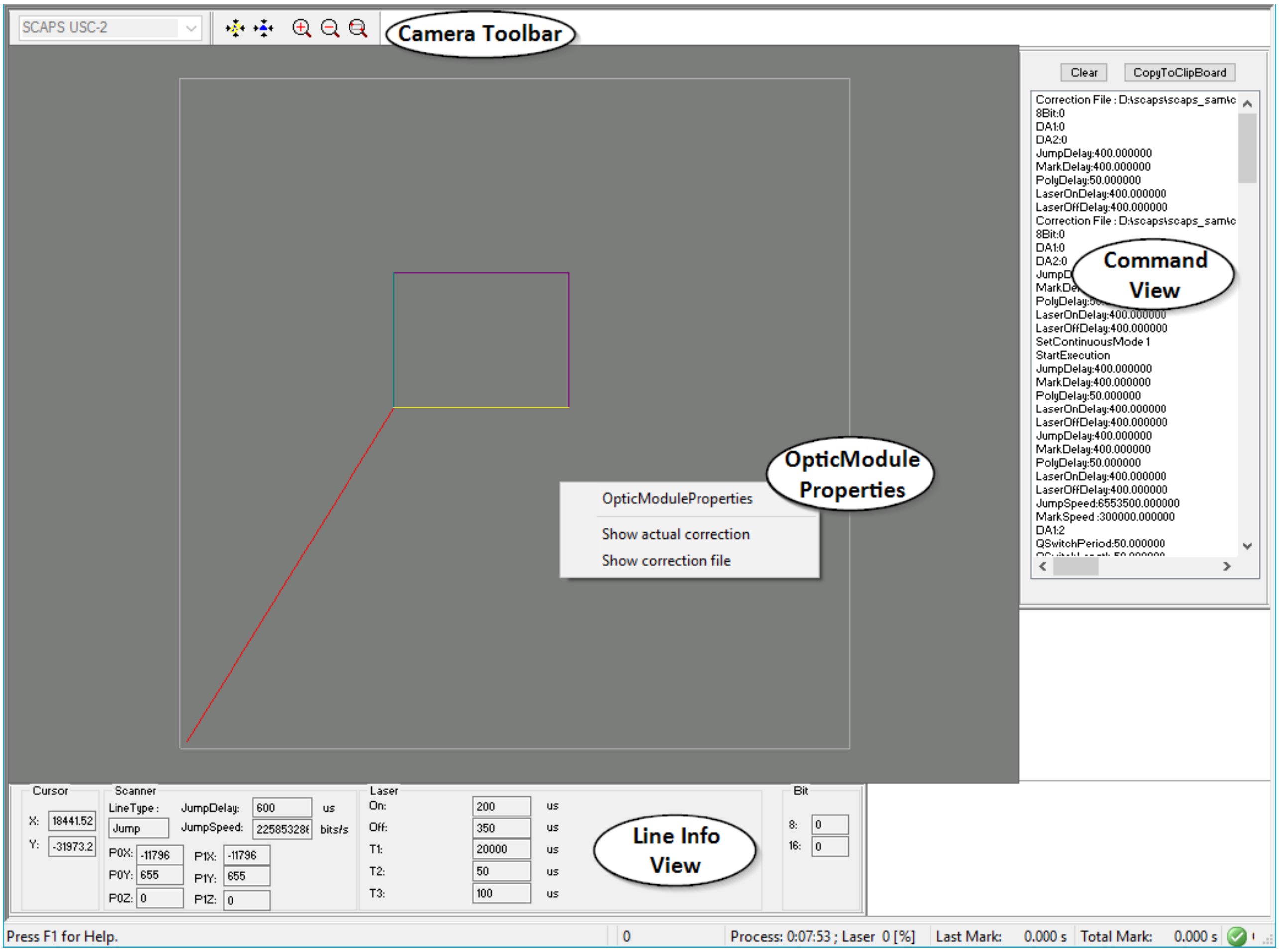


Figure 302: Mark Preview Window

Toolbars: The Preview Window has its own toolbar, because the standard toolbars belong to the Main Window and are not active while the Preview Window is in the foreground. The Camera Toolbar is just the *light* version of the Main Window's *Camera Toolbar*. So refer to chapter Camera Toolbar.

Views: Command View and Line Info View

OpticModule Properties: When clicking with the right mouse button in the Preview Window, the OpticModuleProperties will appear.

13.3.1.1 Command View

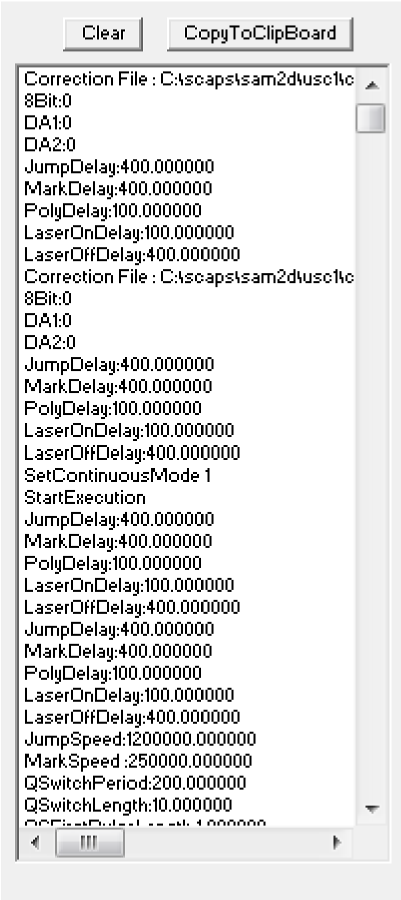


Figure 303: Command View

The Command View displays the first 1000 commands that are being sent to the preview window in a list box.

The displayed commands are compatible to the commands sent to the controller card in hardware mode.

If the button *CopyToClipBoard* is pressed, the content of the list box is copied to the clip board. This makes it easier to search for special commands.

NOTE

This box is only enabled in simulation mode.

13.3.1.2 Line Info View

The Line Info View shows the line settings for the line to which the mouse pointer is moved. The specific line will be highlighted. This box is enabled in simulation mode only.

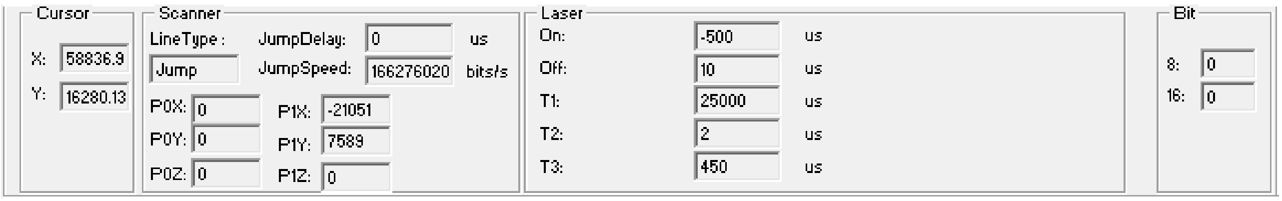


Figure 304: Line Info View

Cursor: Current cursor position in bits.

Line Type: Line type of the marked line like Jump, Mark, PolyA, etc.

P0X-P1Z: Start- and end-point co-ordinates of the selected line.

Delay: Scanner delay at the end of the line.

Speed: Speed of the scanner.

Laser: Laser on/off delay settings during execution of this line and additional the parameters T1 to T3.

13.3.1.3 OpticModuleProperties

If MultiHead license is active: By clicking with the right mouse button in the preview window the following dialogs are being opened:

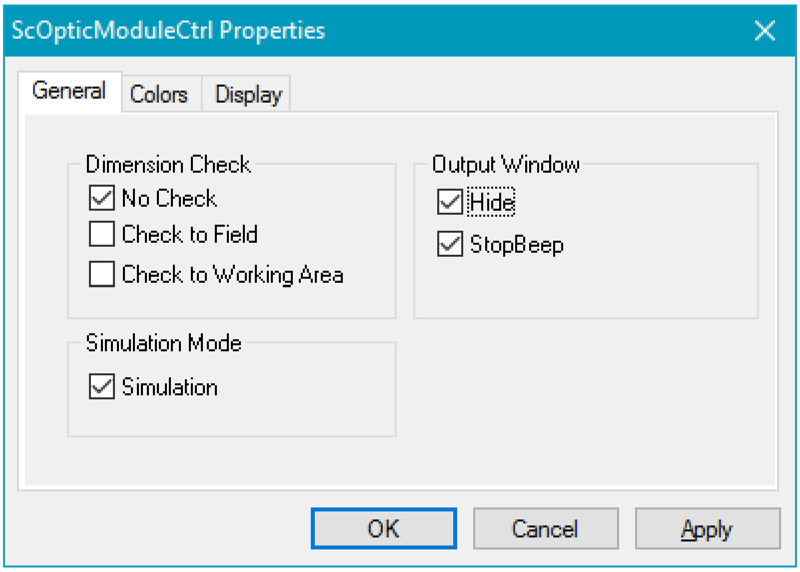


Figure 305: General of ScOpticModuleCtrl

Dimension Check: With these options, the user can check whether all the entities are inside of the Field or Working area before marking is started. If No Check is chosen, the marking will be executed until it gets out of the boundary.

Simulation Mode: This function is only designed for a user who has his own simulation application. Simulation Mode means there is no hardware output. The whole marking process will be simulated by software only, for example the time of Mark could be evaluated.

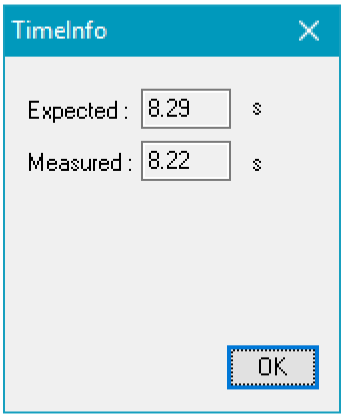


Figure 306: Time Info

NOTE

The Simulation Mode will be automatically switched off by clicking on Mark in mark-dialogue of SAMLight.

Output Window: This is also only designed for a user who has his own Start-Mark-Application. The output window refers to the Mark trigger window. In SAMLight, it is default that the mark trigger window is being hidden. But in the user's own application, he could set it up here and decide if he wants a beep when he clicks stop.

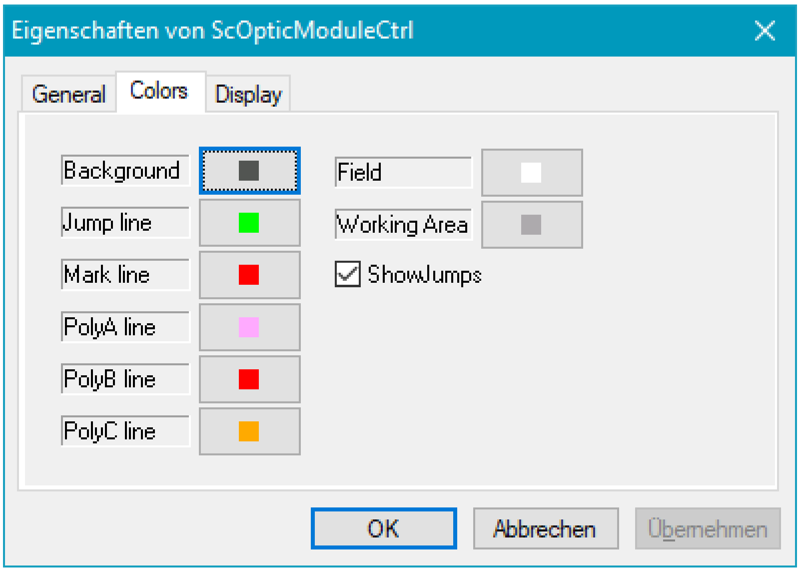


Figure 307: Colors

In the Window/1 Preview, click Mark → Preview, then the entities in Window/2 Main are plotted in different colors, which are defined in this dialog.

Default preview colors:

Line type	Color	RGB HEX	RGB DEC
Background		555555	085 085 085
Jump line		00ff00	000 255 000
Mark line		ff0000	255 000 000
PolyA line		ffaaff	255 170 255
PolyB line		ff0000	255 000 000
PolyC line		ffaa00	000 170 170
Field		ffffff	255 255 255
WorkingArea		aaaaaa	170 170 170

Table 34: Default preview colors

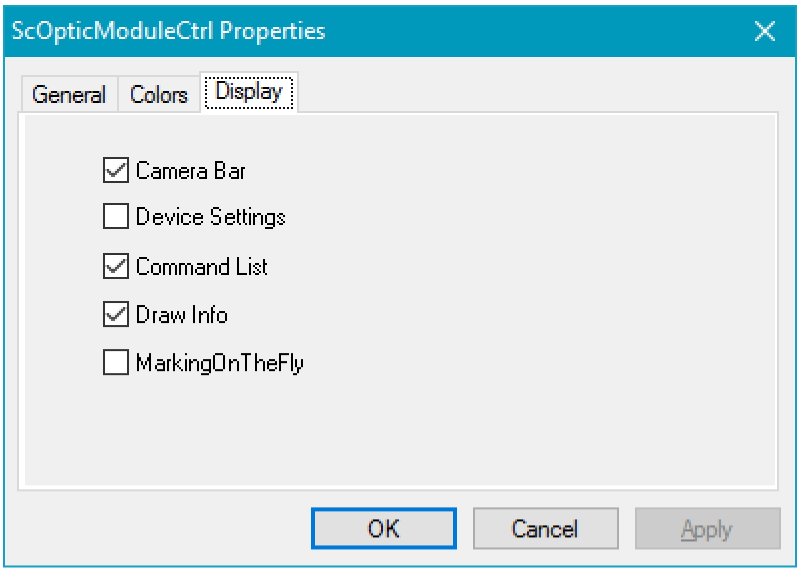


Figure 308: Display

With these check boxes, the user can activate or deactivate some toolbars or windows in the Preview Window.

13.4 SAMLight Job IO Selection

If the I/O mode is set to SAMLight Job IO Selection, loading of a job file can be invoked via input control. In this mode - according to the bit pattern of the external inputs and according to the related number that is created out of this pattern - a predefined job is loaded from disk and available for marking afterwards. If a job is loaded that way, it has to be marked using an external trigger. The appropriate option for the external trigger start has to be set before.

There are six hardware inputs, whereof the last 4 inputs define a number which identifies the job file:

Input	Function
Opto_In_0	Start Mark (Trigger)
Opto_In_1	Stop Mark (external stop)
Job selection inputs	see table 36

Table 35: Input bits used to control the process

Furthermore the extensional board (available on USC-2 and USC-3) gives a possibility to use the 8 digital inputs to load job files, see table 36.

Input Type	Job selection inputs	Max. jobs
SAMLight Job IO Select	USC: Opto_In_2..5 RTC: Digital_In_0..3	15
SAMLight Job IO Select Ext	USC-2/-3: Digi_In_0..7	255

Table 36: Job selection inputs for SAMLight Job IO Selection modes

The job files that will be used should be saved in the folder *<SCAPS>\jobfiles (in the installation directory)* and must have the following name structure: **jn_nnnn.sjf**

Here *jn* stands for any freely definable job name and *nnnn* stands for the decimal job number that is related to the input pin pattern (see table 37).

Opto_In_5 (MSB)	Opto_In_4	Opto_In_3	Opto_In_2 (LSB)	Job name
0	0	0	0	Empty Job
0	0	0	**1**	jn_00**01**.sjf
0	0	**1**	0	jn_00**02**.sjf
0	0	**1**	**1**	jn_00**03**.sjf
0	**1**	0	0	jn_00**04**.sjf
0	**1**	0	**1**	jn_00**05**.sjf
0	**1**	**1**	0	jn_00**06**.sjf
0	**1**	**1**	**1**	jn_00**07**.sjf
1	0	0	0	jn_00**08**.sjf
1	0	0	**1**	jn_00**09**.sjf
1	0	**1**	0	jn_00**10**.sjf
1	0	**1**	**1**	jn_00**11**.sjf
1	**1**	0	0	jn_00**12**.sjf
1	**1**	0	**1**	jn_00**13**.sjf

Opto_In_5 (MSB)	Opto_In_4	Opto_In_3	Opto_In_2 (LSB)	Job name
1	**1**	**1**	0	jn_00**14**.sjf
1	**1**	**1**	**1**	jn_00**15**.sjf

Table 37: Job selection table

With SAMLight Job IO Select Ext, the job numbers are coded according to the same principle with Digi_In_0 (LSB) and Digi_In_7 (MSB).

NOTE *Job number 0 is defined as an empty job. Job files with this number will be ignored.*

Process control output on USC-cards are defined as follows:

Output	Function
Opto_Out_0	*MarkingActive a.k.a. Mark In Progress (MIP)*
Opto_Out_3	Software (Trigger) ready → *StartMark* can be set
Opto_Out_4	New job was loaded successfully, signal will be reset with next external start (*StartMark*)

Table 38: Output bits used to control the process.

NOTE *If the bit configuration will be changed during marking, the current marking process will be aborted.*

14 Splitting

This option allows you to split a job into pieces and then mark them part by part. That automatic split marking is useful e.g. for marking round objects or objects that are bigger than the working area. Jobs are always split completely as long as there are no entities marked as non-splittable.

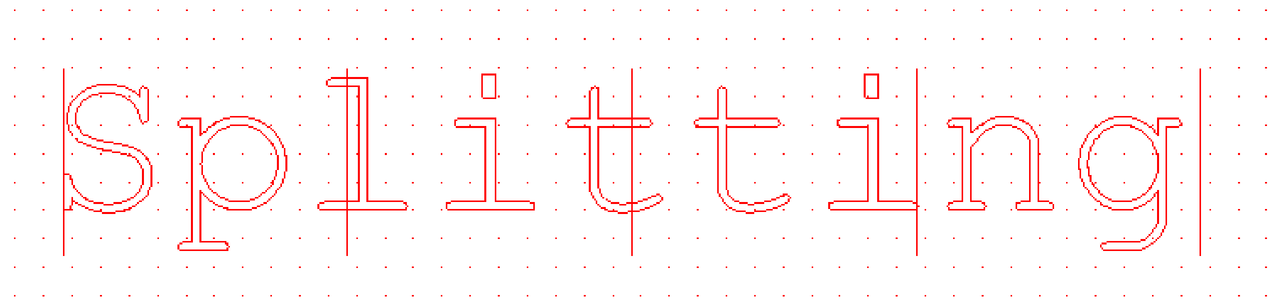


Figure 309: Example of a split job

The Splitting Dialogs can be reached via the Extras Toolbar or by selecting the appropriate menu item. If the extra toolbar is not visible, it can be activated at *System → Settings → View → Toolbars → Extras.*

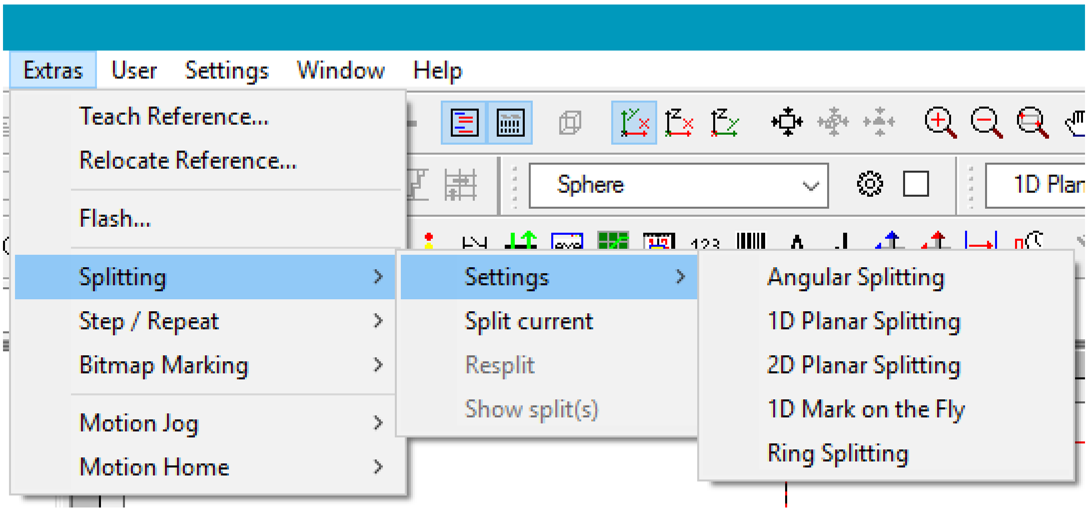


Figure 310: Splitting modes

Splitting Mode	Splitting Option
Angular	• Vector Marking - Fixed Split Size • Vector Marking - Entity Based Splitting • Vector Marking - Character Based Splitting
1D Planar	• Vector Marking - Fixed Split Size • Vector Marking - Entity Based Splitting • Vector Marking - Character Based Splitting
2D Planar	• Vector Marking - Fixed Split Size • Vector Marking - Entity Based Splitting
1D Mark on the Fly	• Vector Marking - Fixed Split Size • Vector Marking - Entity Based Splitting • Bitmap marking
Ring	• Vector Marking - Fixed Split Size • Vector Marking - Entity Based Splitting • Vector Marking - Character Based Splitting • Bitmap marking

Table 39: Splitting modes and options

You can save the splitting settings within the job file. Please refer to the Job Properties Topic for more information.

Rotational marking (*Angular Splitting* mode) can be done by performing the following steps:

- Enable external motion control to perform the automatic rotation of the round object that has to be marked.
- Define the diameter of the object.
- Define a rotational angle that describes the segment that should be marked at the same time.
- Define the motion controller axis that has to be used for the rotation movements. That axis only defines the drive that has to be used for the motion, it does not influence the axis the job is split for.
- Turn on the splitting mode to enable the current job for rotational marking.

For planar marking (*Planar Splitting* modes), the following steps are necessary:

- Enable external motion control to perform the automatic movement of the flat object that has to be marked.
- Define the size of a split part that has to be marked at the same time.
- Define the motion controllers axis that has to be used for the planar movements. This axis only defines the drive that has to be used for the motion. It does not influence the axis the job is split for.

- Turn on the splitting mode to enable the current job for marking it part by part.

For *Mark on the Fly* of split parts, the following steps are necessary:

- You need a *Marking on The Fly* featured license of the SAM software.
- Define the size of one split part that has to be marked on the fly.
- Turn on the splitting mode to enable the current job for marking it part by part.

The Motion Control for the first two modes can be activated at *Menu Bar → Settings → System → Extras → Motion Control*. Additionally, the motion control configuration file has to be configured for the used controller as described in motion control .

If all of the Rotation Axis or Motion Axis radio buttons are disabled, there is a misconfiguration with the motion controller and the application will be unable to perform the correct rotational or planar movements.

To activate splitting mode go to *Extras → Splitting → Split current* or use the related icons in the Extras Toolbar . This mode can be turned off by clicking on this menu button again. Then the job is restored to non-split mode.

When the splitting of a job was successful, its appearance in the *View 2D* is changed. Now there are several additional lines. They are the cutting edges that have been created according to the values *Total Diameter* and *Splitting Angle* (in angular mode) or *Splitting Size* (in planar mode). These splitting lines can be changed to optimize the result. Click on such a line and enter the object level 2 using the level toolbar. Now you can select a single splitting line. To move such a line, click on the small blue box and then drag it in the desired direction. This functionality is identical to that of the normal geometries, with the sole exception that a rotation or scaling of the splits is not possible.

When such a line is moved beyond one of the neighbouring cutting lines, it is automatically forced to a position back within its two neighbours. Interleaving the cuts that are represented by these lines is an invalid operation. If there are parts of the job that are located outside of the outer cutting lines after editing, they will not be marked.

After dragging a cutting line to a new position, the split job is recalculated automatically. This operation may require some seconds, depending on the complexity of the job. Marking such a job can be done like before. Here, all the split parts are marked one by one until the complete job is done. Editing the cutting lines resets the current marking position so that the next marking operation starts with the first segment of that job.

Under *Extras → Splitting → Show split(s)* you can get a preview of all the created split parts.

To get the correct marking result, it is very important that the axes of the motors correspond to the axes of the split job. Normally, they should both form a right handed coordinate system.

If a motor movement is desired, that is independent of the splitting, all motion control objects have to be defined as non-splittable entity, see chapter Entity List.

For the correct Red pointer function in the splitting mode, you have to make settings for Split Red Pointer repetitions in the Mark Dialog --> Advanced...

14.1 Splitting Settings

Each splitting mode has its own splitting settings dialog. The splitting settings can be opened by two different paths. Via *Extras → Splitting → Settings → Splitting Mode.* Alternatively, the splitting settings can be opened via the extras toolbar. The following documentation contains all relevant parameters for all splitting modes. They are divided in the *general* and *mode* section.

General:

Splitting Option: Splitting is divided into splitting modes and splitting options. In total five different modes and four different options are available. Those are listed in the several splitting mode subchapters and explained in the splitting options. The user can choose the splitting option in the respective splitting settings dialog. It can be set while selecting the wanted splitting option from the drop down menu.

Axis Speed: This is a general parameter of planar [mm/s] and angular [deg/s] operation modes. It defines the speed used for the connected drive between the split parts. The speed defined here overrides the speed defined in the *Control* submenu of the entity property page.

MOTF Speed: The moving speed can be selected. It can maximally be the *Maximum MOTF Speed.* At this point it is important, that the moving speed is relative to the marking speed of the current split part, so that it is finished before it is out of the scanning field.

Maximum MOTF Speed: Calculates internally the maximum possible MOTF speed for the marking. This parameter is read-only an can not be edited.

Split Repetitions: Defines how often the marking of each split part is repeated before the motion device moves to the next split part. The individual Mark Loop Count values for each entity is ignored in splitting mode. If repetition for each entity is wished, one can use Pen Path instead.

Job Repetitions: After marking the last split part, the scanner starts again with the first part. This process is repeated according to the number of this total repeat value.

Split Overlap: With this option you can increase the area of a single split part. The overlap is done in positive splitting direction. Only constant vectors that end within the added space are marked. For an example see Visualization of Special Options.

Group Width: It is possible to group entities to decrease the marking time. The *Rotation Axis* dimension of the group outlines will not exceed this value. Entities with a greater *Rotation Axis* dimension than the grouped entities width will not be grouped with other entities. Not available if *Top Level Entities as Split Parts* is checked.

Start Position: Enable the *Activate* checkbox to define start values for the splitting. Define the start position of the splitting routine for the *axis*. The value you put in this field defines the absolute position. With the button *Get Current Position,* it is possible to set the value of the current motion controller as start value. When using a defined start position, the *Mark Trigger* window is not available.

Ring Center Position: Define a starting position for the ring center in X- / Y- direction. (available for ring splitting)

Job Repetitions Gap: Here you can define a gap between two job repetitions. (only 1D MOTF splitting)

Workpiece Movement: Invert all movements, the workpiece is moving instead of the scan head.

Automatic Start Position: Provided that the split is centered on the working area and that the motion axes are on position 0,0. The user does not have to specify where the split should begin. The motion device moves automatically to the start position of the split. (only 2D planar splitting)

Move Back to Start Position: The scan head returns to the starting position after each marking. If unchecked, the scan head stops at the position where the marking finished and the next marking will start from there.

Keep Mark In Progress Active: Without this checkbox, the *OptoOut0* (*DigiOut0* for RTC cards) is set to 1 every time a split is marked and it is set back to 0 during the axis movement between the split parts. If this checkbox is activated, the marking signal via *OptoOut0* (*DigiOut0* for RTC cards) would stay at 1 as long as the complete job (including all splits and axis movement) is marked.

Cog Vectors in Overlap Area: If the entity contains many vectors that reach into the next split part, the quality of the marking sometimes depends on the position of each split. If all lines are split on a uniform edge, heating issues can occur and the marking quality may suffer. The marking quality can be enhanced if those lines are split on a cogged edge instead of on a uniform one. Activate this checkbox to split with a cogged edge. For an example see Visualization of Special Options.

Avoid Unnecessary Splits in Overlap Area: This option optimizes the positioning of the split lines to accelerate the marking process by avoiding unnecessary splits. The order of the split marking process can be changed by this option. For an example see Visualization of Special Options.

Do Not Recalculate Split Lines: Re-split does not change the split lines. The entities can be exchanged.

Center Whole Job on Current Axis Position: Start position of the marking is always the same while using this option. The first split part will be always marked at the same position.

Skip empty Splits: If this option is enabled then the scanner will not move to empty split parts. (only 2D planar splitting)

Top Level Entities as Split Parts: Top level entities will not be split into smaller parts, but will be marked as a whole. This checkbox is always active as the entity based splitting mode only marks top level entities uncut. This means that the top level entity needs to fit into the working area. Split parts will be set in between of top level entities.

Order by Start Border: The marking order depends on the position of the entity. The order in the entity list will be ignored. Example: If the split order is from left to right, then the entity which is most left positioned will be marked first.

Ungroup Text: The characters of a ScWinTextChars2D entity can be ungrouped. In this case, each character of a text is taken as an individual entity for the marking process.

Endless Job Repetition: All split parts are grouped and endless repeated. Enables the function *Job Repetition Gap* (only 1D MOTF splitting)

Center Split Parts: This option allows you to center all split parts.

Mode:

Axis: The axis that defines the splitting direction can be set here.

Horizontal Axis: The motion axis that performs the horizontal movements can be defined here. You cannot use the same axis as *Vertical Axis*.

Vertical Axis: The motion axis that performs the vertical movements can be defined here. You cannot use the same axis as *Horizontal Axis*.

Split Order: This button defines the splitting and marking order of the split parts. There are four possibilities: from bottom to top, from left to right, from top to bottom and from right to left. Switch between options by clicking on the button.

Split Size: The size of one single split part can be defined here. This size needs to be smaller than the working area in the same direction.

Split Width: You can define the horizontal split width here. The split width needs to be smaller than the working area in the horizontal direction and defines the resulting horizontal part size.

Split Height: You can define the vertical split height here. The split height needs to be smaller than the working area in the vertical direction and defines the resulting vertical part size.

Diameter: Define the diameter of the object that has to be marked. In case of a cylindrical object this would be the diameter of the base surface.

Height: Transverse dimension of the ring.

Margin: Shows an optical helping line that reduces the height. No influence on marking output.

Split Angle: This angle in degrees describes the size of one split part.

Repetitions over 360°: If the total diameter and the resulting surface of the cylinder is big enough, more than one copy of the job can be marked on the surface. The different markings are distributed homogeneously over the total generated surface of the cylinder.

Z Tilt Compensation: This feature is to be used with Ring splitting tilt compensation in Global settings → Extras → Splitting.

Keep Bitmap Size: Bitmap is not scaled in order to get a size that results in the specified marking angle, the source bitmap is left untouched.

Move Forward to Start Position: Move to start position after marking has been finished.

Tolerate Larger Split Jobs Than 360°: This allows a continuous rotation with an angle greater than 360°.

14.2 Splitting Mode

SAMLight supports five different splitting modes. The following subchapters will give an overview about the possible editable parameters. The explanation of each parameter is found in the Splitting Settings chapter.

- Angular Splitting
- 1D Planar Splitting
- 2D Planar Splitting
- 1D Mark on the Fly Splitting
- Ring Splitting

14.2.1 Angular Splitting

The Angular Splitting mode is for marking on round objects. The object is always rotated between two marked split parts to step by step mark the whole circumference of the object. To use this mode, at least one rotational (angular) axis of the connected *motion controller* needs to be configured.

NOTE

If an option is grayed out in the splitting settings dialog, it is not possible to use it with the current set splitting option. The availability can be looked up in the following table.

Checked but grayed out parameters are not used.

Angular Splitting Parameters	Vector Marking		
	Fixed Size Splitting	**Entity Based Splitting**	**Character Based Splitting**
Axis Speed	✔	✔	✔
Split Repetitions	✔	✔	✔
Job Repetitions	✔	✔	✔
Split Overlap	✔	—	—
Group Width [a]	—	✔	—
Start Position	✔	✔	✔
Workpiece Movement	✔	✔	✔
Move Back to Start Position	✔	✔	✔

Angular Splitting Parameters	Vector Marking		
	Fixed Size Splitting	Entity Based Splitting	Character Based Splitting
Keep Mark In Progress Active	✔	✔	✔
Cog Vectors in Overlap Area	✔	✔	—
Avoid Unnecessary Splits in Overlap Area	✔	—	—
Do Not Recalculate Split Lines	✔	—	—
Top Level Entities as Split Parts [b]	—	✔	—
Order by Start Border [b]	—	✔	—
Ungroup Text [b]	—	✔	—
Axis	✔	✔	✔
Split Order	✔	✔	✔
Diameter	✔	✔	✔
Split Angle	✔	—	—
Repetitions Over 360°	✔	✔	✔
Tolerate Larger Split Jobs Than 360°	✔	✔	✔
Move Forward to Start Position	✔	✔	✔

Table 40: Angular Splitting Parameter

[a]: This is not available if *Top Level Entities as Split Parts* is checked.
[b]: Cannot be used in combination with *Group Width.*

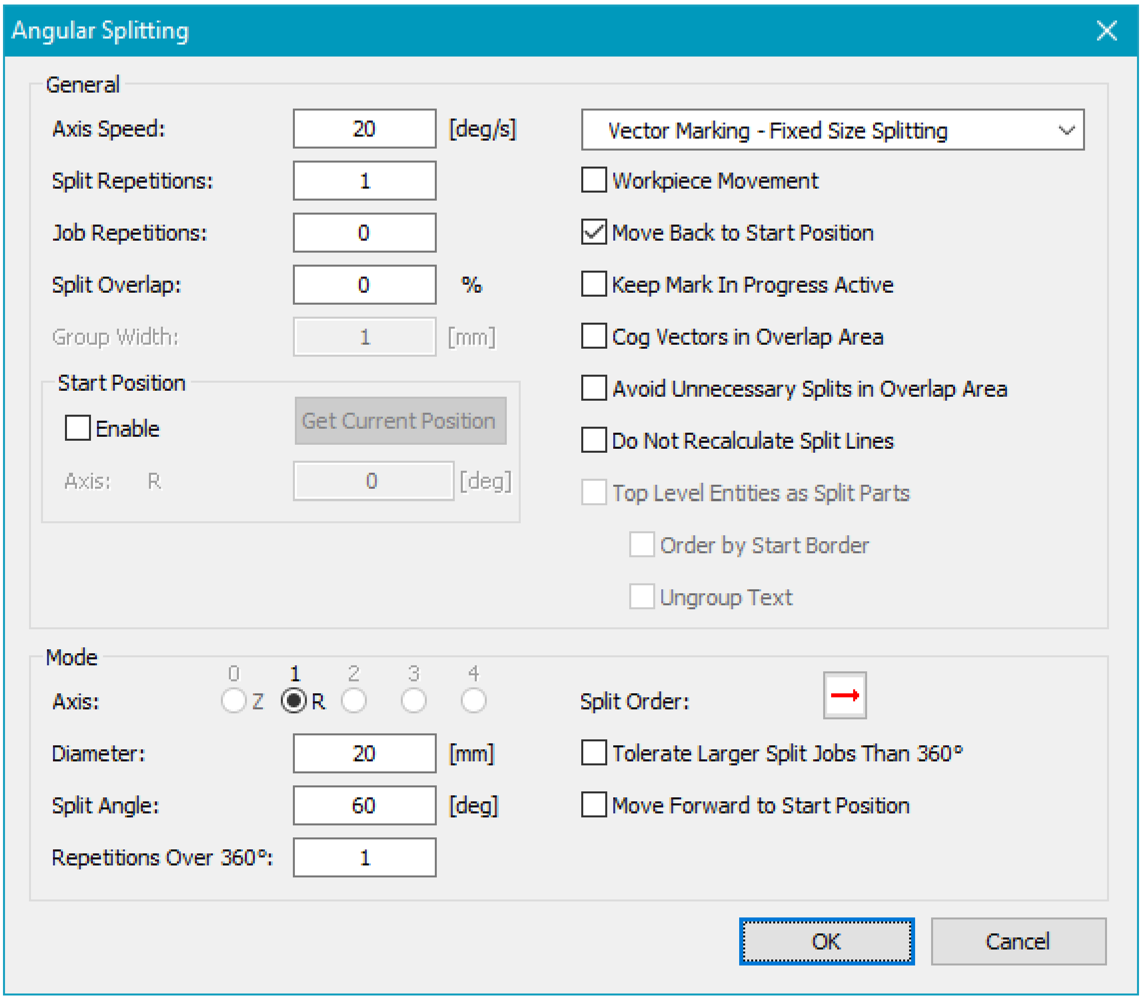


Figure 311: Angular Splitting Dialog

Figure 312: Rotation Object

14.2.2 1D Planar Splitting

The *1D Planar Splitting* mode is for splitting the current job in one direction, either X or Y, related to the contents displayed in the *View 2D*. To use this mode at least one axis of the connected *motion controller* has to be configured for planar operation.

NOTE

If an option is grayed out in the splitting settings dialog, it is not possible to use it with the current set splitting option. The availability can be looked up in the following table.

Checked but grayed out parameters are not used.

1D Planar Splitting Parameters	Vector Marking		
	Fixed Size Splitting	Entity Based Splitting	Character Based Splitting
Axis Speed	✔	✔	✔
Split Repetitions	✔	✔	✔
Job Repetitions	✔	✔	✔
Split Overlap	✔	—	—
Group Width [a]	—	✔	—
Start Position	✔	✔	✔
Workpiece Movement	✔	✔	✔
Move Back to Start Position	✔	✔	✔
Keep Mark In Progress Active	✔	✔	✔
Cog Vectors in Overlap Area	✔	✔	—
Avoid Unnecessary Splits in Overlap Area	✔	—	—
Do Not Recalculate Split Lines	✔	—	—
Top Level Entities as Split Parts [b]	—	✔	—
Order by Start Border [b]	—	✔	—
Ungroup Text [b]	—	✔	—
Axis	✔	✔	✔
Split Order	✔	✔	✔
Split Size	✔	—	—

Table 41: 1D Planar Splitting Parameters

[a]: This is not available if *Top Level Entities as Split Parts* is checked.
[b]: Cannot be used in combination with *Group Width.*

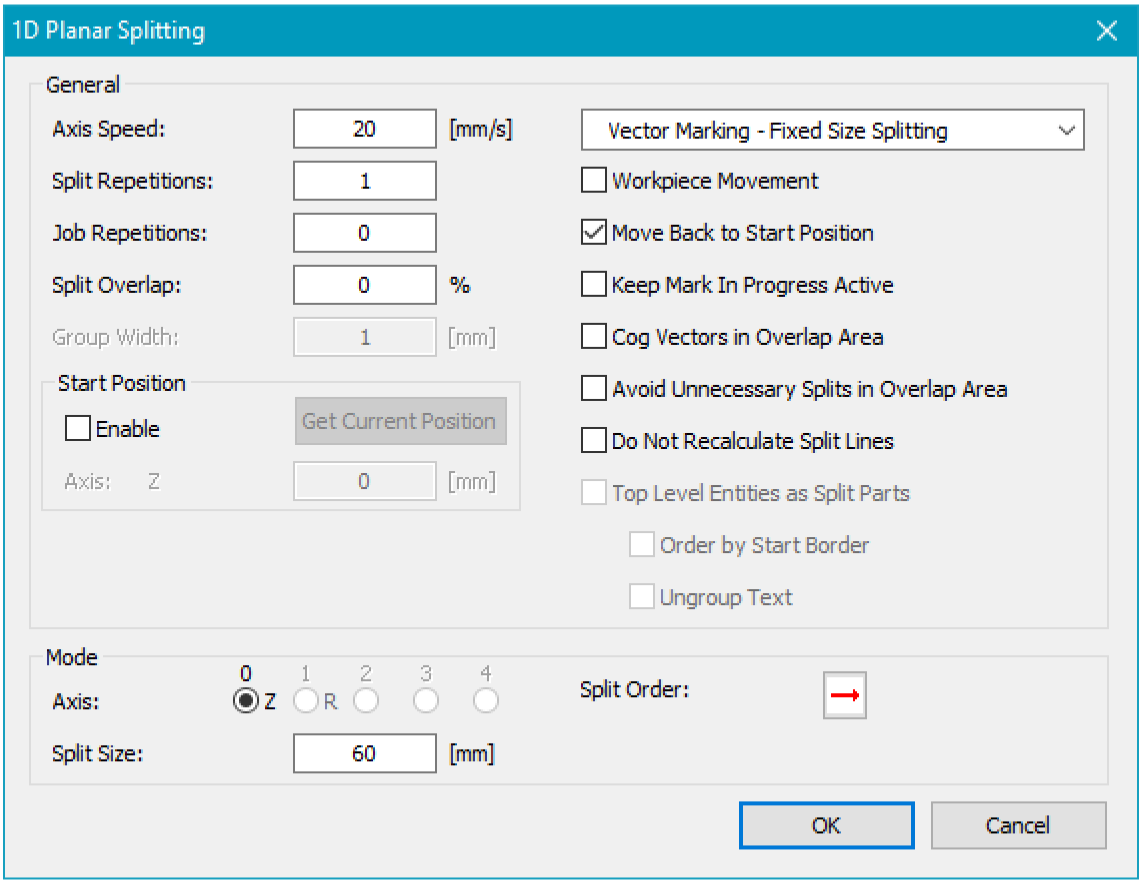


Figure 313: 1D Planar Splitting Dialog

14.2.3 2D Planar Splitting

The 2D mode is for splitting the current job in two directions. Therefore, no split axis definition is necessary here. When using this splitting mode, a XY-table should be used to place the working piece at the correct position. To use this mode at least two axes of the connected *motion controller* have to be configured for planar operation.

NOTE

If an option is grayed out in the splitting settings dialog, it is not possible to use it with the current set splitting option. The availability can be looked up in the following table.

Checked but grayed out parameters are not used.

2D Planar Splitting Parameters	**Vector Marking**	
	Fixed Size Splitting	**Entity Based Splitting**
Axis Speed	✔	✔
Split Repetitions	✔	✔
Job Repetitions	✔	—
Split Overlap	✔	—
Start Position	✔	✔
Workpiece Movement	✔	✔
Automatic Start Position	✔	✔

2D Planar Splitting Parameters	Vector Marking	
	Fixed Size Splitting	Entity Based Splitting
Move Back to Start Position	✔	✔
Keep Mark In Progress Active	✔	✔
Cog Vectors in Overlap Area	✔	—
Avoid Unnecessary Splits in Overlap Area	✔	—
Do Not Recalculate Split Lines	✔	—
Skip Empty Splits	✔	—
Top Level Entities as Split Parts	—	✔
Ungroup Text	—	✔
Horizontal Axis	✔	✔
Vertical Axis	✔	✔
Split Width	✔	—
Split Height	✔	—
Split Order/Preview	✔	—

Table 42: 2D Planar Splitting Parameters

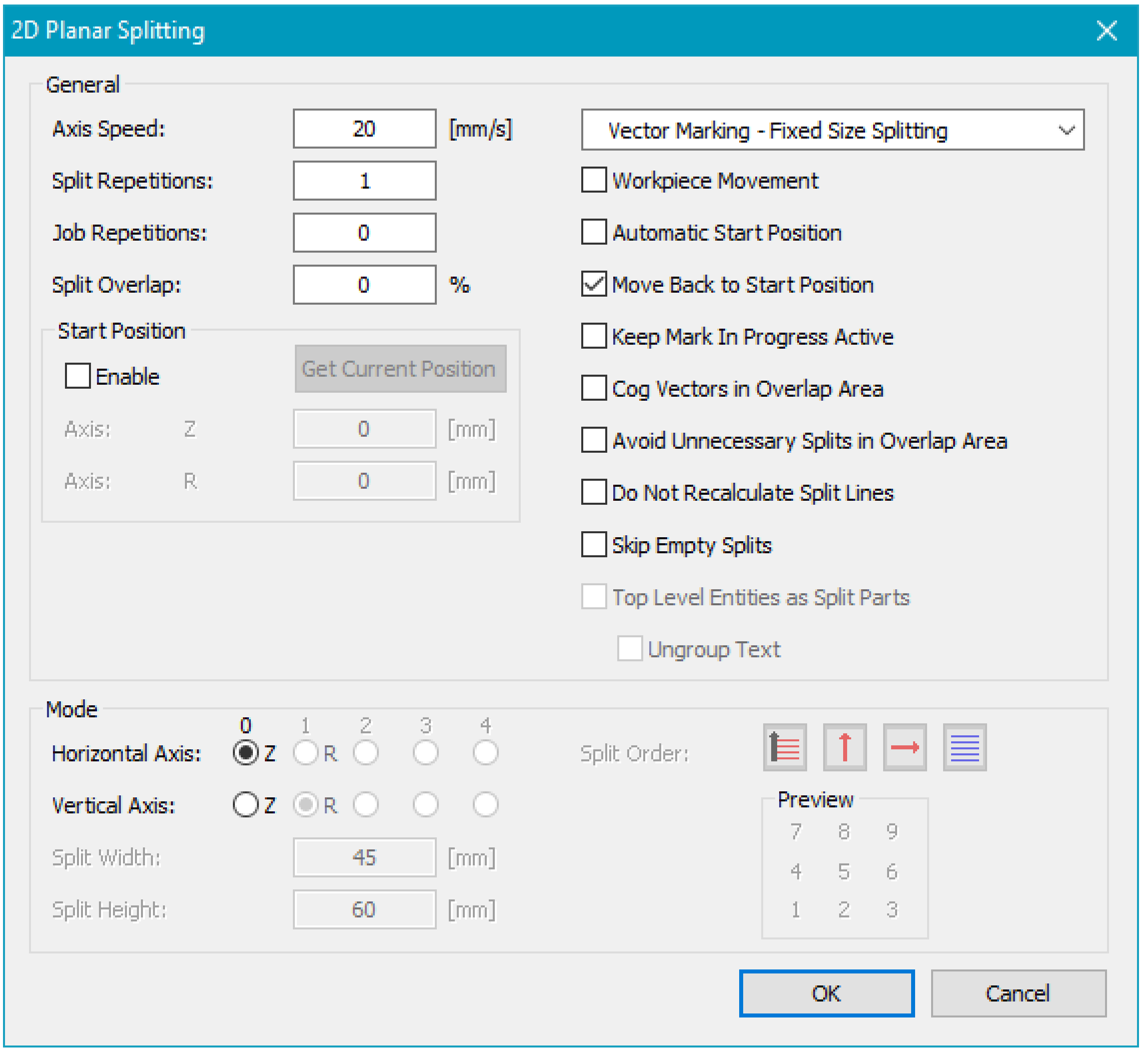


Figure 314: 2D Planar Splitting Dialog

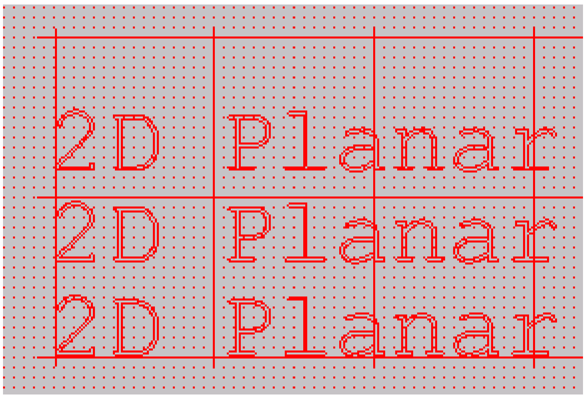


Figure 315: 2D Planar Split

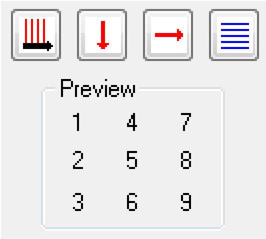


Figure 316: Order of split parts Marking

Here, the splitting routine performs marking one column after another. The scan head always goes back to the top, the vertical movement direction during a split is always the same. This can be inefficient depending on the velocity of the motion device.

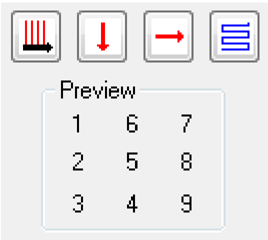


Figure 317: Optimized order of split parts marking

In this case, the splitting routine performs marking of the columns as well, but it will not move back to the vertical start position without marking. Instead, it will perform the marking when the vertical motion device is heading back to the border. This saves time.

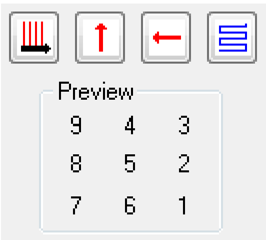


Figure 318: Defining the position of the split parts

With the two buttons in the middle, the position of the first split part can be defined. With the left button the order of the remaining split parts can be defined.

14.2.4 1D Mark on the Fly Splitting

This field is only available for USC cards and only editable if *Marking on the Fly* is enabled within the settings. The *1D Mark on the Fly* mode is similar to the preceding ones, except for the fact that the motion is not controlled by a connected *motion controller*. Here, the object is moved continuously by an external drive like it is known for "on the fly" applications in general.

WARNING *This feature is supported for every USC card, but only for the RTC5.*

NOTE *The license MOTF is required.*

If this splitting mode is active the marking via Start → Mark is disabled. This is because starting mark via the Mark Dialog would have a delay of about 200 ms and this would affect the MOTF marking result because marking would start to late. However, the Red pointer can be activated to see where the MOTF marking will occur. Marking can only be started via Mark → Trigger.

*Split View: Extras → Splitting → Show split(s) → enable "Original positions/ Positions as in job to mark", in this way, a split job file with motions is generated automatically and can be saved in an *.sjf file, which could be converted to *.unf file.*

If an option is grayed out in the splitting settings dialog, it is not possible to use it with the current set splitting option. The availability can be looked up in the following table.

Checked but grayed out parameters are not used.

The external trigger signal can be high during the complete marking process. In previous versions a raising edge of the signal started the process, but the signal had to be go low before the next split part is executed. Now SAMLight does not react to further external trigger signals (start signals)

All motion control objects have to be defined as non-splittable entity, see chapter Entity List.

1D MOTF Splitting Parameters	**Vector Marking**		
	Fixed Size Splitting	**Entity Based Splitting**	**Bitmap marking**
MOTF Speed	—	✔	—
Maximum MOTF Speed [a]	—	—	—
Split Repetitions	✔	✔	✔
Split Overlap	✔	—	—
Group Width [b]	—	✔	—
Job Repetition Gap [c]	—	—	—
Keep Mark In Progress Active	✔	✔	✔
Cog Vectors in Overlap Area	✔	✔	—
Avoid Unnecessary Splits in Overlap Area	✔	—	—
Do Not Recalculate Split Lines	✔	—	—
Top Level Entities as Split Parts [d]	—	✔	—
Order by Start Border [d]	—	✔	—
Ungroup Text [d]	—	✔	—
Endless Job Repetitions	✔	✔	✔
Center resulting splits	—	—	—
Axis	✔	✔	✔
Split Order	—	—	—
Split Size	✔	—	—

Table 43: 1D MOTF Splitting Parameters

[a]: Not editable, only displaying the value.
[b]: This is not available if *Top Level Entities as Split Parts* is checked.
[c]: Editable, if *Endless Job Repetitions* is checked.

[d]: Cannot be used in combination with *Group width.*

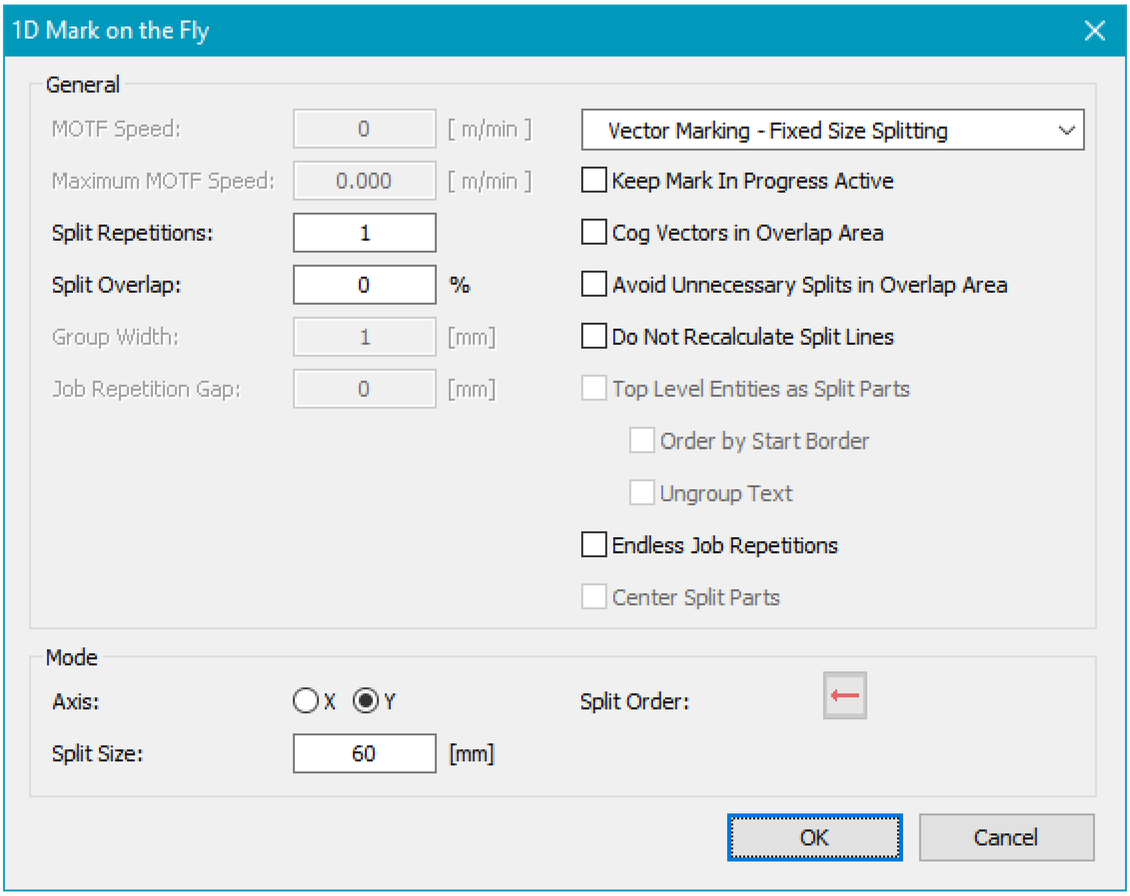


Figure 319: 1D Mark on the Fly Dialog

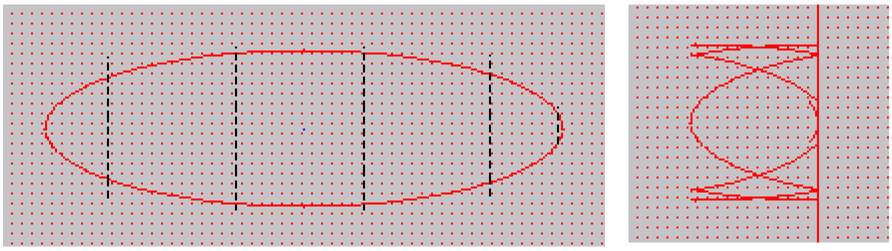


Figure 320: 1D Mark On Fly Split before and after activation of Mark → Trigger

14.2.5 Ring Splitting

The *Ring Splitting* mode is for splitting the current job on a ring.

NOTE

If an option is grayed out in the splitting settings dialog, it is not possible to use it with the current set splitting option. The availability can be looked up in the following table.

Checked but grayed out parameters are not used.

Ring Splitting Parameters	**Vector Marking**			
	Fixed Size Splitting	**Entity Based Splitting**	**Character Based Splitting**	**Bitmap Marking**
Axis Speed	✔	✔	✔	✔

Ring Splitting Parameters	Vector Marking			
	Fixed Size Splitting	**Entity Based Splitting**	**Character Based Splitting**	**Bitmap Marking**
Split Repetitions	✔	✔	✔	—
Job Repetitions	✔	✔	✔	✔
Split Overlap	✔	—	—	—
Group Width [a]	—	✔	—	—
Ring Center Position	✔	✔	✔	✔
Workpiece Movement	✔	✔	✔	✔
Move Back to Start Position	✔	✔	✔	✔
Keep Mark In Progress Active	✔	✔	✔	✔
Cog Vectors in Overlap Area	✔	✔	✔	—
Avoid Unnecessary Splits in Overlap Area	✔	—	—	—
Do Not Recalculate Split Lines	✔	—	—	—
Center Whole Job on Current Axis Position	✔	✔	✔	✔
Top Level Entities as Split Parts [b]	—	✔	—	—
Order by Start Border [b]	—	✔	—	—
Ungroup Text [b]	—	✔	—	—
Axis	✔	✔	✔	✔
Split Order	✔	✔	✔	—
Diameter	✔	✔	—	✔
Height	✔	✔	✔	✔
Margin	✔	✔	✔	✔
Split Angle	✔	—	—	✔
Repetitions Over 360°	✔	✔	—	✔
Z Tilt Compensation	✔	✔	✔	✔
Keep Bitmap Size	—	—	—	✔
Move Forward to Start Position	✔	✔	—	✔
Tolerate Larger Split Jobs Than 360°	✔	✔	✔	✔

Table 44: Ring Splitting Parameters

[a]: This is not available when *Top Level Entities as Split Parts* is checked.
[b]: Cannot be used in combination with *Group Width.*

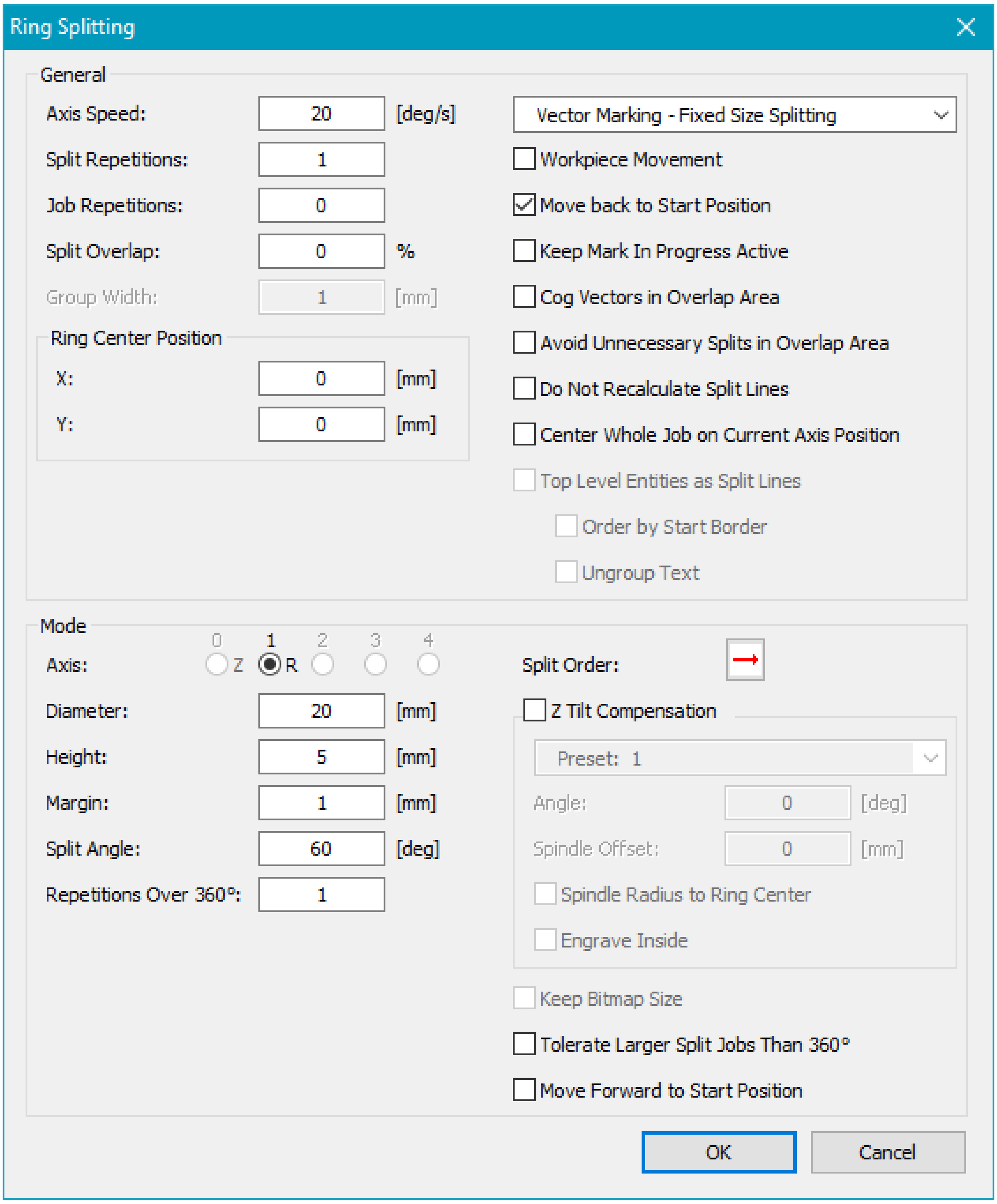


Figure 321: Ring Splitting Dialog

14.3 Splitting Options

SAMLight supports four different splitting modes. The following will give an overview about the differences. Not all Splitting Options are available for each Splitting Mode.

Vector Marking - Fixed Size Splitting:

Regardless of the positions of the entities, the split size is always the same. Some entities may be cut (this can be adjusted manually). The red lines represent the cut into different split parts.

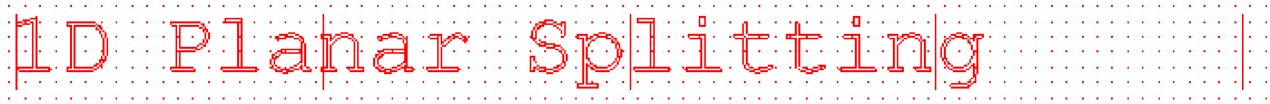


Figure 322: Fixed Size Splitting

Vector Marking - Entity Based Splitting:

This Split algorithm avoids cutting polylines and groups single entities. If the total size of two or more entities does not exceed the set entities width (see above), the entities will be centered and marked together. The dashed lines in the View2D represent the center lines of the grouped entities, the external motion controller will stop at this positions for marking. With *Extras → Splitting → Show split(s)* you can preview the grouped entities. *Entity based splitting* allows use of more than one character string, but e.g. character rotation is not supported with *Entity based splitting*. The black lines represent the middle of the split parts.

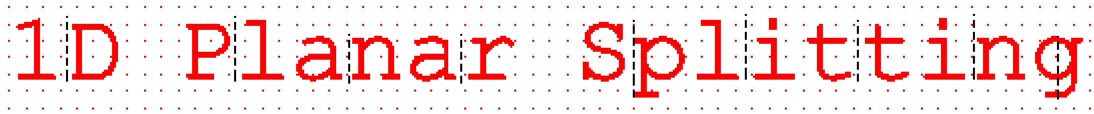


Figure 323: Entity Based Splitting

Vector Marking - Character Based Splitting:

With this option for a job consisting of exactly one text object that consists of one single line, the splitting algorithm will place the splits between the single characters of that text object to avoid that geometry is cut. If you want to split several text objects (maybe even of different font sizes), use entity based splitting with the value of the widest character as *Group width*. The red lines represent the cut into different split parts.

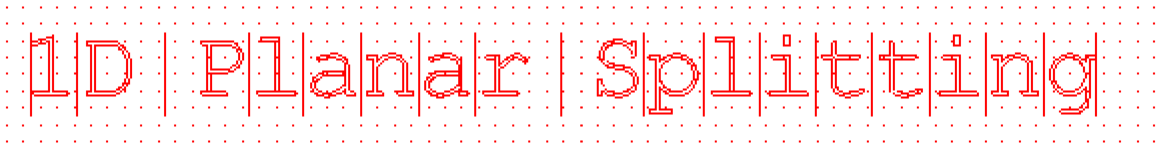


Figure 324: Character Based Splitting

Bitmap Marking:

This option allows for marking a bitmap without the need to split the bitmap into parts manually. The motor moves between every line of the bitmap. It is possible to mark a bitmap on the fly. This feature is only available for USC-2, USC-3 and RTC5. The bitmap dimension may exceed the working area in the direction the belt is moving, but not in the other direction. No split lines are shown here since the bitmap will be marked line by line.

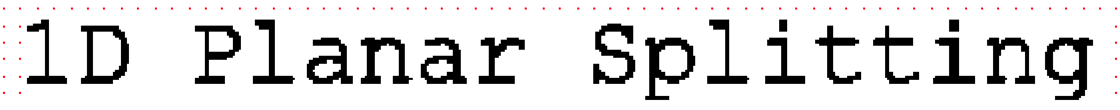


Figure 325: Bitmap Marking

14.4 Visualization of Special Options

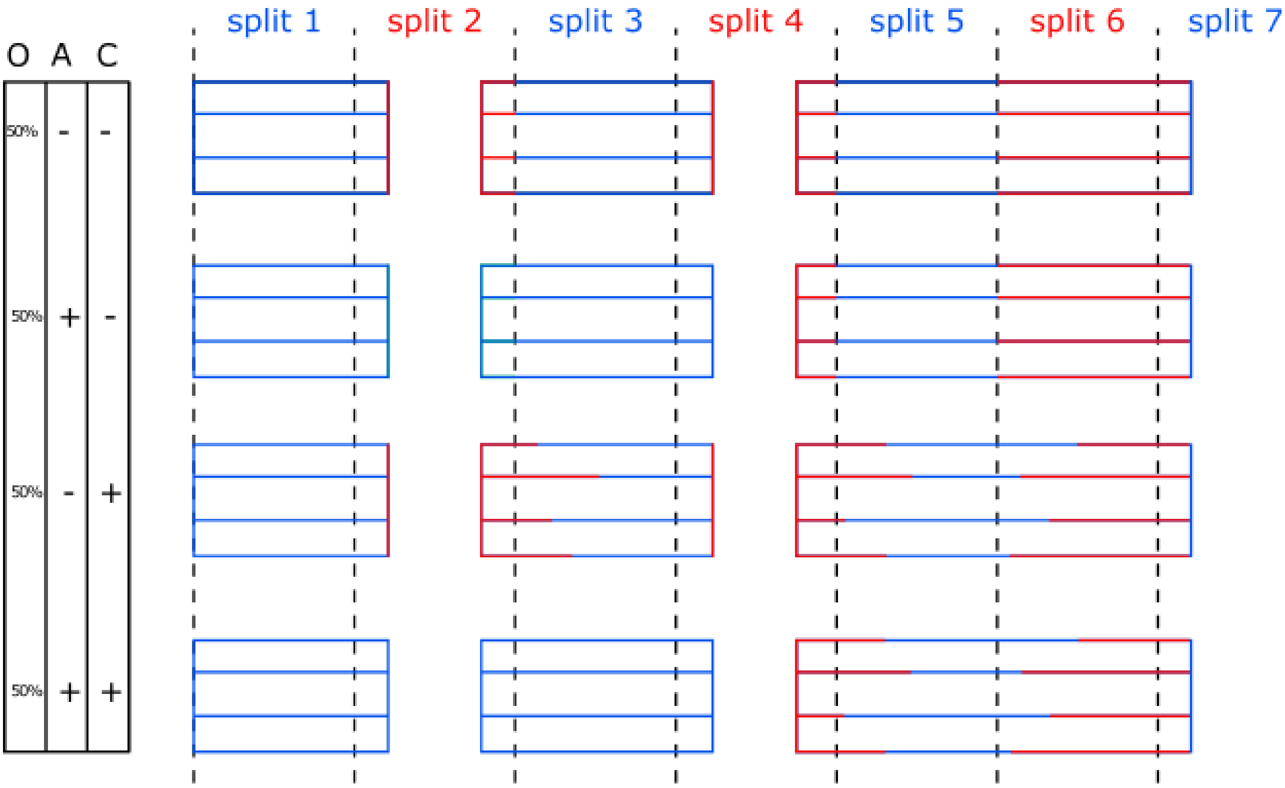


Figure 326: Visualization of Special Options, O Split Overlap, A Avoid unnecessary splits, C Cog Vectors in Overlap Area

For *Vector Marking - Fixed Split Size* it is possible to define a *Split overlap* in % (all Split modes). With this option, the size of the split parts can be increased. The lines will then be grouped optimally and lines that will end within the extra margin will be marked with the part they began in. Lines that would lie within that margin, but would begin in a new split are only grouped in that extra margin when *Avoid unnecessary splits* is selected, additionally.

The example above shows the result of different combinations of *Split overlap (O)*, *Avoid unnecessary splits (A)* and the option *Cog Vectors in Overlap Area* (C) (available for all split modes and options).

14.5 How to Set Split Lines Manually

Split lines can be adjusted manually after SAMLight has calculated the split parts. This option is only available for Fixed Size Splitting.

Select Do Not Recalculate Split Lines:

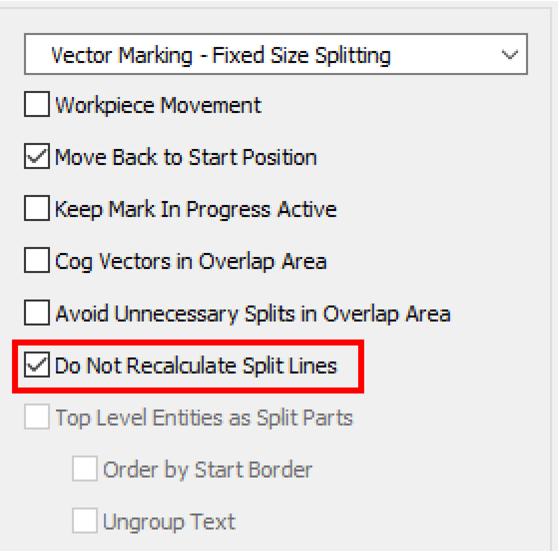


Figure 327: Do Not Recalculate Split Lines

After moving to view level 2 in the toolbar , the split lines are clickable and can be adjusted manually in the direction of the split. This can be useful, when the split lines would cut entities otherwise.

Figure 328: Manually Adjusted Split Lines

15 Option MOTF

This feature "Marking On The Fly" (MOTF) is for marking moving targets on a product line. General information on a MOTF setup is given here.
In sub-chapters, further detail is given on how to work with encoder signals or without them in simulation mode, on the card-specific hardware setup, on how to calibrate a MOTF system and on the possibility of doing endless MOTF or rotational MOTF. Finally, some examples can be found at the end.

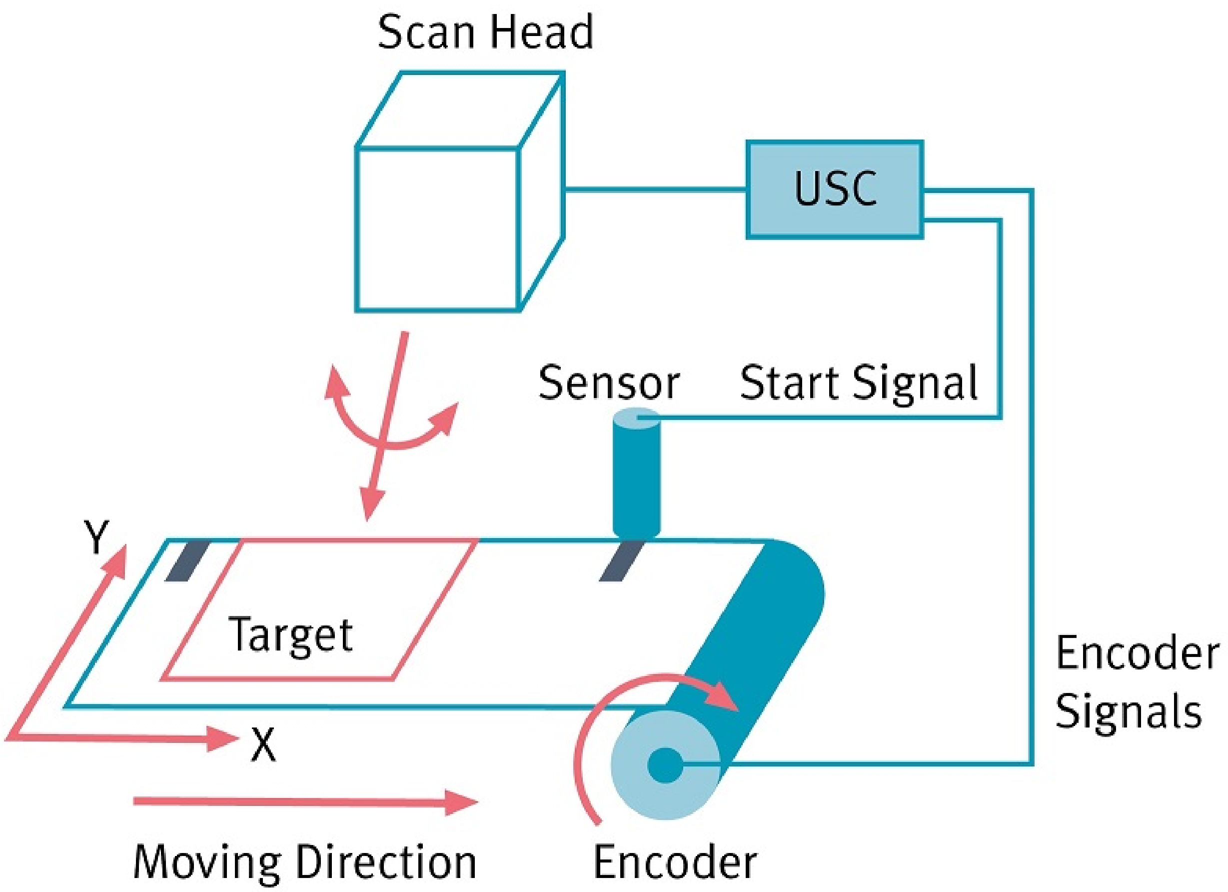


Figure 329: A typical Marking On The Fly (MOTF) setup

The target piece which has to be marked is placed on a moving conveyor belt that surpasses the scan head at a specific position. The scanner has to know when to start marking and how fast the target is moving. Latter is done by the encoder: The movement of the target is converted by the encoder. For a specific distance of the target (in the above example along the X-axis), the encoder gives a specific amount of counts. The information of this conversion is given by the multiplier. Movement is possible in different directions. The distance information from the encoder is sent to the scanner card which appropriately corrects the marking vector such that it fits to the moving target. The starting signal is sent by the trigger sensor: The sensor converts a specific optical label on the conveyor belt into an electrical trigger signal.

The MOTF_CH0 (on the 37-pin connector) and MOTF_CH1 (on the 40-pin connector) signals are filtered by a digital filter unit. For information on the cut-off frequency, see the corresponding hardware manual. After the decoding, a counter counts the incoming count pulses and is incremented or decremented according to signal (see encoder signals for further details). In some application the belt movement direction and speed remains constant. In this case the counts can be generated by an internal simulation generator eliminating the need for an encoder.

In order to calibrate the counter according to the scanner field units [typically in bits, mm or inch], the counter value is multiplied by a user definable signed constant. The resulting MOTF compensation is added to the marking information to form the final signals for scan head control.

15.1 Encoder Signals

The flow chart in figure 330 shows the general MOTF hardware setup with encoder signals.

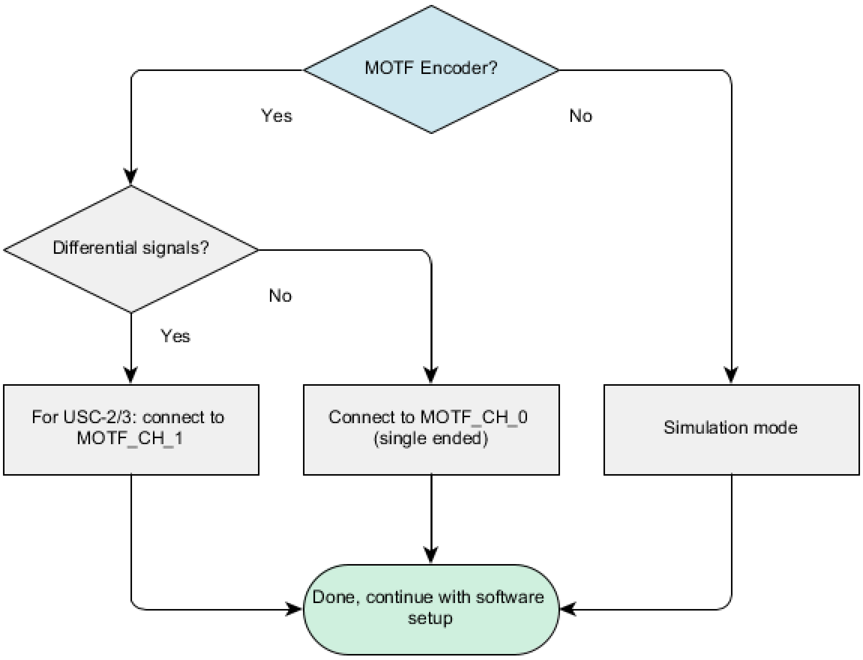


Figure 330: Flow Chart MOTF hardware setup

Verify the GND connection between the USC/RTC card and the encoder to adjust the level. To avoid noisy signals use short cables and check the power supply of the encoder.

The USC cards are designed to handle two 90° shifted encoder signals (track A and track B) delivered by standard commercial encoders. The USC decoder interprets each transition, whether the count impulse is on track A or track B of the respective channel. The interpreted belt direction depends on the phase shift between both tracks (see figure 331).

Phase shift B after A:

If the phase shift between track A and track B is + 90° (refer to figure 331, upper part), the belt is interpreted as moving in the default direction (let's call it forward). In this case, the encoder counts positive which is necessary for advanced MOTF features (like ScMotfOffset). Each falling or rising edge of track A or track B will increment the encoder counter by + 1. That means each track period T will lead to 4 encoder counts in total.

Phase shift B before A:

If the phase shift between track A and track B is - 90° (refer to figure 331, lower part), the belt is interpreted as moving in the non-default direction (let's call it backward). In this case, the encoder counts negative. Each falling or rising edge of track A or track B will increment the encoder counter by - 1. That means each track period T will lead to - 4 encoder counts in total. Please note that for the encoder counting negative, the ScMotfOffset entity will not work properly.

To change the counting direction, the signals of track A and track B can be swapped either by changing the wiring of the tracks or by using Swap A/B in the software. The direction of MOTF compensation may need to be adjusted as well after changing the direction of the encoder counter. Please refer to the corresponding hardware specific sub-chapters for further information.

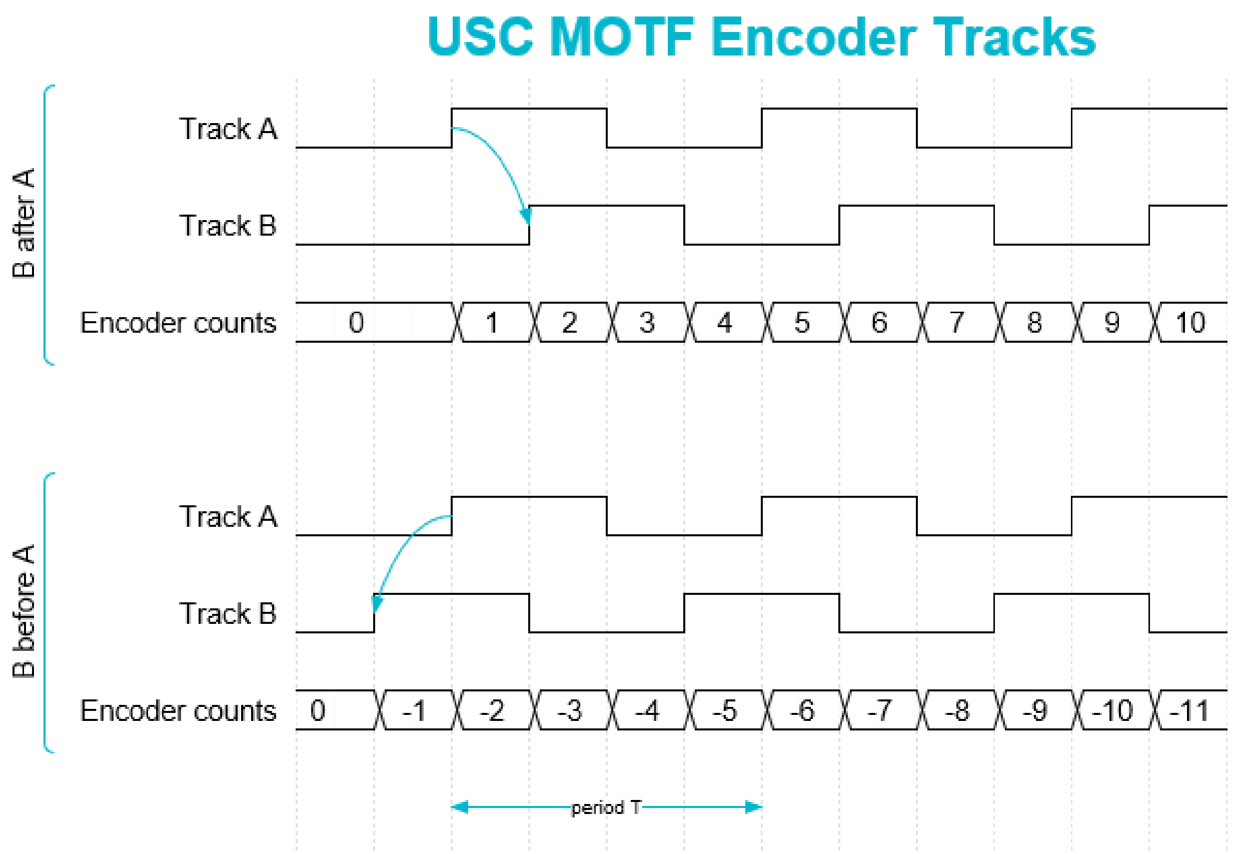


Figure 331: Encoder signals (single ended). Note that each period T consists of 4 encoder counts.

Using encoder signals, it is necessary to specify the conversion between the counts of the encoder and the corresponding distance on the work piece. This conversion is given by the MOTF multiplier in combination with the optic settings.

NOTE

The amplitude of the encoder signals should be in the range of 2.5 V to 5.0 V. The signals should be as low-noisy as possible, since noise could be interpreted as additional pulses. Possibly, you can use a differential encoder and MOTF_CH1 (40-pin connector).

In case of problems, verify the switching threshold levels (cut-off frequency see USC-1, USC-2, USC-3) of the encoder signals with an oscilloscope and/or use differential signals.

15.2 MOTF Multiplier

The MOTF multiplier is the central factor to translate encoder counts into mm. To calibrate this multiplier very well is crucial for correct vector compensation and encoder based distance measurements which are used by ScMotfOffset objects for example. The calibration routine is card specific and described in the following chapters. However, some general remarks and notes are given here.

There are two ways to determine the correct value for the MOTF multiplier:

1. By theory:
   - In the manual of the encoder, the increments per rotation is specified, in the following defined as "I" in the unit counts/360°.
   - The circumference of the wheel that is attached to the encoder is known as "U". The unit is mm/360°. The circumference is calculated over the radius of the encoder with the unit mm.

- Then, the multiplier F is calculated as the ratio of (4 times I) and U. Please find an example for I = 2000 counts/360° and U = 2π * 50 mm/360° below:

$$F = \frac{4 \cdot I}{U} = \frac{4 \cdot 2000 \frac{counts}{360°}}{2\pi \cdot 50 \frac{mm}{360°}} = 25.46 \frac{counts}{mm}$$

- Card specific calculations can be found in the following chapters (Hardware setup).

WARNING

A possible slack between the wheel and the conveyor is not considered. However the calculation gives a good approximation.

The factor I must be multiplied by 4 because each rising and falling edge of both tracks counts. Please have a look at figure 331 for further details on the MOTF signals.

2. By experiment:
   - See USC-1 Specific Calibration
   - See USC-2/3 Specific Calibration

15.3 Simulation Mode

If the belt is moving with a constant speed, the simulation mode can be used instead of connecting an encoder. In simulation mode, the constant speed can be defined in meters per minute (m/min) or in millimeters per seconds (mm/s). All MOTF compensations and ScMotfOffset objects are based on this defined speed. The simulation mode works with an internal frequency of 100 kHz.

NOTE

ScMotfOffset control objects do not work in MOTF simulation mode for USC-1 cards (and RTC3/4).

15.4 Hardware setup

In this chapter, information on how to set up a Marking On The Fly system is given for the following cards:

- USC-1
- USC-2/3
- RTC

For each card, a table gives an overview on the hardware connections available for marking on the fly. Information on how to calibrate a MOTF system are given in chapter Calibration.

15.4.1 Card Specific: USC-1

For the USC-1 card, there is one channel available: the single ended MOTF_CH_0 channel of the 37 pin connector:

USC-1
single ended tracks: Channel 0
37-pin connector: MOTF_CH_0_A
37-pin connector: MOTF_CH_0_B

Table 45: Required MOTF connection for USC-1

Here, the individual options for a USC-1 card are described. To get to the dialog window below (figure 332), select *Menu bar → Settings → System → Optic → Advanced.*

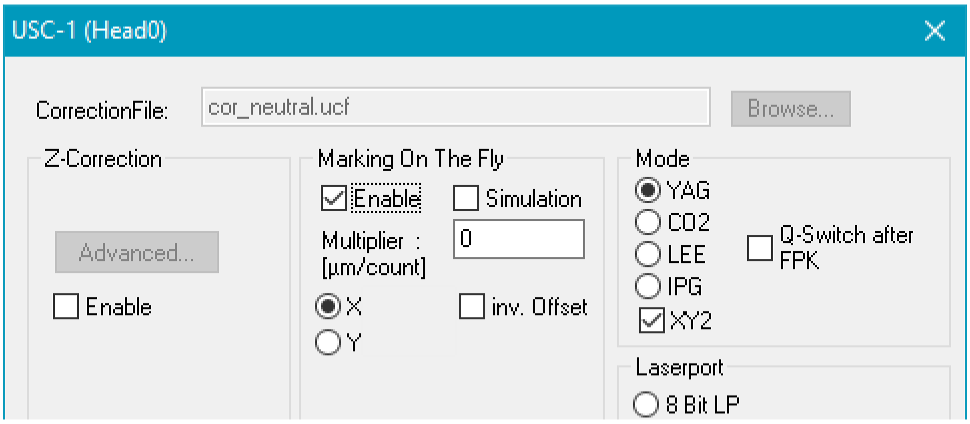


Figure 332: MOTF Settings for USC-1 Cards

Marking On The Fly:

Enable: Activate this checkbox to enable Marking On The Fly.

Simulation: Activate this checkbox to enable Simulation Mode.

Multiplier [µm/count]: Enter the value of the multiplier here in µm/count. If you enter a negative value, the MOTF compensation is switching the direction.
Example of a USC-1 MOTF-Multiplier calculation. In this calculation the radius of the encoder wheel is 50 mm and the encoder is generating 2000 pulses per track (A and B) and full rotation:

$$Multiplier\left[\frac{\mu m}{count}\right] = \frac{circumference\ [\mu m]}{4 \cdot increment} = \frac{2\pi \cdot 50000 \frac{\mu m}{360°}}{4 \cdot 2000 \frac{counts}{360°}} = 39.27 \frac{\mu m}{count}$$

X/Y: Choose the direction in which the target is moving during Marking On The Fly (for neutral optic settings).

inv. Offset: ScMotfOffset control objects can only be positive. It is possible that the MOTF compensation is set up correctly but the MOTF counter value is decreasing (counting negative). In this case, you have to enable this checkbox to be able to use ScMotfOffset control objects.

NOTE

ScMotfOffset control objects do not work in MOTF simulation mode for USC-1 cards.

Only the following configurations allow correct compensation and advanced MOTF features (like ScMotfOffset):

Setup			USC-1 MOTF settings	
MOTF direction	**Phase shift track A - B**	**Orientation of MOTF coordinate**	**Sign of MOTF multiplier**	**Inv. Offset checkbox**
→ or ↑	B after A	normal	positive	unchecked
→ or ↑	B after A	inverted	negative	checked
→ or ↑	B before A	normal	negative	unchecked
→ or ↑	B before A	inverted	positive	checked
← or ↓	B after A	normal	negative	checked
← or ↓	B after A	inverted	positive	unchecked
← or ↓	B before A	normal	positive	checked
← or ↓	B before A	inverted	negative	unchecked

Table 46: USC-1 MOTF settings for different setups

Phase shift track A - B (refer to chapter Encoder Signals):

- B after A: track B is 90° (pi/2) phase-shifted to track A.
- B before A: track B is -90° (-pi/2) phase-shifted to track A.

Orientation of MOTF coordinate:

- normal: the MOTF axis is not inverted in SAMLight → Settings → System → Optic
- inverted: the MOTF axis is inverted in SAMLight → Settings → System → Optic

The orientation of the coordinate system of the UCF correction file has no influence on the MOTF settings.

It is highly recommended to set 'Settings → System → Optic → Rotation' to '0':

- The rotation rotates the entity, but not the MOTF direction.

If a rotation is required, implement the rotation directly in the UCF correction file.

NOTE

If the marking is started with an external trigger (sensor), it is recommended to start the marking in trigger mode (Mark →Trigger) to avoid jitter of the position.

15.4.2 Card Specific: USC-2/3

For the USC-2/3 card, two MOTF channels are available.

- The MOTF signals of Channel 0 (37-pin connector) are single-ended: MOTF_CH0_A and MOTF_CH0_B.
- The MOTF signals of Channel 1 (extension 40-pin connector) are differential: MOTF_CH1_A+, MOTF_CH1_A-, MOTF_CH1_B+ and MOTF_CH1_B-. Do not use this channel single ended!

USC-2/3
single ended tracks: Channel 0
37-pin connector: MOTF_CH_0_A
37-pin connector: MOTF_CH_0_B
differential tracks: Channel 1
40-pin connector: MOTF_CH_1_A+
40-pin connector: MOTF_CH_1_A-
40-pin connector: MOTF_CH_1_B+
40-pin connector: MOTF_CH_1_B-

Table 47: Possible MOTF connections for USC-2/3

Here, the individual options for a USC-2/3 card are described. To get to the dialog window below (figure 333), select *Menu bar → Settings → System → Optic → Advanced.* Enable Marking On The Fly and then go to *Settings*.

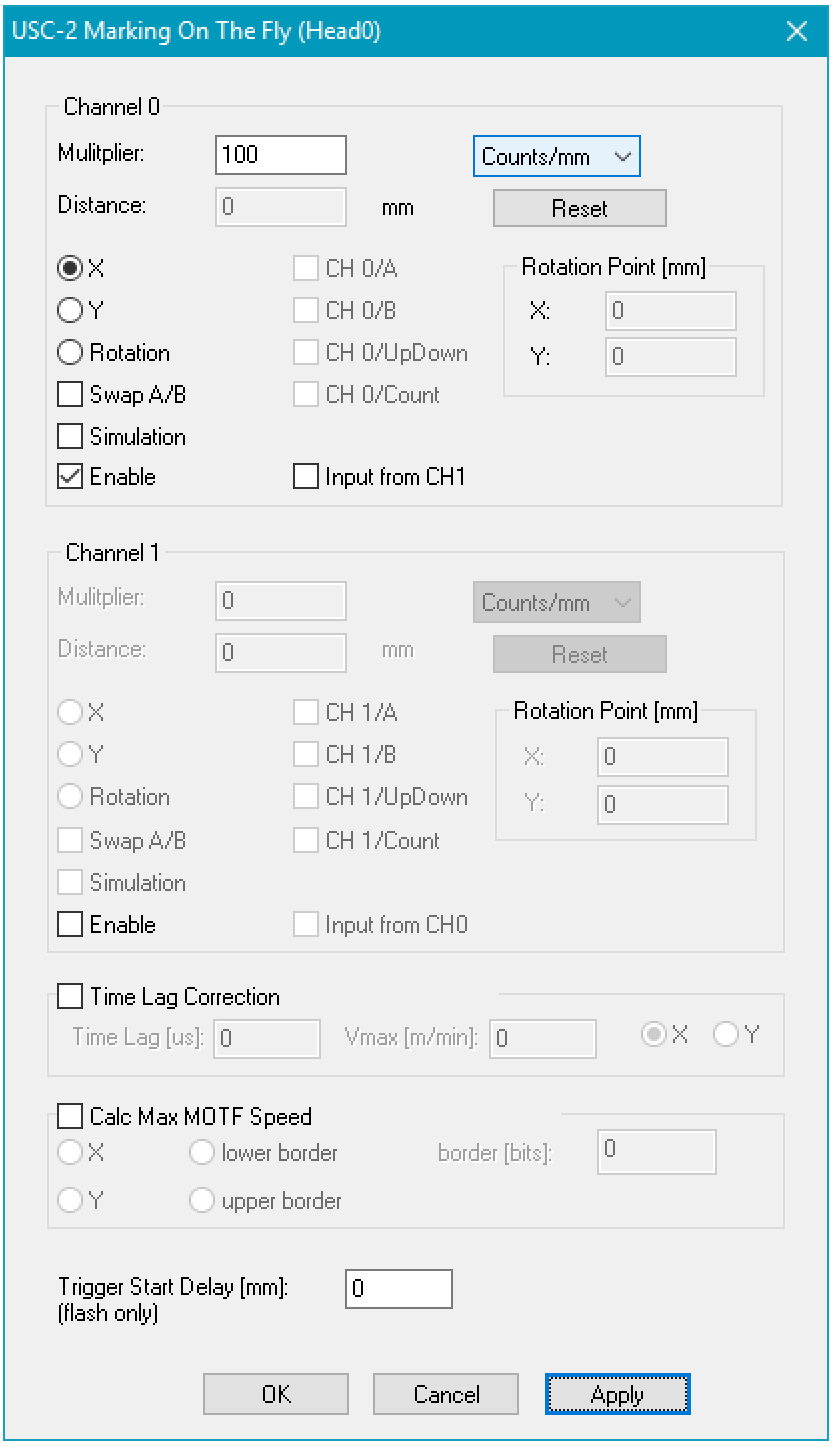


Figure 333: MOTF Settings for USC-2/3 Cards

Channel 0/1: There are two separate channels for MOTF on USC-2/3, Channel 0 and Channel 1.

> **Enable:** Activate this checkbox to enable the desired channel for Marking On The Fly. It is possible to use both (see example rotated scan head).

Multiplier: Unit in [Counts/mm]. When using *Simulation* mode, choose [mm/s] or [m/min] as unit. Using a negative value inverts the direction of the MOTF compensation.
Example of a USC-2/3 MOTF-Multiplier calculation. In this calculation the radius of the encoder wheel is 50 mm and the encoder is generating 2000 pulses per track (A and B) and full rotation:

$$Multiplier\left[\frac{counts}{mm}\right] = \frac{4 \cdot increment}{circumference\ [mm]} = \frac{4 \cdot 2000 \cdot \frac{counts}{360°}}{2\pi \cdot 50 \cdot \frac{mm}{360°}} = 25.46\frac{counts}{mm}$$

Distance: If an encoder is used or if simulation is enabled, the distance of the Marking On The Fly movement is shown here. Normally, this number starts to increment as soon as the encoder or simulation mode is activated and when the belt is moving.

Reset: This button resets the distance counter to 0.

X/Y: Choose the axis in which the scanner is moving during Marking On The Fly (for neutral optic settings).

Rotation: A special feature which is only available for a USC-2/3 card is the rotational Marking On The Fly. Therefore activate the radio button *Rotation*. Then, choose the appropriate units - either Counts/deg or Deg/sec (Deg = Degrees). Finally choose the center of the rotation in *Rotation Point*. You might want to first center the job in the View2D before setting the center X and Y values.

Swap A/B: If activated, the direction of encoder counting will be inverted. The effect of this checkbox is identical to changing the hardware wiring between track A and track B.

Simulation: Activate this checkbox to enable the Simulation Mode.

Input from CH1/0: When activated, the encoder signals of the other MOTF channel are used for MOTF correction. The state of Swap A/B is also taken form the source channel. See Rotated scan head for an example of application.

Time Lag Correction: To compensate the delay of the scan head, a **'Time Lag [µs]'** can be defined. Depending on the current MOTF speed this result in an offset. **'Vmax [m/min] / Time Lag [µs]'** is the maximum of this offset. The **'X'** and **'Y'** radio buttons define the direction of the Time Lag Correction. This feature only works when the encoder pulses result in a positive increase of the distance value. After any changes of the Time Lag Correction, the 'Settings → System → Optic → Advanced → Store' button must be clicked.

Calc Max MOTF Speed: In standalone (flash) mode, it is possible to check the maximal possible MOTF speed of the current job. Steps to use this feature:

- Enable Calc Max MOTF Speed. Only the checkbox must be enabled, the other settings are taken from the MOTF settings above. If both MOTF channels are in use, the CH1 settings are taken.
- Click Settings → System → Optic → Advanced → Store.
- Stop the MOTF belt movement such that there are no more encoder pulses. If the calculation is done while the belt is moving, the result will be less accurate.
- Mark a job in flash mode. Please note that the calculation is only working for jobs without any control objects.
- Start the server in visible mode (<SCAPS>\system\sc_usc_server /v). Select the USC-2/3 card and click on the InfoView button and scroll down to 'MaxMOTFSpeed [m/min]'.
- The 'MaxMOTFSpeed [m/min]' value should be a close approximation of the maximum possible MOTF speed for the current job.

Trigger Start Delay: When a value for the delay is entered, a ScMotfOffset and a ScWaitForTrigger object will be added to the top of the job list. These control objects will delay the job execution until the distance[mm] is reached after the external trigger signal. There must be only one trigger pulse and this pulse must be shorter than the "Distance[mm] / MOTF-Speed[mm/s]". This feature is only implemented for standalone mode (flash mode).

Only the following configurations allow correct compensation and advanced MOTF features (like ScMotfOffset or endless loop applications):

Setup			USC-2/3 MOTF settings	
MOTF direction	Phase shift track A - B	Orientation of MOTF coordinate	Sign of MOTF multiplier	Swap A/B checkbox
→ or ↑	B after A	normal	positive	unchecked
→ or ↑	B after A	inverted	negative	unchecked
→ or ↑	B before A	normal	negative	checked
→ or ↑	B before A	inverted	positive	checked
← or ↓	B after A	normal	negative	unchecked
← or ↓	B after A	inverted	positive	unchecked
← or ↓	B before A	normal	positive	checked
← or ↓	B before A	inverted	negative	checked

Table 48: USC-2/3 MOTF settings for different setups

Phase shift track A - B (refer to chapter Encoder Signals):
- B after A: track B is 90° (pi/2) phase-delayed to track A (of the corresponding channel).
- B before A: track B is -90° (-pi/2) phase-delayed to track A (of the corresponding channel).

Orientation of MOTF coordinate:
- normal: the MOTF axis is not inverted in SAMLight → Settings → System → Optic
- inverted: the MOTF axis is inverted in SAMLight → Settings → System → Optic

The orientation of the coordinate system of the UCF correction file has no influence on the MOTF settings.

It is highly recommended to set 'Settings → System → Optic → Rotation' to '0'.
- The rotation rotates the entity, but not the MOTF direction.

If a rotation is required, implement the rotation directly in the UCF correction file.

Do not forget to press the 'Store' button and save the settings.

In 'sc_usc_server → Info View', the following should be true if MOTF is set up correctly:
- MotfCNT counts positive when 'sign of MOTF multiplier' is positive. MotfCNT counts negative when 'sign of MOTF multiplier' is negative.
- MotfCNTS counts always positive.
- MotfSpeed is always positive.

NOTE *If the marking is started with an external trigger (sensor), it is recommended to start the marking in trigger mode (Mark →Trigger) to avoid jitter of the position.*

15.4.3 Card Specific: RTC cards

Here, the individual options for RTC cards (RTC3, RTC4, RTC5 and RTC6) are described. **Make sure that the hardware license for the RTC card (from SCANLAB) includes Marking on the Fly option.**

For RTC cards, the following MOTF channels are available:

RTC3/4/5/6
differential tracks:
16-pin connector: Encoder X1+
16-pin connector: Encoder X1-
16-pin connector: Encoder X2+

RTC3/4/5/6
16-pin connector: Encoder X2-
16-pin connector: Encoder Y1+
16-pin connector: Encoder Y1-
16-pin connector: Encoder Y2+
16-pin connector: Encoder Y2-

Table 49: Possible MOTF connection for RTC cards

To get to the dialog, choose *Menu bar → Settings → System → Card → Advanced → Driver Settings for [RTC3](), [RTC4](), [RTC5]() or [RTC6]()*. Enable Marking On The Fly and then go to *Settings*.

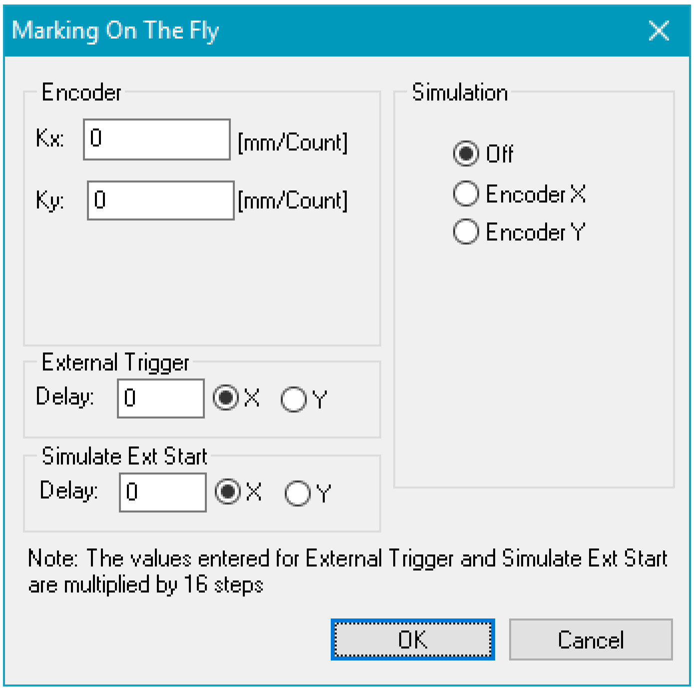


Figure 334: MOTF Settings for RTC Cards

Encoder Kx/Ky: Distance in x/y direction per encoder count. Example of a RTC MOTF-Kx/Ky calculation. In this calculation the radius of the encoder wheel is 50 mm and the encoder is generating 2000 pulses per track (A and B) and full rotation:

$$Kx/Ky\left[\frac{mm}{count}\right] = \frac{circumference\ [mm]}{4 \cdot increment} = \frac{2\pi \cdot 50\,\frac{mm}{360°}}{4 \cdot 2000\,\frac{counts}{360°}} = 0.03927\,\frac{mm}{count}$$

External Trigger: Allows to define an offset between external trigger pulse and the start of the job process. The offset is defined by *Delay* and the unit is encoder steps.

- RTC3/4: *Simulation* must be *Off*. The *Delay* value is internally multiplied by 16 counts.
- RTC5/6: The unit of *Delay* is encoder counts.

Simulate Ext Start: Allows to repeat the job automatically. The gap between the jobs is defined by *Delay* and the unit is encoder steps.

- RTC3/4: The *Delay* value is internally multiplied by 16 counts.
- RTC5/6: The unit of *Delay* is encoder counts.
- The trigger mode (Mark → Trigger) must be used.
- The initial job must be started with an external trigger signal on /START.

- Depending on the direction of the belt, a positive or negative number must be entered.

Simulation: Encoder pulses will be simulated with a constant pulse frequency of 1 MHz.

Off: No simulation
Encoder X: Simulation in X-direction
Encoder Y: Simulation in Y-direction

NOTE

If the marking is started with an external trigger (sensor), it is recommended to start the marking in trigger mode (Mark →Trigger) to avoid jitter of the position.

Simulation can be used with all RTC cards (RTC3/4/5/6). However, MOTF control objects (ScMotfOffset and ScWaitForTrigger) cannot only be used with RTC5/6 cards (not with RTC3/4).

15.5 Calibration

In the following, a general calibration routine is described.
This consists of:

- calibration of the optic system in the static situation, meaning without any movement of the belt
- calibration of the (well-calibrated static system) in the dynamic situation for the USC-1 card
- calibration of the (well-calibrated static system) in the dynamic situation for the USC-2/3 card

The following jobs are useful for calibration: a simple text object like "ABC" to check the orientation, a circle for easy observation of the compensation and a grid to check the accuracy.

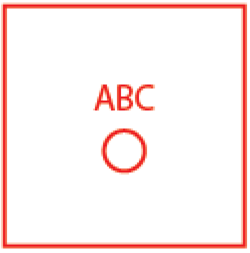


Figure 335: ABC-circle job

ABC-circle.sjf consists of the letters A, B and C in a single laser font and a circle.

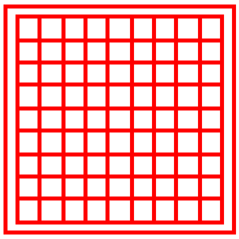


Figure 336: Grid job

Grid.sjf consists of grid vectors (e.g. hatch a square) to check the X and Y aspect ratio. If calibrating the static system, the grid may cover the whole field size.

NOTE *Set Rotation to 0 in Settings - System - Optic (Advanced)! Put a rotation, if necessary, in the correction file because the correction file is applied after MOTF compensation.If you want to use Rotation in the settings, MOTF compensation is necessary in x and y direction, see example rotated scan head.*

15.5.1 Optic Calibration - static setup

Before starting with any Marking On The Fly application, the optic system needs to be well-calibrated in the static situation (no belt movement). A precisely calibrated static system (correction file and optic settings) is absolutely required and really critical for a good marking results in the dynamic system.

It is recommended to start with neutral optic settings:

SAMLight → Settings → System → Optic:

- Disable X invert, disable Y invert and disable XY flip.
- Set X gain to 1.0, Y gain to 1.0 and rotation to 0.0°.

SAMLight → Settings → System → Optic → Advanced → Correction, Settings:

- Set X gain to 1.0, Y gain to 1.0 and rotation to 0.0°.
- If necessary, inversion, flip of axes and rotation can be included in the correction file *.ucf using <SCAPS>\tools\sc_corr_table.

Start (without any movement of your target) to check the calibration in the static situation:

- Make sure your encoder does not send any pulses.
- Mark a job like ABC-circle.sjf: Is the orientation (X and Y axes) of the marking result correct - is the text "ABC" readable?
  - o If not, the correction file must be manipulated. This can be done using <SCAPS>\tools\sc_corr_table (correct the orientation).
- Mark a job like Grid.sjf: Are the X and Y aspect ratios correct? In other words: is 1 mm in the drawing very precisely 1 mm on the target in vertical and horizontal direction throughout the whole working area (not only in the middle of the field)? This is very important for 1D Mark On The Fly split applications, because the start point and the end point have a quite big distance within the scanner field in such an application. This is even more critical with increasing speed of the line and with increasing size of the closed shapes.

- o If not, adapt the field size until the dimension in the direction which you want to use as MOTF axis is correct. Mark again: are the X and Y aspect ratios correct?
    - If not, adapt the gain value of the axis, which is not used as MOTF axis. Mark again: are the X and Y aspect ratios correct?
        - If not, improve the correction file with sc_corr_table. Please find further information on how to work with sc_corr_table online.

NOTE *Set Rotation to 0 in Settings - System - Optic (Advanced)! Put a rotation, if necessary, in the correction file because the correction file is applied after MOTF compensation.If you want to use Rotation in the settings, MOTF compensation is necessary in x and y direction, see example rotated scan head.*

15.5.2 USC-1 Specific Calibration MOTF

Before starting with any Marking On The Fly application, the optic system needs to be well-calibrated in the static situation (no belt movement).
Then, essentially two points need to be checked and adapted:

- MOTF compensation (value of the MOTF multiplier)
- direction of the MOTF compensation (sign of the MOTF multiplier)

1. Check the proper alignment of the scanning field and the scanning direction?

- This can be tested e.g. with marking one line in the moving direction without moving the belt and marking a single point (with a mark loop count of -1) with the slowly moving belt on top of the last marking.
- If the alignment is not good, this can be solved by adjusting the hardware setup or within the correction file.

15.5.3 USC-2/3 Specific Calibration MOTF

Before starting with any Marking On The Fly application, the optic system needs to be well-calibrated in the static situation (no belt movement).

The flow chart in figure 337 shows an overview of the general steps of the MOTF software setup calibration. Please proceed with the steps below.

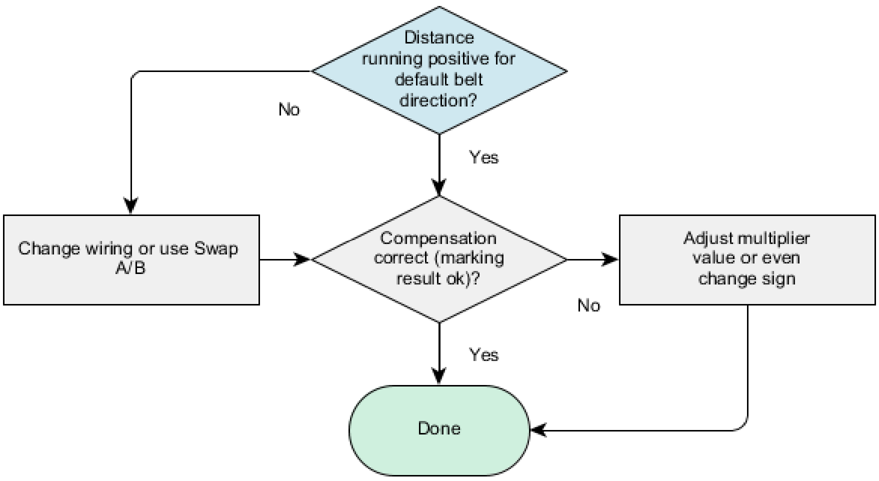


Figure 337: Flow Chart MOTF software setup

1. Check the proper alignment of the scanning field and the scanning direction?

- This can be tested e.g. with marking one line in the moving direction without moving the belt and marking a single point (with a mark loop count of -1) with the slowly moving belt on top of the last marking.
- If the alignment is not good, this can be solved by adjusting the hardware setup or within the correction file.

2. Mark without MOTF settings (without compensation) in SAMLight:

- Start the belt in the default direction with a slow movement. Mark ABC-circle.sjf. It is OK, if the marking result is not good because this marking is a reference to compare with a marking with MOTF compensation (see 338 1)).

3. Activate the MOTF compensation in SAMLight:

- Activate SAMLight → Settings → Optic → Advanced → Marking on the Fly by clicking on the checkbox 'Enable'. Then, please reopen the Advanced dialog.
- Enable Channel 0 for MOTF inputs on the 37 pin connector or enable Channel 1 for the MOTF inputs on the 40 pin extension connector (according to the electrical connection of the encoder).
- Select X-Channel or Y-Channel (or Rotation) according to the movement direction of the belt. Note: due to a flip in the optic settings the correct channel might be not obvious.
- Select the multiplier unit: counts/mm. The units m/min or mm/s are also available (for Simulation Mode with a reference of 100 kHz).
- As a starting value, set the multiplier to the value which can be calculated following the formula here.
- Click on Apply and on the store button in the Optic → Advanced dialog after any MOTF settings changes.

4. Start the belt with a slow movement in the default direction:

- Is the value of 'SAMLight → Settings → Optic → Advanced → Marking on the Fly → Distance' counting positive?
    - o If not, change the wiring of the MOTF tracks (A and B) or use the checkbox "Swap A/B" in Settings → System → Optic → Advanced → MOTF.
  - o Make sure that when driving in the default direction, the distance counter will move in positive direction. Otherwise, the ScMotfOffset control object for example will not work properly.

- Move the MOTF belt for different defined distances. Is the distance value displayed in the software correctly?
  - o If not, adjust the MOTF multiplier value.

5. Mark with MOTF settings (with compensation) in SAMLight:

- Start the belt in the default direction with the same slow movement as in 2. Mark a job like ABC-circle.sjf. Compare the marking result with the result in 2.
  Is the marking result getting better with compensation or getting worse? This can be easily seen when looking at the gap of the circle. There are 3 different cases (see figure 338 2)).
  - o If the result is getting worse (see figure 338 2)a)) - e.g. the start and end of the circle are farther apart, change the wiring of your encoder or change the direction of the compensation by changing the sign of the MOTF multiplier value. See table "USC-2 MOTF settings for different setups" for more details.
  - o If the result is getting better (see figure 338 2)b)) - e.g. the start and end of the circle are coming closer, adjust the MOTF multiplier.
  - o If you run into overcompensation (see figure 338 2)c)) - e.g. the start and end of the circle changed orientation in the direction of the movement, fine tune the MOTF multiplier.

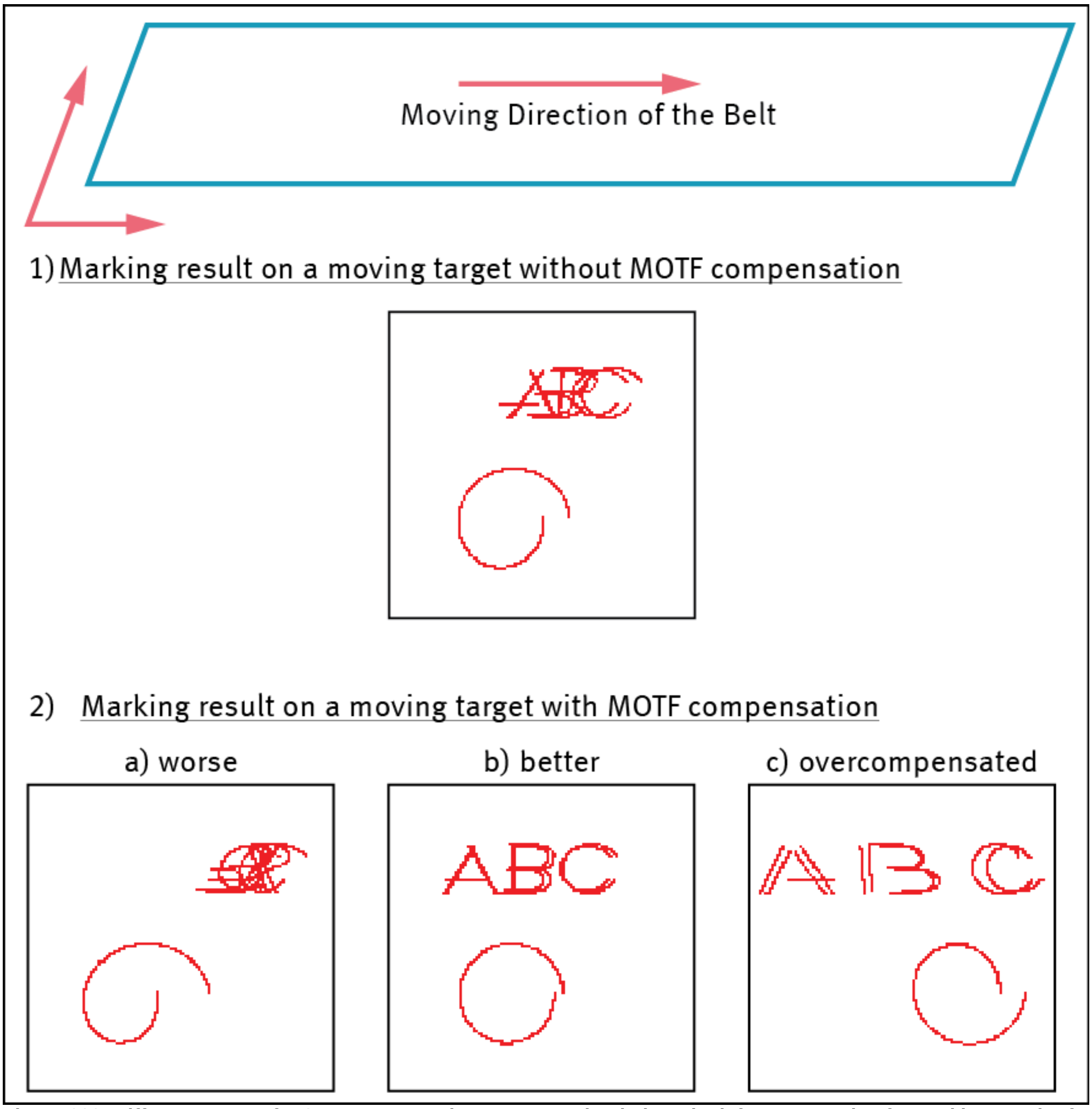


Figure 338: Different cases of MOTF compensation. For a moving belt to the left, an example of a marking result of ABC-circle.sjf without MOTF compensation is shown in a. When the MOTF compensation is activated in SAMLight, the marking result depends on the precise value of the MOTF multiplier. Three different marking results for three different MOTF multiplier values are shown in b.

6. Mark with MOTF settings a job containing 2 entities separated by a ScMotfOffset entity.

- Create a job containing 2 entities, e.g. two vertical lines and center both entities in x and y. Add a ScMotfOffset entity at the beginning and specify a certain distance, e.g. 10mm. Add a ScWaitForTrigger entity after the first line entity, see figure 339 a).
- Start the marking. Are both entities marked?
  - o If you are just seeing the first entity, change the checkbox invert Offset in SAMLight. See table "USC-2 MOTF settings for different setups" for more details. "
- Is the distance between the entities as specified within the ScMotfOffset entity (see figure 339 b) for visualization)?
  - o If not, it could be that the marking time of the first entity was too long for the offset. Try with a larger offset or faster marking speed or smaller entity. Alternatively, the initial external trigger was applied longer than the corresponding offset time. Another possibility is that the 2D calibration of the static setup or the MOTF multiplier is not yet optimal.

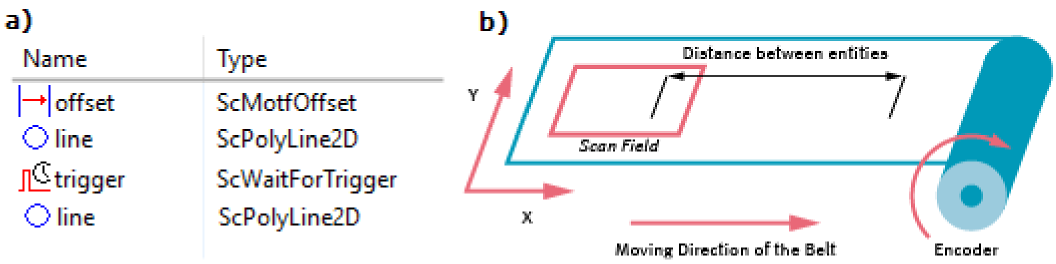


Figure 339: a) Example job containing 2 lines, ScMotfOffset and ScWaitForTrigger. b) Visualization of the marking of a) where the distance between the entities (lines) is defined by the value of the ScMotfOffset entity.

NOTE

It can be helpful to activate the Error Warning Info for Out of field and Overflow MOTF offset.

NOTE

The amplitude of the encoder signals should be in the range of 2.5 V to 5.0 V. The signals should be as low-noisy as possible, since noise could be interpreted as additional pulses. Possibly, you can use a differential encoder and MOTF_CH1 (40-pin connector).

15.5.4 Tips to optimize MOTF performance

Tips for increasing of the maximum possible MOTF speed:

- If using Flash with FCI Calls: Increase MOTF Return Speed (MRS) to get less MOTF dead time (for USC-2/3).
- Job set up: Place the marking object in the opposite of the MOTF direction in the working area to increase the usable MOTF area. Make sure that the gap to the next object is still big enough for the jump.

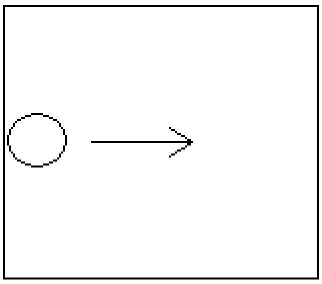


Figure 340: For moving direction of the belt from left to right, place the marking object on the left side of the working area.

- Optic settings: Enable HomeJump and set the position to the start point of the job file to decrease the start jump.

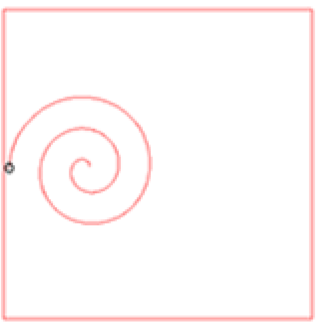


Figure 341: Set the home position to the first point of the spiral, marked with a black circle in the picture.

- Pen settings: Decrease scanner delays and increase scanner speeds, do not forget the HomeJumpStyle (pen 256).
- Use standalone mode (flash mode) to get less MOTF dead time.

15.6 Endless MOTF

SAMLight provides the option to do endless cutting or endless looping, giving the possibility to fold a polyline in order to pre-compensate the MOTF movement.
For these Flash-only-features (only for USC-2/3), the SAMLight Option Flash and Option MOTF are required. Then, the job needs to be prepared in the following way:

1. First, the unfolded polyline is created in SAMLight. The entity must not be more than one single polyline (no jumps allowed). The Y-coordinate of the first and of the last point of this polyline must be identical, the X-coordinate of these two points must be different.
2. Then, this polyline is folded
   a. either while being exported as CNC with the advanced option of endless cut or endless loop (*File → Export*, choose *GCode Files (*.cnc)* as file type and open the *Advanced... dialog*; see figure 343)
   b. or while being saved to the Flash memory. (*Extras → Flash*, activate the checkbox *SJF → CNC → UNF* and open the *settings* dialog on the right; see figure 343).
3. If the polyline was folded by the export function of SAMLight, this CNC file can then be loaded to the USC card (e.g. per FTP server) and convert to UNF file. It can also be imported into SAMLight for the purpose of illustration (see figure 342).

In Endless Cut and in Endless Loop modes, the target speed should be close to a nominal MOTF speed V(MOTF). For MOTF speed variances, a look-up-table for the scanner speed and for the laser power is provided for up to 16 controller cards. Please note that the outline of the polyline can be larger than the field size of your scanner along the MOTF direction before the convolution has been executed. If you want to save the values of the CNC export in your job file, check *File → Job Properties... → MOTF Multiplier, Save in Job / Load from Job*. To save the setting for the look-up-table (see illustration in figure 344) for scanner speed and laser power, you have to select the polyline you want to mark. Then, go to *File → Export*, choose *GCode Files (*.cnc)* as file type and open the *Advanced... dialog.*

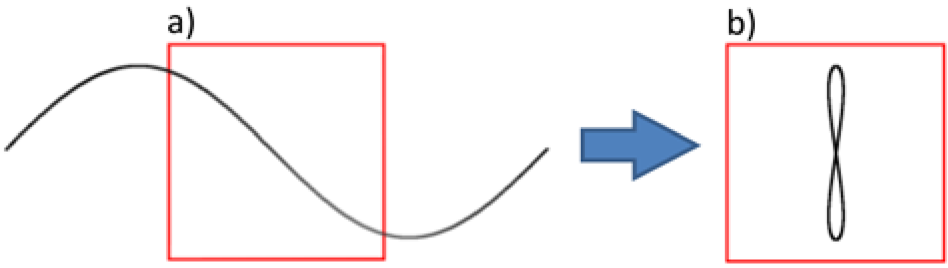


**Figure 342: Illustration of Endless Loop: a) A sine curve with an outline larger than the field size,
b) after CNC export the sine curve is folded to a closed form which fits into the field size and can be marked endlessly.**

NOTE

*In this feature the vector compensation is not based on MOTF encoder values. Instead the vector calculation is pre-calculated by folding the polylines. This is done by generating the *.UNF job for a nominal speed and will be adjusted by internal override parameter which can be defined in lookup tables for speed and laser power. However, the internal override procedure is not done in real time but at the beginning at each job. If the belt speed is not constant this functionality will lead into small distortions of the vectors.*

The following endless MOTF dialog (figure 343) can be reached via *Extras → Flash*, activation of the checkbox *SJF → CNC → UNF* and opening of the *Settings* dialog on the right.

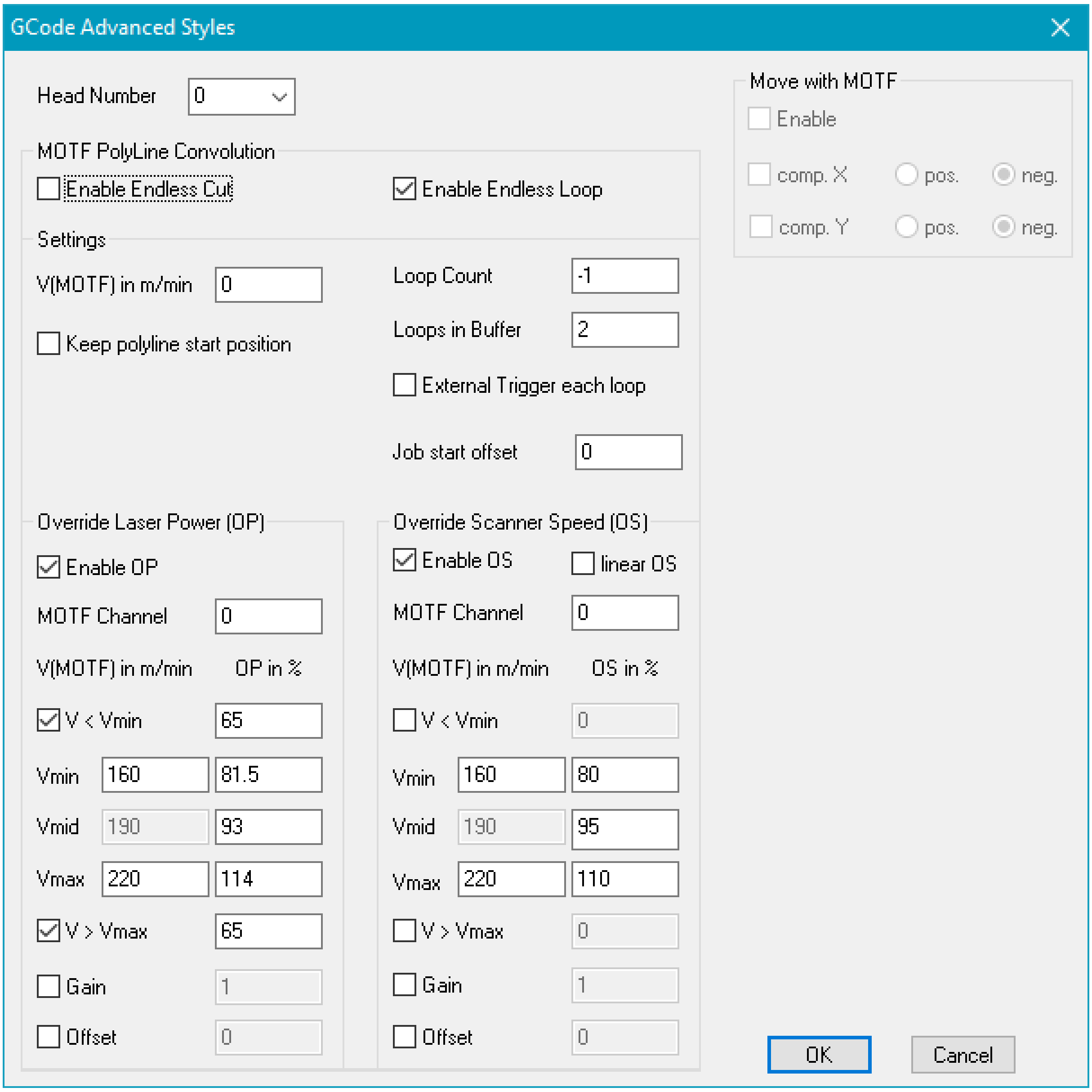


Figure 343: Endless MOTF Dialog

Head Number: Chooses the card number. For single card applications, choose '0'.

Enable Endless Cut: Folds the polyline(s) by the nominal MOTF speed V(MOTF).

> **Requirements of the polyline(s):** The entity must be a single polyline or series of polylines (no jumps between vectors are allowed).
>
> **V(MOTF) in m/min:** Sets the nominal MOTF speed. This speed is used for the folding of the vectors.

Enable Endless Loop: Folds the polyline into a closed form. Laser will be left on all the time during marking.

> **Requirements of the polyline:** The entity must be a single polyline (no jumps between vectors are allowed), the Y-coordinate of the first and the last point of the polyline have to be the same whereas the X-coordinates of these two points must be different.
>
> **V(MOTF) in m/min:** Sets the nominal MOTF speed. This is used for calculating the scanner speed if override speed is disabled.

Keep polylines start position: When disabled, the folded polyline will be centered. When activated, the start position of the polyline will not be changed. In this case, the start point of the polyline needs to be within the valid field.

External Trigger each loop: The USC card waits for external trigger via OPTO_IN_0. This can be used in combination with endless cut.

Loop Count: Define the number of loops the polyline will be marked, type -1 for endless marking.

Loops in Buffer: Buffered jobs are used to avoid gaps in the marking result, but buffered jobs will not be updated by OP / OS values.

Job start offset: A single MOTF offset is added at the beginning of the job. It is not included in the loop and only included if not in trigger mode.

Enable OP, Enable OS: For MOTF speed variances a look-up-table for scanner speed and laser power can be defined. This can be done by defining an lower and upper limit of the MOTF speed. In the range between Vmin and Vmax three nodes are set equidistantly, refer to the picture below. If Override Laser Power (OP) / Override Scanner Speed (OS) is enabled the laser power / scanner speed will be overwritten at the beginning of each job.

linear OS: Uses the nodes of OP for OS as well.

MOTF Channel: Defines the source of the MOTF encoder signal: *0* = MOTF-Channel 0 at the 37 pin connector, *1* = MOTF-Channel 1 at the 40 pin extension connector.

V < Vmin: Defines a constant scanner speed / laser power if MOTF speed is below the range of the look-up-table. If disabled, Vmin is used instead. The internal minimum override value is 20%, so do not use smaller values.

Vmin, Vmid, Vmax: Nodes of the look-up-table. Speed values in [m/min], override value in [%].

V > Vmax: Defines a constant scanner speed / laser power if MOTF speed is above the range of the look-up-table. If disabled, Vmax is used instead.

Gain: All nodes values of the look-up-table will be multiplied by this gain.

Offset: An additionally offset value can be applied to all node values.

The following figure illustrates the look-up-table of speed and power values for endless MOTF.
These setting for the look-up-table for scanner speed and laser power can be saved after selection of the polyline you want to mark via *File → Export*, choose *GCode Files (*.cnc)* as file type and open the *Advanced... dialog.*

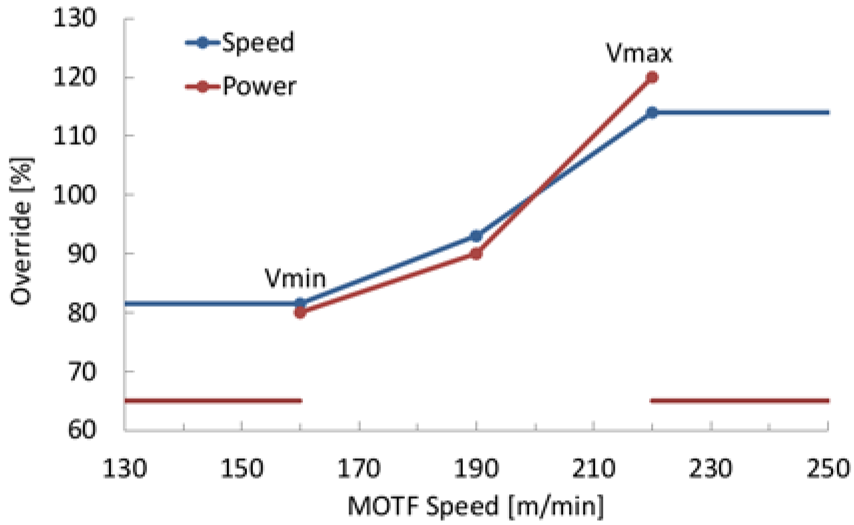


Figure 344: Illustration of Speed- and Power-look-up-table in endless MOTF

15.7 1D MOTF Splitting

The "Marking On The Fly" (MOTF) itself can be seen as a way of enlarging the working area of a scan head. However, the entities to be marked still need to be placed all within the valid field in SAMLight which restricts the widening of the field significantly.

If entities bigger than the working area should be marked on the moving target, 1D MOTF splitting needs to be used. This feature splits the whole job into smaller parts ready to be marked by the scan head within the working area.

15.8 Examples

Here, some examples for MOTF jobs are given.

15.8.1 Trigger based offset

Marking On The Fly can be used to set up an assembly line production with many targets passing by one after another through the scanner field.

Figure 345 shows an example of such an assembly line. Each marking is started by an external trigger signal and the distance between the print mark for the trigger signal and the target is always the same. For **USC-1/-2/-3 and RTC5/6 cards**, the Trigger Control Objects can be used to generate such an offset.

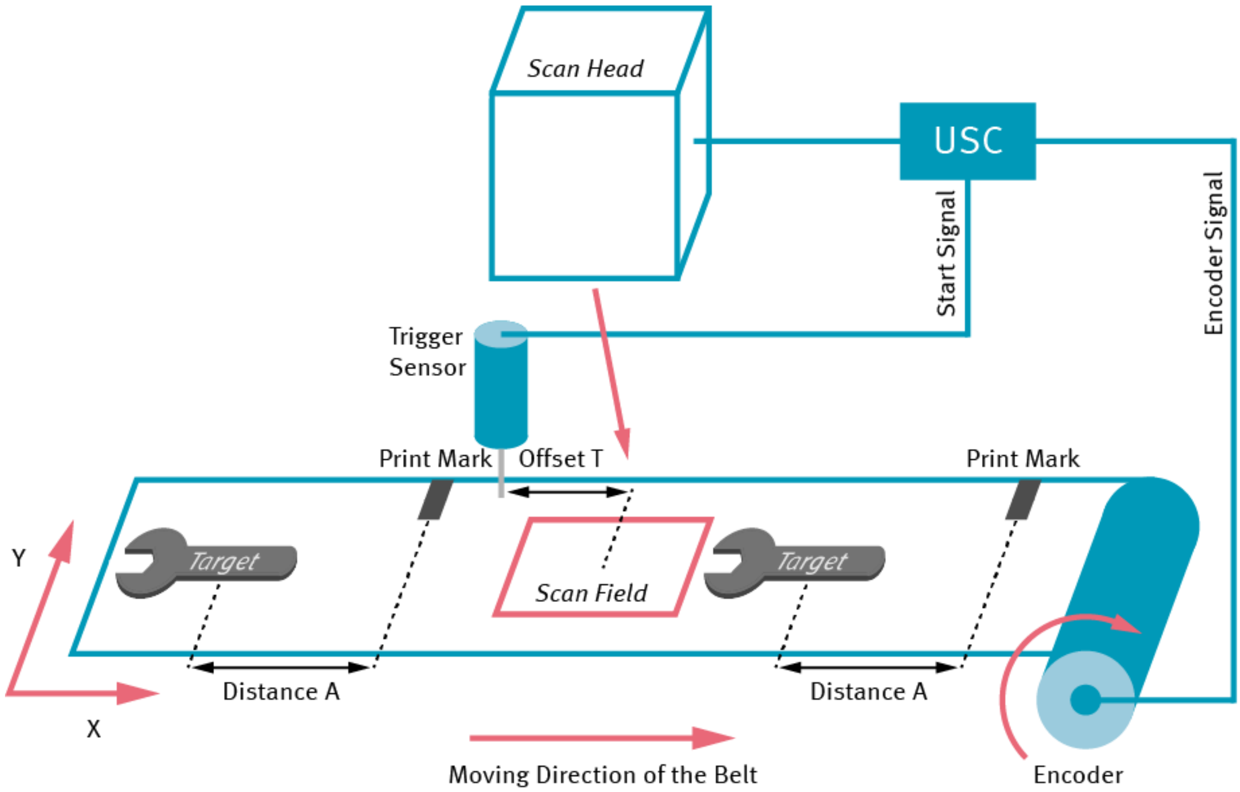


Figure 345: Assembly line with Marking On The Fly: the trigger sensor gives a start signal for each print mark to mark the target. Print mark is located distance A apart from the target.

The job for such a setup can look like in figure 346:

Name	Type
Distance A	ScMotfOffset
Internal Trigger after distance A	ScWaitForTrigger
Target	ScEntities2D

Figure 346: Job for Trigger based offset with Marking On The Fly

Please note that the distance which needs to be specified in the ScMotfOffset entity of the job depends on the positioning of the entity within the scan field (working area in SAMLight) as well as by the positioning of the trigger sensor in respect to the scan field (Offset T in figure 345).

15.8.2 Assembly Line

Marking On The Fly can be used to set up an assembly line production with many targets passing by each after another through the scanner field.

If there is no external trigger signal for each part, but only for the first one and if the distance between the parts is always the same, the assembly line can be realized by the following solution (discriminate between USC and RTC cards).

For USC-1/-2/-3 and RTC5/6 cards, the Trigger Control Objects can be used together with the *Mark Loop Count*. In this case, set up the job as wished for one marking including all the trigger control objects. Then, select those objects in the entity list and group them all together with *Edit → Group*. Finally, set the *Mark Loop Count* of this group to *-1* in the *Entity Info* property sheet.

Name	Type	
Group	ScEntities2D	Mark Loop Count = -1
...		
start distance counting A	ScMotfOffset	
marking Target 1	ScLayer	
wait distance A	ScWaitForTrigger	
start distance counting B	ScMotfOffset	
marking Target 2	ScLayer	
wait distance B	ScWaitForTrigger	
start distance counting C	ScMotfOffset	
marking Target 3	ScLayer	
wait distance C	ScWaitForTrigger	

Figure 347: Example for MOTF Job for an Assembly Line: Target 1, 2 and 3 are repeated with the distances A, B and C in between.

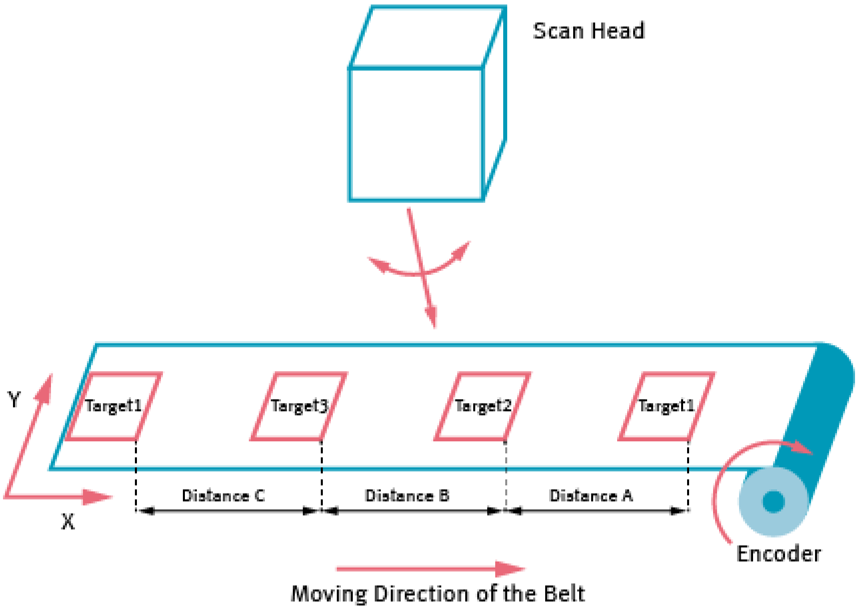


Figure 348: Assembly line with Marking On The Fly

For RTC-3/-4, the trigger control objects are not available. Looping is possible with the Simulate Ext Start feature.

For RTC5/6 cards, the looping can alternatively be done with the Simulate Ext Start feature.

NOTE *You need an encoder for the RTC cards as well as for the USC-1 card. The simulation mode can not be used for this. Only the USC-2 and USC-3 card can handle the simulation mode together with the ScWaitForTrigger and ScMotfOffset objects.*

15.8.3 Rotational MOTF (RMOTF)

Marking on the Fly on a rotary disk is possible for the USC-2/3 cards and serves for marking on the bottom of bottles which are rotated by a disk.

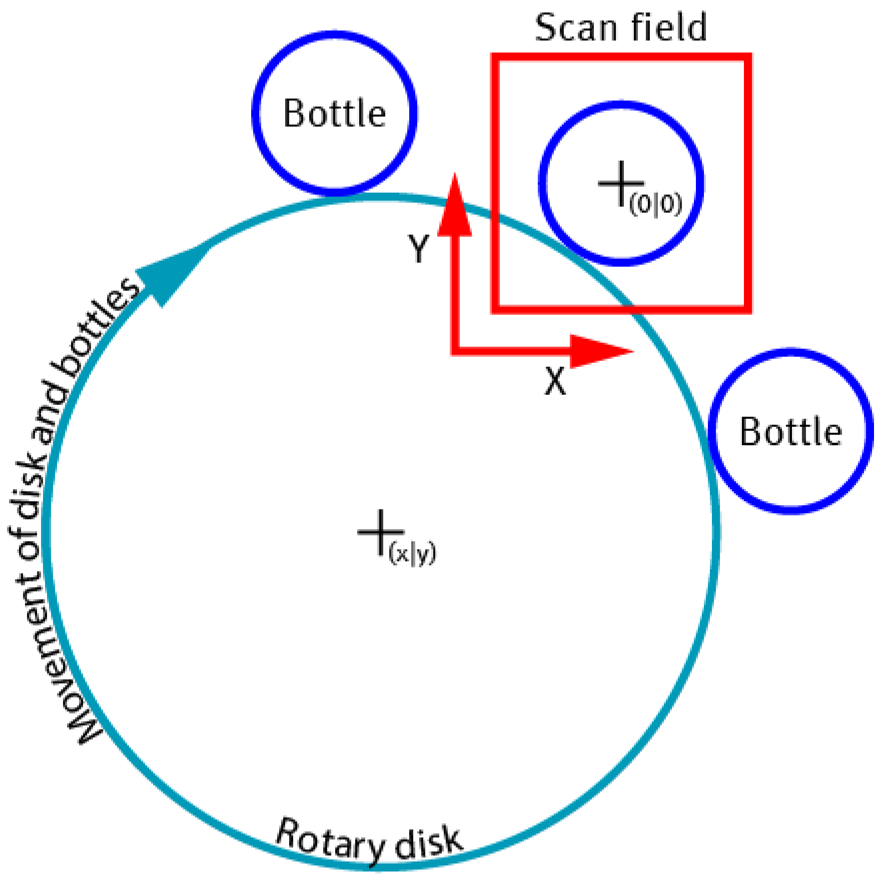


Figure 349: Setup RMOTF

Figure 349 shows a rotary disk (light blue) which is rotating clockwise. Bottles (dark blue) are rotated by this disk and thereby pass by the scan field (red) with the origin of the coordinate system (0|0). The center of the rotary disk is given in respect to the origin within the scan field.

Please follow the following steps to set up such a system.

1. Step one:
   a. Make sure that the proper correction file is applied (in the global settings).
   b. Make sure that the optic is calibrated in the static situation, i.e. 1 mm in the software equals 1 mm at the scanner output. Use the field size value for calibration.
   c. In figure 349, we use a 100 mm scan field, from -50 mm to 50 mm, so that the origin has the coordinates (0|0).
   d. Setup a job that contains a dot with the coordinates of your origin (in our example (0|0), figure 349).
   e. Go to the pen settings and enable the drill mode.
   f. The drill time, i.e. "Duration", depends on the disk speed.

2. Step two:
   a. RMOTF: off → disable Marking On The Fly in the Settings dialog (Settings → System → Optic → Advanced).
   b. Rotary disk: runs with a defined speed (deg / s)
   c. Start marking
   d. The marking result should be a segment of a circle where the distance from the rotary disk center to the origin of the scan field equals the radius.
   e. The length of the segment depends on the drill duration and the disk speed.

3. Step three:

a. RMOTF: on → enable Marking On The Fly in the Settings dialog
b. Rotary disk: stands still
c. Enable MOTF, simulation and choose "Rotation". See figure 349 to determine the correct coordinates of the Rotation Point (the point is (-x|-y) in this example).

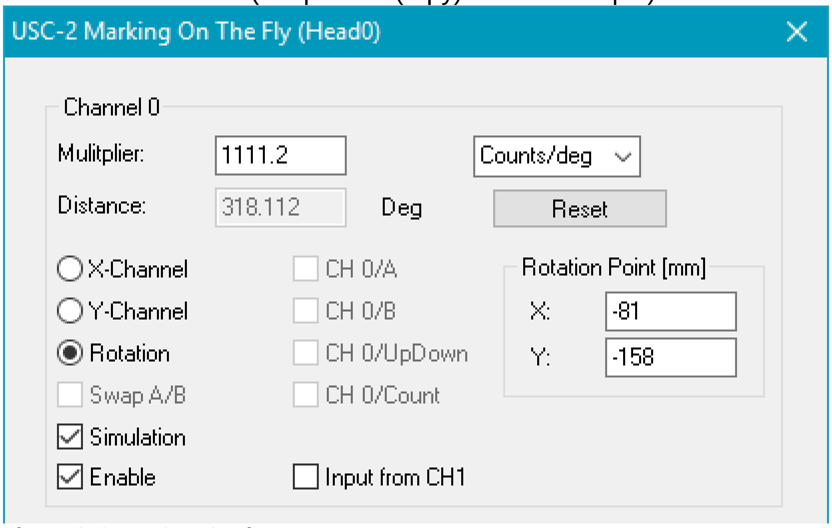


Figure 350: RMOTF Settings

d. Start marking
e. The result should be the same as in step two.
f. If the result is different from step two:
   i. The Multiplier takes effect on the length.
   ii. The Rotation Point takes effect on the radius.

4. Step four:
   a. Repeat step three until you get the same output as in step two. Radius and length must fit.

5. Step five:
   a. RMOTF: on → enable Marking On The Fly in the Settings dialog.
   b. Rotary disk: runs with a defined speed (deg / s).
   c. Correct the Multiplier so that it fits to the disk speed. Note: Consider the algebraic sign!
   d. Start marking
   e. The marking result must be a point!

6. Step six:
   a. RMOTF: on → enable Marking On The Fly in the Settings dialog.
   b. Rotary disk: runs with a defined speed (deg / s).
   c. Remove the point from the job and draw a square.
   d. Start marking
   e. The marking result must be a square!

15.8.4 Rotated scan head

In order to enlarge the marking length possible within the scan field, the scan head can be mounted with an angle between the X-axis of the scan head and the X-axis of the assembly line. In this case, both Channels (Channel 0 and Channel 1) need to be enabled in the software and the channel which is not directly connected to the encoder should use the input from the other channel (activate checkbox "Input from CHx"). To use the maximal possible marking length, the scan head can be rotated 45° in respect to the moving belt. Then, the MOTF multiplier must be adapted. Due to geometric considerations, the MOTF multiplier needs to be reduced by a factor of *v(1/2)* in the case of a rotation of 45°.

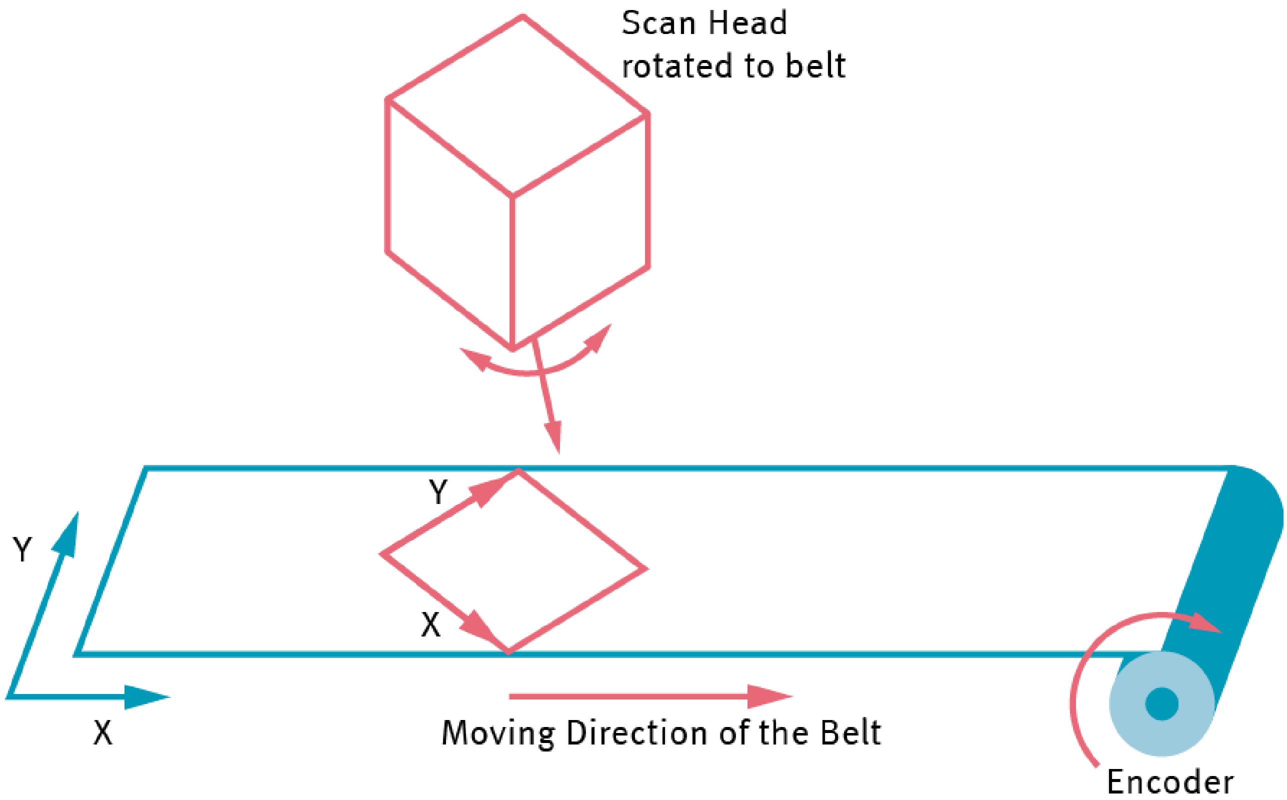


Figure 351: The scan head's coordinate system is rotated in respect to the belt's coordinate system.

16 Option Flash (USC-2/3 only)

The Flash option is available with an USC-2 or USC-3 card. In the past it was also available with an USC-1 card together with an FEB-1 board. The operation of the flash program is the same for these two setups. The FEB-1 (Flash Extension Board) is an optional card to the USC-1 which allows to store marking jobs which then can be executed in a stand alone mode means without a PC involved. The control in stand alone mode is possible via RS232 commands or the USC-3. USC-2 or USC-1 and FEB-1 digital IO's can be used to select the desired job and start/stop the marking process. The job preparation is done with a PC based editor. The jobs are uploaded to the flash over the same way you connected the card to SAMLight (USB or Ethernet).

Figure 352: USC-2 Card

16.1 Supported Objects

The job preparation is done like for normal marking jobs without flash In SAMLight. Not all functions of SAMLight are available in Flash. It is recommended to use the Flash Compatible Mode (FCM) to set up a Flash job.
In the following, supported and not supported objects are listed. The FCM checks the job and settings automatically for incompatibilities.

Flash supported objects/features:

- Entities:
  - o Geometries
  - o Hatch
  - o Dynamic serial numbers (up to 64)
  - o Serial number barcodes [a]
  - o Date time [b]
  - o Text2D
  - o Groups (mark loop count)[c]
- Pen properties:
  - o Skywriting
  - o Power ramping
  - o Drill mode (Use Geometry and Mark Lines as Dots included)
  - o Pen paths
  - o Perforation
  - o Defocus (not for USC-2 with Optic3D)
- Control objects:
  - o ScTimer
  - o ScWaitForInput [d]
  - o ScSetOutput [d]
  - o ScSetAnalogOutput
  - o ScMotionControl
  - o ScMotfOffset
  - o ScWaitForTrigger
  - o ScOverride
- Marking on the fly (MOTF)
- Optic3D (USC-3):
  - o Static 3D vectors
  - o Dynamic 2D objects with static Z offset
  - o 3D MOTF compensation
  - o 3D Surfaces
    - with static vectors
    - with dynamic vectors (not for STL)
- Motion control (stepper type 14)

Flash-only features:

- Flash Control Interface (FCI)
- Grayscale Bitmaps (USC-3 only) with dynamic exchange
- B&W Bitmaps (USC-3 only) with dynamic exchange
- G-Code conversion

Not supported objects/features in flash:

- AnalogIn
- Client Control Interface (CCI)
- I/O:
  - o Control inputs
  - o Emergency Stop
  - o Laser: Control (Timeout), Shutter control
  - o Message Input
  - o Motion Control (except stepper type 14)
  - o SAMLight Status outputs
- Grayscale Bitmaps (USC-2)
- SAMLight options:
  - o MultiHead
  - o Optic3D, 3D Surfaces (USC-2)
  - o SAM3D
- PartCounterMode, Quantity
- Pen settings:
  - o Defocus (USC-2 with Optic3D)
  - o Drill PointPowerMap
- Red pointer
- References to other text elements <$EntityName>
- Control objects:
  - o ScExecutable
  - o ScWaitForInput - Message
- Serial numbers:
  - o as Barcode: types [a], from file, format, human readable text, limits, wounding
  - o as Date time: format [b], language, automatic set/reset of DST (daylight save time)
  - o as Text: line spacing, limits, alignment, swap lines, marking order, barcode reader mode, Merge Character Outline
- Special sequences (pre-/post-processing)
- Splitting, Step&Repeat, Bitmap Rotary
- Use RS-232 as output (USC-3)
- Pen Path for serial numbers and date time entities

[a]: Only the following barcodes can be generated in standalone mode dynamically:

- DataMatrixEx
- QR Code EX,
- Dotcode

- PostNet-A/-C/-Cp
- Code-128 (2)
- EAN-8
- 2 of 5 EX
- Code-39 EX

For DataMatrixEx, QR Code EX and Code-128 (2) it is necessary to activate the checkbox 'Cellmode' in the barcode extended dialog.

[b]: The following date time formats are not supported in flash: %1d, %2d, %3d, %4d, %1M, %2M, %3M, %4M, %1y, %2y, %4y, %1g, %2g, %z, %Z.

[c]: For lasers with adjustable pulse length, the pulse length will be set for each loop in Flash mode. This can lead to differences between Flash Jobs and SAMLight jobs. If this is unwanted, pulse length 0ns needs to be used for the loop. For pulse length 0, the special laser sequence is not executed - meaning the last pulse length stays active. In order to set the correct pulse length, at the beginning of the job, mark the content of the loop once with the desired pulse length.

[d]: When JobIOSelectMode is used, make sure I/O bits are not in conflict with the JobIOSelectMode sync bits.

WARNING

On the flash, the field correction and laser source (CO2, YAG, LEE, IPG, ...) information are global. Mixing up jobs generated with different settings can lead to unpredictable results.

The global settings must be saved to the USC card by clicking the 'STORE' button in Settings→System→Optic→Advanced.

16.2 Flash Jobs and Settings

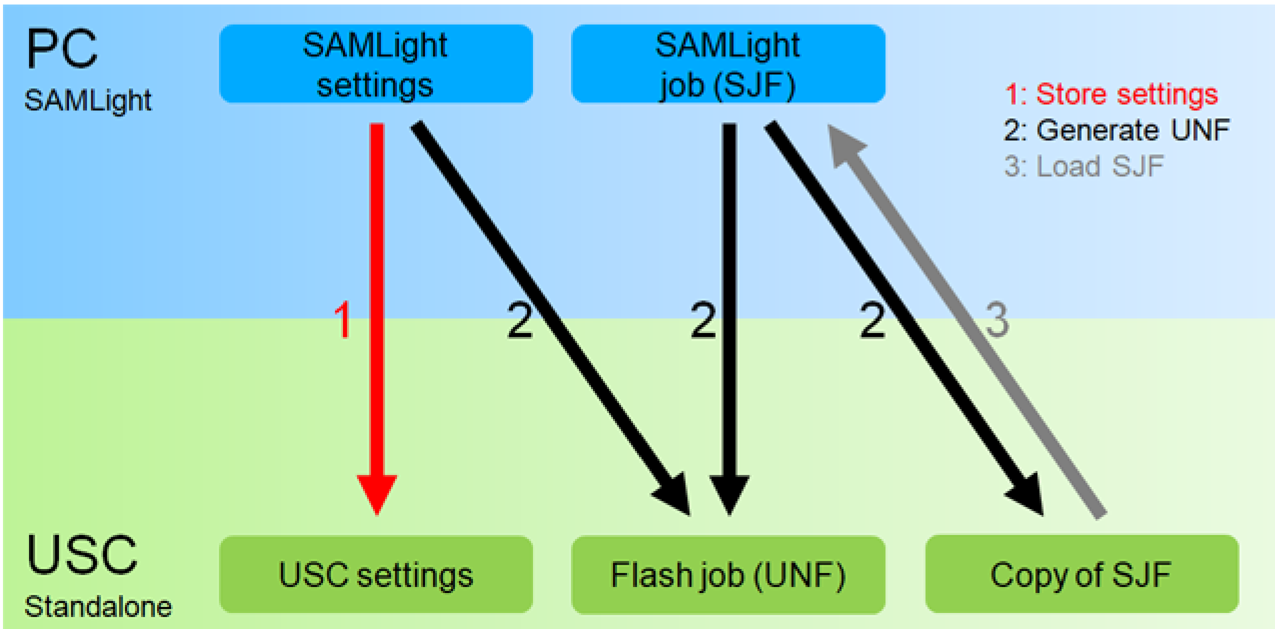


Figure 353: Handling of jobs and settings

The handling of jobs and settings for the USC standalone mode (Flash) as seen in figure 353 is explained in the following:

- **1: Store settings:** For standalone mode, the USC settings are used. Therefore, it is necessary to store the SAMLight settings on the USC card via the Store button. Please refer to figure 354 and figure 355.
- **2: Generate UNF:** For standalone mode, the Flash job (UNF) is created from the SAMLight job (SJF) together with a subset of the SAMLight settings and is saved on the USC card and in <SCAPS>\jobfiles\.

Different SAMLight settings in combination with the same SJF will lead to different UNFs. Furthermore, a copy of the SJF is also stored on the USC card as part of the UNF file. This SJF file cannot be addressed individually.

- **3: Load SJF:** The copy of the SJF on the USC card can be loaded to SAMLight. However, the USC settings and the UNF cannot be loaded to SAMLight.

NOTE *Different SAMLight settings in combination with the same SJF will lead to different UNFs.*

Flash Jobs Handling:

The file formats SJF and UNF differ in several aspects:

- in the SJF format, objects are stored with the information of all position coordinates in mm and thus can be freely edited each time, the job is loaded. When you change the settings of a pen, the change will be valid for all objects which have this pen applied.
- in the UNF format, the information of coordinates of objects is no longer stored in mm. Instead, the position information is converted into bit values. The same is true for pen settings, which means, that the information of pen settings is already converted to direct commands. Thus, you don't have real pen settings any more to change.

A Flash job can be created / loaded via one of the following ways:

- Create UNF from SJF or CNC and load to card:
  o SAMLight → Extras → Flash
  o Client Control Interface Command ScProcessFlashJob
  o G-Code conversion with Flash command CGF

- Load existing UNF to card:
  o SAMLight → Extras → Flash
  o Client Control Interface Command ScProcessFlashJob
  o Visible USC Server → Flash → Load
  o FTP Server and Flash command JLA

Flash Settings Handling:

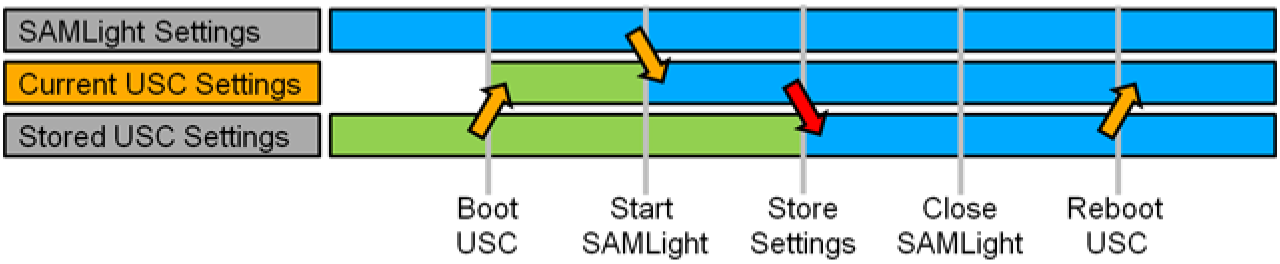


Figure 354: Correct handling: SAMLight and USC settings with proper storing

The current USC settings on the USC card change with different events, see figure 354:

- At boot of the USC card, the current USC settings are loaded from the stored USC settings.
- At SAMLight start, the settings of SAMLight overwrite the current USC settings.
- To avoid loss of these current USC settings after reboot of the USC card (see Fig.355), the current settings need to be stored. Storing the settings on the USC card can be done via the Store.
- The SAMLight settings remain as current USC settings even after SAMLight has been closed.
- The storing ensures that the SAMLight settings and the current USC settings are identical even after a reboot of the USC card.

Common mistakes without storing of settings:

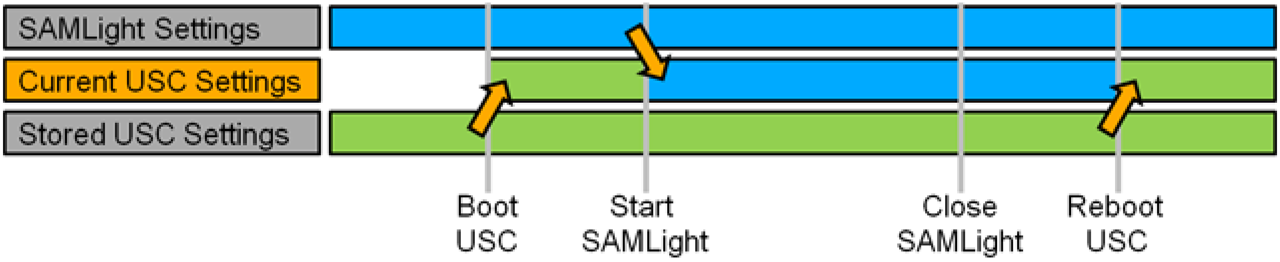


Figure 355: Common mistake: SAMLight and USC settings without storing leading to change of settings after reboot

The current USC settings on the USC card change with different events, see figure 355:

- At boot of the USC card, the current USC settings are loaded from the stored USC settings.
- At SAMLight start, the settings of SAMLight overwrite the current USC settings.
- These SAMLight settings remain as current USC settings even after SAMLight has been closed.
- **But, when the USC card is re-powered, the current USC settings are loaded from the stored USC settings. Then, if the current USC settings do not match the SAMLight settings, the behaviour of the USC card could be different than before.**

16.3 Flash compatible mode in SAMLight

The flash compatible mode (FCM) is meant to:
- help creating jobs for the flash
- check the compatibility of existing jobs with flash
- check the compatibility of existing settings.

The flash compatible mode is reachable via toolbar which needs to be activated in *Settings → System → View → Toolbars*. Once activated, a new toolbar will be visible, see figure 356.

Figure 356: FCM toolbar disabled (gray)

The flash compatible mode itself is enabled and disabled by clicking on the checkbox next to the gray flash. With this activation, options which are not available in flash are grayed out in SAMlight. So, the flash compatible mode helps in creating a valid flash job (*.unf).
In addition, the current settings and a currently loaded job will automatically be checked for compatibility with flash. If no incompatibility was found, the gray flash will turn green. If any incompatibility with flash was detected, the flash icon will turn red and a message box is coming up giving information on the incompatibilities. A click on the red flash will also do the check and bring up the message box.

Figure 357: FCM toolbar enabled - flash incompatible (red)

Figure 358: FCM toolbar enabled - flash compatible (green)

NOTE

FCM can be activated without Flash license so that the user can always check the compatibility.

The message box consists of three columns and can look like the following example:

Item	Location	Reason
Alignment - Vertical	EntityList > ScSerialNumbers2D > "MyText" > Text2D > Extended	must be Default
BarCode type "Code-39"	EntityList > ScSerialNumbers2D > "MyBarcode39" > BarCode	not available

Item	Location	Reason
Cellmode	EntityList > ScSerialNumbers2D > "MyDMC" > BarCode > Advanced	required
Control Input	Settings > System > IO	not available
DateTime Format "%1d"	EntityList > ScSerialNumbers2D > "MyDT" > DateTime > Edit Format	not available
DateTime Language	EntityList > ScSerialNumbers2D > "MyDT" > DateTime >	must be English
Optic3D	Settings > System > Optic > Advanced > Correction	not available

Table 50: Example strings of the FCM message box

The message box can be used to change the settings and the job such that it will be compatible with flash. The column item gives information on which item produces a conflict with flash, the column location gives information on where to find this item and the column reason gives a reason for the incompatibility.

To fix any incompatibility mentioned in the list:

- disable the flash compatible mode (click on the checkbox next to the flash icon)
- navigate to the location
- do changes according to the list.

16.4 Job processing

In general the job preparation will start with the editor which allows to setup fixed and variable geometries together with marking parameters and process flow elements. Also more general operation modes like the laser source, scanner field settings including correction file assignments and other modes like marking on the fly will be defined there.

The prepared job can be saved in SJF format for later reload or it can be converted to UNF format which can be executed by the flash. Uploading and executing of SJF files itself is not possible. By default the editor will attach the SJF file to the UNF file. This allows a reverse processing of a generated UNF file to its original SJF file which finally then can be reimported into the editor.

The general dialog to achieve this functions will be started by *Menu bar → Extras → Flash*.

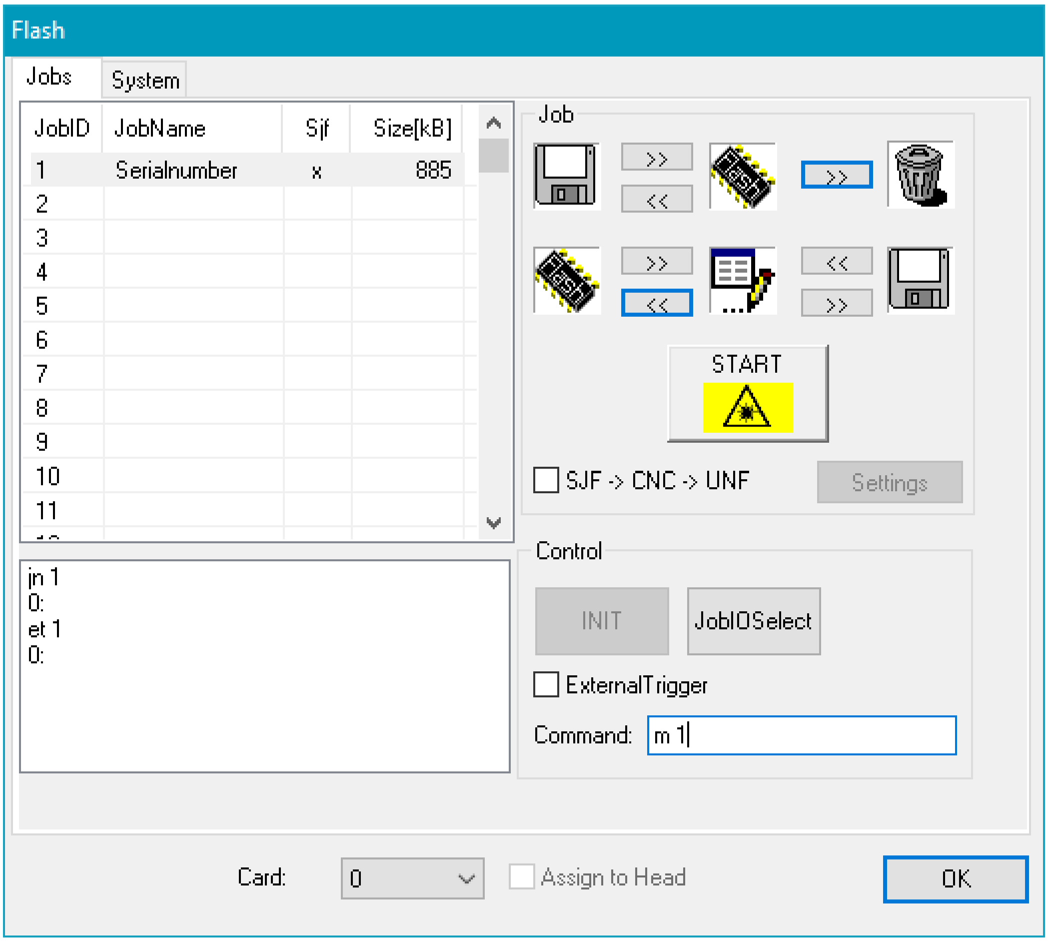


Figure 359: Flash Main Dialog

16.4.1 Up/Download

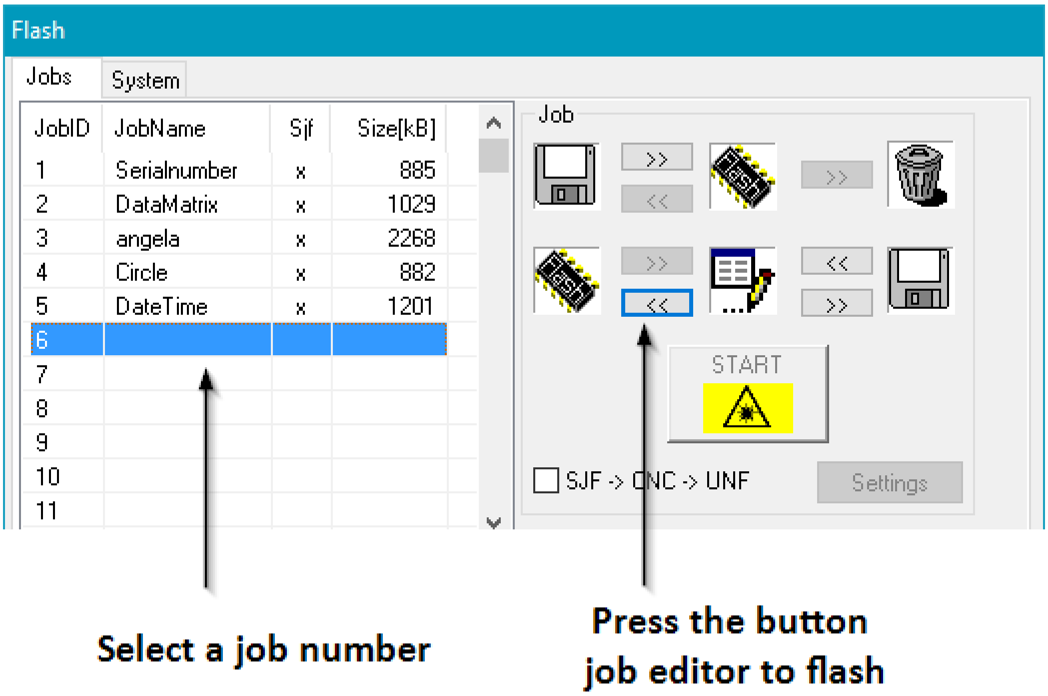


Figure 360: Upload from editor to flash

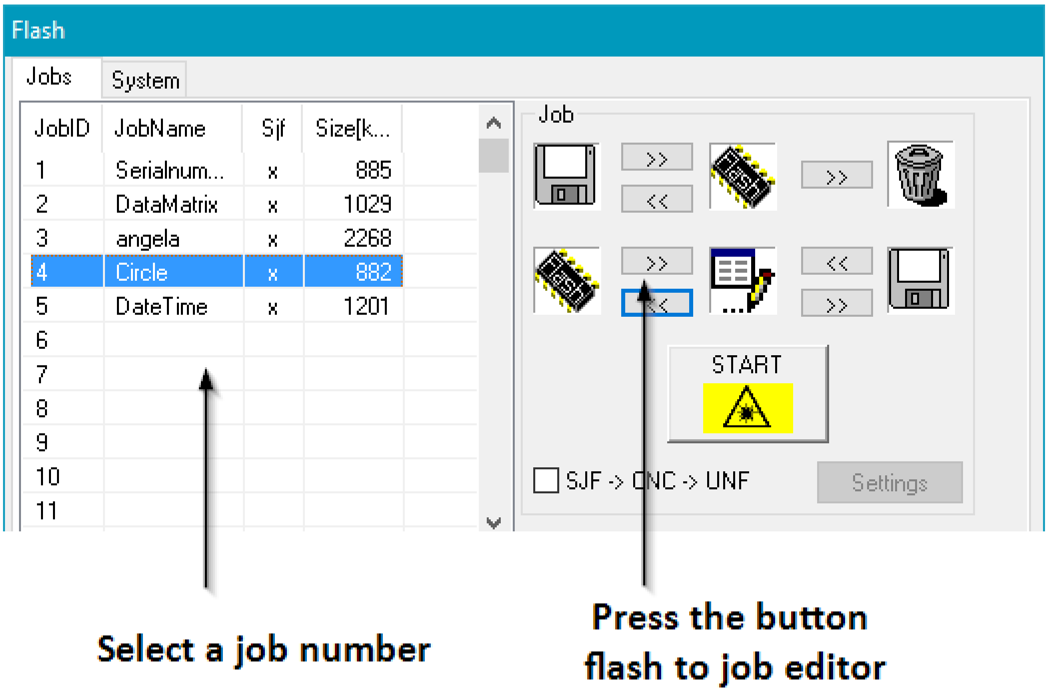


Figure 361: Download to editor from flash

16.4.2 Execution

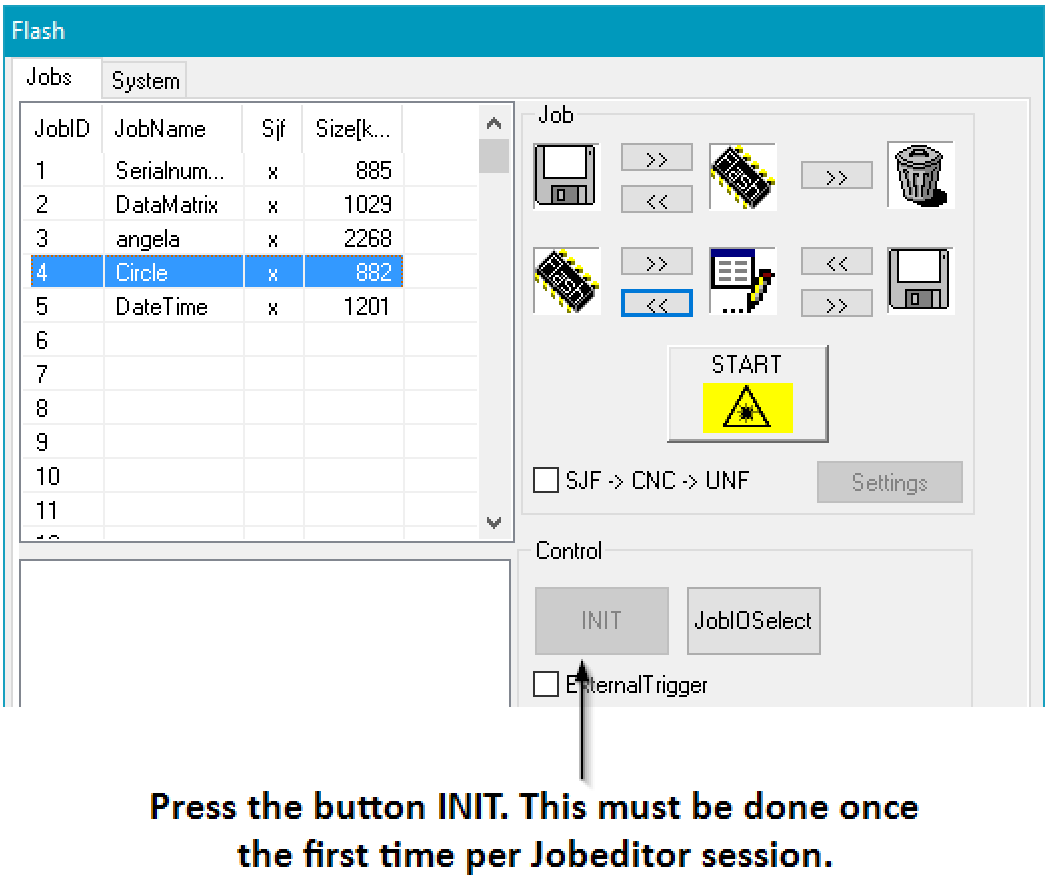


Figure 362: Initialisation of the Flash

The INIT button is only used by USC-1 + FEB-1, it is not used by USC-2 cards so it is grayed out.

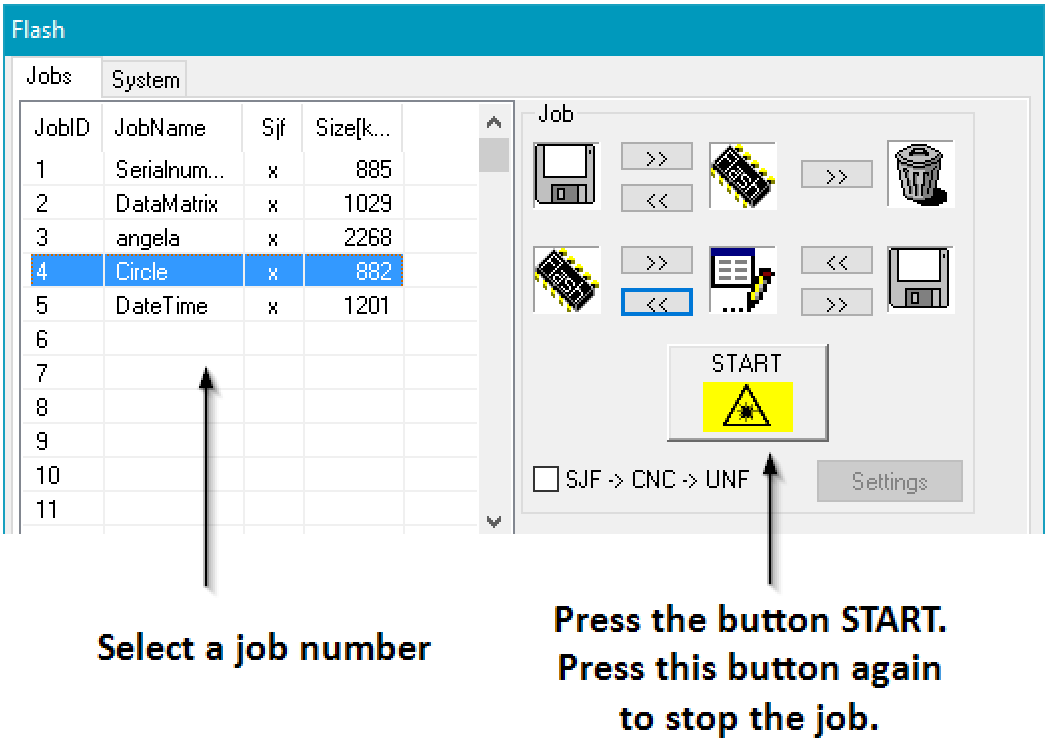


Figure 363: Start/Stop the job

16.4.3 Flash Job IO Selection

By clicking on the button *"JobIOSelect"*, the Flash Job IO Select mode can be activated in Flash mode. Then jobs can be loaded and marked via the OptoIn bits of the USC-2 or USC-3 card. See chapter Flash IO Job Selection.

Once the Job IO Selection mode is entered in the Flash mode, the marking can be started only with an external trigger. To exit the Job IO Selection mode you have to press the "*STOP IOSelect"* button in the flash menu.

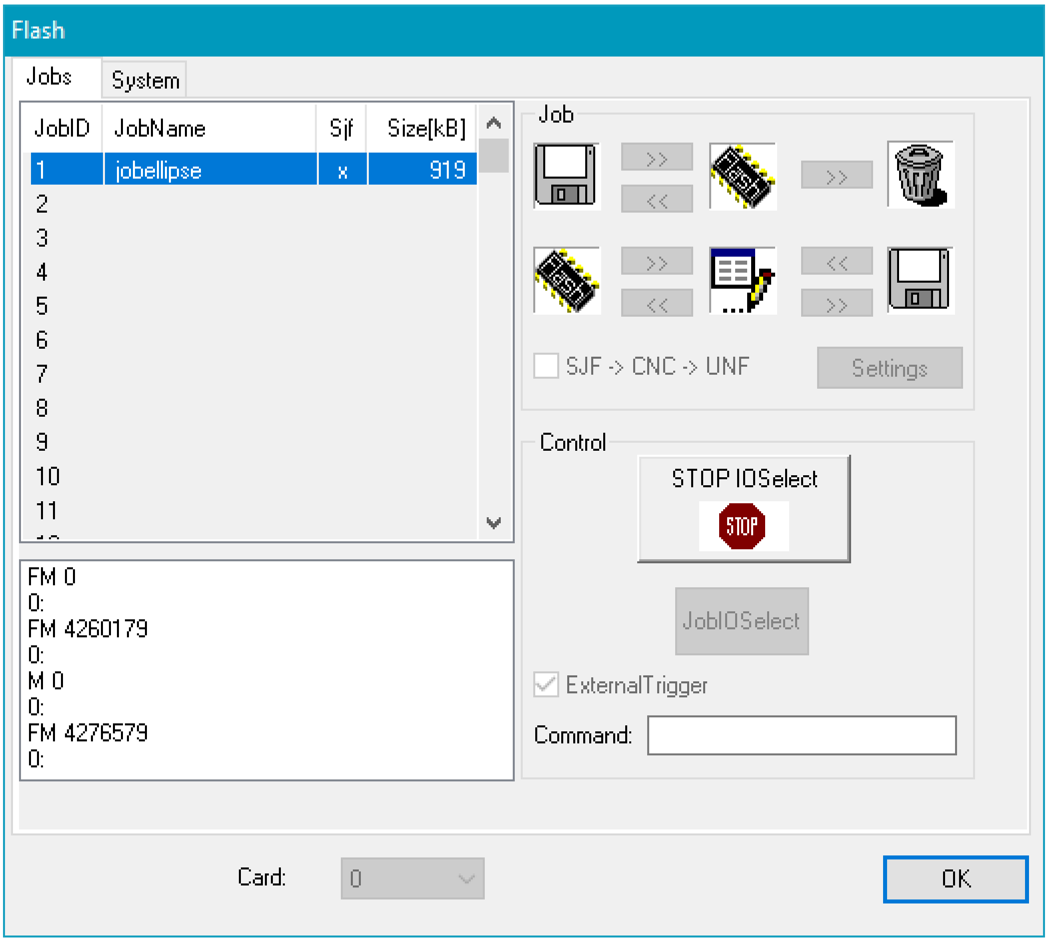


Figure 364: JobIO Selection in Flash Mode

16.5 System

The system property page is used to initialize the flash memory, update the system section and to define the boot configuration in stand alone mode. It also shows some status flags for diagnostics purpose.

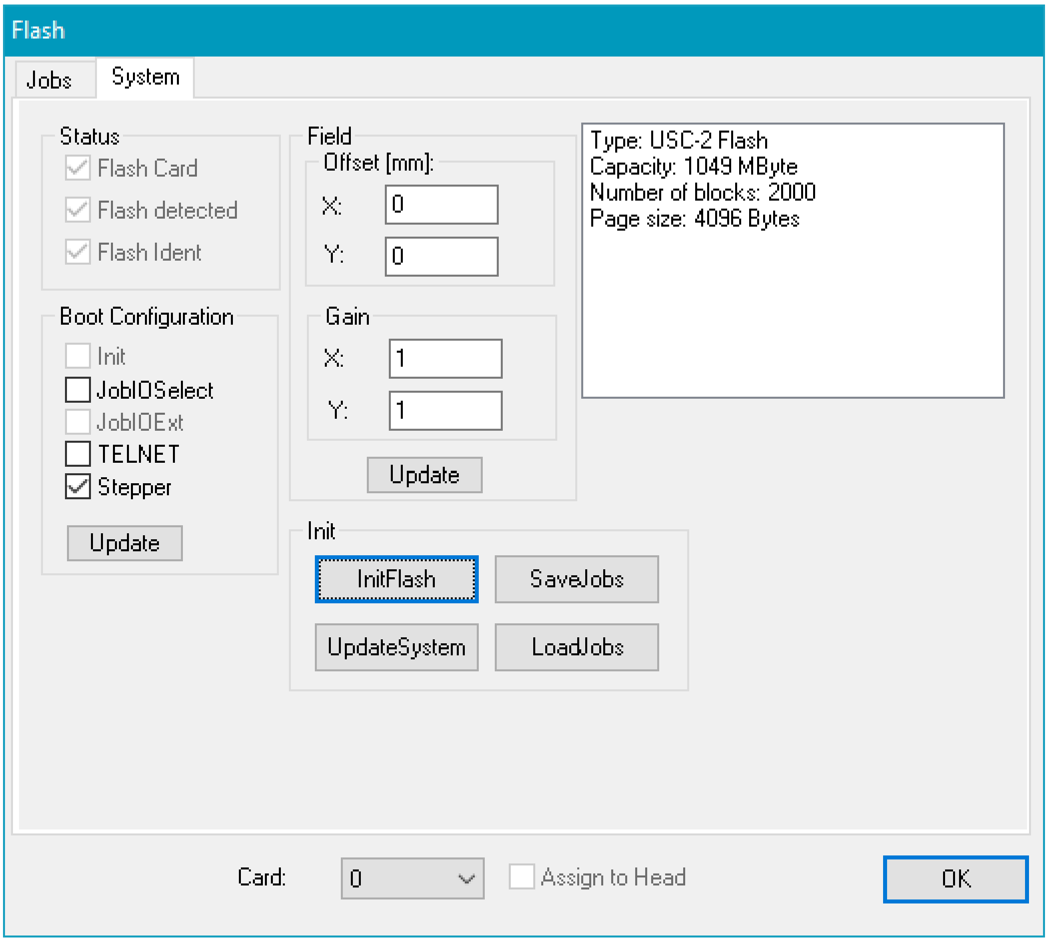


Figure 365: Flash System Dialog

Status:

Flash Card: Shows that a flash card hardware is detected.

Flash detected: The flash software runs and has returned the flash type.

Flash Ident: The flash was successfully initialized and has stored valid information.

Boot Configuration:

InitFlash: Flash card starts and makes an auto initialization. Asks if you want to keep your jobfiles (by using *SaveJobs* and *LoadJobs*). All other settings on the USC-2 card will be restored to default values. Please save your settings to USC-2 again by clicking *Menu bar → Settings → System → Optic → Advanced → Store*.

JobIOSelect: The Flash card starts in *JOBIOSelect* mode.

JobIOExt: In combination with JobIOSelect, this mode enables the inputs [0..7] of the Extension connector, so that 255 jobs can be selected. Only available with USC-2 card.

TELNET: It is possible to send commands to the card via TELNET Port 23. Only available with USC-2 card.

Stepper: Shows if stepper mode is used. You can enable it in USC server (visible mode).

Click Update and repower the USC-2 card for changes to take effect.

Init:

InitFlash: Erases the complete flash. All data is lost.

UpdateSystem: Updates the flash system block only. Job information is preserved.

SaveJobs: Saves the jobs stored on the flash to *<SCAPS>\jobfiles\save_jobs_head_X_YYMMDD*, while X is the head number and YYMMDD the date.

LoadJobs: Loads the jobs saved by *SaveJobs* to the flash.

16.6 MultiCard

The most common application is to run up to six USC-2 cards in standalone mode (flash mode) and use one SAMLight license to prepare job files and do the flash job management of all cards.

WARNING *The feature 'MultiCard' is included in the license 'Flash' and is not compatible with MultiHead.*

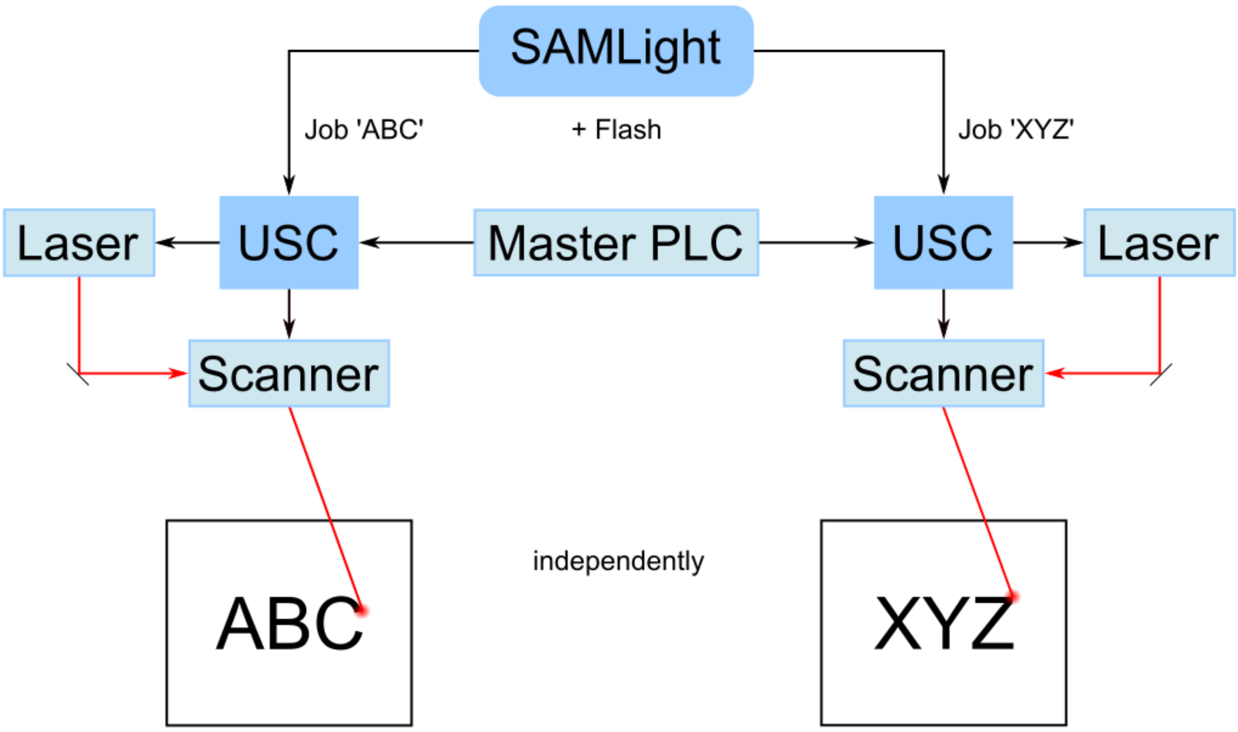
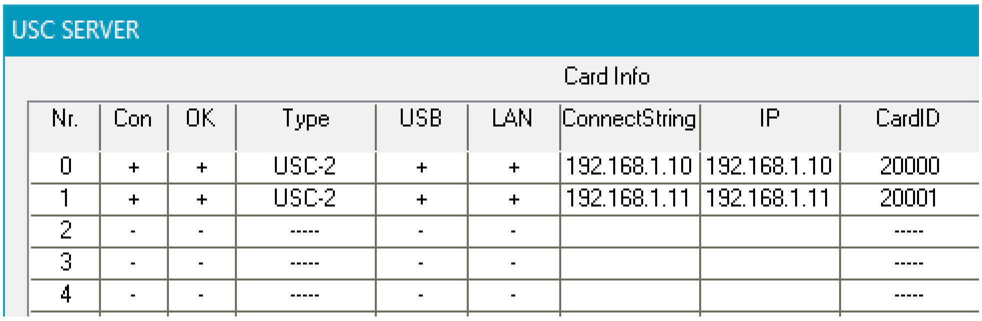


Figure 366: Schematic representation of a MultiCard construction

1. Close SAMLight and start the server in visible mode (<SCAPS>\system\sc_usc_server /v).
2. Connect only one USC-2 card at a time via both USB and Ethernet.
3. Select the USC-2 card in the server and click on the USC2-LAN button and define the Ethernet settings. Leave the dialog with the Connect button. After that you can check the '+' sign in the LAN column of the usc server.
4. Repeat steps 2 and 3 for each USC-2 card.
5. Unplug all USC-2 cards.
6. Define the order of the USC-2 cards in '<SCAPS>\system**sc_usc_card_ids.txt**'. The order of the dongle IDs in this file corresponds to the head (card) number as you can find it in the mark- and flash-dialog.
7. The IP addresses of every USC-2 card should now appear as a connect number in '<SCAPS>\system**sc_usc.cfg**'. The server attempts to connect only to these IP addresses. The order of the IP addresses in this list does not matter.
8. Enable the checkbox 'MultiCard' in sc_usc_server (<SCAPS>\system\sc_usc_server /v).
9. Connect all USC-2 cards via Ethernet, double check the order of the cards in the visible server.
10. Go to <SCAPS>\tools\sc_setup.exe → HardwareSettings and configure each USC-2 card individually and save the changes to the settings file.
11. Do not forget to click on the Store button in the Driver Settings for each card to save the settings to the USC-2 cards as well.
12. Finally the setup is done. Any card connected to the sc_usc_server can be controlled in different ways:
    a. SAMLight: Marking directly in SAMLight (not in flash mode)
    b. SAMLight: Flash job management in Extras → Flash
    c. SAMLight: Flash job management with Client Control Interface calls

d. Telnet: Use the FTP server to add jobs, use Flash Control Interface calls to manage flash joblist
e. sc_usc_server: Flash job management with the Save, Load and Delete buttons in '<SCAPS>\system\sc_usc_server /v → Flash'



Nr.	Con	OK	Type	USB	LAN	ConnectString	IP	CardID
0	+	+	USC-2	+	+	192.168.1.10	192.168.1.10	20000
1	+	+	USC-2	+	+	192.168.1.11	192.168.1.11	20001
2	-	-	-----	-	-			-----
3	-	-	-----	-	-			-----
4	-	-	-----	-	-			-----

Figure 367: MultiCard setup in sc_usc_server

```
Connect0=192.168.1.10        20000
Connect1=192.168.1.11        20001
Connect2=                    0
Connect3=                    0
```

Figure 368: Connect strings in sc_usc.cfg Figure 369: Head (card) assignment in sc_usc_cards_ids.txt

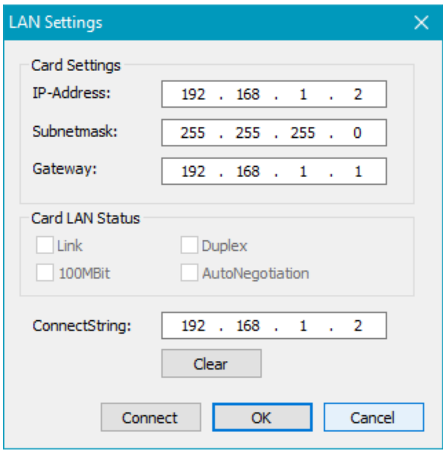


Figure 370: LAN Settings Dialog

Card Settings:

Here you can set the IP address, subnetmask and gateway of the USC-2 card.

ConnectString:

The IP adress that is defined here will be listed in the file 'sc_usc.cfg'. The server will try to connect to those IP adresses only which are listed in this file.

Connect:

By clicking 'Connect' the sc_usc_server will write the card settings to the connected USC-2 card.

17 Multiple Heads

There are several ways to work with more than one card.

- MultiHead: The 'MultiHead' mode requires more than one scanner controller card, each card controlling a scan head and a separate laser. This mode allows the marking of different data on the two or more heads. To use this feature, the optional SAM software license *MultiHead* is required. All heads are sharing one job file which is split on the several scan heads.

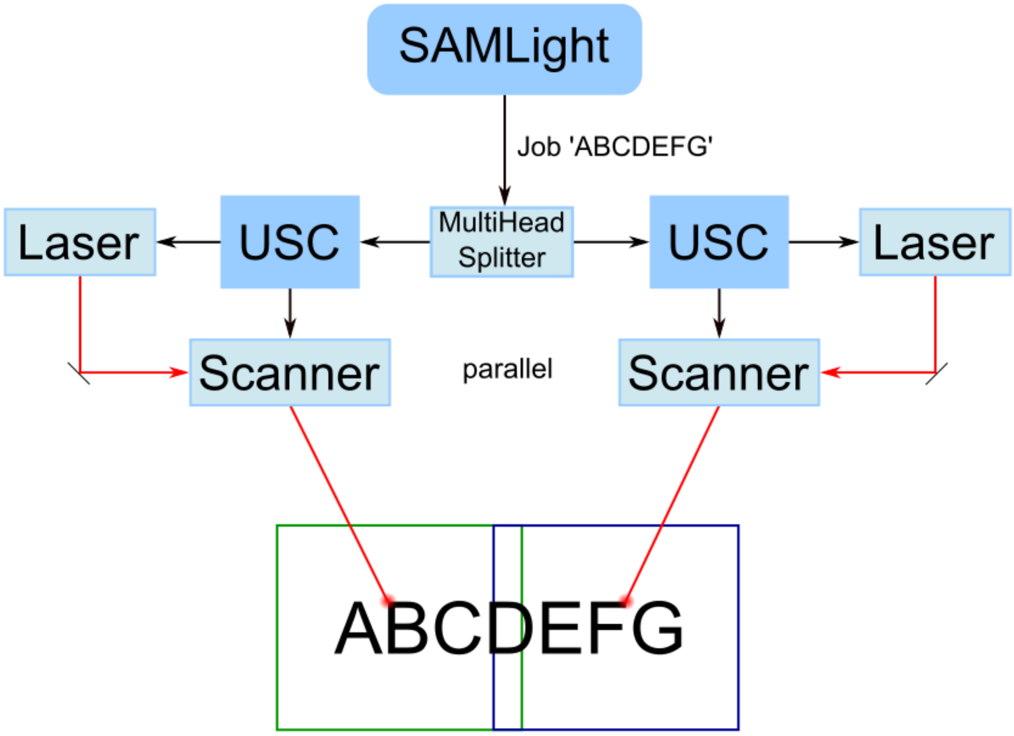


Figure 371: Illustration of a MultiHead construction

- Head2: This mode allows to use two heads connected on one scanner card (if the scanner card is able to do this). Both heads are marking the same data from the same job file at the same time. This mode is included in the SAMLight or the SAM Standard Components, in order to be able to use the second head, the license extension *Head2* is required.

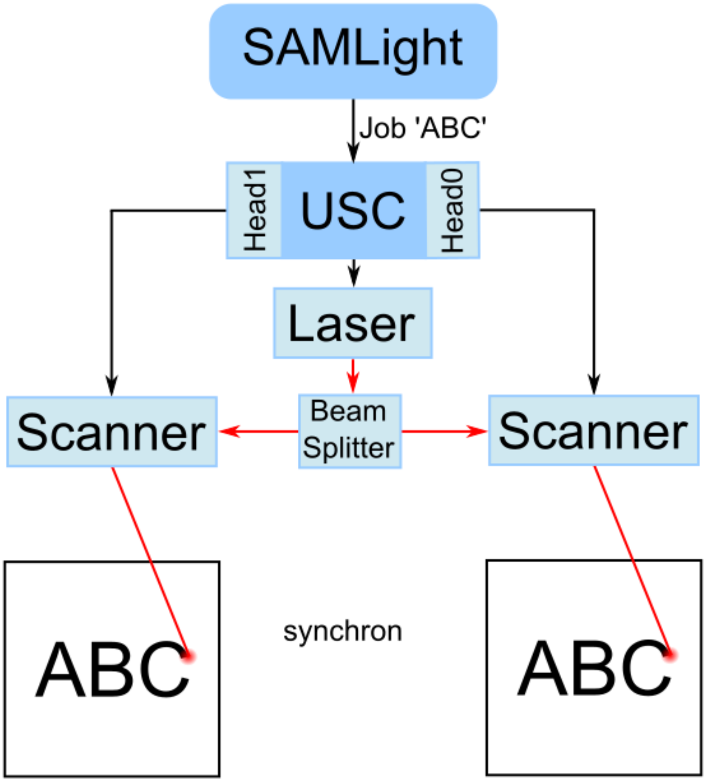


Figure 372: Illustration of a Head2 construction

- MultiCard: The feature 'MultiCard' is included in the license 'Flash' and is not compatible with MultiHead. The most common application is to run up to sixteen USC-2/3 cards in standalone mode (flash mode) and use one SAMLight license to prepare job files and do the flash job management of all cards.

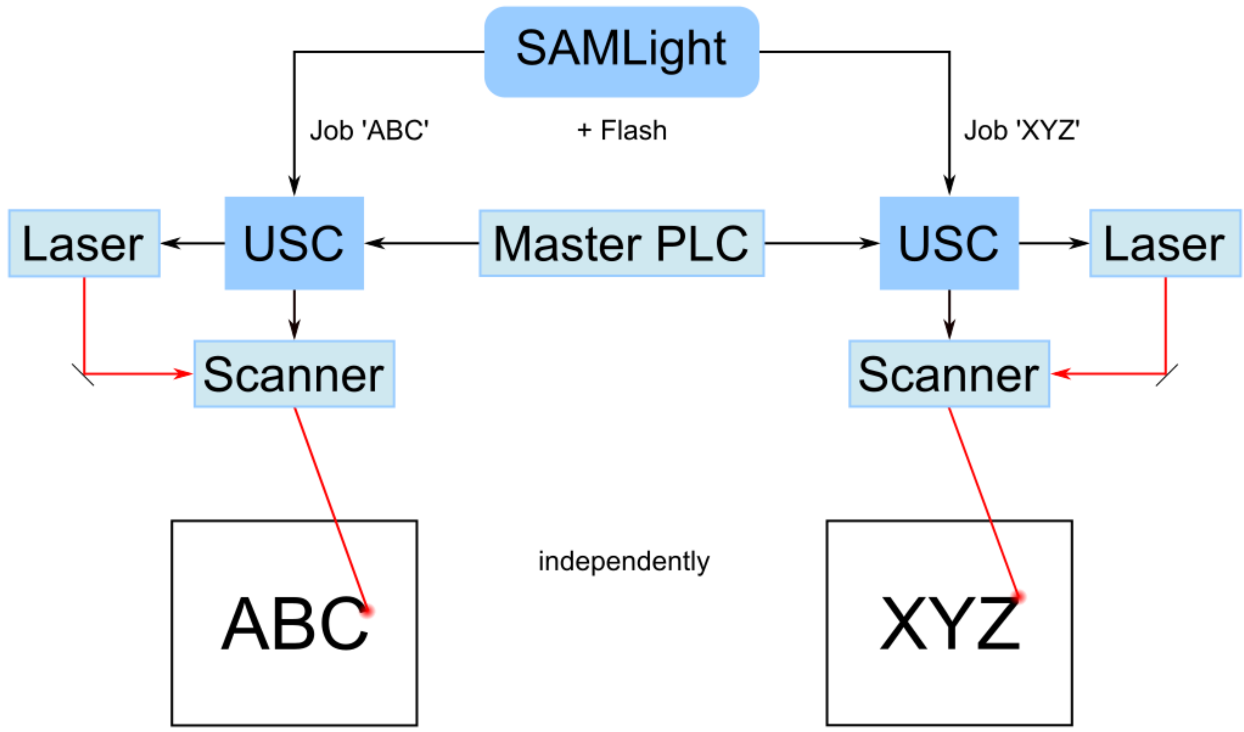


Figure 373: Illustration of a MultiCard construction

- MultiInstance: With the feature 'MultiInstance', more than one instance of SAMLight can be started on the same PC. Command line parameters are used to assign settings and a card to each instance. This mode is included in the SAMLight or the SAM Standard Components, no extra software option is required.

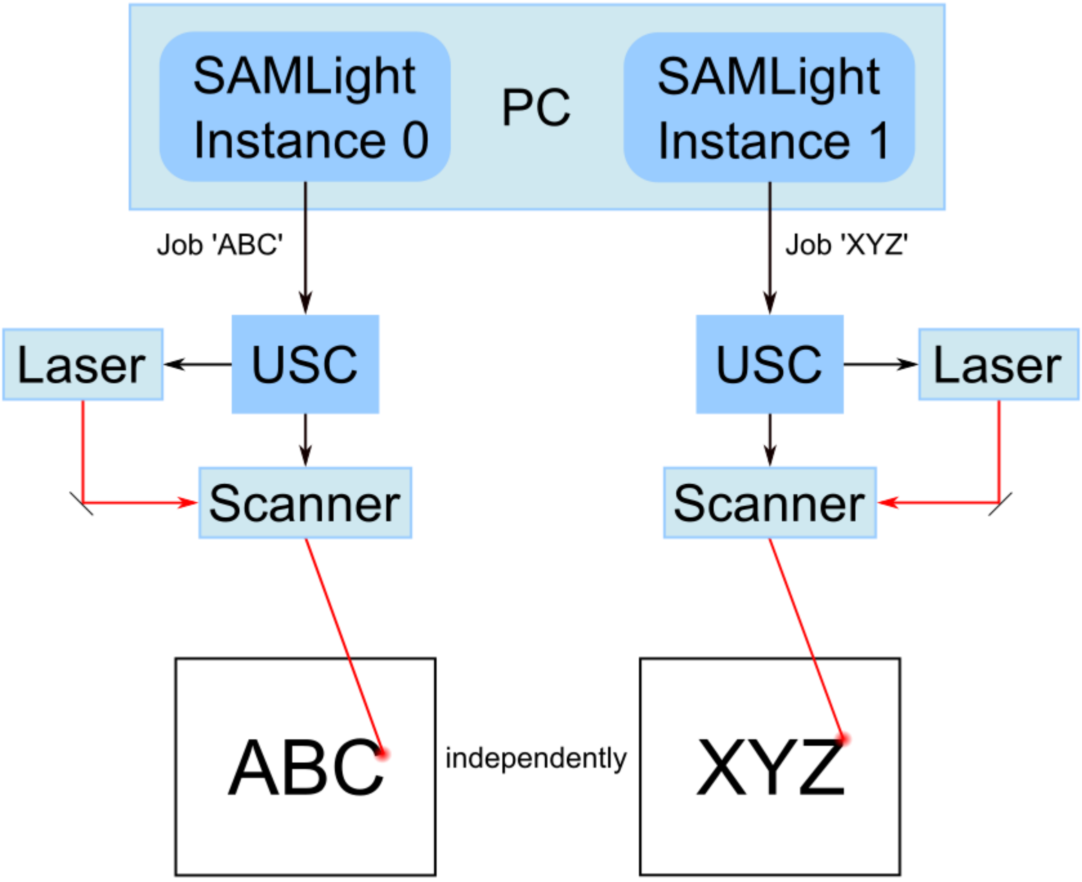


Figure 374: Illustration of a MultiInstance construction

17.1 Option MultiHead

The Option MultiHead allows to build up scanner applications with a simultaneous vector output to up to six scan heads. In this case there will be one job file for all scan heads.

NOTE

For this simultaneous output and the installation of more than one (up to 6) scanner driver cards, the MultiHeadOpticModule license is required. If the license for the MultiHeadView is present, the View2D shows all installed working areas with the overlapping region. The data is edited as there would be one big output. The MultiHeadView provides automatic splitting functionality to see on which head the vectors belong.

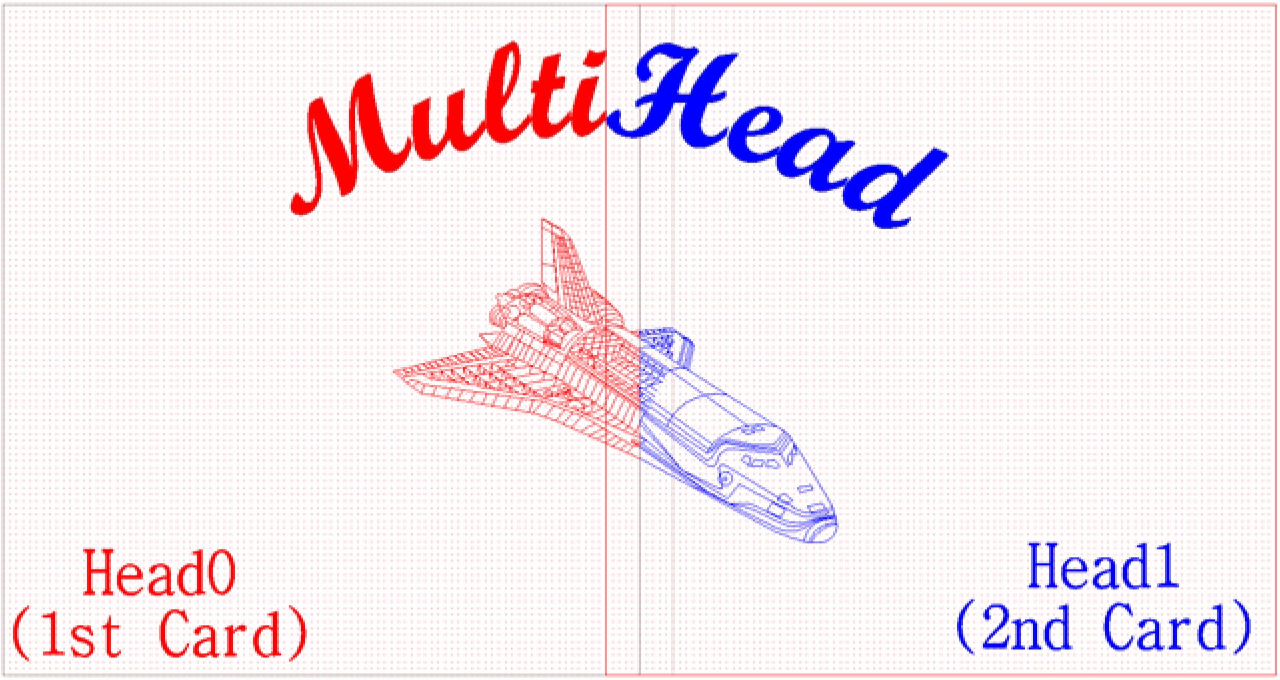


Figure 375: Job with two different working fields for two different scan heads

External trigger signal (MultiHead):

Trigger mode: Each card needs an trigger input signal. All cards must have finished their marking before the next job execution can start.

Mark Dialog with "External Trigger": This is not the same as trigger mode. Here the trigger input for all cards is card 0.

Control Objects (MultiHead): When a ControlObject is part of the job all cards wait for all cards to finish their previous marking process then the ControlObject is executed.

ScMotionControl: Step and direction signals are generated only on card 0.

ScSetOutput, ScOverride, ScSetAnalogOutput: These control objects act on each card.

ScWaitForInput, ScWaitForTrigger: After all cards have finished their previous marking process, each card is waiting for an input signal on itself.

NOTE *Motion Control: the step and direction signals for motion control devices (stepper type 8 and 14) are only provided at the first card (Head0). The necessary settings are defined in the chapter Motion Control Settings.*

NOTE *Marking on the fly (MOTF): the encoder signal for the MOTF compensation has to be provided to each card. The corresponding MOTF settings has to be defined for each card as well. For more information please also refer to your scanner controller card manual.*

NOTE *MultiHead mode currently does not support the Power Ramping and the Speed Ramping functionality.*

17.1.1 Installation

There are four steps to getting started to use multiple heads with multiple cards:

1) Activating software with a password containing the MultiHead license
2) Definition of the hardware settings in the Setup Tool
3) Definition of optic settings
4) Definition of automatic vector splitting in the View2D

17.1.1.1 Password

The software is delivered together with a dongle and a password. To use the multi head components, the password has to include these components. If there is a previous installation on the PC, the following steps are necessary:

1) Install the new software under the same installation folder as the old one. Default folder is C:\scaps\sam2d\.
2) Delete or rename the file sc_##_"dongle_number".scl.
3) During the first start of the SCAPS software there appears a dialog for typing in the new password.

17.1.1.2 Setup Tool

The settings for the software are saved in a *.sam file located in the folder *<SCAPS>\system*. For the Standard2D software the name of this settings file is *sc_settings.sam*. For SAMLight it is *sc_light_settings.sam*. This file also stores the head count and the type of the installed cards. The file can be edited with the *sc_setup.exe* program located in *<SCAPS>\tools*. To do this it is necessary to close all SAM applications before.

Starting *sc_setup.exe* and selecting menu point *Hardware Settings* shows following dialog.

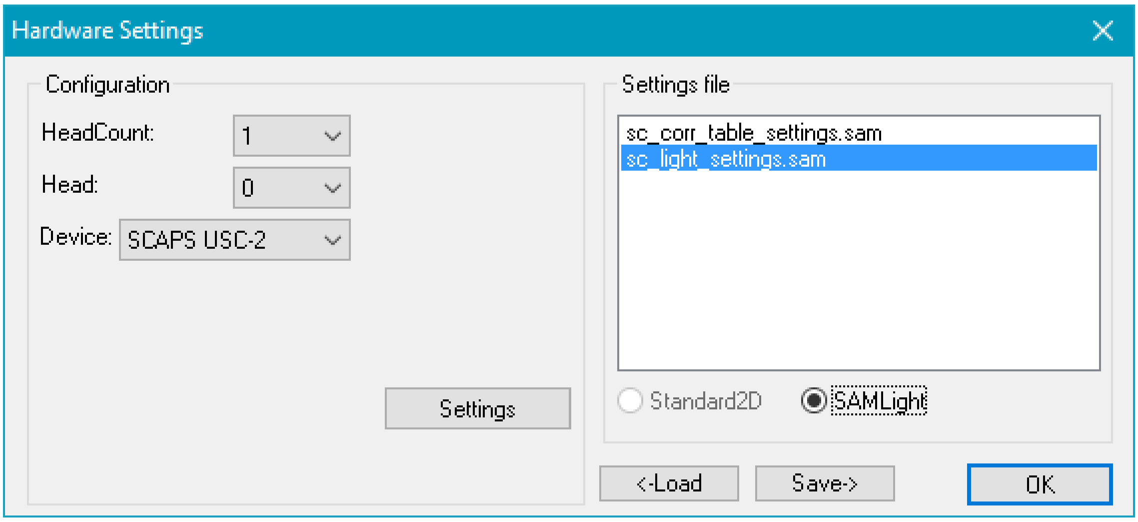


Figure 376: Hardware Settings Dialog

To change your settings, the following steps are required:

1) The software looks for all *.sam files located in <SCAPS>\system\.
2) Select your settings file and press <-Load.
3) Define the total head count.
4) Select the head and the installed device for this head. Repeat this step for all installed heads.
5) Press the Settings button to define the optic settings for the heads.

17.1.1.3 Optic Settings

Get this dialog by clicking on *Settings* at the *Hardware Settings* dialog. See chapter Setup Tool.

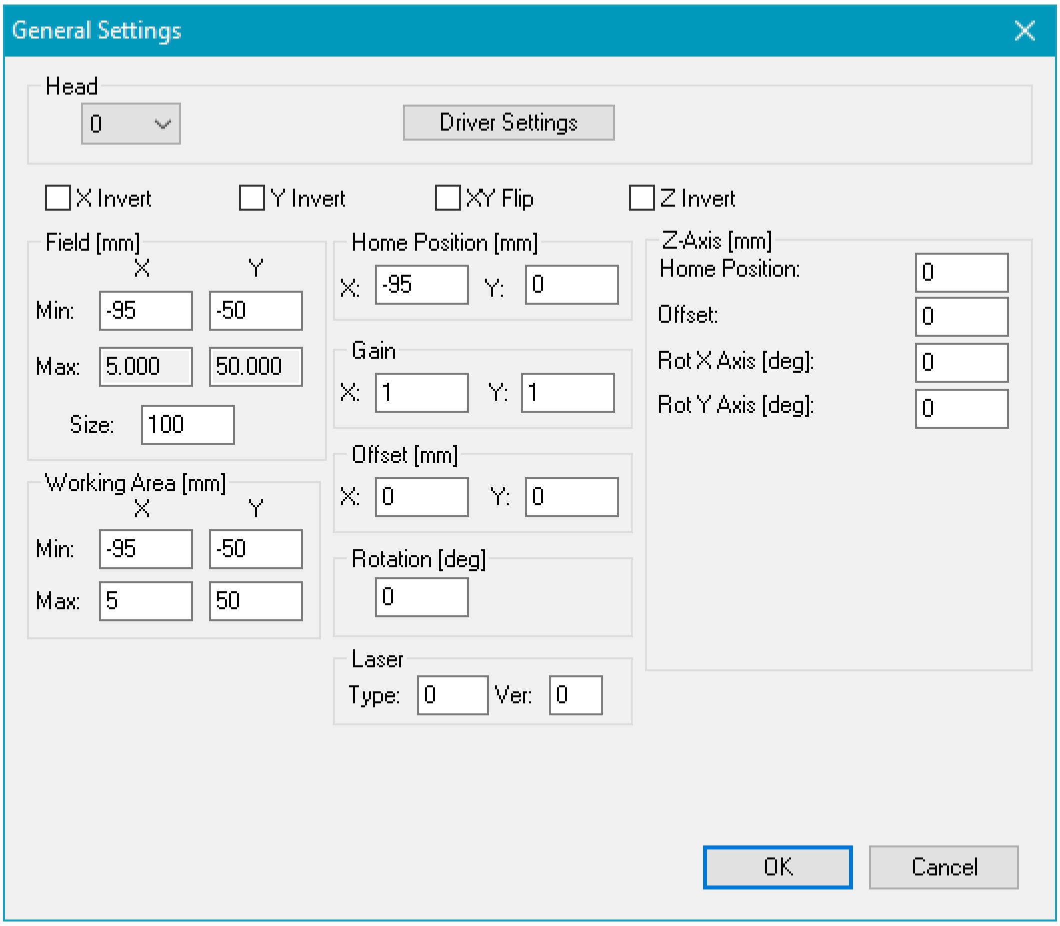


Figure 377: General Settings for Head 0

Select the head with the box at the top. Then define the field and the working area for this head. The values shown in the dialog above define a field with 100 mm size for each head. The total center in the middle of the two fields and an overlapping area of 10 mm in x direction.

The corresponding values for head 1 are shown below:

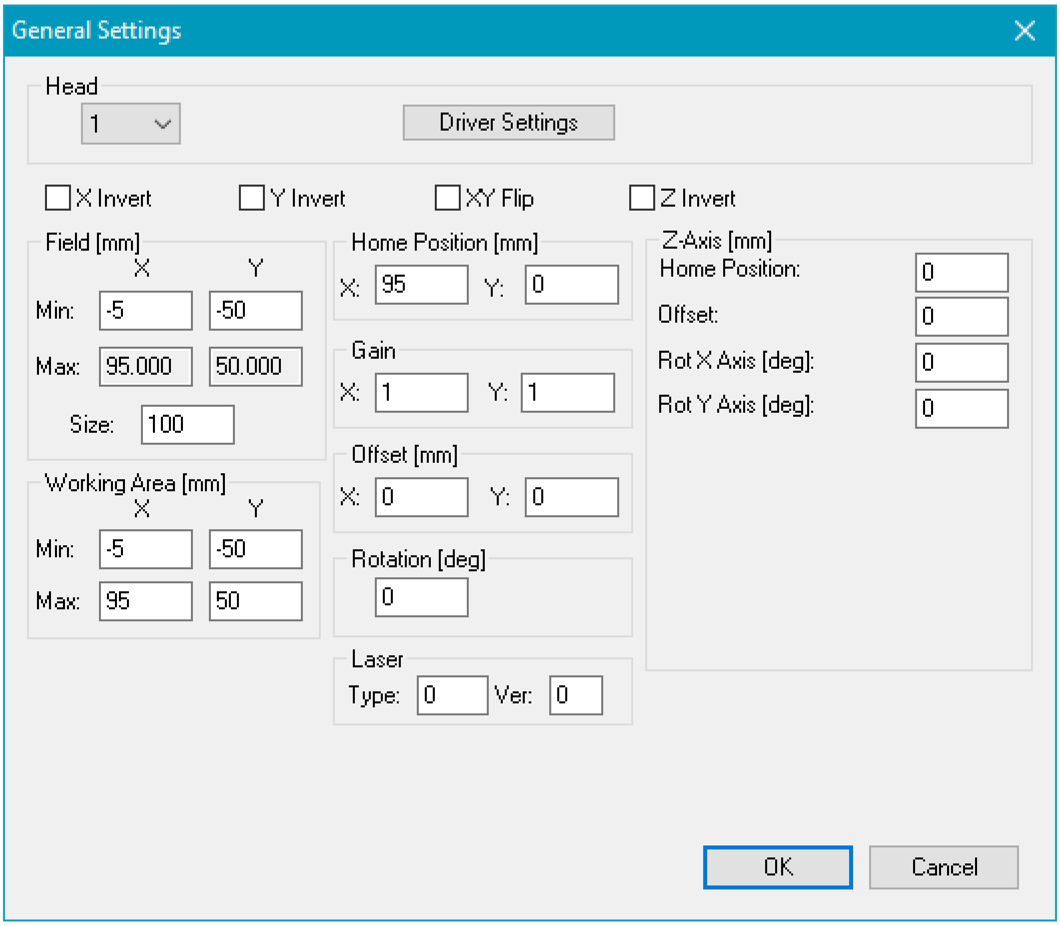


Figure 378: General Settings for Head 1

When the field and the working area have been defined, the driver specific settings have to be defined for each head. Select Head 0 and press button Driver Settings. Here define the specific settings as correction file location, the DLL files etc. for the scanner card. Repeat this step for all installed heads/cards. The optic specific setting can also be changed within the software by selecting *Menu bar → Settings → System → Optic*.

In the View2D for MultiHeads you will see the resulting working area for both heads.

17.1.1.4 View2D

If more than one head is set up, there are some additional functions, such as the algorithm for splitting vectors and the head assignment for each vector.

When the software starts, the View2D shows the working areas for all installed heads and the overlapping region. See the screenshot in the chapter Multiple Cards.

The *split* function can be called directly from the context menu (click right mouse button in the View2D) → *MultiHeadSplit*. There is also an *AutoSplit* function available. It can be activated within the *ViewProperties* dialog. This dialog can be reached by the context menu (right click of your mouse) and selecting *ViewProperties* or by a direct call to the *ViewProperties*:

Figure 379: Context menu (left) and properties (right) of View2D for MultiHead

Checking the *AutoSplitMode* recalculates and splits the vectors after each change of the job. For not too complex jobs this option is very helpful. The head color mode shows the drawings in head specific colors. Once the *Head Color Mode* checkbox is activated, the colors in the view2D for MultiHeads do not tell which pen is used but instead the colors mean which head is used for marking the entity.

NOTE

In case of an error message "Galvos out of range" when the marking should start, the Splitting has not been activated. To solve the problem, right click in the View2D and either do a MultiHeadsplit or activate the AutoSplitMode in ViewProperties → MultiHead.

It might also be a problem of the dimension check. Please select "No Check" in the OpticModuleProperties.

17.2 Option Head2

This mode allows to send the same data to two heads through one scanner controller board.

NOTE

The scanner controller has to have this feature.

Example:

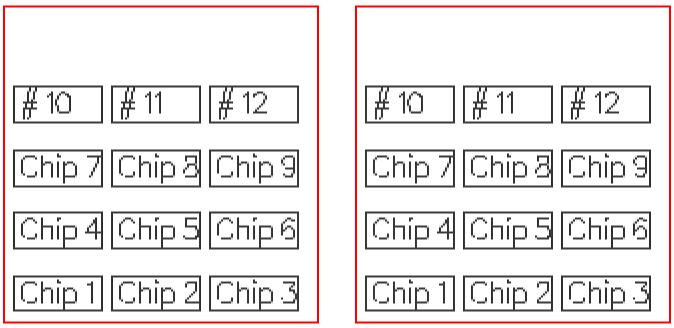


Figure 380: Example for Head2, both scan heads are marking the same vectors

17.2.1 Installation

Here the Settings for the two heads with one card for an USC-2 card are described. The following dialog is achievable via *Menu bar → Settings → System → Optic*.

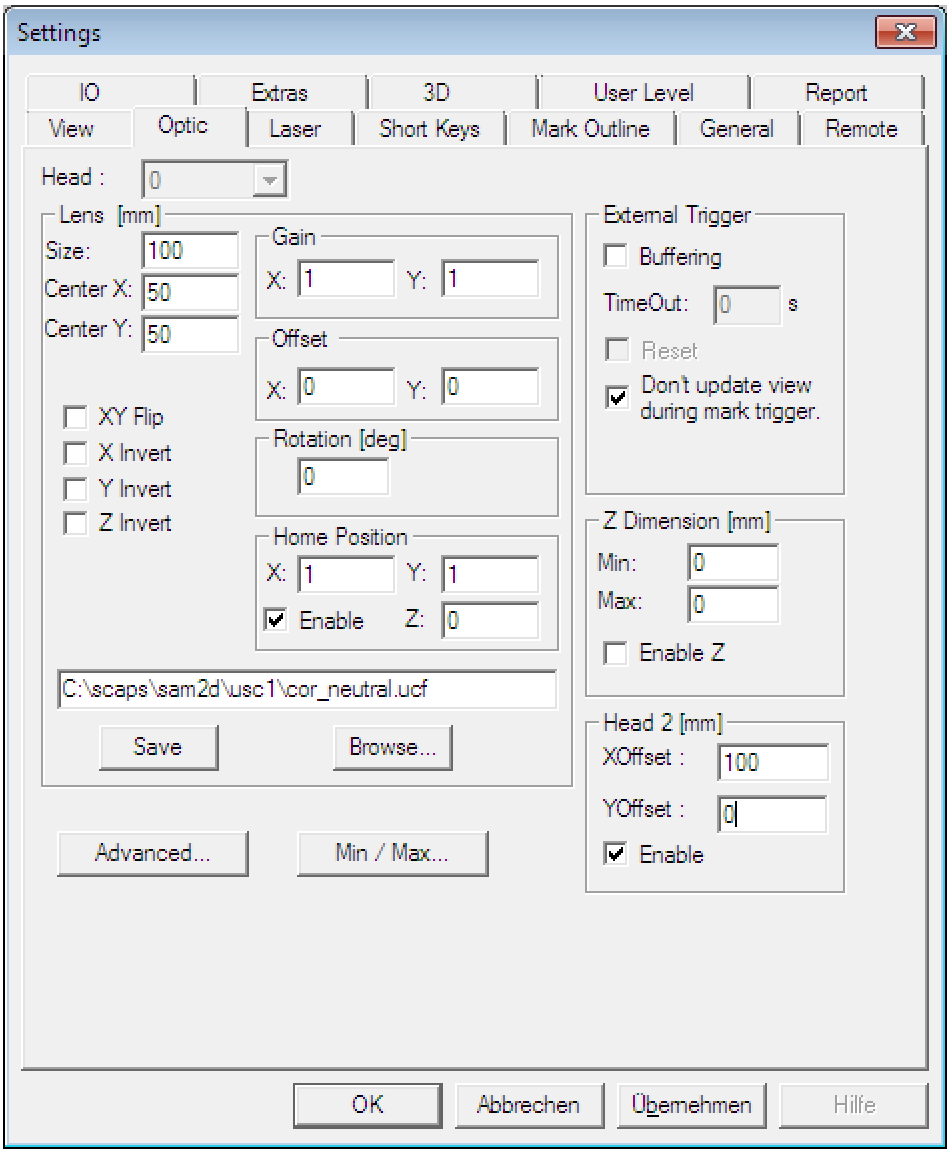


Figure 381: Optic Settings for Head 2

Step 1: Click on *Advanced... → Correction, Settings...* for the correction file dialog. From the drop down box select *2* and activate the *Enable* checkbox right from it. Now click on *Browse* and select the appropriate correction file for the second scan head. After doing this the second scan head is activated.

Step 2: This step is not necessary. In the lower right corner find the field *Head 2 [mm]*. Here enter the offset values for the second head in X and Y-direction. Finally enable the checkbox *Enable* to activate a *X* and *Y Offset* of the second scan head which will only affect the View but not the scanner output.

17.2.2 Fixed Job Offset

It is possible to define a *Job Offset* for the job that is used for the output to the secondary head. If the job offset is the same as the *Secondary Head Offset*, the output takes place at the same relative position inside the two fields. The maximum *Offset* in one direction is ±30000 bits of 65536. That corresponds to ±45 mm of a 100 mm working area or 45%.

Inside SAMLight, the *Job Offset* can be defined within the dialog *File → Job Properties*:

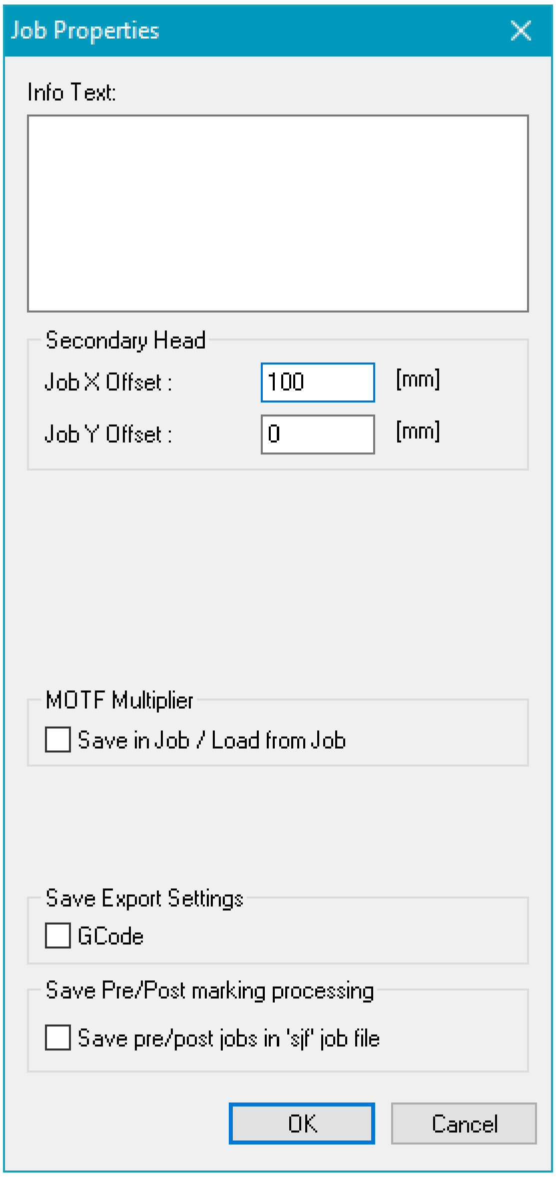
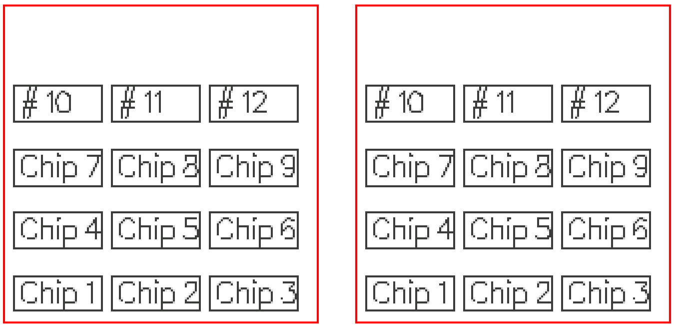
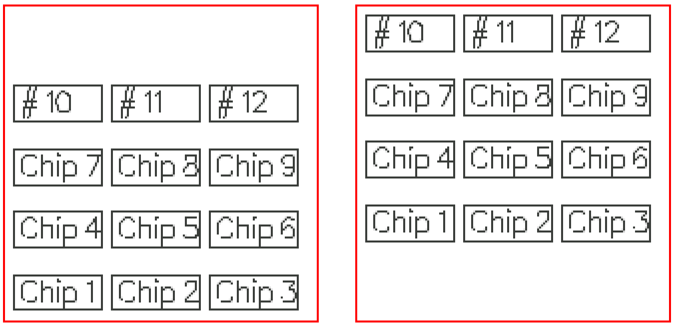


Figure 382: Job Properties Dialog

Inside the View, the job is displayed as follows:



The objects are on the same relative position within the two heads as the Job Offset is the same as the Head Offset. If we change the Job Offset to x = 100, y = 20, we get the following:



The Job Offset is saved within the sjf file.

17.2.3 Variable Entity Offset

For the secondary scan head, a relative offset to the fixed job offset with job properties can be defined with inserting a *ScSetSecondaryHeadOffset* entity. The specified offset in this entity is applied to the subsequent entities in the entity list.

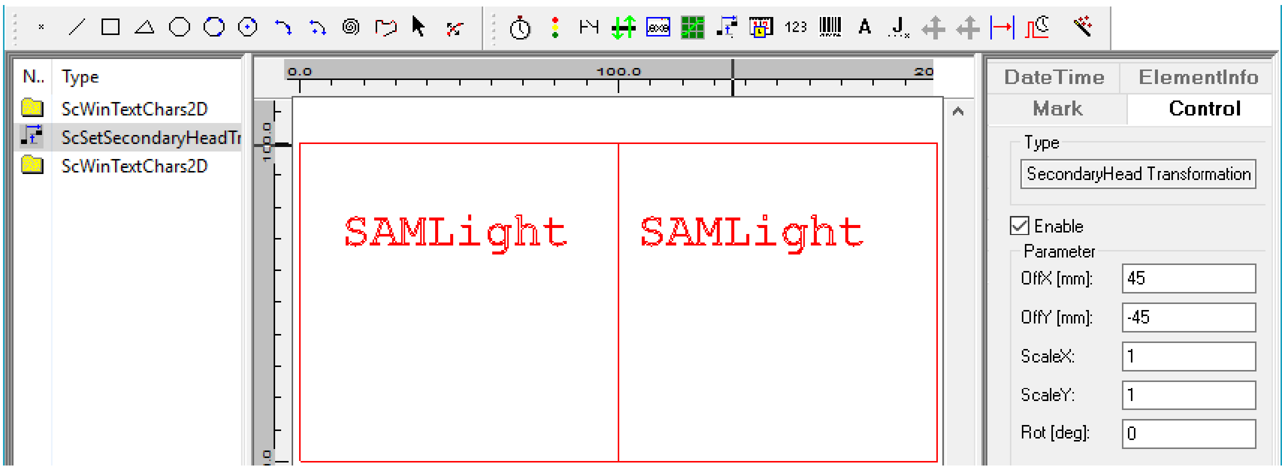
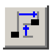


Figure 383: Secondary Head Offset

Click on the  button in the functionality object toolbar and a *ScSetSecondaryHeadOffset* entity is inserted in the entity list on the left. Select this new entity in the entity list and the property page *Control* on the right side becomes active. Click on it. To define an offset, the *Enable* checkbox has to be selected and the relative X and Y offsets have to be entered. Press the *Apply* button. Now a new entity, e.g. a copy of the entity before, is inserted in the entity list behind the *ScSetSecondaryHeadOffset* entity. When marking the entity list, the copy is marked now with the specified offset. A new subsequent *ScSetSecondaryHeadOffset* entity inserted into the entity list overwrites the specified offsets of the *ScSetSecondaryHeadOffset* entity before. Remember that the X and Y offsets of the *ScSetSecondaryHeadOffset* entity are always defined relative to the offset with job properties and that the total offset of *ScSetSecondaryHeadOffset* entity plus the offset with job properties has to be = 45% of the working area.

17.3 MultiCard (USC-2/3 only)

The description of the feature MultiCard can be found in the Option Flash chapter.

17.4 MultiInstance

'MultiInstance' can be used to run two or more instances of SAMLight on the same PC.

Requirements:
- 1 PC
- 2 or more USC cards (each card require a SAMLight license)

WARNING

It is important that the license is on the card. The license for the instance of SAMLight is selected from the card. USB dongles are ignored.

NOTE

If you would like to use 'MultiInstance', your PC need more computing power, due to doubled use of SAMLight.

Instances can not be launched at the same time. Since REL_4_1_5_20220203_0005 the next instance will be opened after the previous has been opened.

For a system with two card, initially two settings files are required. The file sc_light_settings.sam is copied twice and renamed to MultiInstance_0.sam and MultiInstance_1.sam. Then, two shortcuts of SAMLight must be generated, SAMLight MultiInstance 0 and SAMLight MultiInstance 1.

To differentiate between the two shortcuts, each shortcut needs its own settings and its assigned card. This is done using command line parameters. These can be assigned under properties of the respective shortcut.

For SAMLight MultiInstance 0 you will need the following command line parameter:
..\sam_light.exe /MultiInstance /ActiveCard=0 /Settingsfile=MultiInstance_0.sam

For SAMLight MultiInstance 1 you will need the following command line parameter:
..\sam_light.exe /MultiInstance /ActiveCard=1 /Settingsfile=MultiInstance_1.sam

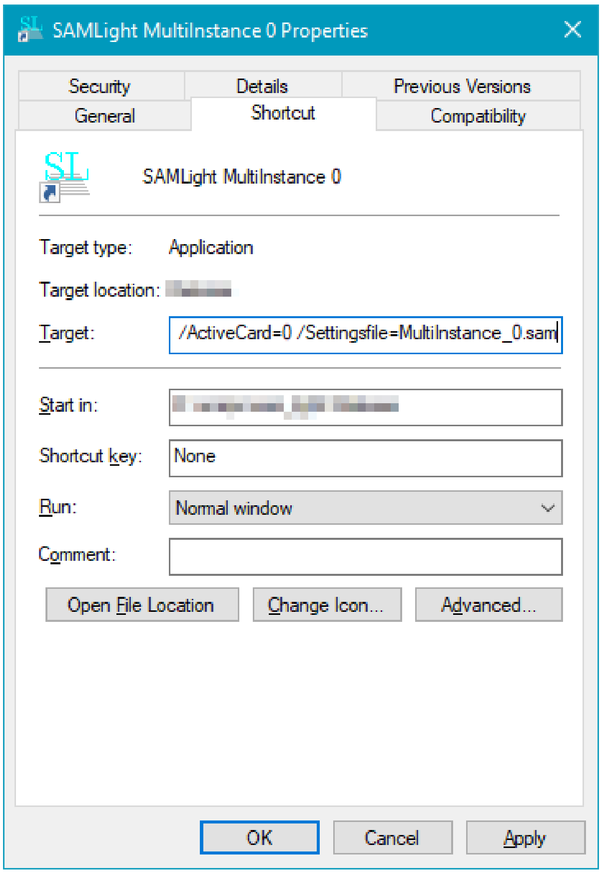


Figure 384: Property Window of SAMLight MultiInstance 0

You will find information on the Command line parameters in the chapter How to → Command Line Parameters.

Settings for MultiInstance in visible USC server:

- Map card for SAMLight MultiInstance 0 to position 0
- Map card for SAMLight MultiInstance 1 to position 1

Each SAMLight instance will show its corresponding Dongle number of the USC-card.

NOTE

Although there are two cards, SAMLight communicates with the cards via one usc server. The information is sent to the cards one after the other. The buffer filling of the cards is sequential. Problems can therefore arise with large amounts of data which can be solved by using one PC per USC card.
Make sure that MultiCard is not checked.

The instance of SAMLight can also be remotely controlled using other software via CCI. In the case of 'MultiInstance', CCI function calls do not work, but CCI TCP with different ports does work. For further information on CCI TCP, please take a look at the CCI manual.

If 'MultiInstance' is used, only motion types 8 and 14 can be used. Command line parameters are also used to assign a motion settings file to the instance. The command line parameter to use is /StepperMotionFile=<*.txt>.

18 Option FlatLense (USC only)

This option enables the Z data channel of the XY2-100 Interface of the USC card. It is required to control 3 axis scan heads. Even if this is an hardware option, it can be enabled by the FlatLense license for every USC card.

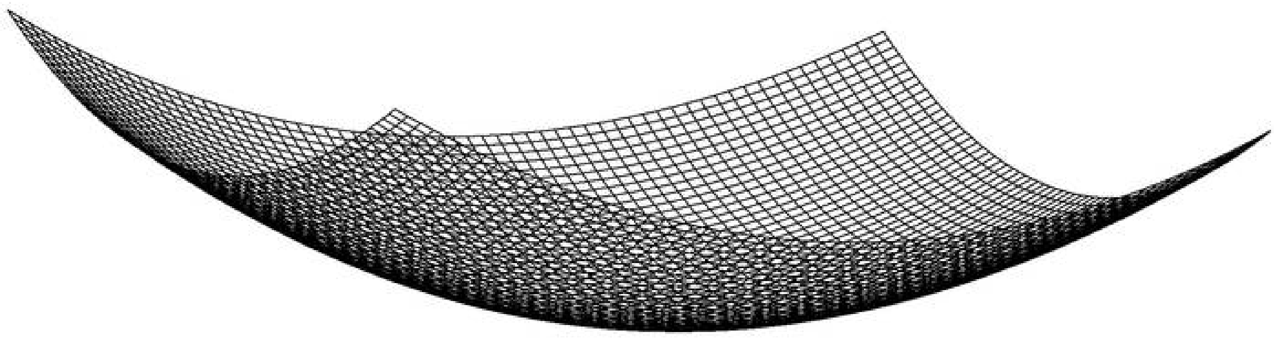


Figure 385: 3D UCF correction file

Flat surface marking without F-Theta lens: If the 3 axis scan head is used for marking on flat parts, the third scanner axis is used instead of a F-Theta objective to keep the laser focus constant on a plane surface, the option FlatLense is required and a 3D correction file has to be used which also contains Z bit values for every correction point.

Defocus: If you have a USC card with the option FlatLense (3D UCF correction file, no Optic3D) the software has no factor to calculate mm to bit. That is why the unit of pen defocus is [2^{16} bit / field size] instead of defined unit in SAMLight.

Real 3D vectors marking: If the 3 axis scan head is used for marking on curved parts (real 3D vectors), both options FlatLense and Optic3D are required.

If a RTC card is installed Real 3D vectors marking is being enabled with the 3D option on the card and the Optic3D option.

19 Option Optic3D

This chapter describes the 3D marking functionality. For example how to set up marking on a curved surface.

19.1 Features

The following features are implemented:

- Wrap objects around 3D Surfaces
- Import of 3D DXF files (DXF Files Version 2)
- Translation of objects in X,Y and Z direction
- Rotation of objects around X,Y and Z axis
- Manual manipulation of points and 3D vectors

19.1.1 3D Surfaces

Toolbar: Enable the 3D surface toolbar in *Settings → System → View → Toolbars*. The toolbar is showing mode drop-down list settings button and enable the checkbox at the right from the cog wheel.

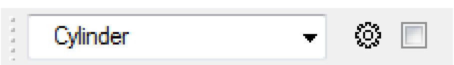


Figure 386: 3D Surfaces toolbar

Mode: Here the 3D Surface mode can be chosen. At the moment *Cylinder, STL file, Tilted Surface and Sphere* is available.

1. Cylinder

2. STL Projection

3. Tilted Surface

4. Sphere

5. Ring

6. Cone

19.1.1.1 Cylinder

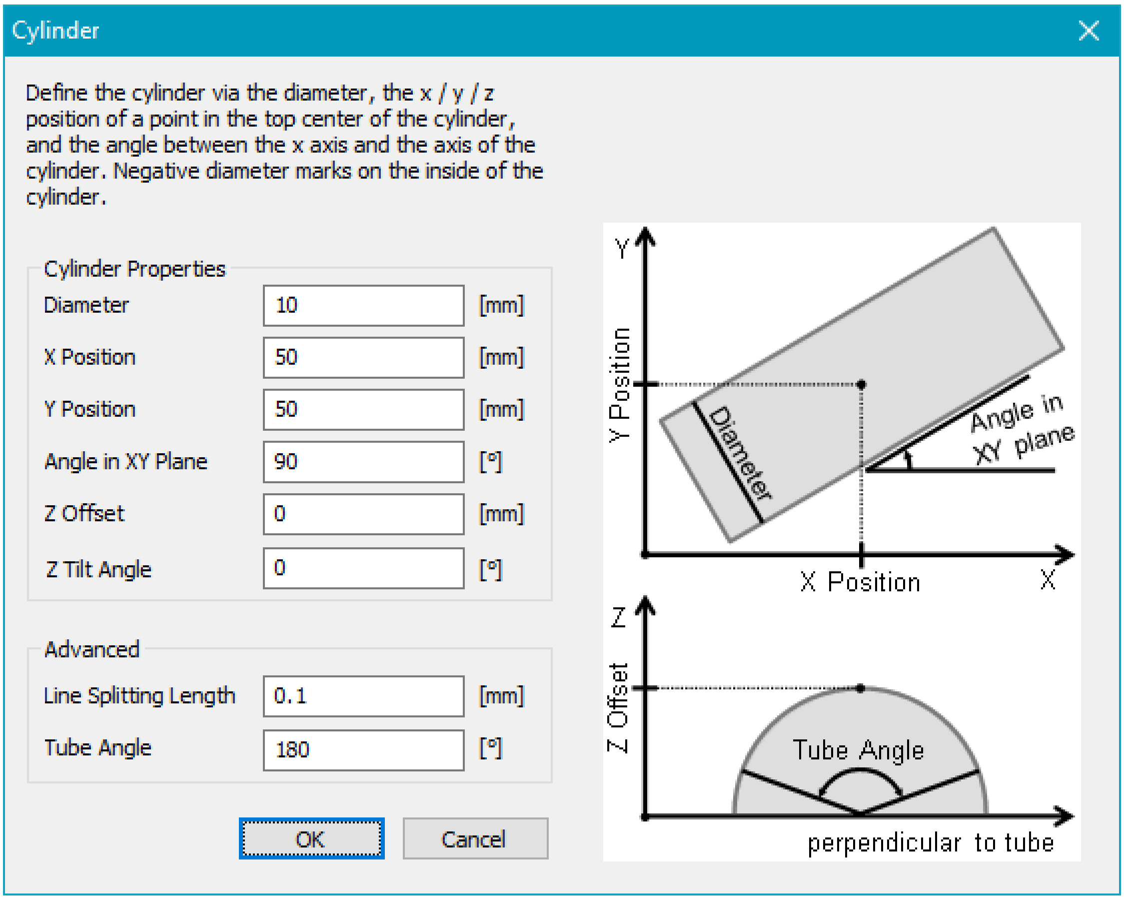


Figure 387: 3D Surfaces Cylinder dialog

Cylinder: Select Cylinder in the mode drop-down list and enable it. Click on  to open the settings dialog.

Cylinder properties: Defines the diameter and position of the cylinder.

Diameter: Defines the diameter of the tube. Negative values are also allowed to mark inside a cylinder.

X, Y Position: Defines the X, Y position of the tube.

Angle in XY Plane: Defines the tube angle in the XY plane.

Z Offset: Defines the Z position of top of the tube. If the *Diameter* is negative this value defines the Z position of bottom of the tube.

Z Tilt Angle: Defines the Z slanting angle.

Advanced: Extended settings of tube marking.

Line Splitting Length: This parameter specifies the distance along a line after which a new point is generated and transformed on the 3D surface. Every individual line is split independently.

Tube Angle: Defines the maximal angle of the tube for the vector bending. This parameter affects the distance of the blue dashed lines in the View2D.

View2D: Four blue dashed lines indicates the location and size of the cylinder in the View2D. The View2D shows the vectors not bent around the tube.

Outer blue dashed lines: Defines the unwrapped size of the *Tube Angle*. Only vectors between these lines will be bend.

Inner blue dashed lines: Defines the wrapped size of the *Tube Angle*.

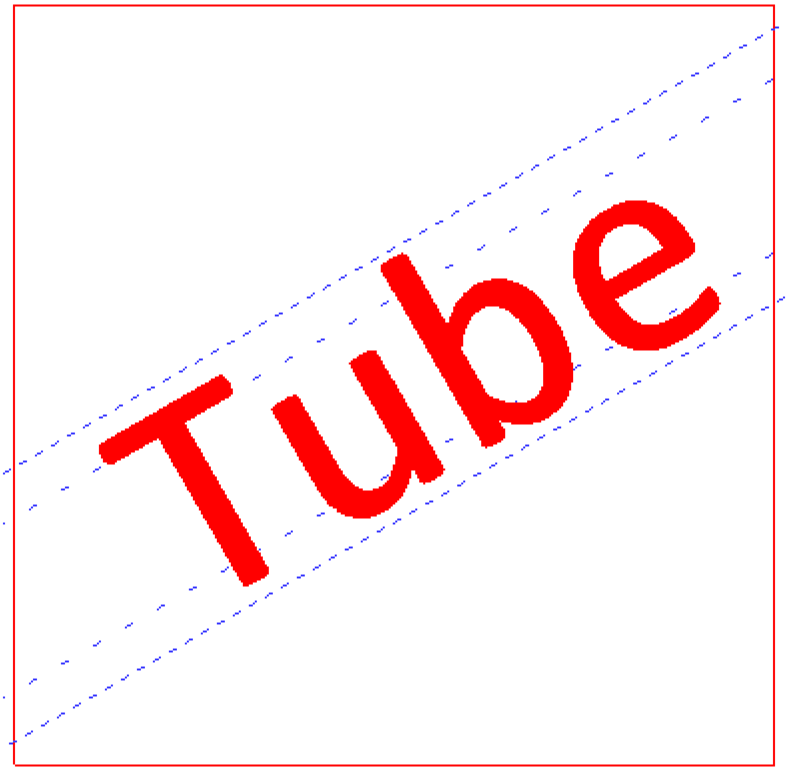


Figure 388: 3D Surfaces Cylinder View2D

3D preview: The bending of the vectors can be seen in the 3D view. Select at least one entity and click the *3D View* button  .

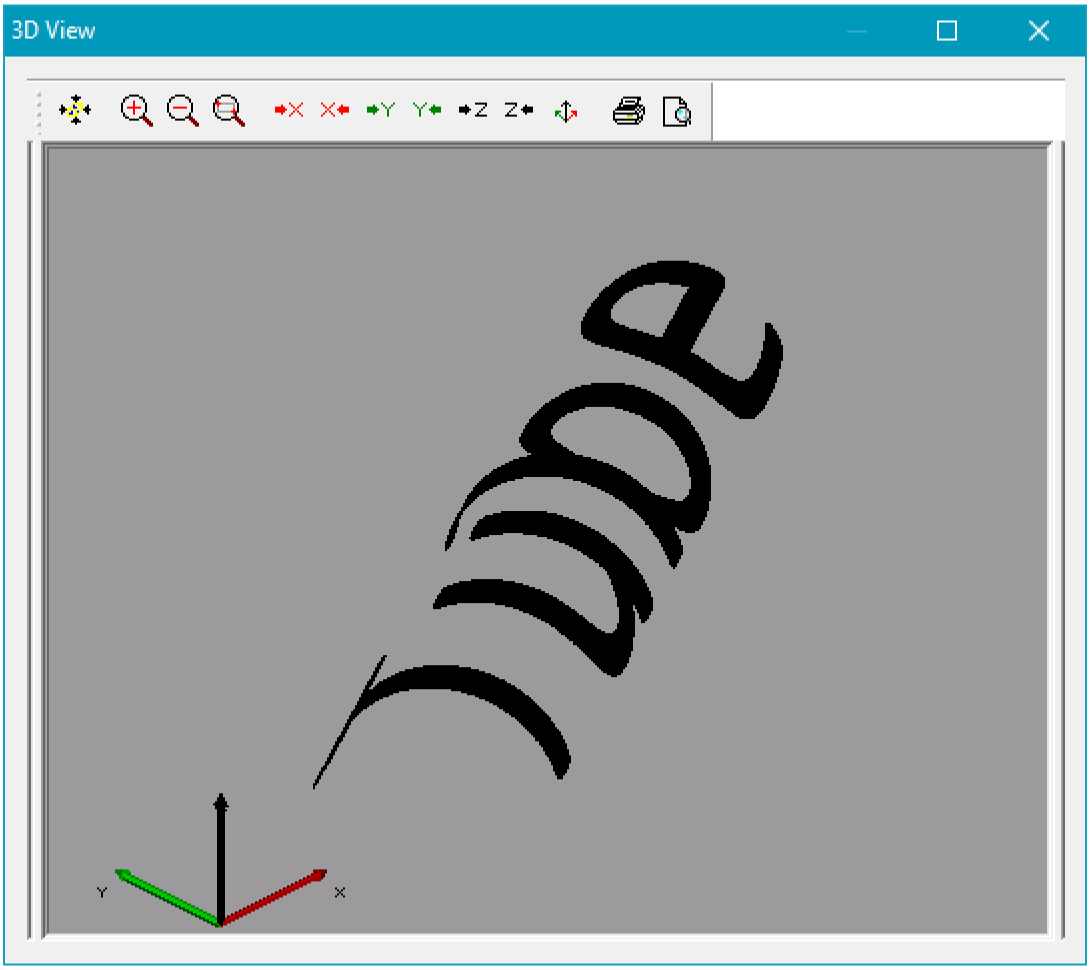


Figure 389: 3D Surfaces 3D preview.

19.1.1.2 STL Projection

NOTE

Only Black&White Bitmaps can be marked on top of a STL projection. Greyscale bitmaps are not supported.

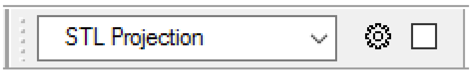


Figure 390: 3D Surfaces toolbar

Click on the cog wheel to open the properties dialog:

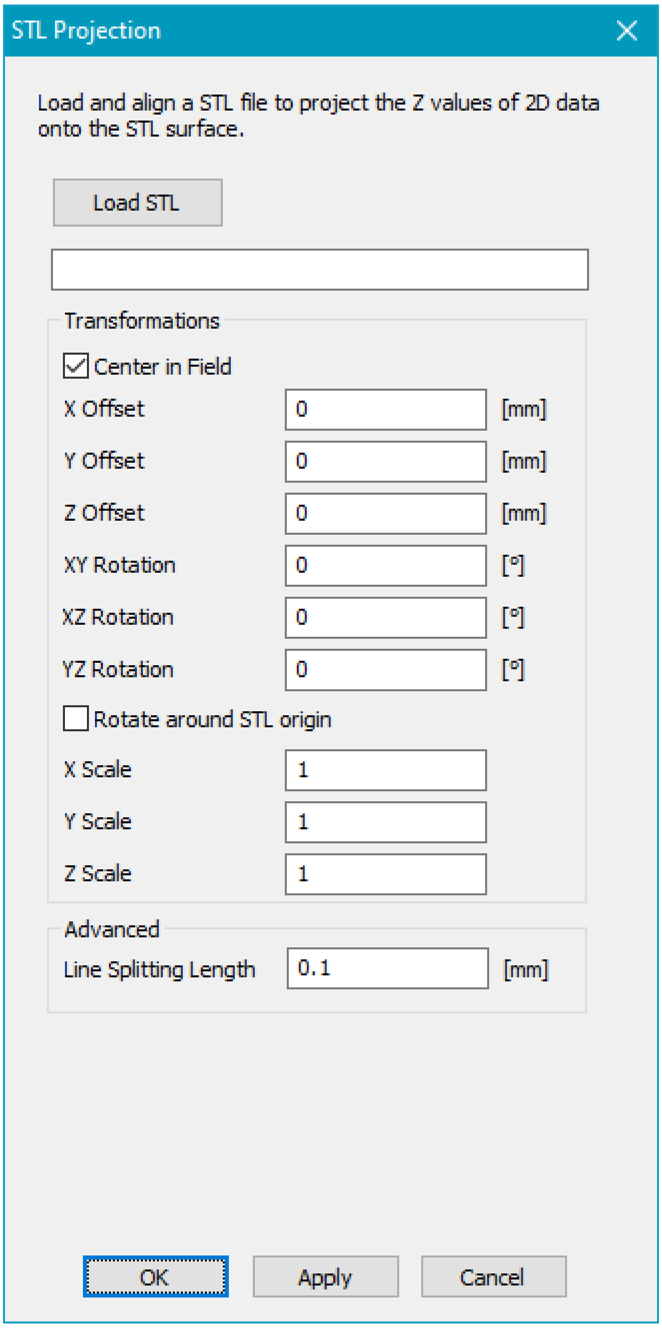


Figure 391: 3D STL property dialog

Click on "Load STL" to load a STL file. This file will be used as the surface on which the 2D vector job should be projected.

Center in Field: The STL file is centered in x and y direction in the SAMLight coordinate system. The top level of the STL file is put at z = 0 of the SAMLight coordinate system.

Offset: The STL file is translated by the given offset. This translation is done after the effect of "Center in Field".

Rotation: The coordinate system of the STL file is rotated in comparison to the SAMLight coordinate system. This rotation is done before the effect of "Center in Field".

Rotate around STL origin: If this checkbox is activated, the STL is rotated around its origin. Otherwise, it is rotated around its middle.

Scale: The STL file is scaled by the given factor. This scaling is done before the effect of "Center in Field".

Line splitting length: This parameter specifies the distance along a line after which a new point is generated and transformed on the 3D surface. Every individual line is split independently.

Enable the checkbox to the right of the cog wheel. Now the STL surface appears in the View 2D.

Below see an example where the 3D surface is a sphere and the 2D job is a simple square:

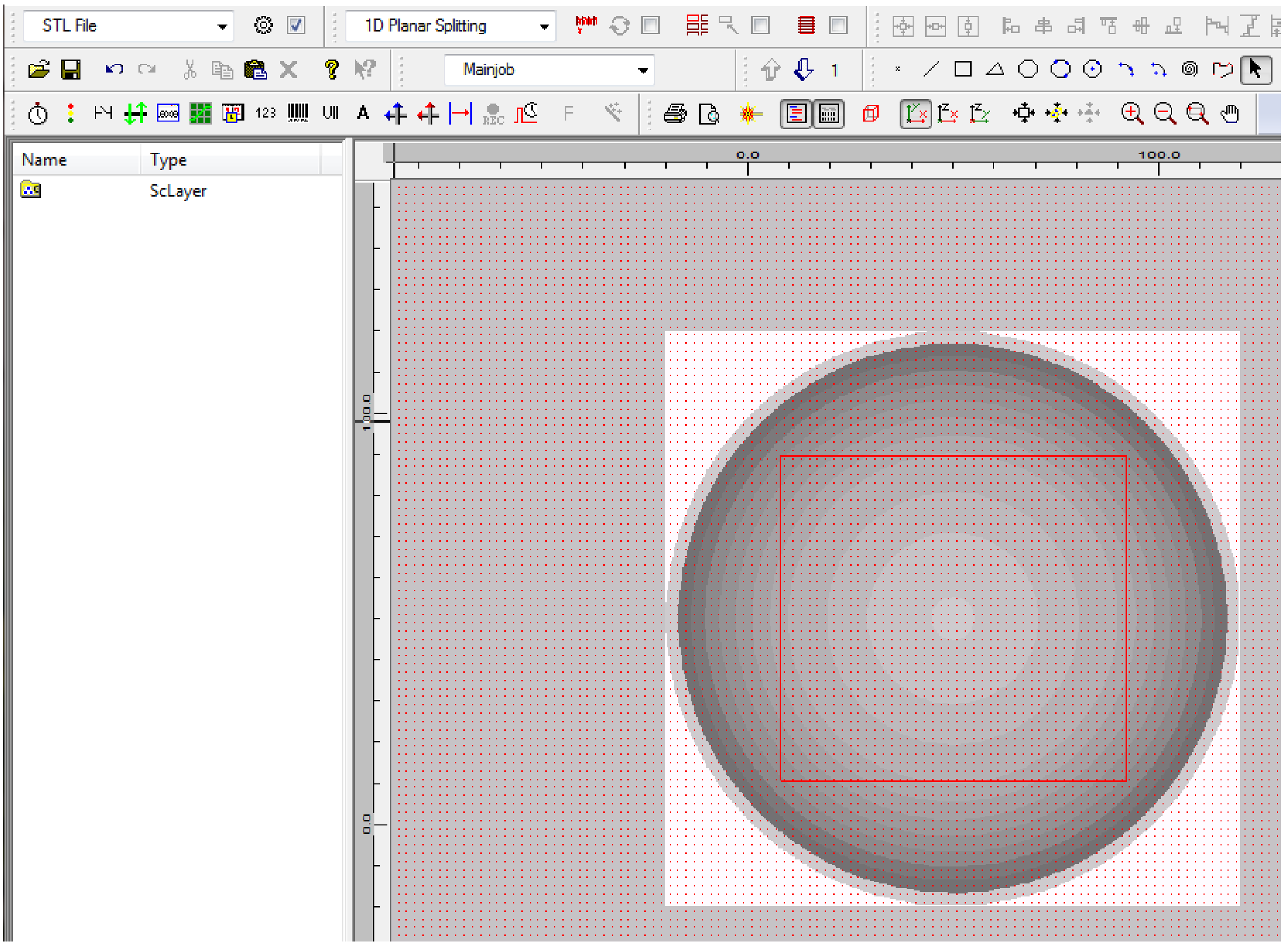
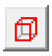


Figure 392: 2D Square projected on 3D STL sphere surface

When clicking on the Optic3D View icon  you can see how the square is projected on the sphere:

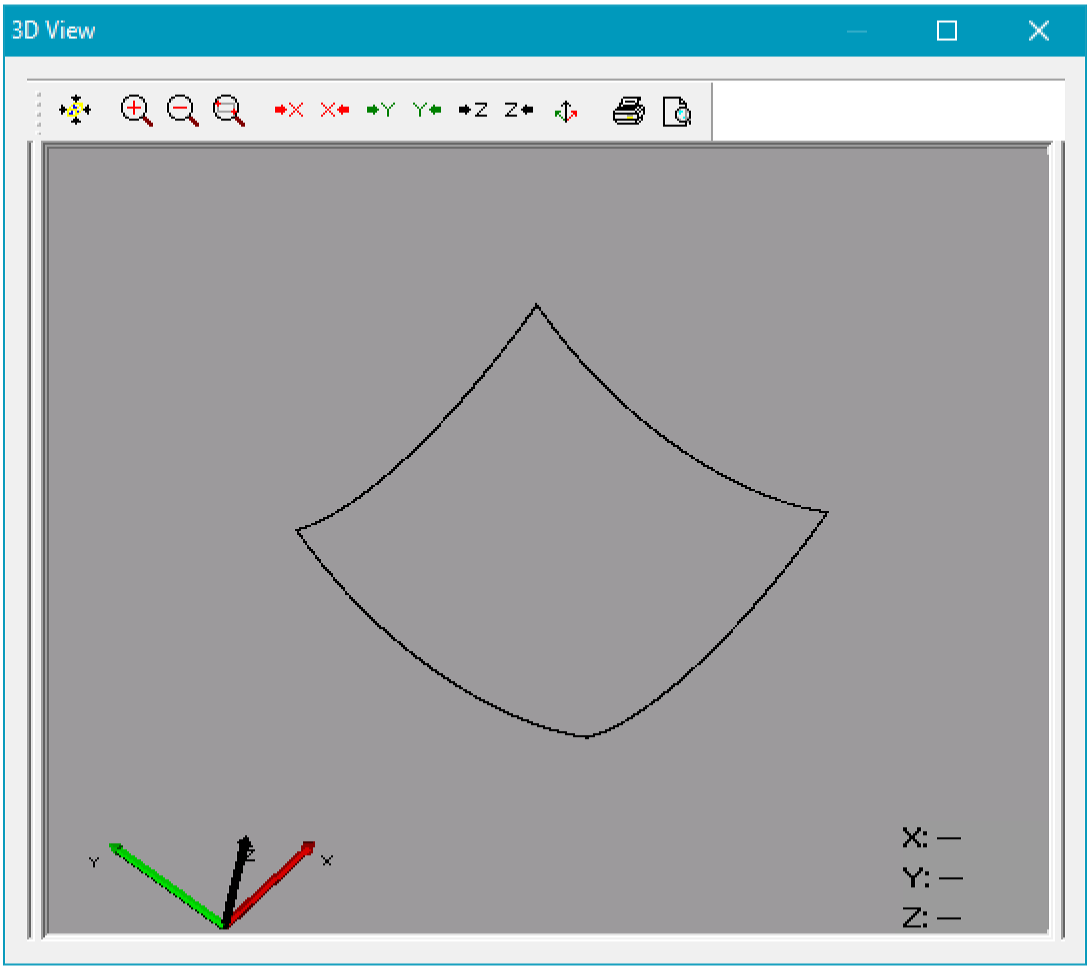


Figure 393: 3D View of the projected square.

19.1.1.3 Tilted Surface

Figure 394: 3D Surfaces toolbar

Click on the cog wheel to open the properties dialog:

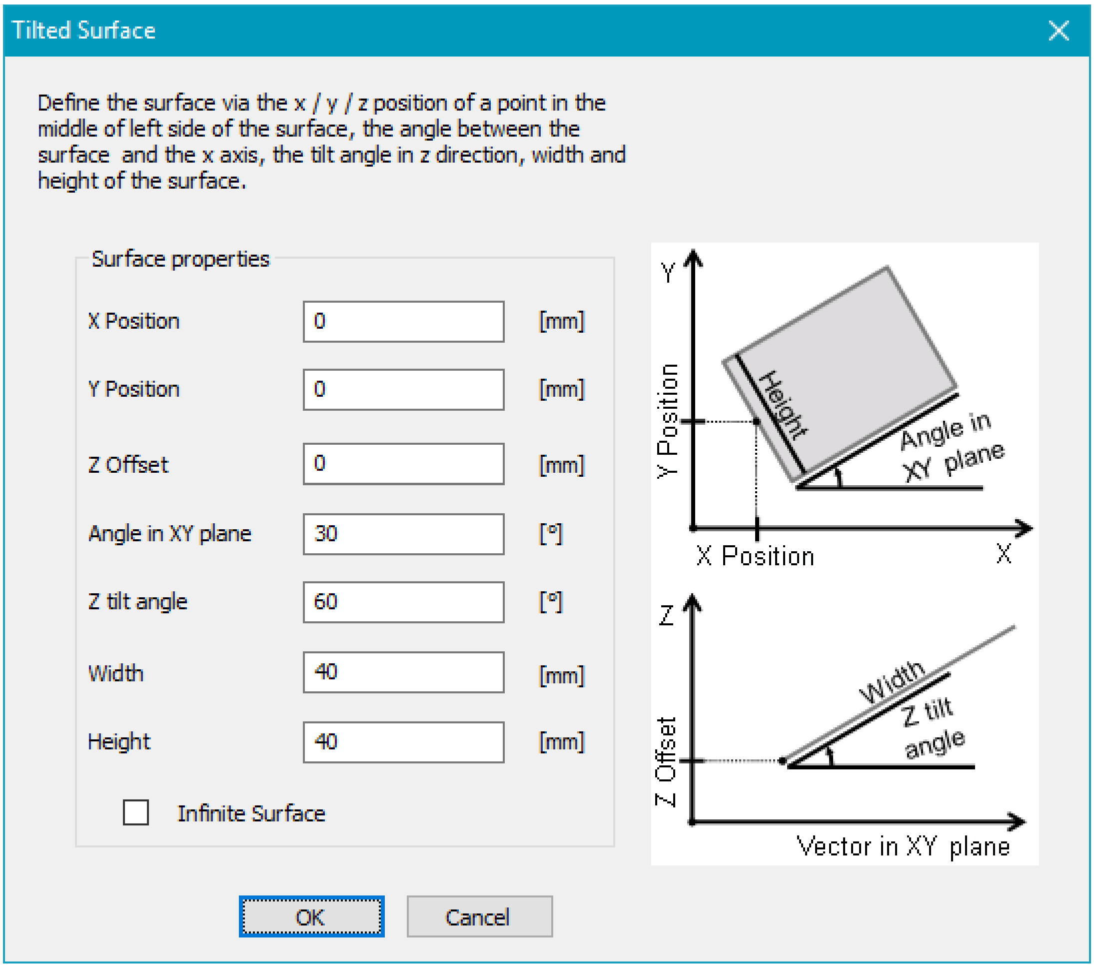


Figure 395: 3D Tilted Surface property dialog

Tilted Surface Properties

X Position: This parameter specifies the x position of the center of one side of the tilted surface.

Y Position: This parameter specifies the y position of the center of one side of the tilted surface.

Z Offset: This parameter specifies the z position of the lower border of the tilted surface.

Angle in XY plane: This parameter specifies the rotation of the tilted surface in the XY plane.

Z tilt angle: This parameter specifies the angle between the tilted surface and the XY plane.

Width: This parameter specifies the width of the tilted surface.

Height: This parameter specifies the height of the tilted surface.

Infinite Surface: Activate this checkbox to create an infinite surface.

Enable the checkbox to the right of the cog wheel to see the tilted surface in the View2D.

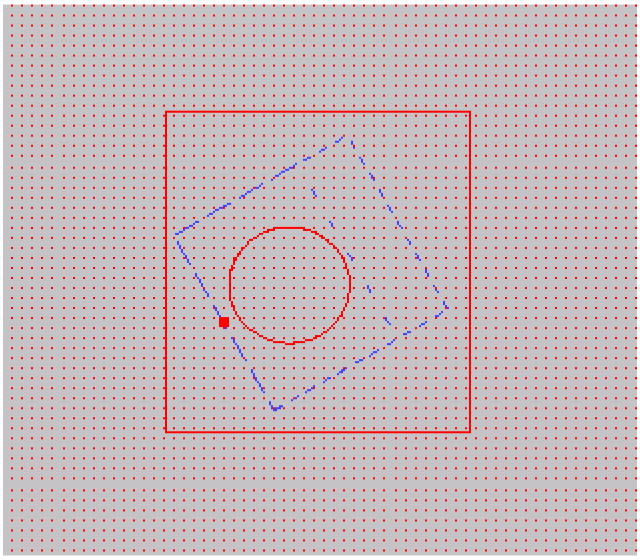


Figure 396: 3D Tilted Surface in View2D

The dashed line in the middle of the square indicates the projection of the end of the surface on the XY plane.

The result is shown in the 3D View:

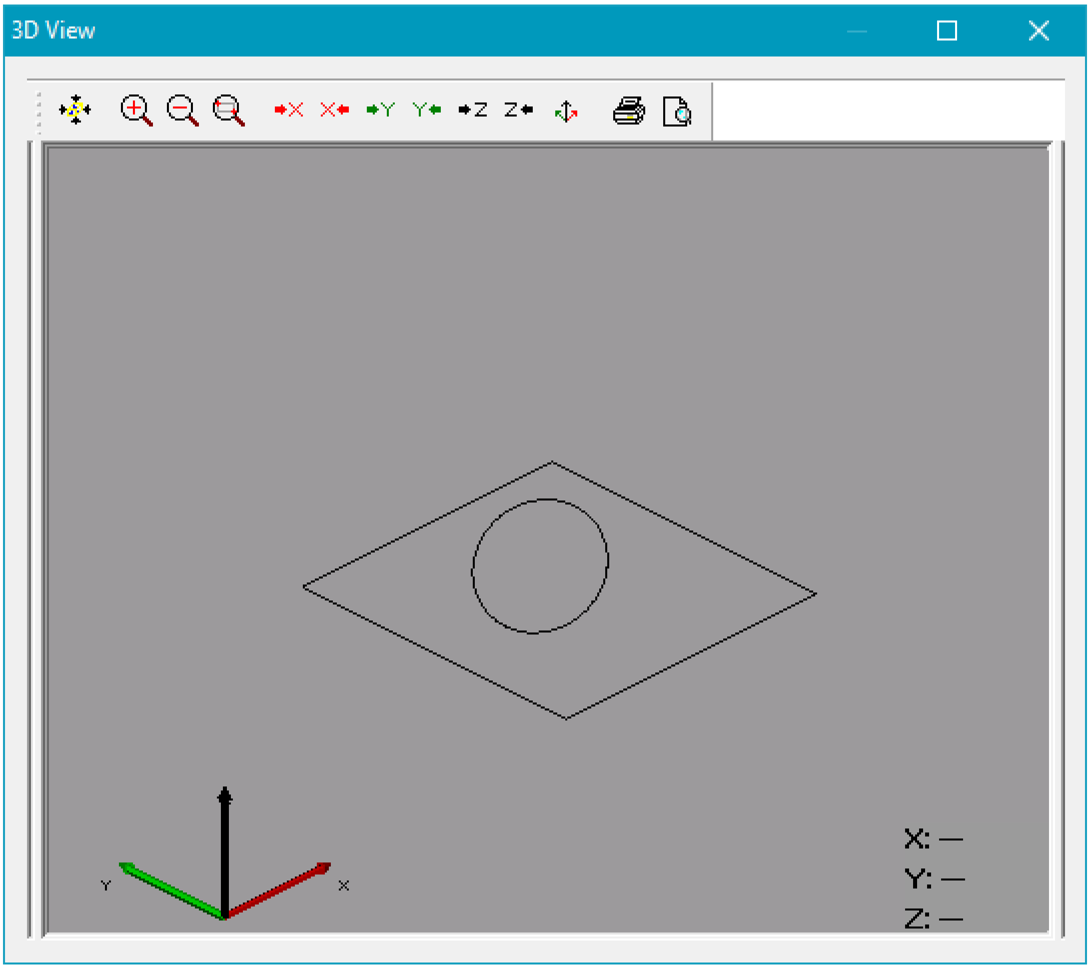


Figure 397: 3D Tilted Surface circle inside of plane XY square in View3D

19.1.1.4 Sphere

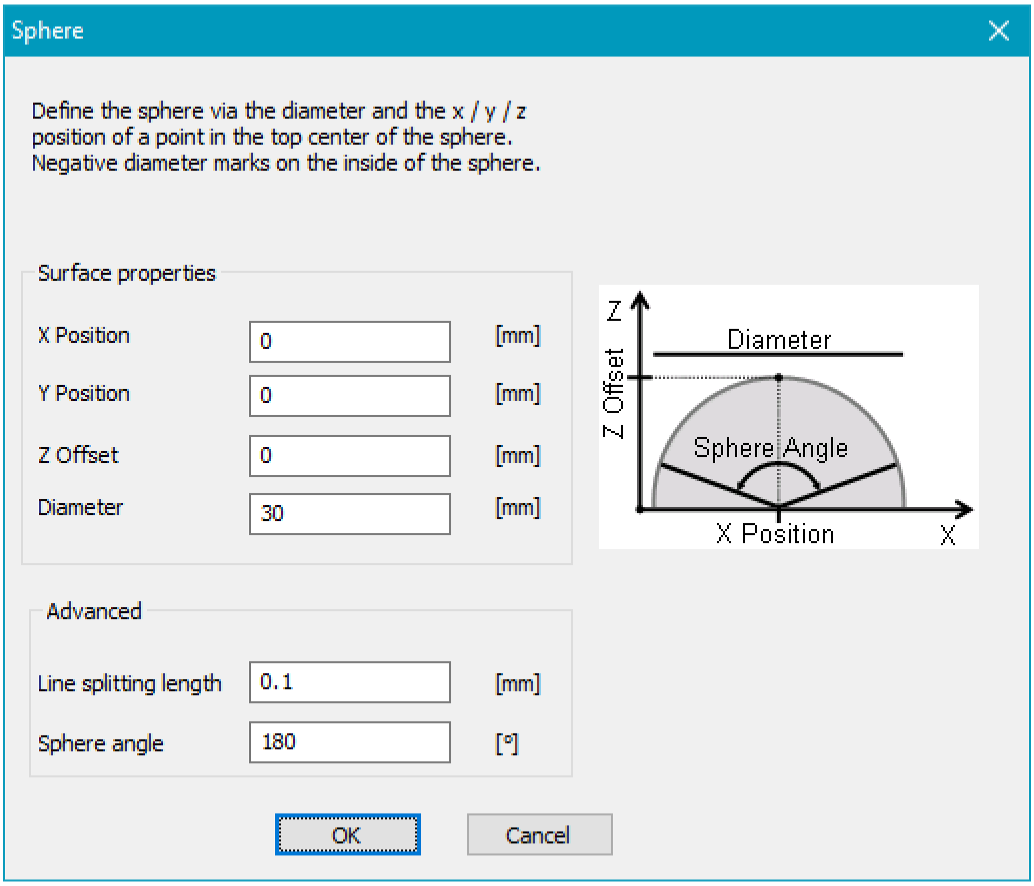


Figure 398: 3D Sphere property dialog

Sphere Surface Properties

X Position: This parameter specifies the x position of the center of the sphere.

Y Position: This parameter specifies the y position of the center of the sphere.

Z Offset: This parameter specifies the z position of the top of the sphere.

Diameter: This parameter specifies the diameter of the sphere.

Advanced

Line Splitting Length: This parameter specifies the distance along a line after which a new point is generated and transformed on the 3D surface. Every individual line is split independently.

Sphere Angle: This parameter specifies the opening angle of the sphere.

Enable the checkbox to the right of the cog wheel to see the sphere in the View2D.

The inner circle represents the diameter of the sphere.

The outer circle represents the full available curved shape of the sphere in 2D as available marking area. The diameter of the outer circle is given by r = diameter * pi/2.

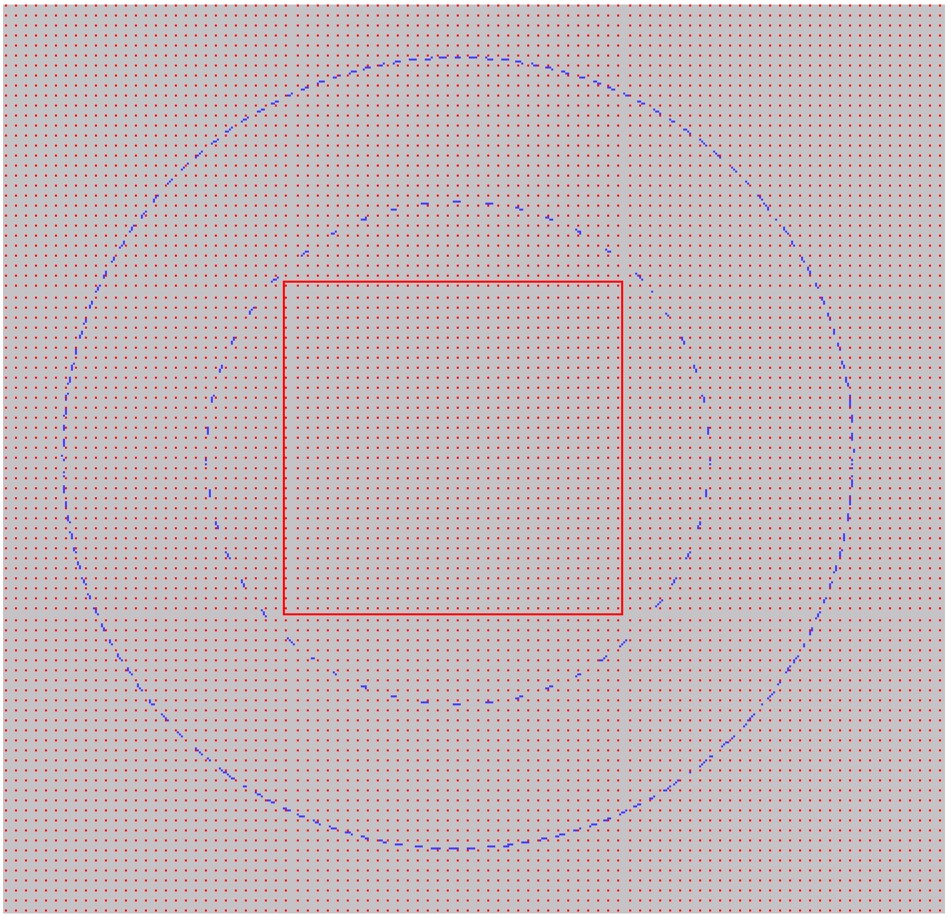


Figure 399: 3D Sphere in View2D

19.1.1.5 Ring

Figure 400: 3D Surfaces toolbar

Click on the cog wheel to open the properties dialog:

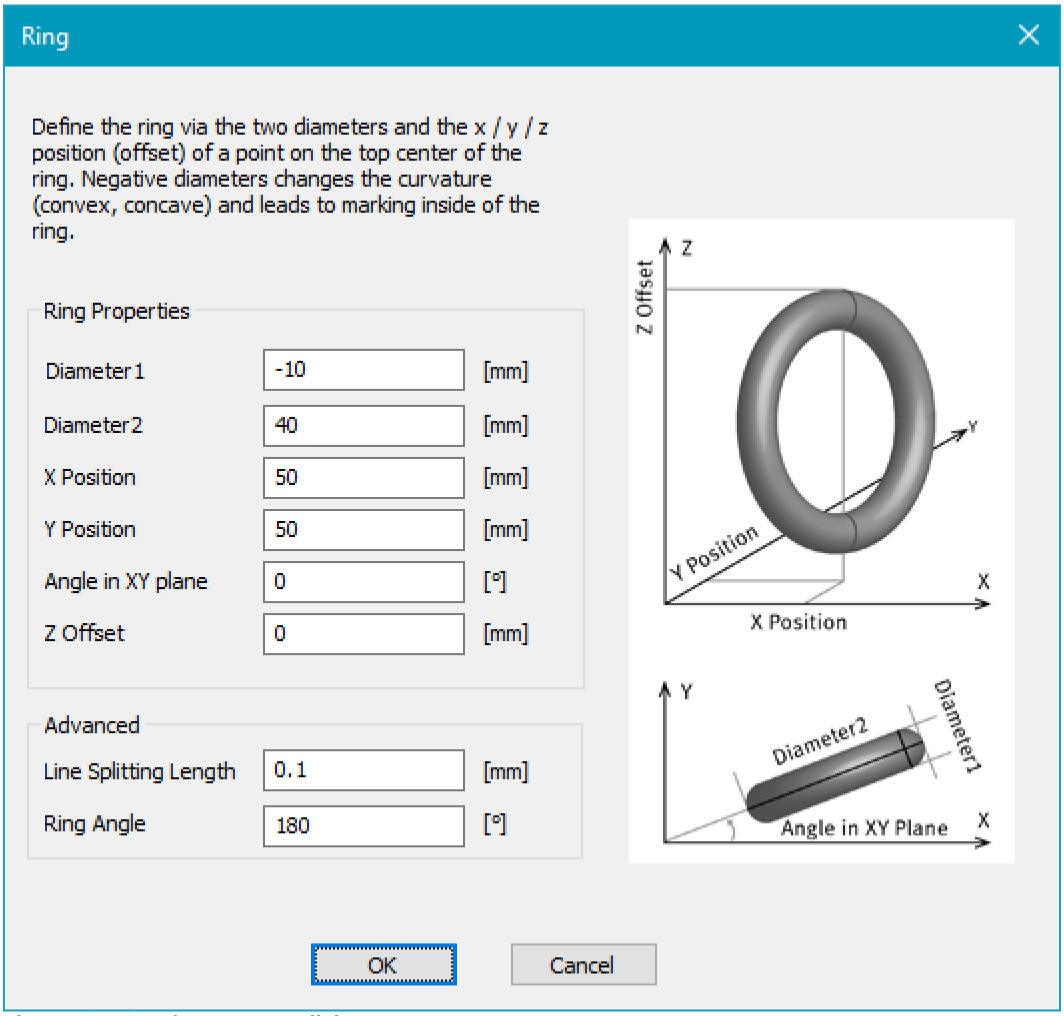


Figure 401: 3D Ring property dialog

Ring Properties

Diameter1: This parameter specifies the width and height of the ring.

Diameter2: This parameter specifies the total size of the ring.

X Position: This parameter specifies the x position of the center of the ring.

Y Position: This parameter specifies the y position of the center of the ring.

Angle in XY plane: This parameter specifies the rotation of the ring in the XY plane.

Z Offset: This parameter specifies the z position of the top of the ring.

Advanced

Line Splitting Length: This parameter specifies the distance along a line after which a new point is generated and transformed on the 3D surface. Every individual line is split independently.

Ring Angle: This parameter specifies the opening angle in the direction of diameter 1 and diameter 2.

Enable the checkbox to the right of the cog wheel to see the ring in the View2D.

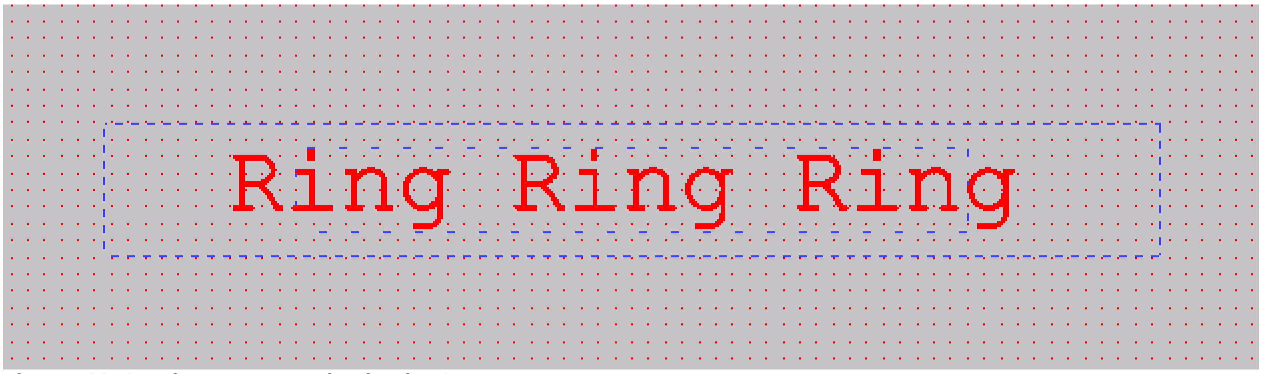


Figure 402: 3D Ring representation in View2D

When clicking on the Optic3D View icon , you can see how the entity is projected on the ring:

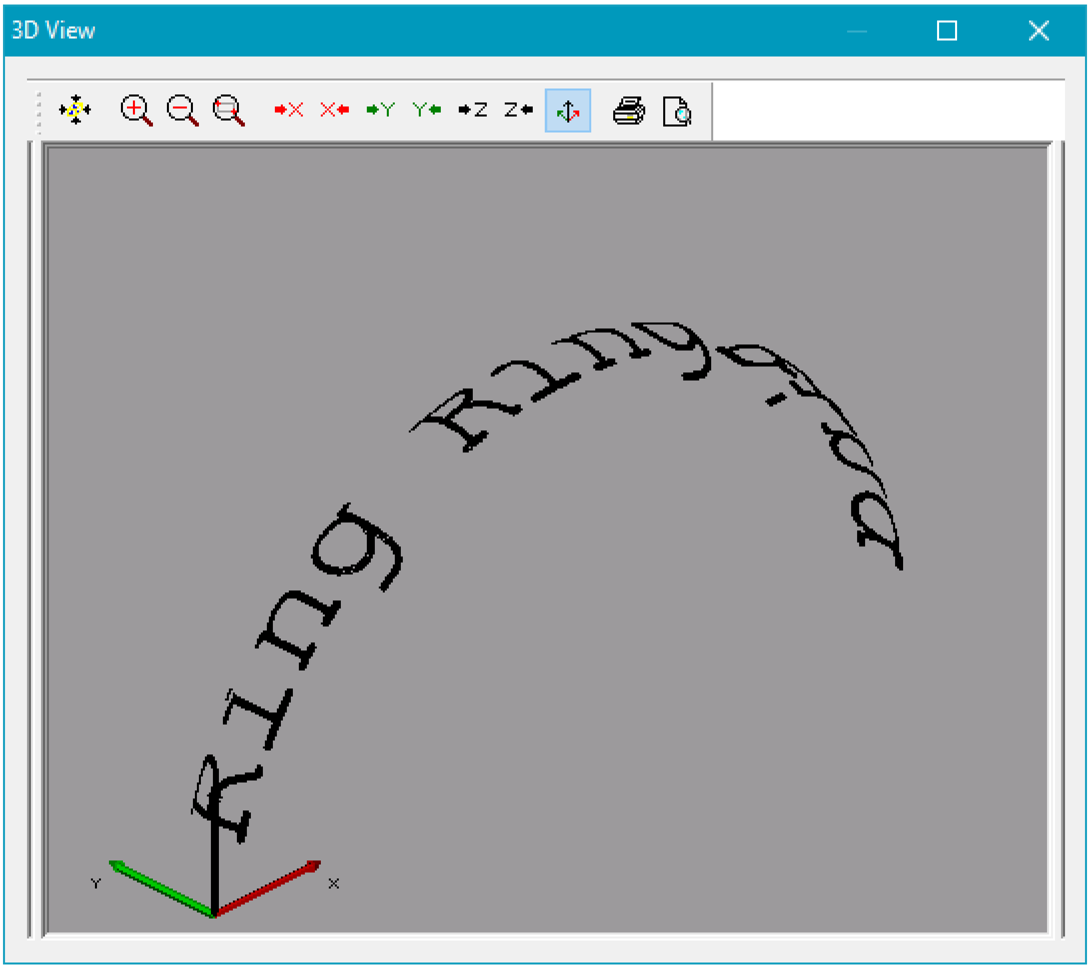
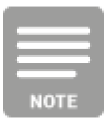


Figure 403: 3D View of the projected entity on the ring.

19.1.1.6 Cone



Only available for USC-3!

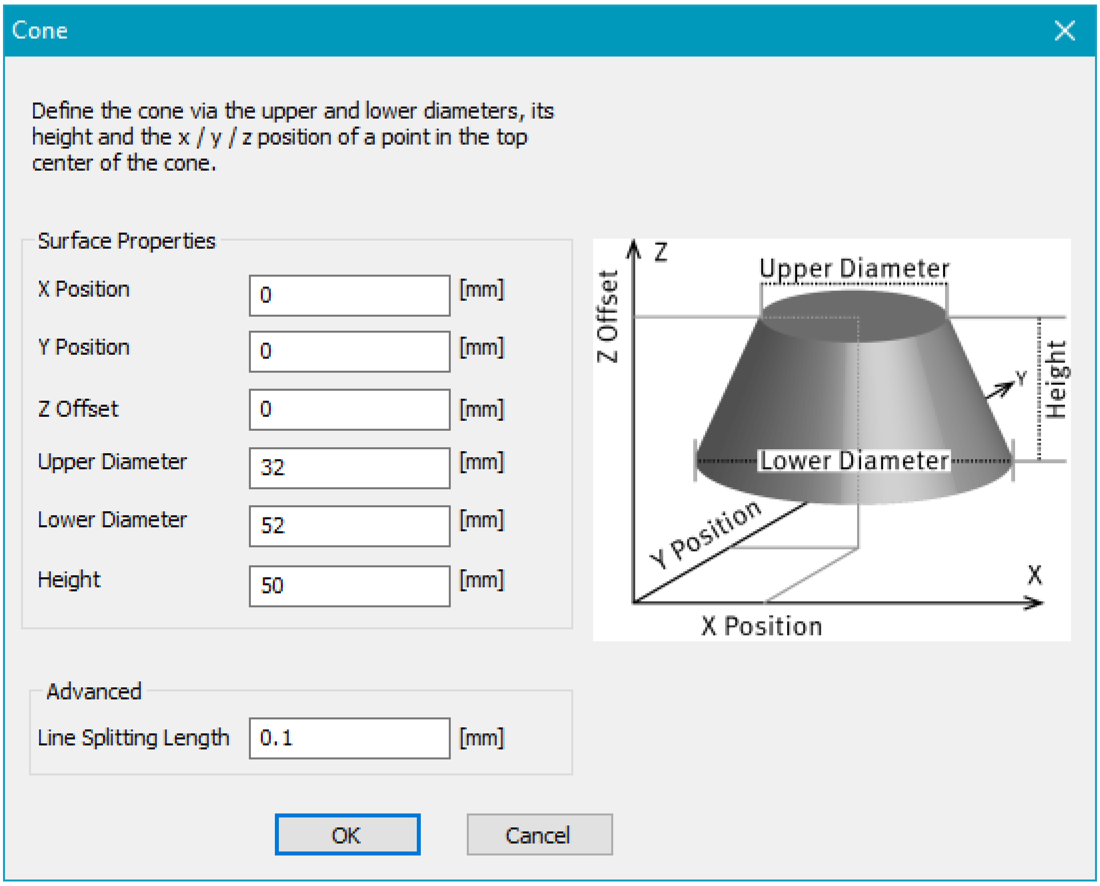


Figure 404: 3D Surfaces Cone Dialog

Surface Properties:

X / Y / Z Position/ Offset: Defines the central point on the upper surface.

Upper diameter: Defines the diameter of the upper surface.

Lower diameter: Defines the diameter of the lower surface.

Height: Defines the height of the cone.

Advanced:

Line Splitting Length: This parameter specifies the distance along a line after which a new point is generated and transformed on the 3D surface. Every individual line is split independently.

19.1.2 Marking on curved parts

With Optic3D 3D DXF files can be imported. A translation of objects in z-direction, their rotation around all axis as well as a manual manipulation of 3D-vectors are additionally possible. The marking of 3D-line structures is only possible with special hardware (suitable scanner controller card / 3-axis scan head). For this a special correction file is needed, see Requirement & Settings. The range of the z-direction depends on the stroke of the scanner. The three axis scanner is used for marking on curved parts, means the scanner can mark lines with varying Z-values. 3D vectors can be imported (*File → Import...*) by a 3D DXF file (*DXF Files Version 2*) and the Z-values of lines / points can be edited.

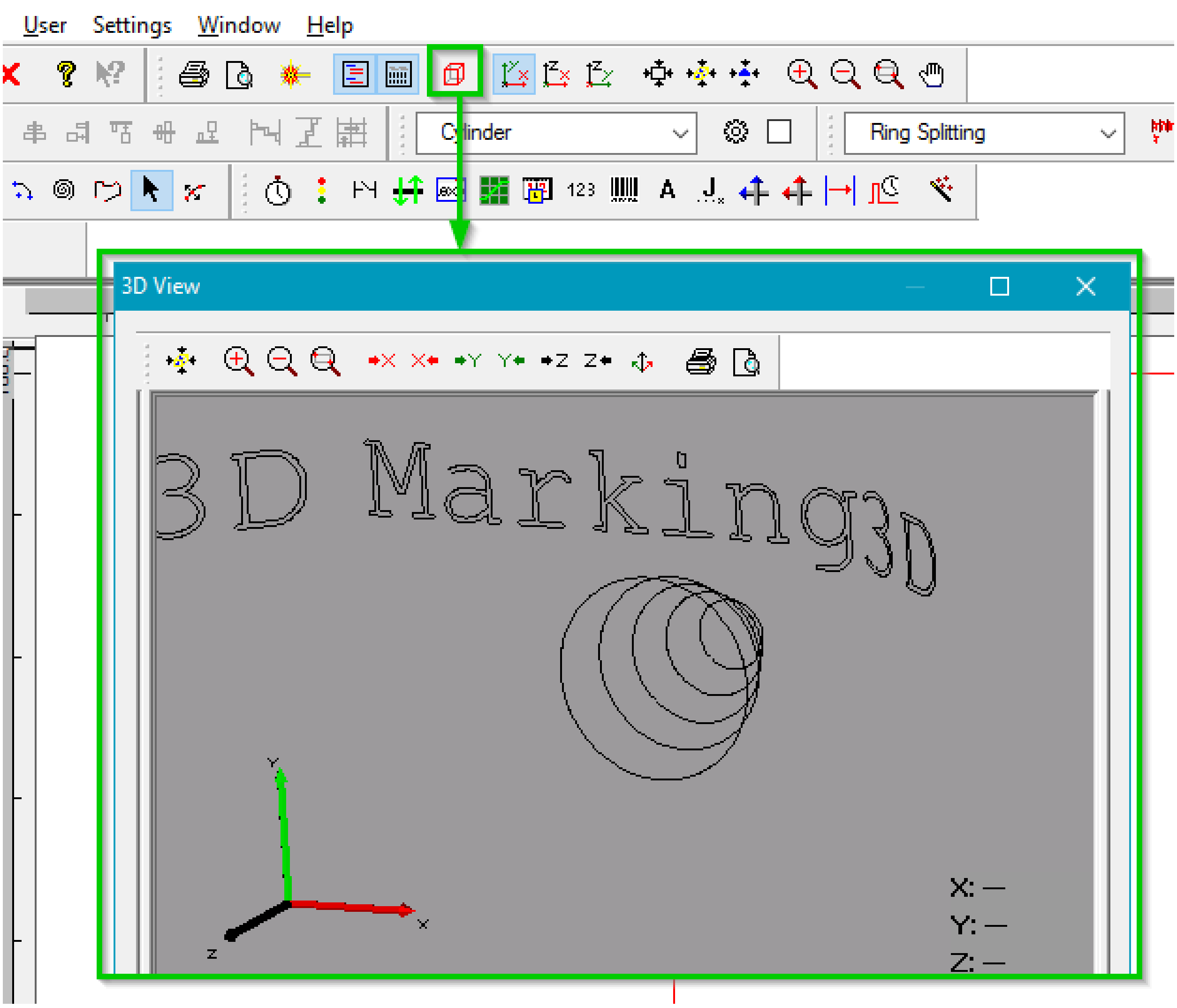
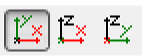


Figure 405: Example of 3D object in optic 3D view

In the 3D View (*cube* button) you can have a closer look at your 3D object by rotating the point of view.

 **3D view buttons:** With these buttons you can view the 3D object in different planes.

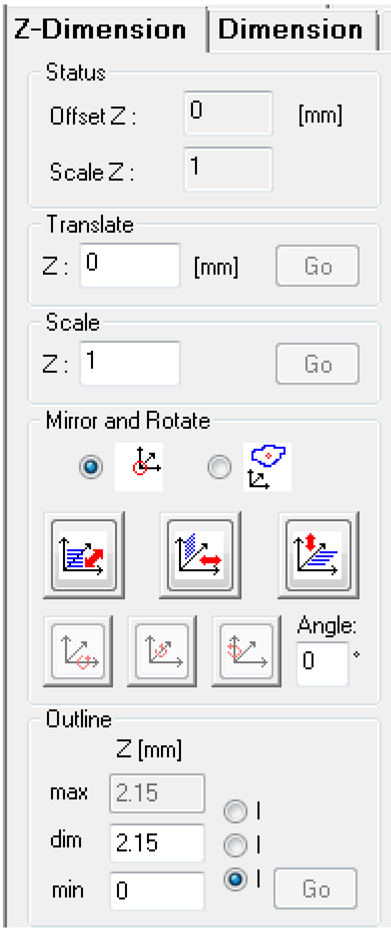


Figure 406: Z-Dimension property page

In the Z-Dimension property page you can translate and scale objects in the z direction and perform mirror and rotation operations.

Status:

Offset Z: Difference of Z value between the original object and the object that has evolved due to transformations.

Scale Z: Difference of Z scaling between the original object and the object that has evolved due to transformations.

Translate:

Z: Translate whole object in Z-direction.

Scale:

Z: Scale whole object in Z-direction.

Mirror and Rotate: Here you can choose the center of rotation.

Mirroring: Mirrors the object on the X, Y or Z plane.

Rotate around X, Y or Z axis: Rotate a specific angle around the X, Y or Z axis.

Outline:

max: Maximum z-Coordinate of the object.

dim: Z-dimension of the object.

min: Minimum Z-Coordinate of the object.

By pressing the *point edit mode* button in the toolbar, you can edit single points of the 3D object.

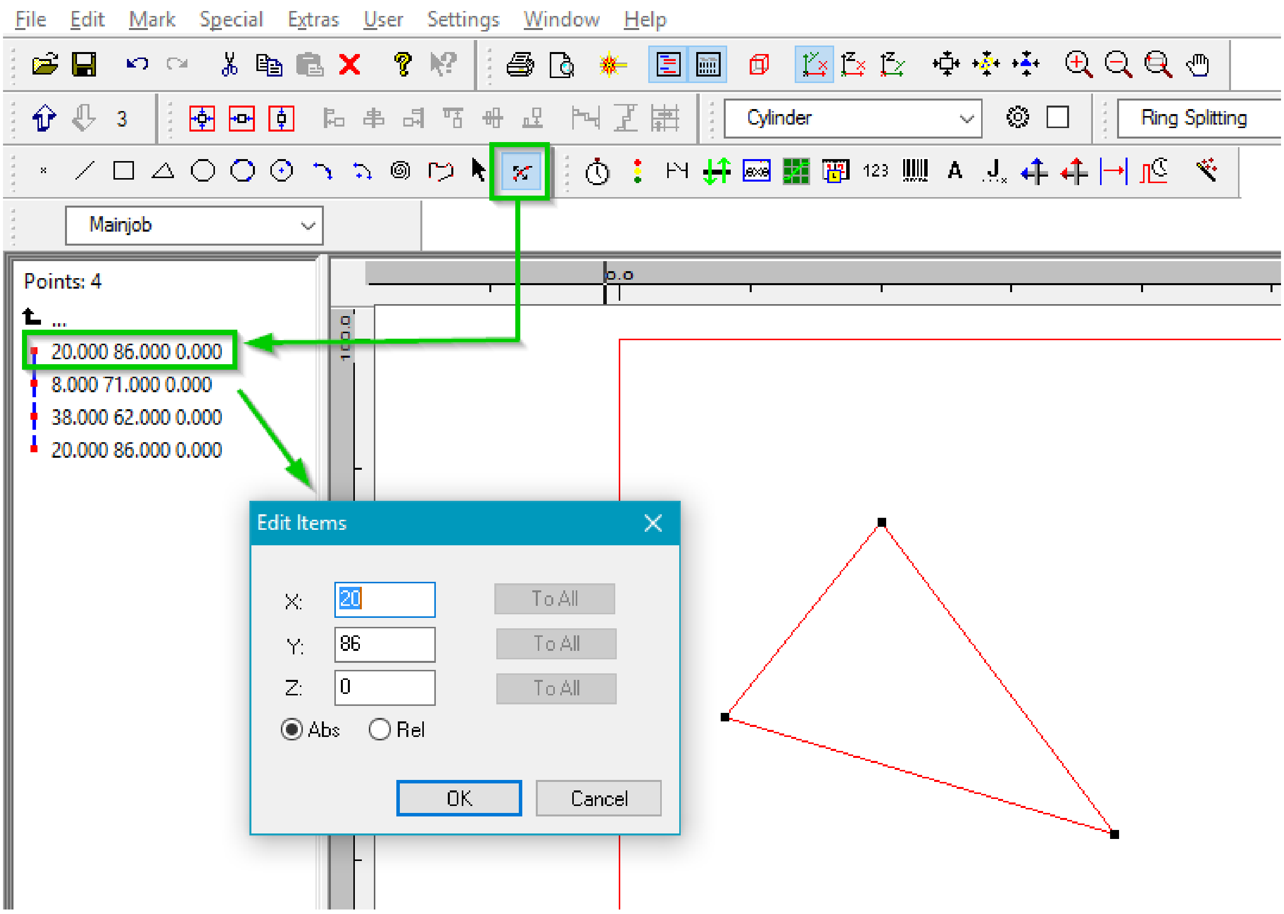


Figure 407: Optic3D point edit mode

19.1.3 Deep Engraving

This feature is used when the object in the editor should be marked several times at different Z-heights, e.g. to do deep engraving on a material. The dialog can be accessed via the menu *Mark → Z-Settings.*

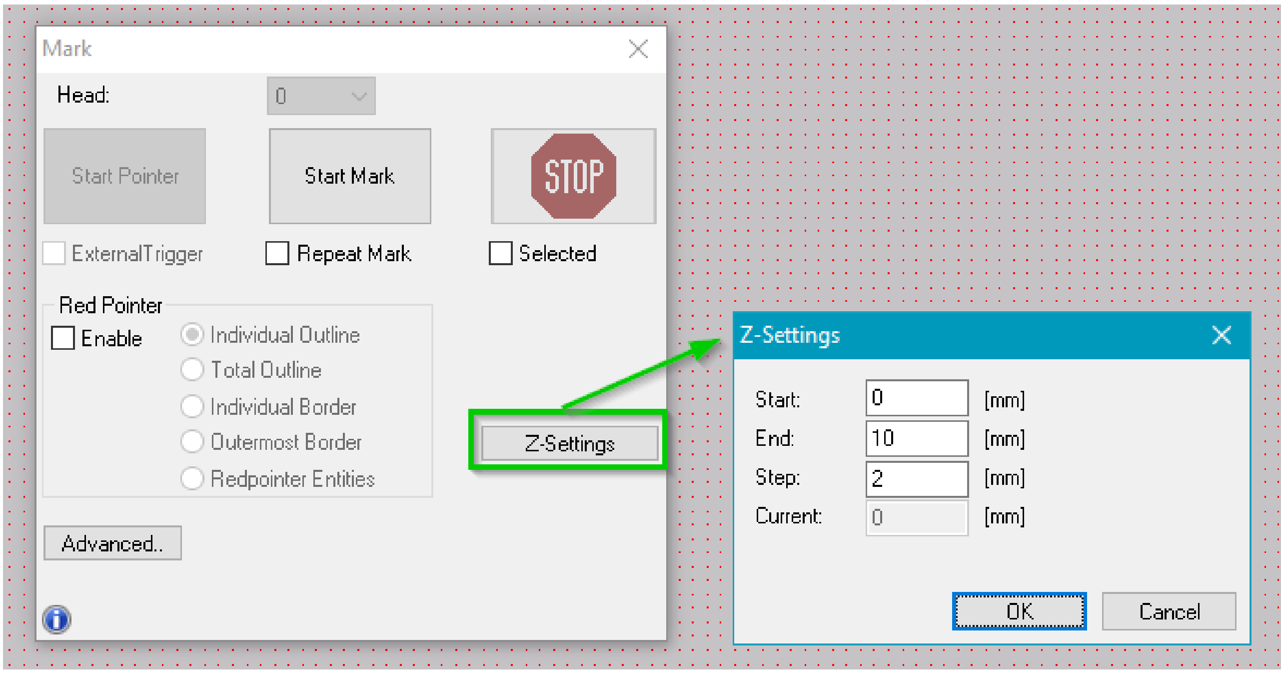


Figure 408: Deep Engraving Dialog

This is how this feature works:

1. Loop: Z = Start
2. Loop: Z = Start + Step*1
3. Loop: Z = Start + Step*2
...
i. Loop: Z = Start + Step*(i-1)
...
n. Loop: Z = End - Step

After last marking the Depth is equal to End, because the last marking is at End - Step and erodes the Step value. The Z-Settings are being stored within the job file.

19.2 Requirements & Settings

In the following the requirements and settings for the different scanner controller cards are described.

To calibrate a 3D system please find further information in the Chapter 3D Calibration.

19.2.1 SCAPS USC cards

Requirements for SCAPS USC-1, USC-2 and USC-3 cards:

- Correction File: 2D ucf file from SCAPS.
- Required SCAPS option: FlatLense and Optic3D.

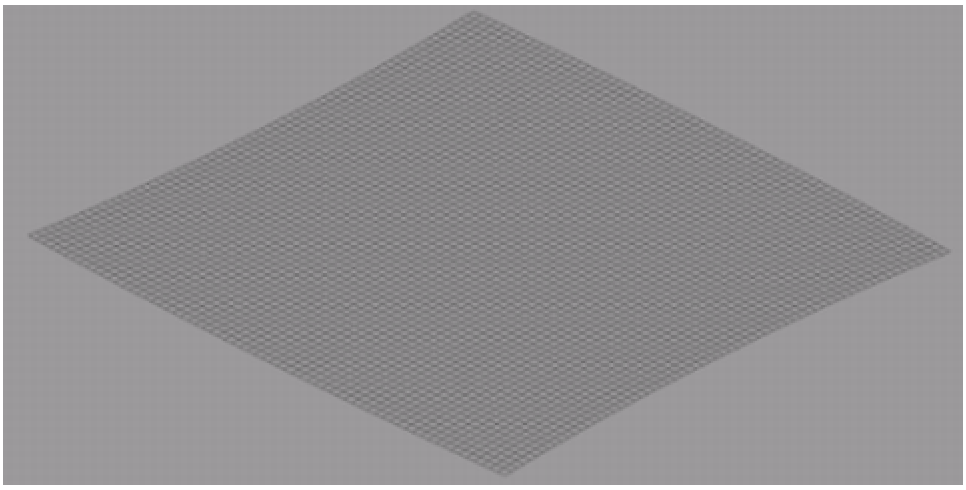


Figure 409: Shape of a 2D Correction File

Settings: For USC cards, enter the following Dialogs *Settings → System → Optic*:

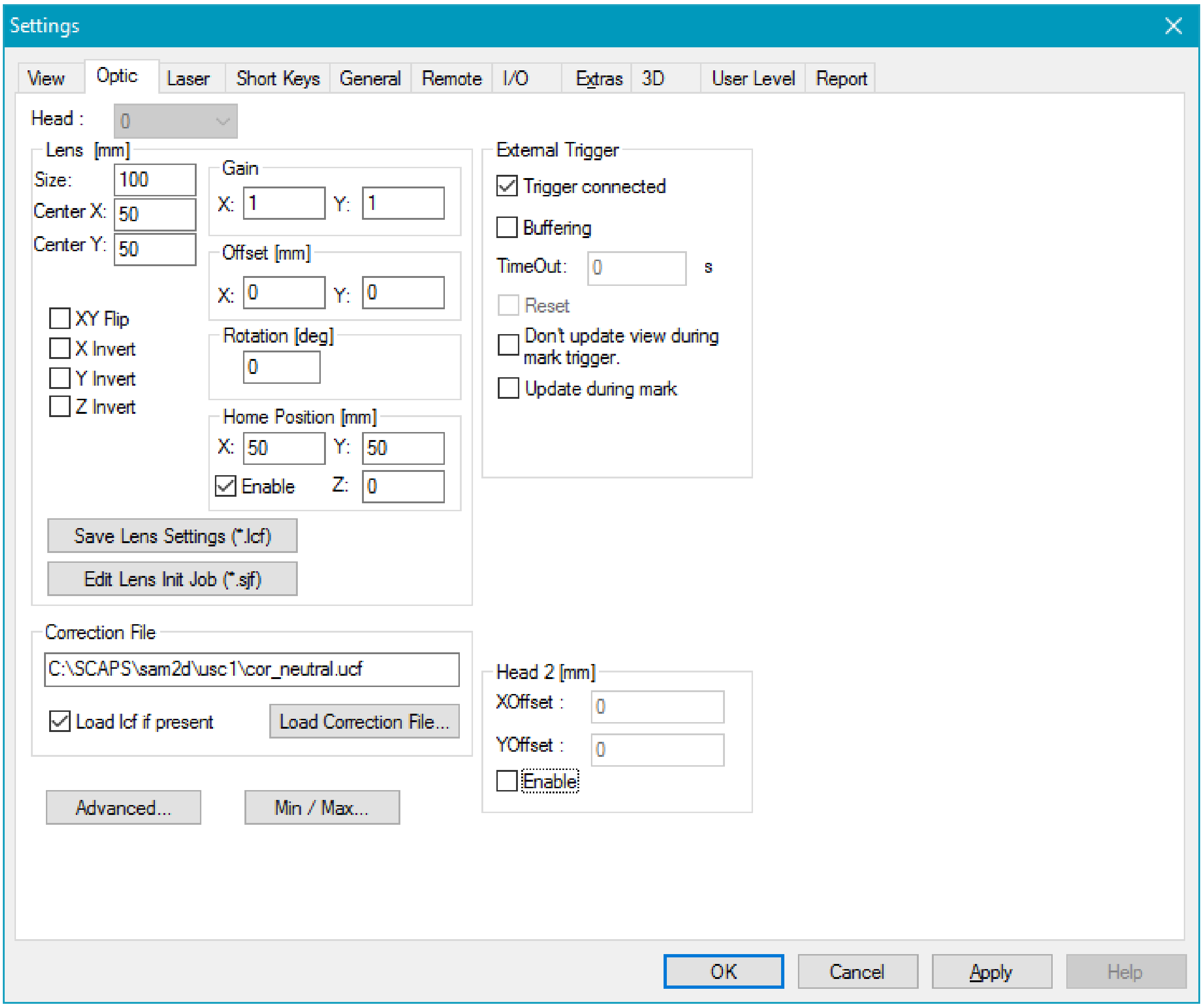


Figure 410: Settings → Optic Dialog for Z-Dimension

In this dialog the XY field size and the correction file can be defined.

Find the following Dialog for USC-1 in *Settings → System → Optic → Advanced → Z-Correction (Enable) → Advanced* and for USC-2 in *Settings → System → Optic → Advanced → Correction, Settings → Z-Correction, Settings.*

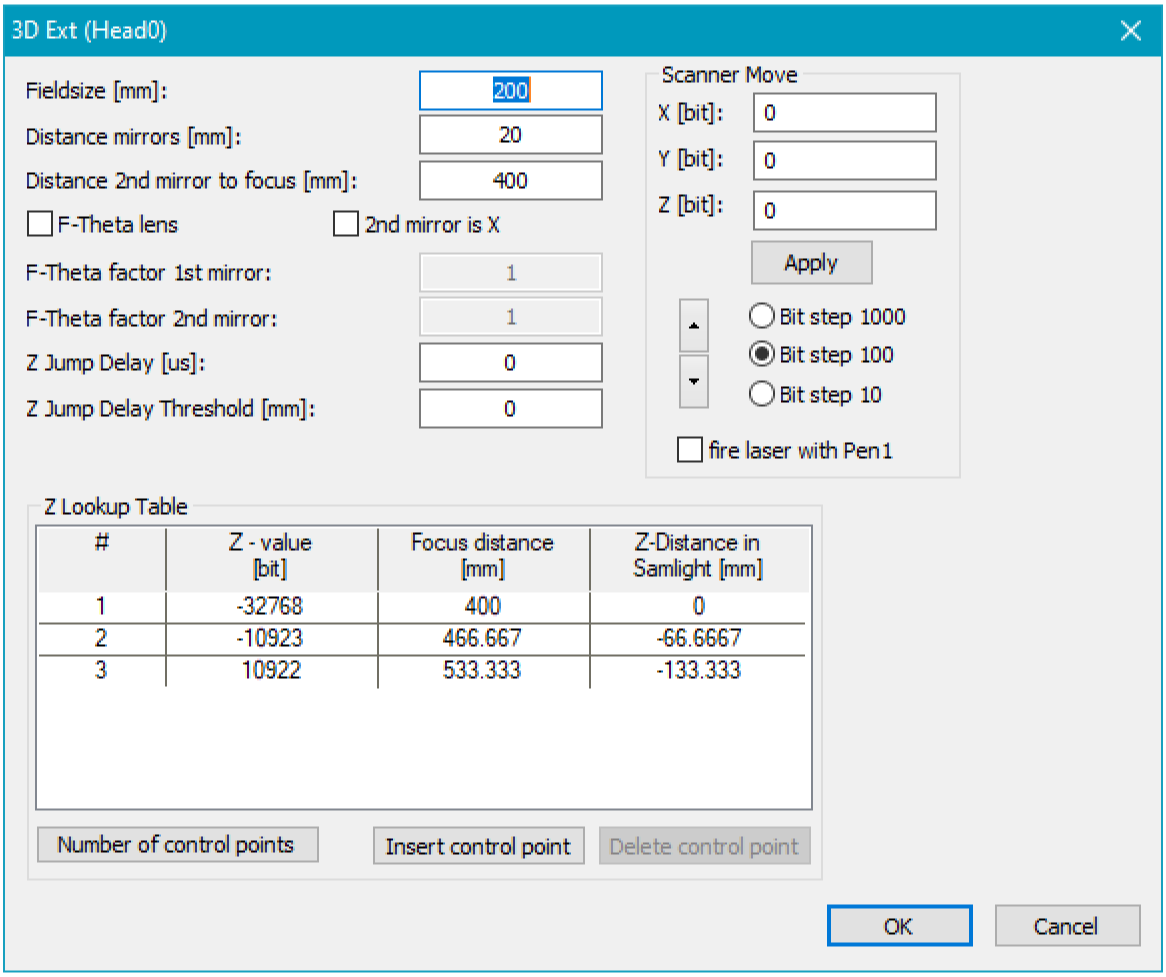


Figure 411: Z-Correction Settings

Fieldsize: The size of the XY field needs to be identical with the field size set in Settings > System > Optic.

Distance mirrors: Distance between the x and y mirror of the scan head in mm. This value is usually given by the scan head manufacturer.

Distance second mirror to focus: Distance between the second mirror to the center of the field in the focus plane.

F-Theta lens: Should be checked if in addition to the 3rd scanner axis, a F-Theta lens is mounted.

2nd mirror is X: Switches x and y values.

F-Theta factor 1st mirror/ F-Theta factor 2nd mirror: These empirical factors can be found following the description in F-theta Factor Calibration.

Z Jump Delay [µs]: Due to the fundamental difference of the Z axis it could be necessary to increase the jump delay after a big change in the Z value. The *'Z Jump Delay'* is added to the normal jump delay when a jump has a Z dimension greater than *'Z Jump Delay Threshold [mm]'*. A value of *'0'* disables this feature.

Z Jump Delay Threshold [mm]: If a jump has a Z dimension greater than this threshold the 'Z Jump Delay [µs]' is added to the normal jump delay.

Z Lookup Table: For compensating the non-linearity of the relation z axis position to focus distance, a lookup table can be defined. The lookup table contains the DAC value plus the distance from second mirror to focus when adjusting this DAC value. The points must be ordered ascending by the 'Focus Distance [mm]' value.

Number of control points: The number of control points can be between 2 and 32.

Insert control point: Insert a new control point. If no line is selected the control point will be added at the bottom, if a line is selected the point will be added above.

Delete control point: Select a control point and delete it.

XYZ Move: A helper function for finding the proper lookup table values.

High / Normal / Fine: Define Step Width for XYZ Move.

Recommended calibration procedure:

- Get the correction file from scan head manufacturer and send it to SCAPS (info@scaps.com) for being converted into a UCF file format containing no z values
- Setup hardware to right working distance (z=0 plane).
- Find the right lens size value to get a correct aspect ratio when marking for example a rectangle. 10 mm in drawing have to be 10 mm on marking (in z=0 plane). The lens value defines how many mm are related to the 65536 bits.
- Get the right mirror mounting distance values.
- Within the z calibration dialog, type in values for field size and the 2 distance mirror values.
- For getting the right calibration values for the lookup table
- Best would be having a z stage for calibration of different z heights. The valid DAC range is from -32768 to +32767. Two general ways to generate the lookup table. Either predefining DAC and varying z or the other way around.
- E.g. you create 3 control points and enter following values for DAC.
- Then you search for the focal planes d1, d2 and d3 with the XYZ Move window. For compensating further non-linearities you have to create further control points in the lookup table.

	DAC [bits]	[mm]
1	-32768	d1
2	0	d2
3	32767	d3

Table 51: Example Z lookup table. d1 < d2 < d3

19.2.2 SCANLAB RTC cards

Requirements for SCANLAB RTC3, RTC4 and RTC5 cards:

- Required SCAPS option:
  - o **Optic3D**
- Required SCANLAB hardware:
  - o RTC board
  - o RTC option "Controlling the Third Axis of a 3-Axis System"
  - o RTC option "Second Scan Head Connector" (if Z channel is wired to 2nd scan head connector)
- Required software:
  - o **32bit 3D** drivers (for example: RTC4D3.hex)
  - o Correction file: **3D** CTB (or CT5) correction file from SCANLAB

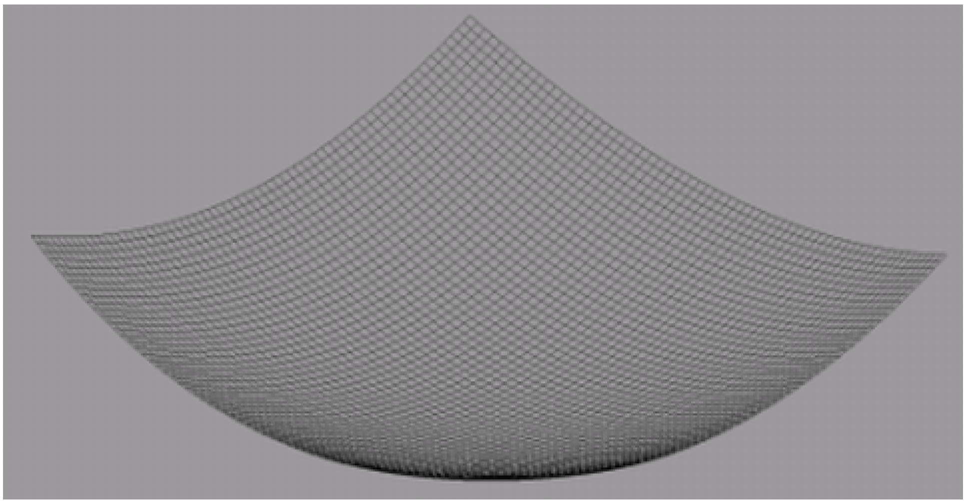


Figure 412: Shape of a 3D Correction File

Settings: For 3D marking with RTC boards, a 3D program file and 3D correction file has to be set. In the 3D Ext dialog the values A,B,C for the Z-table can be defined. You can find this dialog at *Settings → System → Card → Advanced → Driver Settings → 3D Ext*. For the values please ask the scan head manufacturer.

If Z channel is wired to the 2nd scan head connector, enable the 2nd head and assign the same correction file at *Settings → System → Card → Correction.*

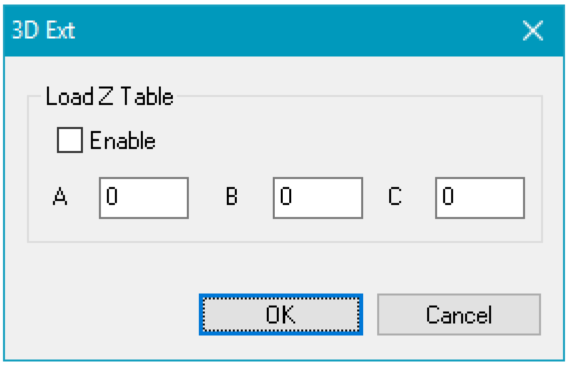


Figure 413: 3D Ext Dialog for RTC cards

20 Option SAM3D

This chapter describes the 3D functionality of SAMLight. Before working in the 3D mode *Menu bar → Settings → System → 3D → General, 3DView* must be checked. Then click on *OK* and restart the software.

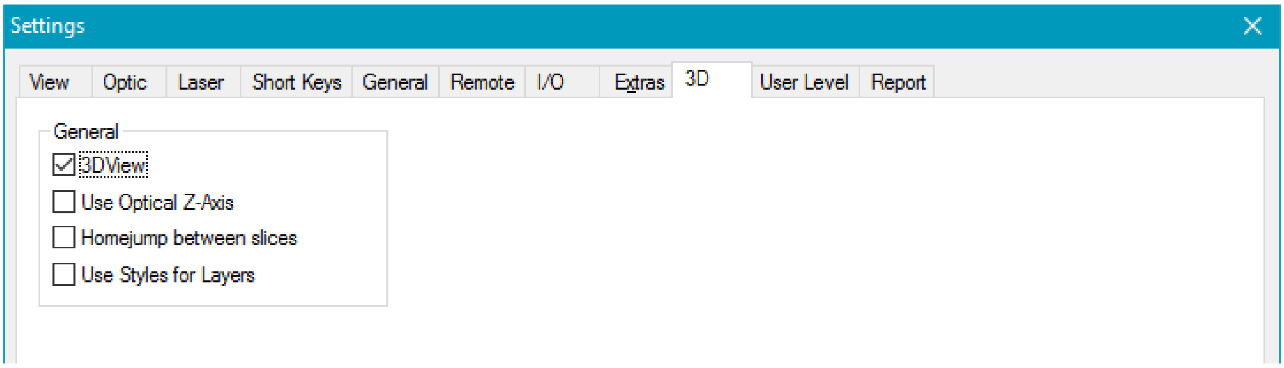


Figure 414: 3D Settings Dialog

General:

3DView: Enables / Disables SAM3D mode. The software must be restarted for the change to take effect.

Use Optical Z-Axis: Check this option, when you want to shift the focus optically with a 3D scan head. Please note: you need a license for Optic3D to use this option.

Home jump between slices: Depending on the settings of the home jump in *Settings → System → Optic → Home Position* there are two different options available:

- If the home jump is enabled a home jump will be performed after each marked slice.
- If the home jump is disabled there will not be a home jump, but the laser power of the HomeJumpStyle is set after each marked slice.

Use Styles for Layers: Enables the Styles property page to assign different pens and hatches to the sliced layers of the 3D object. See chapter "Styles for Layers".

20.1 Main Window

In the following the main window of SAM3D is shown. This window appears after software start.

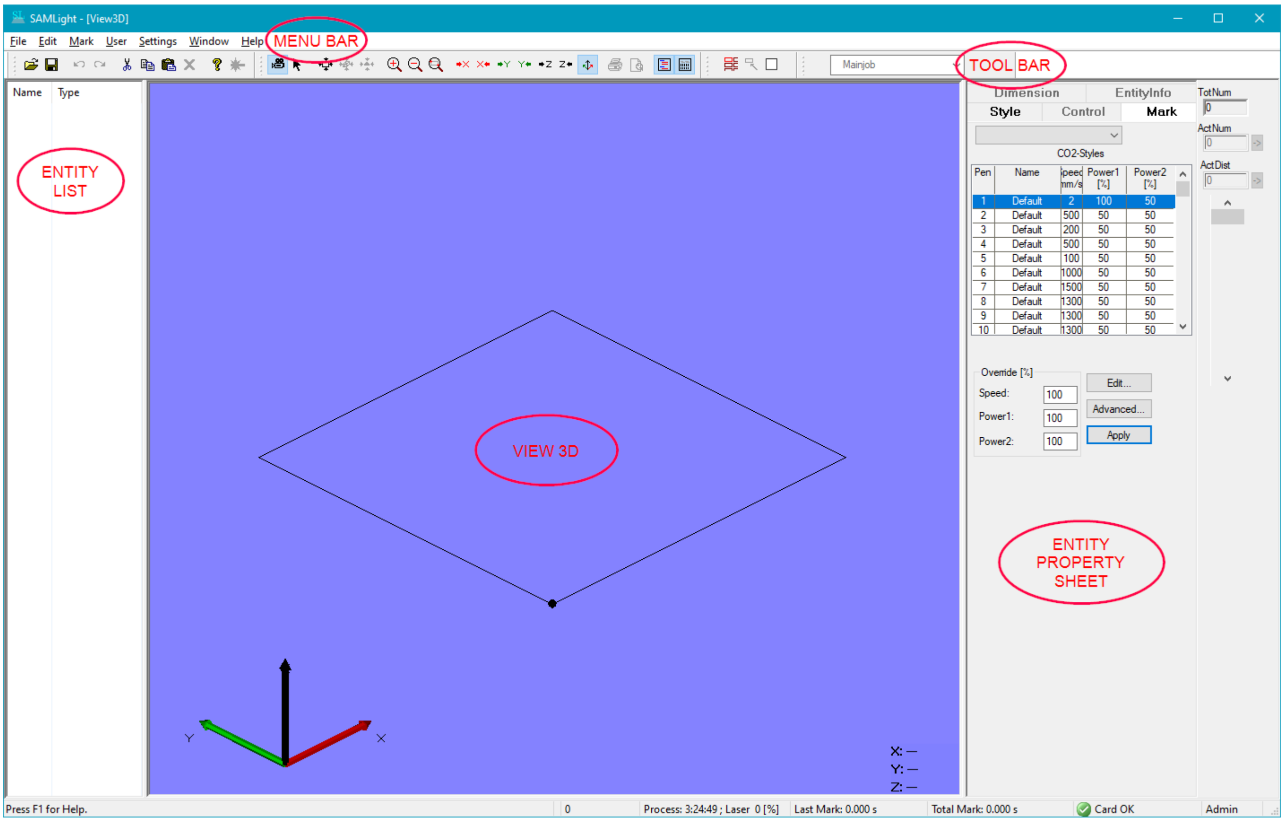


Figure 415: Main Window for 3D Mode

It is very similar to the main window of SAMLight. It consists of the *Menu Bar*, the *Toolbar*, the *Entity List*, the View 3D and the *Entity Property Sheet*. The *Menu Bar* contains only the open and save file functions. The *Toolbar* consists of functions to change the view. The *Entity List*, the *View 3D* and the *Entity Property Sheet* are similar to the items in the SAMLight 2D application. The rotation in SAM 3D is around a vector, which can be defined in Property page→ dimension. The black square in the View 3D shows the working area in the X and Y directions.

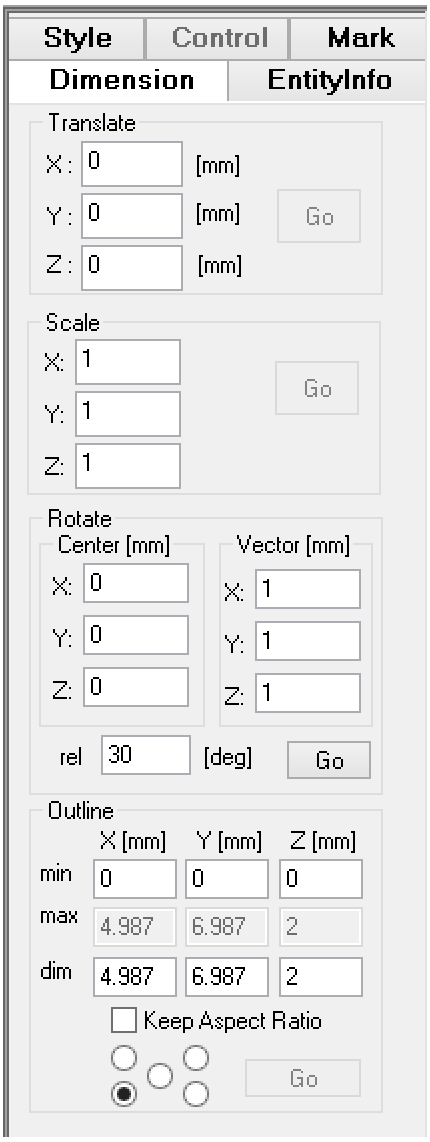


Figure 416: Dimension

Translate: Translation values X, Y and Z are relative values in respect to the current position.

Scale: Scaling values X, Y and Z are relative values in respect to the current size of the entity.

Rotate: Rotates the STL around the axis of rotation by the given *Angle*.

The axis of rotation is defined by two points:

- Point Center (C_*X, C_Y, C_Z*)
- Point Q (C_*X+V_X, C_Y+V_Y, C_Z+V_Z*).
- Here, V_X, V_Y and V_Z is defined by Vector.

Outline:

- **min**: X, Y and Z position at the minimum border.
- **max**: Y, Y and Z position at the maximum border.
- **dim**: Defines the length in X, Y and Z of the outline. If Keep Aspect Ratio is selected, the relation between the values is kept constant after any outline transformation. The radio button activates a selection of the outline fields for modification.

20.2 Job Processing

In SAM3D, it is not possible to create a job like it is in the SAMLight 2D application. A job must be loaded from an *.s3d file or imported from an *.stl, *.cli or *.slc file. To load an *.s3d file, click on *Menu bar → File → Load...* . To import an *.stl, *.cli or *.slc file, go to *Menu bar → File → Import...* . Once a file has been loaded or imported, the 3D object is shown.

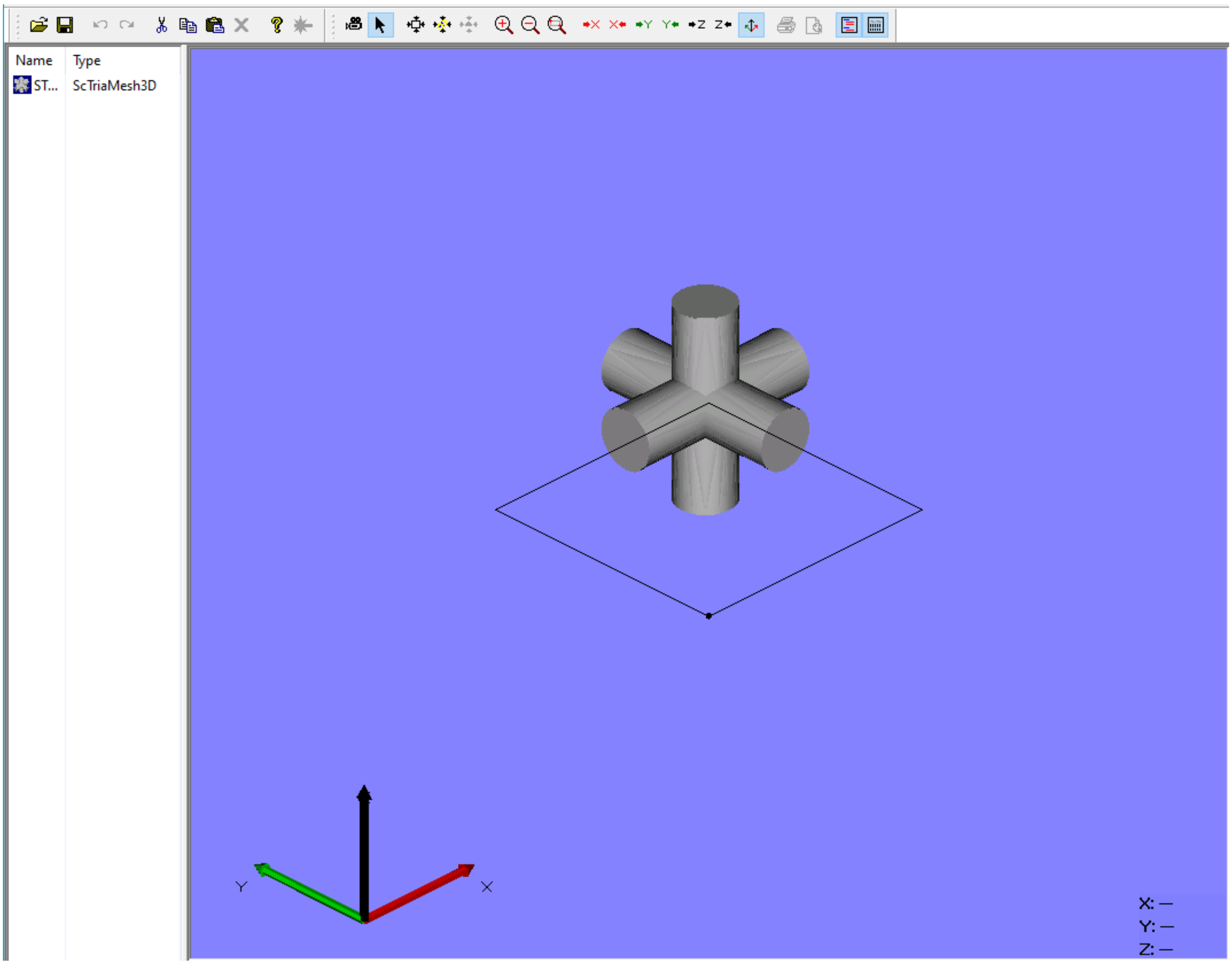


Figure 417: View 3D

On the left side, the entity list shows the name and the type of the 3D object. With a loaded object, actions to modify the view are possible (see Toolbar).

20.2.1 Import Folder

This dialog allows to import several slices in one turn and creates an ScLayerSolid out of them.

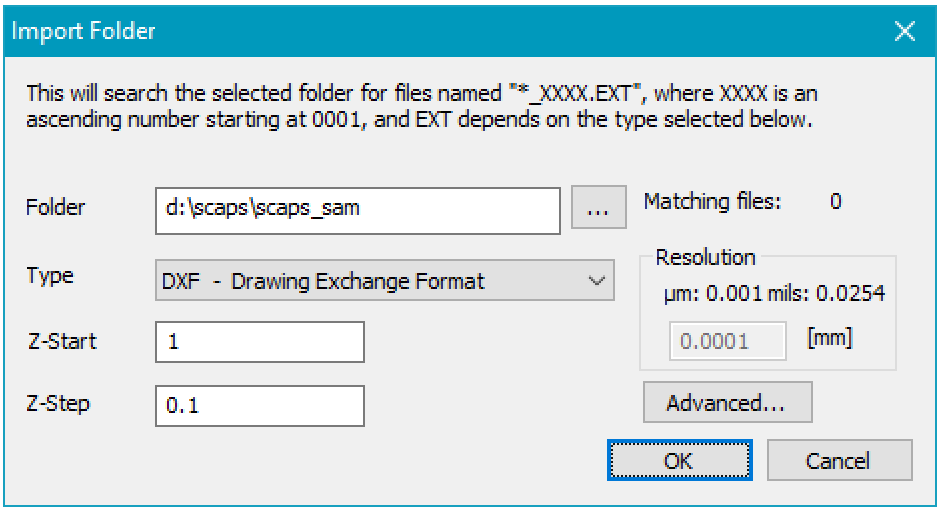


Figure 418: Import Folder Dialog

20.2.2 Toolbar

Toolbar: The Toolbar offers some basic functionality to modify the view of the object.

Camera Mode: If this is activated you can turn around the 3D object by clicking on the View 3D and moving the mouse while the left mouse button is being pressed.

Selection Mode: If activated you can select an object in the view.

Fit View To Working Area: The view is zoomed so that the whole working area is visible.

Fit To Entities: The view is zoomed so that the entities are visible with maximum dimension.

Fit To Selected: The view is zoomed so that the selected entities are visible with maximum dimension.

Zoom In: The objects in the view are enlarged.

Zoom Out: The objects in the view are scaled down.

Custom Zoom: You can choose a region in the view that will be enlarged. Therefore click with the left mouse button on one corner of the desired region. While the mouse button is pressed move the mouse to the other corner and release the mouse button.

View Along Axis: Watch the object along the X,Y or Z axis.

Standard 3D View: Watch the object from a standard 3D view point.

20.2.3 Mouse Mode

There are two different mouse modes available to transform the entities. One is the Transform Camera mode and the other one is the Transform Entity mode. Both transformations work with the mouse while pressing a key on the keyboard:

Camera Transform: Activate Camera Transform by clicking on the Camera symbol in the toolbar. Now the following actions are possible (**LMB = Left Mouse Button**):

- **LMB:** Rotate around the horizontal or vertical axis
- **Ctrl + LMB:** Scale
- **Shift + LMB:** Translate
- **Shift + Ctrl + LMB:** Rotate around the z-Axis

Entity Transform: Activate Entity Transform by clicking the Selection symbol in the toolbar. Now the following actions are possible:

- **LMB:** Select and unselect the entity
- **Ctrl + LMB:** Selection of multiple entities
- **Shift + LMB:** Translate selected entities

- **Shift + Ctrl + LMB:** Rotate selected entities. The rotation axis depends on the view. In the standard view (exceptional ISO) the rotation axis is the axis which comes out of the view. In all other views the rotation axis is the Z-Axis of the workpiece coordinate system.

20.2.4 3D View Properties

Click the right mouse button while the mouse pointer is inside the View 3D to open the View Properties.

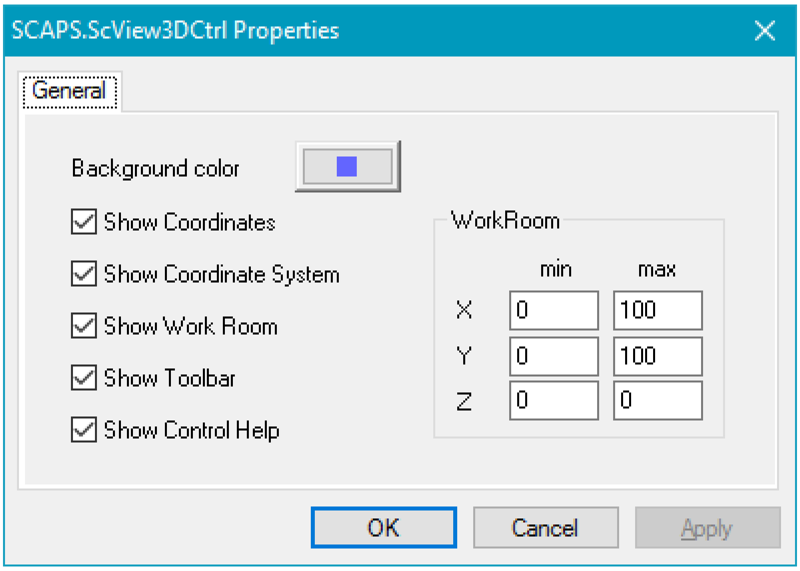


Figure 419: 3D View Properties Dialog

Here, choose the background color, activate or deactivate the options and define a work room.

20.2.5 Slicing

Before you can mark a 3D object, it has to be sliced. This means, it has to be separated into many 2D layers, which then can be marked as normal 2D objects. After marking a layer, a motor has to move the target object to the next z-position or the z-focus of the scan head has to be set to the next z-position. You can choose between these two options by checking or unchecking *Menu bar → Settings → System → 3D → Use Optical Z-Axis*. To slice the object choose *Menu bar → Edit → Slice*. The following dialog appears:

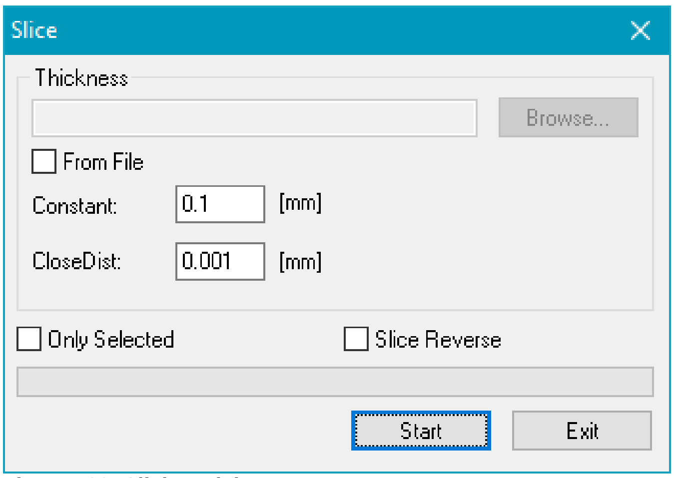


Figure 420: Slicing Dialog

Constant:

Determines the distant between two successive layers.

CloseDis:

If the distance between two polylines is smaller than this value, they will be closed automatically. With "0", the functionality is disabled.

Only Selected:

Only selected objects will be sliced.

Slice Reverse:

The default slice order is from the bottom to the top along the z-axis in positive direction. By activating this option the slice order is reversed. This will also reverse the order for marking. One possible application would be "Deep Engraving".

From File: The slicing information can be read from a file, too. This file is a plain *.txt file with the following structure:

```
2
number of slices
[R]
target_dist;step_width pen_number [hatch_number]
target_dist pen_number [hatch_number]
```

Here the *2* is a version number that corresponds to the internal structure of the file. This field is mandatory. The next line contains a number that is equal to the number of slices that have to be created for that file. This number is used for internal calculations and to provide an expressive progress bar. The third line specifies if all following *target_dist* parameters are relative to the base of the mesh (*R* set) or if the *target_dist* specifies absolute values in the used coordinate system (empty line without *R*).
All following lines describe the slices itself. Here two methods are possible. The first one describes a *target_dist* and a *step_width*. Here the *step_width* is used for the thickness of all slices until the specified target distance is reached. Using this syntax it is possible to define a range of slices just by using one single line. The second syntax provides the possibility to define one single slice. Here the *target_dist* specifies where this single slice has to end. Both methods of defining slices use the preceding slice position as starting point. The thickness of a slice results out of the difference between both values. Additionally a pen number is specified in both cases. This one-based pen number is assigned to the related slices. And as a third, optional parameter, the number of the hatch that has to be applied to that slice can specified. It corresponds to the default hatch styles of the hatch property pane. To use such a predefined hatch, here the required parameters have to be stored and the related hatch 1 and/or 2 has to be enabled.

As an example such a slice definition file can look like this:

```
2
13
R
0.10;0.01 1
0.11 2
0.15 1
0.17 3
```

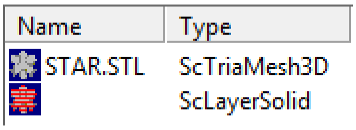
Here the file version number is *2* is used, to exactly specify this format. The number of slices defined within that file is *13* and all given distances are relative to the starting coordinate of the 3D mesh. First there is a range of slices defined. Here, ten slices have to be created from position *0.00* to position *0.10* with a thickness of *0.01*. Pen 1 is assigned to all ten slices. Subsequently, these three slices are created:

- a single slice from 0.10 to 0.11 (= thickness of 0.01), with pen 2 assigned
- a single slice from 0.11 to 0.15 (= thickness of 0.04), with pen 1 assigned
- a single slice from 0.15 to 0.17 (= thickness of 0.02), with pen 3 assigned

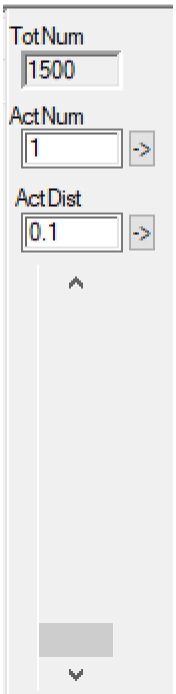
Reverse Slicing direction: If you want to reverse the direction of the slicing from bottom to top to top to bottom write R;S in the second line of the file. The Slice Reverse checkbox must NOT be activated.
Constant: Choose the distance between two successive layers in [mm].

Start: Start slicing and create a new entity which holds the sliced layers.

Once the object has been sliced, a new entity is shown in the entity list, and on the right hand of the main window a slice control becomes active:



The entity which holds the sliced layers is of the type *ScLayerSolid*. The entity which holds the original information of the 3D object is of the type *ScTrialMesh3D*.



The Slice Control allows to access the slices and get information about them.

TotNum: The total number of the slices.

ActNum: The number of the currently selected slice. You can enter the number of a slice and jump to it by pressing the arrow button.

ActDist: The z-distance from the ground of the currently selected slice. You can enter a distance and jump to it by pressing the arrow button.

Scroll bar: Click on the bar and move the mouse up or down to select a slice. The currently selected slice will also be shown in the View 3D as a green contour which indicates the layer.

NOTE

Slicing an entity more than once will lead to a corresponding number of sets of layers (ScLayerSolid entities).The slice distances may vary.
For every set of layers you can define a different hatch, pen etc. If you have more than one set of layers, the total number of slices corresponds to the number of slices with different z-value. If slices from different set of layers have the same z-value, they will be merged in the same layer to avoid motor activities.

NOTE

During slicing, polylines are generated. Depending on the orientation of the polylines, the lines will be displayed in different colors: inner polylines (orientation counter-clockwise) are displayed in red, outer polylines (orientation clockwise) are displayed in green.

The orientation is important for hatching, especially in combination with a beamcompensation. A mixture of inner and outer polylines will lead to a different beamcompensation. If unwanted, the Improved Beam Compensation could help.

NOTE

*While creating an entity of the type ScLayerSolid or loading a *.s3d file containing such one, a set of temporary files with name ddds3dHHH(H).tmp -and since SCAPS installer 4.0.5.0006 s3dHHH(H).tmp, with H being hexadecimal numerals- are generated at <SCAPS>\intermed folder to fluently handle the potential big amount of slicing data. These temporary files are deleted when all entities of the type ScLayerSolid are deleted, or the job or SAM3D are closed.*

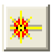
20.2.6 Hatching

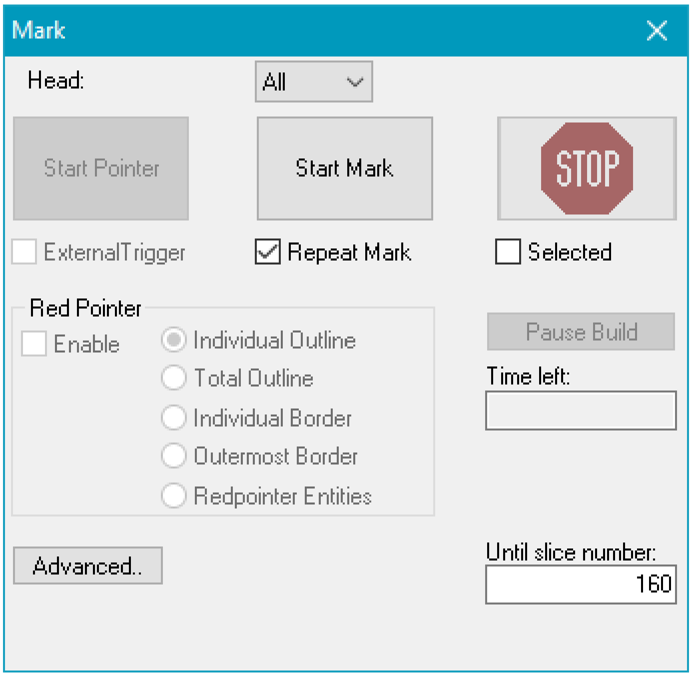
After slicing select the *ScLayerSolid* in the entity list and go to the hatch property page to create hatches for the slices. The SAM3D hatch property page shows one additional value compared to the 2D hatch property page:

Var. Angle: A variable angle from -360° to +360° can be defined to rotate the hatch angle between the slices.

In the mark property page, different pens can be chosen for the outline (PolyLines), Hatch1 and Hatch2.

20.2.7 Marking

By clicking on the mark icon  or by selecting *Menu bar → Mark → Start* the mark dialog is opened:



Start Mark: Starts the marking process. The first slice which will be marked is the number of the currently selected slice.

Repeat Mark: If checked, all slices (layers) will be marked one after another. If not checked only the currently selected slice will be marked.

Selected: If checked only the selected ScLayerSolid entities will be build.

Until slice number: The marking will stop, if this slice number has been reached.

Pause Build: The running build process will be paused after the current slice has been marked completely. If a slice post-process has been defined, it will be executed before pausing. If you click the button again during a paused build process, the next layer will be marked.

Figure 421: 3D Marking Dialog

Selecting *Edit -> show selected Layer* in the *Menu bar* shows the preview of which layer will be marked.

20.2.8 Special Sequences

Like in 2D Mode, it is also possible to define special jobs which are executed before or after the marking process. This can be the movement of configured motion controllers, as well as *ScOverride* entities or entities that wait for an external input signal, or entities that set a special output signal. Since the functionality is exactly the same in 2D Mode, please refer to the chapter Special Sequences in *User Interface → Toolbar*.

For example:

First setup a Mark Preprocessing.

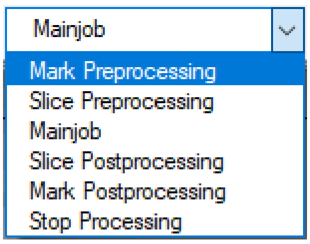
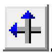


Figure 422: Jobs toolbar

In the window that opens create a Motion Control by clicking on the Motion Control Icon .

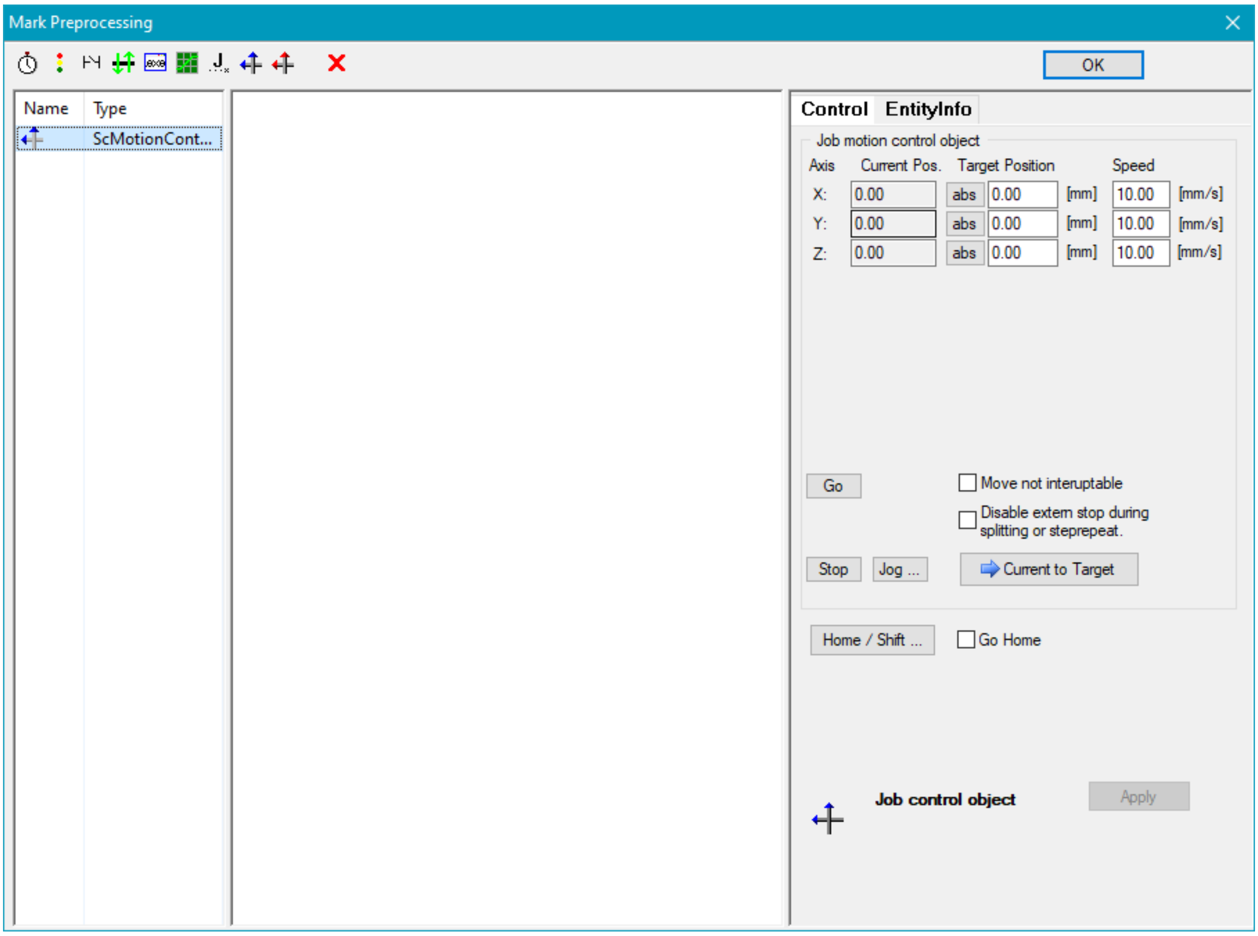


Figure 423: Mark Preprocessing Dialog

In this case, the motor will move to its start position at X=Y=Z=0 before the marking is started. Now, select *Slice Postprocessing* from the *Jobs toolbar*.

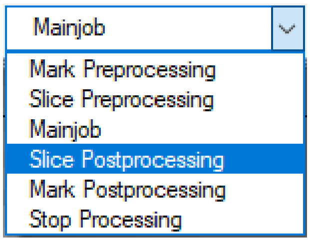


Figure 424: Jobs toolbar

Then, create a motion control entity:

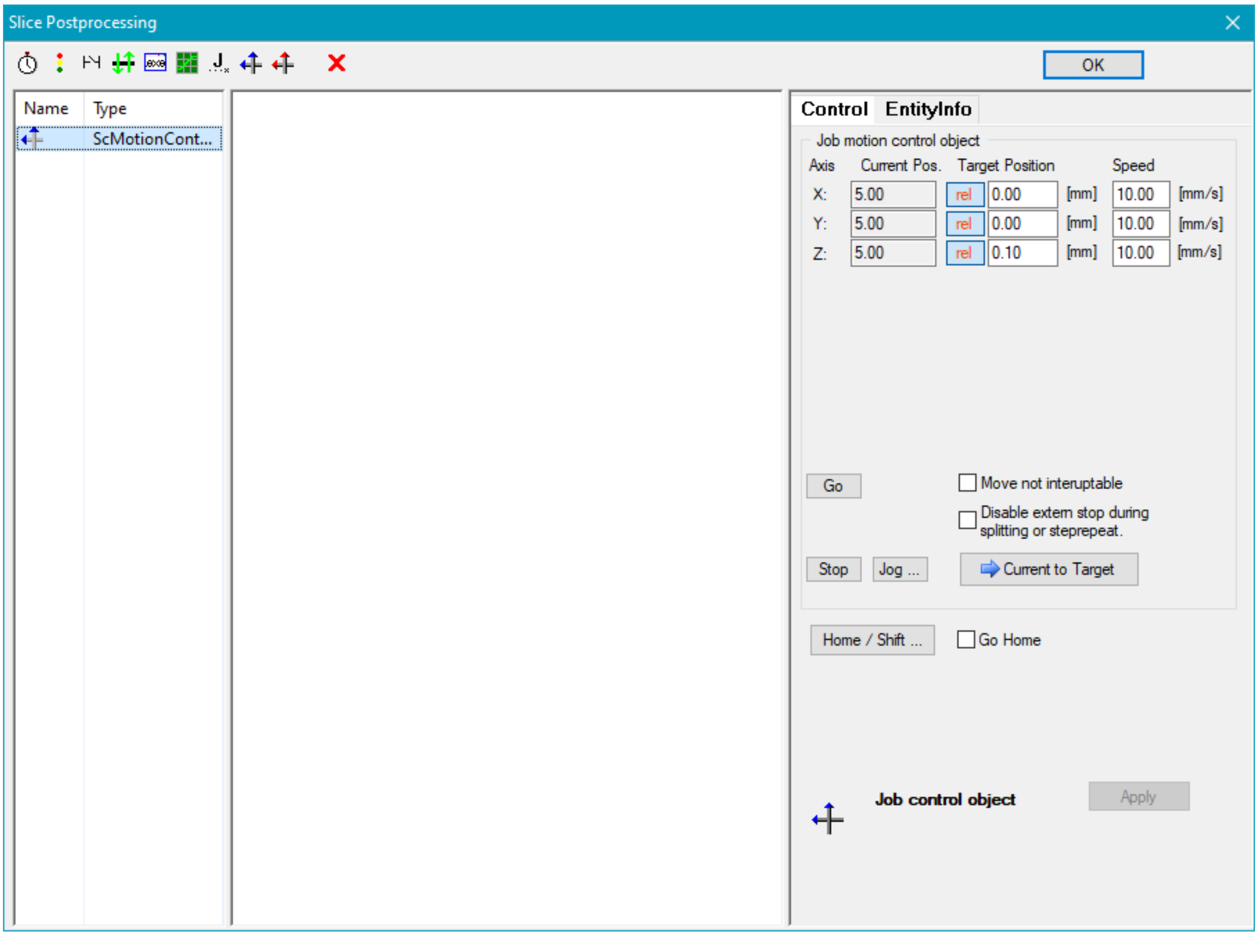


Figure 425: Slice Postprocessing Dialog

In this example, a motion control is inserted after every mark of a slice, which will move the Z-axis *0.1* mm and leave the actual X and Y position. Now, the job is ready to be marked as a complete 3D Object.

Job Properties: It is possible to store the jobs in the Jobs Toolbar when saving the 3D job in a *.s3d file. Therefore, go to File → Job Properties and enable the checkbox "Save 3D pre/post jobs in 's3d' job file".

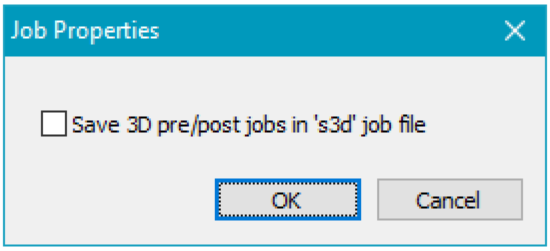


Figure 426: Save 3D pre/post jobs in file

20.2.9 Styles for Layers

To use this feature, the checkbox "Use Styles for Layers" has to be activated in Settings → System → 3D. Then, the Styles property page will be visible. This feature lets you define Pen, Hatch, Mark Loop Count, Mark Contour and Mark Hatch for each Styles, as well as enabling you to use the Upskin and Downskin features. This means that each layer is divided into sub-areas depending on the content of the layer above and below this specific layer. Upskin is called the area of the current layer which is without content in the layer above, Downskin is called the area of the layer which has no content in the layer below.

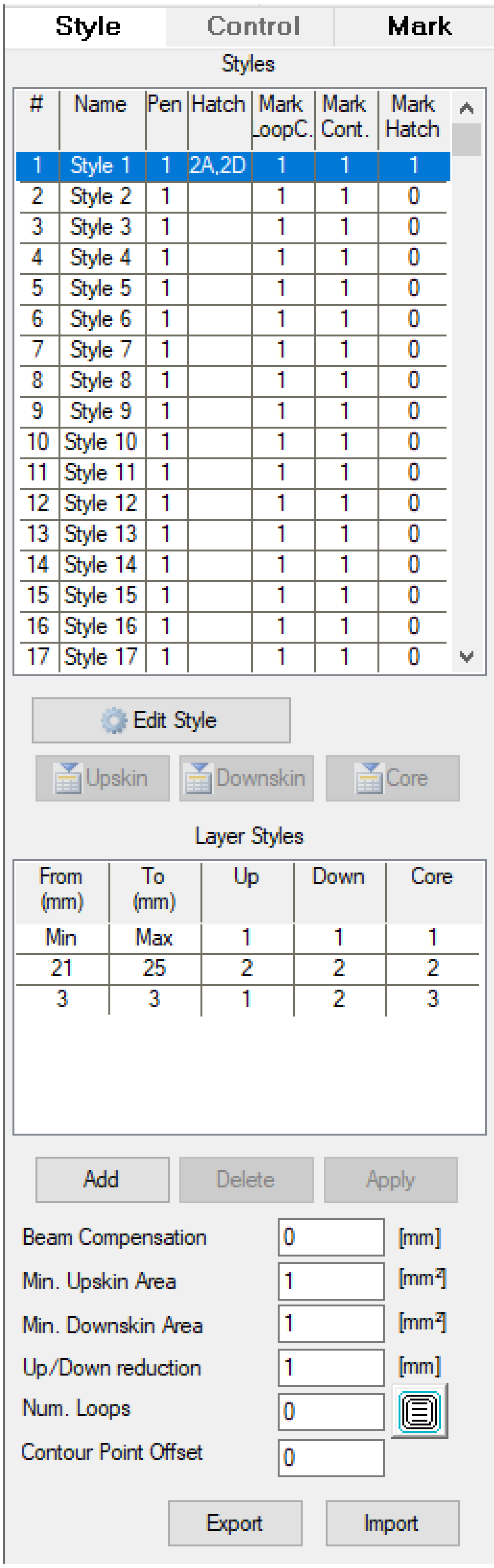


Figure 427: Styles Properties

Edit Style: Opens a dialog in which you can define a Pen, Hatch, and Mark Flags for each Style.

Upskin, Downskin, Core: If a ScLayerSolid entity is available, which is the case if you have sliced a ScTriaMesh3D object, you can define each style to Up- or Downskin or to Core for each Layer. Upskin is the area of the current slice that is not topped by the following slice, Downskin is the area of the slice which has no slice beneath it.

From - To: Defines a range of ActDist for the selected Style. In this case Style 1 is applied for the starting layer to the end layer interrupted by Style 2 for layers between 21 to 25 mm and also interrupted by different settings for layer at 3 mm.

NOTE *The numbers for **From-To** need to go from the smaller to the larger number along the z-axis. Careful if the object sits in negative z.*

Add: Add new From - To definition for Styles.

Delete: Deletes a From - To definition.

Beam Compensation: Reduces marking area for all layers.

Min. Upskin/Downskin Area:
Upskin/Downskin areas smaller than this value will not be created.

Up/Down reduction: Will enlarge the adjacent layers by that value before determining Upskin/Downskin areas. This will reduce the respective areas in order to compensate for irregularities.

Num. Loops: The contour lines can be repeated, the distance is defined by the Beam Compensation. With the button next to Num. Loops, the user can select the order of the contour marking from the inside to the outside or from the outside to the inside. The contour is always marked first.

NOTE *Negative Num. Loops are possible with a negative Beam Compensation and when Improved Beam Compensation is activated. Marking from the outside in is beginning from the contour is this case, and marking from inside to outside is choosing the marking order towards the contour.*

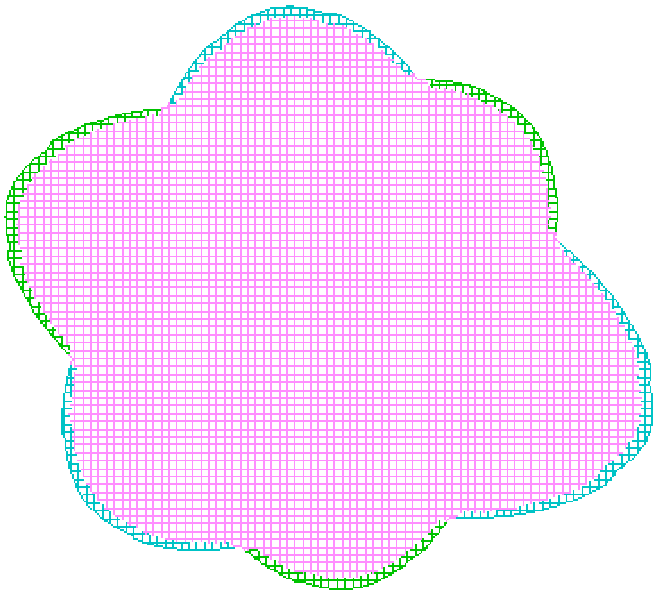
Contour Point Offset: Shifts the starting point for marking the contour lines by the given number of points.

Export / Import: Export or import Styles file. The type of the file is *.txt.

Within one layer (ScLayerSolid entity), the order of marking is the following: Upskin, Downskin, Core. Within each of these areas, polylines are marked first, linearrays are marked second (can be changed by "Mark Hatches First"). Within the Polylines, the order of num loops depends on the selected button (from inside to outside or vice versa).

20.2.9.1 Beam Compensation

The Beam Compensation will reduce the marked area, so that those objects that are not being marked become larger than before. Or said in another way, the marking area of the marked objects is reduced. See example below:



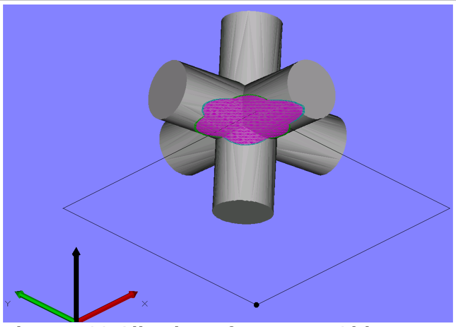


Figure 428:Slice in Reference to Object

Figure 429: Object with no Beam Compensation

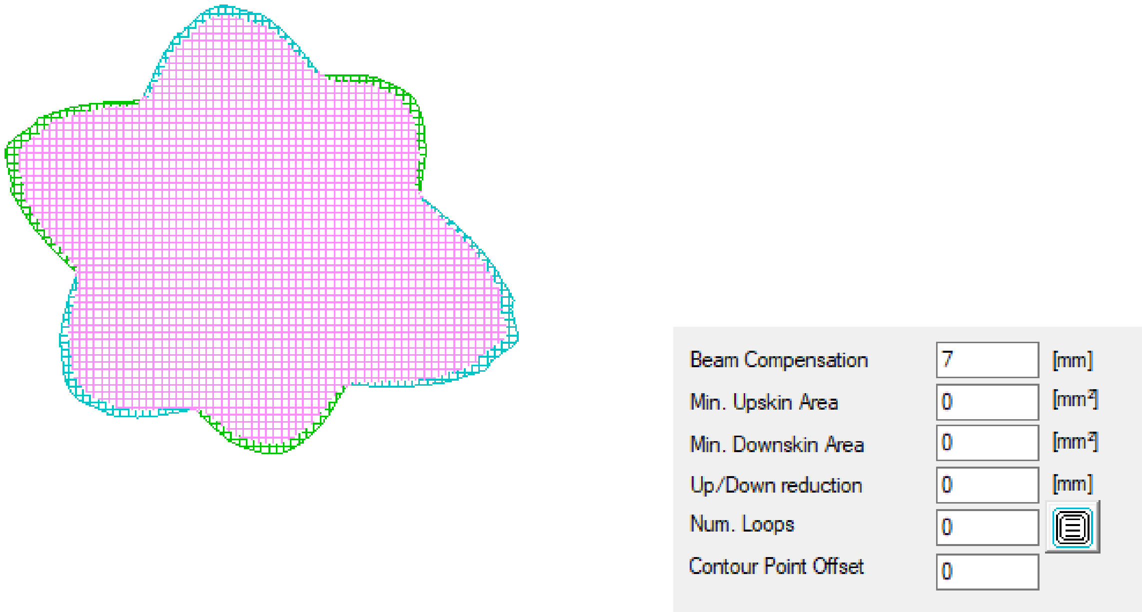


Figure 430: Object with Beam Compensation = 7 mm

For better visualization, the example was sliced with thicker slices. The settings are quite "coarse", accordingly .

20.2.9.2 Handling Up and Downskin

Sometimes, it is necessary to prepare the data regarding Up and Downskin. An example is given below:

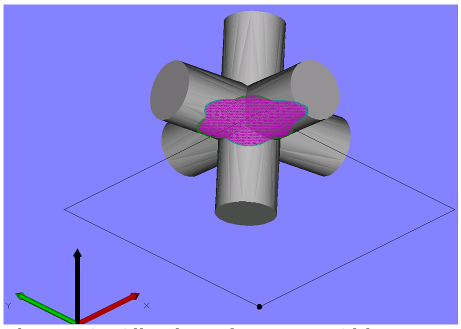
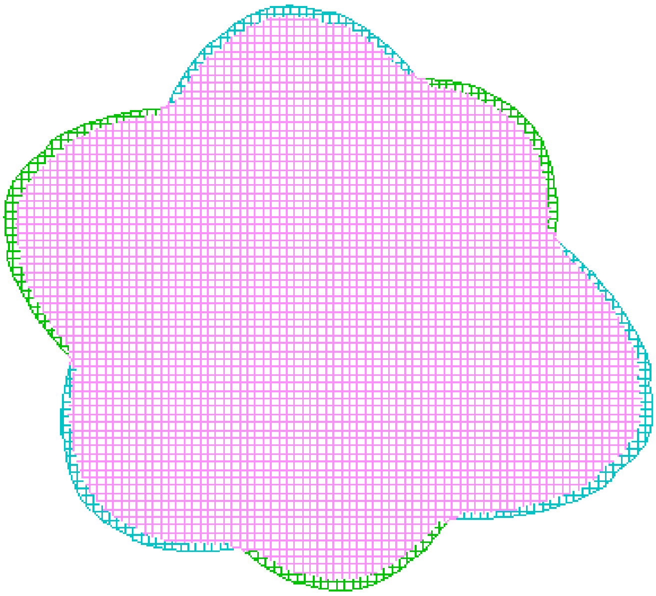


Figure 431:Slice in Reference to Object



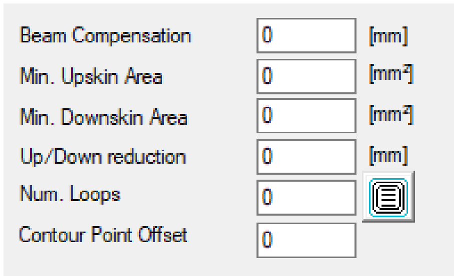


Figure 432: Up and Downskin on a hatched object

In the picture above, a 3D contour is hatched. At the borders of the object, Upskin (blue lines) and Downskin (green lines) areas are visible. To discard these areas, you can define a Min. Upskin or Min. Downskin area. See the following pictures:

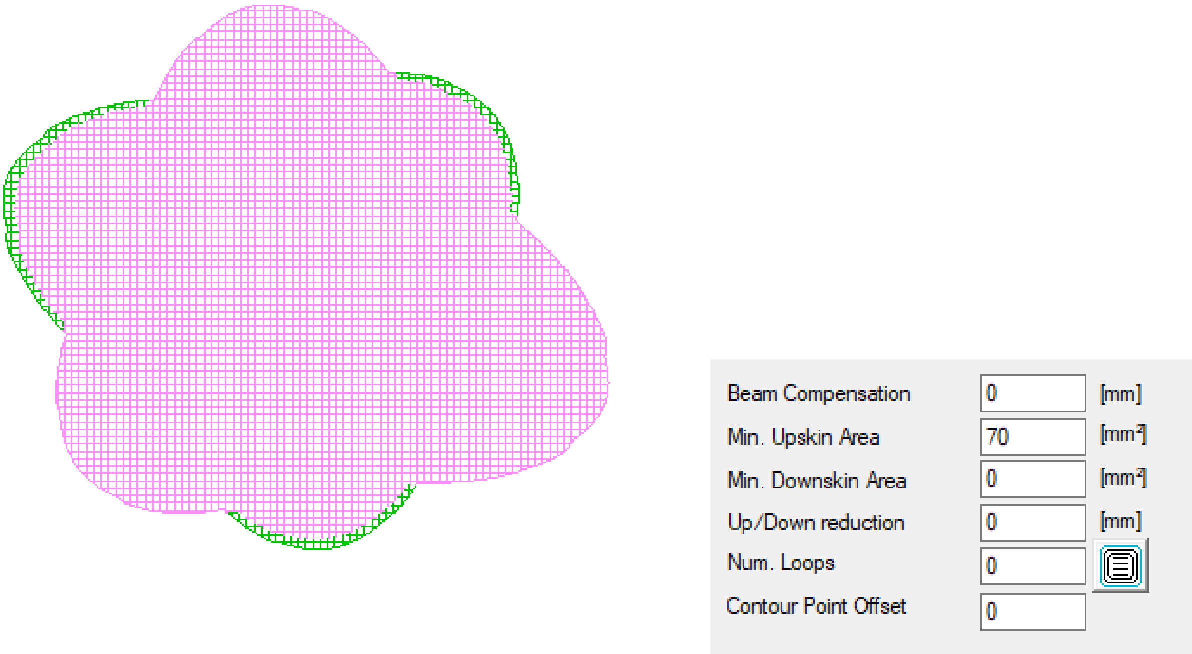


Figure 433: Upskin erased with Min. Upskin Area

In the picture above, it is shown how the blue lines = Upskin are erased by setting Min. Upskin Area = 70 mm².

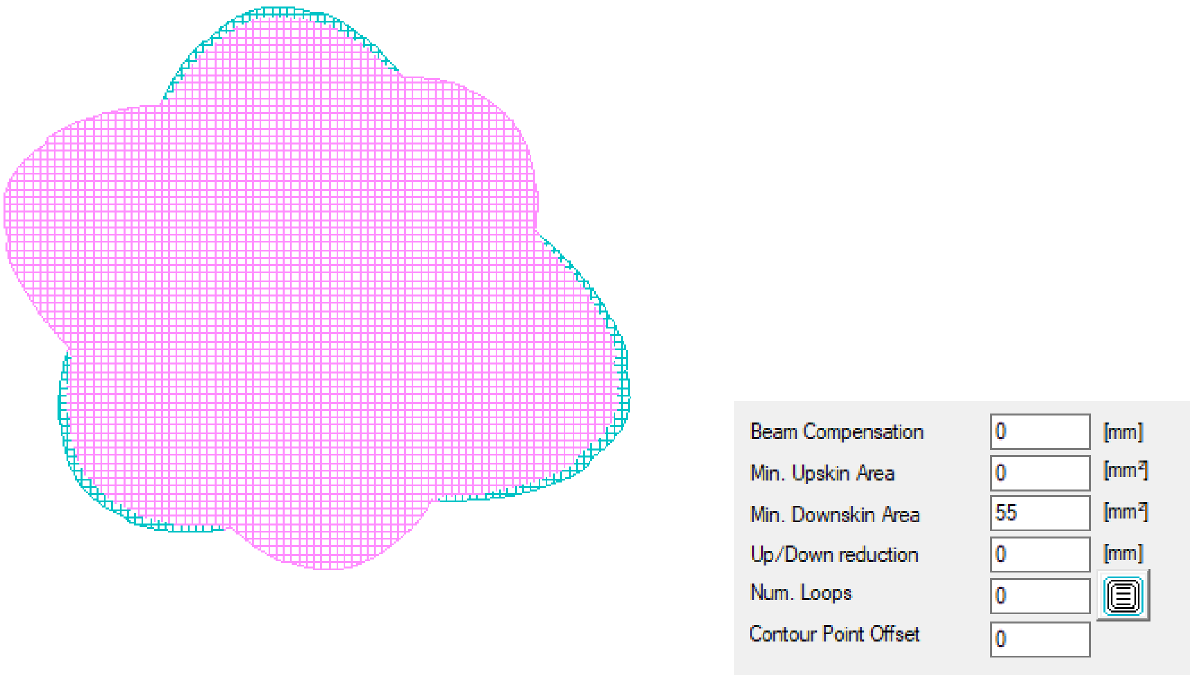


Figure 434: Downskin erased with Min. Downskin Area

In the picture above, it is shown how the green lines = Downskin are erased by setting Min. Downskin Area = 55 mm².

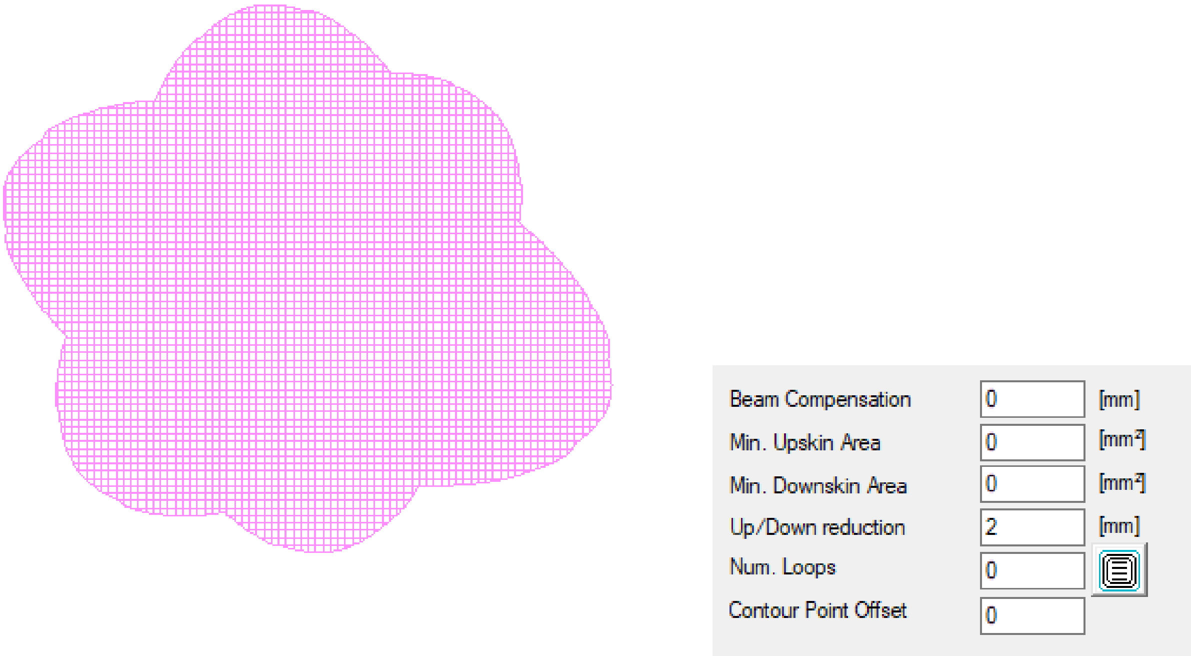


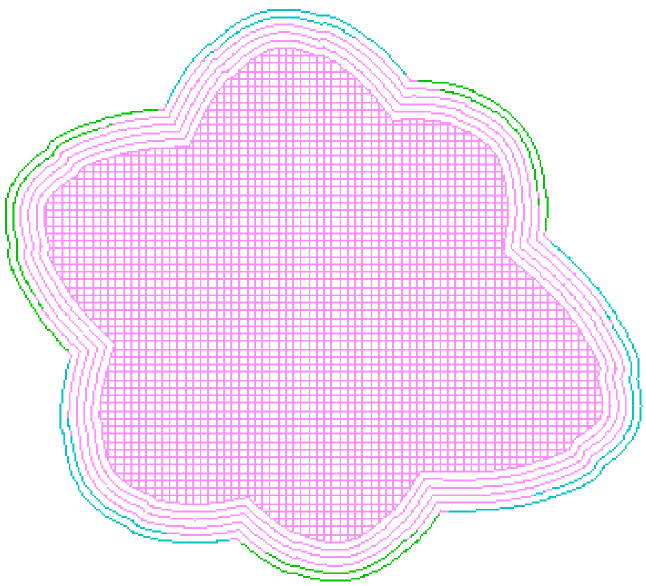
Figure 435: Up and Downskin erased with Up/Down reduction

In the picture above, it is shown how the green lines = Downskin and the blue lines = Upskin are erased by setting Up/Down reduction = 2 mm.

NOTE *For better visualization, the example was sliced with thicker slices. The settings are quite "coarse", accordingly .*

20.2.9.3 Using Num Loops

If necessary, the contour lines can be repeated, aligned to the border line of the hatched object. This can be done by defining a distance of these lines with Beam Compensation and the amount of lines using Num. Loops. Below is an example for Beam Compensation = 1 mm and Num. Loops = 5.



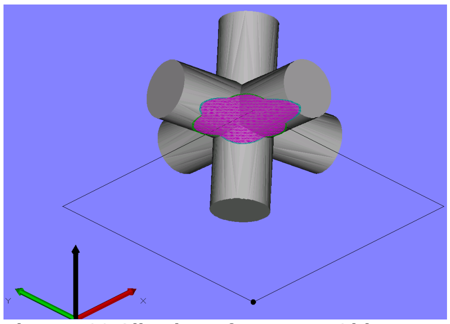


Figure 436:Slice in Reference to Object

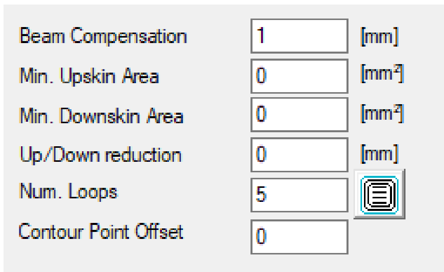


Figure 437: Using Num. Loops for a hatched object

There is a button next to Num. Loops with which the user can select the order of the contour marking from the inside to the outside or from the outside to the inside.
Contour Point Offset shifts the starting point for marking the contour lines by the given number of points.

NOTE *Negative Num. Loops are possible when choosing a negative Beam Compensation. Marking from the outside in is beginning from the contour is this case, and marking from inside to outside is choosing the marking order towards the contour.*

For better visualization, the example was sliced with thicker slices. The settings are quite "coarse", accordingly .

21 Client Control Interface

The Client Control Interface (CCI) can be used to remote control SAMLight from another software. A short overview over the different options of control is given here, examples and a more detailed documentation of all commands can be found at:
https://download.scaps.com/downloads/Software/Programming/Client_Control_Interface/

The **OCX (ActiveX) control SCAPS.ScSamlightClientCtrl** and the **COM control SCAPS.ScSamlightClientCtrlEx** are providing "Function" commands to remote control SAMLight from another software. This allows fast development of automation with different Windows development environments. OCX and COM can be used with a 32-bit or a 64-bit custom software. In addition, there is the possibility to use CCI without the OCX or COM encapsulation. In this case plain "ASCII" commands are sent via TCP from a different system which does not need to be a Windows PC.

In the following, these options are explained in more detail:

1. Remote control SAMLight inside the same Windows PC:	CCI Function commands (OCX or COM) via Function Calls
2. Remote control SAMLight from another Windows PCs:	CCI Function commands (OCX or COM) via TCP
3. Remote control SAMLight from any system supporting telnet-like network connections:	CCI ASCII commands via TCP

1. CCI Function commands (OCX or COM) via Function Calls

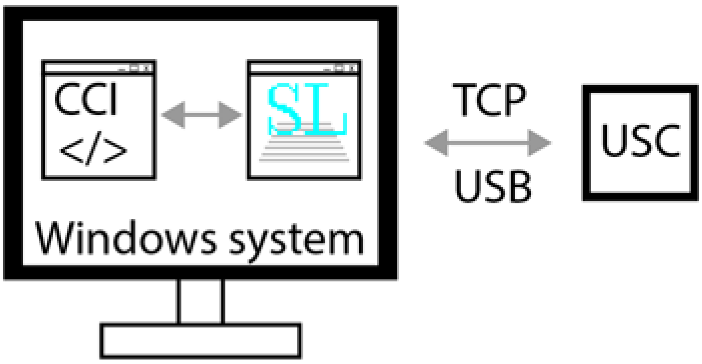


Figure 438: CCI Function commands (OCX or COM) via Function Calls

- SAMLight can be controlled by a CCI application running on the same PC
- SAMLight → Settings → System → Remote → Function Calls must be enabled.
- The client application can use all "Function" commands, see Command Set.

2. CCI Function commands (OCX or COM) via TCP

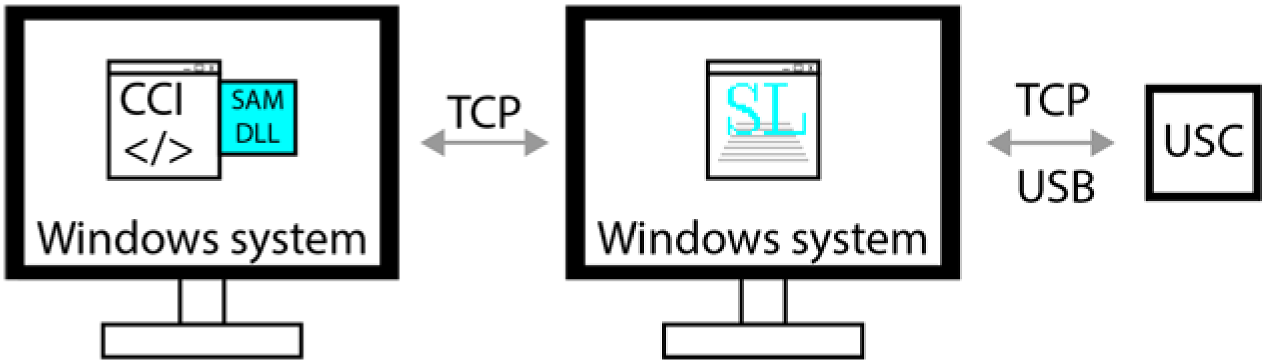


Figure 439: CCI Function commands (OCX or COM) via TCP

- SAMLight can be controlled by a CCI application running on a different PC. The SAM DLL (included in the SAMLight installer) must be installed on the remote controlling PC, meaning that both systems need to be Windows PC.
- The 2 Windows PCs are connected via TCP (Ethernet).
- SAMLight → Settings → System → Remote → TCP must be enabled.
- SAMLight → Settings → System → Remote → Bind To → IP Address must be from the PC running SAMLight. Use 0.0.0.0 to allow remote control from any IP or 127.0.0.1 for localhost.
- The client application can use all "Function" commands, see Command Set.

3. CCI ASCII commands via TCP

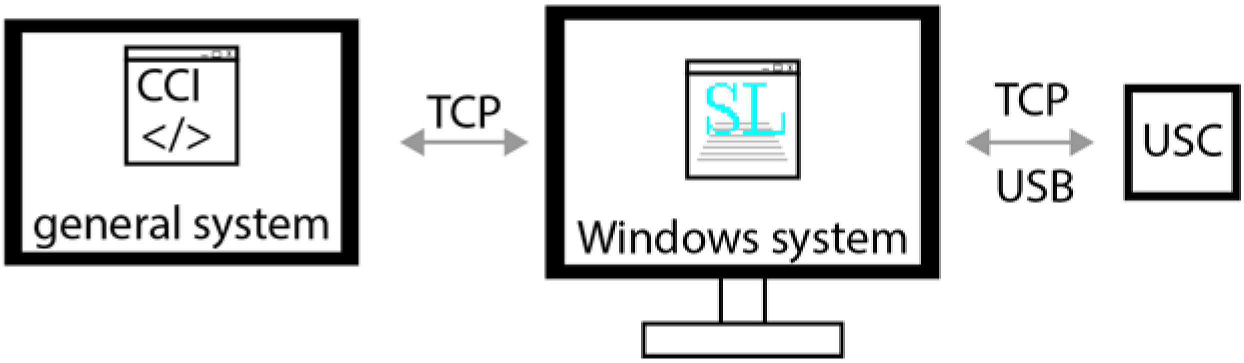


Figure 440: CCI ASCII commands via TCP

- SAMLight can be controlled by a CCI application running on a different system which does not need to be a Windows PC.
- The system and the Windows PC are connected via TCP (Ethernet).
- SAMLight → Settings → System → Remote → TCP must be enabled.
- SAMLight → Settings → System → Remote → ASCII Communication Mode must be enabled.
- SAMLight → Settings → System → Remote → Bind To → IP Address must be from the PC running SAMLight. Use 0.0.0.0 to allow remote control from any IP or 127.0.0.1 for localhost.
- The client application can use all "ASCII" commands, see Command Set.

22 How to

This chapter contains tutorials for special topics.

22.1 Use Simple Fonts

This tutorial describes how to use the simple line fonts (Laser Fonts) in the SAM modules.

22.1.1 Simple Fonts Format

The simple fonts have to be stored as Windows TrueType fonts. They are generated with the SCAPS converter. The SAM software detects the type and only generates lines. The advantage of using the TrueType Format is that the fonts can also be displayed in any Windows software supporting TrueType Fonts.

Figure 441: Example TrueType fonts in SAMLight

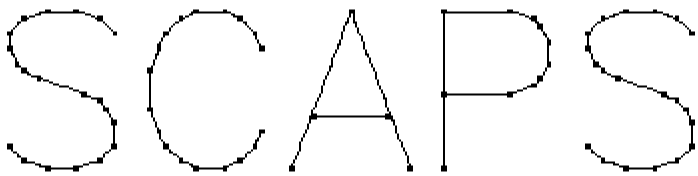
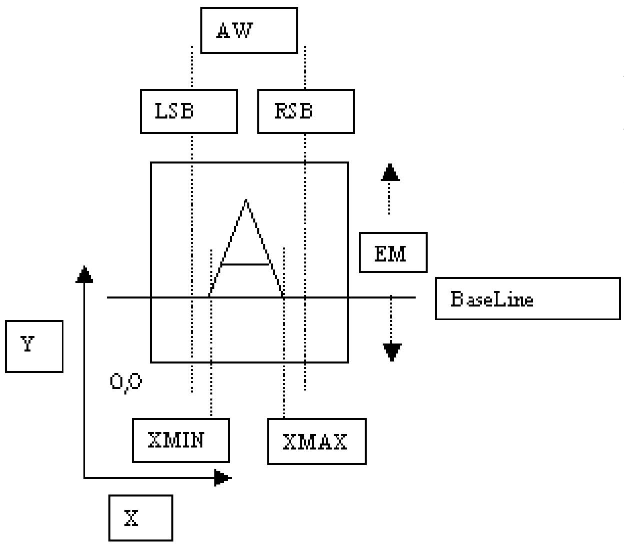


Figure 442: A closer look to a single line font

22.1.2 Generate Fonts

For the generation of "Simple" Windows True Type Fonts, SCAPS provides a converter tool for the translation of ASCII format files (SCAPS Font Format) to Windows True Type.

Each character of the font is defined with respect to a square character cell. The origin of the cell is the lower left corner. Also, each character (or also called Glyph) has a minimum X value XMIN and a maximum X Value XMAX (the bounding box X-co-ordinates).



EM: Size of square character cell.

XMIN: Beginning of the character.

XMAX: End of the character.

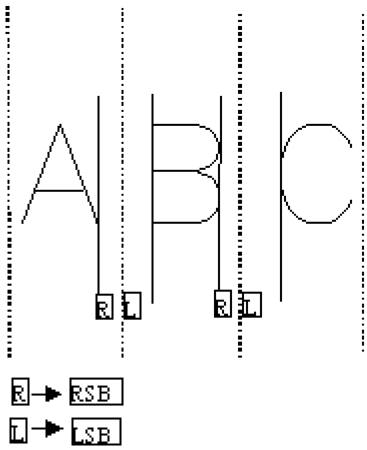
AW: Advanced Width as

AW = LSB + RSB + (XMAX - XMIN) , or

RSB = AW - LSB - (XMAX - XMIN)

In True Type Fonts only AW and LSB are defined. RSB can be calculated according the above equation.

Figure 443: TrueType font parameters 1



LSB: LeftSideBearing – Defines the gap between the RSB of the last character to the beginning of this character (XMIN).

RSB: RightSideBearing - Defines the gap between the end of this character (XMAX) to the LSB of the next character.

Figure 444: TrueType font parameters LSB and RSB

The baseline defines the line between the ascent and descent parameters of the font.

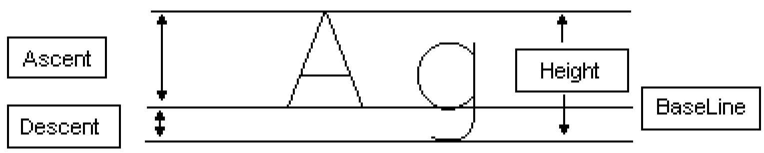


Figure 445: TrueType font parameters Ascent, Descent, Height and Baseline

The sum of Ascent and Descent plus an additional line spacing is the height of the font. BaseLine, Height, Ascent and Descent are global parameters of a specific font design.

22.1.2.1 Scaps Font Format

The SCAPS Font Format is an ASCII font description file format easy to read and to generate. For an example look at the file sc_straight_prop.sff in the folder INSTALLDIR/fonts which is the font used in the example below. Version 2.0 works with a EM of 8000 units in contrast to version 1.0 which works with an EM of 800 units.

Character description:

```
// CHAR 39
```

```
SI1280,640;SP1;PU;PA4000,7040;PD;PA4000,5360;PU;
```

Each character begins with // CHAR #, where # is the ANSI code of the character. The command SI specifies the AW and LSB values as described above. The command SP is optional and will be ignored. In the following line up to the next character description there is the geometrical information of the lines. The point information is stored in HPGL format by using the commands PU, PD and PA.

Header: The header is necessary for the font converter.

```
// SCAPS FONT FILE
// VERSION 2.0
```

Baseline: Y Distance from (0,0)

```
// BASELINE 2000
```

Examples:

```
// CHAR 0
SI8000,0;SP1;PU;PA0,0;PA4720,3440,4480,2960,4480,2480,4960,2000,5440,
2000,5920,2240,6160,2720,6160,3200,5680,3680,5200,3680;PU;PD;PA8000,
0,8000,8000,0,8000,0,0;PU;
// CHAR 33
SI1760,640;SP1;PU;PA4000,7040;PD;PA4000,3680;PU;SP1;PU;PA4000,2480;
PD;PA3760,2240,4000,2000,4240,2240,4000,2480;PU;
// CHAR 34
SI1760,640;SP1;PU;PA4240,7040;PD;PA4000,6800,3760,6320,3760,5840,
4000,5600,4240,5840,4000,6080;PU;
// CHAR 35
SI4880,640;SP1;PU;PA4120,8000;PD;PA2440,320;PU;PA5560,8000;PD;PA3880,
320;PU;PA2440,4880;PD;PA5800,4880;PU;PA2200,3440;PD;PA5560,3440;PU;
// CHAR 36
SI4640,640;SP1;PU;PA3520,8000;PD;PA3520,1040;PU;PA4480,8000;PD;
PA4480,1040;PU;SP1;PU;PA5680,6320;PD;PA5200,6800,4480,7040,3520,
7040,2800,6800,2320,6320,2320,5840,2560,5360,2800,5120,3280,4880,
4720,4400,5200,4160,5440,3920,5680,3440,5680,2720,5200,2240,4480,
2000,3520,2000,2800,2240,2320,2720;PU;
// CHAR 37
SI5600,640;SP1;PU;PA6160,7040;PD;PA1840,2000;PU;SP1;PU;PA3040,7040;
PD;PA3520,6560,3520,6080,3280,5600,2800,5360,2320,5360,1840,5840,
1840,6320,2080,6800,2560,7040,3040,7040,3520,6800,4240,6560,4960,
6560,5680,6800,6160,7040;PU;SP1;PU;PA5200,3680;PD;
// CHAR 38
SI6080,640;SP1;PU;PA6400,4880;PD;PA6400,5120,6160,5360,5920,5360,
5680,5120,5440,4640,4960,3440,4480,2720,4000,2240,3520,2000,2560,
2000,2080,2240,1840,2480,1600,2960,1600,3440,1840,3920,2080,4160,
3760,5120,4000,5360,4240,5840,4240,6320,4000,6800,3520,7040,3040,
6800,2800,6320,2800,5840,3040,5120,3520,4400,4720,2720,5200,2240,
5680,2000,6160,2000,6400,2240,6400,2480;PU;
// CHAR 39
SI1280,640;SP1;PU;PA4000,7040;PD;PA4000,5360;PU;
```

22.1.2.2 Scaps Converter

This tool 'sc_font_converter.exe' converts SCAPS Font Files into Windows True Type Font files. It can be found in the directory <SCAPS>\tools. The standard fonts with character IDs from 0 to 255 can be used. Additional characters can be added. The converter tool maps the EM size of 8000 to a design size of 10 mm. This means, that all numbers and dimensions shown in the converter are valid for a font generated with 10 mm height.

Please note that this tool will only run with a present SCAPS dongle and not with a trial key.

WARNING *Make sure that SAMLight is not running at the same time.*

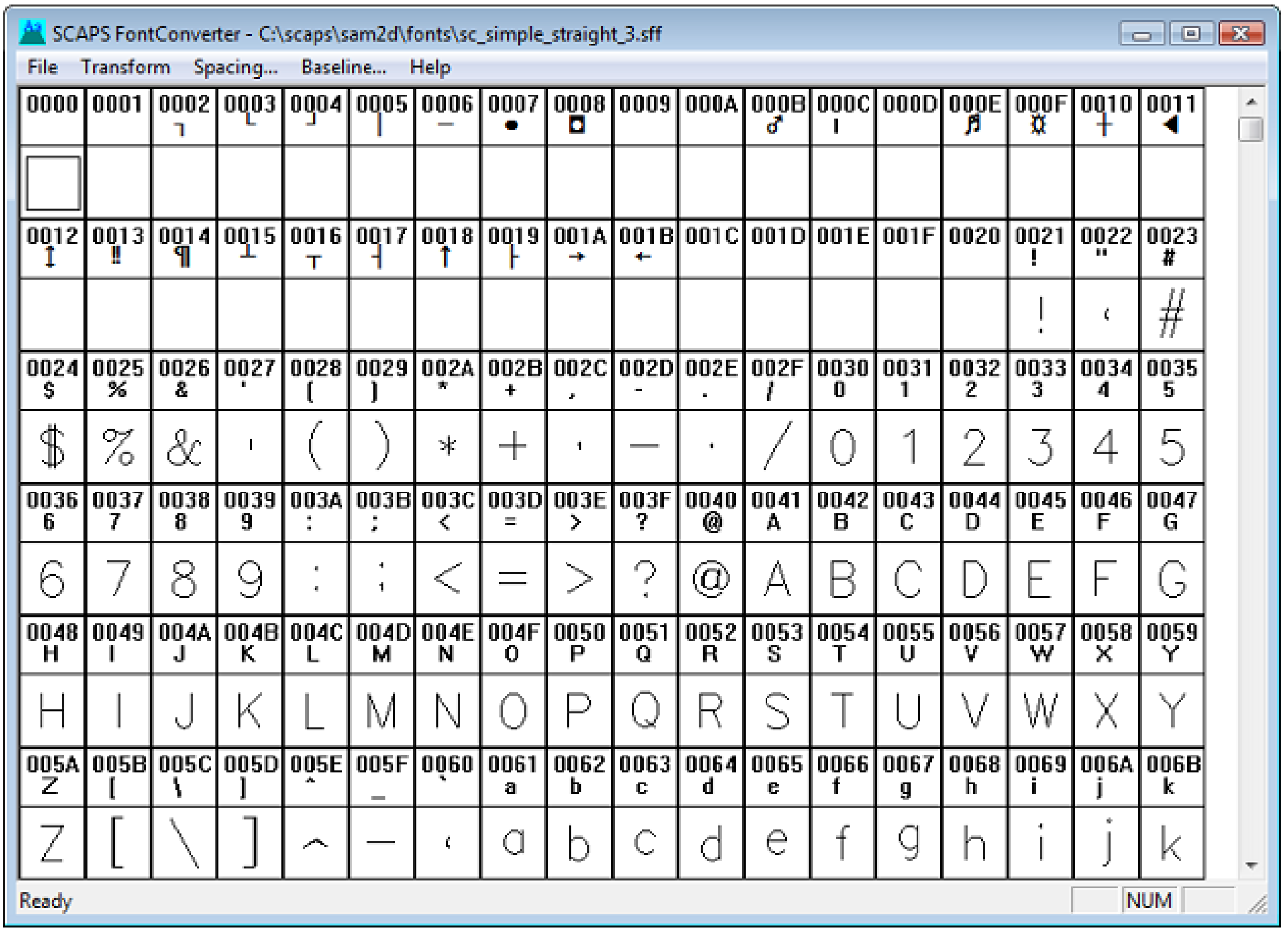


Figure 446: Font Converter Main View

File: The commands New, Load ; Save and Save As are handling *.sff files. Character 0 will always have a rectangle glyph with maximum size ((0,0)(8000,8000)) EM → ((0,0)(10,10)) mm Design, independent from what is stored in the *.sff file. Clicking on the glyph cells highlights the corresponding cell. Double Clicking on the cell activates the Edit View. The command Convert opens the Convert Dialog.

Transform: The dialog scale allows a global uniform character scale. Baseline, AW and LSB parameters will be scaled also. The scale origin is (0,0). Character 0 (the reference character) will not be scaled.

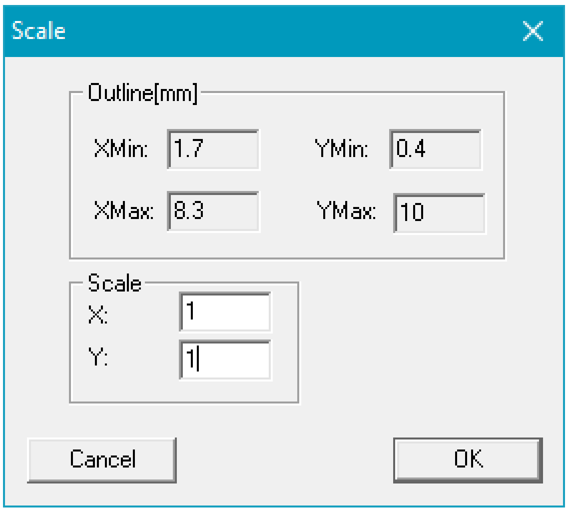


Figure 447: Font Transform Dialog Scale

For version 1.0 Fonts a scaling factor of 10 leads to comparable results with version 2.0. The outline fields show the maximal and minimal values of the font in both directions. The outlines should not exceed 10 mm in both directions. Otherwise you can not calculate with a 10 mm font height.

Spacing: The dialog spacing allows the definition of global calculation parameters for AW and LSB. See also the chapter "Generate Fonts".

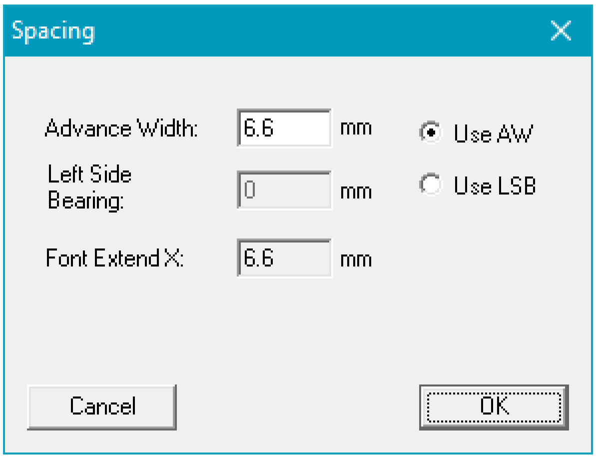


Figure 448: Font Spacing Dialog

Use AW: The AW parameter will be kept constantly. LSB and RSB will calculated to be equal.

AW = LSB + RSB + (XMAX - XMIN)   (LSB == RSB == SB)

SB=(AW - (XMAX - XMIN)) / 2

This setting leads to monospaced fonts.

Use LSB: The LSB parameter will be kept constantly. LSB and RSB will calculated to be equally.

AW = LSB + RSB + (XMAX - XMIN)   (LSB == RSB == SB)

AW=2 * SB + (XMAX - XMIN)

This setting leads to variable spaced fonts. The FontExtend X field shows the X-Dimension (XMAX - XMIN) of the largest character in x-direction inside the font.

Baseline:

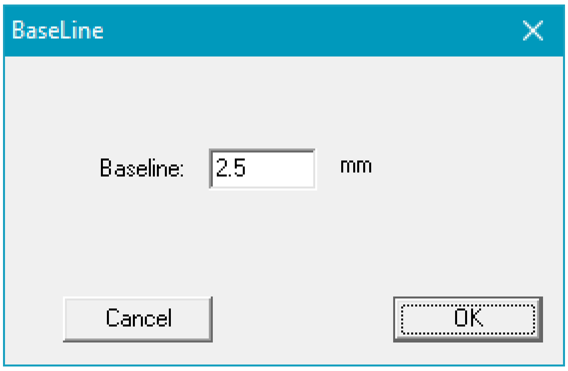


Figure 449: Font Baseline Dialog

With this dialog the user may define the base line of the font. The default value is 2.5 mm.

Edit View: The Edit View allows the editing of the glyph polygons and the AW and LSB parameters for this glyph. You can reach this dialog by double-clicking the corresponding character cell. You can also import a HPGL,SAF or DXF file and attach it to the character. The outline of the character, the AW and LSB should not exceed the 10 mm outline box. The base line cannot be changed because this is a global parameter and applies to all characters in a font. Look to the ScView2DCtrl documentation for more details for the editing functionality.

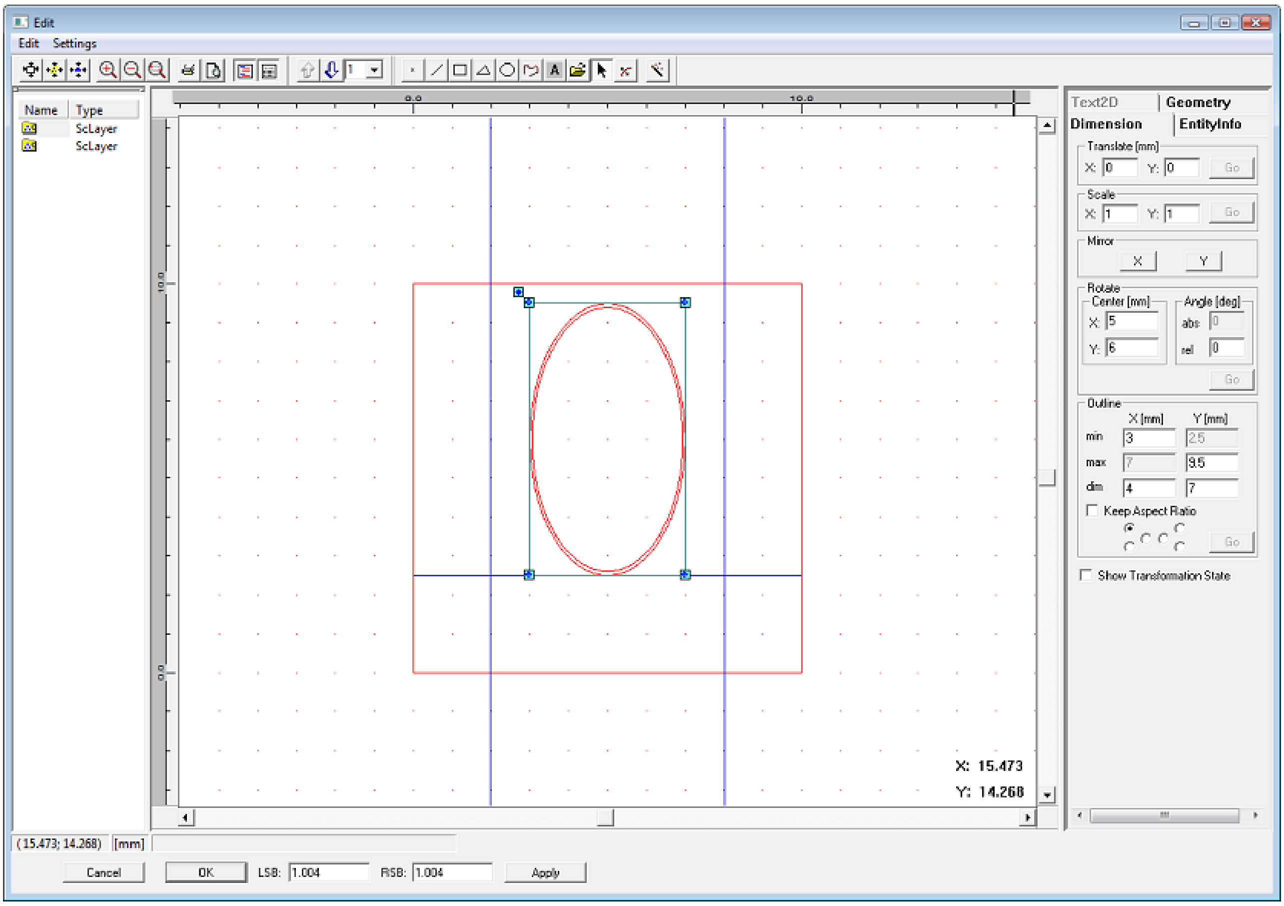


Figure 450: Font Edit View

Convert:

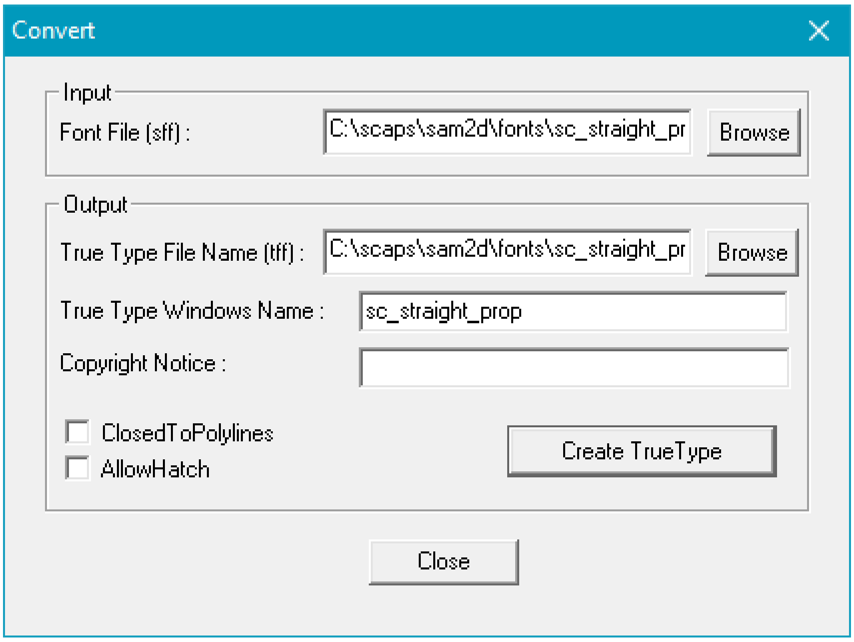
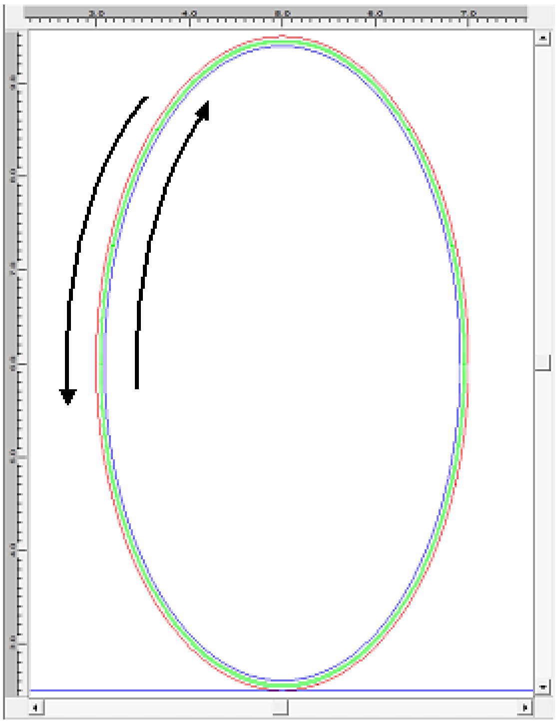


Figure 451: Font Convert Dialog

Parameters:

SCAPS Font File (sff)	File that has to be converted into the True Type Format
True Type file Name (ttf)	True Type Font file that has to be generated
True type Windows Name	The name of the Font
Copyright Notice	Company name of the font designer
ClosedToPolyLines	The True Type font has closed PolyLines. The WinText2D entity generation process creates closed PolyLines out of the character outlines whenever it is possible.

Closed PolyLines will be checked regarding their orientation before they are stored to the True Type Font file.



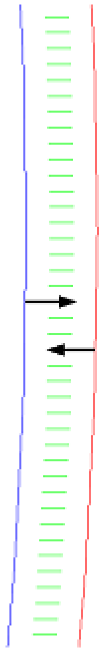


Figure 452: PolyLine Orientation Example

Outer PolyLines are oriented counterclockwise, inner PolyLines in the opposite. This is necessary to allow the Hatch beam compensation algorithm to determine the direction of the compensation as shown above for the letter '0'.

After filling in the parameters and clicking "Create True Type" button the font has to be installed by copying the file into the directory WINDOWS\FONTS. Now the font can be used in SAM2D for the generation of Single Line Characters. You may also have a preview of the font by double clicking on the *.ttf file inside Windows Explorer. Please make sure that a font with the same name is not already installed. In this case Windows will always take the installed font for displaying and you can not see the selected *.ttf file.

The program does not save the current loaded sff file. If you want to convert the current file you have to save it first.

Adding own property pages:

The font converter allows to add own property pages to the Edit View. The ASCII file sc_font_convert.prp located inside the SAM system folder can be manipulated. This works in the similar way as adding a property page by calling of the ScEntityPropertySheet.ScAddPage(Name) command. By default the following pages are added:

```
SCAPS.ScEntity InfoPropertyCtrl
SCAPS.ScDimensionPropertyCtrl
SCAPS.ScGeometryPropertyCtrl
SCAPS.ScText2DPropertyCtrl
```

To add a user property page the following line may be added:

```
SCAPS.ScUserPropertyCtrl.UserPropertyPageName
```

where UserPropertyPageName stands for a valid property page registered on the system.

22.2 Command Line Parameters

The scanner application can be started using different command line parameters. These parameters control its behaviour in different ways. The following values are supported:

command line parameter	description
/ActiveCard=<0, 1, …, 5>	If multiple scanner cards are connected to the PC, this command line parameter allows to select a card to be used with SAMLight. The number of the card is zero based and the valid range is from 0 to 5. Use the command line parameter /SettingsFile=<*.sam> as well, to chose a settings file. A single card settings file is required.
/DisableHomingStopButton	The STOP button during a homing procedure on startup of SAMLight can be disabled.
/DisableMessageBeep	Will suppress the Beep after some SAMLight actions (e.g when marking is finished)
/Hidden	SAMLight starts invisible in the background.
/JobEditor	SAMLight is started in JobEditor mode. Hardware output is not possible in this mode.
/LoadJob=<path*.sjf>	Using this parameter, a job defined by the path can be loaded during startup automatically. Please note that this option can be overwritten by the appropriate settings within the scanner application where you can define a job for loading on startup too.
/MarkTriggerUpdateBeforeEnd	Marking and updating will be done simultaneously, so that there will be no delay after the end of the Marking. (only available for Mark→ Trigger)
/MultiInstance	Multiple instances of SAMLight can be used on one PC. The parameters /ActiveCard and /SettingsFile are obligatory for the definition of the setup. Please note that the performance of multiple instances of SAMLight might be reduced when used on one PC. For high performance setups, the use of one PC per SAMLight instance is recommended instead of MultiInstance.
/noclose	Prevents to close SAMLight via GUI (File->Close, X in top right corner of the window etc) or 'Alt + F4'. SAMLight can only be closed by the CCI command ScShutDown() or the Windows task manager.
/resetwindowpos	Resets the position of the SAMLight if the window was moved outside the screen.
/SettingsFile=<*.sam>	With this parameter, it is possible to define the settings file the software is using. The settings file is always stored in the folder <SCAPS>\system\.
/StartupDelay=<sec>	This parameter delays the startup of the application for the given time period sec (in unit seconds). If SAMLight is started automatically with the Windows start, this option may be necessary to ensure that all required drivers are loaded. Using such a delay assures that the application does not try to access the scanner card before it is made available by the operation system. Using this parameter the splash screen appears with no delay so that the user is informed that everything goes well.
/StepperMotionFile	For motion type 8 or 14 a specific stepper configuration file 'StepperMotionFile' can be used instead of sc_motion_stepper_settings.txt. The motion type (8 or 14) has to be defined in sc_motion_settings.txt. The file 'StepperMotionFile' need to be located in the directory <SCAPS>\system\.
/TriggerMode=<0, 1>	This option can be used only if a job was selected for loading using the preceding parameter. If the TriggerMode is set to 1, the application switches to trigger mode automatically after loading that job.
/UseDirectWrite	Use the later DirectWrite Windows API instead of the older Uniscribe Windows API for text layout and glyph rendering as default. It is used,

command line parameter	description
	e.g. to display an international language correctly in SAMLight.
/3D	The application starts in 3D mode if this option is specified and if the appropriate license is available. This option does not overwrite the appropriate settings. If you want to use this option to toggle the program execution mode the auto save option of the general settings should be turned off, else the temporarily enabled 3D mode would be saved.

Table 52: Available command line parameters for SAMLight

Usage Example: This example describes how to create two icons on the Windows desktop, one starting the scanner application using card number 0 in YAG mode, the other starting it using card number 1 in CO2 mode.

Install the two cards: First the two cards with the drivers have to be installed properly.

Create two settings files: Within windows explorer, go into folder <SCAPS>\system\ and make two copies of the existing file sc_light_settings.sam. Rename them to sc_light_settings_yag.sam and sc_light_settings_co2.sam.

Setup: Start <SCAPS>\tools\sc_setup.exe, go to menu HardwareSettings, select the file sc_light_settings_yag.sam, press "Load" and set up the card for YAG mode. Save the settings and repeat this step for the CO2 file.

Create Windows shortcuts: Create two shortcuts of sam_light.exe. Change the name of the shortcuts to "sam_light YAG" and "sam_light CO2". Right click a shortcut and go to Properties.

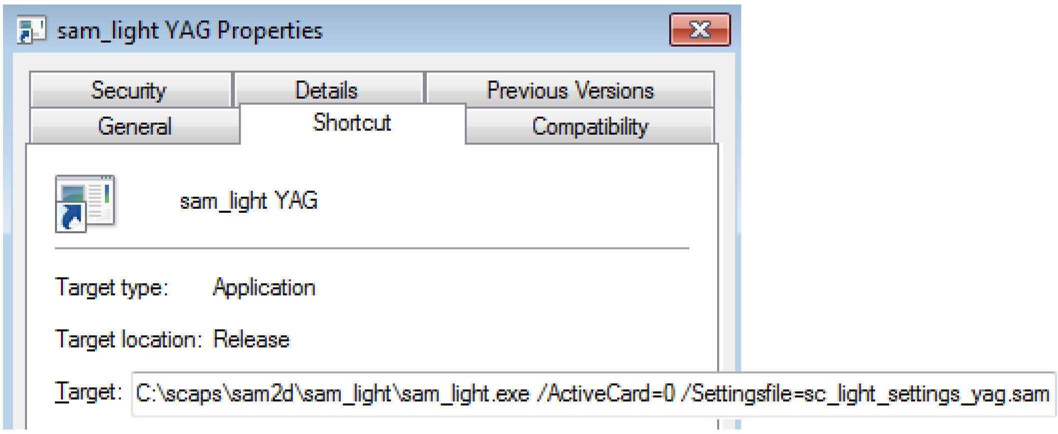


Figure 453: Creating a Shortcut to SAMLight with command line parameters

Set the command line parameters: Inside the property page of the shortcut, define the program arguments. In this example we add the following command line parameters:
`/ActiveCard=0 /Settingsfile=sc_light_settings_yag.sam`

Other settings file: Apply for the "sam_light CO2" shortcut the following command line parameters:
`/ActiveCard=1 /Settingsfile=sc_light_settings_co2.sam`
The entire string in "Target" should be: C:
\scaps\sam2d\samlight\sam_light.exe /ActiveCard=1 /Settingsfile=sc_light_settings_co2.sam

22.3 Customize Program / Language

This chapter might be helpful, if you want to define your own outfit of the application by creating an own application title, a bitmap with your logo and an icon as an application identifier. Another feature explained here is the change of the strings on the windows to provide different languages for the user interface.

22.3.1 Personalize Program

All files to personalize the appearance are stored in the folder *<SCAPS>\system*. The following files can be substituted by personal ones.

sc_light_icon.ico : This is the desktop icon.

sc_light_logo.bmp : This is the startup logo.

sc_light_name.txt : This is the name of the software.

22.3.1.1 Installation of User Data

To install special user data, create a directory with the name *data* in the same directory (of the installation medium), where the installer-exe-file is located.

Valid user data are: (*)

The following settings-, logo-, icon- and help-files:

- sc_light_icon.ico
- sc_light_logo.bmp
- sc_light_name.txt
- sc_light_settings.samsc_settings.sam
- sc_help_sl_english.chm
- sc_resource_settings.sam

Correction files: These are all *.ucf files with corresponding descriptions as *.txt files. A description file called *filename.txt* will be installed only if there exists a correction file called *filename.ucf*, unless the text file is *sc_light_name.txt*.

Resource files: These are *.sam files, the names of which are beginning with *sc_resource*, e.g. *sc_resource_sc_german.sam.* Files in the data directory with other names are not valid and will not be installed.

Controlling the installation of user data: To control the installation of these files, create a text file called *sc_data_info.txt* in the data directory. This file contains a line for each file with the following information :

filename=flag : Filename can be one of the names in **(*)** and where flag can be one of the following values:

ow : Overwrite, the file on the target system will be overwritten with the corresponding file from the data directory

au : Ask user, if the file actually exists on the target system the installer will ask the user to overwrite or not, if the file does not exist on the target system the corresponding file from the data directory will be copied.

no : No overwrite, if the file actually exists on the target system it will not be overwritten, if the file does not exist on the target system the corresponding file from the data directory will be copied.

Files in the data directory which are not listed in *sc_data_info.txt* will be treated as if the no-flag was set. If *sc_data_info.txt* is empty or does not exist then all files in the data directory will be treated as if the no-flag was set.

Example for sc_data_info.txt:

```
sc_light_icon.ico=au
sc_help_sl_english.chm=ow
```

Do not type white spaces in front of the filename or in between filename and "=" or in between "=" and the flag.

Example: Installation of SamLight with a Chinese resource:

create a data directory containing the files:

- sc_resource_sc_chinese.sam
- sc_resource_settings.sam (referring to sc_resource_sc_chinese.sam)
- sc_data_info.txt

where *sc_data_info.txt* contains the lines:

```
sc_resource_sc_chinese.sam=ow
sc_resource_settings.sam=ow
```

22.3.1.2 Customize Laser Names

It is possible to change the Laser Name string in the Mark Property Page and the Pen window captions. Therefore a new text document "override_strings.txt" has to be created in the scaps system folder. Then the user must add LaserName = <Customer Name> to the text document. The entry <Customer Name> will be shown in the Mark Property Page and the Pen window captions.

22.3.2 Customize Language

It is possible to translate almost all strings in SAMLight and redefine appearances of dialogs to generate a SAMLight user interface for your own language and make working at the machine even easier.

This customization of the language can be done with 2 different methods:

- **Resource Editor:** Translation with the *ScResourceManager* which allows to change strings from within the SAMLight GUI. This method is recommended if you need to change only a few strings or if you need to know the exact context of a string.

- **Translation per XML:** Translation using a list of strings (xml file) outside of the SAMLight GUI. This method is recommended if you need to change a lot of strings.

22.3.2.1 Resource Editor

To activate the Resource Edit mode, the following steps are necessary:

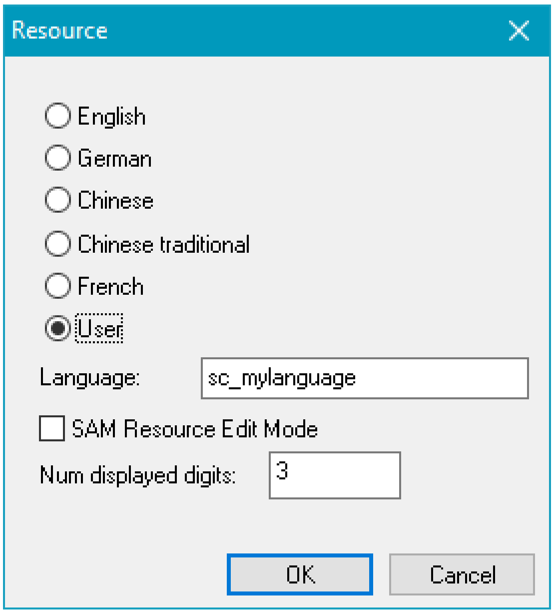


Figure 454: Resource Dialog

1. Close all SAM based applications, check it in the task manager.
2. Start *sc_setup.exe* from folder *<SCAPS>\tools*
3. Click on the menu *Resource*.
4. For Multibyte:
   - o Choose a language
   - o Click ‚*SAM Resource Edit Mode'* to enable the resource edit mode and click ‚OK'.

5. For Unicode:
   - o Choose a language
   - o Click ‚*SAM Resource Edit Mode'* to enable the resource edit mode. and click ‚OK'.

6. Close *sc_setup.exe* and start SAMLight.

7- Do the translations for the user interface, as described in the following.

8. When all translation is done, quit your application and switch off *SAMResourceEditMode* again.

When the SAM based application runs and the *SAMResourceEditMode* is enabled, a *EditResource* button appears on every property page and in every dialog. These buttons are the entry to the *ResourceEditor*. The string in *Language* is used to generate the corresponding resource file. The file containing all user defined resources is located in *<SCAPS>\system* and it is named *sc_resource_mylanguage.sam*. To transfer your resources from one PC to another, you just need to copy this file and to define the language string within *sc_setup.exe*.

Clicking on the *EditResource* button within one dialog shows the the window in figure 455. The strings given in this dialog have to be translated to get a new language appearance.

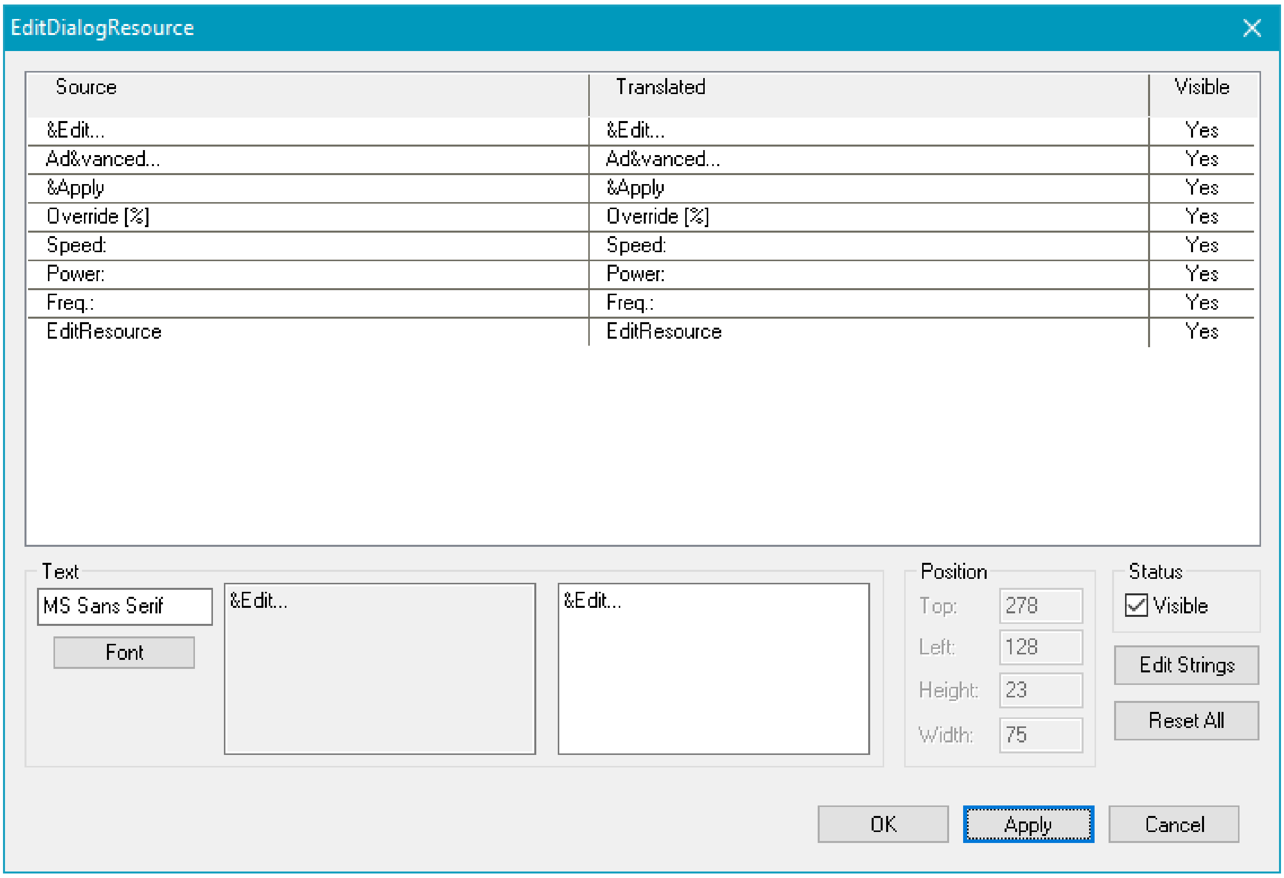


Figure 455: Resource Editor Dialog

List view: On the list view, the user can select the string that will be edited by clicking on it. The list also shows strings of the contents of text boxes. These values do not have to be changed.

Text: The text selected in the view list appears in the *Text* window and can be changed there.

Font: With the button *Font*, it is possible to define a font for the window string. Please make sure that the selected font is also available on the end user system.

Position: The size and the position of the selected window can be changed.

Status: If *Visible* is chosen the selected string will be visible for this language. The status is displayed in the list view.

Edit Strings: This button is the entry to the String Editor for all SAM modules on this PC. Some dialogs have dynamic string setting. For example, the *GeometryPropertyPage* strings are set depending on the type of the selected entity. In these cases, the strings have to be defined in the string editor. For example, the *Rectangle* string is defined in string module *StandardProp, String ID 11*.

Reset All: Imports the default English resources. The current window texts are reset to default English after pressing the *OK* button.

ShiftAll: After pressing *Apply*, all strings of the current window are shifted by the values given in X and Y.

22.3.2.1.1 String Editor

Clicking on the *EditStrings* button within the dialog of the *Resource Editor* shows the following dialog:

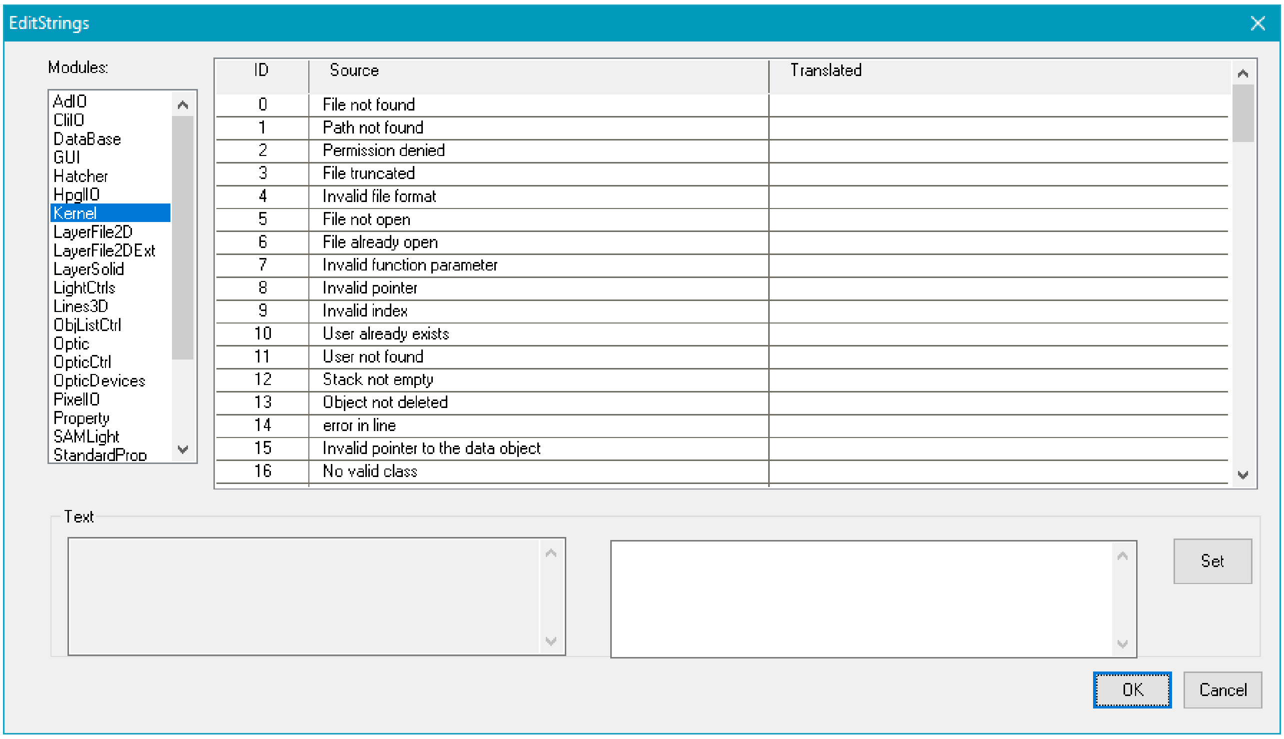


Figure 456: String Editor Dialog

Modules: On the left, there is a list with all activated SAM modules. Every module contains specific strings. To modify the strings, it is necessary to first select the module, second select the string and then change the text.

Set: Clicking *Set*, modifies the corresponding string.

NOTE *The most important strings are inside the Kernel, StandardProp, Optic, OpticCtrl and View2DCtrl modules. It is not possible to modify the menu for the Standard2D application, because this application is delivered with the source code and changes have to be done there. The SAMLight menu can be changed by selecting SAMLight and defining the desired strings. Also for SAMLight it is necessary to modify the LightCtrls string module. The changes of the String Editor are only visible after a restart of SAMLight.*

22.3.2.2 Translation per XML

There is a possibility to use an *.xml table to change the strings in SAMLight. This feature may be beneficial if you are going to translate a big amount of strings in different dialog windows.

1. In the Chapters Creating a new XML-translation and Editing existing XML-translation you will learn how to start translating your source file from the bottom up and how to make changes in your existing translation
2. If you have been using the Resource Editor and want to switch to .XML-translation method, in the Chapter Switching from Resource Editor to XML-translation you will find an instruction on how to do it.
3. SAMLight Updates may contain new functionalities. To be able to translate them into your language, refer to the chapter Resource File Update.

22.3.2.2.1 Creating a new translation

To create a new translation, the steps you need to do are:

1. Define the resource file.

2. Export a list of strings which can be translated. Do the translation.

3. Import the list with the translated strings into SAMLight.

1. Define the resource file:
With every SAMLight version, you get a few resource setting files, which are stored in the directory <*SCAPS*>*system*\ and are named as sc_resource_sc_LANGUAGE. To start translating SAMLight into your language using XML tables, look for a file named sc_resource_sc_default.sam in the system directory and rename it, for example putting in the language you are translating instead of "default". The name of your file may look like sc_resource_sc_mylanguage. This step is important, because every SAMLight update overwrites all standard resource files including the default file, which will be continuously updated by SCAPS with all new functionality. However, your renamed resource file will not be changed.

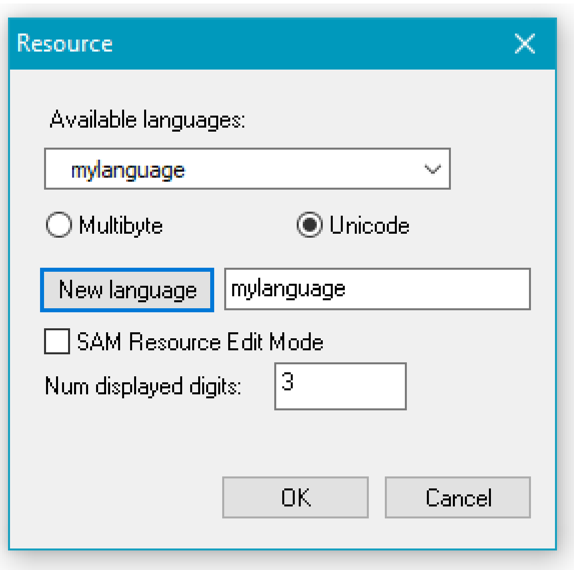


Figure 457: Resource Editor

1. Close all SAM based applications, check it in the task manager.
2. Start *sc_setup.exe* from folder <*SCAPS*>*tools*\.
3. Click on the menu *Resource*.
4. Select *Unicode* and don't choose ‚Available languages' (It doesn't matter which available language you choose).
5. Enter the name of your new resource file e.g. ‚mylanguage'.
6. Click ‚New language'. This creates a new file in folder <*SCAPS*>*system, e.g., \sc_resource_sc_mylanguage_uc.sam'.*
7. Close *sc_setup.exe.*

2. Export a list of strings which can be translated.
Start SAMLight with the command line parameter -export_language (or /export_language) which will generate a file named e.g.,sc_resource_sc_mylanguage_uc.xml' in the folder <SCAPS>\reportfiles, then

close SAMLight. This XML file contains all the strings that are available for the translation. It is strongly recommended to backup this resource-XML file. Then, you can edit the "Translated" and "Visibility" columns using any .xml file redactor (for example Microsoft Excel). **Do not change any other columns!**

Initial Export

By default you will get an .xml file with English strings with an empty translation column. After having filled the translation column, if you want to get an .xml file again with an empty translation column, start SAMLight with two command line parameters: -export_language -initial_export. After the translation is done, go to step 3.

3. Import the list with the translated strings into SAMLight.

After you are done with the translation, save the changes of the sc_resource_sc_mylanguage_uc.xml in the <SAM>\reportfiles repository (don´t forget to make a backup) and start SAMLight with the command line parameter -import_language. SAMLight then ends itself immediately. Start SAMLight again WITHOUT the command line parameter. You will find your newly added translated strings in SAMLight!

The order of strings in your translation file is important. If you copy the "translated"-column somewhere else and paste it back, make sure that you are pasting in the same XML file version as you have used for copying. The reason behind it is, that if new strings are added to the translation, they will be sorted according to the dialog-IDs and not added at the end of the XML-file. So if you use the newest XML file, but paste the "translated"-column from the older one, all translated names may be shifted.

22.3.2.2.2 Editing existing XML-translation

To modify an existing translation, the steps you need to do are:

1. It is highly recommended to backup both .sam and .xml files corresponding to your translation. Please do it before each export or import. Otherwise if anything goes wrong, you might lose your translation!

2. To update your translation (for example in the case you made further changes in the .xml file), replace the old translation .xml-file in <SCAPS>\reportfiles\ with the version you want to use and import this version as explained in the Creating a new translation.

22.3.2.2.3 Switching from Resource Editor to XML-translation

If you're currently using the SAM Resource Edit Mode to translate SAMLight, there's a possibility to switch to .XML-translation method without loosing your previous translations.

1. Make sure that your language is selected in the resource editor. It means that you select your language and can also keep the resource editor activated.
2. Start SAMLight with the command line parameter -export_language (or /export_language) which will generate a file named sc_resource_sc_mylanguage.xml (in Multbyte) or sc_resource_sc_mylanguage_uc.xml (in Unicode) in the folder <SCAPS>\reportfiles. This is the file containing all the strings that are available for the translation, including the strings already translated. It is strongly recommended to backup this resource-XML file. Then, you can edit the "Translated" and "Visibility" columns using any .xml file redactor (for example Microsoft Excel). Do not change any other columns!
3. After you are done with the translation, save the changes of the sc_resources_sc_mylanguage.xml in the <SAM>\reportfiles repository (don´t forget to make a backup) and start SAMLight with the command line parameter -import_language.

To load a new translation file, see Editing existing translation or in the case that your .xml does not contain some strings you need, it may be helpful to update your resource file. You will find the instruction in Resource File Update.

4. If you did new translations in the resource edit mode, you have to -export_language again to update your .XML file. It won't be done automatically.

Pay attention: This method does not work for English under Multibyte!

22.3.2.2.4 Resource File Update

Updating the SAMLight Version

Make sure that you made a backup of sc_resource_sc_mylanguage.SAM and sc_resource_sc_mylanguage.XML before you proceed with updating.

Some SAMLight Updates may contain new strings in the default resource file due to new functionalities or bug fixes. If you want to update your .XML-table with new strings, proceed as follows.

We recommend you to update your source file only in the case that you need to translate some certain strings that are not in your .XML-File yet.

1. Update SAMLight version
2. Create a copy of the updated sc_resource_sc_default.sam and rename the copy according to your language, for example to sc_resource_sc_mylanguage.sam.
3. Make sure that your latest translation-XML file is in the <SAM>\reportfiles directory. Start SAMLight with the parameter -import_language. With this step you integrate your translations in the new sc_resource_sc_mylanguage.sam file. It is important that your latest translation-XML file really contains all translated strings. If you are not sure about this, you can do the export again (before doing step 2.) as described previously.
4. After the import, make sure that your translations were copied to SAMLight by checking some dialogs manually.
5. Close SAMLight and start it again with the command line parameter -export_language. Then, SAMLight will overwrite the current translation-XML with the latest one, containing all previous translations and new strings, if there are any in the new SAMLight version.

It is very important that you first import the old translation into the new SAMLight version. By exporting without importing, you will overwrite your translation-XML with the default .xml, which only contains not translated strings.

22.3.3 Customize Icons

It is possible to replace the icons in SAMLight with your own icons.

To replace the icons, they must be saved with a specific name and in a specific folder.

- The size of the bitmaps must be 16x16 pixels. Big icons can also be used. The size of the bitmaps for the big icons must be 32x32 pixels. They have to be saved in the same folder as the standard bitmaps. In addition, the name of the Big Icon must end with _32, for example the icon for "File Open" is saved as File_Open.bmp for 16x16 pixels and as File_Open_32.bmp for 32x32 pixels.
- The icons must be saved in <SCAPS>\system\images or a subfolder in images.

To use the custom icons, SAMLight must be restarted.

22.3.3.1 Toolbar Icons

Use the color 240-240-240 for transparent background of the bitmap for the toolbar icons.

22.3.3.1.1 File

The folder "main" for the bitmaps for the file toolbar must be saved in the folder "images".

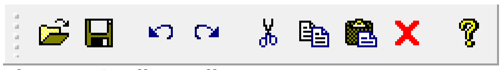


Figure 458: File Toolbar

Bitmap	Name
	File_Open.bmp
	File_Save.bmp
	Undo.bmp
	Redo.bmp
	Cut.bmp
	Copy.bmp
	Paste.bmp
	Delete.bmp
	Help.bmp

Table 53: Bitmaps of File Toolbar

22.3.3.1.2 Camera

The folder "camera" for the bitmaps for the camera toolbar must be saved in the folder "images".

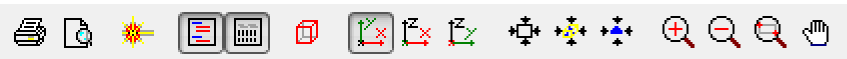
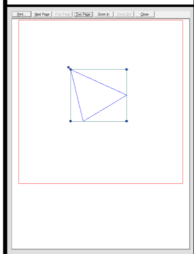


Figure 459: Camera Toolbar

Bitmap	Name
	Print.bmp
	Print_Preview.bmp
	Mark.bmp
	Object_List.bmp
	Property_Pages.bmp
	Job_In_3D_View.bmp
	View_XY.bmp
	View_XZ.bmp
	View_YZ.bmp
	Fit_All.bmp
	Fit_All_Entities.bmp

Bitmap	Name
	Fit_Object.bmp
	Zoom_Plus.bmp
	Zoom_Minus.bmp
	Zoom_Manual.bmp
	Drag.bmp

Table 54: Bitmaps of Camera Toolbar

22.3.3.1.3 View Level

The folder "viewLevel" for the bitmaps for the view level toolbar must be saved in the folder "images".

Figure 460: View Level Toolbar

Bitmap	Name
	Up.bmp
	Down.bmp

Table 55: Bitmaps of View Level Toolbar

22.3.3.1.4 Geometry Object

The folder "GeometricalObjects" for the bitmaps for the geometry object toolbar must be saved in the folder "images".

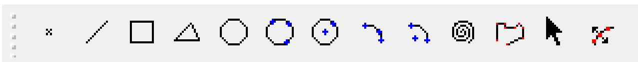


Figure 461: Geometrical Objects Toolbar

Bitmap	Name
	Point.bmp
	Line.bmp
	Rectangle.bmp
	Triangle.bmp
	Circle.bmp
	Circle_three_points.bmp
	Circle_center_radius.bmp
	Arc_three_points.bmp
	Arc_center_begin_end.bmp
	Spiral.bmp
	Polyline.bmp
	Select.bmp
	Select_item.bmp

Table 56: Bitmaps of Geometrical Objects Toolbar

22.3.3.1.5 Functionality Object

The bitmaps for the functionality objects Toolbar must be saved directly in the folder "images".

Figure 462: Functionality Objects Toolbar

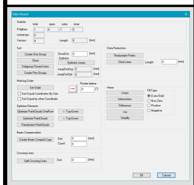
Bitmap	Name
	timer.bmp
	wait_for_input.bmp
	set_output.bmp
	set_override.bmp
	set_executable.bmp
	analog_output.bmp
	SCAPS.ScDateTimePropertyCtrl.bmp
	SCAPS.ScVarEntityPropertyCtrl.bmp
	SCAPS.ScBarCodePropertyCtrlEx.bmp
	SCAPS.ScText2DPropertyCtrl.bmp
	jump.bmp
	motion.bmp
	motion_async.bmp
	mof_offset.bmp
	wait_for_trigger.bmp
	wizard.bmp
	param_finder.bmp

Table 57: Bitmaps of Functionality Objects Toolbar

NOTE

If these icons are replaced with customized icons, the icons in the Entity List of the corresponding entity is replaced as well.

22.3.3.1.6 Align and Spacing

The folder "AlignSpacing" for the bitmaps for the align and spacing toolbar must be saved in the folder "images".

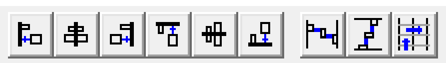


Figure 463: Align and Spacing Toolbar

Bitmap	Name
	Center_Selection.bmp

Bitmap	Name
	Center_Vertical_Selection.bmp
	Center_Horizontal_Selection.bmp
	Align_Left.bmp
	Align_Center_Vertical.bmp
	Align_Right.bmp
	Align_Top.bmp
	Align_Center_Horizontal.bmp
	Align_Bottom.bmp
	Space_Horizontal.bmp
	Space_Vertical.bmp
	Space_Advanced.bmp

Table 58: Bitmaps of Align and Spacing Toolbar

22.3.3.1.7 Extras

The folder "SplitStep" for the bitmaps for the extras toolbar must be saved in the folder "images".

Figure 464: Extras Toolbar

Bitmap	Name
	Splitting_Settings.bmp
	Splitting_Resplit.bmp
	Step_Settings.bmp
	Step_Reset_Pos.bmp
	Bmp_Splitting_Settings.bmp

Table 59: Bitmaps of Extras Toolbar

22.3.3.1.8 3D Surfaces

The folder "3dSurface" for the bitmaps for the 3D surface toolbar must be saved in the folder "images".

Figure 465: 3D Surface Toolbar

Bitmap	Name
	3D_Surface.bmp

Table 60: Bitmaps of 3D Surface Toolbar

22.3.3.1.9 Flash Compatible Mode

The folder "FlashCompatibleMode" for the bitmaps for the flash compatible mode toolbar must be saved in the folder "images".

Figure 466: Flash Compatible Mode Toolbar

Bitmap	Name
	FCM_Not_Checked_Yet.bmp
	FCM_Not_OK.bmp
	FCM_OK.bmp

Table 61: Bitmaps of Flash Compatible Mode Toolbar

22.3.3.1.10 Background Camera

The folder "webcam" for the bitmaps for the background camera toolbar must be saved in the folder "images".

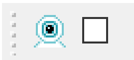


Figure 467: Background Camera Toolbar

Bitmap	Name
	Webcam.bmp

Table 62: Bitmaps of Background Camera Toolbar

22.3.3.1.11 Special Menu

The bitmaps for the special menu toolbar must be saved directly in the folder "images".

Figure 468: Special Menu Toolbar

Bitmap	Name
1	special_menu_1.bmp
2	special_menu_2.bmp
3	special_menu_3.bmp
4	special_menu_4.bmp
5	special_menu_5.bmp
6	special_menu_6.bmp
7	special_menu_7.bmp
8	special_menu_8.bmp
9	special_menu_9.bmp
10	special_menu_10.bmp

Table 63: Special Menu Toolbar

22.3.3.2 Entity List Icons

For each entity type different icons are used in the entity list.

The bitmaps for the entity list must be saved directly in the folder "images".

Bitmap	Name
	analog_output.bmp
	archive_closed.bmp
	archive_parent_list.bmp
	background_archive_closed.bmp
	container.bmp
	container_background.bmp
	container_hidden.bmp
	hatch.bmp
	jump.bmp
	linearray.bmp
	linearrays.bmp
	mof_offset.bmp
	motion.bmp
	motion_async.bmp
	pixelarray.bmp
	pixelarrays.bmp
	point_cloud.bmp
	point_clouds.bmp
	point_end.bmp
	point_middle.bmp
	point_reference.bmp
	point_single.bmp
	point_start.bmp
	polyline.bmp
	polylines.bmp
	set_executable.bmp
	set_output.bmp
	set_override.bmp
	single_line.bmp
	timer.bmp
	wait_for_input.bmp
	wait_for_trigger.bmp

Table 64: Bitmaps of Entity List

The icons of the functionality objects of the entity list corresponds with the entities of the functionality object toolbar.If these icons are replaced with customized icons, the icons in the functionality object toolbar are replaced as well.

22.3.3.3 Overlay Icons

If a special property is assigned to the entity, it is also displayed in the entity list.
The peculiarity of these bitmaps is that it is superimposed on the actual image of the entity. The color white is adopted as transparent.

Beware of wrong placed/scaled images.
If white color (transparent color) is the border color, the pixel in the corners must not be white/transparent.
Otherwise the bitmap is scaled not correct by windows.

It is possible to combine different properties. However, each combination requires its own bitmap.

The bitmaps for the overlay icons must be saved directly in the folder "images".

Bitmap	Name
	OverlayNoEdit.bmp
	OverlayNoEditNoSplit.bmp
	OverlayNoEditRedP.bmp
	OverlayNoEditRedPNoSplit.bmp
	OverlayNoMark.bmp
	OverlayNoMarkNoEdit.bmp
	OverlayNoMarkNoEditNoSplit.bmp
	OverlayNoMarkNoEditRedP.bmp
	OverlayNoMarkNoEditRedPNoSplit.bmp
	OverlayNoMarkNoSplit.bmp
	OverlayNoMarkRedP.bmp
	OverlayNoMarkRedPNoSplit.bmp
	OverlayNoSplit.bmp
	OverlayRedP.bmp
	OverlayRedPNoSplit.bmp

Table 65: Bitmaps of Overlay Icons

22.4 Accelerate SAMLight

For many applications it is important to accelerate the execution speed of SAMLight. The following SAMLight settings show how this can be done:

1st option: Enable the following check boxes at SAMLight menu bar Settings → System → General :

Disable UNDO: Speeds up SAMLight for all entity generating and editing processes as the UNDO and REDO functionality is disabled completely. Depending on the size and complexity of a job, this option can save a considerable amount of memory usage and computing time.

Disable Compression: can help to save big job files on computer systems with low main memory.

Don't update view: can help to save processing time, when entities are updates (date/time and serial number entities, reimport of bitmaps, ...)

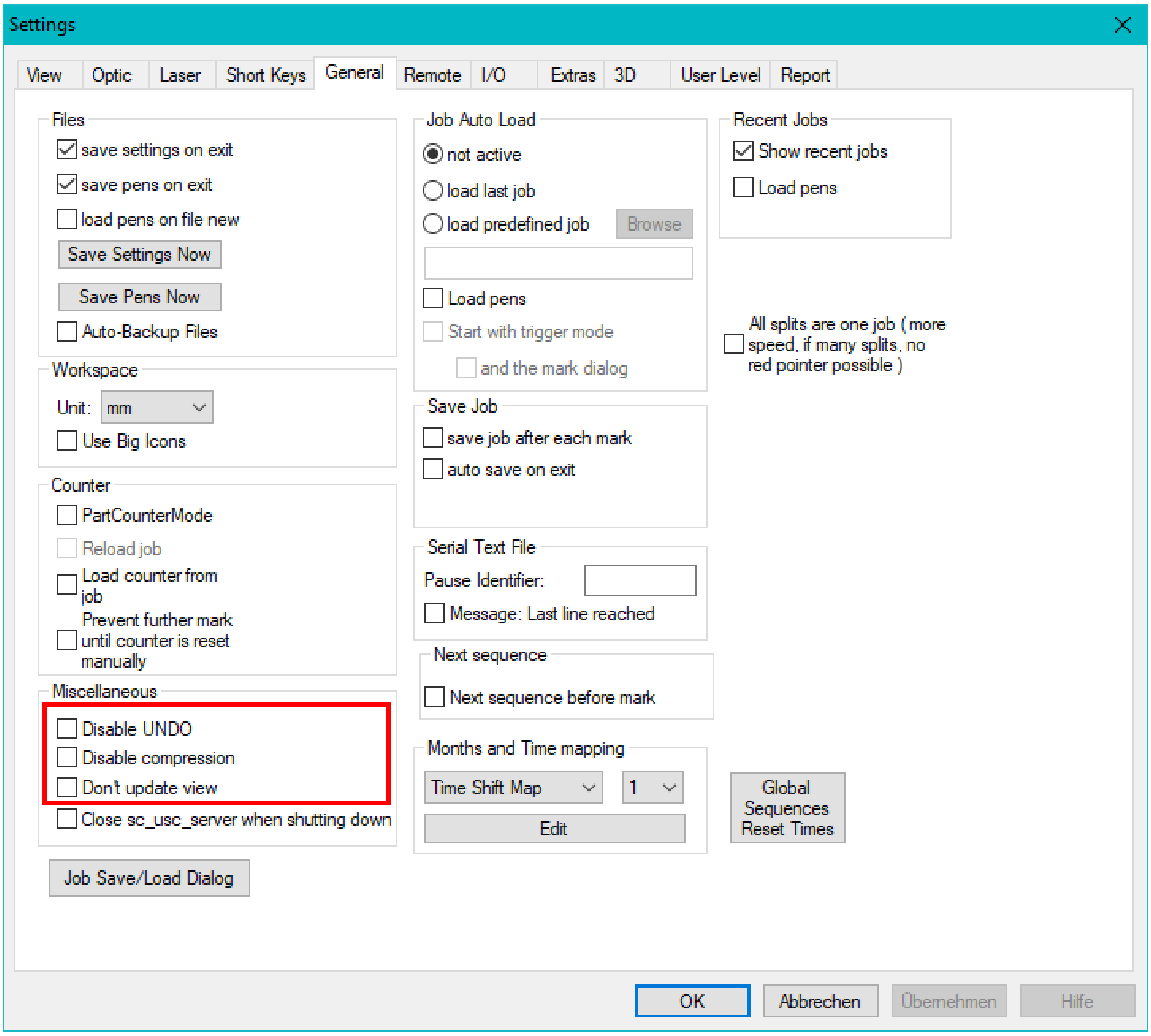


Figure 469: check boxes at General settings for accelerating SAMLight

2nd option: In case the used job file contains text entities like Text2D, serial number, date/time or bar code entities with text the following check box at Text2D property page → Extended settings should be deactivated.

Generate single characters: deactivating this check box can accelerate the generation or all kind of text entities, especially if many text entities are used in a job file.

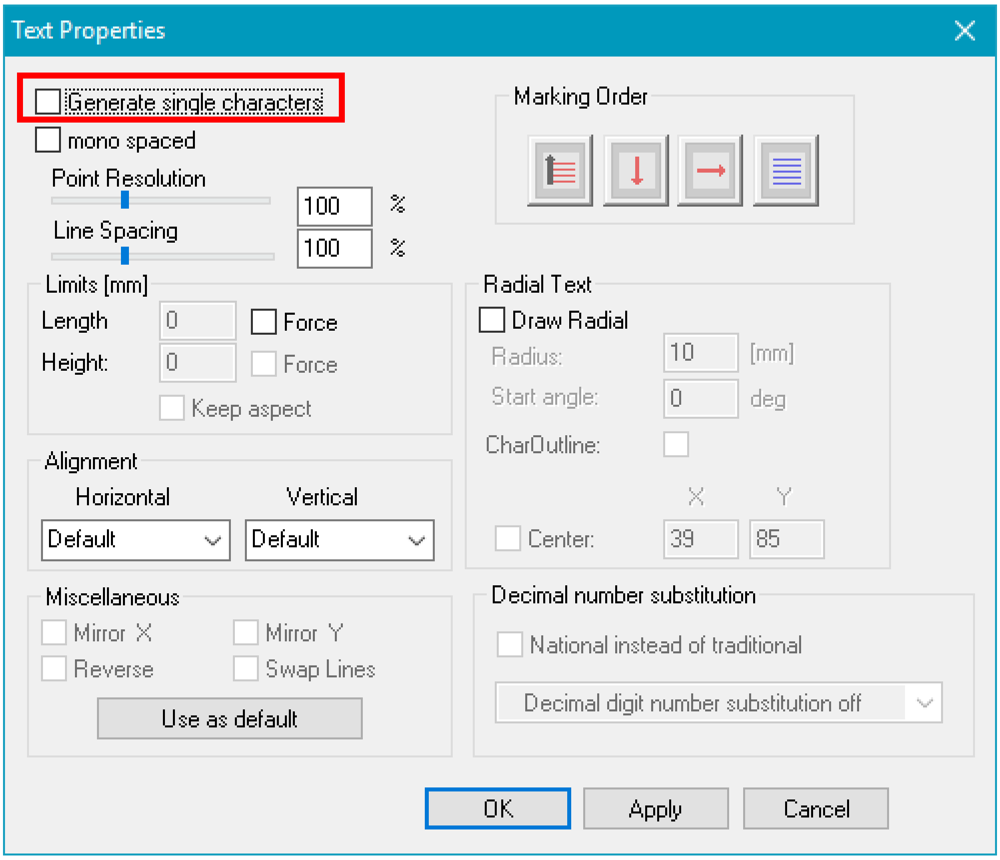


Figure 470: check boxes at Text2D property page Extended settings for accelerating text generation of SAMLight

3rd option: In case SAMLight Client Control programming is used for automating and remote controlling of SAMLight, the SAMLight Client Control commands and constants described at Optimize Performance can be used.

22.5 Generate Dots

For marking a data matrix with a design of dots, the following can be done:

1. Use DataMatrixEx and activate "Generate Dots" in the Extended Dialog.

2. Marking of any other data matrix barcode

    a. Create a data matrix
    b. Calculate the size of a single module of the data matrix
    c. Hatch the data matrix with the parameters (Hatch property page)
        i. Distance = size of one single module
        ii. Start Offset = half of the size of the single module
        iii. Linereduction = very close to the start offset
    d. Enable "Mark Lines as Dots" and set the grid raster values to the module size (Pen property page - Drill)
    e. Enable "Use Geometry" and apply a hatched circle with a size smaller than the module (Pen property page - Drill)
    f. Disable "Contour" for marking only the dots of the data matrix (Pen property page - Misc)

22.6 List of Examples

- Associated Array Copy
- Automate Serialization
- Background Image
- Barcode with variable text and date time object (combined serial number with referenced text)
- Beam Compensation (SAM3D)
- DataMatrixEx
- DataWizard: Combine two Areas
- Date Time Format
- Delays
- Example Loop Count, Beat Count, Beat Offset
- Example for Head2
- Flash Font Codepage
- Hatch Examples
- How to Set Split Lines Manually
- MOTF
- Serial number as barcode
- Special Sequences (SAM3D)
- STL Projection
- Text Alignment
- Upskin / Downskin Handling (SAM3D)
- Using Num Loops (SAM 3D)

- Visualization of Split Options
- Wobble examples

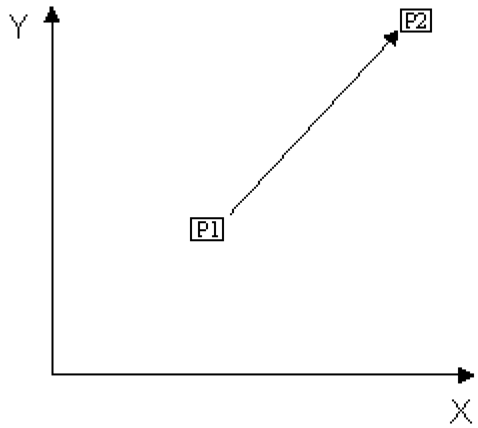
23 Backgrounds

In this chapter miscellaneous theoretical explanations are given for scanner card and program specific functioning.

23.1 Scanner and Laser delays

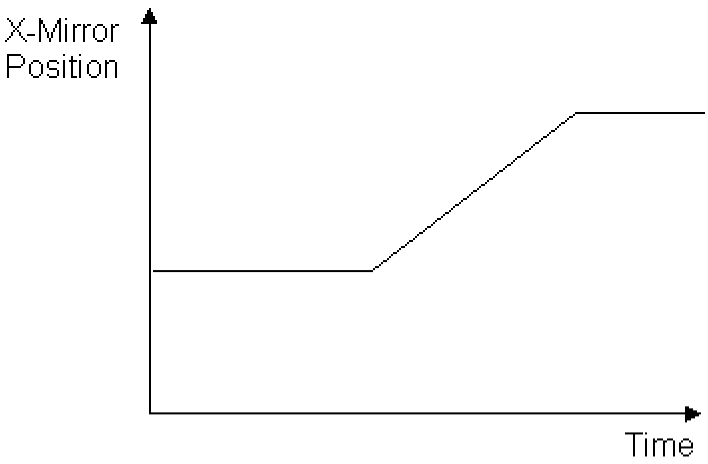
The scanner and laser delays are defined in the laser style parameters dialogs. This chapter gives a short explanation of the delay terms.

Assume that the XY Mirror system is commanded to go from P1 to P2 in XY-plane with a desired speed v.

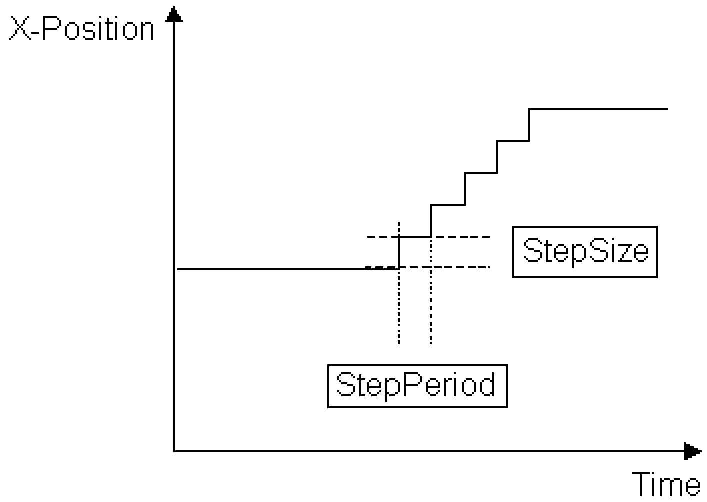


In the following only the X-Mirror is covered, the Y-Mirror is completely analogue.

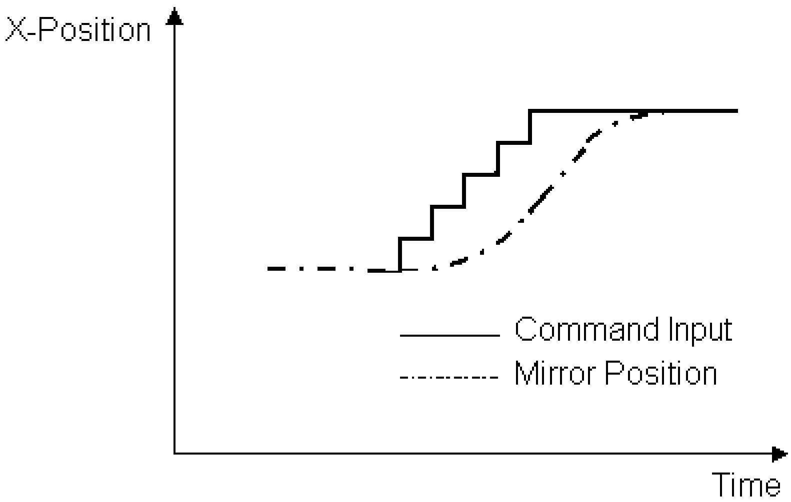
The move command for the X-Mirror looks as follows:



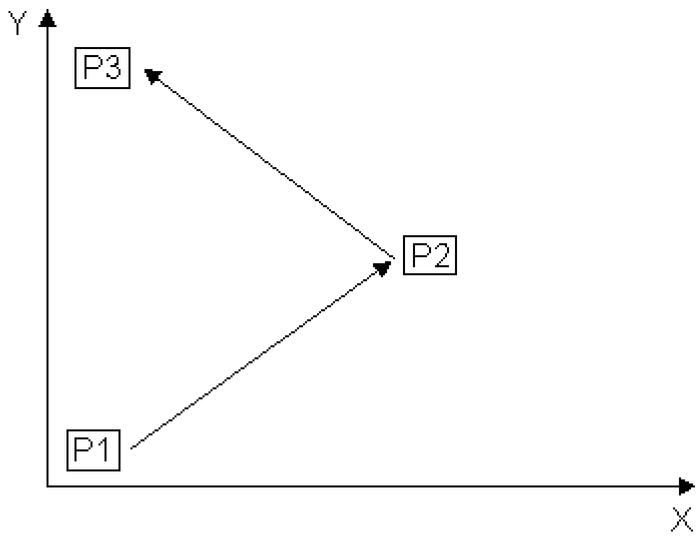
Since the controller card is not able to output values in an arbitrary short time period it has to approximate the desired curve in so called 'microsteps' with a time length of StepPeriod – typically 10 to 50 µs and a position change of StepSize. The requested speed v is the quotient from StepSize/StepPeriod.

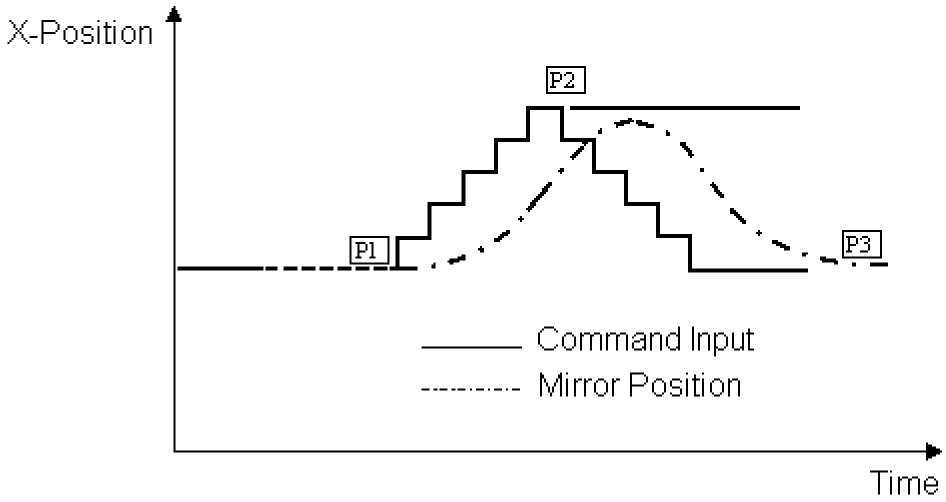


Since the X-Scanner with attached mirror is an inert system it can not follow the controller commands in short time but has some time lag.

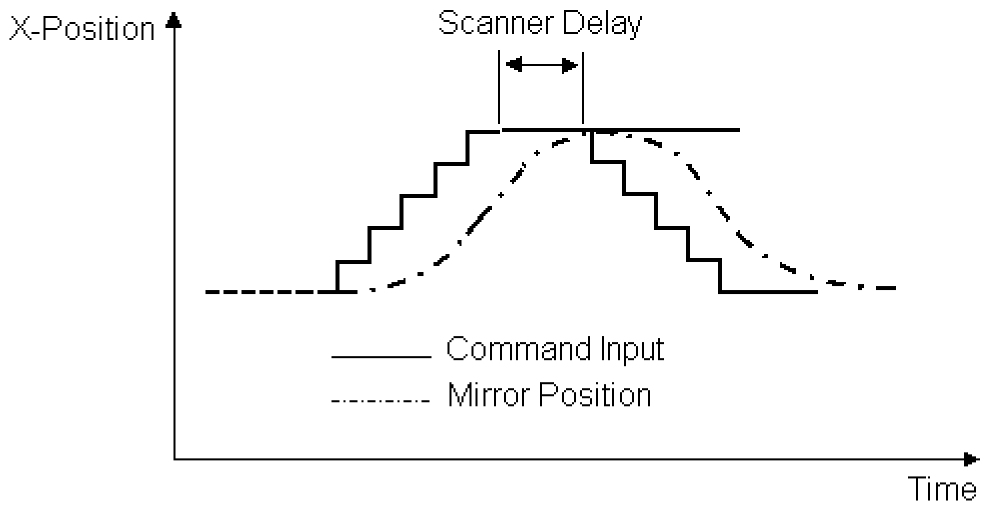


Due to this fact a command input like shown below would lead to the result that the X-Mirror would never reach position P2.





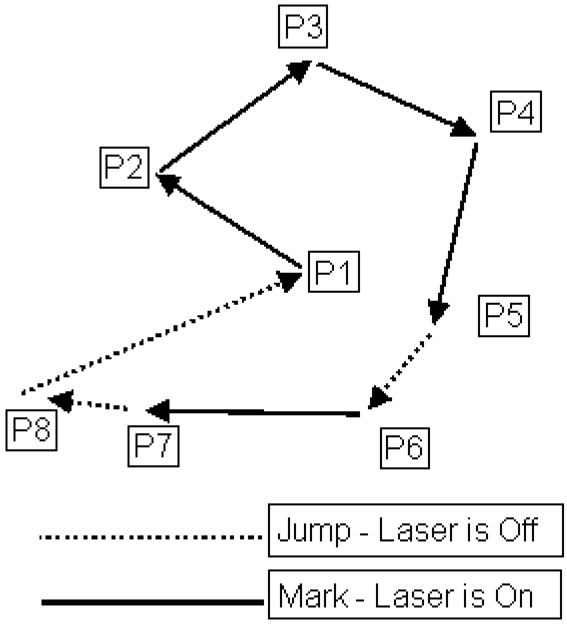
For this reason the controller card inserts a user definable delay between the end of the last vector and the start of the new vector.



There are 3 kinds of scanner delays:

Delay	When are the delays used?
Jump	Points P1,P6 and P8 in the picture below.
Mark	Points P5 and P7 in the picture below.
Poly	Points P2,P3 and P4 in the picture below.

Table 66: Example of scanner delays

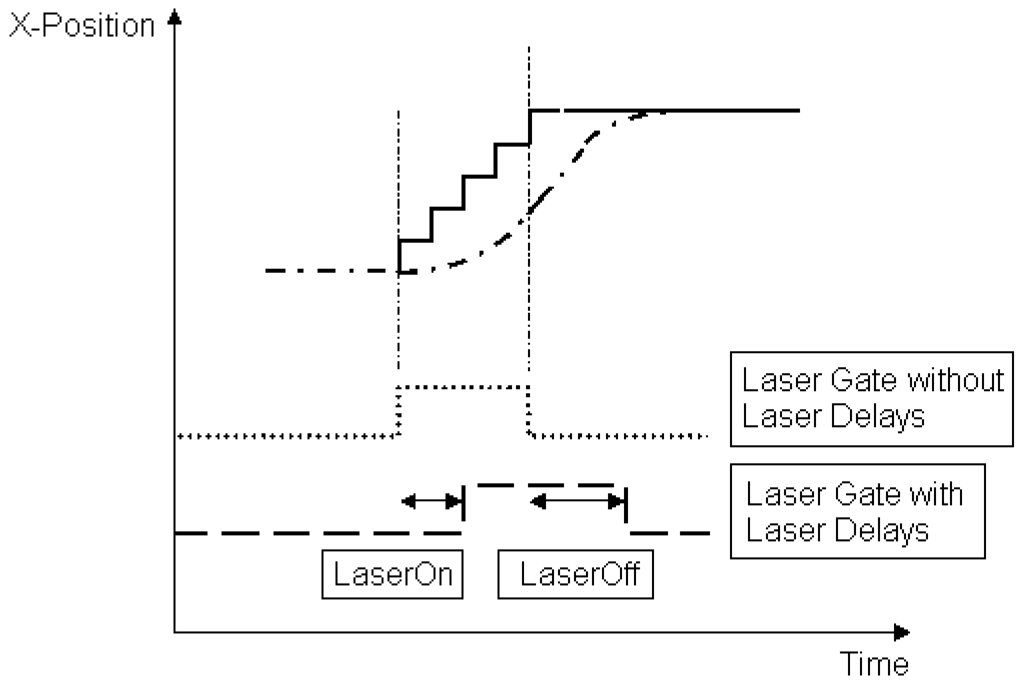


Since the X-Mirror can not accelerate in an arbitrary short time to the requested speed or deaccelerate to zero speed, two additional delays are used to control the delayed switch on and off of the laser gate signal. These are the laser on and the laser off delay.

LaserON delay: Time beginning from the output of the first microstep the controller card waits before it switches on the laser.

LaserOFF delay: Time beginning from the output of the last microstep the controller card waits before it switches off the laser.

Example for laser delays:



23.2 USC Position Transformation

Depending on the scanner controller card and marking mode (SAMLight or standalone) the coordinates stored in SAMLight job file (*.sjf) or standalone job file (*.unf) are transformed like shown in table 67.

	Vector transformation	USC-1 SAMLight	USC-2 SAMLight	USC-2 Flash [OF]	USC-3 SAMLight	USC-3 Flash [OF]	Settings paths and flash notes
PC		SJF	SJF	SJF	SJF	SJF	**SAMLight:** Coordinates saved in SAMLight job file (*.sjf)
PC	CCI ScOpticMatrix	⬇	⬇	✖	⬇	✖	**SAMLight Client Control Interface (CCI) ScOpticMatrix,** adjustable by CCI commands ScOpticMatrixTranslate, ScOpticMatrixRotate, ScOpticMatrixScale and ScOpticMatrixReset
PC	3D Surface mapping	⬇	⬇	✖	⬇	⬇	3D surface **toolbar**
PC	SAMLight lens settings	⬇	⬇	⬇	⬇	⬇	**Settings → Optic**. Order: - XY field center, XY gain, rotation, XY offset (with gain), XY invert, XY flip **sc_setup → z-Offset**
PC	PC Z bit value calculation [O3]	⬇	⬇	✖	✖	✖	**Settings → Optic → Advanced → Correction → Z Correction** **Pen Settings → Misc → Defocus [mm]**
USC				UNF		UNF	**Flash:** Coordinates saved in flash job file (*.unf). SAMLight lens settings are used for UNF generation.
USC	Flash global optic matrix [OF]	✖	✖	⬇	✖	⬇	**Flash:** Standalone only, not used in SAMLight, not accessible via SAMLight. Available Flash Control Interface commands: TMB, SC, RT, TRB
USC	XY head specific optic matrix	✖	⬇	⬇	⬇	⬇	**Settings → Optic → Advanced → Correction** **Flash:** Available Flash Control Interface commands: TRH (InfoView)
USC	MOTF [OM] AnalogIn [AI] Wobble	⬇	⬇	⬇	⬇	⬇	**Settings → Optic → Advanced → Marking on the Fly** **Settings → Optic → Advanced → Analog In** **Pen settings → Scanner → Wobble**
USC	USC Z bit value calculation [O3]	✖	✖	✖	⬇	⬇	**Settings → Optic → Advanced → Correction → Z Correction** **Pen Settings → Misc → Defocus [mm]**
USC	correction file (*.ucf) mapping	⬇	⬇	⬇	⬇	⬇	**Settings → Optic → Advanced → Correction** **Z value** of *.ucf: please refer to [ZV]
USC	FlatLense Defocus [3FL]	✖	⬇	✖	⬇	⬇	**Pen Settings → Misc → Defocus [2^{16} bit / FieldSize]** Add a static value on z output
		XY2-100 signals to scan head					

Table 67: Vector transformations

[OF]: Option Flash required.

[O3]: Option Optic3D required. Optic3D is not possible with USC-2 and standalone mode.

[OM]: Option MOTF required.

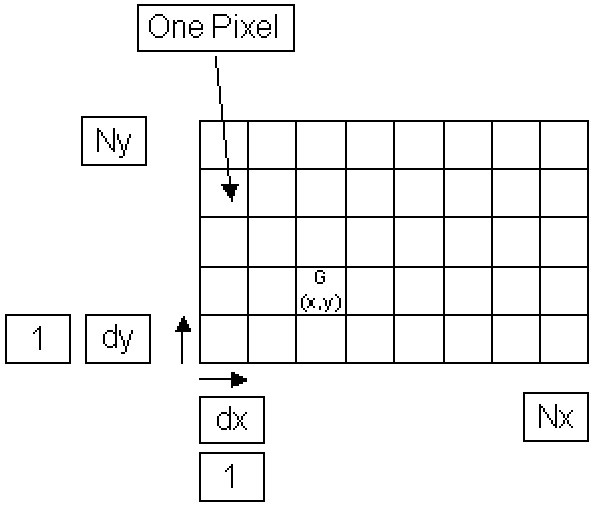
[AI]: AnalogIn is only available for USC-2 / USC-3.

[ZV]: Option FlatLense: Z bit values from UCF file are used. Option Optic3D (+ FlatLense): Z bit values from UCF file are ignored.

[3FL]: Option FlatLense required, if Optic3D license is used, the defocus value is included in Z bit value calculation.

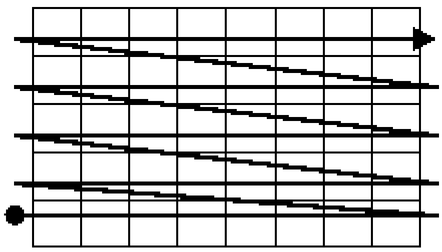
23.3 Pixelmode

This chapter describes how the scanning of bitmaps works with the USC-1 and the RTC3 scanner card. The USC-1and RTC3 card provide a special mode for raster images (Bitmaps).



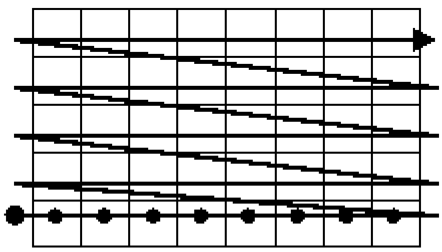
Each pixel inside a bitmap has the same X and Y dimension dx,dy. dx and dy itself may be different. The bitmap consists of Nx pixels in X direction and Ny pixels in Y direction. Each pixel has a Grayvalue G (x,y) from *0* to *1* which is typically transformed to a Grayvalue range from *0* to *255*.

The USC-1 and RTC3 raster modes allow to move the scanner across the bitmap by simultaneously modulating the laser control signal. Within this chapter it is assumed that the scanner movement is performed like shown below.



The scanner starts at the lower left corner, moves over the first line (X-direction) with a defined speed, jumps back to the start of the second line and so on.

Special mode for RTC3 and RTC4 card: If the hardware mode is selected the RTC3 provides two different modes for the movement of the scanner itself. In Mode 0 (shown below for the first line) every pixel position is reached within one scanner step command.



Then the scanner stays at the pixel a certain time before it jumps to the next location. In Mode 1 the scanner moves over the pixels with a constant speed applying many microsteps between the single pixels. In the following scanner mode 1 (constant speed over the pixel line) is assumed.

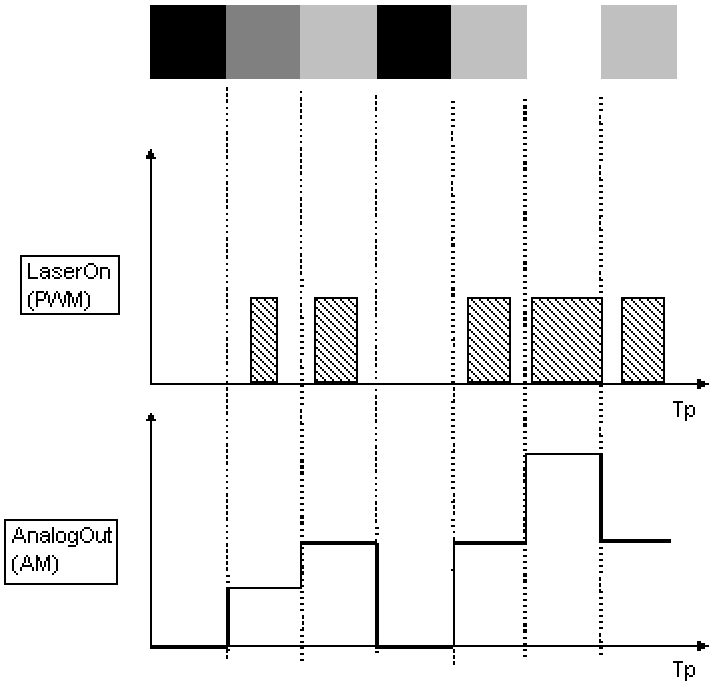
23.3.1 Pulse Modulation

Two modes are provided by the USC-1 and RTC3 to modulate the laser. In general terms they can be described as:

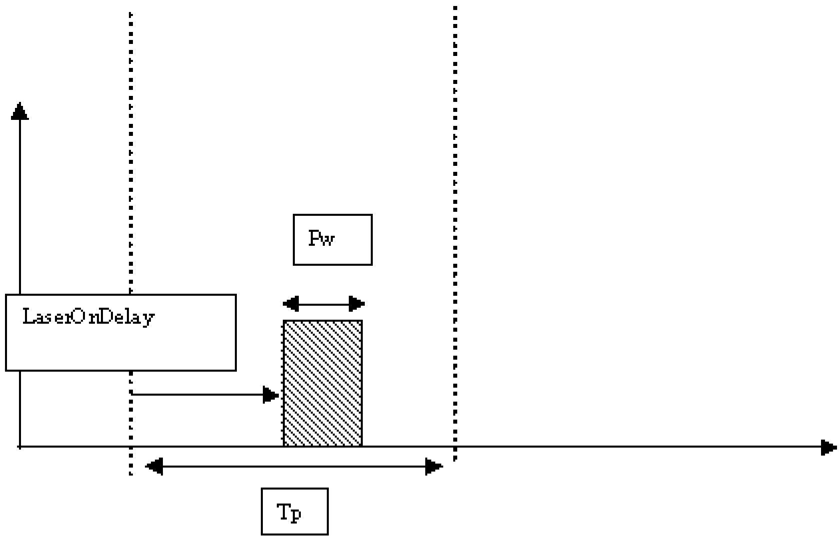
a) Pulse Width Modulation (PWM)
b) Amplitude modulation (AM)

In PWM mode the LaserON Signal is modulated. In AM mode the analog output value of the Laserport is modulated. For a given speed V the time for one pixel is calculated by:

Tp = dx / V



For the scanner card, Tp are multiples of 10 µs. For PWM the pulse with PW inside Tp is calculated according the following formula:



Pw = (Tp - LON - LOFF) * GrayScaleValue

GrayScaleValue	defined from 0 to 1 (normally in 1/256 Steps)
LON	LaserOnDelay
LOFF	LaserOffDelay

LON has the special effect that it offsets the start of the pulse within Tp.

Amplitude Modulation (AM)

For AM the GrayScaleValue will be transformed linearly to the analog output value.

Analogoutput = GrayScaleValue * MaxOutput

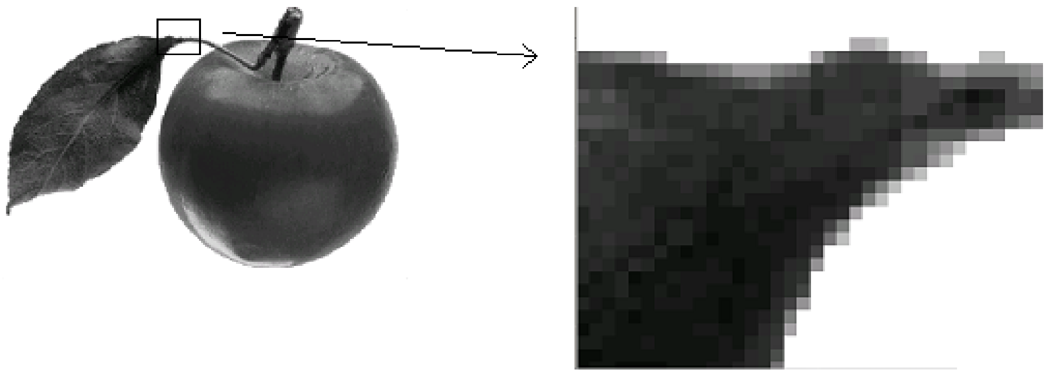
MaxOutput corresponds to maximal achievable Output voltage of the Digital to Analog Converter Output.

NOTE *If one of the following conditions is valid no output takes place:*

- **Tp < 10 OR Tp > 655350**
- **(Tp - LON - LOFF) < = 0**
- **dx == 0 AND dy == 0**

23.3.2 Generating a scanner bitmap

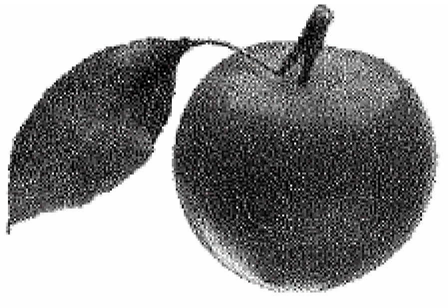
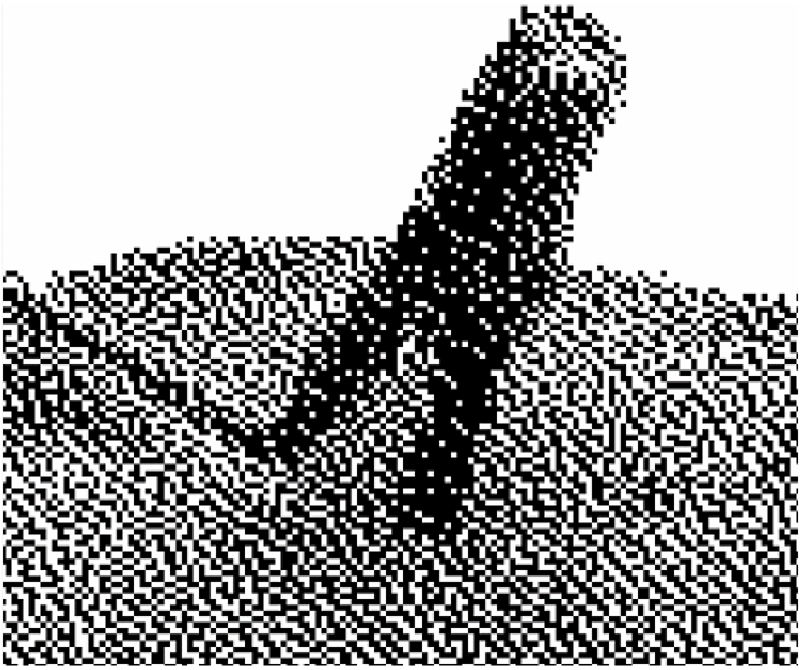
In the following the parameter setting with SAM for the apple.bmp is shown.



The apple.bmp consists of a gray scale bitmap with 255 different possible gray scale values G (x,y). After loading the bitmap the user may scale it according to his needs. Depending on the X and Y dimension and Nx, Ny the bitmap will have a specific dx,dy value for each pixel. To prepare the bitmap for the scanner output the user has to generate a so called scanner bitmap out of the original bitmap. This is done with the help of the Bitmap property page.

For an original Gray Scale bitmap there are two possibilities to generate a scanner bitmap:

a) Error diffusion method:

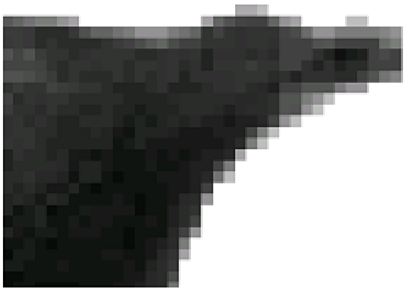



With this method the gray scale values will be approximated by specific placements of black and white pixels to give the impression of gray values. This is a similar method as with a Black/White LaserJet.

b) GrayScale method:

With this method the gray scale values will be kept when transforming them to the scanner bitmap.

Dither step: For both methods a so called *Dither step* parameter can be selected which defines the dx and dy value (which are both equal to Dither step) for each pixel inside the scanner bitmap. The original pixel number Nx,Ny will be changed to new ones Nx=DX/DitherStep with DX as the X Dimension of the bitmap, and Ny=DY/Ditherstep respectively. With this a gray scale bitmap can be transformed into larger pixel size, for example:



23.4 Licenses

the following open source libraries and tools are used in SCAPS products (SAMLight, SAM library, USC and DSD):

Library or tool	License
Clipper	Boost Software License Version 1.0
cpp-httplib	MIT license
CRC++	CRC++ license
Crypto++	Crypto++ license
Cximage	Cximage license
CXXOpts	CXXOpts license
DIME	DIME license
DSPFilters	DSPFilters license
Eigen Library	Mozilla Public License Version 2.0
fatfs	fatfs license
fmt	fmt license
json	MIT license
libaio	GNU Lesser General Public License Version 2.1
libusb	GNU Lesser General Public License Version 2.1
NLopt (without Luksan)	NLopt (without Luksan) license
NSIS	NSIS license
type_safe	MIT license
units	MIT license
zlib	zlib license

Figure 471: Licenses used in SCAPS products (SAMLight, SAM library, USC and DSD)

23.4.1 Boost Software License Version 1.0

```
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:
```

23.4.2 CRC++ license

23.4.3 Crypto++ license

I would like to thank the following authors for placing their works into
the public domain:

Joan Daemen - 3way.cpp
Leonard Janke - cast.cpp, seal.cpp
Steve Reid - cast.cpp
Phil Karn - des.cpp
Andrew M. Kuchling - md2.cpp, md4.cpp
Colin Plumb - md5.cpp
Seal Woods - rc6.cpp
Chris Morgan - rijndael.cpp
Paulo Baretto - rijndael.cpp, skipjack.cpp, square.cpp
Richard De Moliner - safer.cpp

Matthew Skala - twofish.cpp
Kevin Springle - camellia.cpp, shacal2.cpp, ttmac.cpp, whrlpool.cpp, ripemd.cpp
Ronny Van Keer - sha3.cpp

The Crypto++ Library (as a compilation) is currently licensed under the Boost Software License 1.0 (http://www.boost.org/users/license.html).

23.4.4 Cximage license

This copy of the CxImage notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file ximage.h that is included in the CxImage distribution, the latter shall prevail.

If you modify CxImage you may insert additional notices immediately following this sentence.

23.4.5 CXXOpts license

23.4.6 DIME license

23.4.7 DSPFilters license

23.4.8 fatfs license

23.4.9 fmt license

23.4.10 GNU Lesser General Public License Version 2.1

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the

```
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions.  You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

  11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all.  For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices.  Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

  12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded.  In such case, this License incorporates the limitation as if
written in the body of this License.

  13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number.  If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation.  If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

  14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission.  For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this.  Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

                            NO WARRANTY

  15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
```

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

23.4.11 MIT license

MIT License

Copyright (c)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

```
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

23.4.12 Mozilla Public License Version 2.0

```
Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to
    the creation of, or owns Covered Software.

1.2. "Contributor Version"
    means the combination of the Contributions of others (if any) used
    by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.

1.4. "Covered Software"
    means Source Code Form to which the initial Contributor has attached
    the notice in Exhibit A, the Executable Form of such Source Code
    Form, and Modifications of such Source Code Form, in each case
    including portions thereof.

1.5. "Incompatible With Secondary Licenses"
    means

    (a) that the initial Contributor has attached the notice described
        in Exhibit B to the Covered Software; or

    (b) that the Covered Software was made available under the terms of
        version 1.1 or earlier of the License, but not also under the
        terms of a Secondary License.

1.6. "Executable Form"
    means any form of the work other than Source Code Form.

1.7. "Larger Work"
    means a work that combines Covered Software with other material, in
    a separate file or files, that is not Covered Software.

1.8. "License"
    means this document.

1.9. "Licensable"
    means having the right to grant, to the maximum extent possible,
    whether at the time of the initial grant or subsequently, any and
    all of the rights conveyed by this License.

1.10. "Modifications"
    means any of the following:

    (a) any file in Source Code Form that results from an addition to,
        deletion from, or modification of the contents of Covered
        Software; or

    (b) any new file in Source Code Form that contains any Covered
```

```
        Software.

1.11. "Patent Claims" of a Contributor
    means any patent claim(s), including without limitation, method,
    process, and apparatus claims, in any patent Licensable by such
    Contributor that would be infringed, but for the grant of the
    License, by the making, using, selling, offering for sale, having
    made, import, or transfer of either its Contributions or its
    Contributor Version.

1.12. "Secondary License"
    means either the GNU General Public License, Version 2.0, the GNU
    Lesser General Public License, Version 2.1, the GNU Affero General
    Public License, Version 3.0, or any later versions of those
    licenses.

1.13. "Source Code Form"
    means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
    means an individual or a legal entity exercising rights under this
    License. For legal entities, "You" includes any entity that
    controls, is controlled by, or is under common control with You. For
    purposes of this definition, "control" means (a) the power, direct
    or indirect, to cause the direction or management of such entity,
    whether by contract or otherwise, or (b) ownership of more than
    fifty percent (50%) of the outstanding shares or beneficial
    ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
    Licensable by such Contributor to use, reproduce, make available,
    modify, display, perform, distribute, and otherwise exploit its
    Contributions, either on an unmodified basis, with Modifications, or
    as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
    for sale, have made, import, and otherwise transfer either its
    Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License.
Notwithstanding Section 2.1(b) above, no patent license is granted by a
Contributor:

(a) for any code that a Contributor has removed from Covered Software;
    or

(b) for infringements caused by: (i) Your and any other third party's
    modifications of Covered Software, or (ii) the combination of its
    Contributions with other software (except as part of its Contributor
    Version); or

(c) under Patent Claims infringed by Covered Software in the absence of
    its Contributions.

This License does not grant any rights in the trademarks, service marks,
or logos of any Contributor (except as may be necessary to comply with
```

```
the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to
distribute the Covered Software under a subsequent version of this
License (see Section 10.2) or under the terms of a Secondary License (if
permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its
Contributions are its original creation(s) or it has sufficient rights
to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under
applicable copyright doctrines of fair use, fair dealing, or other
equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted
in Section 2.1.

3. Responsibilities
-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any
Modifications that You create or to which You contribute, must be under
the terms of this License. You must inform recipients that the Source
Code Form of the Covered Software is governed by the terms of this
License, and how they can obtain a copy of this License. You may not
attempt to alter or restrict the recipients' rights in the Source Code
Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code
    Form, as described in Section 3.1, and You must inform recipients of
    the Executable Form how they can obtain a copy of such Source Code
    Form by reasonable means in a timely manner, at a charge no more
    than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this
    License, or sublicense it under different terms, provided that the
    license for the Executable Form does not attempt to limit or alter
    the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice,
provided that You also comply with the requirements of this License for
the Covered Software. If the Larger Work is a combination of Covered
Software with a work governed by one or more Secondary Licenses, and the
Covered Software is not Incompatible With Secondary Licenses, this
License permits You to additionally distribute such Covered Software
under the terms of such Secondary License(s), so that the recipient of
the Larger Work may, at their option, further distribute the Covered
Software under the terms of either this License or such Secondary
License(s).

3.4. Notices

You may not remove or alter the substance of any license notices
(including copyright notices, patent notices, disclaimers of warranty,
or limitations of liability) contained within the Source Code Form of
the Covered Software, except that You may alter any license notices to
the extent required to remedy known factual inaccuracies.
```

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the      *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.        *
*                                                                      *
************************************************************************
```

```
************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************
```

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the

```
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

  This Source Code Form is subject to the terms of the Mozilla Public
  License, v. 2.0. If a copy of the MPL was not distributed with this
  file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

  This Source Code Form is "Incompatible With Secondary Licenses", as
  defined by the Mozilla Public License, v. 2.0.
```

23.4.13 NLopt (without Luksan) license

```
NLopt combines several free/open-source nonlinear optimization
libraries by various authors.  See the COPYING, COPYRIGHT, and README
files in the subdirectories for the original copyright and licensing
information of these packages.

The compiled NLopt library, i.e. the combined work of all of the
included optimization routines, is licensed under the conjunction of
all of these licensing terms.  Currently, the most restrictive terms
are for the code in the "luksan" directory, which is licensed under
the GNU Lesser General Public License (GNU LGPL), version 2.1 or
later (see luksan/COPYRIGHT).

That means that the compiled NLopt library is governed by the terms of
the LGPL.

---------------------------------------------------------------------------

Other portions of NLopt, including any modifications to the abovementioned
packages, are licensed under the standard "MIT License:"

Copyright (c) 2007-2011 Massachusetts Institute of Technology

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

23.4.14 NSIS license

Copyright

Copyright (C) 1999-2019 Contributors

More detailed copyright information can be found in the individual source code files.
Applicable licenses

All NSIS source code, plug-ins, documentation, examples, header files and graphics, with the exception of the compression modules and where otherwise noted, are licensed under the zlib/libpng license.
The zlib compression module for NSIS is licensed under the zlib/libpng license.
The bzip2 compression module for NSIS is licensed under the bzip2 license.
The lzma compression module for NSIS is licensed under the Common Public License version 1.0.

zlib/libpng license

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
This notice may not be removed or altered from any source distribution.

bzip2 license

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, Cambridge, UK.

jseward@acm.org
Common Public License version 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Special exception for LZMA compression module

Igor Pavlov and Amir Szekely, the authors of the LZMA compression module for NSIS, expressly permit you to statically or dynamically link your code (or bind by name) to the files from the LZMA compression module for NSIS without subjecting your linked code to the terms of the Common Public license version 1.0. Any modifications or additions to files from the LZMA compression module for NSIS, however, are subject to the terms of the Common Public License version 1.0.

23.4.15 zlib license

```
/* zlib.h -- interface of the 'zlib' general purpose compression library
  version 1.2.11, January 15th, 2017

  Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

  This software is provided 'as-is', without any express or implied
  warranty.  In no event will the authors be held liable for any damages
  arising from the use of this software.

  Permission is granted to anyone to use this software for any purpose,
  including commercial applications, and to alter it and redistribute it
  freely, subject to the following restrictions:

  1. The origin of this software must not be misrepresented; you must not
     claim that you wrote the original software. If you use this software
     in a product, an acknowledgment in the product documentation would be
     appreciated but is not required.
  2. Altered source versions must be plainly marked as such, and must not be
     misrepresented as being the original software.
  3. This notice may not be removed or altered from any source distribution.

  Jean-loup Gailly        Mark Adler
  jloup@gzip.org          madler@alumni.caltech.edu

*/
```

Index