



MANUAL
INSTALLATION AND CONFIGURATION
OF "CUA32" CONTROL UNITS

NEWSON NV

Table of Contents

1 THE CUA32 SYSTEM 4

 1.1 CUA32-MST 4

 1.2 CUA32-SLV 4

2 INSTALLATION 5

 2.1 HOW TO 5

 2.2 SETUP COMMUNICATION BETWEEN HOST AND CUA32-MST 6

 2.2.1 AUTOMATIC “CUA-USB-3G” DRIVER INSTALLATION ON WINDOWS8 / WINDOWS10..... 6

 2.2.2 MANUAL INSTALLATION OF “CUA-USB-3G” DRIVER ON WINDOWS7/ XP 6

 2.2.3 ETHERNET COMMUNICATION WITH “CUA32-MST-XX” DEVICE 6

 2.3 STARTING RHOTHOR.EXE 7

 2.3.1 CONNECTION STRING USB INTERFACE 7

 2.3.2 CONNECTION STRING ETHERNET INTERFACE 7

 2.4 TARGETS 8

 2.4.1 CUA32 TARGET BUS 8

 2.4.2 TARGET MAPPING 9

 2.4.3 TARGET SELECTION 9

3 CONFIGURATION..... 10

 3.1 FRONT END 10

 3.1.1 USB ID 10

 3.1.2 IP..... 10

 3.2 TARGETS 11

 3.2.1 FIELD SIZE 11

 3.2.2 TRACK DELAY 11

 3.2.3 SET POINT FILTER 11

 3.2.4 IN1..IN4, DEDICATED INPUTS..... 12

 3.2.5 IO5..IO8, 5V ANALOG/DIGITAL INPUT/OUTPUTS 13

 3.2.6 IO9..IO16, VIO POWERED INPUT/OUTPUTS 13

 3.2.7 IO17/O17, LASER INTERFACE..... 15

 3.2.8 CH1..CH3, CONNECTION TO DEFLECTORS (DEFLECTION SYSTEM)..... 17

| | | |
|----------|------------------------------|-----------|
| 4 | RUNNING..... | 18 |
| 4.1 | STATUS | 18 |
| 4.2 | SCRIPTING..... | 19 |
| 4.2.1 | HELLO WORLD | 19 |
| 4.2.2 | CREATING A SCRIPT | 19 |
| 4.2.3 | CREATING A FLASH JOB | 20 |
| 4.2.4 | CREATING A BOOT SCRIPT | 20 |
| 4.3 | DIRECTORY | 21 |
| 4.3.1 | DELETE /FORMAT | 21 |
| 4.3.2 | LOAD /SAVE FILE | 21 |
| 4.4 | TARGET CALIBRATION | 21 |
| 4.4.1 | RESET | 22 |
| 4.4.2 | EDIT CALIBRATION..... | 22 |
| 4.4.3 | ADD | 23 |
| 4.4.4 | LOAD /SAVE FILE..... | 23 |
| 4.5 | LASER LINK..... | 23 |
| 5 | DEFLECTORS | 25 |
| 6 | APPENDIX | 27 |
| 6.1 | DIMENSIONS | 27 |

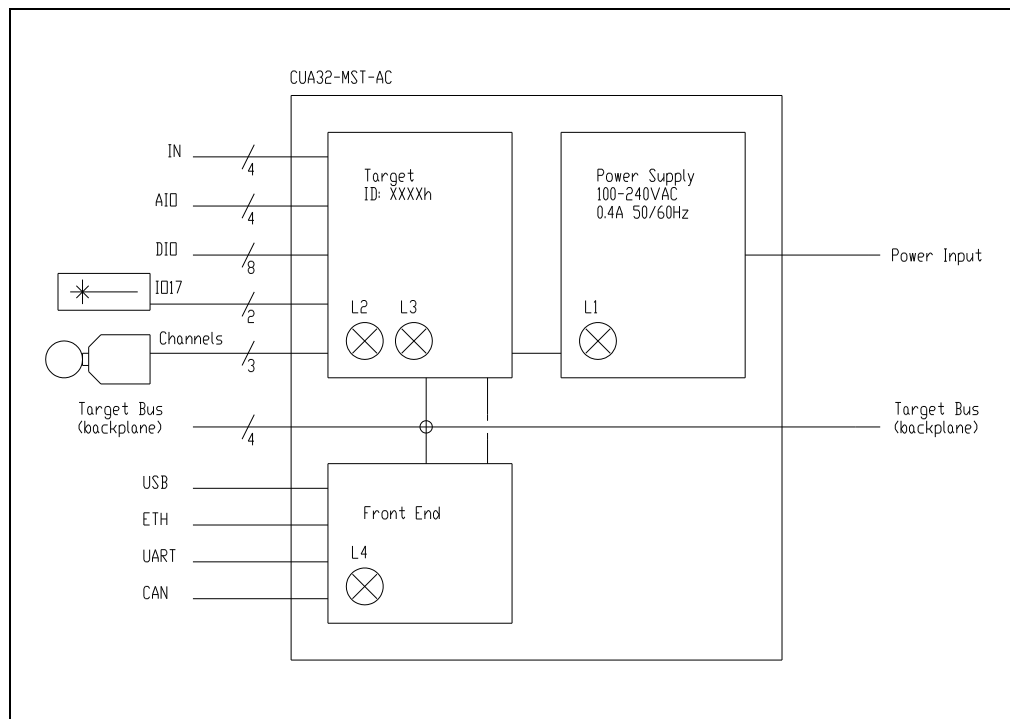
1 THE CUA32 SYSTEM

1.1 CUA32-MST

The key stone of the beam steering system is the CUA32-MST (master) device. It communicates with a host computer, controls a deflection system, triggers a laser and handles several IO events. Supporting streaming and stand alone operation, as a DIN rail compatible system it is easy to integrate in any machine design. The CUA32 master device comprises two functional controllers and a power supply mounted on a connector board.

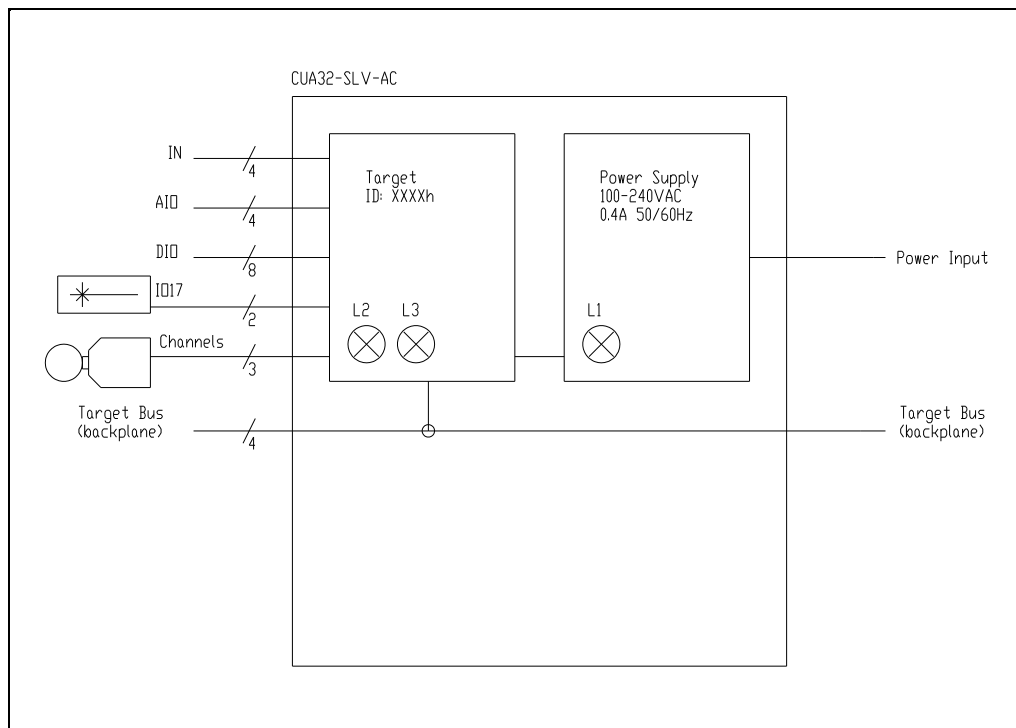
- The Front End controller (CUA32-FE) provides connectivity with a host computer and has a flash disk to store configuration and image files.
- The Target controller (CUA32-TGT) controls a deflection system, laser and several IO signals. Over these IO's it's also possible to control up to three stepper motor drivers making it a 6-axis controller.
- The power supply provides direct connection of the device with mains (CUA32-MST-AC) or with a 24/48 volt DC power line (CUA32-MST-DC).

The Front End controller is connected to the Target controller using a serial bus. This Target bus is also accessible over a back side connector for expansion. Up to 7 additional Target controllers, fitted in slave devices, can be connected and controlled by the master. The bus operates like a multi drop broadcast system. Before being able to access the attached Targets, they need to be mapped to an index. A software tool rthor™ is provided to configure the complete system. Before this tool can be used, communication between host computer and master device need to be established.



1.2 CUA32-SLV

Besides having only the Target controller, a CUA32-SLV (slave) device is similar as a master. Because there is no Front End, a slave device has no flash memory and can not be connected directly to a host computer. Configuration and host connectivity is provided over the external backplane by the connected master device. Throughout this document the names slave device and Target controller are interchanged.



2 INSTALLATION

2.1 HOW TO

- Step 1: mount the CUA32 devices. A CUA32 device is designed to be clicked on a DIN rail. When slave devices are needed, the DIN-rail is also used to house the backplane (Target Bus). When no DIN rail is available the CUA32 devices can be mounted using bolts. Mounting holes are provided by shifting the 2 orange sliders at the backside outwards.
- Step 2: connect the deflectors. RTA-3G Deflectors are connected to the CUA32 devices over SMA coaxial cables. The cables must have 50 ohm impedance and their length is limited to 5 meter.
- Step 3: verify grounding. Only the USB, Ethernet and IO17 pins of the CUA32 device are galvanic isolated from the internal electronics. The CUA32 device is a double isolated device and therefore not connected to ground. However, electronics inside the deflector are connected to the housing over a 100 ohm resistor. This provides shielding for the sensitive electronics. When your setup suffers from ground noise additional connection to ground could be needed. Any 0V signal of the CUA32 devices can be used for this purpose.
- Step 4: connect the laser and additional IO signals
- Step 5: connect and switch on AC power
- Install the rthor software and connect the CUA32-MST device with the computer
- Setup the system ("Config" tab in rthor™)
- Switch on laser and mark Calibration figure or full field Square (Hello World)

2.2 SETUP COMMUNICATION BETWEEN HOST AND CUA32-MST

2.2.1 Automatic “CUA-USB-3G” Driver Installation on Windows8 / Windows10

CUA-USB controllers use a standard windows device driver (Winusb.sys) and can be installed automatically on Windows 8 and later Windows OS versions. When you connect a CUA-USB device, the system reads device information and loads Winusb.sys automatically.

2.2.2 Manual installation of “CUA-USB-3G” driver on Windows7/ XP

For older operating systems, WinUSB installation is part of a driver package which can be downloaded from Newson’s website. This driver package is an unsigned driver installation package. This means you might need to disable driver signing enforcement when installing the CUA32-MST-XX controller on windows x64 versions.

For windows 7

1. Connect the CUA32-MST-XX controller to the computer with an USB cable.
2. Go to start and right-click Computers and select **Manage**. This will bring the Computer Management Window
3. Open Device Manager, right-click the new device and click **Update Driver Software**
4. You have two options here, select **Browse my computer for driver software**
5. Click **Browse** and point to the rthor.inf file of the driver folder (part of the driver package), and then click **Open**
6. Click **Install this driver software anyway**, click **Close**

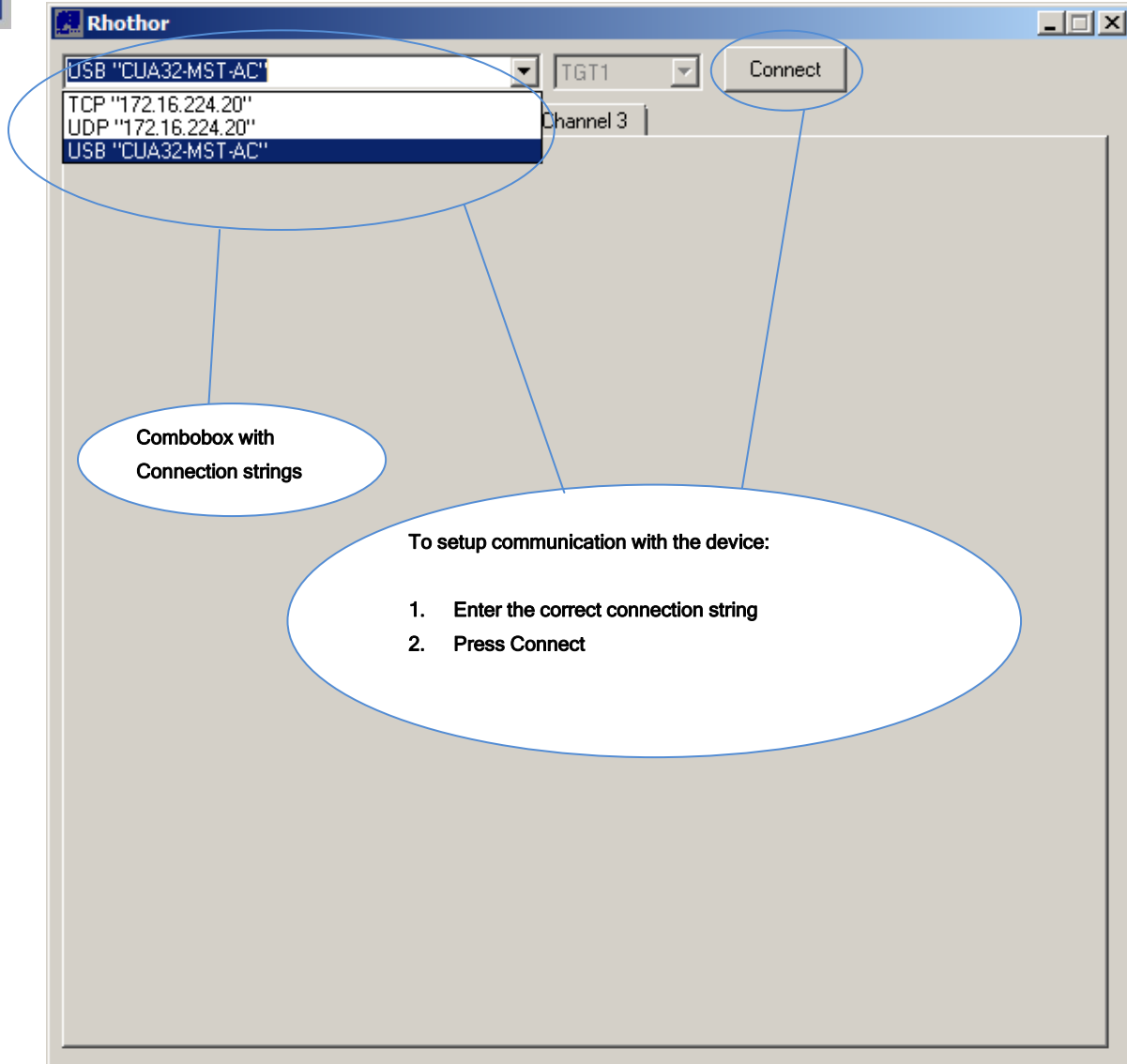
For Windows XP

1. On the **Welcome to the Found New Hardware Wizard** screen verify that **Install from a list or specific location (Advanced)** is selected, and then click **Next**.
2. When the **Please choose your search and installation options** appears, select **Search for the best driver** in these locations. Verify that the item **Include this location in the search** is checked, click **Browse** and point to the rthor.inf file of the driver folder (part of the driver package), and then click **Next**
3. When the **Windows Logo testing** dialog appears, click **Continue Anyway**.
4. On the **Completing the Found New Hardware Wizard** screen, click **Finish**.

2.2.3 Ethernet communication with “CUA32-MST-XX” device

The host can communicate with the CUA32-MST device over an Ethernet connection. Prior, the static IP needs to be set using the USB connection. In order to address the control unit successfully, the IP addresses of the PC must be on the same Local Area Network. Therefore you need to assign a valid static IP address to the PC. For more information on configuring a static IP address in Windows, see the Windows help files. You can access Windows help files by clicking the Windows Start button and selecting Help. Search in Windows help for static IP address.

2.3 STARTING RHOTHOR.EXE



2.3.1 Connection string USB interface

When pulling down the combo box the software will enumerate all valid USB devices connected to the PC. The connection string of each CUA32-MST-XX device connected with USB will have format *USB "name_of_device"*. The name of the device can be modified using the rhothor™ software. You can also use default connection string *USB*. This will connect to the first available CUA32-MST-XX device on the USB bus.

2.3.2 Connection string Ethernet interface

For Ethernet devices you have to enter the valid connection string inside combo box.

The format should be *TCP "ip_of_device"*. As default connection string *TCP "172.16.224.20"* is available inside the combo box. This will setup a TCP connection with the CUA32 device.

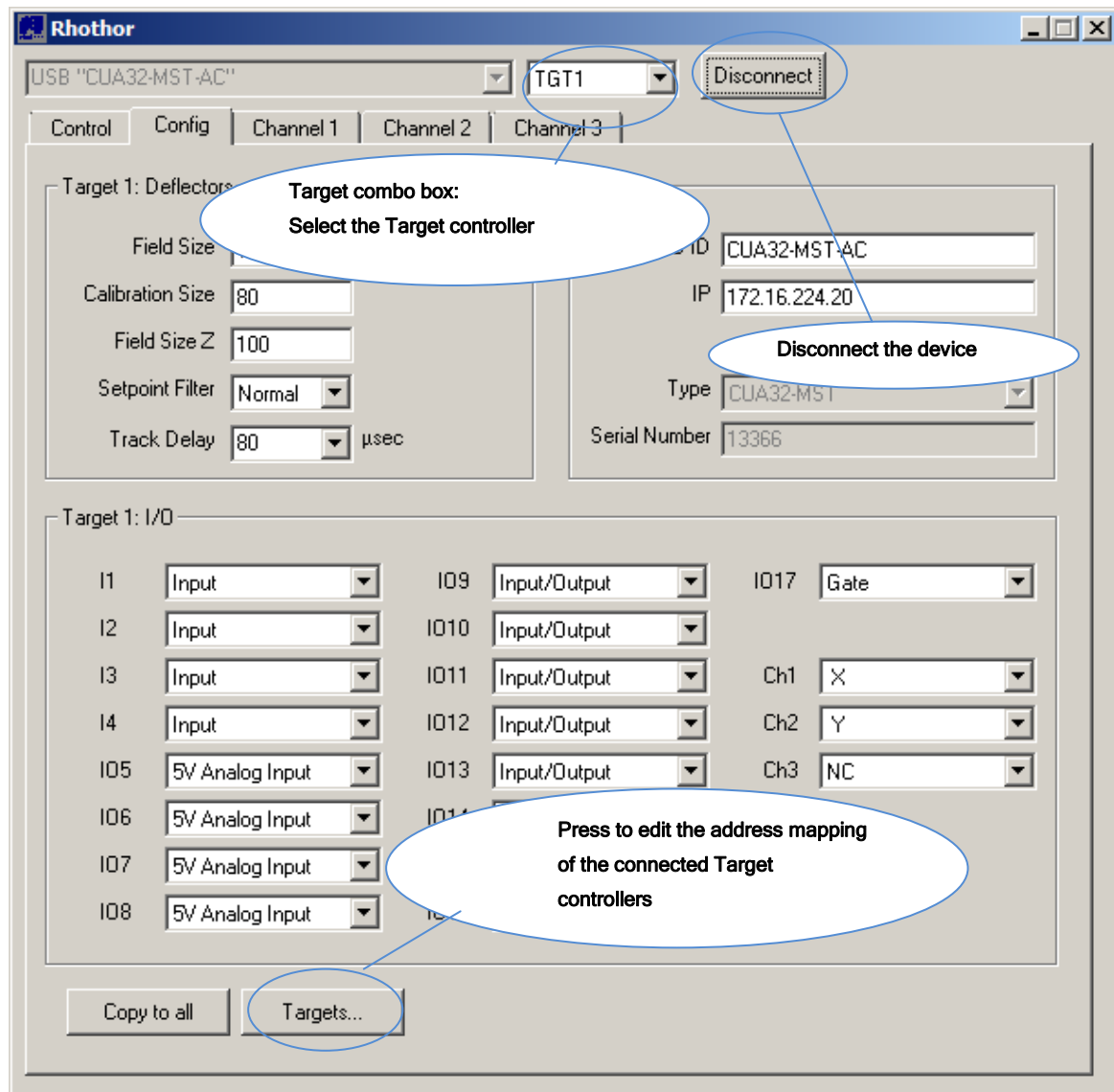
It is also possible to use an UDP connection. The format is then *UDP "ip_of_device"*.

As default connection string *UDP "172.16.224.20"* can be selected from the combo box. With UDP data transmission is not guaranteed. There might get some data lost.

2.4 TARGETS

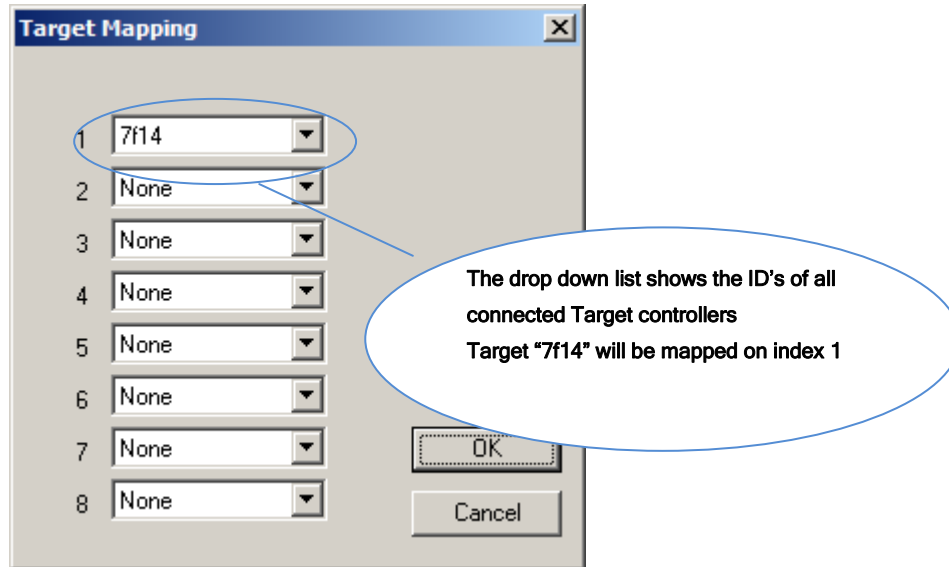
2.4.1 CUA32 target bus

The Front End controller inside CUA32-MST-XX devices sends commands to the connected Target controllers over the Target bus. The first byte of every command is a target byte. To select the command's target a bit masking access scheme is used. Bit 0 of the command byte corresponds with Target 1, bit 1 with Target 2 and so on. One Target controller is mounted inside the master device. So up to 7 additional Target controllers mounted in CUA32-SLV-XX (slave) devices can be added. To operate, all connected Target controllers need to be mapped to an index.



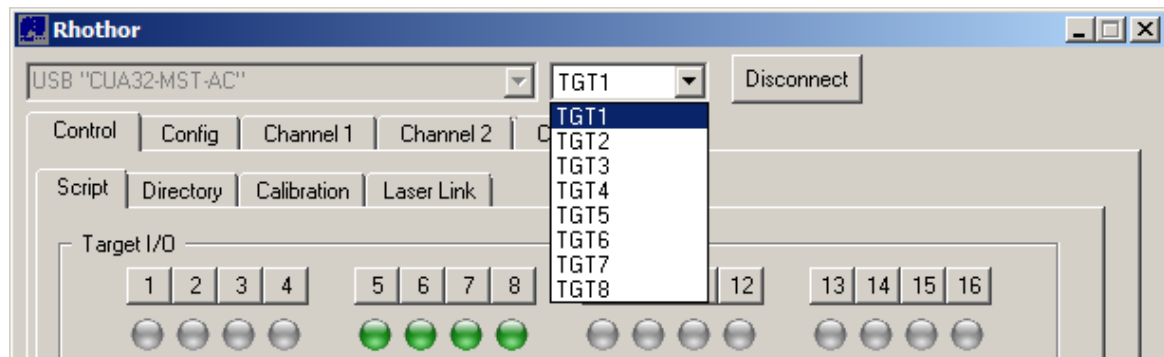
2.4.2 Target mapping

Mapping the connected (Target) controllers with index values, starts by clicking the "Targets..." button to popup the Target Mapping dialog. Every target index (1...8) has a drop down list showing the ID of all controllers attached to the Target bus. A controller is assigned to an index by selecting its ID. When a drop down list has the focus, LED2 of the selected controller will light up. A controller may not be assigned to multiple indexes. The dialog refuses to close when tried.

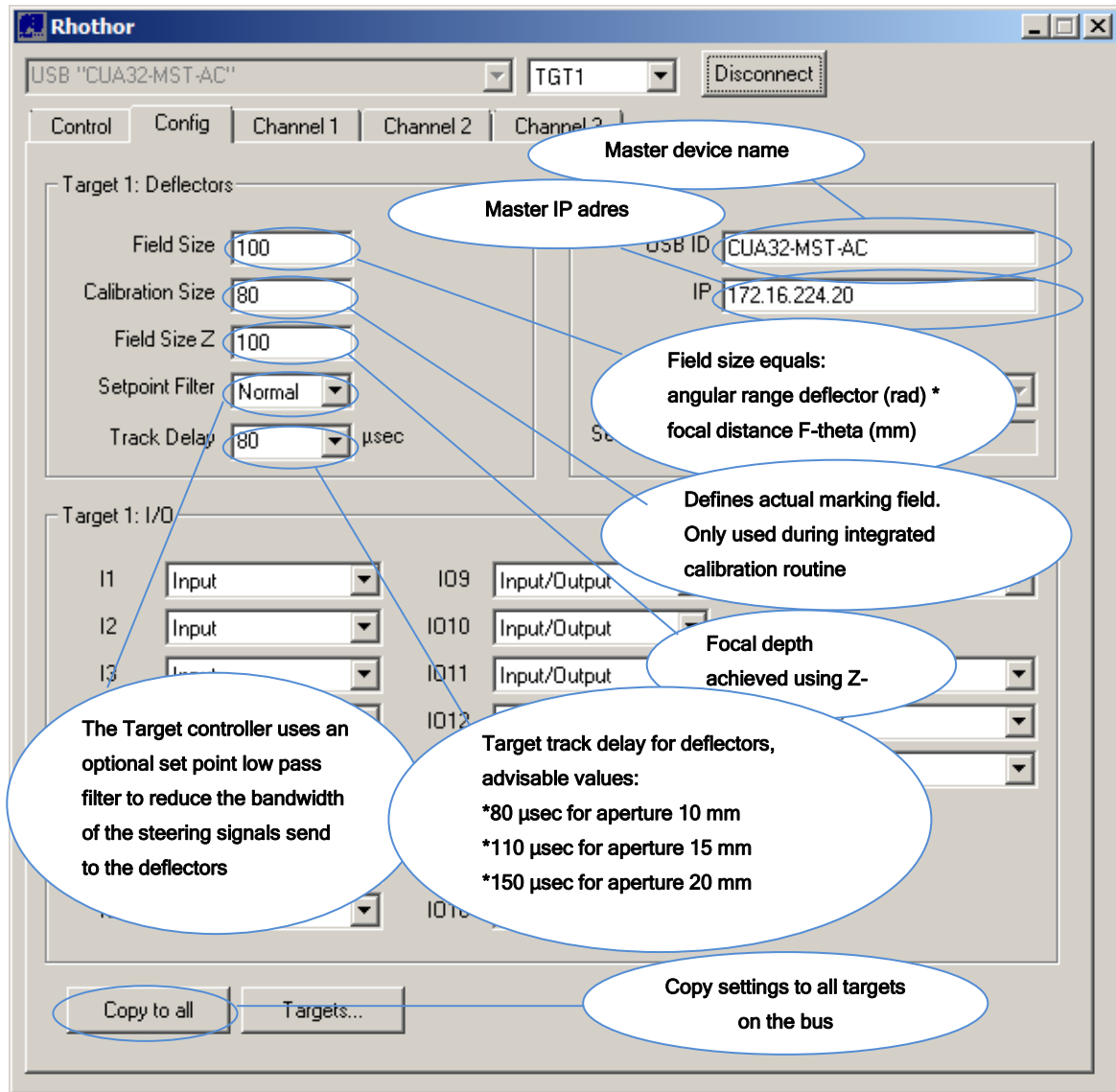


2.4.3 Target selection

Once the CUA32 devices are mapped, their target index can be used to select them. The drop down list box provides an easy way to change the target index.



3 CONFIGURATION



3.1 FRONT END

3.1.1 USB ID

It is possible entering a master device name in the USB ID-Info text field. Using different device names allows addressing when multiple CUA32-MST devices are connected to the same computer over USB.

3.1.2 IP

The host can also interface with a CUA32 master device over an Ethernet connection. However before IP addressing can be used, the master IP address should be set over a USB connection. Once the IP is set, disconnect the device and you can connect now over the Ethernet connection. Make sure your PC is configured with a correct static IP address.

3.2 TARGETS

3.2.1 Field size

Field size is the size of the marking area in mm. This is the area that the system can cover. The range depends on the angular range of the installed rthor™ deflectors and the focal length of the mounted optics.

The value to be entered can be calculated:

$$\text{FIELD SIZE} = \text{FOCAL_LENGTH} * \text{ANGULAR_RANGE} / 1000$$

"focal_length" is the focal length of the used lens.

"angular_range" is the optical scanning range of the installed motors in milliradians.

Example:

If an f-theta 163 mm flat field lens is used on a system fitted with RTA-AR800 motors the field size that has to be entered equals:

$$\begin{aligned} \text{field size} &= \text{focal_length} * \text{angular_range} / 1000 \\ &= 163 * 800 / 1000 = 130 \text{ mm} \end{aligned}$$

Trial and iteration is another way to determine the value that has to be entered. Mark a square and measure its size. The relation between measured size and required size of the square can be used to change the field size setting. The square needs to be positioned at center of the marking area and must be smaller than the field size.

3.2.2 Track Delay

A rthor™ deflection system uses digital regulator technology to control the motors in a closed loop fashion. Track delay is the time delay between set point value and actual value of a closed loop system. In an ideal system this track delay is zero; all steering commands are executed immediately. However zero track delay is not possible. The deflectors need time to accelerate. Track delay allows those motors to execute the steering commands. The track delay value defines the dynamics of the connected deflector. Choosing a low value will increase the deflector's bandwidth resulting in reduced rounding errors. When the track delay is changed, the connected deflectors need to be retuned.

| Aperture (mm) | Recommended Track delay (µsec) |
|---------------|--------------------------------|
| 10 | 80 |
| 15 | 110 |
| 20 | 150 |

Settings values lower than those limits will not damage the system, but give it a tendency to go into oscillation. Furthermore there is a relation between track delay and desired maximum marking speed. If a high maximum marking speed value is needed, the track delay should be increased giving more time to the motors to accelerate.

3.2.3 Set point filter

When there are large changes in the set point stream send to the scanner, it could overshoot. This is the result of the PID controller reacting to the large set point changes. By using a set point filter the bandwidth of the steering signal can be fitted to that of the connected deflector. The set point filter also reduces the mechanical tension forces in both deflector and mirror. To fit all applications and mirror types, the Target controller has a build in configurable set point filter.

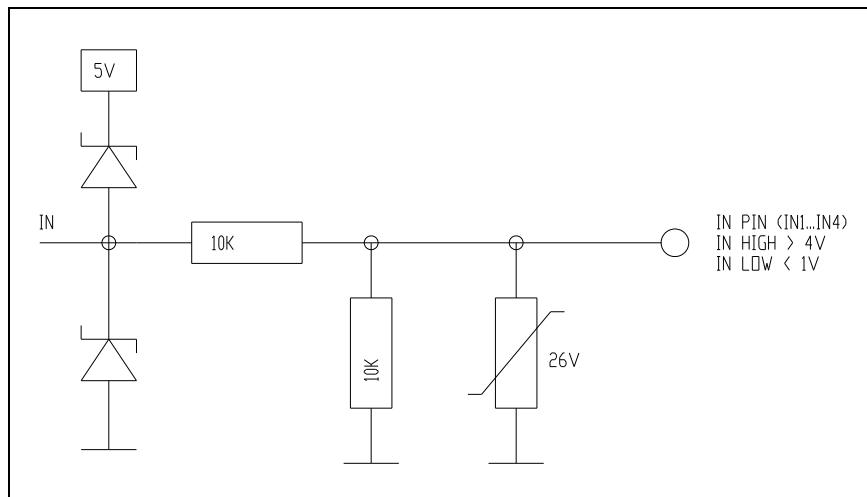
The filter has 4 settings: "None", "Sharp", "Normal" and "Smooth". These settings define the time constant of set point low pass filter. The default value is "Normal".

| Filter | Filter Time Constant |
|--------|----------------------|
| None | - |
| Sharp | 0.5*Track delay |
| Normal | Track delay |
| Smooth | 1.5*Track delay |

Note: When selecting a set point filter, an extra time delay will be added between the steering set point and the actual track (the filter time constant). This delay can be compensated in applications by modifying the laser timings using rhothor™ library function “rtSetLaserTimes”.

3.2.4 IN1..IN4, dedicated inputs

IN1 to IN4 are high impedance input-only connections. They provide on-the-fly functions and can be paired with IO9..IO12 forming differential inputs.



CONFIGURE FOR SINGLE-ENDED CONNECTION TO ENCODERS (ON-THE-FLY)

- a) On-the-fly X, provide phase A and phase B signals on I1 and I2

I1

I2

- b) On-the-fly Y, provide phase A and phase B signals on I3 and I4

I3

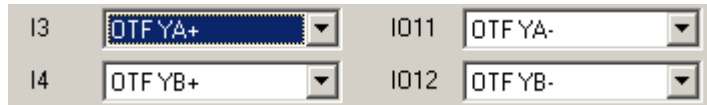
I4

CONFIGURE FOR DIFFERENTIAL CONNECTION TO ENCODERS (ON-THE-FLY)

a) On the fly X, provide differential phase A and phase B signals on I1/IO9 and I2/IO10

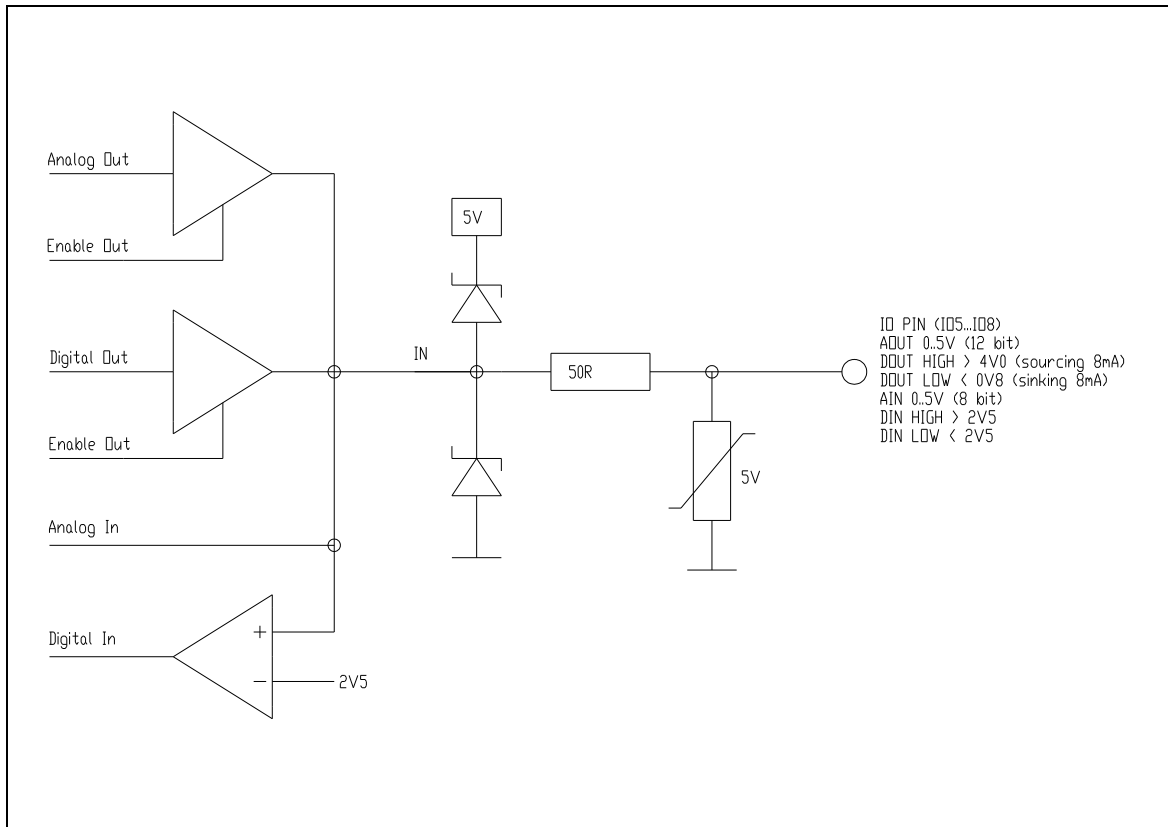


b) On the fly Y, provide differential phase A and phase B signals on I3/IO11 and I4/IO12



3.2.5 IO5..IO8, 5V analog/digital input/outputs

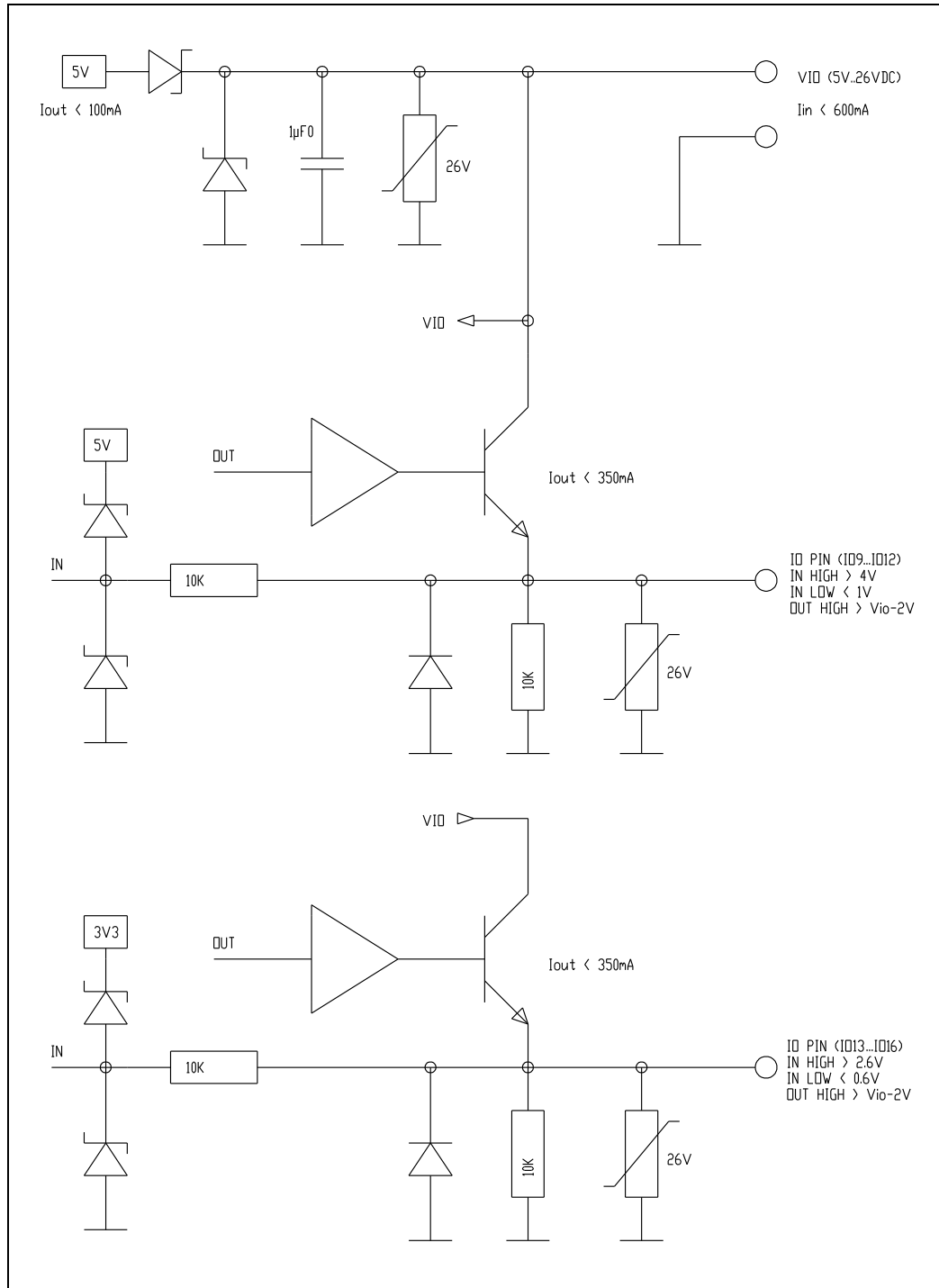
IO5 to IO8 are 5V level interface signals. They have analog and digital functionality.



3.2.6 IO9..IO16, Vio powered input/outputs

The output drivers of IO9 to IO16 are able to provide higher output currents and voltages when fed by an external power supply. This power supply (5 to 24V) should be connected to the VIO pin. The combined output current of all output drivers at the same time should be lower than 600 mA while a single output can source 350mA. The output drivers are protected against thermal overload and short circuit. The outputs are also fitted with a fly by diode to allow connection of inductive loads. When an external power supply is not connected, the drivers are fed by the internal 5V system power supply. In this

configuration the combined output current should be kept below 100 mA. Alternate output functions comprise pulse and direction signals to control stepper motor drivers.

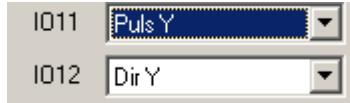


CONFIGURE FOR EXTERNAL AXIS CONTROL WITH PULSE/DIRECTION INTERFACE

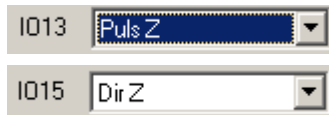
a) X-axis: connect output IO9 to external X axis Pulse input, connect output IO10 to external X axis Direction input



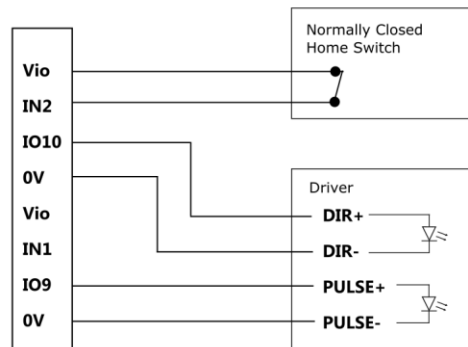
b) Y-axis: connect output IO11 to external Y axis Pulse input, connect output IO12 to external Y axis Direction input



c) Z-axis: connect output IO13 to external Z axis Pulse input, connect output IO15 to external Z axis Direction input



Other Inputs can be used as limit switch, a wiring example for the x-axis:



INFINITE FIELD

In combination with Pulse/Direction signals, single-ended connection with encoders allows infinite field operation. In this mode the CUA32 controller decides which part of the setpoint stream can be handled by the table sending the remainder to the deflection system. On the fly inputs from the encoder ensures accuracy.

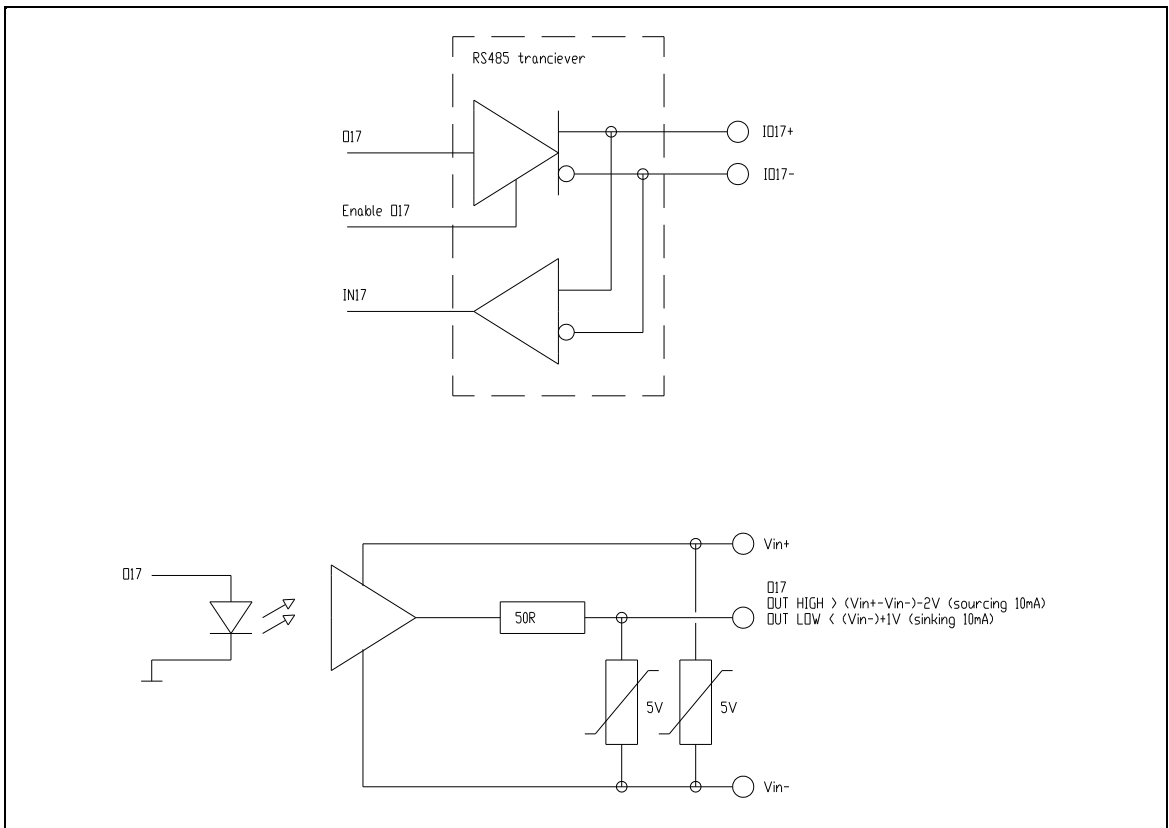
3.2.7 IO17/O17, laser interface

IO17 is a dedicated laser interface pin which provides following operational modes:

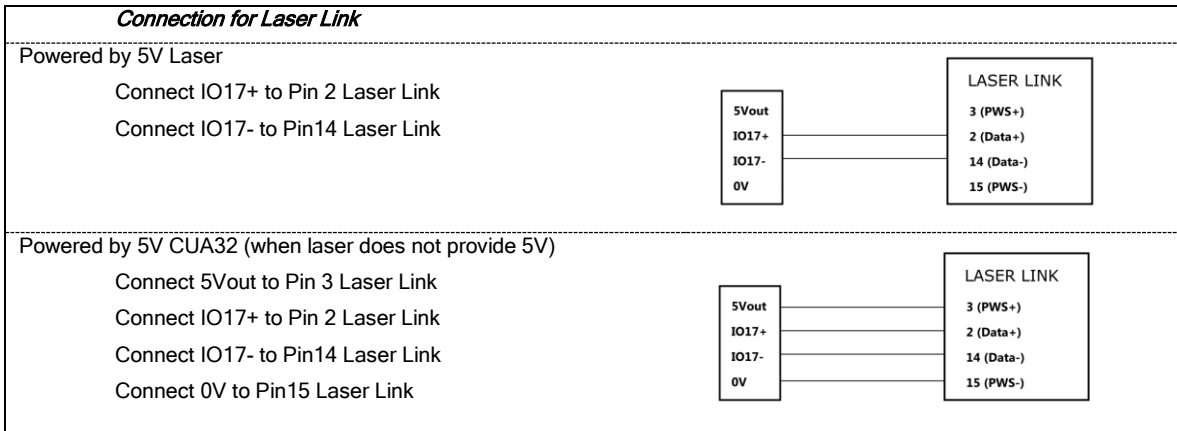
- Laserlink: controlling laser over LLink adapter
- Gate: Laser Gate signal
- Burst: Laser trigger signal
- CO2: Switch between two duty cycles (wakeup pulse)
- Burst FM: Laser trigger signal, frequency modulated in function of speed

A Target controller has build in 3 oscillators to generate the laser signals. In CO2 mode the active and idle block waves are defined by the properties of oscillator1 and 2. In mode Burst/Burst FM the laser trigger (pulse train) is defined by the properties of the third oscillator. Furthermore, in mode Burst FM this frequency is setpoint speed modulated. The frequency changes automatically with the deflector speed keeping the inder dot distance constant.

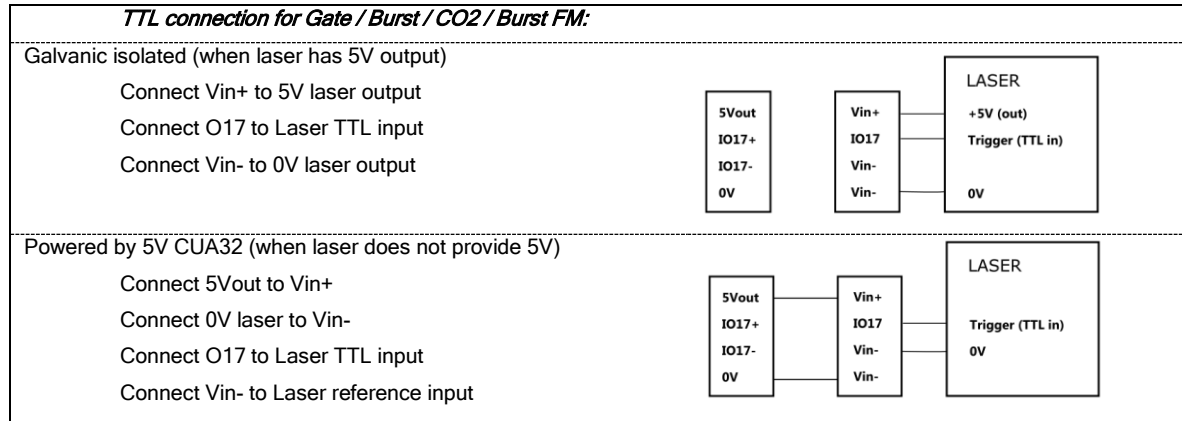
A (TTL compatible) galvanic isolated connection to the laser can be obtained over O17 by powering its driver using an external 5V power supply.



IO17 LASERLINK



IO17/O17 GATE/BURST/CO2/BURST FM



3.2.8 CH1..CH3, connection to deflectors (deflection system)

CUA32 devices can control RTA-3G deflectors and third party deflection systems. RTA-3G deflectors can be connected directly to a CUA32 device. Setpoint and actual positions are exchanged at 20 bit resolution. The build in power supply is able to power up to 3 RTA-3G deflectors. RTBE-XXX devices provide connectivity to third party XY2-100 and analog based deflector systems.

| Ch1 | Ch2 | Ch3 |
|----------------------------------|----------------------------------|----------------------------------|
| NC (no motor connected) | NC (no motor connected) | NC (no motor connected) |
| X (motor X connected) | Y (motor Y connected) | Z (motor Z connected) |
| -X (motor X connected, mirrored) | -Y (motor Y connected, mirrored) | -Z (motor Z connected, mirrored) |
| RTBE (RTBE adapter connected) | | |

4 RUNNING

The control page provides a ready to use platform to run command scripts and verify the overall status.

4.1 STATUS

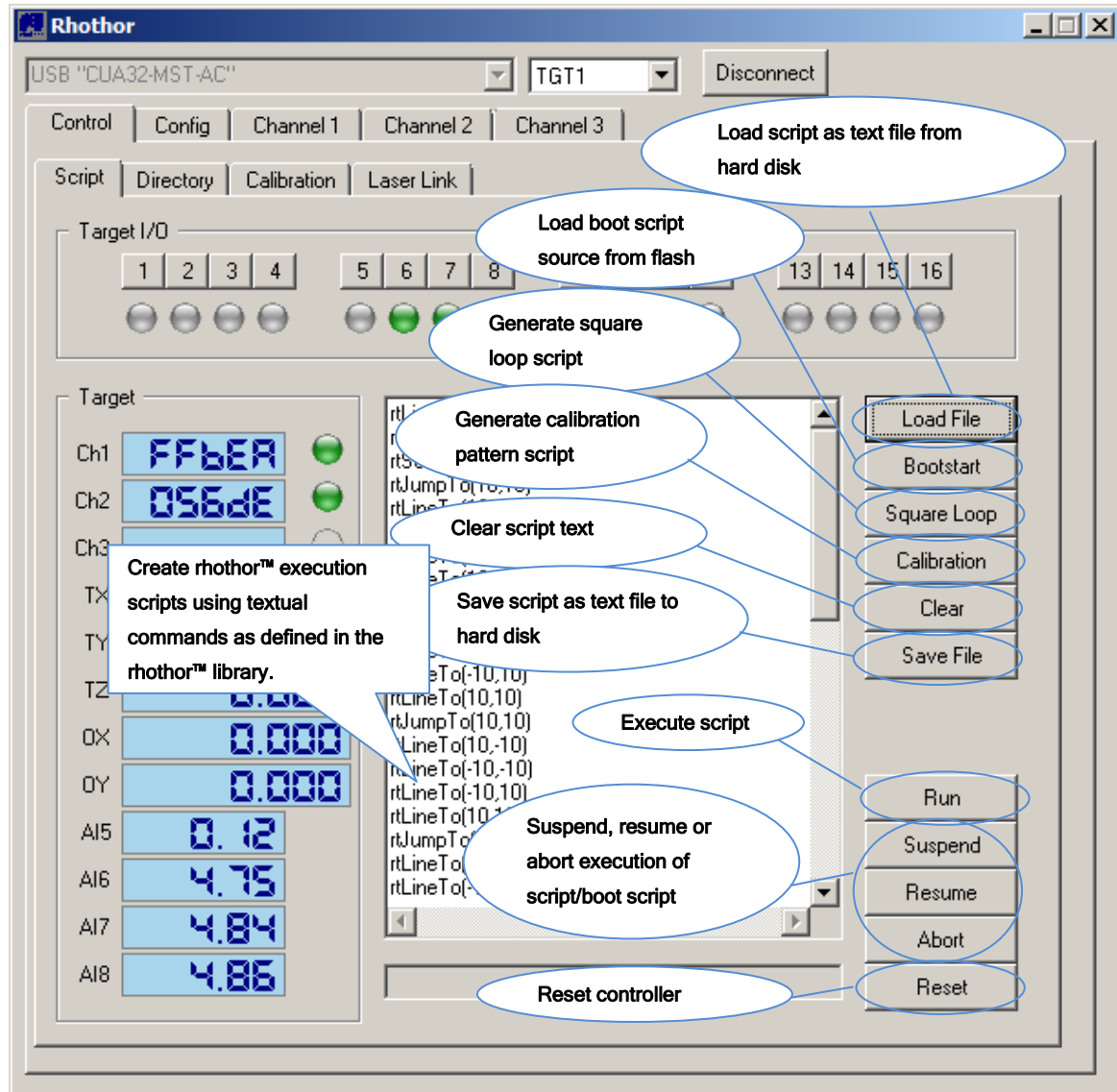
The screenshot shows the Rhothor control software interface. The window title is "Rhothor". At the top, there is a dropdown menu showing "USB 'CUA32-MST-ASH'" and a "Disconnect" button. Below this, there is a "Channel 3" label and a "Change output state of the I/O" button. The main interface is divided into several sections:

- Input/Output status:** A legend indicates that gray represents "Input/Output is Low" and green represents "Input/Output is High". Below this, there are two rows of 16 buttons each, labeled "Target I/O". The first row (1-8) has green buttons, and the second row (9-16) has gray buttons.
- Current scanner position (hex):** A section with three channels (Ch1, Ch2, Ch3) and their corresponding hex values: Ch1: FF6F7, Ch2: 056F8, Ch3: (empty). Each channel has a green LED indicator.
- Current table position (mm):** A section with six channels (TX, TY, TZ, BX, BY) and their corresponding values: TX: 0.000, TY: 0.000, TZ: 0.000, BX: 0.000, BY: 0.000.
- Current encoder feedback input (mm):** A section with four channels (AI5, AI6, AI7, AI8) and their corresponding values: AI5: 4.86, AI6: 4.86, AI7: 4.88, AI8: 4.86.
- Connected deflector:** A legend indicates that no led means "no deflector connected", green means "deflector running no overload", red means "deflector running overload", and gray means "deflector switched off".

Buttons at the bottom right include "Save File", "Resume", and "Abort".

4.2 SCRIPTING

The script text editor is an easy and fast way of controlling the deflection system without the need for extra application software. A script is build up by using the list commands provided by the rthor™ dynamic library interface.



4.2.1 Hello World

Pressing the button "Calibration" loads the editor with a series of commands. When ran, the deflection system will mark the calibration figure. This and the "Square Loop" button can be used as a Hello World marking job.

4.2.2 Creating a script

A rthor™ script starts with an "rtListOpen" command and ends with an "rtListClose" command. When you click the "Run" button the script in the text editor is interpreted and executed. During execution the "Run" button will be disabled. A running job can be stopped at all time by pressing the "Abort" button. The set of commands used in the script editor are the same as the rthor™ software library and are defined in the manual "Applications for CUA-USB and CUA ETH control units".

Script example: marking a square with compensation delays

```

1.  rtListOpen(1)           // start rhothor™ command list
2.  rtSetIO(1,1)           // set job-in-progress IO1
3.  rtSetJumpSpeed(1000)   // set jumping speed
4.  rtSetSpeed(1000)       // set marking speed
5.  rtSetLaserTimes(80,80) // implement gate shift
6.  rtJumpTo(-10,-10)      // jump to position
7.  rtSleep(80)            // implement jump delay
8.  rtLineTo(10,-10)       // mark a line
9.  rtBurst(80)            // implement polygon delay
10. rtLineTo(10,10)        // mark a line
11. rtBurst(80)            // implement polygon delay
12. rtLineTo(-10,10)       // mark a line
13. rtBurst(80)            // implement polygon delay
14. rtLineTo(-10,-10)     // mark a line
15. rtSleep(80)            // implement mark delay
15. rtSetIO(0,1)           // clear job-in-progress IO1
16. rtListClose()         // end rhothor command list

```

4.2.3 Creating a flash job

CUA32 control card has a flash memory which allows storing different marking jobs on the flash. The flash comprises 250 sectors of 256 Kbytes. Flash jobs can be created by starting the script with an “rtFileOpen” command and end it with an “rtFileClose” command.

Script example: creating a job to store in flash

```

1.  rtFileOpen(FlashJob1) // create rhothor™ flash job “FlashJob1”
2.  rtSetJumpSpeed(1000)  // set jumping speed
3.  rtSetSpeed(1000)      // set marking speed
4.  rtSetLaserTimes(80,80) // implement gate shift
5.  rtJumpTo(0,0)         // jump to position
6.  rtLineTo(10,4)        // mark a link
y.  ...                   // #n rhothor commands
z.  rtFileClose()         // end of rhothor™ flash job

```

After creation of the flash job, it will appear in the flash explorer and the index (the job’s first sector) can be used to fetch the job from flash.

Script example: fetching a Flash Job

```

1.  rtListOpen(1)           // start rhothor™ command list
2.  rtIndexFetch(1)         // execute flash job located on flash sector 1
3.  rtListClose()          // end rhothor™ command list

```

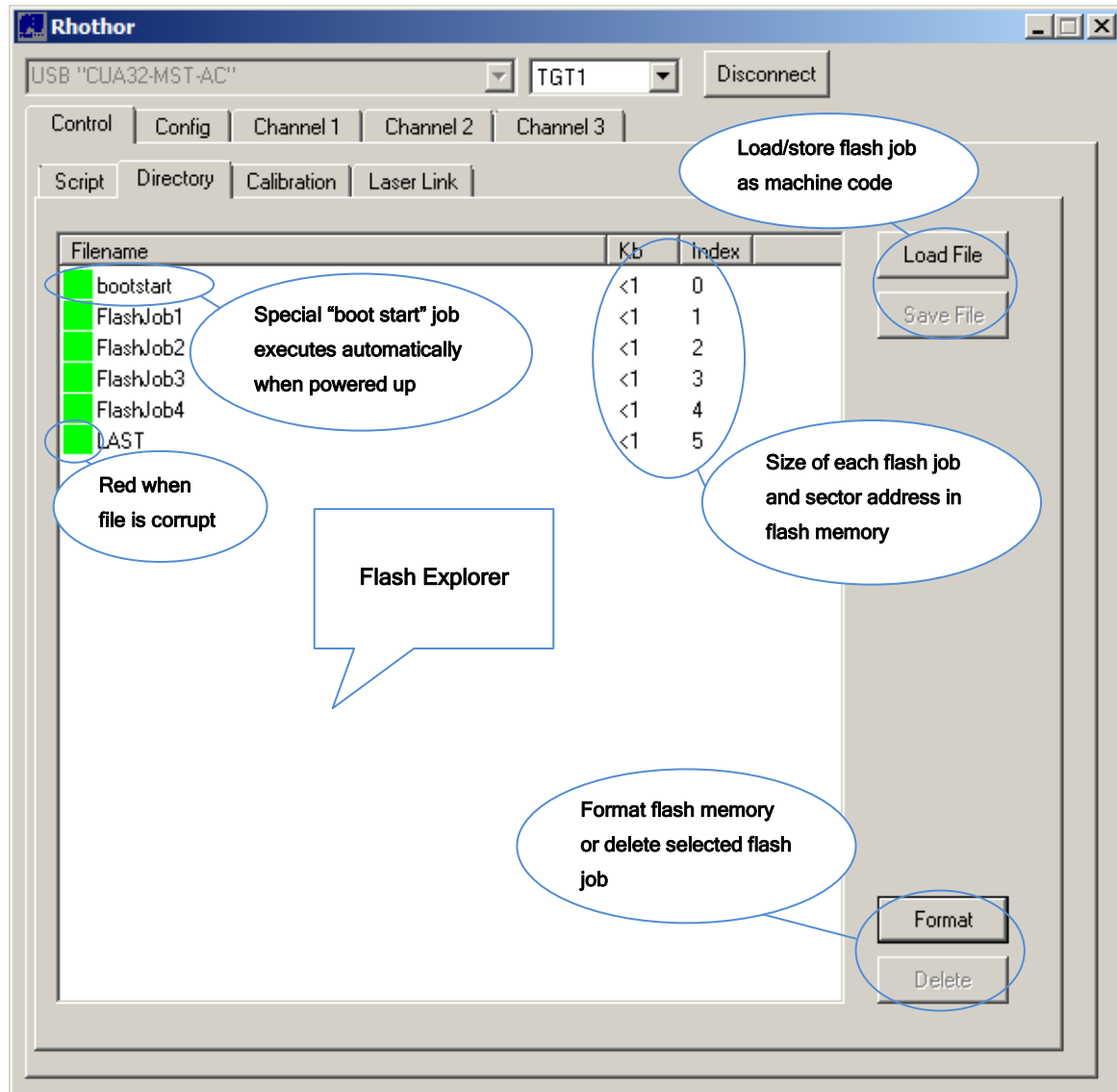
4.2.4 Creating a boot script

When the system is operated without a host computer, it must be able to start list processing automatically. Deflection and other control commands can be stored to flash in a boot sector. When the deflection control system is powered on, it scans the boot sector. When data is found, it will start automatically. When the CUA32 devices are used in streaming mode, a boot script can be used for initialization.

A boot script is created by starting the script with an “rtListOpen(3)” command and ends it with an “rtListClose” command. When the boot script is created with rhothor.exe, the source code will also be stored into the flash memory. This code can be reloaded into the script editor from the flash by pressing the “Load Boot start” button.

4.3 DIRECTORY

In the flash explorer you see all jobs created on the flash memory, including the boot script. The index of each flash job is the start sector of the job. This index is used to execute a certain job. Besides the index also the flash file size is shown.



4.3.1 Delete /Format

The flash memory can be completely formatted by clicking the “Format” button. All data will be erased from the flash. A single Flash job can be selected and deleted from the flash by clicking the “Delete” button.

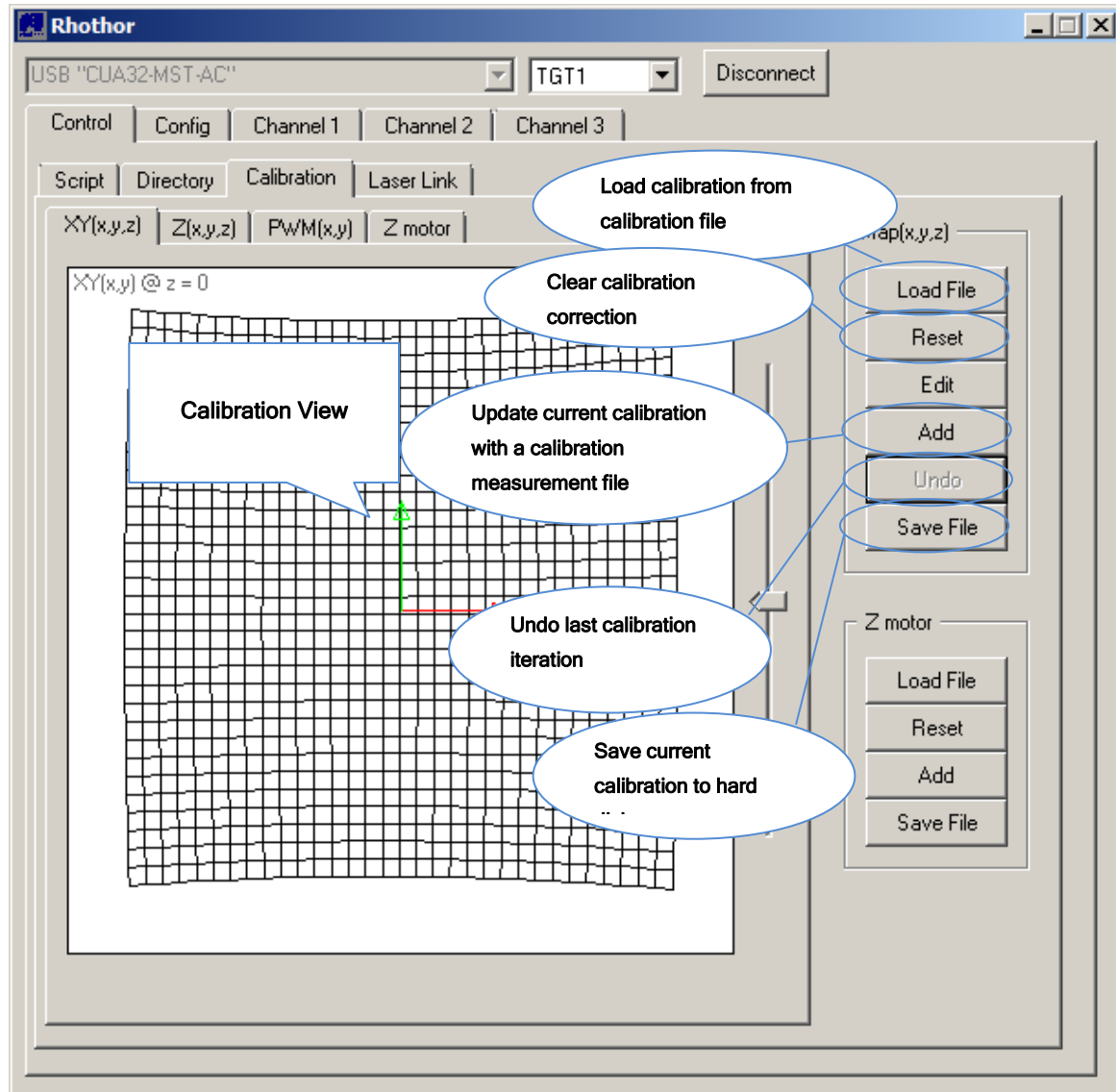
4.3.2 Load /Save File

When the rhothor™ software creates a flash job, it compiles the rhothor™ commands into machine code. Machine code is a block of instructions which can be directly executed by the CUA control board. It is possible to store this block of machine code on the hard disk or to load this instruction block from a file straight into the memory of the flash.

4.4 TARGET CALIBRATION

rhothor™ deflection systems use lookup tables to handle projection distortions. A calibration table is stored for every Target

inside the flash memory of the Front End controller (CUA32 MST) control board. The Calibration tab provides means to view and edit this data. The data shown applies to the target as selected by the Target index (in this case TGT1).



4.4.1 Reset

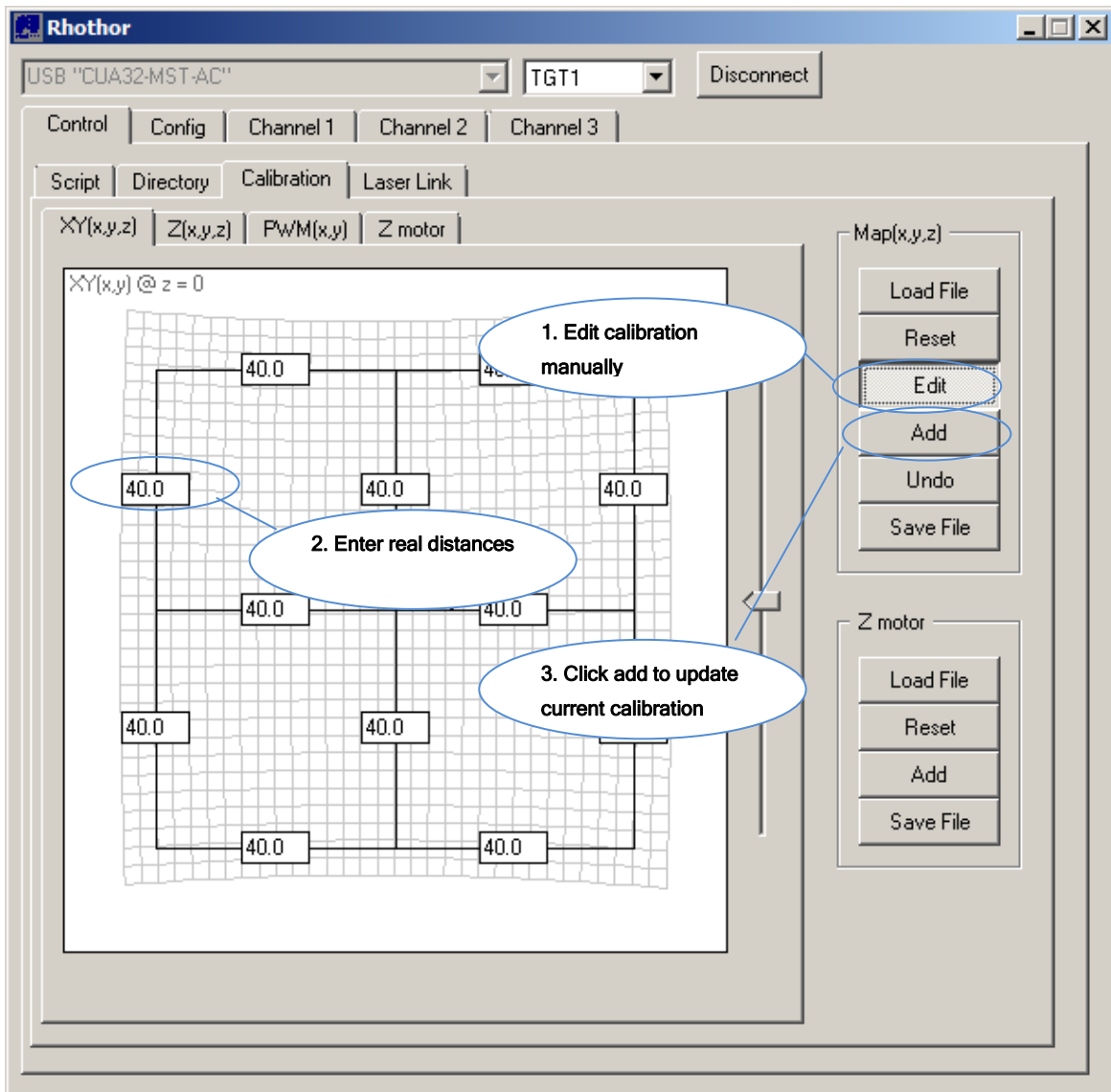
Pressing button "Reset" will reset the calibration for the Target currently loaded in the system.

4.4.2 Edit Calibration

The edit calibration is a manual calibration tool. It provides a quick calibration and is usable when the setup is optically well aligned and absolute accuracy needs are moderate.

To use:

- Use the Target selection combo to select the target
- Mark the calibration figure by pressing the button "Calibration" followed by "Run" on the "Script" tab.
- Go to "Calibration" tab and press button "Edit"
- Enter the real measured distances from a marked sample. Click Add button to compensate the distortion using the filled in data. You can undo the last calibration iteration by pressing the Undo button.



4.4.3 Add

By nature calibration differs from system to system. Therefore errors need to be measured and compensation needs to be adjusted. When the manual 3 by 3 point calibration doesn't result in the desired accuracy this iteration can be done by laser marking a raster of crosses and measure their positions. The result of those measurements has to be stored in a data file. This data file can be added to the current calibration table by pressing the button "Add".

4.4.4 Load /Save File

The calibration lookup table can be stored to a file by pressing the button "Save File". Pressing the button "Load File" will pre-set the calibration table with file data.

4.5 LASER LINK

When IO17 is configured for Laser Link and a Laser Link adapter is attached to the controller, the Laser link can be controlled through the LaserLink tab.

The screenshot shows the Rhothor software interface. At the top, there is a title bar with the name 'Rhothor' and standard window controls. Below the title bar, there is a status bar showing 'USB "CUA32-MST-AC"' and a dropdown menu set to 'TGT1', along with a 'Disconnect' button. The main interface is divided into several tabs: 'Control', 'Config', 'Channel 1', 'Channel 2', and 'Channel 3'. Under the 'Control' tab, there are sub-tabs for 'Script', 'Directory', 'Calibration', and 'Laser Link'. The 'Laser Link' sub-tab is active, displaying a control panel. This panel is annotated with two callouts: 'Change laser settings' and 'Laser State'. The 'Change laser settings' callout points to a group of controls including 'Power (%)' (set to 50), 'Frequency (KHz)' (set to 100), 'Duty Cycle (%)' (set to 50), 'Emission Enable' (with 'Clear' and 'Set' buttons), and 'Guide Laser' (with 'Clear' and 'Set' buttons). The 'Laser State' callout points to a vertical list of status indicators: 'Interlock', 'Emission Enable', 'Guide Laser', 'Alarm 1', 'Alarm 2', 'Alarm 3', and 'Alarm 4', each with a corresponding radio button.

5 DEFLECTORS

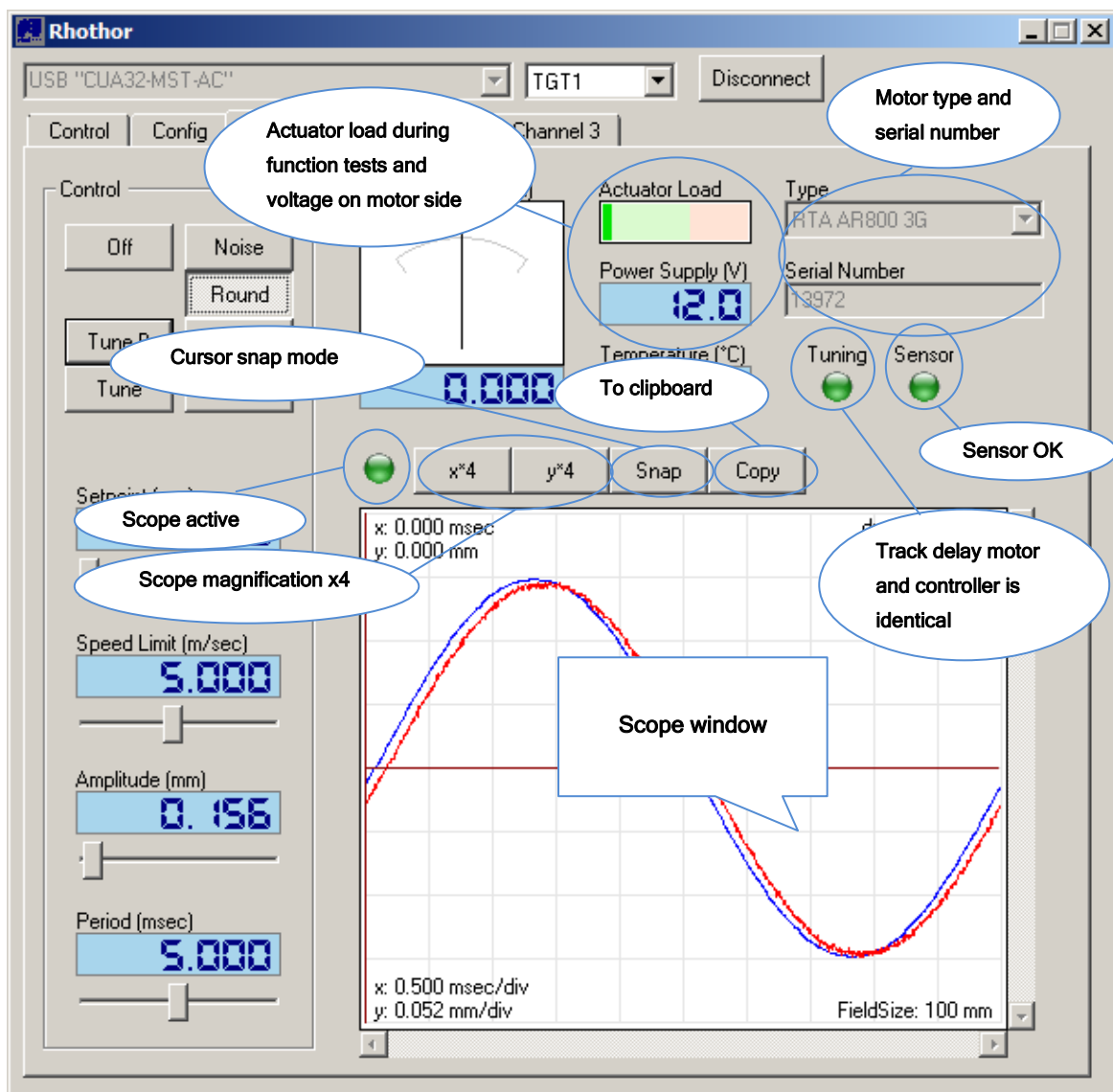
Behavior of the connected deflector can be observed and altered using the “Channel” tab. After setting the target properties on the “Config” tab the connected deflector can be tuned by simply pressing the “Tune” button. Afterwards its behavior can be verified. The deflector can be set-up to execute periodically jumps, rounds and squares. Applied set point speeds can be clipped. This is useful when simulating processing speeds in jump function tests. A set point slider provides full field coverage by changing the center point of the applied test signal.

The screenshot displays the Rhothor software interface for controlling a deflector. The window title is "Rhothor" and the device is identified as "USB 'CUA32-MST-AC'". The interface is divided into several sections:

- Control Panel:** Contains buttons for "Off", "Noise", "Round", "Square", "Jump", "Tune B", and "Tune". A callout bubble points to the "Off" button with the text "Switch off amplifier deflector". Another callout bubble points to the "Noise", "Round", "Square", and "Jump" buttons with the text "Set deflector in test mode, set point function can be: 1. constant at 'Noise' test, 2. cosine at 'Round' test, 3. ramp at 'Square' test, 4. step at 'Jump' test". A third callout bubble points to the "Tune" and "Tune B" buttons with the text "Alternative tuning set for elevathor™ actuators".
- Auto-tune deflector:** A callout bubble points to the "Tune" and "Tune B" buttons with the text "Auto-tune deflector".
- Setpoint (mm):** A digital display shows "0.000" with a slider below it.
- Speed Limit (m/sec):** A digital display shows "5.000" with a slider below it.
- Amplitude (mm):** A digital display shows "0.156" with a slider below it.
- Period (msec):** A digital display shows "5.000" with a slider below it.
- Graph:** A plot showing the set point function. The x-axis is labeled "x: 0.500 msec/div" and "y: 0.052 mm/div". The y-axis is labeled "y: 0.000 mm" and "dy: 0.000 mm". The graph shows a sine wave. A callout bubble points to the sliders with the text "Sliders to set parameters of set point function: offset, maximum speed, amplitude and period." The graph also shows "dx: 0.000 msec" and "dy: 0.000 mm" at the top right, and "FieldSize: 100 mm" at the bottom right.

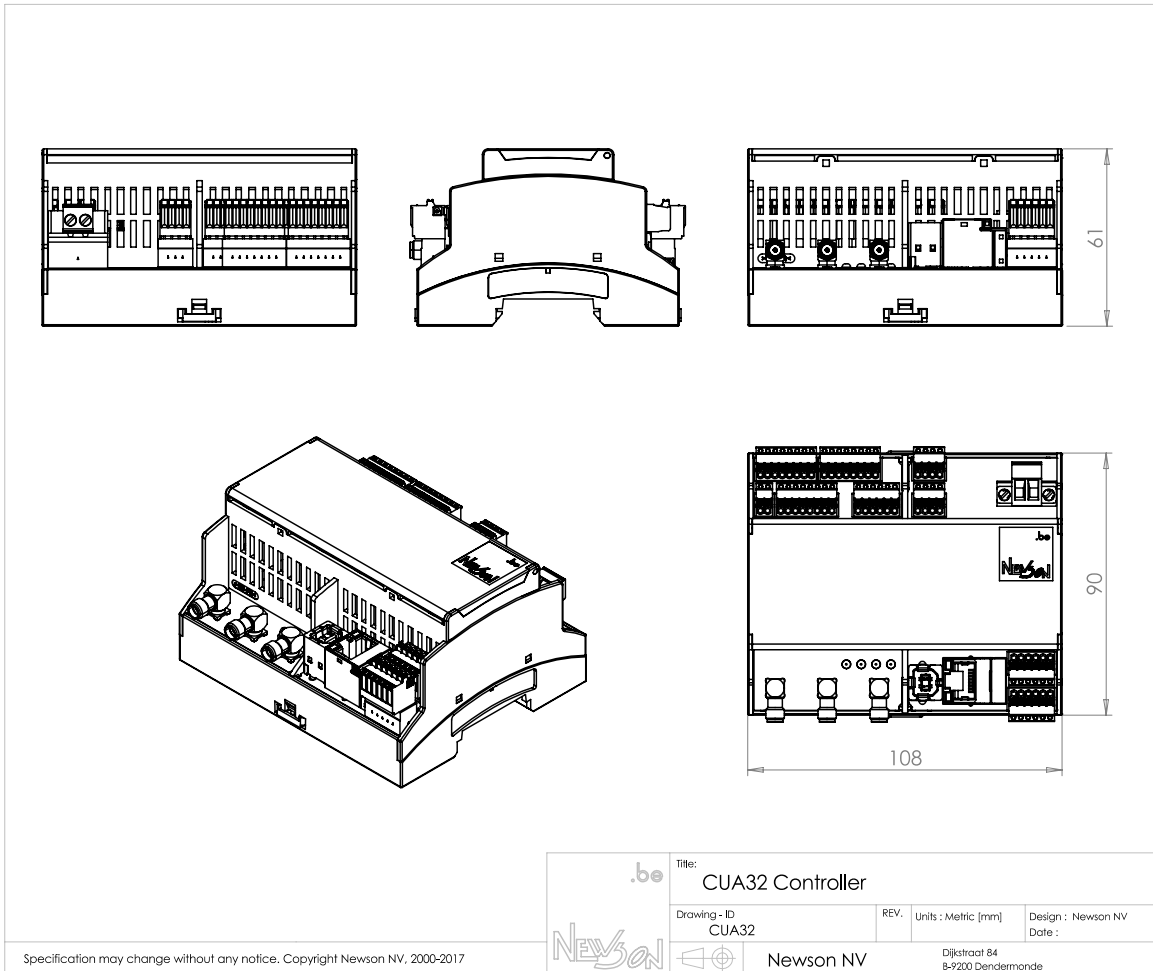
The window shows life deflector status information:

- An actuator Load bar to indicate over steering.
- A voltage meter to indicate the power supply voltage.
- A temperature readout to indication of the deflectors temperature (at CPU core)
- A LED to indicate the status of the internal optical sensor. If this LED turns red the system needs servicing.
- A LED to indicate that the deflector is tuned according to the selected track delay (config page). Whenever the LED is yellow the deflector needs to be tuned.
- A position indicator shows the current motor position.
- Motor type and serial number
- An integrated scope window to analyze behavior of the motor to several function tests. The blue line shows the set point stream send to the motor and the red line show the actual position stream, coming from the feedback of the motor. A green LED indicates when the scope is active.



6 APPENDIX

6.1 DIMENSIONS



6.2 STATUS LED

| | RED | GREEN | TOGGLE |
|------------------------------|--|--|--|
| LED1 Power LED | - | <ul style="list-style-type: none"> • Power on | - |
| LED2 Target LED | <ul style="list-style-type: none"> • Boot | <ul style="list-style-type: none"> • Selected Target during "Target Mapping Dialog" (rthor.exe) | <ul style="list-style-type: none"> • Patching Target |
| LED3 Target LED | <ul style="list-style-type: none"> • Boot • Channel Mode (rthor.exe) | <ul style="list-style-type: none"> • List in process by target | <ul style="list-style-type: none"> • Patching Target |
| LED4 Front end LED | <ul style="list-style-type: none"> • Boot • Flash Access | | <ul style="list-style-type: none"> • Patching • Flash Access |