RepliSLS3D

User manual for 3D SLS/SLM Printer Software

Version 2.1

Table Of Contents

1 Copyright

This document is © by HALaser Systems GmbH. RepliSLS3D, OpenAPC, ControlRoom and RepliSLS3D are registered trademarks.

Other software described here that not belong to the OpenAPC software package is © and TM by their respective owners.

Fedora, RedHat, RedHat Enterprise Linux, RHEL are copyright / trademarks / legal trademarks of Red Hat Inc.

Linux is a trademark / legal trademark of Linus Torvalds.

Ubuntu is copyright / trademark / legal trademark of Canonical.

Microsoft, Windows, the Windows-logo are copyright / trademarks / legal trademarks of Microsoft Corporation.

Panasonic and Minas are copyright / trademarks / legal trademarks of Matsushita.

MDrive and MDrive+ are copyright / trademarks / legal trademarks of Schneider Electronic.

SCAPS, SAM, CCI, FEB and USC are copyright / trademarks / legal trademarks of SCAPS GmbH.

SCANLAB, RTC, RTC3, RTC4, RTC5, RTC6, RTCscanalone and others are copyright / trademarks / legal trademarks of SCANLAB AG.

Raylase and others are copyright / trademarks / legal trademarks of Raylase AG.

ETH6608, Sintec and others are copyright / trademarks / legal trademarks of Sintec Optronics.

PSC, PSC140P, 3rdEye and others are copyright / trademarks / legal trademarks of 3rdEye.

Coherent, Avia and others are copyright / trademarks / legal trademarks of Coherent Inc.

SPI and others are copyright / trademarks / legal trademarks of SPI Lasers Ltd.

Sill and others are copyright / trademarks / legal trademarks of Sill Optics GmbH & Co. KG.

All other trademarks mentioned in this document are trademarks or registered trademarks by their respective owners.

All rights not expressly granted herein are reserved.

2 Legal Issues

LICENSE AGREEMENT

The RepliSLS3D software package that is made available (the "Software") is owned by HALaser Systems GmbH (the "Vendor"). This Software is protected by copyright laws, and is being made available solely for use by you in accordance with the following terms and conditions. Any use, reproduction or redistribution of the Software that is not in accordance with this Software License Agreement is expressly prohibited by law, and may result in civil and criminal penalties.

SOFTWARE LICENSE AGREEMENT

THE VENDOR IS WILLING TO LICENSE THIS SOFTWARE TO YOU ONLY ON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS LICENSE AGREEMENT. This is a legal agreement between you (either an individual end-user or an entity) and the Vendor ("Agreement"). By using this software, you are agreeing to be bound by the terms and conditions of this Agreement. If you do not agree to the terms and conditions of this Agreement, promptly return the software and other items that are part of this product in their original package with your sales receipt to your point of purchase for a full refund, or if you have downloaded this software from the Vendors web site, then you must stop using the software and destroy any copies of the software in your possession or control.

1. Grant of Agreement.

Subject to the terms and conditions of this Agreement, the Vendor and its suppliers grant to you a non-exclusive license to use one licensed copy of the Software and any documentation accompanying this Agreement on one computer. No other rights are granted. The Software is in use if it is loaded on the computer's permanent or temporary memory. For backup purposes only, you may make one copy of the Software. You must include on the backup copy all copyright and other notices included on the Software as supplied by the Vendor. Installation on a network server for the sole purpose of your internal distribution of the Software is permitted only if you have purchased an individual Software license for each networked computer to which the Software is distributed.

2. Restrictions.

The Software may contain copyrighted material, trade secrets, and other proprietary materials of the Vendor and its licensor's. You agree that in order to protect those proprietary materials, except as expressly permitted by applicable law, neither you nor a third party acting on your behalf will:
(I)   decompile, disassemble or reverse engineer the Software;
(II)   modify or create derivative works of the Software;
(III)   transmit the Software, in whole or in part, over the Internet or other network; or
(IV)   sell, distribute, rent, lease, sub license or otherwise transfer the Software to a third party; that does not influence your right to sell the Software together with additional equipment that is controlled by this Software

3. Ownership.

The Software is licensed, not sold, to you for use only under the terms and conditions of this Agreement, and the Vendor reserves all rights not expressly granted to you in this Agreement. The Vendor and/or its licensor's retain title to the Software, and all intellectual property rights therein.

4. Termination.

This Agreement is effective until terminated. Upon any violation of any of the provisions of this Agreement, rights to use the Software shall automatically terminate and the Software must be returned to the Vendor or all copies of the Software must be destroyed. You may also terminate this Agreement at any time by destroying all copies of the Software in your possession or control. If the Vendor makes a request via public announcement or press release to stop using the copies of the Software, you will comply immediately with this request. The provisions of paragraphs 3, 7, 8 and 12 will survive any termination of this Agreement.

5. Limited Product Warranty.

The Vendor warrants to you that the Software will substantially conform to its published documentation and the media containing the Software shall be free from defects in material, each for a period of thirty (30) days from the date of purchase. The Vendors limited warranty is non-transferable and is limited to the original purchaser. This warranty gives you specific legal rights, and you may also have other rights which vary under local laws.

6. Remedies.

The Vendors entire liability and your exclusive remedy for any breach of warranty shall be, at the Vendors option, to:
(a) repair or replace the Software or media, provided that the Software or media is returned to the point of purchase or such other place as the Vendor may direct, with a copy of the sales receipt, or
(b) refund the price paid.
Any replacement Software or media will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. These remedies are void if failure of the Software or media has resulted from accident, abuse, or misapplication.

7. DISCLAIMER OF WARRANTY.

THE WARRANTIES EXPRESSLY SET FORTH IN THIS AGREEMENT REPLACE ALL OTHER WAR-RANTIES. THE VENDOR AND ITS SUPPLIERS EXPRESSLY DISCLAIM ALL OTHER WARRANTIES, IN-CLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD-PARTY RIGHTS WITH RESPECT TO THE SOFTWARE OR MEDIA, AND ANY WARRANTIES OF NON-INTERFERENCE OR ACCURACY OF INFORMATIONAL CONTENT. NO DEALER, AGENT, OR EMPLOYEE OF THE VENDOR IS AUTHORIZED TO MAKE ANY MODIFICATION, EXTENSION, OR ADDITION TO THIS WARRANTY.

Some jurisdictions do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you.

8. LIMITATION OF LIABILITY.

IN NO EVENT WILL THE VENDOR OR ITS SUPPLIERS BE LIABLE FOR ANY COSTS OF PROCURE-MENT OF SUBSTITUTE PRODUCTS OR SERVICES, LOST PROFITS, LOSS OF INFORMATION OR DATA, OR ANY OTHER SPECIAL, INDIRECT, CONSEQUENTIAL, OR INCIDENTAL DAMAGES ARISING IN ANY WAY OUT OF THE SALE OF, USE OF, OR INABILITY TO USE ANY VENDORS PRODUCT OR SERVICE, EVEN IF THE VENDOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO CASE SHALL THE VENDOR ITS SUPPLIERS' TOTAL LIABILITY EXCEED THE ACTUAL MONEY PAID FOR THE VENDORS PRODUCT OR SERVICE GIVING RISE TO THE LIABILITY.

Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you. The above limitations will not apply in case of personal injury where and to the extent that applicable law requires such liability.

9. Export Law Assurances.

You agree and certify that neither the Software nor any other technical data received from the Vendor will be exported outside the Federal Republic of Germany except as authorized and as permitted by the laws and regulations of the Federal Republic of Germany. If you have rightfully obtained the software outside of the Federal Republic of Germany, you agree that you will not re-export the Software nor any other technical data received from the Vendor, except as permitted by the laws and regulations of the Federal Republic of Germany and the laws and regulations of the jurisdiction in which you obtained the Software.

10. Agents and Third Party Purchasers.

If you are acquiring the Software on behalf of another person or entity, you represent and warrant that you have the authority to bind the party or entity for which you are acquiring the Software to the terms and conditions of this Agreement.

11. General Terms and Conditions.

This Agreement will be governed by and construed in accordance with the laws of the Federal Republic of Germany, without regard to or application of its choice of law rules or principles. If for any reason a court of competent jurisdiction finds any provision of this Agreement, or portion thereof, to be unenforceable, that provision of the Agreement shall be enforced to the maximum extent permissible so as to affect the intent of the parties, and the remainder of this Agreement shall continue in full force and effect. This Agreement constitutes the entire agreement between the parties with respect to the use of the Software and supersedes all prior or contemporaneous understandings, communications or agreements, written or oral, regarding such subject matter. The Vendor may, in its sole discretion, modify portions of this Agreement at any time. The Vendor may notify you of any changes by posting notice of such modifications on the Vendors web site(s) or sending notice via e-mail, postal mail or other means. Your continued use of the Software following notice of such modifications shall be deemed to be your acceptance of any such modifications to the Agreement. If you do not agree to any such modifications, you must immediately stop using the Software and destroy all copies of the Software in your possession or control.

12. The Software is protected by Federal Republic of Germany copyright law and international treaty. Unauthorized reproduction or distribution of the Software is subject to civil and criminal penalties.

NOTICE SPECIFIC TO INFORMATION AVAILABLE ON THE WEBSITE OF THE VENDOR:

The information provided is for informational purposes only and is subject to change without notice. It is provided "AS IS" and without any warranty. Any risk arising out of the use of the information on this website shall remain with the reader. IN NO EVENT SHALL THE VENDOR BE LIABLE FOR ANY INDIRECT, CONSEQUENTIAL, INCIDENTAL, SPECIAL, PUNITIVE OR OTHER DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF BUSINESS INFORMATION), ARISING OUT OF OR IN CONNECTION WITH THE AVAILABILITY, PERFORMANCE OR USE OF THE INFORMATION CONTAINED ON THIS WEBSITE, EVEN IF THE VENDOR HAS BEEN ADVISED OF THE POSSIBILITIES OF SUCH DAMAGES.

LINKS TO OTHER WEB SITES

Links to other web sites are available on the Vendors web site and these links allow you to leave the Vendors web site. The linked sites are not under the control of the Vendor, the Vendor is not responsible for the contents of any linked site or links within a linked site. These links are provided as a convenience only and do not imply any endorsement or recommendation by the Vendor.

PORTIONS OF THIS SOFTWARE CONTAIN THE ASSIMP LIBRARY v4.0 OR LATER:

Open Asset Import Library (assimp)

Copyright (c) 2006-2020, assimp team
All rights reserved.

Redistribution and use of this software 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.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the assimp team, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission of the assimp team.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "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 COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

3 Overview

This document describes the full RepliSLS3D software package including all existing functionalities, options and additional software components. Depending on the software variant you really use some of the described features and functionalities may be not available or may be available only in a limited state.

The RepliSLS3D software package is available for different hardware and software platforms. Dependent on the peculiarities of the used operating systems, the exact storage position of the executable, the plug-ins and other required data may differ, here always a storage location is chosen that is a common position for such a software on that operating system. Within this document no operating system specific information are given, here please refer to the user manual of the operating system.

RepliSLS3D is a software specialised on all kinds of laser-based 3D "printing". These processes are known under different terms like selective laser sintering (SLS), selective laser melting (SLM), rapid prototyping, stereolithography and even 3D printing. From its general working principle they all are very similar:

1. a 3D model is imported which describes the object to be built (or to be "printed")

2. the model is sliced into several 2D layers

3. in repeated cycles one layer is marked/printed, then the Z-position is modified and/or new material is added to be used for marking/printing the next layer

4. this is repeated until the whole 3D model is built completely

Here RepliSLS3D can be used to control all necessary steps and operations, it supports following functions:

- import of 3D models in many different formats

- slicing into layers with different modes and parameters in order to fit to the used hardware

- automatic generation of support-structures at positions where object would hang over or parts would have to hover and probably can not be built without these additional structures

- applying of different also very complex fill patterns to both, slices and support structures to create solid models

- applying of different processing parameters to main models body and to support structures which influence the resulting material quality

- perform motion-operations to e.g. shift the Z-axis or apply new material between slices, synchronise with external equipment in order to integrate into complex machinery

- perform also long term build processes without any user interaction

Beside of that it offers the following, non-standard features which are important not only in case of complex build processes with special high-tech materials but also for specialised hardware and applications:

- enhanced mark patterns to keep the material cool by not marking always at the same position

- enhanced fill patterns to keep material cool by not filling one line after an other

- overlapping fill patterns to exactly influence the impact of the laser/tool on the material and the resulting quality

- complex and optionally interleaving pen parameters to handle also difficult materials

- support of many devices to perform the hardware operations between slices and to mark/print the model

- user privilege management to define different roles for users that have access to a device running this software

RepliSLS3D is specialised to and optimised for 3D printing applications as described above. For standard, non SLS/SLM-style laser marking applications, please check out RepliMark which is part of the same soft-

ware package like RepliSLS3D. Full specification and description of RepliMark can be found in a separate manual.

3.1 Security

This application is designed to control laser equipment and/or machinery which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and whenever a software update was installed or whenever some of the relevant parameters have been modified.

Here "software update" not only means updates of the RepliSLS3D software but also modifications with dependent software, like components or drivers for scanner controller cards.

3.2 Quick Start into RepliSLS3D

This section describes how to use RepliSLS3D and gives a short overview about basic set-up and usage of RepliSLS3D and describes with some short steps how to get first results easily. So this section is a short-cut to start into this software very fast but it ignores most of the useful features. These features are described in detail in following sections of the users manual, so it is recommended to use this quick start only to understand the very basic working principles of the software while going deeper into it afterwards.

1. **SECURITY CHECK**: The following steps describe how to set up RepliSLS3D and how to control laser equipment out of it. Thus all laser safety rules and regulations need to be respected, all required technical security mechanisms need to be available and active prior to starting with this software.

2. **Main configuration**: First the scanner controller card or output device that has to be used needs to be selected and configured. To do that, select Menu "Project" menu item "Project settings..." and go to panel "Scanner" of the now opening settings dialogue. Within the selection-list named with "Scanner Card" choose the scanner controller card you want to use. After selecting it, press the button "Configure" and set up the scanner controller card according to its specific needs (choose things like firmware file, correction table, working area size, number of controlled axes or what ever your card requires for proper operation) and according to your local hardware configuration.

3. **Additional devices configuration**: In same settings dialogue configure additional external equipment like motion controllers which are used for moving a Z-stage, for deploying material and for other things more. Here either go to tab-pane "Motion" and configure up to two motion controllers for generic use and/or go to tab-pane "Misc" to configure a focus shifter plug-in for controlling a Z-stage.

4. **Process parameter control**: In the same dialogue but in tab-pane "Additional hardware" a process control plug-in can be configured which is able to measure and influence additional process parameters such as gas pressure, temperature, gas concentration or others. Depending on your specific build process and hardware set-up, such a control plug-in can be used to keep track of track and have an (automated) influence on different process parameters.

5. **Save configuration**: Leave all these settings dialogues by pressing "OK" once all parameters have been entered correctly and select menu "Project" menu item "Save as default configuration". Now whenever you start RepliSLS3D and as long as you do not load a project file with a different hardware set-up, these settings are used by the application.

6. **Set up all operations which have to be done between two slices**: now it is time to generate a separate project file which contains all the operations that have to be done between tow slices (e.g. move the Z-stage via a motor, deploy new material via an other motor, trigger external devices via digital outputs, wait for feedback signals from external devices via digital input signals, …).
This all can be done via the elements in toolbars "Control elements" button. Here several motion/input/output and other elements can be added and configured until a project exists, that performs all operations that are required between two slices. Once this is done, save this project and configure

the path to this file via menu "Project" → "Project settings" → tab-pane "Motion axes" → input field "Process between slices".

7. **Save changed configuration**: Once again leave this settings dialogue by pressing "OK" and select menu "Project" menu item "Save as default configuration". Now whenever you start RepliSLS3D and as long as you do not load a project file with a different hardware set-up, this inter-slice-project is used automatically during building of your models.

8. **Import basic geometry**: Select menu "Project" sub-menu "Import 3D data" to load an existing 3D file in an appropriate format. Within the import dialogue set-up basic parameters which are used for slicing the data later.

9. **Arrange imported model**: Once you have imported the 3D mesh, left-click the related element in entity-tree on the right. There the imported model appears with a name which is similar to the path it has been imported from. Now you can re-arrange it and set it to a position within the build-space using the buttons and input fields in toolbar at upper end of the main window. The imported mesh needs to fit into the working area completely which is symbolised by the blue cube. All parts which are located outside can't be reached by the used marking/output device.

10. **Slice the model into separate layers**: Once this is done, select the root-node (the topmost element) in the entity-tree at the main windows right hand side. Now at the left a panel appears, which gives access to different slicing parameters. Configure it as needed and then press the yellow button "Reslice" to apply your parameter and to let RepliSLS3D slice the model into layers.

11. **Adding a hatch pattern**: Currently only the outline of the slices is shown and only this outline would be processed during laser marking. To mark/print also the inner side of the layers and to create a massive, filled object, a fill pattern has to be added by using the Additional Geometry element "Hatch". It is symbolised by a purple-coloured tool bar icon that consists of several horizontal lines. When this icon is clicked as long as the "Slice"-element is selected in the entity-tree, it is added to all slices automatically. Now the appearance of the slice and the appearance of the application changes:
- the first panel on the main windows left hand side switches to show all the "Hatch" parameters
- the entity-tree on the main windows right hand side now shows the new hierarchical structure of the element with the "Hatch" as sub-element of the primary "Slice" geometry
- the slices within the drawing area are filled with some hatch lines (which are visible only when related option is checked in the slice property panel used in previous step)
This hatch pattern now can be changed by modifying the values within the property-panel "Hatch" on the main windows left hand side.

12. **Save data**: To avoid the currently generated geometry gets lost, the project has to be saved now. To do that, select menu "Project" menu item "Save as..." and save these data at a suitable position using the .BEAMP format.
PLEASE NOTE: this project contains all the hardware definitions that have been done during configuration of the software in first steps. Whenever you change this configuration and then load this project again, the hardware set-up of the project is used instead of the changed one of the application. To avoid that after loading such a project, the menu "Project" menu item "Load default configuration" can be chosen to replace the projects hardware set-up by the local, desired one.

13. **SECURITY CHECK**: Next the scanner controller card together with a possibly connected laser will be accessed for the first time. That means it is opened and initialised and all connected equipment may start working now. Thus it is very important to ensure all security regulations are met and nobody can be injured and no damage can be caused also in case laser output or other motion starts spontaneously and unexpectedly!

14. **Prepare for laser marking**: Now the created geometry has to be sent to the connected laser and scanner system for processing it on some material. To do that the menu "Process" menu item "Mark" has to be selected. It opens the marking dialogue and tries to access the connected hardware. In case it is not possible to access the scanner controller card, an error message is shown and none of the buttons of the mark dialogue are usable (except the "Cancel" button to leave the dialogue). In this case you have to go back to the project settings as described in step 2 and to correct the set-up of your scanner controller card.

15. **SECURITY CHECK**: Next the laser and scanner will be accessed. Since there are some (laser) controllers available that are VERY sensitive to wrong laser signals, it is recommended first to check the output your scanner controller card produces. This can be done e.g. by using an oscilloscope in-

stead of the real hardware. The target equipment should be connected only in case all signals are checked, correct and acceptable by the hardware. More than this it is recommended to repeat this step whenever the something within the complete set-up changed; changes may be caused e.g. by modified parameters, by software or driver updates.

16. **Start marking**: Now marking can be started by pressing the big yellow button (the one with the laser warning sign) in the middle of the marking dialogue. This starts sending of all laser and scanner data to the scanner controller card so that it is able to output motion and laser control data synchronously. Such a marking operation can be stopped by pressing the big red button (with the STOP-symbol) at the dialogues right hand side.
PLEASE NOTE: this stop-button is not a replacement for a real emergency stop, it will try to stop the currently running laser process only via software which – in worst case – may fail for an example due to an abnormal problem or in case of a hardware failure. Thus there must be independent and working emergency stop equipment available in every case!

The result marked/printed after these steps lacks different things:

- the laser parameters are not suitable → this can be fixed by selecting menu "Project" sub-menu "Pen-Parameters" and by creating some pens with suitable parameters, next these pens can be configured to the slices, to the hatches, to the support structures and to the support structures hatches

- no motion is done to build up the model in Z-direction, to refill material and other things more → a separate project is required which performs all the "mechanical" steps which have to be done between two slices; such a project has to be set-up before a 3D model is imported by using the blue toolbar elements which can control different devices, motion stages, focus shifters and which can wait for external signals. When such a sequence is working, it has to be saved as separate project and then can be configured to be used between slices via menu "Project", sub-menu "Project settings", tab-pane "Motion", then at the bottom in field "Process between slices" the path to this project file has to be given.

3.3   User Interface

3.3.1   Menu Items

The menus in RepliSLS3D offers the following functions:

Menu Project:

- New project – create a new project, this deletes all possibly existing data

- Open project – open an existing RepliSLS3D project; this menu item can be used to load data which have been generated previously, it is not intended to add a new 3D mesh for processing

- Import 3D model – add a new 3D model to the currently used project; here a 3D file can be loaded which needs to be sliced in a second step; when this file is loaded while other 3D models already exist in current project, they need to be rearranged in build space in order to not to overlap each other

- Import 2D Image – load a 2D image of one of the supported formats.
When the imported file is a raster (=bitmap) image of type BMP, GIF, PNG, TIFF and others, it will be converted to a point cloud; the resulting geometry is special, it can not be used to cut it into slices and then to process it layer by layer. Different to this it is to be marked as once, it is processed from bottom to top and can be used e.g. for in-glass-marking or to mark within other transparent objects. The image can be imported into one or more different layers where each layer represents different brightness. The related values can be chosen in image import dialogue which opens after selecting the related image.
When the imported file is a vector image of type XPS, Svg, DXF and others, it will be imported as is but remains a flat, 2-dimensional element which can't be cut into slices but only marked as single, flat geometry.

- Save project – save the currently loaded project using the same name, this operation overwrites an existing project without further notice

- Save project as – save the currently loaded project while giving it a (new) name by using a file dialogue

- Save with options – save the currently loaded project with the possibility to specify what this project may contain exactly

- Project settings – open the global settings dialogue where all application and hardware related configurations can be done

- Pen settings – open the dialogue where all material and output related parameters can be configured

- Save as default configuration – save the current hardware and pen settings as default so that these parameters are loaded as default on next start-up of RepliSLS3D

- Load default configuration – when a process file was loaded it possibly can contain an own hardware configuration which does not fit to the current device. Using this menu item the local default configuration can be loaded overriding the parameters of the loaded project

Menu View:

- 2D top – instead of the default, three-dimensional and freely rotatable spatial view this menu item switches over to a 2D view which shows the data from top (which is similar to the scanheads point of view)

- 3D – switches to the three-dimensional, spatial view where the scene can be rotated by holding down the right mouse button during movement and where the scene can be moved by holding down the left mouse button during movement

Menu Edit:

- Auto-arrange 3D meshes – when several 3D model have been loaded using the "Import" function described above, they may be located at non-optimal positions or may overlap each other. Using this function they can be arranged automatically. After selecting this menu item a dialogue opens, where the distance to the working area borders, the distance between the meshes, and alignment in Z-direction has to be specified. These values then are used to arrange the 3D meshes in build space. This operation moves the meshes but does not rotate them. For more details please refer to section "3.7.4 Auto-Arrangement of 3D Meshes in Sliced 3D Mesh Groups"

- Delete element(s) – when an element is selected in windows right hand side element tree, it can be removed completely with this function

- Extrude – this function extrudes a flat, two-dimensional element into the third dimension and creates a slicable 3D mesh out of it. After clicking this menu item, a dialogue opens, where several options can be chosen which influence the extrusion result:
  - Extrusion axis – specifies along which axis the extrusion has to take place, this is done always parallel to one of the three axes X, Y or Z so that it is recommended to align and position the element properly before this function is called; for 3D applications Z-direction would be the first choice, the remaining directions X and Y are of limited use and intended to be used for very specific cases only
  - Extrusion direction – specifies the extrusion direction (up or down, left or right, in our out)
  - Extrusion length – specifies the total extrusion depth, means how much the element shall grow into the selected direction

- Virtual workspace – this function can be used to process vector data which are larger than the available scanheads working area. For a more detailed description please refer to the section "3.3.1.1 Virtual workspace" below.

- Undo / Redo – using these functions an operation step can be undone or redone

Menu Process:

- Build – this menu opens the mark / building dialogue which can be used to start a mark-operation. The name "mark" is somewhat uncommon due to it's origin in laser industry and its double meaning:
  - when the project consists of control elements only triggering external devices which is intended to be executed between slices later, typically no laser output happens and the operation is more some kind of "move" or "test"
  - when the project consists of a complete, sliced 3D model and has a control-project (as mentioned above) configured, the operation is some kind of "build"

- Save build data – instead of sending the build data to the connected devices directly and start a build process automatically, this function can be used to save the raw build data for later processing. This function is different to the normal function to save the project into a file for later usage. Build data saved via this menu item are stored in a format which depends on the used scanner controller card and its plug-in (please refer to section "3.4.1 Scanner Controller Plug-Ins"). The same is true for later usage of these data: it is not possible to load them back into RepliSLS3D, it depends on the configured device how they can be used and processed. Furthermore when the chosen scanner controller plug-in does not support writing of build data, this menu item is disabled and this function is not available.

- Show controller state – this option shows different scanner controller state information; which information are shown depends on the used scanner card and its capabilities but it may consist of
  - digital output data (reflects the data that have been sent to the output but not necessarily the data that really are set as long as the card does not support readback of these outputs)
  - digital input data
  - analogue input data
  - plug-in information
  - scanhead information

Menu Help:

- About – shows application and version information

- License – displays the license which is currently used and which decides what functions are available

- Check dongle – this function can be used in case of problems with license retrieval from the USB dongle.

- Generate bugreport – this function can be used in case of problems with the software to gather different data which are helpful to analyse these problems.
  Please note: this function collects different information which may include sensitive, personal or other confidential data. When you do not agree to share these data, do not use this function. Otherwise you agree that we get, store, analyse and read these data for an undefined time (but typically only as long as it is necessary to check your specific problem).
  In case of a problem with the software, please use this function to generate a bug report. When you experienced a crash, where the software has died completely and therefore this function is no longer accessible, please restart RepliSLS3D, load the last project you used and where you experienced this crash.
  This function creates a ZIP-file where you can specify where to store it. This ZIP-file contains the bugreport-data and has to be sent to us in order to check what causes the problems. We provide a public key for encryption of e-mails and data, so you also can send us such a bugreport fully encrypted.

3.3.1.1 *Virtual workspace*

This is a special kind of group that concatenates several vector elements into one group. Due to the kind of this group it is recommended to use only geometries and not to put other ones like motion elements into it.

When using this virtual workspace, the software is able to handle geometries and bitmaps that are larger than the scanheads available working space. It cuts the contained elements into smaller pieces and combines the resulting tiles with motion control information so that these large and cut geometries and bitmaps can be put together during processing by using e.g. a XY-table that moves to the related positions. So this functionality can be used to handle big working pieces. The cutting operation only applies to X-Y-direction (equal to top view). Thus it is also recommended to switch to 2D top view when applying this function to some elements.

As soon as some geometric elements are grouped together within such a virtual workspace, it's behaviour can be modified using the parameters that are available within the property panel of this group:

– Split horizontal – splits the grouped geometry along X axis, this option can be combined with "Split vertical" to define two-dimensional movements

– Split distance – the split size in horizontal direction

- Split vertical – splits the grouped geometry along Y axis, this option can be combined with "Split horizontal" to define two-dimensional movements

- Split distance – the split size in vertical direction

- Process order – this parameter defines in which order the pieces have to be processed:
  <u>Left to right</u> – horizontal direction always starts from left
  <u>Right to left</u> – horizontal direction always starts from right
  <u>Zig-zag from left</u> – horizontal direction starts from left and performs zig-zag movements
  <u>Zig-zag from right</u> – horizontal direction starts from right and performs zig-zag movements
  <u>Top to bottom</u> – vertical movement starts from top
  <u>Bottom to top</u> – vertical movement starts from bottom
  In case two-dimensional motion is defined both process order selections are combined with each other to describe the 2D motion.

- Motion speed: the movement speed in unit mm/sec that is applied to the motion between the splits.

- Overlap splits: here an as a percentage value can be set that specifies how much the pieces should overlap, means how much of the geometry should exists twice so that it is processed two times at the areas of intersection

- Try split between pieces – here the splitting algorithm tries to find empty spaces between geometries to avoid cutting of lines; in case this option is set, the specified split distance is used as maximum value: a cut is set wherever space between geometries is found or latest after the given split distance. In last case cutting of lines is still done. So in this mode split distance define the absolute maximum size of a resulting piece while the distances where the geometry are cut really can be smaller than this.

- Axis mapping: at the lower end of the property panel a matrix of radio buttons exists where the axes can be mapped to the related movements, there it can be specified which of the axes A, B or C of the two possible outputs OUT2 and OUT3 have to provide the related planar movements in horizontal or vertical direction.

- Save as default – after application start-up new virtual workspaces are created with some default parameters. When these parameters are changed, other virtual workspaces then are created with these changed parameters, but after restarting RepliSLS3D, the initial default parameters are used again. When this button is pressed, the current virtual workspace parameters are saved as new default so that they are used whenever a new virtual workspace is created after RepliSLS3D was started newly. The related data are stored within the users working directory in a file named ".repliuiS-PLT"

After such a virtual workspace has been created and after the correct parameters have been set, the appearance of the related geometry changes (visible only in 2D top view): now it is overlapped by horizontal and/or vertical lines. These lines specify the cutting positions and can be modified manually. They can be dragged with the mouse in horizontal or vertical direction exclusively to change the position where the geometry has to be cut. The related modifications are done as soon as such a cutting line is released.

Please note: the positions of the cutting lines do not influence the parameters given within the property panel but as soon as the parameters within the property panel are modified, the position of the cutting lines are reset to their default. Thus their positions should be changed only in case all other parameters do fit perfectly.

When a virtual workspace was created, it can be disabled in element tree on main windows right hand side: right-click the related root-element and choose "UnGroup Element" in the context menu which opens there.

3.3.2   The Property Tab-Panes

The main window of RepliSLS3D contains some special tab-panes on its left hand side. These panes give access to different parameters and values, they can be used to influence geometries in a very exact way.

3.3.2.1 The tab-pane Of The Currently Selected Element

The very left tab-pane is a dynamic one, it changes its contents depending on the element that is currently selected. As long as no element is selected this panel is empty, no parameters can be changed here. These element-specific values that become visible as soon as something is selected within the drawing area or within the element tree on the windows right hand side are described in detail below together with the different elements.

In case more than one element or in case a simple group element is selected this tab-pane is empty too.

3.3.2.2 The Element Tab-Pane

This panel contains some global parameters of the selected element. Depending on the type of selected element only some of the parameters are accessible:

– Type – this is an informational field and therefore read-only, it displays the type of the element

– Name – here the name of the element is shown and can be changed; this name is auto-generated when the element is created and shown within this panel and within the element-tree on the main windows right hand side.

– Head – in case of a multihead environment (as described in "3.11 Multihead Operations") there is the possibility different controller cards are able to process the same geometries. Here the software typically decides for its own and based on the current global settings (as done in tab-pane "Multihead" as described at "3.4 Project Configuration") which controller is responsible for which part of the geometry. For these cases, where a geometry has to be assigned to a controller card (aka "head") directly, this selection box can be used. Here a head can be selected to process the geometries of the related element. When no head is selected but option "Automatic" is chosen, the default, automatic behaviour as defined in general project settings is used for this element. Here "Automatic" is the default behaviour and should be used unless an other selection is necessary.

– Alignment – in general pen-settings there is one option to align the laser/the milling tool at the inner or outer side of a polygon (please refer to "3.6.3 Laser Pen Parameters" for details). This gives the possibility to cut elements in exactly the size specified by these geometry by taking the cutting width of the laser/tool into account. Instead of defining such an alignment globally via the pen settings, using this selection list it is possible to align a single element in the same way. Means when a different value than "From pen settings" is chosen here, the "Align spot to vectors" parameter out of the used pen is ignored but the local definitions as per these settings are used.

– Process element – this check-box decides if the element has to be processed or not, when it is not checked, the related element acts as some kind of placeholder only, it is not used by the target system for processing the working piece

– Mark contour first – on a standard marking process first the additional element (the "hatch") is marked and at the end the contour is appeded. When this option is set for a element whic hitself is the contour (e.g. a primary element), this order is incersed, then first the contour is marked followed by the hatch

– Trace data – when this option is set, the name, and when available, unique data of this element are used for traceability purposes. This means, when a marking cycle was performed successfully, the related Smart Interface message or trace log entry contains information about this element too. These information consist of the name of the element. When this option is enabled for an input element, beside the name also the possibly unique input data are added to the trace data. For more information please refer to section "3.13 Smart Interface" (when the trace data have to be provided via the Smart Interface) or to section "3.4 Project Configuration" option "Trace log" (in case the trace data have to be written into a log file)

– Skip marks – when this is set to a value greater than 0, the related element is no longer marked on every process cycle but skipped for the given number of times; after this number has elapsed, it is marked once and skipped afterwards again

– Skip offset – this parameter belongs to "Skip marks" and specifies an offset at what mark/skip counting has to start; so with both parameters elements can be created that are marked interleaved:
    Element 1 with skip marks = 1 and skip offset = 0

Element 2 with skip marks = 1 and skip offset = 1
→ results in two elements that are marked alternating, each exclusively

Element 1 with skip marks = 2 and skip offset = 0
Element 2 with skip marks = 2 and skip offset = 1
Element 3 with skip marks = 2 and skip offset = 2
→ results in three elements that are marked alternating, each exclusively

- – Number of points – this is an informational element and can't be changed directly, it displays the number of vector lines that are used to create the selected element

- – Total line length – this field contains an information too, here the total length of the geometry is shown; this value takes the "on-"-lines into account only, "off"-lines (movements with the tool turned of to go to the beginning of the next "on"-line) are ignored here

- – Process time – here the time is displayed that is needed to process the currently selected element; this value is calculated out of the length of "on"-lines, the length of "off"-lines between them and the speed and delay information of the pen that is assigned to this element

3.3.2.3 *The Motion Tab-Pane*

In tab panel Motion it is possible to drive configured motion axes manually and independent from any motion elements in current marking project. Please note: when there are some marking elements used directly within the project, it has to be taken into account that the user may have changed the position of the axes via these manual functions. Thus at least the first motion element of a project should perform a movement to a defined, absolute position or should reference its position.

Within the motion tab-pane following functions are available:

- • OUT2 / OUT3 – specifies which of the two possible motion controllers has to be used

- • Speed – specifies the speed a manual motion operation can be performed with; PLEASE NOTE: when the speed value given here is bigger than the maximum speed of the related axis, this value is clipped internally, means the axis moves with it's maximum speed which may be slower than the nominal speed entered here

- • movement value – the value that can be entered in middle between the arrows specifies the relative movement distance or the absolute motion position to move to (depending on the arrow button that was pressed)

- • Double arrows – these buttons cause a movement as long as the button is pressed

- • Single arrows – these buttons cause a single movement via the distance or angle specified with the number input field in the middle, here the time the button is pressed does not influence the motion

- • Arrow to dot – when this button is pressed an absolute movement is performed, the number input field in the middle between the buttons specifies the target position to move to, the time the button is pressed does not influence the motion

- • Ref – start referencing of the related axis

For planar movement axes the images of the buttons can be changed within the motion axis configuration panel of the general project settings. Depending on the real movement direction (left/right, up/down, in/out) a different symbol can be shown here in order to make it easier for the operator to decide which direction a motion will use. The same is true for the axis names, they can be configured via the axis alias value in general project settings.

3.3.3 The Element-Tree

On the main windows right hand side a tree is shown that represents the current projects structure. It consists of a list of primary elements and their (optional) child elements. During processing on the target system the elements are processed in the same order as they are listed there from top to bottom (including their children).

Every primary element may contain several Additional Geometry elements and several Post-Processing tools. They are listed as child nodes of their Primary Geometry parents within this tree.

All these primary and child nodes can be edited within this tree via a context menu. This menu can be opened by selecting a node within the tree and clicking the right mouse button. Then following functions can be chosen:

- Group Elements – while holding down the Shift- or Ctrl-key it is possible to select a range of elements within the entity tree; in case more than one Primary Geometry element is selected it is possible to put them into a group using this function

- UnGroup Elements – this item can be used when a group element is selected, it resolves the group, pulls all element out of it into the hierarchical level where the group existed before

- Delete Element(s) – deletes an element completely; in case this is applied to a primary element all its sub-elements are removed too

- Duplicate – the currently selected Primary Geometry element is copied and inserted at a position that is slightly shifted according to the original element

- Move To Top – the selected element is moved to the top position within its same hierarchically level (means a child of a primary element will stay a child of it but it will become the first one)

- Move Up – the selected element is moved one position up but also within the same hierarchically level

- Move Down – the selected element is moved one position down within the same hierarchically level

- Move To Bottom – the selected element is moved to the bottom position within its same hierarchically level (means a child of a primary element will stay a child of it but it will become the last one)

- Import 3D Model – this option is active only when a single Sliced 3D Mesh Group element is selected, it gives the possibility to import an additional 3D Mesh into this Sliced 3D Mesh Group; after successful import this Sliced 3D Mesh Group keeps two more 3D Meshes and creates slices that consist of all of their 3D data, the 3D Meshes which can be positioned within the working area independent from each other

- Clean between slices – this option is active only when a single Sliced 3D Mesh Group element is selected; when this option is set, such a 3D Slice Cleaner element is generated on construction of the Sliced 3D Mesh Group. For more information about this type of element which can be used for cleaning purposes between slices, please refer to section "3.7.5.2 3D Slice Cleaner"

- Auto-arrange – this option is active only when a single Sliced 3D Mesh Group element, when a group or when several primary elements are selected, it gives the possibility to arrange several 3D Meshes or child elements within a group; for a detailed description of the auto-arrange function please refer to section "3.7.4 Auto-Arrangement of 3D Meshes in Sliced 3D Mesh Groups" below

- Merge 3D Meshes – when several 3D slice groups are selected, this option can be chosen to merge their 3D meshes into one element. After this operation has been finished, all the selected 3D slice groups are combined into one containing all the 3D meshes as one single object

- Collapse all – closes all nodes in the current element tree; when all nodes are collapsed and the whole tree is selected (with shift-key held down) only the top level primary elements are selected but not their sub-elements

- Expand all – opens all nodes in the current element tree

3.4   Project Configuration

When starting with a new project it has to be configured first so that the editing environment of RepliSLS3D reflects the real processing conditions and hardware set-up.

This can be done in global settings, the related dialogue can be found in menu "Project" sub-menu "Project settings...". There a dialogue opens which offers several configuration options within different tab panes. The following parameters can be set and modified within these panes:

In tab-pane "User interface":

- Visual grid size – this is a helper grid that is used and visible within the working area as an array of of points with a distance in X and Y direction as specified here; this grid is visible in 2D view only

- Snap to grid – when this check box is selected, all geometry that is drawn or re-positioned within the working area of the 2D view, automatically snaps to a position that is defined by the visual grid; this option gives the possibility to position elements very exact

- Background colour – this option changes the visual representation of the editor and does not have any influence on the generated data, it specifies the colour of the background within RepliSLS3D's drawing area

- Working area border – the working areas (as defined in scanner controller settings for every scan-head) are shown as rectangle within the drawing area of the application, here the colour of the bounds of related rectangles can be changed; similar to the previous option this one also does not influence the generated data

- Editing area border – beside the working area, which specifies the range a single scanhead can cover, an editing area exists which is a visual help showing the user where drawing is allowed for the current configuration. This area exists only once for a configuration, it can be equal or smaller than the working area of a single scanhead (in single head environments), it can include several scan-heads working areas (in multihead environments) or it can be larger than the available scanheads working area (e.g. when it has to be used together with the functionality of Active Split Group).Here the colour of the bounds of this area can be changed; similar to the previous option this one also does not influence the generated data

- 3D mesh colour – this option changes the colour of the mesh of an imported 3D model within RepliSLS3D's drawing area

- 3D support mesh colour – this option changes the colour of the support structures around a mesh of an imported 3D model after it has been sliced

- Editing area shape – here the shape of the editing area can be selected. By default it is rectangular (or cuboid in 3D mode), alternatively it can be switched to a cylindrical shape (a circle in 2D mode)

- Editing area upper left – specifies the upper left corner of the global editing area within the coordin-ate system of the application

- Editing area size – specifies the size of the global editing area within the coordinate system of the application; this is for the cuboid shape width, height and depth and for the cylindrical shape dia-meter and depth (aka height of the cylinder)

In tab-pane "Scanner" it is possible to configure the scanner controller or any other output device as suppor-ted by RepliSLS3D. In case these components are set up properly according to the used hardware environ-ment it is possible to start building and motion operations directly from within RepliSLS3D:

- Scanner Controller Selection – this combo box has to be used in multihead mode where RepliSLS3D controls more than one scanner card at the same time. The selection with this combo box influences the settings of all following parameters. Please note: in multihead and singlehead mode the first scanner controller always has to be set, it is not possible to start e.g. with settings for the second scanner controller and leave the first one empty.

- Scanner Controller – here a scanner card can be chosen out of a list of available scanner controllers. As soon as a scanner card is selected for usage within RepliSLS3D, it can be set-up by pressing the button "Configure". Although this configuration is done for the scanner controller card all the para-meters like the working area size and position are used to set up RepliSLS3D too. For an overview about available scanner controller plug-ins please refer below.

- Initialise on start-up – when this option is set, the scanner controller card and – when available and configured – motion-, laser- and process-controllers are initialised on start-up of the software. Nor-mally this is done when the Mark-dialogue is opened for the first time, but when this option is en-abled, it happens when RepliSLS3D is executed.
PLEASE NOTE: In general it is not necessary and not recommended to use this option since it causes possibly unnecessary accesses to the hardware. Only in very rare cases where some ex-ternal equipment has to be initialised as early as possible (e.g. when some hardware needs to be heated up), this function should be used

- Home position – when this option is set, the scanner is moved back to the XY-coordinates given right beside this checkbox whenever a marking cycle was finished; using this function the head always ends up in a defined position on regular marking cycles, no matter what the last marked vector really was.

- Override manual stop – when a marking operation is stopped manually (e.g. by pressing the stop-button in marking dialogue or by sending a stop via one of the programming interface) all operations are cancelled as fast as possible and a probably used home position is ignored. When this option is enabled, the software tries to reach the home position also in case of a stop so that markign always ends at this defined position.

- Pilot Laser – this section is related to the configuration of a pilot laser that does not perform marking operations but points to the position where the main laser would influence the working piece (as some kind of preview-laser). This pilot laser can be used out of RepliSLS3D's marking dialogue directly (please refer below for more details); the related parameters are enabled only in case a scanner controller card was chosen and configured properly

- Output Port – here a port can be chosen that has to be used to turn the pilot laser on and off

- Output Number – here the bit number at the previously chosen output port has to be selected which shall be used to toggle the pilot laser;
PLEASE NOTE: some scanner controller cards use some of their output bits for specific, reserved purposes, these bits should not be used for the pilot laser to avoid any unwanted and undefined behaviour. As a second precondition it is necessary to connect the pilot lasers on/off input with the output that was selected here.

- Timeout – here a timeout (in unit minutes) can be specified which turns off a pilot laser after this time. Using this function it can be ensured a user does not (accidentally) let a pilot laser run over hours or independently

- Invert Output Logic – inverts the output value for the signal that should turn the pilot laser on, this option needs to be set for some specific pilot lasers to invert their behaviour

- Always on – when this option is set, the pilot laser is kept on all the time, means it is not modulated and not turned off during jumps. Only before next real marking operation it is ensured the pilot laser is turned off since some laser types can't turn on the main laser as long as the pilot is still on.

- Correction Factor / Correction Offset – pilot lasers normally have a different colour and therefore a different wavelength than the main laser. In case some optics are used within the scan head, this may result in variances in the marked positions due to a different light refraction factor. Using these parameters such variances can be compensated. The correction factor is a factor that modifies the size of the pilot lasers output in X and Y direction while the correction offset modifies its absolute position

- Ready For Marking Signal – here a signal can be defined that is emitted on a digital output of the scanner card as soon as the software is ready for marking

- Output Port – here a port can be chosen that has to be used for the "Ready For Marking" signal

- Output Number – here the bit number at the previously chosen output port has to be selected which shall be used to toggle the "Ready For Marking" signal;
PLEASE NOTE: some scanner controller cards use some of their output bits for specific, reserved purposes, these bits should not be used for this signal in order to avoid any unwanted and undefined behaviour.

- Invert Output Logic – inverts the output value for the ready-signal

- Mark dialogue opened signal – here a signal can be defined that is emitted on a digital output of the scanner card as long as the marking dialogue is open

- Output Port – here a port can be chosen that has to be used for the "mark dialogue open" signal

- Output Number – here the bit number at the previously chosen output port has to be selected which shall be used to toggle the "marking dialogue open"" signal;
PLEASE NOTE: some scanner controller cards use some of their output bits for specific, reserved

purposes, these bits should not be used for this signal in order to avoid any unwanted and undefined behaviour.

- Follow-up time – here a time range (in unit seconds) can be specified the "mark dialogue open" signals stays turned on after this dialogue window was closed. When it is reopened again before this time elapses, the output stays active without interruption

- Invert Output Logic – inverts the output value for the active-signal

- Marking Active Signal – here a signal can be defined that is emitted on a digital output of the scanner card as long as marking is in process; this signal is NOT identical to the laser gate which turns laser on and off during such a marking process
PLEASE NOTE: this signal is generated from within RepliSLS3D and independent from the features of the used scanner controller. Since some controller cards support this feature natively, these options can be disabled and therefore are not accessible. In such a case the marking-active-signal can be configured within the settings of the scanner controller card directly.

- Output Port – here a port can be chosen that has to be used for the "Marking Active" signal

- Output Number – here the bit number at the previously chosen output port has to be selected which shall be used to toggle the "Marking Active" signal;
PLEASE NOTE: some scanner controller cards use some of their output bits for specific, reserved purposes, these bits should not be used for this signal in order to avoid any unwanted and undefined behaviour.

- Invert Output Logic – inverts the output value for the active-signal

- Interlock Input – this section provides functionality to watch the state of an interlock / emergency stop circuit.
PLEASE NOTE: this function is NOT intended to perform any security-related functions, it neither is able to implement emergency stop functionality nor is it able to turn of any hardware in case of an emergency! All these emergency-stop-functions have to be done in hardware only and apart from any software or connected control boards! This function is only able to watch the state of external emergency/interlock circuitry and to ensure reinitialisation of the connected controllers when an external event has occurred, was handled by the external hardware and is no longer active!
When this function is used, the input selected with the options below, is expected to be HIGH. In case it is not, no marking operations are possible. When it goes to LOW during a marking operation is running, it is stopped. Once the input was LOW, a possibly opened marking dialogue has to be closed and all connections to used devices are shut-down automatically. Due to the shut-down of the connected devices, as soon as the input is back at HIGH, a complete initialisation of all hardware is performed including referencing operations (if necessary). So this function can ensure all these devices are in a proper and defined state also after an emergency stop.

- Input Port – here an input port can be chosen where the interlock/emergency stop signal can be read at

- Input Number – here the bit number at the previously chosen input port has to be selected which shall be used to watch the emergency stop/interlock signal

When more than one scanner/head was configured, it is possible to do some multihead-related configurations in tab-pane "Multihead". When only one scanner controller/scanhead exists, these settings do not have any influence on marking operations:

- Preferred head usage – within this matrix it is possible to define which head should be used for what kind of operations. So as long this is really possible for a head and related geometries, the head then will be used for this operation. Here following options can be chosen for a head:
- Automatic (default) – the geometries are assigned to this head, which is suited best to process them. So in case of partial overlapping working fields the automatic operation tries to spread the geometries over the heads in a way so that as long as possible marking operations are done while switching as seldom as possible from one head to an other
- Contour – the head should try to mark only contours. Only when a contour is located in an area of

an other scanhead and outside of the working field of the selected scanhead, or when an other head in "Automatic" mode fits better, this option is ignored and an other scanhead will mark contours

- Hatch – the head should try to mark only hatches. Only when a hatch is located in an area of an other scanhead and outside of the working field of the selected one, or when an other head in "Automatic" mode fits better, this option is ignored and this other scanhead will mark the related hatch

- Require trigger – when marking data are available for and have to be sent to this head, a trigger is inserted into the stream of data. So when this option is set, the change from one head to an other is neither done automatic nor do they run in parallel permanently, but an external trigger signal is needed (ExtStart) in order to start output of marking data at the head where this option is set

- Wait for tool – this option is related to milling output devices which have the ability to send some kind of "Tool change"-command. When this option is set and the software detects a change of the pen during a marking cycle, a) a tool-change command is sent to the related controller and b) a message box pops up for the head where this option is set. This message box asks the user to change the tool and the marking process will not continue until this message box is confirmed

In tab panel Motion Axes it is possible to configure up to two different motion controllers with up to four independent axes each. Both possible motion controllers are identified via names OUT2 and OUT3. These two identifiers are used within RepliSLS3D whenever it can be chosen which one of the two controllers has to be used:

- Motion Controller OUT2 or OUT3 – this combo box can be used to select a motion controller that would be accessed whenever a motion control information is assigned to output OUT2 or OUT3 (please refer to the description of the Primary Element "Motion" and other motion-related functionalities below). The parameters that are set for such a motion controller are NOT handed over to RepliSLS3D fully, so when some of the axes of a motion controller are not enabled here within its settings, they are still be usable and configurable within the related RepliSLS3D elements.

- Bind axis to slices Z-position – this function is related to marking of sliced 3D data. This option should be set for all these axes, which perform a Z-movement which has to have the same height as the slice thickness of a 3D slice group which is processed. Then the height of the slice is set as motion distance automatically – no matter, what value the related element is configured really. So when a 3D model is sliced with changed parameters, the motion settings of a globally configured process project (an external project which is executed between slices automatically) do not need to be changed manually.
When overwriting the motion distance of such a motion element, only the sign of the distance is kept, means when a negative movement was specified, the negative movement direction will be performed – but using the current slice height.
This option can be set for all axes which in any way are related to a motion operation which depends on the height of a slice.
Following preconditions have to be met in order to let the motion distance of an axis automatically be set to the current height of a slice:
- the option "Bind axis to slices Z-position" needs to be set
- within the related motion element the axis motion needs to be set to "relative"
- a positive or negative relative motion distance needs to be set
- the motion element is part of a sliced 3D group (as child of the related slice) or it is part of a project which is processed between slices (please refer to parameter "Process between slices" below)
When one of this conditions is not true, the related axis will be left untouched and a motion operation will be performed using exactly these values which are set for this element.

- Auto-move back to start position: when this option is set for an axis, its position at the beginning of a marking cycle is stored. As soon as the marking operation has ended, the axis will be moved back to this starting position automatically.

- Key controlled – configure single axes for control via keyboard operation. When this option is enabled, these axes can be moved out of mark-dialogue or "Motion"-panel by pressing ALT-key plus one of the cursor keys up/down/left/right. Then these axes are moved by the step distance and the speed as entered in related input fields.

Here it is possible to configure up to two axes for this operation mode. When the related checkbox is set for a longitudinal or vertical axis, it can be controlled by they key-combination ALT plus cursor up/down. When the checkbox is set for a horizontal axis, it can be moved step-wise by the key-combination ALT plus cursor up/down in motion panel or mark dialogue.

• Reference axis on start-up – here these axes can be specified that have to be referenced on application start-up or on first opening of devices automatically

• Referencing order – this option is related to the referencing-function which can be enabled with the reference-checkbox as described above. Using these drop-down-lists, an order can be specified in which the axes shall be referenced. The smaller a number is, that is selected for an axis here, the earlier the related axis will be referenced. When different axes make use of the same order number, they are either referenced at the same time or the underlying motion-controller decides which order is suitable.

• Axis alias and orientation – here for the default axes A, B, C and D an own name can be set that is shown e.g. within mark dialogue to identify the axis; additionally for planar axes it can be specified in which direction movement is done, the setting made here influences what kind of arrows are shown for axis movements in mark dialogue

• Don't show warning when referencing axes – when some axes are configured to perform referencing/homing operation on start-up, this may result in an unexpected, surprising and therefore possibly dangerous movement. Because of this prior to referencing operations an information is shown the user has to confirm in order to start the referencing operation. When this option is set, this information is NOT shown, referencing of axes starts immediately and WITHOUT ANY WARNING! Thus this option has to be used carefully and only in case the whole machine is secure and covered completely.

• Process before slice group – here a path to an existing .BEAMP project can be given that is used before building of a 3D slice group starts; this project can contain a processing sequence that is used for initial operations that are necessary before a slice-by-slice processing can start; such a project is nothing more than a RepliSLS3D project that consists of different motion and I/O operations only performing all that is necessary as initialisation. Such a project has to be set-up only once and then can be configured here in order to let RepliSLS3D use it automatically before a building process.

• Process between slices – here a path to an existing project can be given that is used when 3D meshes are sliced; this project can contain a processing sequence that is performed between every slice; such a project is nothing more than a RepliSLS3D project that consists of different motion and I/O operations only performing all that is necessary between two slices. Such a project has to be set-up only once and then can be configured here in order to let RepliSLS3D use it automatically during building process.

• Process after slice group – here a path to an existing .BEAMP project can be given that is executed once after building of a 3D slice group has ended; this project can contain a processing sequence that is used for final operations that are necessary after a 3D model has been built; such a project is nothing more than a RepliSLS3D project that consists of different motion and I/O operations only performing all that is necessary as deinitialisation. Such a project has to be set-up only once and then can be configured here in order to let RepliSLS3D use it automatically after a building process.

Remaining plug-ins to access and control "Other hardware" are concatenated in related tab-pane:

• Scanhead Selection – this combo box is dedicated to multihead configuration too, different to the other combo box which can be found in "Scanner" tab-pane, it is assigned to a scan head. This differentiation is made because one scanner controller / one plug-in defined above may access several scan heads independently, therefore head-specific parameters have to be set separately. The selection made with this box belongs to the following parameters:

• Laser – here a laser controller plug-in can be selected that communicates with a laser independent from the scanner card. As soon as a laser controller is selected for usage within RepliSLS3D, it can be configured by pressing the button "Configure". This function is useful in case a laser is used that requires control via a different communication channel that does not exist on a scanner controller card. This can be e.g. an Ethernet connection or a serial interface which is used to send control commands to the laser.

- Z-Shifter – here a plug-in can be selected that communicates with a special motion axis to be used for shifting the Z-position, expanding the beam or shifting the lasers focus independent from the scanner card. As soon as such as Z-Shifter is selected for usage within RepliSLS3D, it can be configured by pressing the button "Configure".

- Process Controller – here plug-ins can be selected that are able to control additional process parameters (beside the laser marking related ones). Possible parameters are pressure, temperature, gas flow and others where the current value is read from an external device. Depending on the possibilities and settings of the used plug-ins, other devices can be controlled or switched and/or the currently running marking process can be halted or continued. Also depending on the used plug-ins, the pen settings dialogue may be extended by an additional tab-pane as soon as such a process control plug-in is enabled. This additional tab-pane then contains parameters which are handed over to this plug-in during runtime and dependent on which pen is currently selected.
  As soon as such as Process Control plug-in is selected for usage within BeamConstruct, it can be configured by pressing the button "Configure". There all parameters can be set according to the capabilities of this plug-in.
  Here several process-control plug-ins can be selected. The combination of these plug-ins allows to access different sensors which are connected via different interfaces. Typically a plug-in supports exactly one type of sensor and in most cases is able to measure one value via this sensor. Now by combining several plug-ins, several types of sensors can be combined and used. So these plug-ins provide some kind of tree-structure:
  → process-control plug ins (several can be chosen and combined freely in this panel)
      → one or more values measured and displayed by this plug in (such as temperature, gas pressure, gas concentration,…); typically one plug-in is responsible for one sensor and one measured value
          → up to four parameters that belong to each measured and displayed value (such as lower threshold, upper threshold, setpoint, …), which can be configured in settings of the plug in (here their default values can be set) and which can have different values for each pen (the pen-dependent values can be set in pen settings dialogue as soon as such a process-control plug-in was set up)

Tab-pane "Misc" contains options and configuration possibilities that do not belong to any of the previous ones logically:

- Auto-optimise vectors on marking – RepliSLS3D is able to handle a very high accuracy internally – which is in very most cases much higher than what the controller hardware and scanhead are able to produce on output. When this option is set, the vector data are analysed before they are sent to the scanner controller(s) and all data that are beyond the resolution of the used hardware are removed. Depending on the data to be marked this can save nameable amount of marking time and data.

- Auto-slice 3D-meshes on change – when a 3D-model is imported to be sliced into layers there are several operations that require reslicing of these data. Depending on the size of the 3D model and the thickness of slices this can be a time-consuming operation and therefore has to be triggered by the user manually. This behaviour can be changed, when this option is set, reslicing of the 3D mesh is done automatically after every operation that requires it. This option should be used only in case the used 3D models are simple enough or the computer hardware is powerful enough, elsewhere it can result in long lasting delays during editing of 3D models and sliced data.

- Show Tab-Panes in Mark-Dialogue – this area consists of a set of checkboxes that correspond to the different tab-panes shown in Mark-Dialogue. Here it is possible to uncheck some of them in order to hide the related tab-panes. This function can be used to not to show unused functions to the end user

- The sub-section "Limits" gives the possibility to configure allowed ranges and behaviour of some pen parameters.
  Here for each scanhead the minimum and maximum value that can be specified within pen settings dialogue can be predefined. The head these limits have to be applied for can be selected via the combobox right above the input fields. The allowed range of following values can be configured for each head separately:
  - Pen Frequency

- Jump Speed
- Mark Speed

Beside of that the behaviour of the related slider can be changed, when check box "Logarithmic" is set, the sliders no longer act linear.

Option "Allow unrestricted laser on/off delays" disables some logic checks in pen settings: Normally the delays applied to a pen have to follow some logic rules which are automatically checked when the user modifies their values. When this option is set, these logic checks are disabled and illegal laser delay values can be set for a pen. Since this may result in some very strange and undefined behaviour during marking, this function should be used only in some very rare cases where it is clear what happens when such illegal laser parameters are used!

- Parts Counter / Enable – enables the parts counter; this function counts how many parts already have been processed and informs the user when a limit was reached; in this case the user has to confirm this limit and needs to reset the counter to continue marking; confirmation and resetting can be done in mark dialogue

- Default part number – here a number of default parts can be specified that is used for the parts counter, it can be overwritten by the user in mark dialogue and requires the parts counter option to be enabled

- Power down laser / Enable – enables a function which gives the possibility to power down a laser when it is unused for a given time

- Power down idle time-out – here a value in seconds can be set which specifies after what idle time a connected laser has to be turned off; this requires the power-down option to be enabled

- Trace log – similar to the Smart Interface (as described in section "3.13 Smart Interface"), this option lets the software write trace information into a log file. This log file can be specified here and makes use of CSV format. Once enabled, all Elements, where the option "Trace data" is enabled, are written into this log file in pairs of columns consisting of the name of the element and the assigned data that have been marked. For serial number elements, these assigned data are the serial number used for a marking operation

Tab-pane "Custom dot" contains configuration possibilities to set up one or more custom dots which then can be used in elements for creating geometries:

- Custom dot – gives the possibility to choose fro ma list of dot configurations which then can be edited using the parameters below

- Custom dot name – here the selected custom dot can be renamed to give it a unique, easy to identify name

- Custom Dot Style – the following parameters are responsible for defining a style for a customised dot layout that is used by several geometric elements; this drop-down box selects the type of geometry, here polygon (includes a circle) or spiral can be chosen to be used as replacement for simple dots

- Edges/Segments – specifies the number of the edges or segments the custom geometry has to consist of

- Outer Radius – sets the size of the custom geometry

- Inner Radius – this parameter is used only in case of a spiral, it specifies the inner size of it

- Slope – here the same is true, this parameter applies to a spiral only and can be used to set the steepness the spiral grows with

- Draw CCW – here the drawing direction of the custom geometry can be inverted

- Inside To Outside – when this option is set for a spiral, it is drawn starting from its inner point

All parameters from this dialogue are stored within local default settings and within every .BEAMP project file. Thus it may happen a project file is loaded that contains different hardware settings and therefore is not compatible to the local system. To avoid such troubles two functions can be found in menu "Project" that handle the local default data:

- Save as default configuration – saves the current set up as default, whenever the application is started or a new project is generated, this configuration is used automatically; this not only includes the settings done within the project settings but also the number of pens and pen settings

- Load default configuration – loads a saved default configuration, overwrites the current project settings and appends the loaded pen definitions to the currently used ones

3.4.1   Scanner Controller Plug-Ins

This section describes all output-related plug-ins. Although the main category is "scanner controller" this can also include tools and output devices which not necessarily make use of laser scanner controllers only. Such a plug-in can be chosen and configured in general project settings in tab-pane "Scanner".

3rdEye(R) PSC140P:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used to access the 3rdEye PSC140P scanner controller card for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.
Basic:

- Use Board Number – in case more than only one board is used within the same host computer this value specifies which of the boards has to be used by this plug-in

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of the PSC140P and to the manuals of the laser device vendor

- Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

- Correction file – the .pcf correction file to be used for operation (provided by the vendor of the scan head)

- Firmware file – the .hex firmware file to be used to operate the scanner controller card (provided by 3rdEye)

- Stand By Frequency – frequency the laser has to be driven with in standby mode

- Stand By Length – pulse length the laser has to be driven with in standby mode

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates

- On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.
This plug-in is located in flowplugins/libio_psc140p.

G-Code Controller (Ethernet):

WARNING: This plug-in is designed to control laser and other equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

This plug-in can be used to access laser-controller or other similar devices that accept G-Code (CNC) data transferred via Ethernet. This is a generic plug-in which does not expect a specific device but provides generic G-Code commands conform to the standard which normally should work with all G-Code-compatible devices. Following parameters can be configured:

- IP and Port – IP and port number of the device to connect with

- Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode

- Laser type – here the type of used laser has to be set; this parameter corresponds to the option of custom M7xx commands described below

Laser Type	Signals
Generic laser	This is for generic laser types which do not require special control signals and therefore do not make use of any of the M7xx commands; this mode has to be used when the connected controller handles all laser signals itself
CO_2	stand-by frequency and length are used for initialisation (tickle-pulse after opening the device)
	default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by)
	additional definitions in "CO_2"-tab-pane
YAG Mode 1/2/3	stand-by frequency and length are used for initialisation (tickle-pulse after opening the device)
	default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by)
	first pulse killer supported at laser on (mode description see manual of E1701C)
Laser Mode 4	no predefined power control
	definition of continuously running frequency in "Laser Mode 4"-tab-pane
	continuously running frequency can be changed by parameter at beginning of mark
SPI (HS)	stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency)
	frequency can be changed by parameter at beginning of mark
	default power control via A0 analogue output
	simmer control requires laser controller plug-in and serial interface communication
	additional definitions in "SPI"-tab-pane
MOPA (MO, Latch, LP8)	to be used for digitally controlled MOPA laser types which require main oscillator and power control via latched LP8 port
	stand-by frequency is used for initialisation and running independent from jump/mark

Laser Type	Signals
	(continuously running frequency with 50% pulse-width at LaserA output)
	frequency can be changed by parameter at beginning of mark
	default power control via latched LP8 digital outputs
	additional definitions in "MOPA" tab-pane
Generic mill	To be used when no laser but a mechanical tool like a mill is used
	no setting of power, frequencies or laser delays necessary
	maximum speed for Z axis limits speed of tool when diving into the material
	maximum increment for Z axis (in mm) can be set for automatic calculation of iterations that are necessary for milling into material of a specified thickness

- Number of axes – specifies if 2D or 3D data have to be submitted to the device

- Working area – defines the upper left coordinate of the field that has to be used by the scanner and the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Custom initialisation commands – here it is possible to add own, device-specific G-Code commands that are sent during initialisation of the device

- Job start commands – here it is possible to add own, device-specific G-Code commands that are sent to the device at the beginning of each marking cycle

- Tool change commands – here it is possible to add own, device-specific G-Code commands that are sent to the device whenever the pen is changed; this command is typically used in milling mode and can be used to invoke a tool-change at the machine in order to reflect the different pen

- Laser on commands – commands configured here are sent to the device each time the laser has to be turned on; this function can be used to configure own, custom sequences to switch on the connected output device which can be – dependent on the chosen mode – a mill or a laser

- Laser off commands – commands configured here are sent to the device each time the laser has to be turned off; this function can be used to configure own, custom sequences to switch off the connected output device which can be – dependent on the chosen mode – a mill or a laser

- Jump mode – here a mode can be chosen to perform fast jumps to a position while the laser or tool is turned off. It can be chosen between G-Code command "G00" which performs an as fast as possible movement, and "Jump speed" which performs a normal, linear movement using the jump speed specified below

- Include custom M7xx laser commands – when this option is set, some laser-specific commands are included into the generated G-Code which contain parameters and settings which are required for and specific to laser marking operations. Following commands are supported:
M700 – the laser type, here a decimal number has to be set which specifies the laser type in detail

Laser type	M700 decimal parameter	Corresponding hexadecimal value
CO2	1207959553	0x48000001
YAG1 (Q-Switch and FPK at the beginning)	3355443202	0xC8000002
YAG2 (Q-Switch after FPK at the beginning)	2818572291	0xA8000003
YAG3 (Q-Switch after variable Q-Time)	2818572292	0xA8000004
CRF (Continuously running frequency)	1073741829	0x40000005

MOPA	1207959560	0x48000008

M701 – laser frequency in unit Hz (generic laser types only)
M702 – laser power in unit %*1000 (generic and MOPA lasers only)
M703 – laser on delay (A) and laser off delay (B) in unit µsec
M704 – jump speed (A) and mark speed (B) unit mm/min
M705 – jump delay (A), mark delay (B) and in-polygon delay (C) in unit µsec
M706 – reserved for future use
M707 – working area position X and Y in unit specified by G20/G21 command
M708 – jump delay (A), mark delay (B) and variable in-polygon delay (C) in unit µsec (maximum delay at 180 degrees angle)
M709 – working area size in X, Y and Z direction in unit specified by G20/G21 command
M710 – pulselength in unit nsec (MOPA and CRF lasers with variable pulse width only)
M711 – wobble amplitude in X-direction (X) in unit µm, wobble amplitude in Y-direction (Y) in unit µm, wobble frequency (C) in unit mHz (Hz*1000)
M712 – reserved for future use
M713 – FPK (first pulse) in unit nsec (YAG lasers only)
M714 – simmer current in unit %*1000 (SPI laser only)
M715 – stand-by frequency (A) in unit Hz and stand-by pulsewidth (B) in unit nsec (CO_2 and YAG lasers only)
M716 – reserved for future use
M717 – waveform number (SPI laser only)
M718 – laser frequency in unit Hz (A) and laser pulsewidth in unit µsec (B, needs to fit to period of current laser frequency)
M719 – switch main oscillator (MOPA lasers only), 1 for on, 0 for off
M720 .. M799 – reserved for future use

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates in relation to output

This plug-in is located in flowplugins/libio_sc_gcode.

G-Code Controller (Serial):

WARNING: This plug-in is designed to control laser and other equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

This plug-in can be used to access laser-controller or other similar devices that accept G-Code (CNC) data transferred via serial interface. This plug-in is identical to the previous one (G-Code Controller (Ethernet)) except the following functionality and features:

- serial communication – instead of an IP and port number a serial port is used for communication, thus in general settings tab-pane instead of IP and port number the name and parameters of the serial port have to be configured

- Use GRBL syntax – when this option is set, the plug-in expects to receive responses from the connected controller/sends control commands to the controller in GRBL format

- Position tolerance – when operating in GRBL-mode, two conditions have to be met in order to assume a motion operation is at its end: GRBL returns "Idle" and the returned XY(Z) position is equal to the last requested position. Depending on the resolution and stepwidth of the connected motor it may be possible GRBL is not able to exactly reach the intended target position because it would be located between two steps. For such cases a position tolerance has to be defined. The value given here in +- mm specifies how much the actual position returned from GRBL is allowed to differ from the last given, nominal position while still being a valid target position.

This plug-in is located in flowplugins/libio_sc_gcode_serial.

HALaser E1701A Scanner Controller:

WARNING: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after any software update.

This plug-in can be used to access the HALaser Systems E1701A scanner controller cards for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.

Basic configuration parameters:

- IP or Serial Interface - here it has to be configured how and where the controller is accessible. In case of Ethernet connection the IP (in style aaa.bbb.ccc.ddd) has to be entered here, in case of USB-connection the name of the serial interface has to be given ("COMx" for Windows, "/dev/tty-ACMx" for Linux where "x" is the number of the serial interface the controller was connected with)

- Ethernet Password – this parameter has to be given only in case Ethernet connection is used; here a password can be specified in order to identify connection to a controller. The same password has to be set within the related E1701 scanner controller. When password specified here and password configured at connected controller do not fit to each other, connection is closed and plug-in is not able to use this controller.

- Analogue voltage range – the E1701A is used to control analogue scanheads. Using this parameter the voltage range this scanhead works with can be selected in range 2,5..10 V (equal to +-2,5 V +-10 V control voltage at the cards outputs)

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of E1701 controller and to the manuals of the laser device vendor; within the plug-in the following laser types are available which make use of different parameters:

Laser Type	Signals
CO_2	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - additional definitions in "CO_2"-tab-pane
YAG Mode 1/2/3	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - first pulse killer supported at laser on (mode description see manual of E1701)
Laser Mode 4	- no predefined power control - definition of continuously running frequency in "Laser Mode 4"-tab-pane - continuously running frequency can be changed by parameter at beginning of mark

Laser Type	Signals
SPI (HS)	• stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency) • frequency can be changed by parameter at beginning of mark • default power control via A0 analogue output • simmer control requires laser controller plug-in and serial interface communication • additional definitions in "SPI"-tab-pane
MOPA (MO, Latch, LP8)	• stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency with 50% pulse-width at LaserA output) • frequency can be changed by parameter at beginning of mark • default power control via latched LP8 digital outputs • additional definitions in "MOPA"-tab-pane

• Additional power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here; this is an additional value which does not override or replace the default of the selected laser type.
PLEASE NOTE: the possible outputs listed here require specific variants of E1701 scanner card and/or specific extension boards which are optional. Please refer to E1701 scanner controller manual for details which outputs are available <u>together</u> with base- and extension boards.

• Bitmap marking mode – using this combobox the marking mode for scanner bitmaps can be chosen; "Fast" uses the jump speed as long as the laser is turned off within a bitmap and moves with mark speed continuously elsewhere, "Jump'n'shoot" jumps to a pixel position, then shoots with the laser and jumps to the next, "Laser gate modulated" does not make use of any power outputs but turns on/off the laser gate (black and white marking only)

• A0 Correction Factor: this parameter applies to analogue output A0 of LP8 Extension-board only. It can be used for analogue output voltages in range 0..5V but requires an external power supply which provides 5V exactly (voltages provided via USB are sometimes below of 5V). This factor can be used to adjust the output voltage by giving a correction factor. It of course can not pull the A0 output up to 5V when power supply delivers less than 5V but it can be used to adjust voltages below this limit.

• Correction file – a correction file to be used for operation (provided by the vendor of the scan head). Here several formats are supported by the card directly and can be used without any conversion.

• Stand By Frequency – frequency the laser has to be driven with in standby mode (depends on selected laser type); if applicable this frequency is emitted after initialisation of the plug-in

• Stand By Length – pulse length the laser has to be driven with in standby mode (depends on selected laser type); if applicable this pulse width is used after initialisation of the plug-in

• Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

• Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

• Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

• Mirror X / Mirror Y – mirror the input coordinates in relation to output

• Gain X / Gain Y – linear correction factor in X and Y direction to stretch or shrink the output

• Rotation – rotates the output by the given angle (in unit degree) using the centre of the working area as rotation point

• Slant X / Slant Y – trapezoidal correction angle along X and Y axis

- Use A3 as Digi Out – when this option is set, the analogue output A3 is operated as it would be a digital output

3D:

- Use third axis – this checkbox enables 3D marking mode

- Field depth – the Z-size of the field in unit mm; this value is used for displaying the correct working area in RepliSLS3D and for calculation of the additional Z correction

- Z distance – the distance between last mirror of the scanhead and middle position of the working area in unit mm; this value is used for calculation of the additional Z correction

Signals:

- Mark-In-Progress output – specifies a digital output to be used as "Mark in progress" signal. When this option is set, the related output is set to HIGH as long as a marking operation is running. This option can be used only together with Digi I/O Extension board.

- Wait-External-Trigger output – specifies a digital output to be used as "Wait for external trigger" signal. When this option is set, the related output is set to HIGH as long as the controller has halted operation because it is waiting for an ExtStart signal.

- Laser Ready Input – specifies a digital input to be used as "Laser Ready" signal. When this option is set, marking is possible only in case a HIGH signal is detected at the input specified here. This option can be used only together with Digi I/O Extension board.

- Invert LaserGate – inverts the output logic of the LaserGate output

- Invert LaserA – inverts the output logic of the LaserA output

- Invert LaserB – inverts the output logic of the LaserB output

- Marking on-the-fly – can be used to enable marking on-the-fly operations, here the desired mode can be chosen; "1D" makes use of one encoder input (two signal lines) and compensates in one direction, here the direction and strength is specified by one of the following parameters, "2D" makes use of two encoder inputs (four signal lines) and compensates in both directions X and Y, here the strength is specified by the two following parameters, "Simulated" is intended to be used for testing purposes, here MOTF-functionality is available without an external encoder signal, it is generated by the controller card internally

- On-the-fly factor X / On-the-fly factor Y – specifies the encoder factor to be used along the related axis, the value has to be given in unit µm/inc (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.

This plug-in is located in flowplugins/libio_e1701a.

HALaser E1701D Scanner Controller:

WARNING: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after any software update.

This plug-in can be used to access the HALaser Systems E1701D scanner controller cards for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.

Basic configuration parameters:

- IP or Serial Interface – here it has to be configured how and where the controller is accessible. In case of Ethernet connection the IP (in style aaa.bbb.ccc.ddd) has to be entered here, in case of USB-connection the name of the serial interface has to be given ("COMx" for Windows, "/dev/tty-ACMx" for Linux where "x" is the number of the serial interface the controller was connected with)

- Ethernet Password – this parameter has to be given only in case Ethernet connection is used; here a password can be specified in order to identify connection to a controller. The same password has to be set within the related E1701 scanner controller. When password specified here and password configured at connected controller do not fit to each other, connection is closed and plug-in is not able to use this controller.

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of E1701 controller and to the manuals of the laser device vendor; within the plug-in the following laser types are available which make use of different parameters:

Laser Type	Signals
CO_2	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - additional definitions in "CO_2"-tab-pane
YAG Mode 1/2/3	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - first pulse killer supported at laser on (mode description see manual of E1701)
Laser Mode 4	- no predefined power control - definition of continuously running frequency in "Laser Mode 4"-tab-pane - continuously running frequency can be changed by parameter at beginning of mark
SPI (HS)	- stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency) - frequency can be changed by parameter at beginning of mark - default power control via A0 analogue output - simmer control requires laser controller plug-in and serial interface communication - additional definitions in "SPI"-tab-pane

Laser Type	Signals
MOPA (MO, Latch, LP8)	• stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency with 50% pulse-width at LaserA output) • frequency can be changed by parameter at beginning of mark • default power control via latched LP8 digital outputs • additional definitions in "MOPA"-tab-pane

- Additional power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here; this is an additional value which does not override or re-place the default of the selected laser type. PLEASE NOTE: the possible outputs listed here require specific variants of E1701 scanner card and/or specific extension boards which are optional. Please refer to E1701 scanner controller manual for details which outputs are available together with base- and extension boards.

- Bitmap marking mode – using this combobox the marking mode for scanner bitmaps can be chosen; "Fast" uses the jump speed as long as the laser is turned off within a bitmap and moves with mark speed continuously elsewhere, "Jump'n'shoot" jumps to a pixel position, then shoots with the laser and jumps to the next, "Laser gate modulated" does not make use of any power outputs but turns on/off the laser gate (black and white marking only)

- A0 Correction Factor: this parameter applies to analogue output A0 of LP8 Extension-board only. It can be used for analogue output voltages in range 0..5V but requires an external power supply which provides 5V exactly (voltages provided via USB are often below of 5V). This factor can be used to adjust the output voltage by giving a correction factor. It of course can not pull the A0 output up to %v when power supply delivers less than 5V but it can be used to adjust voltages below this limit.

- Correction file – a correction file to be used for operation (provided by the vendor of the scan head). Here several formats are supported by the card directly and can be used without any conversion.

- Stand By Frequency – frequency the laser has to be driven with in standby mode (depends on selec- ted laser type); if applicable this frequency is emitted after initialisation of the plug-in

- Stand By Length – pulse length the laser has to be driven with in standby mode (depends on selec- ted laser type); if applicable this pulse width is used after initialisation of the plug-in

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates in relation to output

- Gain X / Gain Y – linear correction factor in X and Y direction to stretch or shrink the output

- Rotation – rotates the output by the given angle (in unit degree) using the centre of the working area as rotation point

- Slant X / Slant Y – trapezoidal correction angle along X and Y axis

3D:

- Use third axis – this checkbox enables 3D marking mode

- Field depth – the Z-size of the field in unit mm; this value is used for displaying the correct working area in RepliSLS3D and for calculation of the additional Z correction

- Z distance – the distance between last mirror of the scanhead and middle position of the working area in unit mm; this value is used for calculation of the additional Z correction

Signals:

- Mark-In-Progress output – specifies a digital output to be used as "Mark in progress" signal. When this option is set, the related output is set to HIGH as long as a marking operation is running. This option can be used only together with Digi I/O Extension board.

- Wait-External-Trigger output – specifies a digital output to be used as "Wait for external trigger" signal. When this option is set, the related output is set to HIGH as long as the controller has halted operation because it is waiting for an ExtStart signal.

- Laser Ready Input – specifies a digital input to be used as "Laser Ready" signal. When this option is set, marking is possible only in case a HIGH signal is detected at the input specified here. This option can be used only together with Digi I/O Extension board.

- Invert LaserGate – inverts the output logic of the LaserGate output

- Invert LaserA – inverts the output logic of the LaserA output

- Invert LaserB – inverts the output logic of the LaserB output

- Turn off power during jumps – this parameter is relevant for lasers with an unusual high leakage that can't be turned off by LaserGate completely. This typically results in thin marking lines for all movements where the laser should have been turned off. When this option is set, the current power is set to 0 before a jump is performed in order to minimize that leakage during scanner movement with LaserGate at LOW (means no emission). This ensures proper marking result also with such faultily lasers, but extends the marking time (up to 10 usec for every jump)

- Halt with ExtStart – when this option is checked, the ExtStart-input is used to halt the current marking operation. To allow marking, ExtStart needs to be at HIGH. When it goes to LOW, marking is halted the next time the laser is turned of (typically at the next jump). When ExtStart goes back to HIGH, markign is continued at the position where it was halted before except the whole operation was stopped in the meantime. When this mode is enabled, external triggers still can be applied, as they react on a rising edge at ExtStart, the input signal has to go to LOW and back to HIGH in such a case.

- Marking on-the-fly – can be used to enable marking on-the-fly operations, here the desired mode can be chosen; "1D" makes use of one encoder input (two signal lines) and compensates in one direction, here the direction and strength is specified by one of the following parameters, "2D" makes use of two encoder inputs (four signal lines) and compensates in both directions X and Y, here the strength is specified by the two following parameters, "Simulated" is intended to be used for testing purposes, here MOTF-functionality is available without an external encoder signal, it is generated by the controller card internally

- On-the-fly factor X / On-the-fly factor Y – specifies the encoder factor to be used along the related axis, the value has to be given in unit µm/inc (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.

This plug-in is located in flowplugins/libio_e1701.

HALaser E1803D Scanner Controller:

WARNING: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after any software update.

This plug-in can be used to access the HALaser Systems E1803D scanner controller cards for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.

Basic configuration parameters:

- IP or Serial Interface – here it has to be configured how and where the controller is accessible. In case of Ethernet connection the IP (in style aaa.bbb.ccc.ddd) has to be entered here, in case of USB-connection the name of the serial interface has to be given ("COMx" for Windows, "/dev/tty-ACMx" for Linux where "x" is the number of the serial interface the controller was connected with)

- Ethernet Password – this parameter has to be given only in case Ethernet connection is used; here a password can be specified in order to identify connection to a controller. The same password has to be set within the related E1803 scanner controller configuration file. When password specified here and password configured at connected controller do not fit to each other, connection is closed and plug-in is not able to use this controller.

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of E1803 controller and to the manuals of the laser device vendor; within the plug-in the following laser types are available which make use of different parameters:

Laser Type	Signals
CO_2	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - additional definitions in "CO_2"-tab-pane
YAG Mode 1/2/3	- stand-by frequency and length are used for initialisation (tickle-pulse after opening the device) - default power control and laser on/off via PWM control (longer pulses cause laser emission, short tickle pulses keep laser in stand-by) - first pulse killer supported at laser on (mode description see manual of E1803)
Laser Mode 4	- no predefined power control - definition of continuously running frequency in "Laser Mode 4"-tab-pane - continuously running frequency can be changed by parameter at beginning of mark
SPI (HS)	- stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency) - frequency can be changed by parameter at beginning of mark - default power control via A0 analogue output - default simmer control via A1 analogue output - additional definitions in "SPI"-tab-pane
MOPA (MO, Latch, LP8)	- stand-by frequency is used for initialisation and running independent from jump/mark (continuously running frequency with 50% pulse-width at LaserA output) - frequency can be changed by parameter at beginning of mark - default power control via latched LP8 digital outputs - additional definitions in "MOPA"-tab-pane

Laser Type	Signals
MOPA (MO, A0)	• to be used for analogue controlled MOPA laser types which require main oscillator and power control via analogue interface • default power control via A0 analogue output • RS232 serial interface connection used to read and reset error states • additional definitions in "MOPA"-tab-pane

- Additional power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here; this is an additional value which does not override or replace the default of the selected laser type.

- Bitmap marking mode – using this combobox the marking mode for scanner bitmaps can be chosen; "Fast" uses the jump speed as long as the laser is turned off within a bitmap and moves with mark speed continuously elsewhere, "Jump'n'shoot" jumps to a pixel position, then shoots with the laser and jumps to the next, "Laser gate modulated" does not make use of any power outputs but turns on/off the laser gate (black and white marking only)

- Correction file – a correction file to be used for operation (provided by the vendor of the scan head). Here several formats are supported by the card directly and can be used without any conversion.

- Stand By Frequency – frequency the laser has to be driven with in standby mode (depends on selected laser type); if applicable this frequency is emitted after initialisation of the plug-in

- Stand By Length – pulse length the laser has to be driven with in standby mode (depends on selected laser type); if applicable this pulse width is used after initialisation of the plug-in

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates in relation to output

- Gain X / Gain Y – linear correction factor in X and Y direction to stretch or shrink the output

- Rotation – rotates the output by the given angle (in unit degree) using the centre of the working area as rotation point

- Slant X / Slant Y – trapezoidal correction angle along X and Y axis

3D:

- Use third axis – this checkbox enables 3D marking mode

- Field depth – the Z-size of the field in unit mm; this value is used for displaying the correct working area in RepliSLS3D and for calculation of the additional Z correction

- Z distance – the distance between last mirror of the scanhead and middle position of the working area in unit mm; this value is used for calculation of the additional Z correction

Signals:

- Mark-In-Progress output – specifies a digital output to be used as "Mark in progress" signal. When this option is set, the related output is set to HIGH as long as a marking operation is running.

- Wait-External-Trigger output – specifies a digital output to be used as "Wait for external trigger" signal. When this option is set, the related output is set to HIGH as long as the controller has halted operation because it is waiting for an ExtStart signal.

- Laser Ready Input – specifies a digital input to be used as "Laser Ready" signal. When this option is set, marking is possible only in case a HIGH signal is detected at the input specified here.

- Invert LaserGate – inverts the output logic of the LaserGate output

- Invert LaserA – inverts the output logic of the LaserA output

- Invert LaserB – inverts the output logic of the LaserB output

- Turn off power during jumps – this parameter is relevant for lasers with an unusual high leakage that can't be turned off by LaserGate completely. This typically results in thin marking lines for all movements where the laser should have been turned off. When this option is set, the current power is set to 0 before a jump is performed in order to minimize that leakage during scanner movement with LaserGate at LOW (means no emission). This ensures proper marking result also with such faultily lasers, but extends the marking time (up to 10 usec for every jump)

- Halt with ExtStart – when this option is checked, the ExtStart-input is used to halt the current marking operation. To allow marking, ExtStart needs to be at HIGH. When it goes to LOW, marking is halted the next time the laser is turned of (typically at the next jump). When ExtStart goes back to HIGH, markign is continued at the position where it was halted before except the whole operation was stopped in the meantime. When this mode is enabled, external triggers still can be applied, as they react on a rising edge at ExtStart, the input signal has to go to LOW and back to HIGH in such a case.

- Marking on-the-fly – can be used to enable marking on-the-fly operations, here the desired mode can be chosen; "1D" makes use of one encoder input (two signal lines) and compensates in one direction, here the direction and strength is specified by one of the following parameters, "2D" makes use of two encoder inputs (four signal lines) and compensates in both directions X and Y, here the strength is specified by the two following parameters, "Simulated" is intended to be used for testing purposes, here MOTF-functionality is available without an external encoder signal, it is generated by the controller card internally

- On-the-fly factor X / On-the-fly factor Y – specifies the encoder factor to be used along the related axis, the value has to be given in unit µm/inc (this is typically not needed in 3D SLS applications)

- A0/A1 0% voltage – these fields can be used to define the allowed voltage range for A0 and A1 analogue outputs. Here 0% defines the output voltage which has to be set at 0% power values.

- A0/A1 100% voltage – these fields can be used to define the allowed voltage range for A0 and A1 analogue outputs. Here 0% defines the output voltage which has to be set at 100% (means full) power values.

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.

This plug-in is located in flowplugins/libio_e1803.

Makeblock XY-Plotter:

Warning: This plug-in is designed to control laser or other dangerous equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

This plug-in can be used to control a Makeblock XY-plotter by sending appropriate G-Code commands to it via serial interface. It is independent from the tool used at this plotter, which can be a laser or any other hardware that can be toggled by straight on/off signals. The plug-in provides a basic configuration panel where the serial interface parameters can be set according to the specifications of the laser to be connected.

Within a second panel the default mark-speed to be used can be configured.

This plug-in is located in flowplugins/libio_makeblock_xy.

Printer driver controlled:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used for all scanner controllers that can be accessed via a standard system printer driver. So within the configuration dialogue no laser/scanner related parameters can be set but values for line width, orientation and others:

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates

This plug-in is located in flowplugins/libio_sc_printer.

SCANLAB(R) RTC ScanAlone:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and whenever a software update was installed.

This plug-in can be used to access the SCANLAB RTC ScanAlone scanner controller card for controlling a scan head and a connected laser in real-time. This plug-in supports the direct operation mode only, no writing of laser marking data onto the card or operating in stand-alone mode is supported.
Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.
Basic:

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of the RTC ScanAlone and to the manuals of the laser device vendor

- Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

- Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode, this option depends on the capabilities of the used scanner controller card

- Correction file – the .ctb correction file to be used for operation (provided by the vendor of the scanhead)

- Stand By Frequency – frequency the laser has to be driven with in standby mode

- Stand By Length – pulse length the laser has to be driven with in standby mode

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates

- On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.

This plug-in is located in flowplugins/libio_rtcscanalone.

SCANLAB(R) RTC4:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used to access the SCANLAB RTC4 scanner controller card for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.
Basic:

- Use Board Number – in case more than only one board is used within the same host computer this value specifies which of the boards has to be used by this plug-in

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of the RTC4 and to the manuals of the laser device vendor

- Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

- Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode, this option depends on the capabilities of the used scanner controller card

- Correction file – the .ctb correction file to be used for operation (provided by the vendor of the scan head)

- Firmware file – the .hex firmware file to be used to operate the scanner controller card (provided by SCANLAB)

- Stand By Frequency – frequency the laser has to be driven with in standby mode

- Stand By Length – pulse length the laser has to be driven with in standby mode

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

– Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

– Mirror X / Mirror Y – mirror the input coordinates

– On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card (this is typically not needed in 3D SLS applications)

Default:

Here several default parameters can be defined that are used in case the direct axis control inputs X, Y and Z are used or in case control data are used that do not contain motion/speed/delay information

– Jump Speed – the speed the scanners position has to be changed with when the laser is turned off

– Mark Speed – the speed the scanners position has to be changed with when the laser is turned on

– Laser Off Delay / Laser On Delay / Jump Delay / Mark Delay / Polygon Delay – these delays are used during marking operation, they become active when the laser is turned off or on, when the scanner jumps to an other position, when it moves to an other position while marking or when the movement direction is changed during marking; for a detailed description of these values, their usage and their influence to the marking result please refer to the RTC4 manual

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.
This plug-in is located in flowplugins/libio_rtc4.

SCANLAB(R) RTC5:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used to access the SCANLAB RTC5 scanner controller card for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.
Basic:

– Use Board Number – in case more than only one board is used within the same host computer this value specifies which of the boards has to be used by this plug-in

– Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of the RTC5 and to the manuals of the laser device vendor

– Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

– Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode, this option depends on the capabilities of the used scanner controller card

– Correction file – the .ct5 correction file to be used for operation (provided by the vendor of the scan head)

– Firmware data directory – the directory that contains the firmware files to be used to operate the scanner controller card (provided by SCANLAB)

– Stand By Frequency – frequency the laser has to be driven with in standby mode

– Stand By Length – pulse length the laser has to be driven with in standby mode

– Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

– Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

– Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

– Mirror X / Mirror Y – mirror the input coordinates

– On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.
This plug-in is located in flowplugins/libio_rtc5.

SCANLAB(R) RTC6:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used to access the SCANLAB RTC6 scanner controller card for controlling a scan head and a connected laser in real-time. Within the configuration dialogue the plug-in and the scanner controller card have to be configured according to the connected devices.
Basic:

– Use Board Number – in case more than only one board is used within the same host computer this value specifies which of the boards has to be used by this plug-in

– Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the manual of the RTC6 and to the manuals of the laser device vendor

– Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

– Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode, this option depends on the capabilities of the used scanner controller card

– Correction file – the .ct5 correction file to be used for operation (provided by the vendor of the scan head)

– Firmware data directory – the directory that contains the firmware files to be used to operate the scanner controller card (provided by SCANLAB)

– Stand By Frequency – frequency the laser has to be driven with in standby mode

– Stand By Length – pulse length the laser has to be driven with in standby mode

– Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

– Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates

- On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.
This plug-in is located in flowplugins/libio_rtc6.

SCAPS(R) USC-1/2:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured. Beside of that some laser equipment can be damaged in case it is controlled with wrong signals. Thus it is highly recommended to check the output generated by this plug in using e.g. an oscilloscope to avoid problems caused by wrong configurations. This should be done prior to putting a system into operation for the first time and prior to putting a system into operation after a software update was installed.

This plug-in can be used to access the SCAPS USC1 and USC-2 scanner controller cards via the generic Scanner Controller Interface for controlling a scan head and a connected laser in real-time. The usage of this interface may require additional licenses provided by the vendor of the USC scanner controller cards.
Within the configuration dialogue the plug-in, the scanner controller card and the interface to the card have to be configured according to the connected devices.
Basic:

- Laser type – the type of the connected laser, the field that was selected here directly corresponds to the laser tab panes where some additional, laser-specific settings can be done; for a detailed description of the available options please refer to the SCAPS manuals and to the manuals of the laser device vendor

- Power output – the laser output power can be controlled via an additional output; the kind of output to be used can be defined here

- Number of Axes – specifies if the scanner controller has to operate in 2D or 3D mode, this option depends on the capabilities of the used scanner controller card and on the available SCAPS licenses

- Correction file – the .ucf correction file to be used for operation (provided by the vendor of the scan head or by SCAPS)

- Stand By Frequency – frequency the laser has to be driven with in standby mode

- Stand By Length – pulse length the laser has to be driven with in standby mode

- Field Left Position / Field Upper Position – defines the upper left coordinate of the field that has to be used by the scanner

- Field Size – the total size of the field that is used by the scanner, this parameter together with the left and upper position specifies the coordinates input vectors have to be located within, in case there are input data which define a position outside this field they will not be marked

- Swap X and Y – swaps the input X and Y coordinate to exchange X and Y axis values of the scanner

- Mirror X / Mirror Y – mirror the input coordinates

- On-the-fly factor X / On-the-fly factor Y – specifies the factor in X or Y direction for marking on-the-fly operation; this option depends on the capabilities of the used scanner controller card and on the available SCAPS licenses (this is typically not needed in 3D SLS applications)

All following tab-panes contain laser-specific definitions that are used depending on the selected laser type. For information about these parameters and their usage please refer to the manuals and specifications of the used lasers.

This plug-in is located in flowplugins/libio_samsci.

3.4.2   Motion Controller Plug-Ins

Following these plug-ins are described, which can be used to control external motion stages. RepliSLS3D supports up to two different motion controllers with up to four axes each – which sums up to up to eight independent axes which can control many different external devices. Motion plug-ins can be configured in tab-pane "Motion" of global settings.

HALaser E1701M Stepper Motor Controller:

Using this plug-in up to four axes of E1701M stepper motor controller by HALaser Systems can be accessed. These axes can be moved independently. The controller can be accessed either via USB serial interface or via Ethernet.

Within the configuration panels of the plug-in there are two major set-up possibilities. The first panel provides general settings and parameters that apply to the plug-in and controller globally:

- IP or Serial Interface – here it has to be configured how and where the controller is accessible. In case of Ethernet connection the IP (in style aaa.bbb.ccc.ddd) has to be entered here, in case of USB-connection the name of the serial interface has to be given ("COMx" for Windows, "/dev/tty-ACMx" for Linux where "x" is the number of the serial interface the controller was connected with)

- Ethernet Password – this parameter has to be given only in case Ethernet connection is used; here a password can be specified in order to identify connection to a controller. The same password has to be set within the related E1701M motion controller. When password specified here and password configured at connected controller do not fit, connection is closed and plug-in is not able to use this controller.

- Swap X and Y – this option is valid only in case the plug-in is fed with coordinates and not with separate position values for each axis, when it is set, X and Y coordinate are exchanged. When plug-in is used out of RepliSLS3D, this option is inoperable

The settings within the axis-panels apply for one specific axis only. Here different settings for each axis can be given:

- Enable Axis – an axis is used and can be moved by this plug-in only in case this checkbox is set.

- Axis Mode – an axis can be operated in two modes: planar or radial. In planar mode it moves over a distance (using unit mm), in angular mode it moves to an angle (in unit degrees)

- Resolution Factor – this value has to be used to specify conversion factor from shown position value (mm in planar mode, degrees in angular mode) to incremental position of the used motor (depending on its incremental resolution, step-configuration of used driver and used gearbox)

- Low Limit – the lower limit (in unit mm) is a soft limit specifying the minimum position the axis is allowed to move to in planar mode; in case of angular mode this value is ignored.
PLEASE NOTE: when an axis was not referenced yet, this limit is useless since the real position of axis is not known to E1701M controller. So as very first an axis always should be referenced!

- High Limit – the higher limit (in unit mm) is a soft limit specifying the maximum position the axis is allowed to move to in planar mode; in case of angular mode this value is ignored.
PLEASE NOTE: when an axis was not referenced yet, this limit is useless since the real position of axis is not known. So as very first an axis always should be referenced!

- Maximum Speed – this parameter can be used to limit maximum speed an axis can move with (in unit mm/sec for planar mode and deg/sec in angular mode). Independent from which speed value is specified during operation, this limit is never exceeded

- Acceleration Mode – specifies the shape of the ramp used during acceleration, here it can be chosen between linear (smooth acceleration), exponential (strong but not so smooth mode which may over-shoot at end of acceleration phase) or s-shaped acceleration (very smooth but not very fast accelerating)

- Acceleration – factor that specifies how strong acceleration is for chosen acceleration mode

- Deceleration Mode – specifies the shape of the ramp used during deceleration, here it can be chosen between linear (smooth deceleration), exponential (strong but not so smooth mode which may lose increments at beginning of deceleration phase) or s-shaped deceleration (very smooth but not very fast)

- Deceleration – factor that specifies how strong deceleration is for chosen deceleration mode

- Stop Deceleration – factor that specifies strength of deceleration for all stop events (stop by limit switch, reference switch or by stop-command)

- Invert Direction – all axis use a "positive" and "negative" movement direction which are used to describe the motion direction. When this option is set, "positive" and "negative" are exchanged which may be useful to make UI-layout/appearance of these directions more convenient to user

- Reference Mode – after first start-up or when a motion error occurred where increments may got lost, referencing is required for an axis. To do this, a reference switch is required at a defined mounting position (refer to related parameter below). Using reference mode parameter a sequence of movements in relation to this reference switch can be chosen that has to be performed on referencing.

- Reference Timeout – here a value in unit seconds can be specified. When referencing could not be finished after this time has elapsed, it will be cancelled since it seems axis is not able to find reference switch and/or movement is not possible for some reasons

- Reference Signal Input – here an input can be specified that is used as switch defining the referencing position

- Invert input logic – inverts the logic of the reference input to react on LOW level instead of HIGH

- Reference Speed 1/2/3 – referencing is done in up to three different steps (depending on used reference mode). Speeds for these steps should be slow in order to not to overshoot when reference position is reached. In case of more than one referencing step, these speed values should decrease for every step. Using these three parameters reference speed values can be specified. When a speed is given for a step that is not used in chosen reference mode, this value is ignored.

- Referenced Position – after referencing was finished successfully, the current axis position is set to the position value specified here

- Limit Switch 1/2 Input – beside the soft limits it is possible to define hardware-controlled position limits via limit switches. When such limit switches are used and connected to E1701M controller, the inputs they are connected with can be specified here. Whenever one of these switches sends a HIGH signal, current movement is stopped using stop-deceleration

- Invert input logic – inverts the logic of the limit switch input to react on LOW level instead of HIGH

- Auto-leave Limit Switch – when a limit switch is hit, normally no more motions are allowed until this switch is left. When this option is set and an axis hits a limit switch, it automatically moves back until this switch is left.
PLEASE NOTE: when this option is set, a possibly unexpected motion may start in opposite direction!

- Encoder – E1701M controller board supports up to two external quadrature encoders that provide position information. When such an encoder is connected to decoder input 0 (digital inputs 0 and 1) or decoder input 1 (digital inputs 2 and 3) these inputs can be enabled here for an axis. When an external encoder is activated newly. the axis position on start-up is set to 0. To specify a different position, reference mode "No referencing" has to be chosen, a referencing position has to be set and referencing has to be performed. This does not cause any referencing operation but immediately sets the current, encoder-based position to the one specified as "Referenced Position".
Alternatively a standard referencing operation as described above can be performed too.
PLEASE NOTE: an encoder can be used for one axis only, when the same decoder input is configured for more than one axis, the results are undefined.

- Encoder Factor – this factor is used for calculation of axis incremental position, here the ratio between encoder pulses and motor increments has to be specified

This plug-in is located in flowplugins/libio_e1701m_stepper.

IO Sync Motor Trigger (Scannercard):

This plug-in is not a motor driver in common sense. Instead of submitting motion and position information, it sends a digital signal to a connected device which then can be used to trigger external equipment. This digital signal stays at HIGH until the external device responds with an other signal. As soon as this signal is received, the software assumes the motion is completed and continues with other operations. This plug-in sends and receives the digital signals via a connected scanner controller card and its digital IOs.

The plug-in is able to control up to four axes via these digital on/off signals These axes will operate sequentially, two or more axes will not be triggered at the same time.

As global parameter only the digital output port and the digital input port of the connected scanner controller card have to be chosen. Here is is up to the user to select IO ports that really exist for the used scanner controller, in case an IO is specified that is not available, no data can be send to it or no input information can be received. Which bit of the selected IO's is used for which purpose and which axis, can be specified within the axis-related configuration panels:

- Motion start output – the output bit at the previously chosen output port type that is used to send the command to an external device to start a motion operation. This output stays at HIGH until the external device signals the motion operation was completed

- Motion done input – the input bit at the previously chosen input port where the external device signals the end of a motion operation

- Reference start output – the output bit at the previously chosen output port type that is used to send the command to an external device to start a referencing operation. This output stays at HIGH until the external device signals the referencing operation was completed

- Reference Timeout – a timeout value in seconds after what a referencing operation will be cancelled no matter if the connected device signalled a "motion done" or not

- Referenced Position – the metric position (in unit millimetres) that has to be set at the referenced position

This plug-in is located in flowplugins/libio_scanctrl_iosync.

Schneider/IMS MDrive+:

This plug-in is able to control up to three axes of a Schneider/IMS MDrive+; there is also a macro available that encapsulates the low level functionalities described here. These axes can be moved independently.

This drive can be controlled via a serial interface where all axes are connected to this line. Independent from the fact if one or more axes are used, all of the used drives have to be pre-configured in the following way before operating them within a RepliSLS3D environment. Following the configuration steps are described together with the commands that have to be sent to the drive to do this configuration:

1. A unique device identifier has to be set for each axis (`DN="X"` where "X" is the identifier).

2. The echo mode has to be set to mode "2" (`EM=2`).

3. The party mode has to be enabled (`PY=1`).

4. These parameters have to be stored to the drives permanently (`XS^J`).

5. When the homing/referencing functionality has to be used, the input S1 of the drive has to be connected to the home switch.

When this configuration was done successfully, the drives can be set up within the configuration panel of the plug-in. First of all the data of the serial interface have to be defined. Here the correct port name and the in-

terface settings have to be given in order to establish the communication to the drive. Next for every used axis several parameters have to be given:

- Axis Mode – specifies the operational mode of the drive, here out of the two options planar (expects linear input data) and rotational (expects angular input values in unit degree) can be chosen

- Factor – specifies a factor that is used to calculate a metric value in millimetres or an angular value in degrees out of the drive-internal position information that is given in unit "increments", the exact factor depends on your hardware set-up and the used gear

- Low Limit – specifies the lower limit in unit millimetres where the axis is allowed to move to at minimum

- High Limit – specifies the upper limit in unit millimetres where the drive is allowed to move to at maximum

- Axis Name – is the device name that was stored within the drive in order to identify it

- Axis Home Timeout – is a time-out value in seconds that described after what time referencing has to be stopped when no home switch can be found

When a movement is running at an axis it can't be overwritten by a new position value immediately, such a new position information would be rejected by the plug-in. In this case the axis first has to be stopped and after the related speed-output indicates that this axis really stopped (by emitting a speed value of 0) the new position can be set at the POS-input.

This plug-in is located in flowplugins/libio_ims_mdrive2.

Stepper Motor Driver (Scannercard):

Using this plug-in a stepper motor can be controlled via a connected scanner controller card. This plug-in does not control such a card directly but it generates output data that can be handled by a scanner controller card.

The plug-in is able to control up to three axes via two output signals "Step" and "Direction". These axes will operate sequentially, two or more axes will not move at the same time.

As global parameter only the digital output port and the digital input port of the connected scanner controller card have to be chosen. Here is is up to the user to select IO ports that really exist for the used scanner controller, in case an IO is specified that does not exists, no data can be send to it or no input information can be received. Which bit of the selected IO's is used for which purpose and which axis can be specified within the axis-related configuration panels:

- Axis Mode – specifies the operational mode of the drive, here out of the two options planar (expects linear input data) and rotational (expects angular input values in unit degree) can be chosen

- Factor – specifies a factor that is used to calculate a metric value in millimetres or a rotational value in degrees out of the drive-internal position information that is given in unit "increments", the exact factor depends on your hardware set-up and the gear you are using

- Low Limit – specifies the lower limit in unit millimetres where the axis is allowed to move to at minimum

- High Limit – specifies the upper limit in unit millimetres where the drive is allowed to move to at maximum

- Acceleration – a value that specifies how smooth the drive should start its movements

- Deceleration – a value that specifies how smooth the drive should stop its movements

- Step Signal Output – the output bit at the previously chosen output port type that is used for the pulsed step signal; here it has to be made sure that an output is used only once, overlapping mappings would cause problems during operation

- Direction Signal Output – the output bit at the output port that was chosen in general configuration panel, this output bit is used to define the movement direction of the axis; here it has to be made sure that an output is used only once, overlapping mappings would cause problems during operation

- Invert Direction – inverts the direction output / the movement direction for this axis

- Reference Mode – here it can be specified if a home/referencing operation has to be done or not and if yes which method of referencing has to be used; the chosen mode defines using which direction the software shall search for the reference switch and using which direction the switch shall be left afterwards

- Reference Signal Input – the input bit at the previously chosen input port where the reference switch is connected to

- Reference Speed (Enter) – the speed that has to be used to search for the reference switch

- Reference Speed (Leave) – the speed that has to be used to leave the reference switch once it was found; the correct referenced position is that position, where the reference switch just turns off

- Referenced Position – the metric position (in unit millimetres) that has to be set at the referenced position

This plug-in is located in flowplugins/libio_scanctrl_stepper.

3.4.3 Laser Controller Plug-Ins

Typically lasers are controlled by the used scanner controller card fully. But in case of some specific scanner cards which do not provide all required hardware interfaces or in case of specific lasers, which need additional control commands via a separate communication channel, a laser control plug-in can be used. It can be selected and configured in general application settings in tab-pane "Misc".

Coherent(R) Avia Ethernet Laser Controller:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

Using this plug-in a Coherent Avia laser can be controlled via Ethernet. To access the serial interface of the target laser an Ethernet to serial converter is required. The plug-in provides a basic configuration panel where the network parameters can be set according to the configuration of the TCP/IP to serial converter. A second panel gives the possibility to enter several laser-specific parameters. By default the plug-in can be used to set fixed frequency and power that are used all the time. On "Basic" configuration panel this behaviour can be changed. There are checkboxes to influence the behaviour of the plug-in:

- Control power during process – when this checkbox is set, the power-values defined in pen parameters are sent to the plug-in before the related vector data are sent to the scanner card

- Control frequency during process – when this checkbox is set, the frequency-values defined in pen parameters are sent to the plug-in before the related vector data are sent to the scanner card

This plug-in is located in flowplugins/libio_coherent_avia.

Generic Laser Controller:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

Using this plug-in a laser can be controlled via serial interface ASCII commands. It is able to connect to a laser controller via COM-port/serial interface or TCP/IP, send predefined commands to it and receive a response from the laser. The received data can be interpreted in a minimum way. The plug-in provides a basic configuration panel where the serial interface parameters can be set according to the specifications of the laser to be connected.
As parameter "Interface" following values can be set:

- "COMx" or "/dev/ttyx" in case a local serial interface has to be used, here "x" is the number of the serial port

- "U1" in case the serial interface of a connected scanner controller card has to be used, in this case all data are routed to and from this controller card

- IP in style "aaa.bbb.ccc.ddd" in case network connection via TCP/IP has to be used

A second panel gives the possibility to enter the commands that have to be sent to the laser. Here several commands can be submitted. Every command has to be separated by specific characters that specify carriage return and/or line feed. Here for a carriage return the special character sequence "`[CR]`" has to be added for a line feed "`[LF]`" can be used. Please note: also in case only one laser command has to be sent, the CR/LF termination characters have to be set at the end. Which characters have to be used depends on the requirements of the used laser. The "Commands" configuration panel offers following configuration possibilities:

- Response End Character – specifies which characters are expected in every response from the laser; if "none" is chosen, the plug-in tries to receive some data after every command transmission but does not check the contents

- Device Open Command – here the command can specified that is sent to the laser when everything is initialised and when all connected devices are opened

- Device Open Delay – this is an optional delay that can be specified to let the plug-in wait after the Device Open Command is sent and before all other operations (like opening of other devices) are continued; this can be useful to let the laser start up/settle completely

- Ready for Marking Command – this command is sent to the laser whenever it has to be ready for marking, it is issued e.g. in case RepliSLS3D opens the Mark dialogue

- Ready for Marking Delay – this is an optional delay that can be specified to let the plug-in wait after the Ready for Marking Command is sent and before all other operations (like start of marking operation) are continued; this can be useful to let the laser start up/settle completely

- Job Start Command – this command is sent to the laser prior to submission of first vector data to the scanner card

- Job Start Delay – this is an optional delay that can be specified to let the plug-in wait after the Job Start Command is sent and before sending of marking/vector data begins

- Laser On Command – this is a very specific command that should be used only in some very special applications, it is issued whenever the laser has to be turned on; this is done by the scanner card normally, so the user has to ensure that both commands don't overlap. Please note: Since serial communication is quite slow and not very responsive this way of turning on the laser is quite inaccurate, it should be used only when exact laser timing does not matter

- Laser On Delay – this is an optional delay that can be specified to let the plug-in wait after the Laser On Command is sent and before sending of marking/vector data begins

- Laser Off Command – this corresponds to the Laser On Command and is a very specific one that should be used only in some very special applications, it is issued whenever the laser has to be turned off; this is done by the scanner card normally, so the user has to ensure that both commands don't overlap. Please note: Since serial communication is quite slow and not very responsive this way of turning off the laser is quite inaccurate, it should be used only when exact laser timing does not matter

- Job End Command – this command is sent to the laser after submission of vector data to the scanner card has finished

- End Ready for Marking Command – this command is sent to the laser whenever it no longer needs to be ready for marking, it is issued e.g. in case RepliSLS3D closes the Mark dialogue

- Device Close Command – here the command can specified that is sent to the laser when everything is de-initialised and when all connected devices are closed; this command is invoked e.g. when the main application is closed

This plug-in is located in flowplugins/libio_lc_generic.

SPI G4 Laser RS232 Controller:

Warning: This plug-in is designed to control laser equipment which may effect a person's health or may otherwise cause damage. Prior to installation and operation compliance with all relevant safety regulations including additional hardware-controlled safety measures has to be secured.

Using this plug-in a SPI G4 laser can be controlled via serial interface ASCII commands. It is able to connect to such a laser controller via COM-port/serial interface, send predefined and freely definable commands to it and receive responses from the laser.

The plug-in provides a basic configuration panel where the serial interface parameters can be set according to the communication parameters of the lasers serial interface.

A second panel gives the possibility to specify behaviour of the plug-in and to enter custom commands that have to be sent to the laser. Every of these custom commands has to be separated by specific characters that specify carriage return and/or line feed. Here for a carriage return the special character sequence "[CR]" has to be added for a line feed "[LF]" can be used. Please note: also in case only one laser command has to be sent, the "[CR][LF]" termination characters have to be appended.

The "Commands" configuration panel offers following configuration possibilities:

- Device Open Command – here a custom command can specified that is sent to the laser when everything is initialised and when all connected devices are opened

- Device Open Delay – this is an optional delay that can be specified to let the plug-in wait after the Device Open Command is sent and before all other operations (like opening of other devices) are continued; this can be useful to let the laser start up/settle completely

- Ready for Marking Command – this custom command is sent to the laser whenever it has to be ready for marking, it is issued e.g. in case RepliSLS3D opens the Mark dialogue

- Ready for Marking Delay – this is an optional delay that can be specified to let the plug-in wait after the Ready for Marking Command is sent and before all other operations (like start of marking operation) are continued; this can be useful to let the laser start up/settle completely

- Job Start Command – this command is sent to the laser prior to submission of first vector data to the scanner card

- Job Start Delay – this is an optional delay that can be specified to let the plug-in wait after the Job Start Command is sent and before sending of marking/vector data begins

- Control power during process – when this checkbox is set, the plug-in controls the power of the laser, means it sends the related command to it automatically on every change of output power and before vector data are sent that would use this new power value

- Control waveform during process – when this checkbox is set, the plug-in controls the used waveform, means it sends the related command to it automatically on every change of waveform and before vector data are sent that would use it

- Job End Command – this optional, custom command is sent to the laser after submission of vector data to the scanner card has finished

- End Ready for Marking Command – this optional command is sent to the laser whenever it no longer needs to be ready for marking, it is issued e.g. in case RepliSLS3D closes the Mark dialogue

- Device Close Command – here an optional, custom command can specified that is sent to the laser when everything is de-initialised and when all connected devices are closed; this command is invoked e.g. when the main application is closed or when a user opens the global settings dialogue in RepliSLS3D

This plug-in is located in flowplugins/libio_lc_spi.

3.4.4   Z-Shifter/Focus-Shifter Plug-Ins

The following plug-ins can be used together with Z-shifter elements in order to change the focus/z-position via a specialised device at defined positions within a process control job (which itself is intended to perform control tasks between two slices of a marked 3D model). Plug-ins of this type can be configured in tab-pane "Misc" of the global application settings.

HALaser E1701M Focus Shifter Motor Controller:

Using this plug-in up one axis of E1701M stepper motor controller by HALaser Systems/OpenAPC Project Group can be accessed. It is intended to be used as Z-shifter/focus shifter primarily. For independent axis control of up to four axes please check description of E1701M Stepper Motor controller.

The controller can be accessed either via USB serial interface or via Ethernet.

Within the configuration panels of the plug-in there are two major set-up possibilities. The first panel provides general settings and parameters that apply to the plug-in and controller globally:

- IP or Serial Interface – here it has to be configured how and where the controller is accessible. In case of Ethernet connection the IP (in style aaa.bbb.ccc.ddd) has to be entered here, in case of USB-connection the name of the serial interface has to be given ("COMx" for Windows, "/dev/tty-ACMx" for Linux where "x" is the number of the serial interface the controller was connected with)

- Ethernet Password – this parameter has to be given only in case Ethernet connection is used; here a password can be specified in order to identify connection to a controller. The same password has to be set within the related E1701M motion controller. When password specified here and password configured at connected controller do not fit, connection is closed and plug-in is not able to use this controller.

The settings within the axis-panel applies for the Z-axis:

- Resolution Factor – this value has to be used to specify conversion factor from shown position value (mm in planar mode, degrees in angular mode) to incremental position of the used motor (depending on its incremental resolution, step-configuration of used driver and used gearbox)

- Low Limit – the lower limit (in unit mm) is a soft limit specifying the minimum position the axis is allowed to move to in planar mode; in case of angular mode this value is ignored.
  PLEASE NOTE: when an axis was not referenced yet, this limit is useless since the real position of axis is not known to E1701M controller. So as very first an axis always should be referenced!

- High Limit – the higher limit (in unit mm) is a soft limit specifying the maximum position the axis is allowed to move to in planar mode; in case of angular mode this value is ignored.
  PLEASE NOTE: when an axis was not referenced yet, this limit is useless since the real position of axis is not known. So as very first an axis always should be referenced!

- Maximum Speed – this parameter can be used to limit maximum speed an axis can move with (in unit mm/sec for planar mode and deg/sec in angular mode). Independent from which speed value is specified during operation, this limit is never exceeded

- Acceleration Mode – specifies the shape of the ramp used during acceleration, here it can be chosen between linear (smooth acceleration), exponential (strong but not so smooth mode which may overshoot at end of acceleration phase) or s-shaped acceleration (very smooth but not very fast accelerating)

- Acceleration – factor that specifies how strong acceleration is for chosen acceleration mode

- Deceleration Mode – specifies the shape of the ramp used during deceleration, here it can be chosen between linear (smooth deceleration), exponential (strong but not so smooth mode which may lose increments at beginning of deceleration phase) or s-shaped deceleration (very smooth but not very fast)

- Deceleration – factor that specifies how strong deceleration is for chosen deceleration mode

- Stop Deceleration – factor that specifies strength of deceleration for all stop events (stop by limit switch, reference switch or by stop-command)

- Invert Direction – all axis use a "positive" and "negative" movement direction which are used to describe the motion direction. When this option is set, "positive" and "negative" are exchanged which may be useful to make UI-layout/appearance of these directions more convenient to user

- Reference Mode – after first start-up or when a motion error occurred where increments may got lost, referencing is required for an axis. To do this, a reference switch is required at a defined mounting position (refer to related parameter below). Using reference mode parameter a sequence of movements in relation to this reference switch can be chosen that has to be performed on referencing.

- Reference Timeout – here a value in unit seconds can be specified. When referencing could not be finished after this time has elapsed, it will be cancelled since it seems axis is not able to find reference switch and/or movement is not possible for some reasons

- Reference Signal Input – here an input can be specified that is used as switch defining the referencing position

- Invert input logic – inverts the logic of the reference input to react on LOW level instead of HIGH

- Reference Speed 1/2/3 – referencing is done in up to three different steps (depending on used reference mode). Speeds for these steps should be slow in order to not to overshoot when reference position is reached. In case of more than one referencing step, these speed values should decrease for every step. Using these three parameters reference speed values can be specified. When a speed is given for a step that is not used in chosen reference mode, this value is ignored.

- Referenced Position – after referencing was finished successfully, the current axis position is set to the position value specified here

- Limit Switch 1/2 Input – beside the soft limits it is possible to define hardware-controlled position limits via limit switches. When such limit switches are used and connected to E1701M controller, the inputs they are connected with can be specified here. Whenever one of these switches sends a HIGH signal, current movement is stopped using stop-deceleration

- Invert input logic – inverts the logic of the limit switch input to react on LOW level instead of HIGH

- Auto-leave Limit Switch – when a limit switch is hit, normally no more motions are allowed until this switch is left. When this option is set and an axis hits a limit switch, it automatically moves back until this switch is left.
  PLEASE NOTE: when this option is set, a possibly unexpected motion may start in opposite direction!

- Encoder – E1701M controller board supports up to two external quadrature encoders that provide position information. When such an encoder is connected to decoder input 0 (digital inputs 0 and 1) or decoder input 1 (digital inputs 2 and 3) these inputs can be enabled here for an axis. When an external encoder is activated newly. the axis position on start-up is set to 0. To specify a different position, reference mode "No referencing" has to be chosen, a referencing position has to be set and referencing has to be performed. This does not cause any referencing operation but immediately sets the current, encoder-based position to the one specified as "Referenced Position".
  Alternatively a standard referencing operation as described above can be performed too.
  PLEASE NOTE: an encoder can be used for one axis only, when the same decoder input is configured for more than one axis, the results are undefined.

- Encoder Factor – this factor is used for calculation of axis incremental position, here the ratio between encoder pulses and motor increments has to be specified

Sill(R) Focus Ethernet:

This is a one-axis motion plug-in to access a Sill focus shifter via Ethernet. Since the hardware is controlled via serial interface an Ethernet to serial converter is required in order to use this plug-in. In first configuration panel the TCP/IP configuration of that converter has to be set. In second panel the motion parameters can be set the focus shifter has to be operated with. Here a conversion factor from distance (in unit µm) to internal representation of the focus shifter has to be given. Beside of that the upper and lower movement limit (in unit µm) can be specified as well.

This plug-in is located in flowplugins/libio_sill_focus.

3.4.5   Process Control Plug-Ins

Following section describes plug-ins and supported hardware which can be used to control different process parameters. Such a plug-in can be chosen in general project settings in tab-pane "Misc".

E1803D PID control loop:

This plug-in can be used together with the E1803D scanner controller card to access and use the integrated PID controllers of this scanner card. It can be used for controlling different parameters like temperature, pressure or concentration. The measured data have to be provided via the analogue inputs AIn0, AIn1 or AIn2 of this scanner card (this may require an appropriate extension board). These inputs directly correspond to the control loops PID0, PID1 and PID2 which can be configured separately:

- Enable PIDx/AInx – enable the related PID controller for usage

- Type of value – here it can be chosen which type of data have to be measured; this does not have an influence on the behaviour of the PID control loop but is used to display proper names and measurement units within the main application

- Value at 0V – specify the temperature which belongs to an analogue input value of 0V

- Value at 5V – specify the temperature which belongs to the maximum possible analogue input value of 5V

- Lower value – the lower threshold for halting a building operation, when the measured value falls below of this, marking is halted until measured value is back in range "lower value" .. "upper value"; this value can be overwritten by parameters specified in pen settings

- Upper value – the upper threshold for halting a building operation, when the measured value rises beyond of this, marking is halted until measured value is back in range "lower value" .. "upper value"; this value can be overwritten by parameters specified in pen settings

- P – proportional part of the PID control loop

- I – integral part of the PID control loop

- D – differential part of the PID control loop

- Default setpoint – the default setpoint to be used on initialisation, this value can be overwritten by parameters specified in pen settings

- Control output – here the output can be chosen which has to be used to emit an output value depending on the current measured input value and the state of the control loop. It can be chosen between two analogue outputs AOut0 and AOut1 and the digital outputs from DOut0 to DOut7. Due to its limitations to an on/off state, the digital output should be used only for simple, slowly reacting environments, elsewhere analogue outputs are to be preferred

- Mode – this option can be used to configure the control mode and is only relevant when analogue outputs have been chosen. Here following options are available:
  5V +/- regulation – in this mode a 5V output value at the analogue output means setpoint and measured value are identically and no modification is necessary in process parameters. When measured value falls below of the setpoint, the output is regulated to values smaller than 5V (decay). When measured value rises above of the setpoint, the output is regulated to values bigger than 5V (amplification)
  0V + regulation – in this mode a 0V output value at the analogue output means setpoint and measured value are identically and no modification is necessary in process parameters. When measured value falls below of the setpoint, the output stays at 0V. When measured value rises above of the setpoint, the output is regulated to values bigger than 0V (amplification)

In RepliSLS3D, the pen parameters are extended by an additional tab-pane "PID ctrl loop" that offers some parameters that can be set to individual values for each pen. Depending on the number of PID controllers

here up to three sets of parameters can appear. During marking these related values are sent to the plug-in whenever then pen changes modifying the current control loop and processing parameters:

- Lower value – the lower threshold for halting a building operation, when the measured value falls below of this, marking is halted until measured value is back in range "lower value" .. "upper value"; this value can be overwritten by parameters specified in pen settings

- Upper value – the upper threshold for halting a building operation, when the measured value rises beyond of this, marking is halted until measured value is back in range "lower value" .. "upper value"; this value can be overwritten by parameters specified in pen settings

- Default setpoint – the default setpoint to be used on initialisation, this value can be overwritten by parameters specified in pen settings

This plug-in is located in flowplugins/libio_e1803_pid.

Generic analogue temperature:

This plug-in can be used together with a scanner controller card that provides analogue input(s) which can be used for measuring temperatures and to react on the measured values accordingly. The behaviour of the plug-in can be configured. The first configuration-panel "Basic" provides following options:

- Analogue input port – here the port of the scanner controller card has to be selected where the temperature sensor is connected with. Please ensure the correct port is chosen here, since this plug-in is generic, it has no information about the scanner controllers capabilities and the analogue port which are really available, therefore also ports are available for selection, which may not exist on the current scanner card

- Temperature at 0 – specify the temperature which belongs to an analogue input value of 0

- Temperature at max – specify the temperature which belongs to the maximum possible analogue input value

The second panel "Thresholds" gives the possibility to define the behaviour of the plug-in via following options:

- Lower Temperature – here a default temperature can be set which is used to turn on an external device (e.g. a heater), when measured temperature falls below of the limit specified here, the device is turned on, this value can be overridden by pen parameters

- Upper Temperature – here a default temperature can be set which is used to turn off an external device (e.g. a heater), when measured temperature rises above the limit specified here, this device is turned off, this value can be overridden by pen parameters

- Additional Digital Output Port – this parameter can be used to configure the digital output port of a connected scanner controller card to be used for switching the external device
  Here several options are given which depend on the hardware capabilities of a connected scanner controller. For an E1803D controller card, the outputs are assigned as follows:
  - "Laserport 1 (8 bit)" is the laser port output on the laser interface of E1803D
  - "Digital 1 (8 bit)" can be found on the digital interface of the E1803D
  - "Digital 2 (8 bit)" can be found on the Intelli-IO extension of the E1803D
  - "Digital 1 (16 bit)" is not available on E1803D, it is e.g. the output of RTC scanner controller cards

- Additional Digital Output Bit – this parameter can be used to configure the pin number of the digital output port configured with previous parameter

- Stop Process on Temperature Deviation – when this option is set, a running marking process is stopped, when the temperature is above the upper temperature limit or below of the lower temperature limit byt the given amount of "Temperature deviation". The process is continued only when temperature is back within the given range (plus/minus the "Temperature deviation").

In RepliSLS3D, the pen parameters are extended by an additional tab-pane "Heating" that offers some parameters that can be set to individual values for each pen. During marking these related values are sent to the plug-in whenever then pen changes:

- Lower Temperature – a temperature which is used to turn on an external device (like a heater), when measured temperature falls below of the limit specified here, the device is turned on
- Upper Temperature – a temperature which is used to turn off an external device when measured temperature rises above the limit specified here

This plug-in is located in flowplugins/libio_generic_ana.

Optris LT Pyrometer:

This plug-in can be used together with an Optris LT pyrometer with serial interface to read temperature data from it and to react on the measured values accordingly. The behaviour of the plug-in can be configured. The first configuration-panel "Basic" provides possibilities to set-up the serial interface, here the same parameters have to be specified as the Optris LT serial port is running with. Also to be configured in the first panel is the protocol format to be used when communicating with the pyrometer, here the option to be selected depends on the exact device which is used.

The second panel "Thresholds" gives the possibility to define the behaviour of the plug-in via following options:

- Lower Temperature – here a default temperature can be set which is used to turn on an external device (e.g. a heater), when measured temperature falls below of the limit specified here, the device is turned on, this value can be overridden by pen parameters
- Upper Temperature – here a default temperature can be set which is used to turn off an external device (e.g. a heater), when measured temperature rises above the limit specified here, this device is turned off, this value can be overridden by pen parameters
- Additional Digital Output Port – this parameter can be used to configure the digital output port of a connected scanner controller card to be used for switching the external device
- Additional Digital Output Bit – this parameter can be used to configure the pin number of the digital output port configured with previous parameter
- Stop Process on Temperature Deviation – when this option is set, a running marking process is stopped, when the temperature is above the upper temperature limit or below of the lower temperature limit byt the given amount of "Temperature deviation". The process is continued only when temperature is back within the given range (plus/minus the "Temperature deviation").

In RepliSLS3D, the pen parameters are extended by an additional tab-pane "Heating" that offers some parameters that can be set to individual values for each pen. During marking these related values are sent to the plug-in whenever then pen changes:

- Lower Temperature – a temperature which is used to turn on an external device (like a heater), when measured temperature falls below of the limit specified here, the device is turned on
- Upper Temperature – a temperature which is used to turn off an external device when measured temperature rises above the limit specified here

This plug-in is located in flowplugins/libio_optris_lt.

3.5 Machine Configuration

There is a set of machine-dependent parameters which are not stored within each project file or in default project configuration, but locally on each machine where RepliSLS3D is installed at. These parameters can be configured via related dialogue in menu "Project" sub-menu "Machine settings..." which offers several configuration options within different tab panes. The following parameters can be set and modified within these panes:

In tab-pane "User interface":

- Expand / Move entity tree to tab panel – this option saves some space in X-direction; the entity tree that is shown on main windows right hand side normally will be moved to the left side and becomes a normal tabbed panel there; changes to this parameter do require restarting of the software in order to take effect

- Security / No warning message at start-up – in general it is not recommended to use this option, it makes sense in some very rare cases only: when this checkbox is set, the warning message about the possible dangers of this application is no longer shown. This means, prior to turning of this warning, other ways of informing the user have to be established in order to avoid unwanted and possible dangerous usage of this option.
  Changes to this parameter do not require saving of the current configuration as default one, the related information is stored automatically and always only locally, it does not belong to a project file which can be given away.

- No license progress bar on start-up – on application start RepliSLS3D checks for a valid license. Depending on the speed of the used computer this can take some seconds. During this time a progress bar is shown. When this option is set, this progress bar is no longer used but license checking is still done – so with this option the delay at application start up is still there but without a visual feedback.

- Switch back to old OpenGL 1.1 – the 3D view within the application makes use of modern OpenGL for fast drawing also of complex geometries and 3D models. On some graphics hardware and in case of poor quality of drivers of such graphics card this can cause issues resulting in wrong or corrupt displaying of models. In such cases it is recommended to turn on this option to switch back to the slower OpenGL version 1.1

- Hide support structure settings – when this option is set, the parameters to generate support-structures in 3D Slice Group tab pane is hidden from the user and no support structures can be generated any more

- Disable centre point – every element that contains some geometry comes with a centre point that is used for rotations of this geometry. This centre point can be edited, means it can be moved to any suitable position relative to the geometry itself. When this option is enabled, this centre point is no longer visible when an element is selected and it no longer can be repositioned.

- Language – by default this parameter is set to "Automatic", means the application chooses the language the underlying operating system is configured for. When other translations are available, they are listed in the related combobox and can be chosen to be used instead. Then on next start of the application, this language is used for the user interface.

- Menu item shortcuts – the area below of these parameters contains an editor that gives the possibility to edit the shortcuts of RepliSLS3Ds menu items. On lower left hand side the menu item to be edited can be selected in a tree view. Next right beside that tree the new shortcut has to be typed in field "New Shortcut" and then can be added using button "Add". The new shortcut is shown in the list right beside the menu tree view and can be removed from there using button "Remove". After the settings dialogue is left with "OK" these new menu shortcuts are applied to the menus.
  These settings are locally only and do not become part of a project.

The tab-pane "Toolbars" gives the possibility to configure which tools are shown in the toolbars and which are hidden in order to not to be accessible by the user. With the selection list at the top, the toolbar to be configured can be chosen. Next the list at the right shows the tools that are accessible. Selecting such an entry gives the possibility to move it to the left (the list of hidden and therefore not accessible tools) by pressing the button with the arrow to the left. In same way it is possible to chose a hidden tool and move it to the right list by pressing the button with the arrow to the right.

After changing the tools within the toolbars, the software needs to be restarted.

Please note: tools, which already have been disabled by the related customisation functions (as described in "Error: Reference source not foundError: Reference source not found"), are not visible here, neither in the "shown" nor in the "hidden" section. Furthermore, in case of custom installations, where specific tools have to be disabled on set-up of the software, it is recommended to not to use the toolbars-function out of the machine settings, but to use the customisation functions out of the section Error: Reference source not found too.

The "Smart Factory" tab-pane offers functionalities which all are related to integration into automated production environments:

- Machine name – here a name can be set which identifies a software installation/a machine uniquely. By default an auto-generated name is set here which can be replaced by a human-readable value. This name is used for all Smart Factory related functions, e.g. within all Smart Interface state messages to identify the machine it comes from and within the relevant Hermes messages that require an unique identification too.

- Enable Smart Interface – when this checkbox is set, the Smart Interface is enabled and the software can be integrated into a Smart Factory / Industry 4.0 environment. For details regarding this interface please refer below

- Enable MQTT connection – when this checkbox is set, state messages are sent to a suitable broker using the MQTT protocol and as configured in next field

- Broker – here IP and port number of the broker to connect with and to publish the MQTT state messages to has to be set

- Topic basename – using this field a topic for the MQTT messages can be configured. It will consist of a base-part of the topic which always is expanded by the machine name configured above. The base-part of this topic can be configured freely here but always has to start and to end with a slash ("/")

- Enable Hermes automatic control – when this checkbox is set, support for machine integration according to the Hermes-standard is enabled (for details please refer to https://www.the-hermes-standard.info). This enables several configuration parameters which can be used to specify connection to other machines within the same line. Here it depends on the availability of previous and/or next machine if RepliSLS3D acts as end of line machine, begin of line machine or something in between. If the next or previous machine does not connect according to the Hermes specification, RepliSLS3D automatically switches to end/begin of line mode.

- Previous machine – here IP and port number of the machine which transfers working pieces to RepliSLS3D can be configured

- Next machine – here the port number of the machine RepliSLS3D transfers working pieces to can be configured. Since the next machine has to connect with RepliSLS3D actively, here no IP has to be specified. Instead of this the local IP must be known at the next machine.

- Conveyor axis – using the following comboboxes it can be specified which motion controller and which of its axes has to be used as conveyor to feed the working pieces. Here also the movement direction of the conveyor can be set.

- TransportFinished distance – when the conveyor axis moves to transport a working piece to the next machine, a TransportFinished command has be sent to the next one, as soon as this transport is complete. Using this field the motion distance can be specified which has to be used to move a working piece out of the current machine. This parameter is used only when RepliSLS3D does not act as end-of-line machine.

- Enable Hermes remote configuration – according to the Hermes standard there is a possibility to send configuration data from a remote host and to overwrite local information as they may have been specified here. When this checkbox is set, overwriting of the local configuration is allowed and possible remotely. When it is not set, the Hermes-configuration of this machine can be changed only by direct manual and local interaction.

Since all parameters from this tab-pane are related to the local machine the software is running at, they are never stored within a .BEAMP project file but in a separate, local configuration file. These data are stored automatically when the configuration dialogue is left via "OK".

In tab-pane "Users" access to functions can be limited depending on users, their roles and privileges. By checking the box "Enable User Management" this functionality is enabled and some predefined settings are provided. For all existing, preconfigured users the password is similar to the user name and should be changed prior to operating it on the target system.
The users-tab-pane lists the available users on the left hand side. By selecting one of them, the related privileges and settings are shown on the right hand side of the pane. Once a user is selected, its privileges can

be changed, the login and full name as well as the state and the comment can be modified. To activate such changes after setting them, the button "Apply" has to be pressed, elsewhere the changes of a users parameters are not activated.

The check-boxes on upper right side of this tab-pane define what a user can do after logging into the software successfully:

- Supervision – this is a special privilege that applies only to user "Supervisor"; as long as this user is logged in, all functionalities are enabled, no restrictions exist

- Manager Users – a user with this privilege is allowed to modify the users and its privileges within this tab-pane; this should be combined with the "Modify Settings" privilege in order to give the related user access to the configuration dialogue

- Exit Application – this privilege gives a user the possibility to leave the application

- Modify Settings – a user with this privilege is able to enter the project configuration dialogue and to change the global parameters and settings of an installed software

- Edit Geometry – when this privilege is set for a user, this user will be able to modify loaded geometry and to edit the data of the currently loaded project

- Load Project – this gives a user the possibility to load an other project, this privilege is similar to "load recipe" like it may be named in other software

- Save Project – a user with this privilege is allowed to save the current project to disk; this privilege should be combined with an other privilege that gives the possibility to change the project settings, to edit the loaded project or to load/insert an other project, elsewhere it would not make sense for this user to save an unchanged project only

- Control Marking – when this privilege is set, a user is allowed to start and stop the marking/building process

Comparing to all the other configuration parameters the user privilege settings are not applied to the current project but to the local installation only. This means a project file that is transferred to an other installation of RepliSLS3D where no user management is enabled can be loaded and used without any limitations.

The user data are stored in the default application data directory of the used operating system within a sub-folder "beamhome". The exact position depends on the used operating system. The data are neither encrypted nor protected, so deletion of the directory and its contents would disable the user management completely. Thus it has to be ensured by using the operating systems functionalities that no user is able to access this folder without permission (e.g. by disallowing users to exit the application and by locking the whole system as long as RepliSLS3D is running). This is done by intention to offer the possibility to restore a system just by deleting the "beamhome" directory when some relevant passwords got lost.

In case the user management was activated, on next software start-up all functions are disabled until a valid user logs in by using the correct password. Logging into the software can be done by pressing the user-symbol in toolbar:

3.6   Pen Settings

Every element of a sliced group (like slice contour, slice hatch, support structure contour or support structure hatch) has a pen assigned to its geometry. This pen contains all information the related element will be marked with: scanner speeds, delays, laser power and frequency and others more. These parameters can be set within the pen settings dialogue that opens when the menu "Project" menu item "Pen settings..." is selected.

On top of that dialogue the head can be selected this pen applies to. This is important for multihead environments where more than one scanner controller/more than one independent working area is used. There will always exist the same number of pens for all heads but the same pen number can have different parameters for different heads.

Below of this a pen can be chosen for editing, the same combo box can be used to create a new pen, in this case the option "Append new pen..." has to be selected there. After selecting an existing or a new pen all its

parameters can be changed within the lower part of the dialogue. It is separated into some sections that contain different categories of pen parameters.

All tab-panes that hold these categories have one common functionality for selected parameters: by pressing the button "To all pens" a parameters value can be applied to all pens that currently exists. So in case a specific pen parameter has to be changed globally, with this function it is not necessary to edit it manually within every single pen.

PLEASE NOTE: there is no feedback from a possibly connected laser or scanner equipment to this dialogue, so the application can't know if a value is suitable for the connected equipment or not. Although there are some logical and range checks that try to avoid heavy mistakes it is up to the user to keep all parameters within valid and useful ranges.

3.6.1   General Pen Settings

This panel contains some basic pen parameters. On top of it an input field can be found that can be used to rename the current pen, there its name can be changed and the new name can be applied by pressing the return key.

Beside of that the following parameters can be edited here:

- Power – the output power in range 0..100% the laser has to be driven with

- Frequency – the frequency in unit Hz the laser has to be driven with; the allowed frequency range can be configured within project configuration dialogue

- Mark Speed – the speed of the scanner that is used during marking (motion while laser is turned on); the allowed speed range can be configured within project configuration dialogue

- Jump Speed – the scanners speed that is used during jumps (motion while laser is turned off); the allowed speed range can be configured within project configuration dialogue

- Spot Size – the diameter of the lasers spot on the surface, this value should be as exact as possible because it is used for different calculations

- Draw Off-Movements – this option is related to the visual representation of a project within RepliSLS3D only; when this option is set not only these parts of the geometry are shown within the drawing area where the laser is turned on but also the jumps where the laser is turned off; this option can be helpful to optimise a project and to re-arrange elements to perform as less jumps as possible

- Use as default Pen – the pen where this option is set for is used as default pen for all newly created geometries. More than this, whenever a new pen is created the parameters of this pen will be set to the values specified for the default pen. Within a project only one pen can be default, so whenever this option is selected for a pen, it is disabled, resetting this option can be done only by choosing an other pen as default one.

- Use for Pilot-Laser – the parameters of this pen are used to control the pilot-laser. This means the speed and delay settings are used whenever the pilot laser is started. Since this laser can be turned on and off only, all values like power, frequency and some delays are disabled, they are not relevant for the pilot laser. Within a project only one pen can be used for the pilot laser, so whenever this option is selected for a pen, it is disabled. Resetting this option can be done only by choosing an other pen for the pilot laser.

- Pen On-Colour – this colour is used within the drawing area for all geometries

- Pen Off-Colour – this colour is used when "Draw Off-Movements" are enabled to show the laser jumps within the drawing area

- Use for coloured scanner-bitmaps – when this checkbox is set the current pen is used for coloured scanner bitmaps; this option should be enabled only
1) when the current pen parameters cause a coloured result on current material
2) when the colour of this result is specified using button "Colour after marking"; PLEASE NOTE: setting the resulting colour requires a calibrated monitor in order to have exact results

3.6.1.1 Pen Parameter Wizard

It is sometimes difficult to find parameter-combinations that fit to a special material or application. Thus the pen dialogue offers a special wizard that creates a matrix out of different values where the user can choose a suitable result from. To open this wizard, the button "Parameter Wizard" on upper left corner of the pen settings windows has to be pressed.

There a window opens where a 4x4 to 10x10 matrix of small marking areas is symbolised by a matrix of buttons. On upper part of the window the parameters to be used have to be selected and their range to be checked can be entered. These values are spread over the matrix in order to find best parameters. Next the yellow "Mark Start" button right beside the button matrix has to be pressed in order to start output to the material. Alternatively it is possible to use the button "As project" in order to not to mark out of the pen parameter wizard directly but to use the generated pen parameter matrix as current project which then is editable like a simple 2D marking project.

ATTENTION: by pressing the mark-button a laser operation is started! This may operate with up to 100% power and therefore is potentially dangerous! Prior to opening the pen parameter wizard and prior to starting output out of this wizard all security measures have to be ensured!

As soon as marking has finished, the area with the best marking result has to be chosen and the related button in pen parameter wizard has to be pressed.

To make it easier to find the correct position there is one circle and one triangle used for marking, these symbols identify orientation and direction of the matrix uniquely. So to find the position of the best result it is possible to first count from triangle to circle until the correct column is found and then count the row number. When the same is done in pen parameters button matrix the exact position is found and this button can be pressed. This button-press opens a popup-menu where one of the following options can be chosen for further processing:

  – Refine this result – some values close to the chosen position are used to create a new matrix for a second, more accurate calibration step

  – Go back from this result – this option performs the opposite operation of the previous one, it creates a new parameter matrix with a larger range of values

  – Use parameters for pen – when this item is chosen the parameter wizard will be closed and the found parameters that belong to the selected field are set to the current pen.

For the other marking parameters apart the selected ones the pen parameter wizard always used values of the currently selected pen, only the selected ones are varied and modified by this function.

3.6.2   Scanner Pen Settings

This tab-pane contains all scanner-related pen parameters:

  – Jump Delay – time the scanner waits when a jump was executed; this delay applies at the end of a jump to let the scanner finish its movement
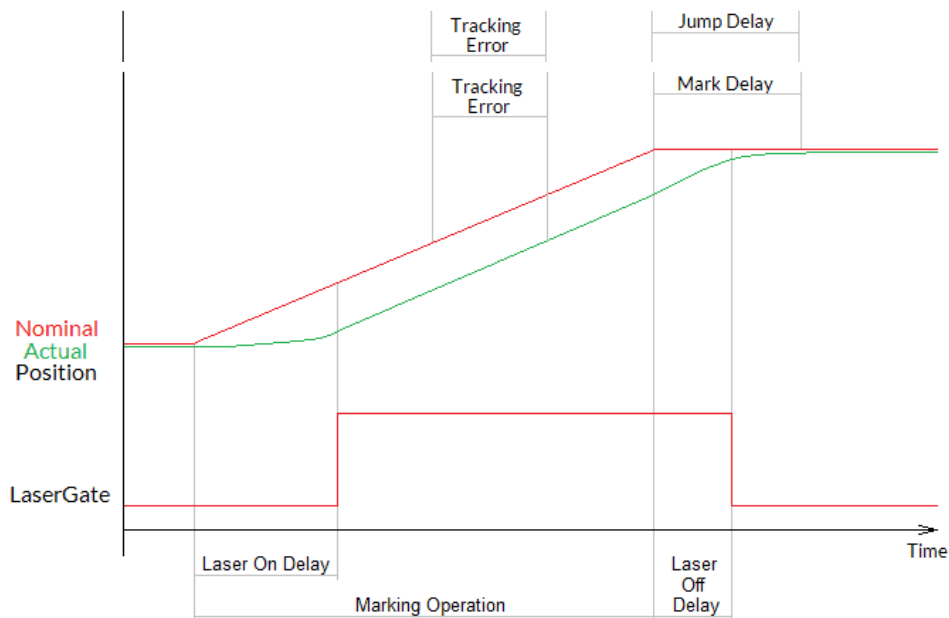
– Mark Delay – time the scanner waits when a marking operation was executed; this delay applies at the end of a marked line to let the scanner finish its movement

– Polygon Delay – the time the scanner has to wait for every corner within a polygon when the laser is turned on and stays turned on

– Use variable polygon delay – this option belongs to parameter Polygon Delay. When it is set, there is no static delay used in every corner of a polygon, but a variable one which depends on the current angle between two lines of the polygon. The sharper the angle is, the longer the used polygon delay becomes. This option is available only when a scanner controller card is configured which supports this feature.

– Auto-generate delays – the delay values described above are essential to get high quality marking results also in case of an as high as possible marking speed. Since the evaluation of the correct values is a difficult and time-consuming process, RepliSLS3D offers a possibility to set some default values which can be used as starting point for optimising these delays.
After pressing this button, a dialogue opens, where the used scan head can be selected out of a list of predefined types. Then some base delay values are generated for this scan head.
In case a used scanhead is not listed there, option "Other" has to be chosen, which requires to enter one single value for delay calculation: Every scan head vendor should provide a value named "time lag", "tracking error" (or something similar) within its technical documentation. This value describes the specific inertia of this scan head and needs to be entered here.
When this dialogue is left by pressing "OK", RepliSLS3D calculates some basic delay parameters (including not only the jump- and mark delay but also laser-on and -off delay as per section "3.6.3 Laser Pen Parameters" below)that fit to the scan heads mechanical properties and often can be used directly. Alternatively these delays can be used for further optimisations for these applications, where a more strong and exact set-up of the scan head is required.
For more details about the purpose of all these delays and their influence please refer to your scan heads or scanner controller manufacturers manual.

– Wobble mode – when wobble is enabled the linear movement of the scanner during marking is distorted by an additional frequency that lets the scanner oscillate around the predefined position. This functionality can be used e.g. to warm up larger areas of a material, for welding or similar things. To enable wobble, a mode has to be selected and both, a wobble frequency and amplitude have to be set.
Following wobble-modes are supported:
Straight: wobbling is done with fixed values for frequency and amplitude resulting in a regular wobble shape, in this mode the range-parameters as well as the update frequency parameter are ignored
Fuzzy: the wobble parameters are randomly modified which results in a fuzzy-looking wobble-pattern; for this mode at least one of the range-parameters as well as the update-frequency value have to be set

62

Sweep: the wobble parameters are changed linearly varying from one extreme (as configured by the "+-" parameters) to the other

- Wobble Frequency / Wobble Amplitude – when both of these values are set to 0, the wobble mode is turned off, independent from what mode is configured. To do some wobbling, both parameters amplitude and frequency have to be greater than 0. The wobble frequency specifies how fast to oscillate and the wobble amplitude how much to vary over the original position. Wobble has to be supported by the used scanner controller card in order to have any effect during marking.

- +- Wobble Frequency / +- Wobble Amplitude – these parameters apply to the wobble frequency and amplitude when a non-straight wobble mode is configured. They specify a range, which defines, how strong the related wobble parameters have to be changed. When both are set to 0, this is similar to wobble-mode "straight".

- Wobble update – when a non-straight wobble mode is set, this parameter specifies, how fast the wobble-parameters have to be changed

The given delay values have a direct influence on the marking result and need to be adjusted carefully. Wrong delay values can result in inaccurate geometries, burn-in points at some specific positions and/or unnecessary long marking times. Out of the visible result it can be evaluated which delays are to small or too big by checking following rules (please do also refer to the images below which make use of some characters as example for possible marking results since there the effect is more obvious):

Parameter	Value too big	Value too small	Remark
Laser On Delay	First part of a mark vector (following a jump vector) is not marked	Burn-in point at the beginning of a mark vector that follows a jump vector	See pictures below
Laser Off Delay	Burn-in point at the end of a mark vector and before the next jump vector	Last part of a mark vector (before a jump vector) is not marked	See pictures below
Jump Delay	Mark time is longer than necessary	Geometry is damaged at the end of jump (transient oscillation)	
Mark Delay	Mark time is longer than necessary	The end of a vector is turned towards the direction of the following jump vector	See picture below
In-Polygon Delay	Burn-in points at the corners	The corners of the polygon are not sharp but rounded and at wrong position	
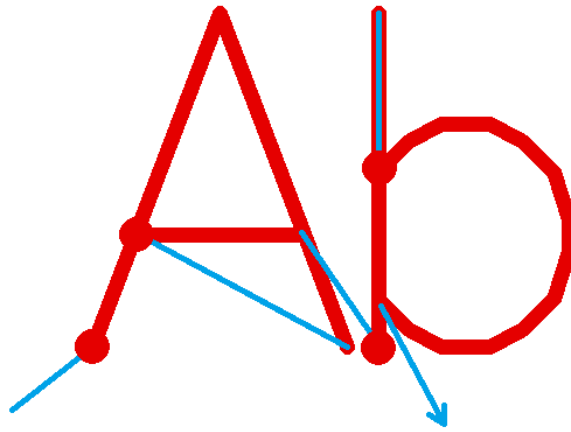
Laser-on delay is too short, laser is switched on too early while mirrors are still slow or accelerating, thus it fires on the same position for a too long time which results in burn-in points; please refer to section "3.6.3 Laser Pen Parameters" below for this parameter
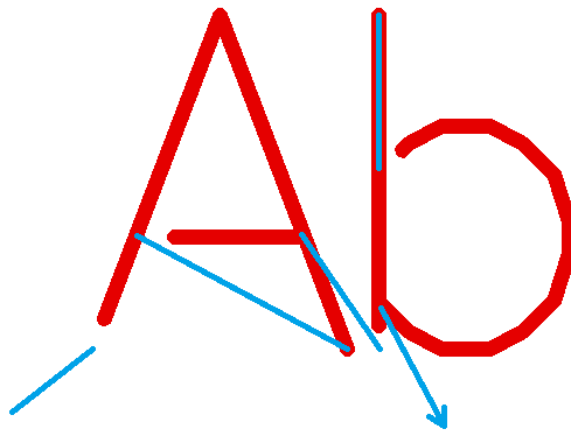
Laser-on delay is too long, laser is turned on too late while scanhead already started marking of the next vector, thus parts of the mark geometry are missing at the beginning; please refer to section "3.6.3 Laser Pen Parameters" below for this parameter
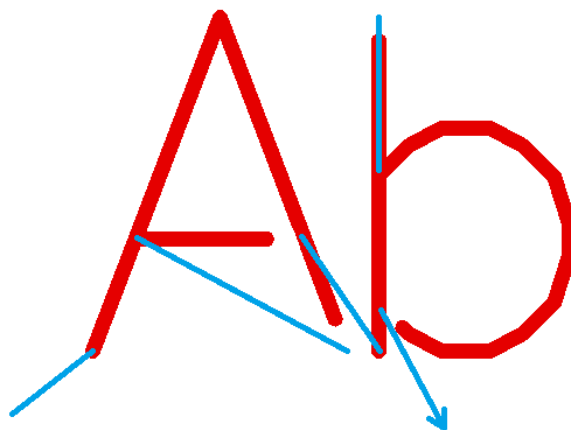
Laser-off delay is too short, laser is turned off before the end off the current mark vector is reached, thus parts of the mark geometry are missing at the end; please refer to section "3.6.3 Laser Pen Parameters" below for this parameter
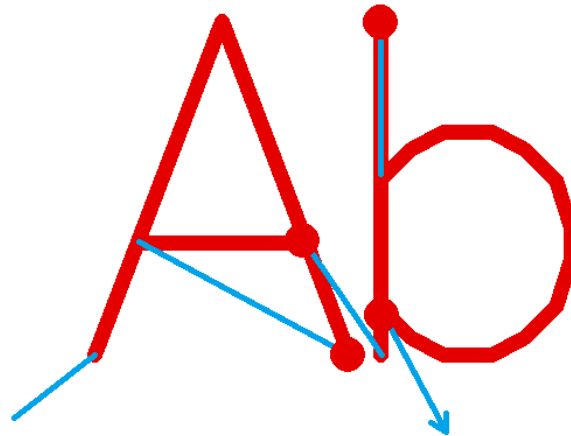
Laser-off delay is too long, laser is turned off too late while end point of the vector is already reached and mirrors of the scanhead slow down, this results in burn-in points at the end of the mark vectors; please refer to section "3.6.3 Laser Pen Parameters" below for this parameter
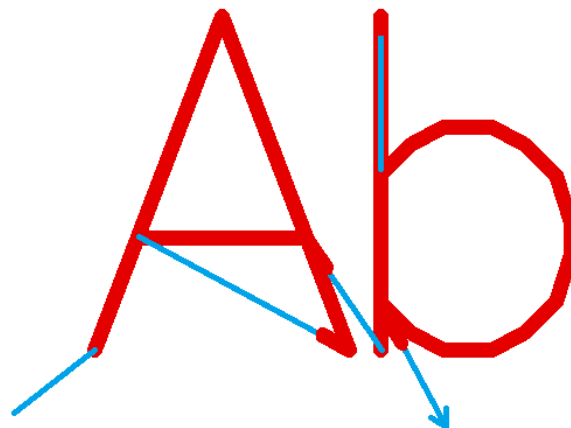
Mark delay too short, motion of next jump vector is started too early and before end position of previous mark vector was reached, thus the laser is still on when motion for the following jump vector is performed
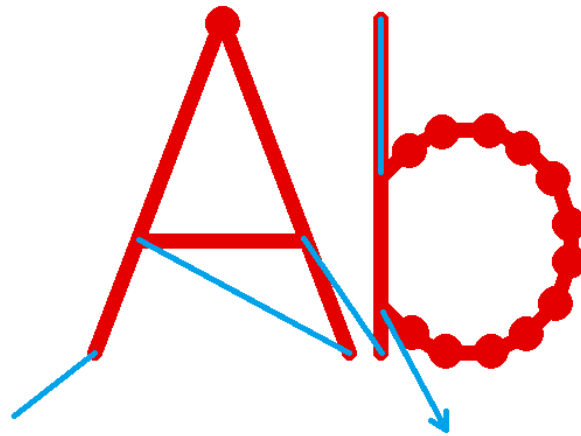
In-polygon delay too long, scanner stops for a too long time at the position of a corner point within a polygon, thus burn-in points appear at these positions.
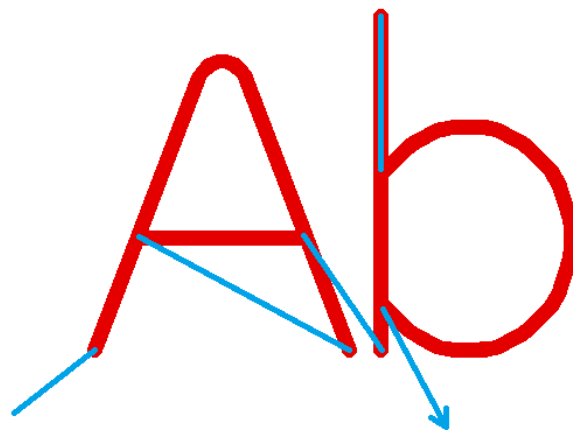
In-polygon delay too short, mirrors are not able to arrive at the positions of the corner points within a polygon before the next mark vector starts, thus the whole geometry is rounded, the vectors are pulled into direction of the next one

3.6.2.1   Sky-Writing

As described in previous section, it is necessary to apply delays in order to have clear and sharp corners within polygons. Depending on the used hardware and its capabilities these delays can become quite long, slowing down the total marking speed. To avoid this, the "sky-writing" mode is available (on selected scanner controller hardware which supports this feature or has required capabilities). When this mode is active, the configured polygon delays are no longer used which waste some marking time. On the other hand the vectors to be marked become longer which itself again needs some time. So it depends on the used hardware and its specific capabilities if sky-writing really saves time during marking.

Sky-writing avoids delays and rounded, blurred corners by starting vectors earlier, ending them later and turn on/off the laser at exactly the position where the polygons line begins/ends:
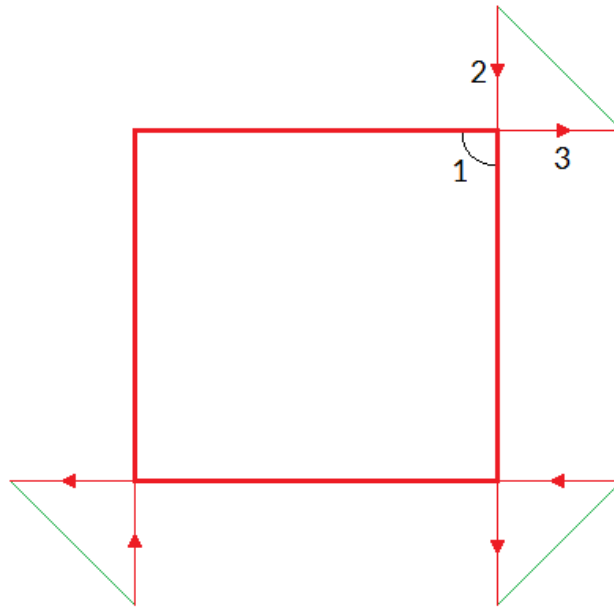
In this picture the thick red lines symbolise the marks where laser is turned on. The thin red lines show the movements where vector lines are extended by a given length but where laser is already turned off. For a better understanding small arrows in the jump lines show the marking direction. The thin green lines show the jumps, only at these positions the chosen jump speed is used, all other movements are done with mark speed.

Here each line of the rectangle starts earlier, turns on the laser at the point where the polygon line has to begin, turns it off at the point where it has to end and then continues a bit in the same direction. Next a jump is done to the position where the next extended vector line has to start. Now when the scanhead would distort vectors because there is no more in-polygon delay active, this distortion would happen outside the rectangle and at a position where the laser is turned off. Thus the distortion would no longer be visible in the marking result when the lengths given for the polygon are long enough:
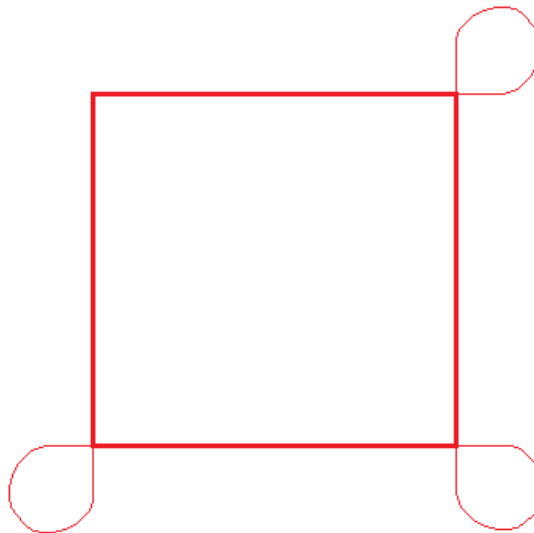
In this image the real scanner movements are shown. Here the movements at the corners are rounded and distorted due to the high speed, but these distortions do not influence the marking result since they happen when the laser is already turned off. In this specific example the lengths the lines are extended by could be shortened because there is still a longer straight line before and after every line.

Sky-writing requires following parameters to be configured:

- activation threshold – marked with a "1" in picture above: this is a value that specifies at which angle between two lines within a polygon the sky-writing mode needs to be activated, if an angle is equal or smaller (means sharper) than this value, sky-writing is used to keep the marking result clear, for bigger (means flatter) angles the normal in-polygon delay (as described above) is used

- fade-in distance – marked with a "2" in picture above: this value specifies the length a line has to be extended with at the beginning, means this is the size which is added before laser is turned on. This value should be as small as possible in order to not to waste marking time. At the borders of the current working area it may be possible this value can't be applied fully because the starting point would be located outside of the valid range of the scanhead.

- fade-out distance – marked with a "3" in picture above: this value specifies the length a line has to be extended with at the end, means this is the size which is added after the laser was turned off. This value should be as small as possible in order to not to waste marking time. At the borders of the current working area it may be possible this value can't be applied fully because the end point would be located outside of the valid range of the scanhead.

3.6.2.2 Overspeed

An other method of getting a higher speed out of a scanhead is the overspeed-functionality. With this feature the galvos of a scanhead are driven with a speed that is far beyond of its limits and normally results in a totally distorted marking result. Here the scanner controller card takes care about the movement of the galvo and drives it with a specific speed curve that avoids these distortions. When this feature is used, the exact improvement highly depends on the used marking data. The more long vectors it contains and the less inertly a scanhead is, the higher is the possibility to find vectors that can be driven with overspeed and the better is the possible improvement in speed.

To make use of the overspeed-function, some specific requirements needs to be met:

- the used scanner controller card supports these speed curves for overspeed movements

- scanhead bases on galvo-technology

- scanhead does not artificially limit the maximum speed

The overspeed-function can be enabled and configured in pen settings, tab-pane "Scanner". Here following parameters can be set:

- Enable overspeed – enables the overspeed-function for the related pen; when this checkbox is set, the following parameters need to be configured:

- Tracking error – this is a scanner-specific value (also named "time lag" sometimes) which describes the inertia of the mechanical parts of the used scanhead. This value is fixed for each type of scanhead and is typically provided by the vendor of the scanhead.

- Jump speed factor – this value is the overspeed factor which is used for jumps. Normally jumps are done using the speed defined in tab-pane "General". When the overspeed-feature is enabled and when a jump vector is suitable to be driven faster, it no longer is moved with the nominal jump speed but with the nominal jump speed multiplied by the factor given here.
The maximum value this factor can have depends on the used scanhead but typically is in range 3..9

To adjust a scanhead for usage with the overspeed-function and to find suitable parameters, please perform the following steps:

1. Create a test-project that represents a typical application but also contains some large jumps (at least halve of the working area is recommended). These large jumps later are used to find the maximum overspeed factor

2. Start with the overspeed-function disabled.

3. Use the test project from step 1 to find the proper scanner and laser-delays as well as the nominal maximum jump and mark speed the scanner can be driven with while it still provides acceptable marking results with no distortions (this is the normal procedure of setting scanner parameters, so please refer to previous sections about adjustment of these delays).

4. Enable the overspeed-function for the pen(s) used in the test project and set the tracking error value according to the technical data of the used scanhead.

5. Increase the jump speed factor until you find a value where the marking results start to become distorted, here especially the marking results which are located close to the long jump vector(s) are important and have to be checked

3.6.3 Laser Pen Parameters

This panel contains several parameters that belong to the used laser and the parameters/signals that are required to operate it. So depending on the laser type that was configured for the current project these options are enabled that contain useful parameters for proper laser operation. For a detailed description of these parameters and their meaning please refer to your laser vendors manual.

When an output device is used, which does not make use of a laser but of an other kind of tool, all laser-specific parameters are disabled and don't need to be configured.

PLEASE NOTE: some laser types need to be operated using relatively tight parameter ranges. When they are controlled with wrong parameters the results may be an undefined behaviour or the laser hardware may be damaged in worst case. So when setting up new parameters or when changing them or after a software update was done, it is highly recommended to (re-)check these laser parameters and the output signals of the scanner controller in order to avoid problems with the laser. Here "software update" not only includes updates of the RepliSLS3D software package. Also in case the software or drivers that for an example belong to the scanner controller card are modified, it is recommended to perform these checks.

This tab-pane contains all laser-related pen parameters:

– Laser On Delay – the time that is set between start of a mark movement and the moment the laser is switched on; so this parameter specifies if and if yes how much the laser should be turned on delayed in case a marking operation is started

– Laser Off Delay – the time that is set between the end of a mark operation and the moment the laser is turned off; so this parameter specifies how much turning the laser off has to be delayed in case a marking operation ends
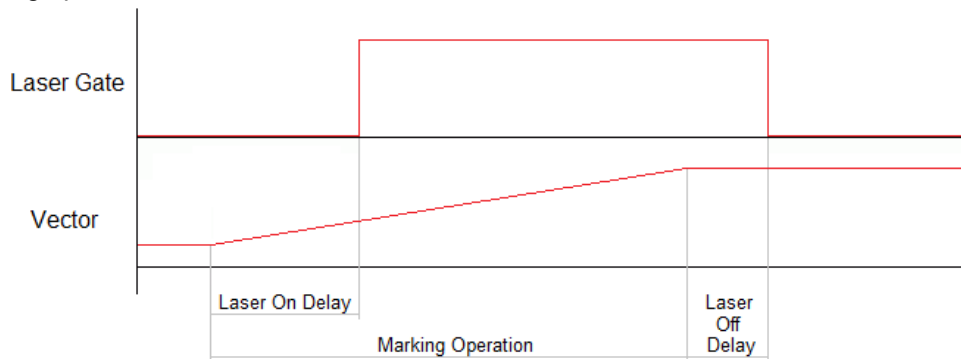
– Dot-length – this is a convenience function which shows and calculates the number of pulses a dot consists of with the current settings. The value given here depends on the frequency and the current laser on and laser off delays.
So when the value in this field is changed, the corresponding laser off delay is changed automatically. On the other hand, when frequency, laser on delay or laser off delay are changed, the number of pulses shown in this field are changed according to the modified values.

3.7 Process 3D Data

RepliSLS3D supports two methods of building 3D models: 2.5D, where the Z-axis is "simulated" by a stage changing the Z-position of the working area and current layer to be processed and real 3D where a 3D-capable scanhead is used performing also the Z-movement.

3.7.1 Slicing 3D Data

As a first step a ready-to use 3D model has to be imported. Currently RepliSLS3D supports the supports the 3D Manufacturing Format 3MF format, the 3D Studio 3DS format, the Object File Format OFF, the Render-Ware RWX format, the Surface Tessellation Language STL in binary and ASCII format, the WaveFront OBJ format and the PLY 3D printer format and others more. Beside of that the Common Layer Interface CLI format is supported too, since this is a file format that already contains sliced data it can be used for processing but is not a real 3D format where a slicing operation can be applied to.

More directly supported 3D file formats will follow in future. In case 3D data are available using a different file format than the ones described above, a converter application can be used. Here we recommend to use AccuTrans 3D which is available at http://www.micromouse.ca/ and supports nearly every 3D file format that exists. AccuTrans 3D itself is available for an extremely low price.

After selecting a 3D file for import, a configuration dialogue opens where some initial slicing parameters can be configured. All the slicing parameters can be changed later within the application:

- Import Scale Factor – a factor that can be used to scale the whole 3D object during loading

- Slice Distance – the distance between every slice

- Slice Offset – specifies an offset from top/bottom of the 3D model where slicing has to start; in case this value is set to 0.0 it may result in a first slice that contains nearly no relevant data (dependent on the used 3D model)

- Reduction – expects a value in unit degrees to perform a lossy data reduction on sliced data; whenever an angle between two vectors of a layer is smaller than the angle given here, they will be concatenated to one common vector. For small angle-values and in most cases this results in a memory consumption which is dramatically lower and processing speed therefore much higher while there is no visible loss in quality of the result.

- Close lines – this parameter can be used to try to repair malformed or defect 3D models. When two not connected lines are located closer to each other than the distance specified here, the software tries to connect them. So this function tries to close holes in the imported 3D model. Please note: when the value is set to too big values, it may happen that the wrong open ends of slice lines are connected to each other – resulting in a damaged slice.

- Slice Direction – here the slice direction for the 3D model can be selected, "Top To Bottom" may be used e.g. for depth-engraving where operation starts on top of the model and moves into the material, "Bottom To Top" can be used e.g. for rapid prototyping applications where processing starts from bottom and builds up a model slice by slice

- Real 3D / Make use of Z-coordinate – when the scanner controller card is configured for 2D only (so no third axis is enabled), slicing of a 3D model is done in a way where the Z-axis always stays at 0. In this mode for Z-movements some additional actions are necessary such as adding a motion operation between each slice. In this mode, due to the lack of the third dimension, the "Real 3D" option is disabled and can't be used.
  When the scanner controller operates in 3D mode (e.g. with a real 3D scanhead connected), the Z-coordinate can be calculated by the slicing function so that the model can grow in Z-direction controlled by the scanhead. To do that, this option has to be set. When operating in 3D mode but this "Real 3D" option is not enabled, the Z-coordinate is no longer used for building the 3D model and again stays at 0 and again requires a separate Z-shifting. This is useful e.g. in cases where the 3D scanhead is used for keeping the focus but not for doing real Z-movements.

- Split to tiles – the working area is split into (invisible) tiles that are used to rearrange the contours of the sliced 3D mesh during marking. Using these tiles it can be avoided to heat up the material at a specific position too much, it lets the position where contour is marked make additional interruptions and jumps to other place in order to continue marking at a distant position (means a distant tile). When this option is set, the following two parameters can be used to specify behaviour of this feature. For a more detailed description and usage examples please refer below.

- Tile Size – the size of a tile in working area, this value should be such a fraction of the working area size, that lets all tiles fit fully into whole working area

- Tile Skip Size – skip distance between tiles, this value decides how many tiles in horizontal direction should be skipped before marking continues in next tile; remaining tiles are processed on next cycle; this value should be much smaller than the total number of tiles that fit into the working area

- Clean between slices – when this option is set, a 3D Slice Cleaner element is generated on construction of the Sliced 3D Mesh Group. For more information about this type of element which can be used for cleaning purposes between slices, please refer to section "3.7.5.2 3D Slice Cleaner"

When this configuration dialogue is left by pressing the "OK" button the 3D model is loaded and sliced according to the specified parameters. Dependent on the size of the model, the number of slices that result out of this size, the slice distance and the complexity of the models geometry this operation may take a while. When the slicing operation is finished a "Sliced 3D Mesh" group element is added to the current project that contains the 3D model and the slices.

3.7.2   Slicing the imported model – Sliced 3D Mesh

After importing a 3D model as described above, the entity-tree on main windows right hand side shows a hierarchic structure with the imported 3D-mesh as child of a top-level slicing group element. This is a "Sliced 3D Mesh Group" which contains all data and parameter required for slicing and building the 3D model as well as all steps which have to be performed between slices.

When this kind of grouping element is selected a special property panel opens on the windows left hand side. There several parameters can be changed. Some of these parameter changes lead to a recalculation of the slices which means the currently existing slices, their sub-elements and possibly added motion-elements are removed from this group. So it is recommended first to optimise all the slicing parameters and then add more data and elements to such a group.

The Sliced 3D Mesh Group property panel itself is split into two major parts. Section "Model" contains all parameters that belong to the 3D mesh and the related data. Section "Support" is related to (automatic) creation of additional structures that are intended to help creating the model itself, especially at positions where parts of the model hang over and would not be stable during the phase of marking of single slices.

In section "Model" following parameters can be changed:

- Slice Distance – the distance between every slice; when this value is changed all slices and possibly existing additional elements are removed and the slices are generated again

- Slice Offset – specifies an offset from top/bottom of the 3D model where slicing has to start; in case this value is set to 0.0 it may result in a first slice that contains nearly no relevant data; when this value is changed all slices and possibly existing additional elements are removed and the slices are generated again

- Reduction – expects a value in unit degrees to perform a lossy data reduction on sliced data; whenever an angle between two vectors of a layer is smaller than the angle given here, they will be concatenated to one common vector. For small angle-values and in most cases this results in a memory consumption which is dramatically lower and processing speed therefore much higher while there is no visible loss in quality of the result.

- Close lines – this parameter can be used to try to repair malformed or defect 3D models. When two not connected lines are located closer to each other than the distance specified here, the software tries to connect them. So this function tries to close holes in the imported 3D model. Please note: when the value is set to too big values, it may happen that the wrong open ends of slice lines are connected to each other – resulting in a damaged slice.

- Slice Direction – here the slice direction for the 3D model can be selected, "Top To Bottom" may be used e.g. for depth-engraving where operation starts on top of the model and moves into the material, "Bottom To Top" can be used e.g. for rapid prototyping applications where processing starts from bottom and builds up a model slice by slice; when this value is changed all slices and possibly existing additional elements are removed and the slices are generated again

- Real 3D / Make use of Z-coordinate – when the scanner controller card is configured for 2D only (so no third axis is enabled), slicing of a 3D model is done in a way where the Z-axis always stays at 0. In this mode for Z-movements some additional actions are necessary such as adding a motion operation between each slice. In this mode, due to the lack of the third dimension, the "Real 3D" option is disabled and can't be used.
  When the scanner controller operates in 3D mode (e.g. with a real 3D scanhead connected), the Z-coordinate can be calculated by the slicing function so that the model can grow in Z-direction controlled by the scanhead. To do that, this option has to be set. When operating in 3D mode but this

"Real 3D" option is not enabled, the Z-coordinate is no longer used for building the 3D model and again stays at 0 and again requires a separate Z-shifting. This is useful e.g. in cases where the 3D scanhead is used for keeping the focus but not for doing real Z-movements.

– Slice Pens – this table can be used to define patterns for the pens that have to be assigned to every slice. There separate pens can be specified for contour and hatch of a slice. First of all the check box in front of a row of the list has to be selected to enable this entry. Then the pens can be selected via the combo box. Here the list rows have different meanings:
- "First Slice" is used for the very first slice only (that can be on top or bottom of the model dependent on the slice direction)

- "Last Slice" is used for the very last slice only (that can be on top or bottom of the model dependent on the slice direction)

- 1..5 (all other rows) within that list define an order of pens that is repeated until the last slice is reached. Here patterns of up to 5 different pen combinations can be defined. Each pen definition (=row of settings) comes with an additional input field "Repeats" which defines how often the current contour/hatch pen combination has to be repeated before the next row is processed. Using this function it is possible to have more than one slice with the same pen definition before switching to the next variant.

- DIn – when this option is set to a value different than "Off", an override-function is enabled. Here "Din" defines a digital input on the used scanner controller card. As soon as this input is set to "HIGH" during marking, none of the pen-definitions "First", "Last" or "1..5" are used for marking, but the pens for contour and hatching that are defined within this row. When the input is "LOW" during marking or when the digital input function is set to "Off", the regular pen definitions are used. When a scanner controller card comes with different digital input ports, the digital input bit is read fro mthe first port according to this order: Digital 1 (8 Bit), Digital 2 (8 Bit), Digital 1 (16 Bit), Digital 2 (16 Bit), Digital (32 Bit)
Since the pen-definitions for sliced geometry have to be done here, the related option within the "Element" tab-pane is disabled, pens can be assigned only via this pen mapping table.

– Show All Slices – in case this check box is selected all slices that belong to a 3D model are shown within the drawing area of RepliSLS3D. This function has to be handled with care since a large amount of slices can slow down the application dramatically, here it is recommended to use "Display reduced" parameter to show only every nth number of slice

– Hide hatch from view – A hatched model can – dependent on the number of hatch lines calculated for a slice – slow down graphic visualisation quite dramatically. For such cases it is possible to enable this option which then disables drawing of the hatch patterns of the slices of a 3D slice group in order to have a faster and more smooth graphical representation of the current project

– Slice during building – normally complete slicing of a model is done in advance and for the complete model. Depending on the complexity of a model such an operation can consume quite a lot of time while no other operations are possible. When this option is checked, only the very first slice is precalculated and all other ones are done during marking: while a slice is marked and the related interslice (motion)operations are done, the next slice is calculated and the related hatch-patterns are applied in parallel. Thus a lot of handling time can be saved since both operations – marking of a slice and completely preparing the next slice – are done fully parallel.
This function can be used only with models where no support-structures are generated, means activating this option automatically disables generation of support-structures.
After setting this option, slicing and adding additional elements to the generated slices has to be done exactly in the same way as before. But with this option set, only the very first slice is shown. All others are generated during marking and therefore are not available in advance.

– Display Reduced – reduces the number of slices that are shown in case option "Show All Slices" is selected, the bigger this number becomes the less slices are shown within the drawing area

– Split To Tiles – when this option is set, the working area is split into (invisible) tiles that are used to rearrange the contours of the sliced 3D mesh. Using these tiles it can be avoided to heat up the material at a specific position too much, it lets the position where contour is marked make additional jumps in order to continue marking at a distant position (means a distant tile). When this option is set, the following two parameters can be used to specify exact behaviour of this function. After en-

abling this option the 3D mesh is re-sliced but there will be no visible representation of the changes in working area, it can be noticed during marking only. This function influences only the contour of the slices, hatches are not interrupted and broken into tiles in order to avoid damaged marking results. To reduce heating of material during hatch processing, please refer to description of the hatcher below; there exists a similar functionality to skip hatch lines. Together with the following parameters this function can be used to e.g. mark every second tile:
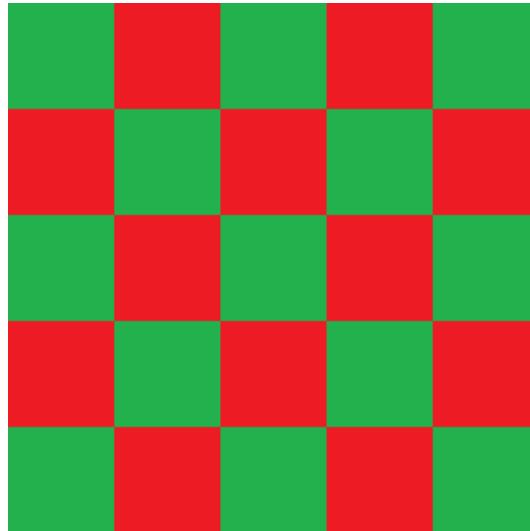
Assumed the working area has a size of 100 mm x 100 mm, for this behaviour "Tile Size" needs to be set to 20 mm and "Tile Skip Size" has to be set to 2. With these settings first all these contours are marked that have vectors within red tiles, finally all the remaining contours that have a vector in green tiles are marked. Tiles itself are processed from upper left to lower right corner row by row. It is also possible to have even bigger distances between tiles so that they do not touch at their edges:
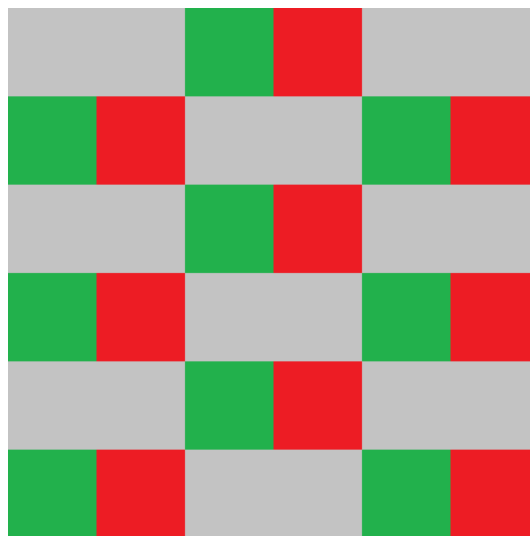
Parameters for a 90 mm x 90 mm working area are 15 mm for "Tile Size" and 4 for "Tile Skip Size". This results in a behaviour where marking starts with these contours that own vectors in red tiles, next the contours in green tiles are marked. This is continued until the whole working area was processed (in this example by processing the remaining two tile sets in grey areas with two additional steps).

– Tile Size – the size of a tile in working area, this value should be such a fraction of the working area size that lets the tiles fit fully into whole working area

- Tile Skip Size – skip distance between tiles, this value decides how many tiles in horizontal direction should be skipped before marking continues in next tile; remaining tiles are processed on next cycle; this value should be much smaller than the total number of tiles that fit into the working area

Section "Support" offers the following options and parameters which are available only when a model is created from bottom to top (so this option is not available when doing deep-engraving with a Sliced 3D Mesh Group):

- Enable Support Structures – this checkbox has to be enabled in order to let the software automatically create support structures, when it is turned off, the model will be processed as is and with no additional geometries. When enabled, additional geometries are calculated and added to the model automatically at positions where an overhang could cause problems. These additional geometries are only contours and need to be extended by hatch-elements in order to get them filled and stable. Here for both, contour and hatch suitable pens and related pen-parameters have to be chosen in order to not to melt together the support structures and the model's structures.

- Distance to Model – here a distance can be specified that always has to be used at positions where the contour of model and support structure are neighbouring within a slice; using this distance it can be ensured the support structure can be removed easily after processing. When this value is too large, the effect of supporting the model may fizzle out.

- Enlarge Structure – the value given here can be used to make the support structures larger than necessary in order to overlap the model completely; this value has no influence on the distance to the model, it expands the support structure to its outer borders.

- Style of support structures – the type of the support structure to be generated can be selected with this combobox, here following options are available:
<u>Enclose model completely</u> – when this option is set, the support structure is created always, will cover the whole model from top to bottom and will exist anywhere independent from the fact if there would be some gravity effects making it necessary to have support structures
<u>All-over structures</u> – the support structure is created from bottom to top until it is covered and replaced by parts of the model itself, working like a shadow that would be cast by the model itself. So in this mode support structures are generated at these parts of the model only, where negative effects caused by gravity are to be expected. Here no support structures are generated when there are (stable) structures of the model itself available. The generated support-structures are all-over, solid areas.

- Down to bottom of editing area – when this option is set, support structures are not only generated until bottom end of the 3D model but down to the lower end of the whole editing area (which in this case should be located below of the lower end of the 3D model)

- Support Pens – this table can be used to define patterns for the pens that have to be assigned to every slice's support structure. There separate pens can be specified for contour and hatch. First of all the check box in front of a row of the list has to be selected to enable this entry. Then the pens to be used for contour and hatch of a support structure can be selected via the combo box. All rows within that list define an order of pens that is repeated until the last slice is reached. Here patterns of up to 5 different pen combinations can be defined. Since the pen-definitions for sliced geometry have to be done here, the related option within the "Element" tab-pane is disabled, pens can be assigned only via this pen mapping table. Alternatively the pen-definitions for hatches can be disabled. To do that, the checkbox above the "Hatch"-column has to be unset. Now an own pen can be assigned to every hatch via the common way (pen-assignment in "Element" tab-pane). The difference between both methods: while pen assignment via the table in "Support"-section assigns the same pen to all hatches of a slice (which additionally may alter over slices), the pen-assignment via "Element" tab-pane can be used to assign different pens to each individual hatch that stay the same over slices.

Beside of that there is a slider available on the property panels left hand side: there the currently shown and used slice can be selected and it can be used to scroll through the sliced model. The same is true for the input field below of that slider: there a number of the current slice can be entered or chosen.

Such a Sliced 3D Mesh Group is shown as a group element with one sub-element for every slice within the element tree on the windows right hand side. Comparing to other groups it behaves a bit different. So it is possible to put new elements (like a motion control element or I/O elements) into that group directly. To do that, either the group node itself or one of its slices has to be selected and the Primary Element to be added

has to be chosen within the tool bar. Now a dialogue is opened where the user can select if that element has to be added as a single one only, below of every slice or for a range of slices. This way it is possible to set motion controlling elements for every slice easily. More than this: all elements that are added via such a range will belong together afterwards: whenever a parameter of one of the elements is changed, the parameters of all others that belong to the same range (and therefore have the same number in its name) are modified too.

The same is true for Additional Elements: when there is a hatch added to a single slice the user can choose to apply it to a range of slices or to all slices. When the parameter of one of these hatches is changed the change is applied to all the other hatches that have been created together with this first one.

Using this way the same operations and parameters can be applied to all slices easily.

Between every slice-subelement of the group there is also a "Process"-element created automatically. This element does expect only one parameter: the path to an existing project file. This project file is executed at the position of every "Process" element and can contain processing-operations that are necessary between every slice (such as motion operations to move a Z-table, setting of digital outputs and waiting for digital inputs a to synchronise with external devices). Using such an external project it is not necessary to add these control-steps – which normally are always the same for a machine – manually for every newly sliced element.

3.7.3   Filling a 3D Model

After slicing a 3d mesh only the contours are marked. To make it solid, a fill pattern has to be applied using the so-called "Hatcher". To do this, the slice or the support element of the sliced 3D model has to be selected in entity-tree and the purple toolbar-icon has to be pressed in order to apply some hatch patterns to it. Here several different hatches can be applied to slices and their support structures separately.

Hatching will work only with closed polygons and might fail on specific geometries that do not describe a closed area non-ambiguous. In such cases the resulting hatch pattern may be incomplete or may be located outside of the polygon. This can happen e.g. in cases where a damaged 3D model was imported.

The hatcher offers following parameters:

– Line Style – the style the lines the text consists of is created with, here following options exist:
Continuous: the geometry consists of closed, connected lines with as less as possible points
Connected Lines: the geometry consists of several short lines including additional points, all these lines are connected
Dashed (simple): the geometry consists of several short lines including additional points, every second line is drawn that results in a dashed line style; this mode starts with a line at every beginning of a hatch line
Dashed (interleaved): the geometry consists of several short lines including additional points, every second line is drawn that results in a dashed line style; this mode starts with a line and a space alternative at every new beginning of a hatch line
Dashed (continuous): the geometry consists of several short lines including additional points, every second line is drawn that results in a dashed line style; this mode starts with a line or a space at the next beginning of a hatch line depending on the previous hatch line, the pattern is kept continuous
Dotted: no lines are drawn, the geometry consists of a set of not connected dots
Dotted (Custom): no lines are drawn, the geometry consists of a set of not connected points in custom dot style as configured within the parameter settings (please refer to section 3.4 Project Configuration above)
Dotted pairs: no lines are drawn, the geometry consists of a set of not connected pairs of dots with the given distance between each pair and halve of the distance between the dots of a pair
Dotted pairs (custom): no lines are drawn, the geometry consists of a set of not connected pairs of dots with the given distance between each pair and halve of the distance between the dots of a pair using the custom dot style as configured within the parameter settings (please refer to section 3.4 Project Configuration above)

– Distance – whenever a non-continuous line style is selected, this distance value specifies the length of the additional points that form the selected line style

– Offset – whenever a non-continuous line style is selected, the related pattern is calculated in a way so that it starts at the beginning of the related hatch line. Dependent on the filled shape this may res-

ult in line patterns which are aligned in a way which is not intended. Using this parameter an additional offset can be added to each line incrementally, so that the beginning of the hatch pattern is shifted by this offset. As soon as the real, accumulated offset value becomes bigger than the line distance defined with previous parameter, the distance is substracted from the current offset value

– Align dots to raster – whenever a non-continuous line style is selected, the related pattern is calculated in a way so that it starts relative to the beginning of the related hatch line taking an optionally set offset-value into account. When the current hatch angle (as described below) defines an exact horizontal or vertical line (0, 90, 180 or 280 degrees) and this option is set, the pattern for the line is no longer calculated relative to the beginning of the line but aligned to the coordinate 0,0 of the underlying coordinate system. This is useful e.g. for polygons with different edge angles where e.g. a dot-pattern would result in an irregular, uneven distribution of dots.

– Hatch Style – here the hatching mode is defined, depending on the selection made here a different pattern is used to fill a polygon
<u>Lines, forward</u> – filling is done with parallel lines that go all into the same direction
<u>Lines, forward/backward</u> – the polygon is filled with parallel lines that change their direction
<u>Connected lines</u> – this mode extends the previous one, here the parallel lines are connected and form some kind of waved lines
<u>Zig-zag</u> – these lines are no longer parallel, they exchange their direction alternative and are connected at their ends directly
<u>Inner lines</u> – this mode repeats the outline of the hatched polygon in its inner side as often as it fits with the chosen hatch distance
<u>Outer lines straight</u> – this mode is similar to "Inner lines" but not a plain hatch mode that can be used to really *fill* a geometry. It repeats the outline of the hatched polygon on its outer side as often as specified by the value given in field "Number of Lines". At corner points where additional lines are necessary to close the polygon, the shortest way is used by adding a straight line.
<u>Outer lines rounded</u> – this mode is similar to "Inner lines" but not a plain hatch mode that can be used to really *fill* a geometry. It repeats the outline of the hatched polygon on its outer side as often as specified by the value given in field "Number of Lines". At corner points where additional lines are necessary to close the polygon, a round shape is created. Please note: this adds a lot of polygon points and may – dependent on the geometry – increase memory and processing time consumption dramatically. So this mode should be handled with care and only in cases where it is really necessary to use this special shape!

– Hatch Angle – this angle specifies the orientation of the hatch lines according to the coordinate system, it does not apply to the inner/outer lines hatch modes

– Angle Offset – this parameter applies only in case the hatcher is set as sub-element of a group that contains more than one geometric element to be hatched; in this case only the first primary element within that group is hatched with the original hatch angle, for all following elements the hatch angle is modified by this offset. So this parameter can be used to have some kind of rotating hatch angle for several elements that are concatenated together within a group. It does not apply to the inner/outer lines hatch modes.

– Repeat – this option becomes enabled when an angle offset is defined with previous parameter. When "Repeat" is set to values greater than 0, the hatch generation is repeated for the given number, each with a different angle according to the given offset. So this function can be used to create complex hatch patterns with rotating angles while using one hatch element only.

– Hatch Distance – this is the distance that is kept between the hatching lines, for the zig-zag hatching mode it applies to the ends of the lines

– Hatch Offset – normally hatching starts at the border of a polygon; this parameter allows to specify an offset (in positive and negative direction) to modify the position where hatching really starts

– Align hatch to raster – when the current hatch angle (as described above) defines an exact horizontal or vertical line (0, 90, 180 or 280 degrees) and this option is set, the hatch lines are no longer calculated relative to the beginning of the element to be hatched but aligned to the coordinate 0,0 of the underlying coordinate system. Together with the function "Align dots to raster", a custom dot style and the "Dotted (Custom)" hatch mode this function can be used to generate support structures that consist of single pins instead of filled areas.

– Number of Lines – this input field is activated only when "lines" hatch styles are used; it specifies the maximum number of inner or outer lines to be calculated; when this value is smaller than the number

of inner lines required to fill a geometry completely, inner part of the element will contain a non-hatched hole

– Interleave Lines – when this parameter comes with a value greater than 1, hatch lines are not marked one after an other but by dropping that number of lines after each, that is specified by this parameter. The remaining lines then are marked in additional cycles skipping the same number of lines in between. This mode slows down marking process caused by many additional jumps but gives the possibility to let the material not heat up too much. This can be useful in case of temperature-sensitive materials, there the currently marked areas get some time to cool down until the lines in their direct neighbourhood are processed. As an example: instead of marking lines 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 with an interleave-value of 4 it would mark lines 1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12.

– From bounding box – hatching is started from the (not visible) bounding box, when this option is set, hatch lines start outside of the polygon; this function should not be used in cases where the bounding box is the same as the shape of the hatched geometries which is true for all rectangles which are aligned parallel to the working area border

– At border – when this option is set, polygons are no longer filled completely but only at the borders, the given distance in unit specifies the "thickness" of the resulting border hatch in unit mm:
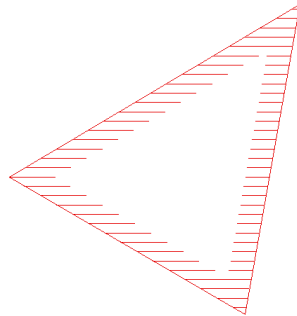
– Beam Compensation – the laser beam that is used to process a working piece itself has an own size that results in the width of the lines that are processed by it; for the hatcher it might be necessary to compensate this lasers spot size to avoid parts of the working piece to be processed more than once by the overlapping beam, this compensation can be chosen here:
Use spot-size from pen – here the tool compensation sizes are calculated automatically out of the spot sizes of the used pens
Custom –when this option is used a custom value for compensation can be entered

– Split To Tiles – when this option is set, the hatch lines are split into tiles that are aligned to the working area and that subdivide the hatch pattern additionally. Using these tiles it can be avoided to heat up the material during hatching too much. When this option is set, the following parameters can be used to specify exact behaviour of this function. After enabling this option, the hatches are split and reorganised but there will be no obvious visible representation of the changes, it can be noticed during marking or via the voordinate values only.
Together with the following parameters this function can be used to e.g. mark every second tile:
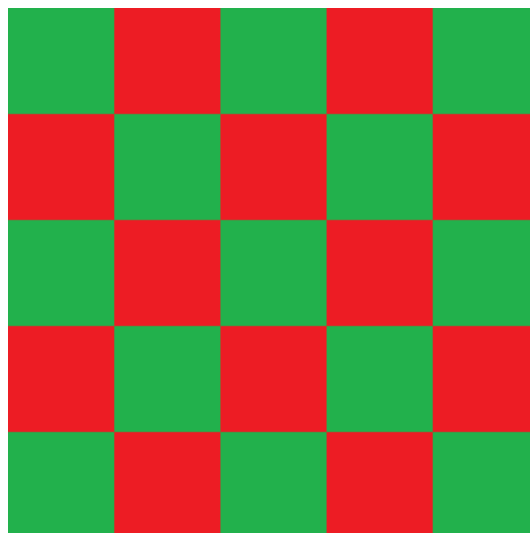
Assumed the working area has a size of 100 mm x 100 mm, for this behaviour "Tile Size" needs to be set to 20 mm and "Tile Skip Size" has to be set to 2. With these settings first all these areas are marked that have vectors within red tiles, finally all the remaining areas that have a vector in green tiles are marked. Tiles itself are processed from upper left to lower right corner row by row.
It is also possible to have even bigger distances between tiles so that they do not touch at their edges:
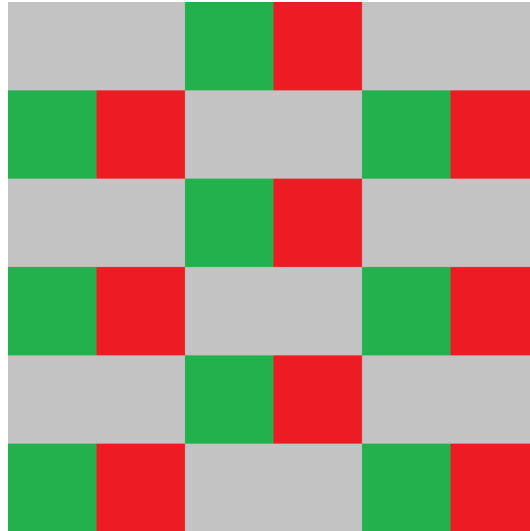
Parameters for a 90 mm x 90 mm working area are 15 mm for "Tile Size" and 4 for "Tile Skip Size". This results in a behaviour where marking starts with these areas of the hatch that own vectors in red tiles, next the areas in green tiles are marked. This is continued until the whole working area was processed (in this example by processing the remaining two tile sets in grey areaswith two additional steps).

– Tile Size – the size of a tile in working area, this value should be such a fraction of the working area size that lets the tiles fit fully into whole working area

– Tile Overlap – when this parameter is set to a value greater than 0, it specifies how much neighbouring lies should overlap, when it is smaller than 0, the value sets a gap between such neighbouring tiles using the given size in unit mm

– Tile Skip Size – skip distance between tiles, this value decides how many tiles in horizontal direction should be skipped before marking continues in next tile; remaining tiles are processed on next cycle; this value should be much smaller than the total number of tiles that fit into the working area

– Tile drop offset – when this parameter is set to a value greater than 0, the hatcher no longer creates a complete set of tiles resulting in a closed hatch for the complete element. Instead of this, the parameter specifies at which tile the function should start dropping the tiles so that there exist non-hatched parts in the whole geometry. This function can be used to combine several hatches with different hatch angles in tiles. As an example: when one hatch is set with a hatching angle of 0 degrees and a tile drop offset of 1 and the next hatch on the same element is set with a hatch angle of 90 degrees and a tile drop offset of 2 (while both use a tile skip size of 2), the result is a closed, tiled hatch with alternating hatch angles of 0 and 90 degrees in the different tiles.

– Hide hatch from view – this option does not influence generation of data, so no matter if it is set or not, the marking result will stay the same. When this checkbox is set, the hatch is not shown in the drawing area in order to make displaying of the whole project faster. This can be useful in case of very complex hatch patterns with millions of single lines or dots

– Save as default – when this button is pressed, the current hatch parameters are saved as default values. On next application start and on first usage of the hatch element, these saved parameters then will be used instead of the predefined ones

– Save as default – when this button is pressed, the current hatch parameters are saved as default values. On next application start and on first usage of the hatch element, these saved parameters then will be used instead of the predefined ones

3.7.4   Auto-Arrangement of 3D Meshes in Sliced 3D Mesh Groups

A Sliced 3D Mesh Group can hold several 3D meshes. They appear as several sub-elements of type "3D Mesh" in element tree. These 3D Meshes can be positioned with the standard functions in toolbar on windows upper side in order to place them at suitable positions in working area. Alternatively it is possible to let RepliSLS3D arrange these meshes automatically. To do this, the menu "Edit" menu-item "Auto-arrange 3D meshes" has to be selected. Now a dialogue opens which provides options to arrange the 3D meshes in X- and Y-direction and to align them at a specific Z-height. This is done via the configuration possibilities within the dialogue:

• Arrange in X and Y – when enabled, the 3D Meshes are arranged on working area using the following parameters

• Distance to borders – specifies the minimum distance the 3D Meshes need to have to the border in both, X and Y direction of the working area

• Distance between meshes – specifies the minimum distance the 3D Meshes need to have between each other

• Arrange in Z – when enabled, the 3D Meshes are aligned at a specific height using the following options and parameters

• Align geometries at top or bottom – here it can be selected if all the meshes need to be aligned at their upper or lower position

• Arrange in Z – this is an option which can be enabled when 3D Meshes have to be top or bottom aligned at a specific, absolute height; when it is set, the height has to be specified in unit mm; in case this option is not set but arrangement in Z is enabled, all meshes are aligned at the highest/lowest top/bottom position of that 3D Mesh, that is located at the highest/lowest position (according to its outlines top/bottom border)

3.7.5   3D Elements

Following all element types are described which are directly related to a (sliced) 3D model. All the elements described here are children of a Sliced 3D Mesh Group as described above in section "3.7.2 Slicing the imported model – Sliced 3D Mesh"

3.7.5.1   3D Mesh
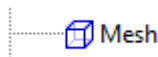
This element is a 3D model which can be created by drawing a 2D geometry ("3.9 Geometry Editing") and extruding it ("3.3.1 Menu Items") or by importing an already existing 3D model. Several 3D meshes can be assigned to one parent Sliced 3D Mesh Group and will be sliced and built all together. The 3D Meshes below of such a Sliced 3D Mesh Group can be rearranged freely in order to use the build-space as effective as possible and offer the following options in their property tab-panes:

• Show 3D Mesh – when this option is set, the related geometries are shown not only in 3D view but also in 2D top view; by default this option is disabled because it may slow down 2D graphics view nameable, instead of this there only the outline of the 3D mesh is shown

• Support structure model – this option has to be set when a 3D model was imported which acts as support-structure of an other 3D mesh which is also part of the current project. In such a case the

support-generation functions of RepliSLS3D are deactivated and the support pen settings out of the property pane of the Sliced 3D Mesh Group are applied to the slices of this 3D mesh

3.7.5.2   3D Slice Cleaner
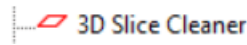
This kind of element typically is used in case of deep engraving applications but is flexible enough to be used in other areas too. It is not related to any geometry directly. When set within a Sliced 3D Mesh Group, during processing of the slices it is executed from time to time and can be used to clean up the current working area. The geometry of this element consists of the outline of the concatenated previous slices since the last slice cleaner was executed. Combined with Hatch elements (that can be added freely to it), it can be used to clean exactly that area that was processed during the last slices. The 3D Slice Cleaner and its hatch sub-elements all can make use of own pen settings.

Following parameters and settings are available in the properties tab-pane of this element:

- Clean every x. slice – defines the number of slices after that a cleaning cycle has to be performed. Here "x" specifies the number of slices which are processed without interruption and where the geometries are concatenated. Then after the xth slice has been finished, that geometry which is the sum of the previous slices is used to perform a clean-cycle – using the pen settings and hatch which is assigned to the 3D Slice Cleaner element.

- Derive cleaning shape from – here a mode can be chosen how the cleaning shape has to be created. Following options are possible:
  - Concatenated slices – in this mode the shapes of all slices are concatenated so that the resulting cleaning shape is the sum (aka "boolean add") of all slices which have been processed until current position; the resulting, current cleaning shape can be extended in size via the "Enlarge by" parameter (please refer below).
  - First slice – when this mode is selected, only the very first slice is used to generate a cleaning shape from for the complete 3D model. When the first slice has no or a too small geometry, the next slice which has a valid geometry is used. The resulting, current cleaning shape can be extended in size via the "Enlarge by" parameter (please refer below).
  - Generic rectangle – instead of a complex shape which is calculated out of the slices of the related 3D model, in this mode a simple rectangle is used as cleaning shape. The rectangle itself is the 3D Slice Cleaner element, means it can be positioned and dragged directly within the drawing view or its size and position can be entered in the tab-pane "Geometry". So in this mode editing of the rectangle to be used is possible in the same way like dragging and scaling can be done like for any normal element. Please note: rotation of the shape is not supported yet.
  - Generic circle – instead of a complex shape which is calculated out of the slices of the related 3D model, in this mode a simple circle is used as cleaning shape. The circle itself is the 3D Slice Cleaner element, means it can be positioned and dragged directly within the drawing view or its size and position can be entered in the tab-pane "Geometry". So in this mode editing of the circle to be used is possible in the same way like dragging and scaling can be done like for any normal element.

- Enlarge by – this parameter is related to the concatenated geometry of the cleaner-element and is valid for modes "First slice" and "Concatenated slices". The value given here in mm specifies the amount this geometry has to be enlarged in order to cover an area which overlaps the one which was used by the slices effectively..

3.7.5.3   Embedded Process
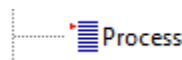
This element is used to point to a sub-project which consists of control-elements and which is executed during build-process once after every slice. So such a sub-project can be used to perform mechanical operations which are necessary between slices (like applying of new power, moving of Z-stage, switching of external devices, waiting for external devices,…) and are described in detail in section "3.8 Inter-Slice Processing".

3.7.5.4   Slice

This element neither has some specific parameters nor can be edited directly, it is generated when imported models are sliced. It is a place-holder for every slice which is generated by this step.

This element can have an unlimited number of hatch-elements to create solid slices and a Support-element symbolising support-structures generated for this model.

3.7.5.5   Support Structure
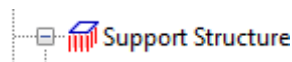
This element neither has some specific parameters nor can be edited directly, it is generated when imported models are sliced and it is always a children of a Slice element. It is a place-holder for every support structure which is generated by this step.

This element can have an unlimited number of hatch-elements to create solid support elements.

3.7.5.6   Hatch
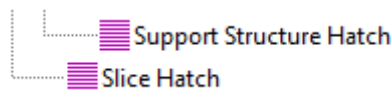
This element can exists several times below of a Slice or below of a Support Structure element. It is used to fill the slices and to create solid layers. For details please refer to section "3.7.3 Filling a 3D Model"

3.8   Inter-Slice Processing

Between two slices it is necessary for most applications to perform some additional steps in order to access external hardware (e.g. to move the Z-stage, to provide new material, to clean up working area, to wait for external equipment to finish their operation and many things more). Such sequences have to be defined in a completely own project by building it up using the blue toolbar button elements below of the "control"-symbol:

After the processing sequence works as expected, it can be saved as an own project and then configured in project settings (tab-pane "Motion") in order to be used between slices in a build process.

The sequence itself can be tested separately – simply by selecting menu "Process", sub-menu "Mark" and then letting this project "Mark".

The different elements can be added to the project simply by clicking the control-tool in the toolbar which opens a drop-down menu of available control-elements. Then they appear in the entity-tree on the main window's right-hand side in the order they will be processed with. After selecting one of the elements its parameters can be changed in the property-panel on the main window's left hand side.

Following elements are supported:

3.8.1 Delay

At the position where this element is added, the full process is delayed for the given time. Thus this element can be used whenever a delay is necessary e.g. to let material become dry, let material become colder or whatever.

This element expects exactly one parameter:

– Delay – the delay (in unit seconds) the full process has to stop

PLEASE NOTE: the accuracy of the delay given here depends on the used hardware. When the scanner controller supports delays, it is handled by it directly and is normally very exact. When the scanner controller does not support this, the delay is done within RepliSLS3D and much less accurate since timer resolution of the host operating system is quite low and transmission times to and from scanner controller card are unpredictable.

3.8.2 Motion

At the position where this element is placed additional motion axes (apart from the axes of the used scanner) can be driven.

The motion can be influenced by the parameters that have to be set for this element. First of all up to three axes can be controlled by the element using following parameters:

– Enable – specifies if an axis has to be changed by the following parameters or if it has to be left untouched, axes that are not enabled stay at their current position

– the new position to move the axis to; this field is highlighted in yellow when the current settings of the axis may lead to a situation where the position value entered here is not used but overwritten during marking. This is possible e.g. in cases where an axis is bound to the height of a slice while marking 3D slice groups. For additional information please refer to axis settings as described in section "3.4 Project Configuration".

– Speed – the speed for the movement to the given position

– Relative – defines a non-absolute movement, means the axis does not drive to the given position but changes its current position by the given distance

– Stop axis – the motion of an axis is stopped immediately, this option is available only for motion controllers that support this function and ignores the given position and speed values

– Move axes to home-position – the axis is moved to its home / reference position, this option is available only for motion controllers that support this function and ignores the given position and speed values

– Don't wait for end of motion – in normal operation mode full processing is halted as long as a motion operation is running. When this option is set, this is no longer the case: then the motion operation is started and without waiting for its end, the following elements of the current project are executed in parallel to the possibly still running motion.
For such an asynchronous motion operation some thing shave to be noted:
1. Prior to any other motion operation to be started, the current one needs to be stopped by adding an other Motion element with the "Stop axis" option set.
2. At the end of the current project it is recommended to add an other Motion element with the "Stop axis" option set in order to leave the project with a defined state

3. This option is always valid for all axes of a motion controller, it is not possible to wait for the end of one axis and to let an other axis work asynchronously, so it can be turned on and off for all axes together only

All other parameters influence the whole motion element:

– Pre-Motion Delay – this delay is done before the motion is triggered

3.8.3   Custom Output

Most scanner controller cards provide several analogue and/or digital outputs. As soon as a scanner controller card is configured within the project settings this element can be used to send values to these outputs. This Primary Element does not contain any geometry so clicking the related tool-bar button immediately adds it to the current project.

The exact layout of the related property panel highly depends on the capabilities of the used scanner controller card. There options are shown depending on the outputs the used card provides.

So it is possible that following types of outputs can be enabled for modification and set within this element:

– Laserport – a numerical value according to the valid range of the (LP8) laserport (e.g. 0..255) can be entered, this value is sent to the card during processing in case the related check box is enabled

– Pilot laser – a digital output which normally is used to turn on and off the pilot laser automatically and therefore is assigned to the laser functionality directly. When such an output interface is available and not used anywhere else by the current configuration, it can be set to HIGH ("1") or LOW ("0")

– LaserB – a digital output which normally is used to control the laser timing automatically and therefore is assigned to the laser functionality directly. When such an output interface is available and not used anywhere else by the current configuration and configured to be used as general purpose output, it can be set to HIGH ("1") or LOW ("0")

– Main oscillator – a digital output which normally is used to control the laser automatically and therefore is assigned to the laser functionality directly. When such an output interface is available and not used anywhere else by the current configuration, it can be set to HIGH ("1") or LOW ("0")

– Digital – several buttons are shown that influence the related output ports bits and can be toggled into different states: "X" means this bit is left unchanged, "1" sets a bit to HIGH and "0" clears it; here the first button is the highest bit while the last button is always bit 0

– Analogue – a numerical value according to the resolution of the analogue output (e.g. 0..1023) can be entered, this value is sent to the card during processing in case the related check box is enabled

– Pulse Output – when this option is not set the output(s) simply is/are set according to the settings that are done above. In case this option is enabled and a time in unit milliseconds is given, the output it switched to the defined state only for this time, after that it is set back to its previous state. So this option can be used to emit only pulsed signals instead of a permanent output value.

PLEASE NOTE: The scanner controller card has to be set up and configured BEFORE this element is used for the first time. This is necessary because the capabilities of the scanner controller card need to be known for this element so that it can show the correct output options that belong to the card. When a completely different scanner controller card is selected for a project, it is necessary to save this project, close and restart the application to force a re-initialisation, load the project again and then check if all Custom Output elements still use the correct ports and values.

This element is symbolised by this toolbar icon:

3.8.4   Custom Input

Most scanner controller cards provide several digital inputs. As soon as a scanner controller card is configured within the project settings, this element can be used to read values from these inputs and to halt pro-

cessing of current project until a predefined state is found there. This Primary Element does not contain any geometry so clicking the related tool-bar button immediately adds it to the current project.

The exact layout of the related property panel highly depends on the capabilities of the used scanner controller card. There options are shown depending on the outputs the used card provides.

So it is possible that following types of inputs can be enabled and used within this element:

– Digital – several buttons are shown that read the related input ports bits and can be toggled into different states: "X" means this bit is ignored, "1" expects a HIGH value at this input and "0" a LOW; here the first button is the highest bit while the last button is always bit 0

– Display message during wait – when this option is set, a text can be entered which then is shown during operation as long as the software is waiting for the input signal to arrive as specified by the options above. This text is shown in a top-level window which is closed only when mark operation is stopped or when the expected input signal is detected and processing of the current project continues

PLEASE NOTE: The scanner controller card has to be set up and configured BEFORE this element is used for the first time. This is necessary because the capabilities of the scanner controller card need to be known for this element so that it can show the correct input options that belong to the card. When a completely different scanner controller card is selected for a project, it is necessary to save this project, close and restart the application to force a re-initialisation, load the project again and then check if all Custom Input elements still use the correct ports and values.

This element is symbolised by this toolbar icon:

3.8.5   External Trigger

Whenever the control-project reaches this elements position during execution, all output is stopped until an external trigger signal is detected. Here the external trigger according to the used scanner controller cards capability is used. Within the elements tab-pane the controller card has to be chosen, where the external trigger will appear.

PLEASE NOTE: starting the geometry after a trigger is received is done by the scanner controller, the time between detection of the external trigger signal and output of the next geometry depends on the used scanner card and typically is less than some microseconds.
Waiting for a trigger after the last geometry has been sent (or after the last Primary Element was processed completely) is controlled by the application. Here the reaction time highly depends on the used operating system and the available computing power. So it is recommended to ensure a gap of of at least 20 milliseconds after the last element was processed until the external trigger may appear.

3.8.6   Z-Shifter

At the position where this element is placed, a focus shifter or beam expander is accessed. The related Z-Shifter plug-in has to be configured in projects hardware settings.

The focus/z-position can be influenced by the parameters that have to be set for this element:

– Position – the new position to move the z-position to

– Speed – the speed for the movement to the given position

– Relative – defines a non-absolute movement, means the axis does not drive to the given position but changes it current position by the given distance

– Move axes to home-position – the axis is moved to its home / reference position, this option is available only for motion controllers that support this function and ignores the given position and speed values

All other parameters influence the whole motion element:

– Pre-Motion Delay – this delay is done before the motion is triggered

3.9 Geometry Editing

Within RepliSLS3D it is possible to draw simple, two-dimensional geometries which then can be extruded into third dimension to become a 3D-mesh which is slicable as described above. Such hand-drawn meshes can be used e.g. to create inverted models (by surrounding existing geometries completely) or to add some construction details.

Adding a geometry can be done by selecting a shape below of the blue "draw"-toolbar-button:

To create a new primary element following steps have to be performed:

1. switch to 2D view, for some of the elements: switch to 2D top view

2. select the related button below of the "draw"-tool within the tool-bar, now the toolbar top button symbol changes to the one of the selected shape

3. click into the working area to position the element; depending on the geometry that is created by the related function it can be drawn now with some additional clicks into this area

4. now the new element is selected and its parameters are shown within the first, element-specific tab-pane on the main windows left hand side

When the parameters of such an element are changed within its tab-pane, all following elements of the same type use exactly these parameters as default values.

When a shape was selected accidentally and does not have to be drawn, clicking the toolbar-button again disables this drawing function. As a visual feedback the toolbar buttons image switches back from the shape to the "draw"-symbol as shown above

Following the available elements, the steps to create them within the working area and their parameters are described.

3.9.1 Triangle

To draw a triangle all three corners have to be specified separately by single left mouse clicks for each. First the starting point is defined, with the next click an other corner of the triangle is specified and the last one completes the triangle by setting the third corner of it.

Drawing functionality can be extended by pressing one of the keys "Shift" or "Ctrl" after specifying the first corner point of the triangle (first click): when Shift is held down, a horizontal line is drawn next, and vertical deviations of current mouse position are ignored. When Control is held down, a vertical line is drawn, and horizontal deviations of current mouse position are ignored.

Within the triangles property tab-pane following options are available:

– Line Style – the style the triangles lines are created with, here following options exist:
Continuous: the geometry consists of exactly tree points that are connected by three completed lines
Connected Lines: the geometry consists of several short lines connecting the three corners including additional points, all these lines are connected and therefore form a closed shape

– Distance – whenever a non-continuous line style is selected this distance value specifies the length of the additional points that form the selected line style

3.9.2   Rectangle

When drawing a rectangle the upper left and the lower right corner of it are specified by two separate mouse clicks. The resulting rectangle will always be parallel to the coordinate axes.

Drawing functionality can be extended by pressing the "Shift"-key after specifying the first corner point of the rectangle (first click): when this key is held down, a square is created instead of a rectangle. Horizontal distance from first point specifies the side length of the square.

Within the rectangles property tab-pane following options are available:

–   Line Style – the style the rectangles lines are created with, here following options exist:
    Continuous: the geometry consists of exactly four points that are connected by four completed lines which form a rectangle
    Connected Lines: the geometry consists of several short lines connecting the four corners including additional points, all these lines are connected and therefore form a closed, rectangular shape

–   Distance – whenever a non-continuous line style is selected this distance value specifies the length of the additional points that form the selected line style

–   Nominal width – this is the width of the rectangle (in X-direction) as done by this primary element; so this value does NOT reflect possible scale factors done in "Geometry"-tab-pane, the real geometry width is shown only there

–   Nominal height – this is the height of the rectangle (in Y-direction) as done by this primary element; this value does NOT reflect possible scale factors done in "Geometry"-tab-pane, the real geometry height (after all transformation operations have been applied) is shown only there

–   Corner radius – when this value is greater than 0 and when the rectangle is large enough, it is created with rounded corners

–   Segments – when the rectangle is created with rounded corners, this value specifies of how much segments all these corners consist and therefore how smooth they are

3.9.3   Circle

This primary element can be used to create circles and arcs. Starting point is always a circle that is drawn in two steps: the first left mouse click into the drawing area specifies the centre point of it, the second one specifies the radius (via its distance from the first point).

Next the circle can be modified further:

–   Line Style – the style the circle is created with, here following options exist:
    Continuous: the geometry consists of several short, fully connected lines that form a closed circle; for this mode the parameter "Segments" specifies how much of these short lines are used to create this circle, the more lines are used the more smooth the circle becomes
    Connected Lines: the geometry consists of several short lines that form the circle, comparing to the previous option here the parameter "Distance" influences the length of these lines and therefore the resolution; here the shorter the lines are the more segments this circle consists of and the more smooth it becomes

–   Distance – this parameter decides how long the resulting lines of the chosen line style have to be, it is not valid for the continuous line style where the length would be always equal to the circumference

–   Radius – the radius of the circle

–   Second radius – this is an optional parameter. When enabled, a radius value can be entered here which is used to create a second circle. Both, the primary and the second radius can be used to form specific shapes like rings.

–   Segments – here it can be chosen how much segments the circle has to consist of, this parameter influences the accuracy of the resulting circle; for the line style "Dashed (Arcs)" it influences the smoothness of the resulting arcs while the "Distance"-parameter influences their length

- Start Angle and End Angle – these two parameters can be used to change the circles starting and end angle, it can be used to draw arcs (in case the resulting total angle is smaller than 360 degrees) or to let a circle be drawn several times (in case the resulting total angle is a multiple of 360 degrees)
- Draw CCW – normally the circle has a clockwise orientation, means it is drawn to right direction; when this check box is set, the drawing direction is reversed

3.9.4 Star

This element is a special variant of a circle, it consists of two radii where every second point the element consists of, belongs to an other radius. Resulting from that stars can be drawn and modified by its parameters. Such a star is drawn in two steps: the first left mouse click into the drawing area specifies the centre point of it, the second one specifies the first radius. While drawing the star the second radius is set automatically to value smaller than the first radius.

After the star is drawn it still can be modified interactively. Here every second point (the ones that belong to the second radius) is marked by a small, blue cross. This cross can be dragged with the mouse in order to change the second radius independent from the first one and in order to apply an offset to the angle of these points.

Next the star can be modified via its options:

- Line Style – the style the circle is created with, here following options exist:
  Continuous: the geometry consists of single, continuous lines
  Connected Lines: the geometry consists of several short lines including additional points, all these lines are connected
- Distance – this parameter decides how long the resulting lines of the chosen line style have to be, it is not valid for the continuous line style
- Radius 1 – the first radius that is valid for all odd-numbered points
- Radius 2 – the second radius that is valid for all even-numbered points
- Spikes – here it can be chosen how much spikes (or rays) the star shall have,
- Angle offset – here an offset can be specified for the points that are assigned to the second radius in order to create non-regular spikes
- Draw CCW – normally the star has a clockwise orientation, means it is drawn to right direction; when this check box is set, the drawing direction is reversed

3.9.5 Polygon

A polygon consists of an undefined number of edges (but requires at least three). Drawing it can be done by left clicking at the position where a corner between the connecting lines has to be positioned. This can be done several times, the process of drawing this polygon can be finished by right-clicking the mouse. Then the last corner and the first position where the user left-clicked are connected by a line.

Drawing functionality can be extended by pressing one of the keys "Shift" or "Ctrl" after specifying the starting point of the polygon (first click): when Shift is held down for one of the following points, a horizontal line is drawn next, and vertical deviations of current mouse position are ignored. When Control is held down, a vertical line is drawn, and horizontal deviations of current mouse position are ignored.

Following options are available to modify the parameters of the polygon:

- Line Style – the style the polygons lines is created with, here following options exist:
  Continuous: the geometry consists of lines that use exactly one starting and one end point where one single line is drawn between
  Connected Lines: the geometry consists of several short lines including additional points, all these lines are connected
- Distance – whenever a non-continuous line style is selected, this distance value specifies the length of the additional points that form the selected line style

3.9.6 Text

This primary geometry can be used to draw some text using an existing TrueType or special laser font in 2D top view. When a project, which contains this element, is moved from one computer to an other or from one operating system to an other it has to be made sure a used font is available on the other system too (this applies to TrueType fonts only since the fast laser fonts are part of RepliSLS3D). Otherwise the application tries to find a similar font which might be only somehow similar, but exact positions, sizes and outlines may be got lost in such a case.

Drawing such a text is quite easy: only one single click into the top view drawing area is necessary to specify the position of the text, there a new geometry is set using a default text. This text can be changed together with some other parameters:

–   Line Style – the style the lines the text consists of is created with, here following options exist:
    Continuous: the geometry consists of closed, connected lines with as less as possible points
    Connected Lines: the geometry consists of several short lines including additional points, all these lines are connected
    Dashed: the geometry consists of several short lines including additional points, every second line is drawn that results in a dashed line style
    Dotted: no lines are drawn, the geometry consists of a set of not connected dots
    Dotted (Custom): no lines are drawn, the geometry consists of a set of not connected points in custom dot style as configured within the parameter settings (please refer to section "3.4 Project Configuration" above)

–   Distance – whenever a non-continuous line style is selected, this distance value specifies the length of the additional points that form the selected line style

–   Text – here a multi-line text can be specified that has to be drawn; a modified text is applied as soon as the text input area is left, either by clicking an other element or by pressing the Tab-key

–   Unicode – instead of characters here space-separated Unicode values can be entered either as decimal numbers or as hexadecimal numbers with a leading "0x". On pressing "Return" the entered numbers are converted to text and shown as their character-representation in input area "Text"

–   Font Type – here the type of font that has to be used for this text element can be chosen, following options are available to select from:
    TrueType: the standard fonts that are installed on the system can be used. When this option is selected, the combo box directly below of this one is enabled to select a font out of it; all these fonts are in no way optimised for output via a laser scanner system, their intended target is publishing, therefore their lines and paths are optimised for quality which may result in a slow output
    Laser Vector: this special type of fonts is optimised for laser marking operations. They consist of as less as possible lines and are designed for single-lined outputs like they are done with lasers. Using these fonts marking of texts is much faster than with the TrueType fonts. When this font type is selected the font list below of this combo box is modified to show the available laser fonts to choose from.

–   Standard outline – this option is enabled only in case TrueType font type is selected and it is enabled by default. When a normal TrueType font is used, it always should be selected.
    This option becomes useful when a non-standard TrueType font is used that is designed as laser font. Typically the characters of a TrueType font consist of an outline that forms the shape of the character and can be filled with a hatch pattern. Due to the lots of lines such a font is slow for laser marking. A faster, speed-optimised font would consist of single lines only which itself define the shape and typically do not need to be filled. Now when this option "Standard outline" is unset, such special fonts can be used (as TrueType does not know anything about laser marking, the option has to be set/unset explicitly by the user). Such fonts can be drawn in every standard TTF-editor and will look like laser fonts as soon as they are used in RepliSLS3D and as soon as this option is disabled:
    From left to right: normal, outlined shape of a standard TrueType font, single line shape of a laser-marking-optimised font, view of a single line font when drawn into a TTF editor (the filled area is a wrong preview of most external editors and will be gone automatically as soon as this font is used in
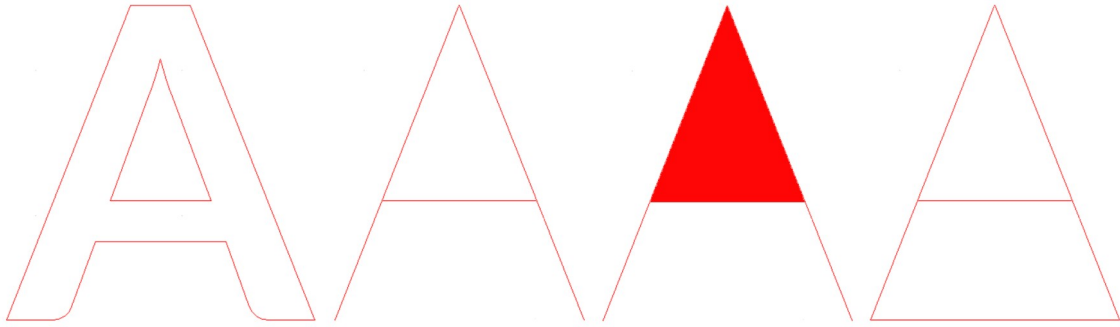
RepliSLS3D and as soon as the option "Standard outline" is disabled there), wrong representation of a laser-TrueType-font when the option "Standard outline" was not unset

– Font – this list of available fonts can be used to select a specific font and style for the text. Depending on the font type chosen above, here a specific font can be selected. In TrueType mode the number and types of available fonts depends on the operation system and which ones are installed there. In Laser Vector mode the internal laser fonts are listed here, these fonts are delivered with RepliSLS3D and do not depend on the fonts installed on the system.

– Alignment – the text is always drawn from a fixed position that is specified by the left mouse click when setting the text initially; using this list it can be chosen how the text has to be aligned to this position (left, centred or right)

– Orientation – this list specifies the direction the single characters of the text have to be drawn

– Spacing – this value can be used to modify the distance between the characters; here a value of 100% is the distance that is specified by the used font, smaller values set the characters closer to each other, larger ones stretch it to a bigger distance

– Kerning – this value can be used to influence the kerning distance for fonts that support it, here 100% is equal to the kerning value that it specified by the font, bigger values set the characters with a kerning value closer to each other, values smaller than 100% set a bigger distance between them; this parameter is different to the "Spacing"-value, it only influences character-combinations where kerning is defined for

– Arc Length – when creating geometry out of the available font information depending on the style of the chosen font different round parts have to be created (like the two arcs that are needed for creating a letter "B"). This parameter specifies the segment length for these round parts, the smaller it is the more accurate the result is but the more data are used to hold it.

– Nominal height – here the nominal height of a line of text can be specified in unit mm; this does not only include all possible characters and their heights but also the distance between two lines of the chosen TrueType font. So the height specified here in most cases will be always bigger than the vector height of every possible characters of the chosen font.
Beside of that this is the nominal value on creation of the text, means no additional scaling operations which are done with this element or with optional parent groups elements are included in this value.

– Set monospaced – when this option is checked, equal distance is set between all characters independent from their shape and independent from the distance information of the used font

– Lock size – this option can be used to force a text element to always keep a specific size. When this option is enabled, the current geometric size of the element is stored. Whenever the text changes by entering a new text, the element is scaled in a way so that the size is left unchanged. Depending on how much the real length of the texts changes, this means the characters itself are shrunk or stretched

3.9.7 Barcode

Using this function several types of 1D and 2D barcodes can be created in 2D top view. The size of the created barcode can be defined freely, when creating this element a rectangle is drawn exactly in the same way like it is done for the Rectangle Primary Element. This rectangle describes the outline of the barcode. For

some barcode types that use a fixed aspect ratio between width and height the predefined outline is modified automatically.

The complete layout and structure of the barcode is defined via the parameters that can be set within the elements tab-pane:

– Data – the text or numeric data that are encoded within the barcode, here it depends on the chosen barcode type if and which characters, numbers or combinations of them are allowed and will result in a valid barcode. This means it is <u>not</u> possible to enter any kind of data for any type of barcode! For most barcode types special formatting rules need to be respected, elsewhere no barcode is created and the primary element stays empty.
For these barcode types that support carriage return and/or line feeds ("\r" and/or "\n"), the related characters can be added by setting some special placeholders. So a "[CR]" within the data will be replaced by a carriage return character before the barcode is encoded and a "[LF]" will be replaced by a line feed character.

– Type – here the type of barcode can be chosen, depending on this type several of the following options are available and can be used for further definitions that are necessary for this type

– Mode – this is a barcode-type-specific option, here an encoding mode can be chosen for some of the barcodes

– Error Correction – this is a barcode-type-specific option, here an error-correction level can be set

– Size – this is a barcode-type-specific option, here the resulting size can be specified (which not influences the dimension of the barcode but the encoded data size)

– Bar Width – this parameter influences 1D barcodes only, here the size of the resulting bar lines can be modified; this value might be changed depending on the method of material processing in order to get a readable barcode

– Segments – this parameter is used only for the Maxicode barcode, it specifies the number of segments and therefore the smoothness of its circles

– Fade In Size – this parameter is used for inverted barcodes, it defines the size of the additional frame that is drawn around the barcode to get an inverted representation

– Force to Square – this option applies to DataMatrix barcodes only, in case it is set, the resulting barcode is always a square and never a rectangle

– Merge Cells – several barcode types may consist of cells (rectangles or squares) that are located directly beside each other; this option merges such cells wherever that is possible without destroying the barcode information, it can be used to save vector data

– Draw Cells as Dots – when this option is enabled, the cells some barcode types consist of are replaced by plain dots

– Draw Cells as Custom Dots – when this option is enabled, the cells some barcode types consist of are replaced by the custom dot geometry as defined within the settings (please refer to section "3.4 Project Configuration" above)

– Invert – barcodes created by this element consist only of the outlines, together with the Hatching Additional Geometry element these outlines can be filled. Depending on the used material processing method these filled areas may result in lighter or darker areas. In case the result is not readable by barcode scanners the light and dark areas can be exchanged by this option, here an additional border is drawn around the barcode so that the hatching function generates an inverted fill pattern

– Invert with embedded normal – this option combines the normal mode and mode "Invert" into one element. When this mode is enabled, the visual representation of the barcode is switched to "inverted" with the possibility to define fade in size for the additional bounds created around it. This inverted barcode will make use of the pen assigned to the element as usual.
Additional (but not directly visible) a non-inverted, normal barcode is created too making use of exactly the same input data like the inverted one. This non-inverted barcode is located at exactly the same position overlapping its inverted pendant perfectly. When this mode is enabled, the pen to be used for the embedded, non-inverted (=normal) barcode can be selected using the pen combobox which is shown right below the checkbox of this mode.
This special mode can be used e.g. in cases where both, inverted and normal barcode needs to be marked with different pen parameters in order to have a good contrast on resulting material. With

this mode it is not necessary to manually overlap two separate barcodes and it is guaranteed they both have the correct position and the same input data.

3.10   Marking / Building

"Marking" in sense of RepliSLS3D in general means "outputting some data to connected devices" aka "building". This can be:

- sending of a control sequence to connected devices to test an inter-slice operation

- starting a complete build process including (laser-)marking of the slices and performing the inter-slice operations as defined for the "Process" element

The mark dialogue / build dialogue can be opened via menu "Process" menu item "Build". When this dialogue is opened for the first time or in case the hardware settings have been changed after it was used the last time, as first operation the connections to scanner controller cards and the – optionally configured – motion devices are established. Motion axes that are configured to reference on start-up are moved to its reference/home position. All these operations may take some time dependent on the connected hardware.

In case this could be completed successfully, the dialogues mark-buttons can be used afterwards. It provides a button for starting the pilot laser (requires fully configured pilot laser outputs in project settings) on the left side of the dialogue, a mark-button in the middle that starts the main laser and a stop-button on the right side that stops the main or pilot laser. Below of them some options can be found that influence the marking operation:

- Wait for external trigger – using this option building is started only under two conditions: first the start-button is pressed, next an external trigger signal is applied. So using this function start of building can be controlled by external hardware.

Additional marking-related functionalities are arranged in separate tab-panes that are grouped according to the functionality they contain. tab-pane "Pilot Laser" contains functions for accessing the preview/pilot-laser that can be started by the upper left mark button:

- Mode – this combo box gives the possibility to choose the pilot laser marking mode, "Off" disables usage of the pilot laser "Manual" expects that the user starts the pilot laser manually by pressing the big pilot-laser-button on the windows upper left hand side and "Auto" re-starts the pilot laser after every marking process automatically. Also in mode "Auto" the pilot laser has to be started manually for the first time because of security reasons.

- The radio buttons below this combo box can be chosen to select what the pilot laser shall show in its preview, "Use total projects outline" draws a simple rectangle that includes the whole project, "Use individual elements outline" draws a simple rectangle for every element within the current project (except dots, lines and triangles which are shown fully) and "Use real element shapes" draws the full shape of all elements except hatches.
  The pen – and therefore the related speed and delay values – that is used for the pilot laser can be defined within the pen settings dialogue (please refer above).

Tab panel "Manual" offers direct access to the position of scanhead. Here mirrors can be moved to a specific position:

- In upper combobox the scanhead can be chosen this operation has to be performed on (valid only for multihead configurations, elsewhere "Scanhead 1" has to be used

- four arrows give the possibility to move the scanner position in positive and negative X- and Y-direction; the input field between these arrows specifies the jump length every button press has to cause; pressing these buttons lets the scanner jump immediately

- In fields "X" and "Y" coordinates can be selected the scanner has to be moved to, by pressing button "Jump" motion to these positions is started

- Below of the camera-button there is a small pilot-laser-button which gives the possibility to turn on and off the pilot laser (only available in case a pilot output is configured in hardware settings). To-

gether with the coordinate buttons it can be used e.g. for referencing purposes – when the pilot laser permanently points to a fixed, defined position which also can be changed as needed.

In tab panel Geometry the current projects vector data can be manipulated:

- Arrows left/right/up/down – using these arrows the whole geometry of the project can be shifted into the given direction (in top view for 3D mode); the shift step size in unit mm is taken out of the input field below of these arrows

- Arrows rotation left/right – using these arrows the whole geometry of the project can be rotated into the given direction (in top view for 3D mode); the rotation step size in unit degrees is taken out of the input field below of these arrows

- Button Reset – when this button is pressed all changes in position and rotation of the current projects geometry are taken back, it is reset to its default values

In tab panel Motion Axes it is possible to drive configured motion axes manually and independent from any motion elements in marking project. Please note: when there are some marking elements used directly within the project, it has to be taken into account that the user may have changed the position of the axes via these manual functions. Thus at least the first motion element should perform a movement to an absolute position or should reference its position. Within the mark dialogue following motion functions are available:

- OUT2 / OUT3 – specifies which of the two possible motion controllers has to be used

- Speed – specifies the speed a manual motion operation can be performed with; please note: the value entered within this field is not clipped to the maximum possible speed immediately, since every axis can have a different maximum speed it is decided on motion start which of these limits has to be used; thus it may be possible an axis moves with a lower speed than specified within this field simply because is maximum speed is limited

- movement value – the value that can be entered in middle between the arrows specifies the relative movement distance or the absolute motion position to move to (depending on the arrow button that was pressed)

- Double arrows – these buttons cause a movement as long as the button is pressed

- Single arrows – these buttons cause a single movement via the distance or angle specified with the number input field in the middle, here the time the button is pressed does not influence the motion

- Arrow to dot – when this button is pressed an absolute movement is performed, the number input field in the middle between the buttons specifies the target position to move to, the time the button is pressed does not influence the motion

- Ref – start referencing of the related axis

For planar movement axes the images of the buttons can be changed within the motion axis configuration panel of the general project settings. Depending on the real movement direction (left/right, up/down, in/out) a different symbol can be shown here in order to make it easier for the operator to decide which direction a motion will use. The same is true for the axis names, they can be configured via the axis alias value in general project settings.

The third tab panel Parts is related to parts and part counting, some of its buttons and functionalities will be enabled only when part counting is enabled for the current project (can be done in general project settings dialogue). It offers the following functions:

- When part counting mode is enabled the large counter shows the current number of parts already marked. When the limit specified in project settings or in input field "To produce" is reached, no more marking operations are possible until the user presses button "Reset Counter"

The tab panel "Progress" shows some information regarding the current marking progress, its functionality is limited to some specific applications. First of all it is feed with timing and operational data only when sliced 3D marking data are processed. Next it will not show the real marking state of the connected controller but only the state of data sent to the scanner card. So when a scanner controller card with a large buffer is used, it may hold some none-processed data while for this progress bar they are already finished. So this progress information is mainly suitable for long-running processes (where relative deviation between mark-dialogue-estimated and real marking time is quite small) but not for getting a process forecast for fast marking operations. For last case it is still recommended to use the function "Simulate" in menu "Process".

Beside of that it offers two buttons to control the current process – which also are available in case of some specific applications:

Pause and resume – when pressed for the first time, the currently running process is halted. This happens not immediately but the next time the software is able to perform a halt without causing damage to the current marking process. How fast this halt is done, depends on the used geometries and the capabilities of the used scanner controller card.

When pressed for the second time, a previously halted process is resumed and processing is continued seamless.

Override all stops – this button can not be used to halt a currently running process but to resume it. Different to the standard halt/resume button explained at the left, this also includes stops which have been invoked by process-control plug-ins. This function has to be handled with care: when it is used to override a stop, it may be possible a process is continued while being outside the defined process parameter ranges which itself may end up with an undefined or unusable marking result!

3.11 Multihead Operations

In many cases RepliSLS3D will use one scanner card that uses exactly one rectangular working area. In this configuration there will be one scan head performing movements for this working area:
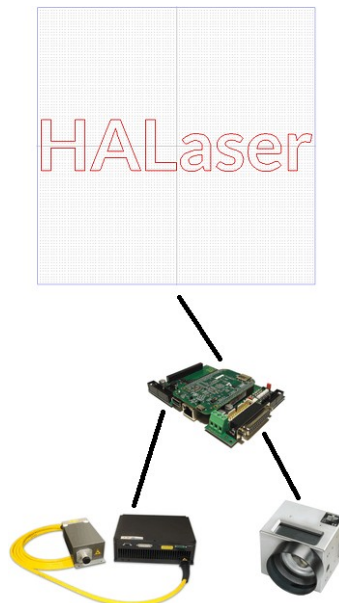
In some special cases this so called singlehead operation mode can be extended to make use of a second (and optionally third, fourth,…) working area and a second (optionally third, fourth, …) scan head where both heads perform exactly the same movement. This mode is often called "secondary head", it simply doubles the output but does not allow to mark different, independent geometries with these two heads. Although multiple heads are used in this mode, there is still one laser and one scanner controller card used, so there is no independent laser modulation possible for each of these multiplied working areas:
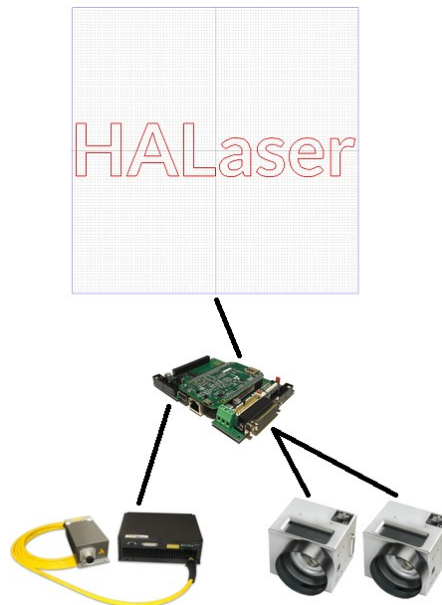
Full independence of all used heads is possible only with a real multihead mode. In this mode RepliSLS3D knows about more working areas that can overlap each other or that can exist beside each other. But all these working areas are completely independent, every area is assigned to an own scan head that performs separate movements and every connected laser is modulated separately:
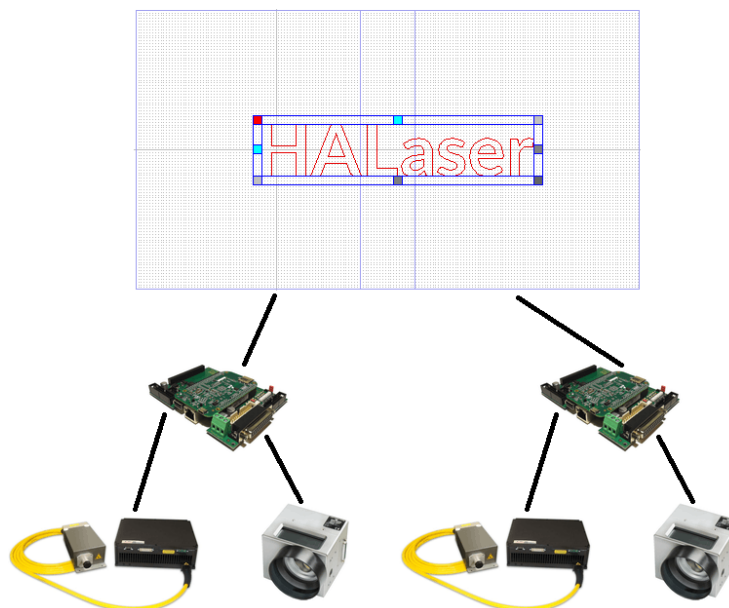
This multihead mode can be used to:

- mark geometries that are larger than a single working area (in this case it is recommended to let the working areas overlap a bit to profit from vector data optimisations RepliSLS3D performs automatically)

- mark large, complex geometries faster (in this case big overlapping areas have to be configured to let both heads work on same area in parallel)

- mark independent geometries faster (in this case working areas do not need to overlap each other)

In last case it is recommended to use scan heads with the same performance and lasers with same power, elsewhere it may happen one marking area slows down the full marking process.

In first two cases different lasers and scan heads can be used but it is recommended to have the more powerful lasers and the faster scan heads configured at the first position.

Setting up of a multihead configuration is quite easy and can be done in general project settings (please refer to detailed description of settings in section 3.4 Project Configuration above). There in tab-pane "Scanner" two combo boxes can be found. One is for selecting a scanner controller and one for selecting a scan head. Both can be separate things but don't have to, this depends on the used scanner controller / the used scanner controller plug-in. The difference between both:

- the scanner controller is a piece of hardware / a plug-in that accesses this scanner controller device, where more than one scan head can be connected to.

- the scan head is assigned to a single working area and a single laser that is used for this working area, it is possible to have more than one scan head connected to a scanner controller.

In many cases there will be a 1:1 relation where scanner controller selection is equal to scan head selection, but in case one scanner controller plug-in is able to control more scan heads independently, this separation between "controller" and "head" is necessary.

The same can be found in pen settings (please refer to description above). There the scan head can be selected, means different pen parameters can be set for every working area – independent from the fact which scanner controller plug-in really controls this head.

To configure a multihead-environment, following steps have to be done in global settings of RepliSLS3D:

- go to tab-pane "Scanner" within project configuration dialogue

- in upper combobox select the first scanhead (which is default)

- below of it choose a scanner controller card and configure it, optionally change the field left and top position within the scanner controller card's settings

- select the next scanhead in the combobox on top of the scanner settings tab-pane in global settings

- choose a scanner controller card and configure it → this step now creates a new working area which is assigned to this second scanner controller card and the laser and scanhead it is connected with, so in configuration of this scanner controller the field left and upper size have to be set to appropriate values in order to overlap or to not to be connected with the working field from the first scanner controller card; as soon as a second scanhead/scanner card is configured, BeamConstruct is in mutlihead-mode automatically

- optionally continue this procedure with a third, fourth, fifth,… scanner controller card and related lasers/scanheads/working fields

- optionally go to tab-pane "Multihead" within project configuration dialogue to configure a specific, preferred behaviour for each of the scanheads created in previous step

3.12   Customise RepliSLS3D

3.12.1   Rebranding

RepliSLS3D can be modified in a way so that it can be delivered together with machinery or additional equipment using an own brand. Here following attributes can be changed:

- the applications name, this value replaces the name RepliSLS3D completely

- the homepage which is shown in "About"-dialogue

- the splash image shown during application start-up and in "About"-dialogue

- the application icon shown in task-bar and main window (not supported on all operating systems)

- the copyright as shown in "About"-dialogue; this custom copyright does not replace the default one but is added to it

- the images shown in toolbar and entity-tree

To customise the application in this way a plain text file with the name `replisls.oem` has to be created in UTF-8 format. To become active it has to be placed in installation directory of the application for Windows op-

erating systems (typically `C:\Program Files\RepliSLS3D`) or in `/usr/local/replisls3d` for Linux systems. Within this file every single line contains one customisation definition:

1. the name of the application

2. the homepage URL to be shown in "About"-dialogue

3. path to the splash image in PNG format

4. path to the icon image in PNG format

5. copyright text to be shown in "About"-dialogue

As an example:

```
Beamy Raymanipulator
https://www.replisls3d.com
replioem.png
replioem_icon.png
Copyright (c) 2017 by LaserFreak
```

This file (in UTF-8 encoding) placed within the directory described above renames RepliSLS3D to "Beamy Raymanipulator", shows https://www.replisls3d.comreplisls3d.com within the "About"-dialogue as well as the additional copyright information. As splash image for the startup-screen the image out of file replioem.png is used. When this files name is given without a absolute path the application tries to find it in installation directory (Windows) or in directory `/usr/local/replisls3d` (Linux). The same is true for replioem_icon.png which is used as image for the applications icon.

3.12.2 In-application images

The images that are shown within toolbar and entity tree can be replaced by own ones too. Here two different methods have to be used to replace all of them.

To replace the toolbar- and entity-tree images replacement images in PNG-format have to be put into to the plug-ins location but with different names. The base name is always the name of the plug in without extension. For the toolbar image a "1.png" has to be added to this base name, for the entity-tree image a "2.png" has to be added. As an example: The images of the motion plug-in have to be replaced. It can be found in folder "priplugins" with the file name "libio_pri_motion.dll" or "libio_pri_motion.so". Here two images "libio_pri_motion1.png" with a size of 25x25 pixels and "libio_pri_motion2.png" with a size of 15x15 pixels have to be put into the same folder. After next software start the motion element appears with the new images.

To replace all other images PNG files with special names have to be placed in folder "icons":

- "newprj.png" for the "New"-symbol (25x25 pixels)

- "openprj.png" for the "Load project"-symbol (25x25 pixels)

- "saveprj.png" for the "Save project"-symbol (25x25 pixels)

- "saveprjas.png" for the "Save project as"-symbol (25x25 pixels)

- "quit.png" for the "Quit application"-symbol (25x25 pixels)

- "zoomin.png" for the "Zoom in"-symbol (25x25 pixels)

- "zoomout.png" for the "Zoom out"-symbol (25x25 pixels)

- "zoomworkarea.png" for the "Zoom full editing area"-symbol (23x23 pixels, for compatibility reasons the name of the image contains "work" but not "editing")

- "zoomselected.png" for the "Zoom selected entity"-symbol (25x25 pixels)

- "pointdrag.png" for the "Point drag mode"-symbol (25x25 pixels)

- "openmarkdlg.png" for the symbol to open the mark dialogue (25x25 pixels)

- "dlgmark_pilot.png" for the pilot laser symbol in mark dialogue (113x113 pixels)

- "dlgmark_mark.png" for the mark-start symbol in mark dialogue (113x113 pixels)

- "dlgmark_stop.png" for the stop button symbol in mark dialogue (113x113 pixels)

- "penwizard.png" for the pen parameter wizard button in pen settings dialogue (55x55 pixels)

- "rotate_move_left_xpm.png", "rotate_step_left_xpm.png", "rotate_step_right_xpm.png", "rotate_move_right_xpm.png" and "rotate_to.png" for rotational movement buttons in "Motion" tab-pane and mark dialogue (17x17 pixels)

- "arrow_move_down_xpm.png", "arrow_step_down_xpm.png", "arrow_step_up_xpm.png", "arrow_move_up_xpm.png" and "vertical_to.png" for planar, vertically oriented movement buttons in "Motion" tab-pane and mark dialogue (17x17 pixels)

- "arrow_move_out_xpm.png", "arrow_step_out_xpm.png", "arrow_step_in_xpm.png", "arrow_move_in_xpm.png" and "longit_to.png" for planar, longitudinally oriented movement buttons in "Motion" tab-pane and mark dialogue (17x17 pixels)

- "arrow_move_left_xpm.png", "arrow_step_left_xpm.png", "arrow_step_right_xpm.png", "arrow_move_right_xpm.png" and "horiz_to.png" for planar, horizontally oriented movement buttons in "Motion" tab-pane and mark dialogue (17x17 pixels)

3.13   Smart Interface

The interface described here is read-only. It can be used to implement Smart Factory (Industry 4.0) applications and provides different status informations which give the possibility to react on events during marking/production.

When this interface is used, it is possible to watch processing of one or more RepliSLS3D-controlled machines without the need to have an eye on them all the time. As soon as something important happens which requires user interaction, a notification is sent via the Smart Interface. This notification can be received by a suitable device (e.g. a smartphone-application) which then informs the user that something has to be done at a specific machine. Now the operator can check that machine.

3.13.1   Accessing the Smart Interface

The smart interface data are provided via an unencrypted, unauthenticated TCP/IP socket using port number 11355. RepliSLS3D sends several UTF-8-encoded XML documents via this interface. These documents provide state information but the interface does not accept any data sent by the connected client. So independent from what (unauthorized) client connects here, it is not possible to cause any dangerous operation via this connection.

This interface accepts an unlimited number of incoming connections. But to avoid an unnecessary high load on a running instance of RepliSLS3D, it is recommended to not to connect with too much clients to this interface. For security reasons it is highly recommended to not to make this interface available via the Internet but only within a protected, walled-off production network.

The data provided at the Smart Interface all come in XML format and all have one common basic structure:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg Time="2017-08-17T15:17:31" ID="901B0E5A06A2">
 …
 </BCMsg>
```

The first line is mandatory and specifies the beginning of a new XML document according to the XML specification. The second line starts the root-node `BCMsg` of every Smart Interface message. This root-node can contain two optional attributes `Time` specifying the time the message was sent (ISO 8601 time format, local time) and `ID` uniquely identifying the Machine/software installation this message was sent from.

Dependent on the kind of message which is sent, it may be extended by some more sub-nodes and attributes (symbolised by "..." in the example above).

Following the messages are described which are supported by the Smart Interface. PLEASE NOTE: although these messages are described as separate XML structures, they all can be combined within one `BCMsg`-node:

Alive Message

This message is sent latest 40 seconds after last transmission in order to signal connected clients the software is still alive. When no messages have been received by a client for a longer time (e.g. for more than 150 seconds), the client should close the connection and try to reconnect. This alive-message does not provide any further information and therefore comes with an empty `BCMsg`-node:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg/>
```

Status Message

This message is sent whenever the operational state of the application changes. It comes with a sub-node `Data` which contains an attribute `State`:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Data State="<state>"/>
 </BCMsg>
```

Here `<state>` can have following values:

`idle` – software is started but marking dialogue is not open and no mark operation is running

`ready` – mark dialogue is open but no marking operation is active yet

`marking` – a marking process is active

`paused` – a sliced 3D project was halted by pressing the "Pause"-button in mark dialogue; when this project is continued later and no error happens, it switches back to state "`marking`"

`error` – an error occurred, no marking is possible at the moment; in case this state is used, an additional Attribute `Text` provides the error text related to the current error (using the language the software is running with)

Project Message

This message specifies which project is loaded at the moment. It comes with a sub-node `Data` which contains an attribute `Project`:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Data Project="<pathname>"/>
 </BCMsg>
```

Here `<pathname>` is the full path to the project file which just was loaded. When it is empty, the "New" function was used and the last project was flushed completely.

Immediate Trace Messages

This message is submitted as soon as an element is about to be marked where the "Trace data" option is set:

```
<BCMsg Time="2020-02-07T09:11:19" ID="B0C0904A5F15">
  <Trace Name="<name>" ID="<identifier>"/>
</BCMsg>
```

The body encapsulates exactly one nodes of type `<Trace>` that contains an attribute `Name` which keeps the name of the element in project which is about to be marked. For input elements an additional attribute `ID` is contained which holds the input data which have been processed.
Please note: the information provided here only inform about the fact the current element is about to be marked. Since this software can't check if e.g. the working piece really was in place and was properly aligned, it does not necessarily mean the marking result was as expected, so additional inspection may be necessary with the working piece in order to ensure the correct result. Beside of that at this point it is also not known if the marking cycle was interrupted, possibly before the element announced with this message was processed. So to get a summary about the successful completion of a marking operation, refer to the "Number-Of-Parts Message" below.

Number-Of-Parts Message

This message returns information about the current part number which has been produced and corresponds to the parts-parameter in global settings. When a `<parts>`-value of 0 is sent, the parts-counter was reset by the operator. Additionally there can be some optional attributes `ID` which uniquely identifies the current working piece and `ProdID` which uniquely identifies the product which is currently process. These identifiers typically are set externally, e.g. when RepliSLS3D is integrated into an automated production line where previous machines provide these data via a suitable communication interface (like a Hermes connection):

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Data Parts="<parts>" ID="<id>" ProdID="<prodid>">
   <Trace Name="<name>"/>
   <Trace Name="<name>" ID="<identifier>"/>
  </Data>
 </BCMsg>
```

The `<Data>`-section may encapsulate one or more nodes of type `<Trace>` when the "Trace data" option is enabled for elements in the current project. Every `<Trace>` section contains an attribute `Name` which keeps the name of the element in project which has been processed successfully by the last marking operation. For input elements an additional attribute `ID` is contained which holds the input data which have been processed. So this functionality can be used e.g. for traceability purposes. Since the Number-of-Parts message is emitted only in case a marking cycle was performed successfully, did not fail due to an error and was not interrupted by a stop-operation, the data contained in this message can be used for further automated processing of the produced working piece. Together with an input element that provides unique data, it can be identified uniquely.
These trace information are also submitted immediately with an own message as soon as the related ele-

ment is about to be marked. For more information please refer to the "Immediate Trace Message" above. Please note: the information provided here only inform about the fact a marking cycle has be performed successfully. Since this software can't check if e.g. the working piece really was in place and was properly aligned, it does not necessarily mean the marking result was as expected, so additional inspection may be necessary with the working piece in order to ensure the correct result.

Number-Of-Slice Message

This message is sent when a sliced 3D SLS/SLM model is built. Here the slices are announced as they are submitted to the scanner controller card. So this does not necessarily mean the current slice specified by value <slice> is finished, but it was handed over to the scanner card for production. The second attribute MaxSlices provides information about the maximum number of slices this model consists of:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Data MaxSlices="<maxslices>" CurrSlice="<slice>"/>
 </BCMsg>
```

Process Data Message

This message is sent when a process control plug-in is used and when this plug-in sends measured process data. These process data are provided via this interface:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Process Name="<name>" Unit="<unit>" Idx="<index>" Value="<value>"/>
</BCMsg>
```

The data are encapsulated in a Process-tag which can exist several times within a XML package and which contains the following attributes:

Name – the name of the measured <value>

Unit – the measurement unit the <value> is given with

Idx – an index value in range 0..3; a plug in can measure up to four different values, this index specifies which of these values is provided

Value – the measured <value> which is further specified by the other attributes

Notification Message

Using this message, generic information can be submitted to a client. Such a notification message contains some text which is useful to the user and which can have different priorities:

```
<?xml version="1.0" encoding="UTF-8"?>
 <BCMsg>
  <Msg Src="<sender>" Text="<shorttext>" Prio="<prio_number>"
      PrioText="<prio_text>" Desc="<detailtext>"/>
 </BCMsg>
```

Different to the preceding ones this message provides different attributes within a `Msg`-tag:

- `Src` specifies the sender of this notification message, here either a unique machine identifier is provided (in case it is available) or a generic description

- `Text` is a short variant of the notification text, it acts as some kind of headline to give an overview what the notification is about

- `Desc` is the complete, detailed notification message

- `Prio` is a number and `PrioText` is a text corresponding to the number, both specify the priority this notification has:
  - 1 = "Fatal error"
  - 2 = "Error"
  - 3 = "Warning"
  - 4 = "Info"
  - all values >4 = "Other"

3.14 MQTT Data Interface

The interface described here is one-way only. The application connects to a MQTT broker (as configured in "3.5 Machine Configuration") and sends state information to this broker which then can be used by other clients which are connected to the same MQTT data broker.

When only this interface is used, it is possible to watch processing of one or more RepliSLS3D-controlled machines without the need to have an eye on them all the time. As soon as something important happens which requires user interaction, a notification is sent via the MQTT Data Interface. This notification can be analysed by a suitable client application which then informs the user that something has to be done at a specific machine. Now the operator can check that machine.

This interface gives extended possibilities to automate production with RepliSLS3D according to the concept of smart factories.

For more details about the MQTT protocol, please refer to https://mqtt.org

3.14.1 MQTT Messages

RepliSLS3D sends several MQTT-messages to a connected broker automatically when a related event occurs. These messages provide state information but the interface does not accept any data sent by the broker RepliSLS3D has connected to. So independent to what (possibly unauthorized) client the software may connect to, it is not possible to cause any dangerous operation via this connection.

The data provided via the MQTT data interface are structured according to the MQTT specification and contain information as described below. Here the field "Topic" always contains a part `<basetopic>`, this is the topic as configured in "3.5 Machine Configuration". Unless otherwise noted in description below, all data come as plain ASCII-texts.

Status Message

This message is sent whenever the operational state of the application changes:

Topic: `<basetopic>/state/`

Data: `IDLE` – software is started but marking dialogue is not open and no mark operation is running

`READY` – mark dialogue is open but no marking operation is active yet

`MARKING` – a marking process is active

PAUSED – a sliced 3D project was halted by pressing the "Pause"-button in mark dialogue; when this project is continued later and no error happens, it switches back to state "MARKING"

ERROR – an error occurred, no marking is possible at the moment; in case this state is used, an additional error-message is submitted

Error Message

This message is sent whenever the operational state of the application changes to or from "ERROR":

Topic: `<basetopic>/error/`

Data: an error text in case a previous status message signalled an error state or an empty string in case no error happened or in case a previous error was reset

Project Message

This message specifies which project is loaded at the moment:

Topic: `<basetopic>/project/`

Data: the full path to the project file which just was loaded. When it is empty, the "New" function was used and the last project was flushed completely.

Number-Of-Parts Message

This message returns information about the current part number which has been produced. When a value of 0 is sent, the parts-counter was reset by the operator:

Topic: `<basetopic>/parts/`

Data: the number of parts (as human readable ASCII-text)

Number-Of-Slice Message

This message is sent when a sliced 3D SLS/SLM model is built. Here the slices are announced as they are submitted to the scanner controller card. So this does not necessarily mean the current slice is finished, but it was handed over to the scanner card for production:

Topic: `<basetopic>/slices/`

Data: `<curr_slice>/<max_slices>` - the slice information (as human readable ASCII-text) where `<curr_slice>` is the slice currently processed/sent to the controller and where `<max_slices>` is the total number of slices to be processed in current project.

4 ScannerAdjust

This section describes ScannerAdjust, a tool to create and modify scanhead correction tables.

4.1 Overview

When a scanhead – a system with two mirrors that deflect a laser in X and Y direction – is used, the resulting movement on the working area can be distorted. Here a few major reasons for distortions can be found:

1. a pincushion distortion that is caused by the mirrors and the used optics

2. a 3D focal distortion that happens because the distance between last mirror and working area differs dependent on the position where the laser hits it – the distance is shorter in the middle directly below of the mirror and longer on the borders of the working area, resulting in a loss of focus for some of the positions

3. a spatial correction because the scanhead is not mounted in parallel to working area because it has to work inside of a concave working piece

To correct such distortions special correction tables are used. In most cases these correction tables are specific to and delivered together with a scanhead. They use a vendor-specific file format that has to be converted depending on the used software/the used scanner controller plug-in. In other cases no such correction file exist and has to be generated from scratch. As a third case for some very high accuracy applications existing correction files have to be modified so that they fit to the used hardware exactly.

All these three tasks can be done with ScannerAdjust correction file tool that is described here. The resulting correction table afterwards can be saved in RepliSLS3D HD (.bco) correction file format – a special format that provides much better accuracy than most other formats..

4.2 Usage

To start with generation of a new correction file either the menu-item "Project" → "Calculate new" or menu-item "Project" → "Interpolate new" has to be selected. Each of them opens a separate dialogue which provides a different method of generating a basic correction table.

Calculate new: Here a dialogue is openedwhere parameters for focal and pincushion distortion can be set. Additionally it can be specified which of both corrections have to be applied to the correction table. So for plain 2D applications or hardware set-ups where no third axis or no possibility to change the focus exists, only pincushion distortion parameters can be set. For a real 3D set-up focal correction parameters can be applied too. When none of the two options is set, a neutral correction file is generated. For a detailed description about how this dialogue can be used, please refer section "Error: Reference source not foundError: Reference source not found" below.

Interpolate new: When it is not intended to calculate a new correction based on some optical parameters, this option can be used. Here a dialogue opens, where the working area size and a grid of coordinates can be entered. This grid represents X and Y positions spread over the working area regularly. The grid is already filled with nominal coordinates that represent the positions to be marked. Now a similar Grid has to be marked without the use of any correction and then the real, actual marking positions of these points have to be entered into the related input fields. When clickong OK-button, a correction file is generated which bases on these actual coordinates and contains a field, that corrects these positions to reach the nominal ones.

Alternatively it is possible to load an already existing correction file via menu Project → Open or by using toolbar button "Load Project".

Next this correction table can be modified by specifying one or more nominal and actual positions. To get the values required for this, the current correction file has to be used for marking. During the marking operation one or more single points have to be marked. These points should be located at positions where the accuracy has to be increased or at positions spread over the whole working area. After marking the positions it has to be measured where the points have been marked really. The resulting values have to be entered in single or multiple point correction dialogue that can be opened via menu Edit → Correct Single Point or menu Edit → Correct Multiple Points. After entering the nominal coordinate(s) – the coordinates of the points within RepliSLS3D – and the actual coordinate(s) – the real coordinates as measured after marking – the current correction table is modified at the given position(s) in order to reflect the deviation(s) at this/these point(s). This operation has to be repeated with the same or varied point coordinates until the resulting correction table offers the required accuracy.

Please note: these correction operations influence the correction table only in X- and Y-direction. A possibly existing focal correction is damaged by these operations. Because of that for full 3D correction the focal correction has to be applied again after this: By selecting menu Edit → Recalc Focal Correction the related correction is re-calculated according to the changed positions of 2D correction coordinates.

Additionally the ScannerAdjust tool provides the same "Geometry"-panel like it is known from RepliSLS3D. So when the current correction table – that is shown in the middle of the application – is selected, here several operations can be performed. It can be scaled, mirrored, centred, rotated, the total position can be changed and other things more.

4.3 Correction Calculation Dialogue

When menu "Project" → "Calculate new" is selected, a dialogue opens, where focal and pincushion distortion parameters can be set. There exists an other dialogue that provides only focal correction parameters and that can be selected via menu Edit → Recalc Focal Correction. This dialogues parameters are a subset of the parameters of the correction definition dialogue and therefore are not described separately.

This dialogue provides following general parameters:

- Z-Distance – the shortest distance between last mirror and working area

The correction dialogue provides following parameters for pincushion distortion:

- Checkbox Pincushion Distortion – the 2D / pincushion distortion parameters are applied to the newly generated correction table only when this checkbox is set

- X-angle – the distortion strength/angle in X direction

- Y-angle – the distortion strength/angle in Y direction

This dialogue provides following parameters for focal correction:

- Checkbox Focal Correction – the depth / focal distortion parameters are applied to the newly generated correction table only when this checkbox is set

- Working Area Size – the size of the working area; independent from its real shape here always a squared size is assumed and the biggest used size has to be entered here

- Maximum Z-Range – the maximum range the Z-axis is able to move; this value is typically smaller than the horizontal working range of the scanhead in X and Y direction and therefore has to be specified separately

4.3.1 Angle Calculation Wizard

As the X-angle and the Y-angle values out of the correction calculation dialogue (as described above) are not easy to evaluate, there is a wizard available which helps to get a first rough guess, what values are appropriate for an actual distortion. To access this wizard, first click one of the buttons "X-angle" or "Y-angle" (the text right beside the related input fields for the angle values).

Now an other dialogue opens, where distances have to be entered, which can be measured directly within the previously marked but still distorted square:
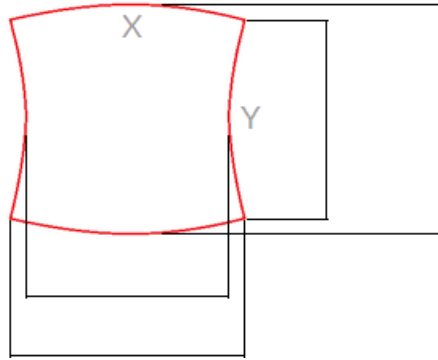
Here two values have to be entered for each of the directions X and Y: one for the distance of the corner points at the edges and one at the maximum/minimum expansion of the related edges. These values have to be entered into the related input fields. The measurement unit does not matter here, but needs to be exactly the same for all four fields.

After all four values have been entered, the wizard dialogue can be closed with "OK" and some suitable angles, calculated out of these measurements, are handed over to the parent correction calculation dialogue.

Please note: these angle values are quite good, but not yet perfect suitable. So they are a good starting point for generating a first correction, but they still need some fine-tuning afterwards (e.g. via function described in "4.7 Import Scanned Grid").

4.4   Correction Interpolation Dialogue

When menu "Project" → "Interpolate new" is selected, a dialogue opens, where actual marking positions can be entered that are set in relation to expected nominal positions. The order to generate a correction file with this method is as follows:

1. Generate a 2D RepliSLS3D-project which marks dots, crosses or other appropriate geometries at the coordinates shown in the interpolation dialogue

2. Mark this project without any correction involved and measure the exact positions of these geometries

3. Enter the used working area size within the interpolation-dialogue

4. Enter all the measured, acutal positions in the input fields that are preallocated with the nominal positions

5. Leave the Dialogue with "OK" to calculate a suitable correction

The entered coordinates are stored and available again when the application is started again in order to let the user continue working with already entered data. There are only two conditions under what the grid of actual coordinates is reset to its nominal defaults:

•   the working area size is changed to a different value

•   the button "Reset to nominal coordinates" is pressed

Please note: this method of generating a correction file is somewhat rough and therefore should be used only in cases where accuracy is not important or when additional steps are done in order to refine the resulting correction table. In all other cases it is highly recommended to use function "Calculate new" instead.

4.5 Correct Single and Multiple Points Dialogue

These two functions can be used to fine-tune an already existing correction table. So with them it is not possible to generate a complete correction from scratch but to do some fine-tuning on an existing correction file which is already close to be perfect. In other words: these functions are intended to be used in cases where an extreme accuracy is required which can't be provided by a correction file which is only calculated and does not base on real-world measurements at specific, critical points.

Both, the single and the multiple point dialogues expect the same values and behave the same except the following:

- the single point dialogue expects exactly one nominal and one actual coordinate value pair

- the multiple point dialogue asks for a CSV file that contains pairs of nominal and actual coordinate values and shows an editable list of them within the dialogue;
  The CSV-file contains comma-separated values in format
  `nominal x, nominal y, actual x, actual y,`
  using dot-separated floating point numbers

The dialogues offer the following fields and input values:

- Nominal Position X, Nominal Position Y – the nominal coordinate(s) in unit mm

- Actual Position X, Actual Position Y – the actual, measured coordinate(s) in unit mm

- Working Area Position X and Y – the top left position of the working area in unit mm

- Working Area Size – the size of the working area in unit mm, here a squared working area is assumed

- Influence scale – a factor which decides how much the correction at a single point has to influence the surrounding area in the correction table to be generated. Here a value of 1.0 means, the influence on surrounding area has to be as small as possible and will be exactly the difference between nominal and actual position. Every other value will be multiplied to the difference between these two points and expand the area around the nominal and actual point accordingly.
  When this scale factor is too small and the difference between nominal and actual position is too big, it may happen one correction grid point overlaps a previous/following correction grid point which would lead to undefined results and has to be avoided. In such a case the scale factor has to be increased.
  When a too big influence scale value is chosen in case of the multiple points dialogue, it may happen the influenced area is too big and overlaps a nearby nominal/actual position resulting in an over-compensation or invalid modification of this neighbour-point. In such a case the scale factor has to be decreased.

When the dialogue is left with "OK" the current correction table is modified according to the given nominal and actual value(s).

4.6 Spatial Correction Dialogue

This correction is intended for cases where the scanhead is not mounted in parallel to the working area but is inclined (e.g. in order to mark in inner side of a ring). It requires a correction that already fits to the used scanhead and optic, the spatial correction then can be applied to it as additional parameter. Thus following steps have to be performed in order to get an exact spatial correction:

- mount the scanhead in parallel to the working area in a way where the distance between scanhead and working area is the same length like the shortest distance between scanhead and working area when mounted inclined

- check if the correction for this scanhead fits and possibly modify it using the functions described above

- once you can mark an exact square that does not show any distortions, measure the size of that square and mount the head inclined

- mark the same square again; now that side of formerly square where distance between working area and scanhead is biggest has to be measured

The resulting two values – the normal (expected) length of the square measured in first step and the new, bigger length of the square measured after inclining the scanhead – have to be entered in dialogue for spatial correction. As an other information that side of the current correction table where the square has been distorted and where its edge is longer than before has to be marked using the radio buttons.

After pressing OK-button the spatial correction that stretched the squares width with growing distance will be applied to the correction table. Now the result is a rectangle. To correct its length to get back to a square, the gain factor of this direction has to be changed in scanner card settings (within RepliSLS3D) or the width/ height of the whole correction table has to be changed in Geometry-Panel of ScannerAdjust directly.

4.7   Import Scanned Grid

CorrCorrect provides the possibility to update and enhance an existing base-correction by using an image of an array of dots that have been marked with this base correction which may look like this:
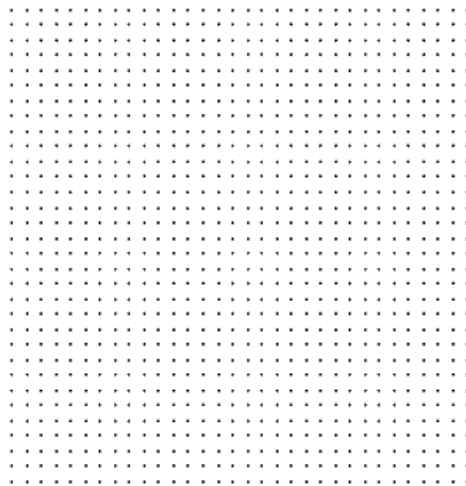
Correction enhancement based on such an image can be done with following simple steps:

- create a base-correction (as described in "Error: Reference source not foundError: Reference source not found") or use the correction file provided by the vendor of your scanhead

- mark a regular, squared raster of dots with this correction (e.g. on a sheet of white or black paper), here all neighboured dots need to have exactly the same distance in X- and Y-direction, the grid needs to have the same number of dots in X- and Y-direction and there have to be dots in each of the four outer corner positions of the used working area (a suitable example can be downloaded at https://halaser.systems/dotgrid.beamv)

- digitize the marking result (e.g. by using a flatbed scanner) to get an image file

- in CorrCorrect load the base-correction this grid of dots was marked with

- go to menu "Edit" → "Apply scanned grid" and import the image from previous step; here all common raster image formats like PNG, BMP, GIF and others are supported

Now CorrCorrect loads this image, analyses the positions of the dots and adjusts the current correction table in order to reflect the deviations which still exist. After saving the modified correction table these steps can be repeated with that new correction to iteratively improve the quality of the result.

During analysis of the image, CorrCorrect uses the actual positions retrieved out of the imported image and the related nominal positions (which are defined by the total number of dots and the fact that they originally describe squares) to check the differences and to calculate what a correction table has to look like in order to equalise these deviations. Here CorrCorrect is able to adjust the typical pillow- or galvo-distortion as well as

any other non-regular distortions. But it does not adjust offsets or a rotation of the scanhead along the Z-axis. These deviations later have to be adjusted either manually by adding them to the resulting correction table or by configuring the related offset- and rotation-parameters in settings of the used scanner controller card.

For the grid of dots following is important:

- the neighbouring dots need to form squares, so they all need to have the same distance in X- and Y direction

- the upper left/upper right/lower left/lower right dot need to be located at the outer corners of the working area

- after marking, no vertical dots are allowed to overlap some other column's vertical dots in X-direction and no horizontal dots are allowed to overlap some other row's horizontal dots in Y-direction; when this is the case in resulting image, the base-correction needs to be improved or the distance between the dots needs to be increased

- there should be no dirt on the surface or scanned image which can be misinterpreted as dot

When some of these conditions are not met, the grid-detection-mechanism may not work properly and may misalign some of the scanned dots positions. This will result in a damaged correction table.

The overall accuracy of the resulting correction table can be influenced by following parameters:

- laser spot size: the laser spot size and resulting from that the marked dots should be as small and as sharply delineated as possible

- contrast: in both, the marked material and the scanned image there should be an as good as possible contrast, here it would be best to have black dots on a white surface or white dots on a black background

- image quality: the scanned image should have an as high as possible resolution (large dpi-number) without any noise (noise can be misinterpreted as dots)

- rotation: the marked material should be aligned as exact as possible in both cases, when marked with the laser and when scanned in the flatbed scanner in order to not to have the dots containing a rotation which does not exist; the detection algorithm of CorrCorrect tries to automatically unrotate a grid of points, but this always results in additional inaccuracies due to rounding errors of the unrotation-function, so it is always a good idea to avoid such additional calculations in order to get a better accuracy.

- Image file format: do not use a file format which makes use of lossy compression algorithm (like in JPEG) as this changes the image, but something that has no or a lossless compression (like PNG, GIF or BMP)

When a scanned and loaded image is not accepted as grid and an error message appears, which tells, no squared grid was found, please check the following points:

- Does the used image contain transparency? In some images a transparency channel is used which – although it is transparent – has the same colour like the dots which makes it impossible to detect anything in the image. So please ensure there is no transparency in your image or make sure an image format is used which does not support transparency.

- Is there noise in the image, caused e.g. by some dirt on the marked sheet or by dust that appears on the scanned image? Such spots which are no regular marking dots can be misinterpreted, resulting in a grid with an illegal grid size. So please ensure a clean and accurate environment is used while creating such a scanned grid.

- What resolution is used? When scanning (aka digitising) the marked sheet, please use the best native resolution your flat-bed scanner offers in order to ensure the marked dots appear not only as single pixels in the image but as clear and large dots. This ensures they can be detected as wanted marking dots uniquely and are not filtered out as potential noise.

In case the grid detection failed, the related error message gives the possibility to show the detection result. When pressing "yes" to get this result, a dialogue opens where the original image (converted to greyscale) is shown. On top of that image the detected dots are drawn in green. Here one has to check:

- Are there some green markings in the image at positions where no dot was marked originally? This points to dirt on the surface or image which is misinterpreted as dot

- Are there dots in the original image which do not contain a green surrounding, means which have not been detected as regular dot? This points to the fact the resolution of the input image is too low so that the dots within the image are too small in order to be detected as suitable area.

4.8 ScannerAdjust Cookbook

Following some steps are described to get a suitable and accurate correction file that fits to a custom scan-head hardware set-up. Since every combination of scanhead with a specific lens results in an own, specific distortion, a separate correction file is needed for every scanhead-lens-combination. For the same combination typically the same correction file can be used.

By performing following steps a suitable correction file can be created:

1. Generate a project within RepliSLS3D that consists of a square which is nearly as large as the whole working area (edge length should be about 96% of the maximum working area size that is possible with a scanhead and can be 100% in a second step when an already created correction file has to be verified and refined).

2. Add a horizontal and a vertical line to this project, both should be exactly centred vertically and horizontally to the working area, crossing each other in exact centre.

3. Mark this project on a material where the marking lines can be seen as clear and sharp as possible.

4. Check the marking result, the edges of the square and the position of the horizontal and vertical lines (please refer to example in image below).
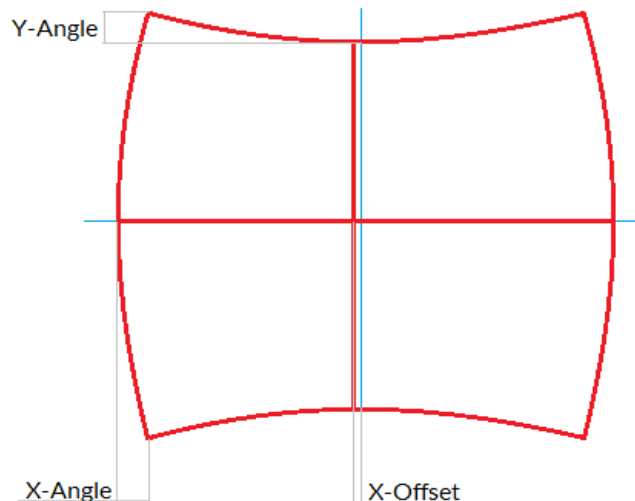
Now switch over to ScannerAdjust. When the square is distorted at top and bottom edge (distance marked in upper left corner in picture above), this distortion has to be corrected with the "Y-Angle" parameter in Correction Definition Dialogue. The stronger this deviation is, the larger the Y-Angle value should be.
When the square is distorted at left and right edge (distance marked in lower left corner in picture above), this distortion has to be corrected with the "X-Angle" parameter in Correction Definition Dialogue. The stronger this deviation is, the larger the X-Angle value should be.
To get some starting values for these two angles at least at the very first try, it is possible to use the angle calculation wizard as described in "4.3.1 Angle Calculation Wizard"

When the horizontal line is not marked at exact centre position of the working area (distance marked in lower middle position in picture above), after creation of a new correction file via Correction Definition Dialogue it has to be selected and shifted in vertical position. This can be done in tab-pane "Geometry" of ScannerAdjust by moving it against the direction of its deviation (depending on the strength of distortion this line may also be bent a bit, in this case the distance from centre point has to be used for shifting).

When the vertical line is not marked at exact centre position of the working area, after creation of a new correction file via Correction Definition Dialogue it has to be selected and shifted in horizontal position. This can be done in tab-pane "Geometry" by moving it against the direction of its deviation (this deviation is not shown in example picture above).

In some cases it also may be necessary to rotate the correction table (e.g. when the marking result and its distortion is rotated by 90 degrees comparing to the example picture above). The rotation angle of 90 degrees can be entered in tab-pane "Geometry" once. Smaller/other rotation angles that compensate mounting accuracies should not be set here but by using parameter "Rotation" in scanner card settings.

5. Save the generated correction file, configure RepliSLS3D to make use of it and mark the same geometry again. Depending on the result you may now continue with 4.) to refine the correction or finish here when the marked square does not show any distortions.

Some important things that have to be mentioned for the process of creation of correction tables:

- There is no direct relation between the pincushion distortion of an edge of the marked square and the angle value to be set to adjust it, means there is no simple formula to calculate the angle-value out of a measured distance in marking result. Thus it is necessary to try in several steps to find out the correct values. When a given angle does not fully remove a distortion, it has to be set to a larger

value in next step. When a used angle over-compensates a distortion and causes an edge to be bent in other direction, it has to be set to a smaller value.

- A correction table shown in ScannerAdjust always has a base-size of 65,535 x 65,535 mm. This is an artificial size value that has to be used by ScannerAdjust because this application does not know anything about used hardware set-up and real working area size. Thus a measured X- or Y-offset in real working area always has to be set in relation to this artificial working area size. The offset value to be entered within ScannerAdjust then has to be a value calculated out of the real offset and the relation between real working area size in RepliSLS3D and 65,535 mm artificial working area size in ScannerAdjust.

- After saving a new correction file, it is mandatory to let RepliSLS3D close a possibly existing connection to a scanner controller card. This can be done by going into project settings. RepliSLS3D loads a correction file only once on initialisation of scanner controller card, thus a new initialisation has to be forced this way. When this is not done, RepliSLS3D will continue working with a preceding correction table and will not notice there are any new data saved into the same file.

5 BeamControl

As described above, RepliSLS3D is a fully featured editor with loads of functions to create projects, manipulate geometries, modify parameters and even to control the whole process of building a 3D model. Although it contains a user management which gives the possibility to hide functions from users which are not privileged to do specific things, often it does not make sense to place this fully equipped software on a machine which does nothing more than marking already existing projects.

Here BeamControl comes into account (not to be confused with AndroidBeamControl which is an app for an Android-device and intended to watch running instances of RepliSLS3D or even BeamControl). This software is stripped down to the minimum functionality that is needed for loading a project and building it. It has a plain and clear user interface offering only these functions which are really needed, comes with big buttons and input fields that allow usage with a touch-screen and even provides a user management to not to let users start potential dangerous operations which are not privileged to do that.

BeamControl comes as separate executable which is located in RepliSLS3D installation directory (Windows) or in standard search path $PATH (Linux). It is up to the machine integrator to start this application automatically and in foreground.

Index