

PowerXP Compact Motorized Attenuator

User Manual



Contents

PowerXP	3
1. Introduction	3
1.1. PowerXP short description	3
1.2. PowerXP Maxi main specifications	4
1.3. General safety requirements	4
1.4. Regulation	4
1.5. Operating and storage conditions	5
2. Operation Principle	5
3. Package Contents	5
4. PowerXP Software Description	6
4.1. “AltOne” software	6
4.2. Computer requirements	6
4.3. Software installation	6
4.4. Program first run	9
4.5. “Select Device” tab page	9
4.6. “Calibration” tab	10
4.7. “PowerXP” tab	12
4.8. “Settings” tab	13
4.9. “User Manual” tab	14
4.10. “About” tab	14
5. PowerXP Controller Hardware	15
5.1. Controller specifications	15
5.2. Controller usage	15
6. Serial and Ethernet Communication	16
6.1. Serial protocol description	16
6.2. Ethernet protocol description	16
6.3. Command execution	16
6.4. Controller commands	19
7. Main Dimensions	21
7.1. PowerXP	21
7.2. PowerXP Controller	21

PowerXP

November 2022

Copyright © Altechna, UAB. All rights reserved.

No part of this manual, including the products and software described in it, may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form or by any means, except documentation kept by the purchaser for backup purpose, without the express written permission of UAB Altechna (hereinafter Altechna).

Product warranty or service will not be extended if:

1. The product is repaired, modified or altered. Unless such repair, modification or alteration is authorized in writing by Altechna.
2. The LOT number of the product is defaced or missing.

Altechna provides this manual “as is” without warranty of any kind, either express or implied, including but not limited to the implied warranties or conditions of merchantability for a particular purpose. In no event shall Altechna, its directors, employees or agents be liable for any indirect, special incidental, or cosequential damages (including damages for loss of profits, loss of business, loss of use or data, interruption of busines and the like), even if Altechna has been advised of the possibility of such damages arising from any defect or error in this manual or product.

Specifications and information contained in this manual are furnished for informational use only, and are subject to change at any time without notice, and should not be construed as a commitment by Altechna. Altechna assumes no responsibility or liability for any errors or inaccuracies that may appear in this manual, including the products and software described in it.

This manual is directly connected to firmware version. Latest firmware, software and manual versions can be downloaded from www.altechna.com.

Products and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies and are used only for identification or explanation and to the owner’s benefit, without intent to infringe.

1. Introduction

This user manual is designed to help to install and operate PowerXP. Before installing and operating PowerXP please read installation and operation instructions carefully. Safety instructions must be read carefully. If there are any questions about contents of this manual, please contact info@altechna.com. Altechna reserves the right to update contents of this manual without any notification.

1.1. PowerXP short description

Compact motorized attenuator - PowerXP is a computer-controlled laser beam attenuation device. It attenuates a free space laser beam/pulse continuously without introducing additional energy fluctuations. PowerXP is controlled by a computer via USB.

1.2. PowerXP Maxi main specifications

Typical application	High power CW and pulsed lasers
Time between min and max	<0.2 s
Resolution	0.003125 deg
Damage threshold	>20 J/cm ² @ 1064 nm, 10 ns, 10 Hz
Attenuation range	<0.1 – >99%

1.3. General safety requirements

Compact motorized attenuator is designed to operate in conjunction with laser systems. All applicable rules and regulations for safe operation of lasers must be known and applied while installing and operating PowerXP. The customer is solely responsible for laser safety while using PowerXP as a standalone device or integrated into system. The customer must consider protective measures.

While assembling or operating PowerXP, do not stare at the direct or scattered laser light even with safety goggles. All parts of the body must be kept away from the laser radiation. While adjusting laser beam through PowerXP, laser power must be kept as low as possible. Hazardous laser radiation can increase while optical components or instruments are used in combination with PowerXP. Appropriate eye protection must always be worn at all times. Electrical safety requirements must be complied while assembling and operating PowerXP.

1.4. Regulation

Attention!

The following statement applies to the products covered in this manual, unless otherwise specified herein. The statement for other products will appear in the accompanying documentation.

These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can create radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference with radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment causes harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna
- Increase the separation between the equipment and receiver
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected
- Consult the dealer or an experienced radio/TV technician for help

Altechna is not responsible for any radio or television interference caused by modifications of this equipment or the substitution or attachment of connecting cables and equipment other than those specified by Altechna. The correction of interference caused by unauthorized modification, substitution or attachment will be treated as responsibility of the user.

Attention!

Cellular phones or other radio transmitters are not recommended to be used within the range of three meters of this unit since the electromagnetic field intensity may then exceed the maximum allowed disturbance values according to IEC 61326-1.

1.5. Operating and storage conditions

For proper PowerXP functioning please use the assigned controller (found in the same package). Using unassigned controllers might be harmful to the device.

Environmental conditions that must be hold while storing, servicing and operating are:

- Storage temperature should be between -30 °C and +30 °C
- Operating temperature is 25 °C ± 15 °C
- PowerXP must be protected from humidity, dust and corrosive vapors to avoid damaging optical components and electronics
- Avoid strong static electricity and electromagnetic fields

2. Operation Principle

PowerXP incorporates a rotating quartz $\lambda/2$ phase waveplate and one polarizer which separate s-polarized and p-polarized beams (fig. 1). The intensity ratio of the two beams may be continuously varied without alteration of other beam parameters by rotating the waveplate. Proper functioning of PowerXP requires optimal configuration of optical elements regarding to incident laser beam polarization contrast. Higher incident laser beam polarization contrast leads to higher PowerXP output polarization contrast.

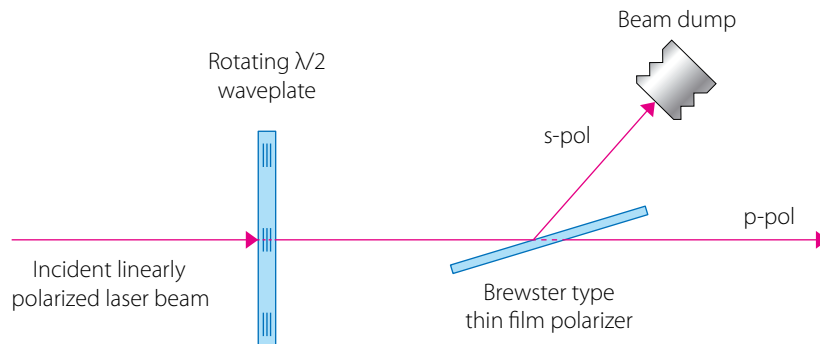


Figure 1. PowerXP operation principle. The intensity of the incident beam is varied by means of a $\lambda/2$ phase waveplate and a polarizer.

3. Package Contents

- Compact motorized attenuator - PowerXP
- Controller
- USB cable (2 m length recommended but no longer than 3 m)
- 12 V power supply
- USB flash drive with software, user manual and SDK files
- Polarizer – **(already in the attenuator)**
- Waveplate – **(already in the attenuator)**

4. PowerXP Software Description

4.1. “AltOne” software

“AltOne” software is dedicated software for all Altechna products. It is designed to automatically detect different Altechna products by the product serial number.

Software can operate only when connected to “USB” or “RS232” ports. When communicating through “Ethernet” port you should established connection by oneself considering your system’s configuration.

Software can be downloaded from product page:

<https://www.altechna.com/products/powerxp-compact-motorized-attenuator/>

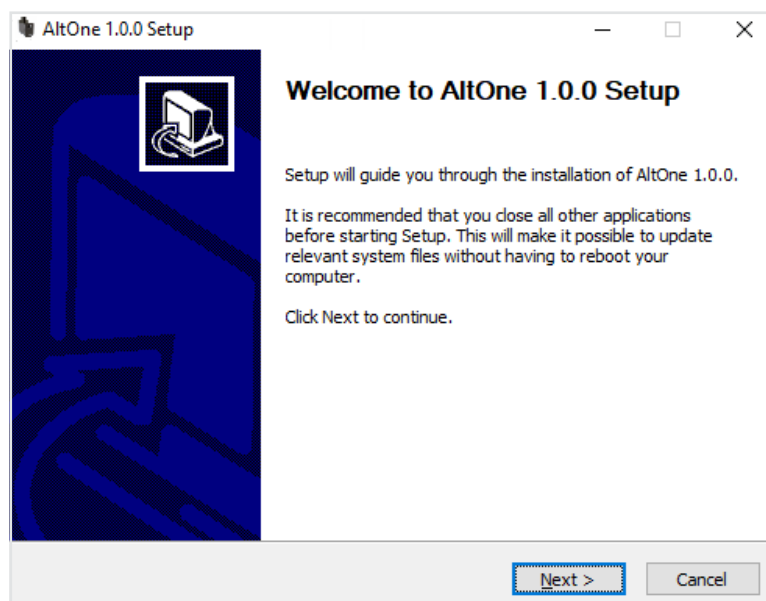
4.2. Computer requirements

- Free USB port or RS232 port. PowerXP is compatible with USB 1.1, USB 2.0 and USB 3.0 Computer administrator rights (only for installation)
- Operating systems supported: Windows XP sp3, Windows Server 2003 sp2, Windows Vista sp1, Windows Server 2008, Windows 7, Windows 8 and Windows 10.
- Microsoft.Net framework 4.0 redistributable (installs automatically)
- Microsoft Visual C++ Redistributable 2019 or later (installs automatically)
- Adobe Acrobat Reader 7.0 or above.

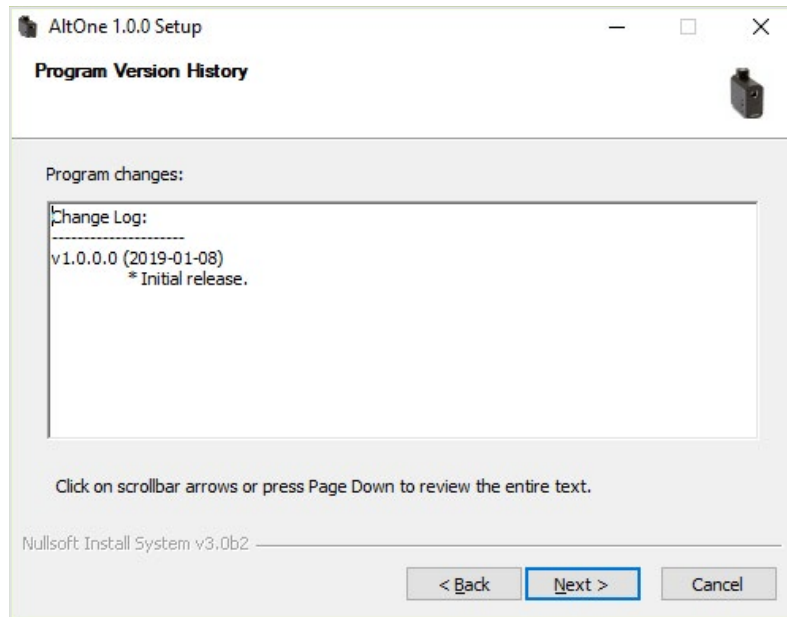
4.3. Software installation

“AltOne” is dedicated software for all Altechna products. All products from Altechna should work with this software and it is not required for multiple installations.

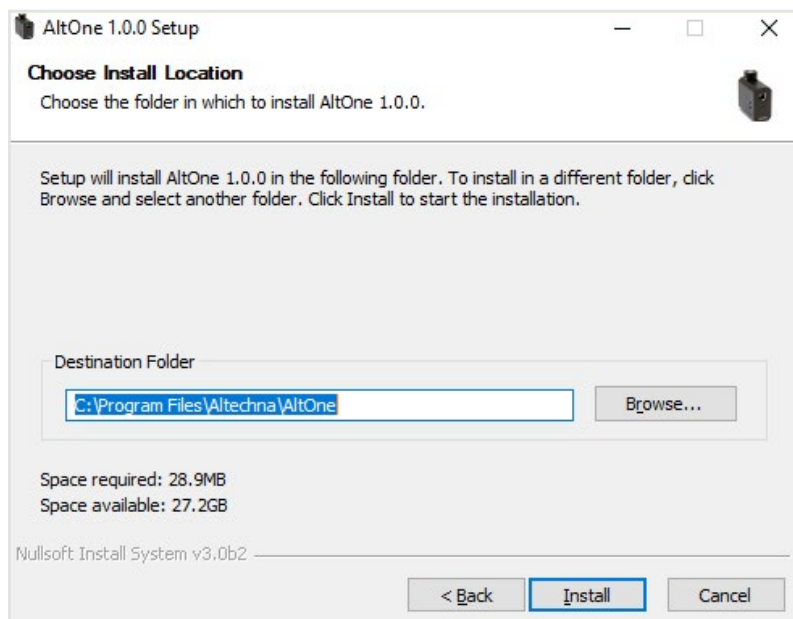
1. Run the “AltOne-Setup.exe” installation file. In case the software is installed on an operating system that does not meet requirements, only USB drivers will be installed. Click “Yes” to continue.
2. Installation window will appear, click “Next” to continue:



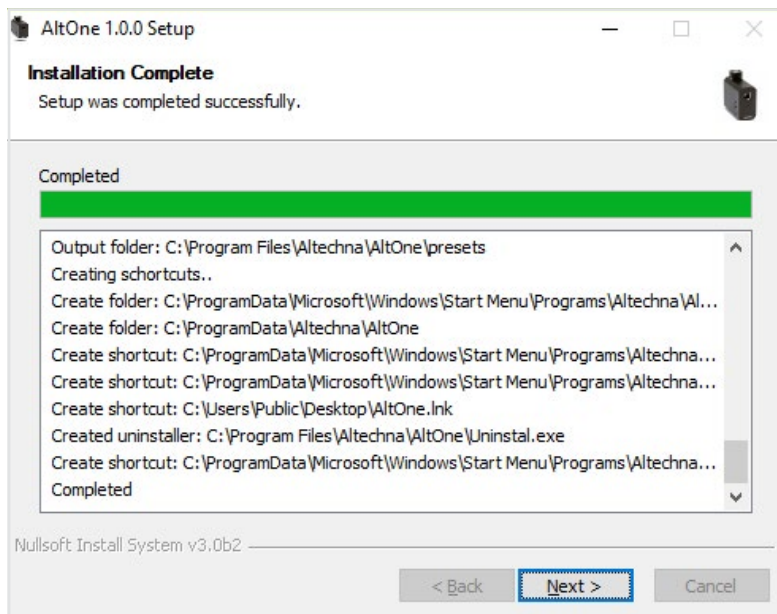
3. Click “Next”:



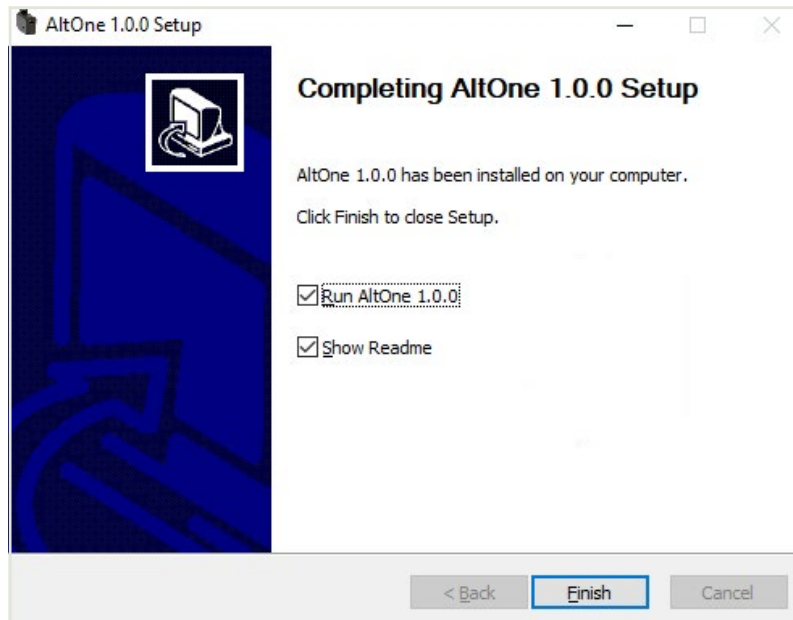
4. Select installation directory and click “Next” to begin installation:



5. “AltOne” software requires “Microsoft.NET Framework 4 Client Profile” or higher version to be installed. Setup will offer to download it, choose “Yes” if active internet connection is available. Choose “No” to download it from www.microsoft.com/en-us/download/details.aspx?id=17113 and install it manually. One should choose “No” if newer version of Microsoft.NET framework is unwanted, or working internet connection is not available. This dialog will not appear if framework is already installed. Download size is 42 Mbytes.
6. Setup will download “Microsoft .NET Framework 4 Client Profile”.
7. After download is finished, “Microsoft .NET Framework 4 Client Profile” will be installed, wait for it to complete. This can take more than 10 minutes on slower machine.
8. Setup will finish by installing drivers. Click “Next” to continue:



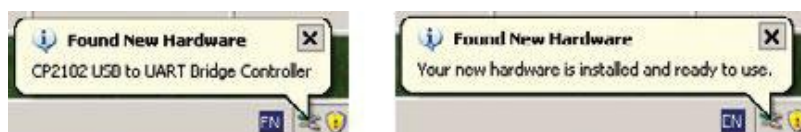
9. Click “Finish” to end installation. The program cannot be opened if only drivers were installed (see step 2 for details).



10. “AltOne” software icon will appear on “All Users” desktop and “All Users” start menu.

When using PowerXP with USB:

1. Connect PowerXP to its controller.
2. Connect PowerXP controller and PC via USB cable.
3. Plug in PowerXP power supply jack.
4. Windows will detect new hardware. Wait until windows configures new device.



4.4. Program first run

Launch “AltOne” program using “AltOne” icon on desktop or from “Start Menu → All Programs → Altechna → AltOne → AltOne”. AltOne “Select Device” window will appear. At least one device must be displayed on the list. If the list is empty, please check USB/RS232 cable, power connection. “Status” LED must be active if power is OK. Click refresh button to rescan all ports.

4.5. “Select Device” tab page

“**Select Device**” tab contains a list of currently connected and powered attenuators. This window is used to choose device to work with if there are several controllers connected to a single computer. Functions of this window are described in the picture below.

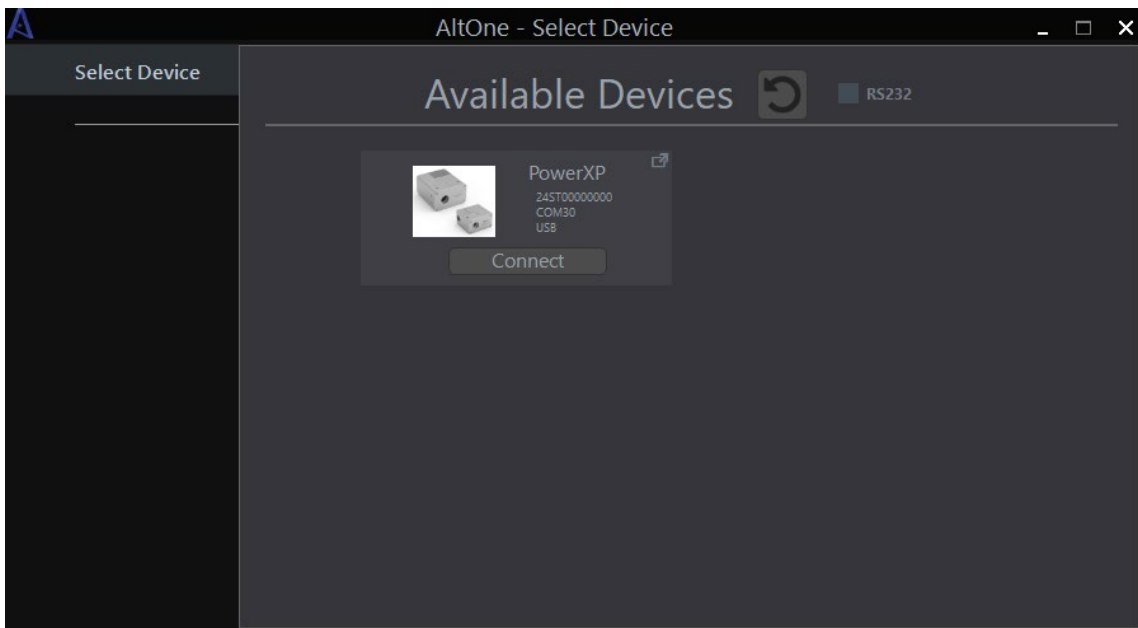


Figure 4. a) AltOne “Select Device” tab. It is displayed every time program is started. There is one device (PowerXP) attached to PC in shown screenshot. Click “Connect” to load control tabs for it.

Every device in “Select Device” tab list has information about connected device. This information contains:

- Device picture
- Device type
- Device serial
- Port name
- Connection type (“USB” or “RS232”)

Every device in the list can be connected using “Connect” button.

In case of multiple device connection at the same time use pop-up icon “” in the top right corner to open device in separate window.

RS232 connection

In case of connecting device via RS232, you should add devices manually. To do that check “RS232” check box near the “Refresh” button, then dropdown box and button “add” will appear. When you click on dropdown box, software will populate it with all ports available to PC. Then you just select specific port and push button “add” to add new RS232 device to a “Select Device” window.

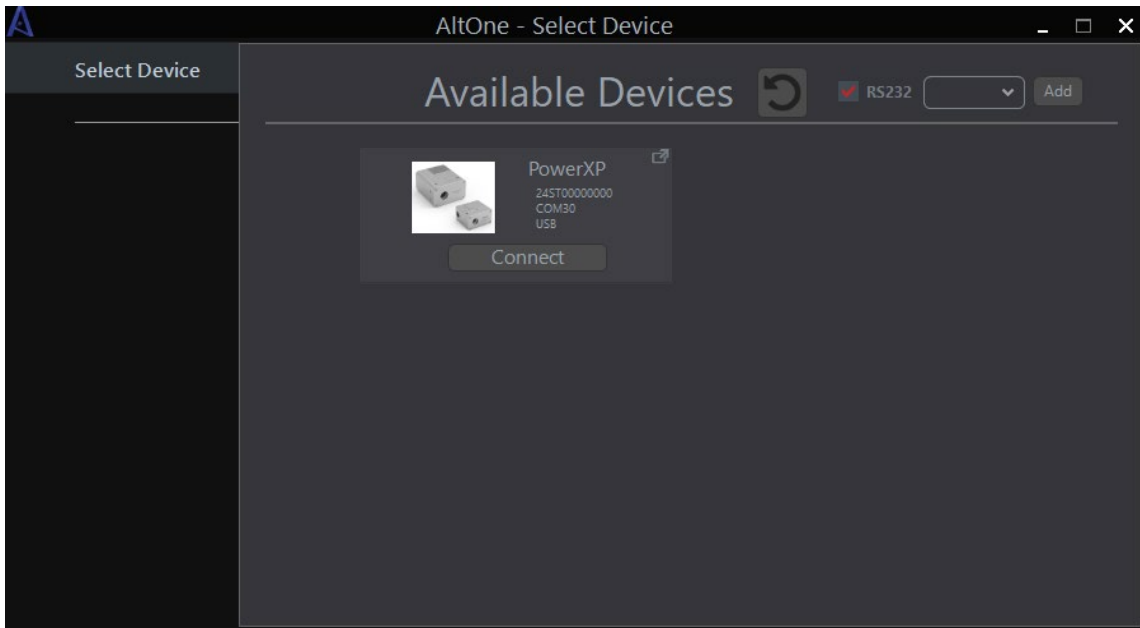


Figure 4. b) AltOne “Select Device” tab. RS232 connection controls are on the top right corner.

4.6. “Calibration” tab

After successful connection to your device you should be automatically redirected to main control tab (Figure 5.).

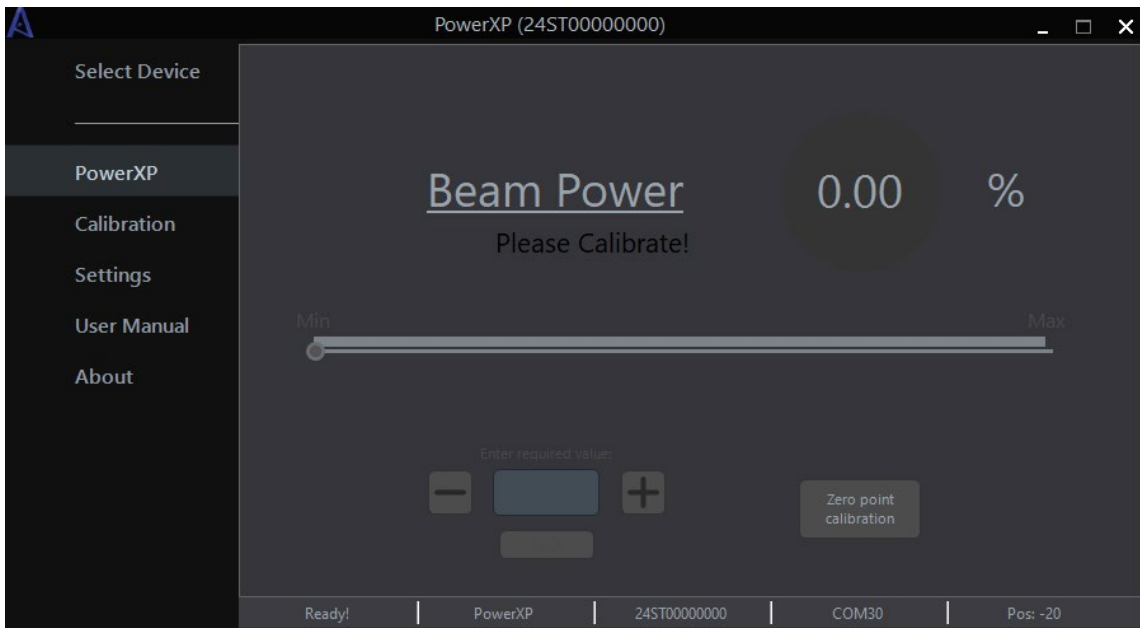


Figure 5. PowerXP tab.

If your device is not calibrated all controls will be disabled. To calibrate your device, go to “Calibration” tab.

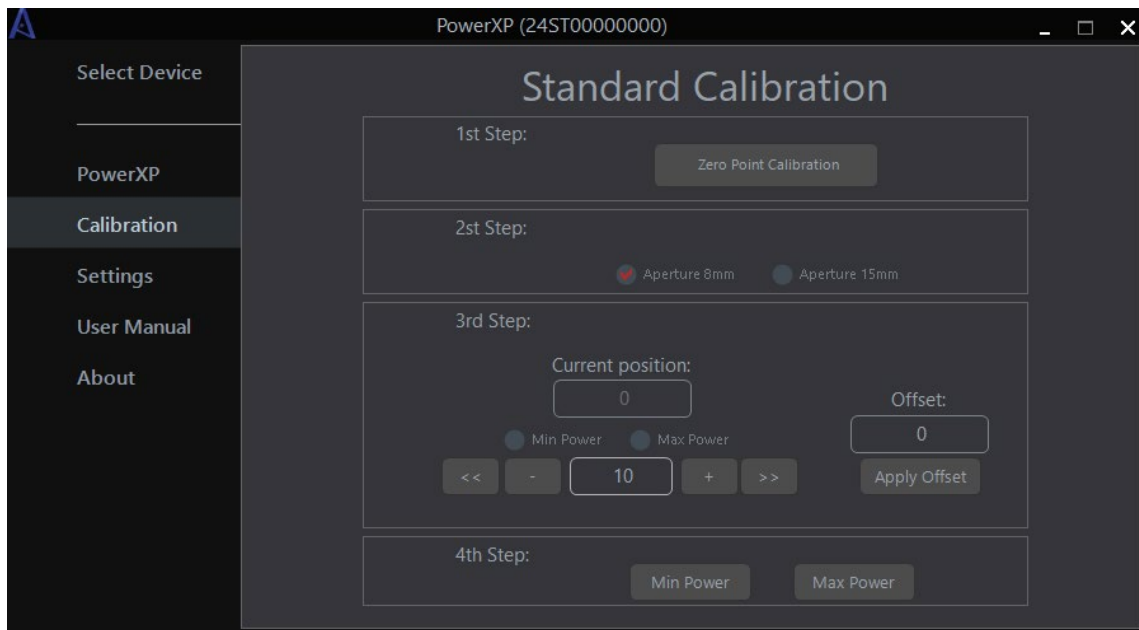


Figure 6. Calibration tab.

Calibration procedure:

In order to correctly change the output power, the software needs to know the angle of the $\lambda/2$ waveplate's fast axis. It can be fixed in any angle with respect to the rotator. The purpose of calibration is to fix angular offset between the $\lambda/2$ waveplate and the rotator hardware zero position.

1. On "1st Step" click on "Zero Point Calibration". This will drive the waveplate holder to hardware limit switch (zero position). Step 1 must be performed every time the device is turned on or after extensive period of usage to eliminate any accumulated position error.
2. At "2nd Step" select correct clear aperture (CA) of currently connected attenuator. Waveplate CA could be 8 or 15 mm in diameter.
3. Rotate waveplate by using buttons in section "3rd Step", and measure laser power after the attenuator. Numbers on the text box represent motor moving distance (expressed in stepper motor steps) and "+"/"-" buttons represent rotation direction. One step equals to 0.003125 degrees.
Find minimum or maximum signal power and select the appropriate check box ("Min Power" or "Max Power"). Angular offset is recalculated and shown in "Offset" field every time check box selection is clicked. At this moment, main calibration goal is completed and "Apply Offset" button should be clicked (if necessary, offset value can be entered manually to an offset value box).
Further step is not mandatory, but should be considered for convenience.

Note

It is advised to search for lowest power (max attenuation) position, because usually it can be spotted more easily and accurately.

You only need to find minimum or maximum power only once. When you push "Apply Offset" offset value is saved to internal memory of the controller and is retrieved even if device has been powered off.

4. Calibration result can be verified by using buttons in "4th Step" panel. By clicking "Min Power" or "Max Power" motor will go to minimum or maximum power position. If everything is properly calibrated, you should see minimum or maximum attenuation.

4.7. “PowerXP” tab

“PowerXP” tab is used to change laser power after PowerXP attenuator. Main components of this window are described below (Figure 7).

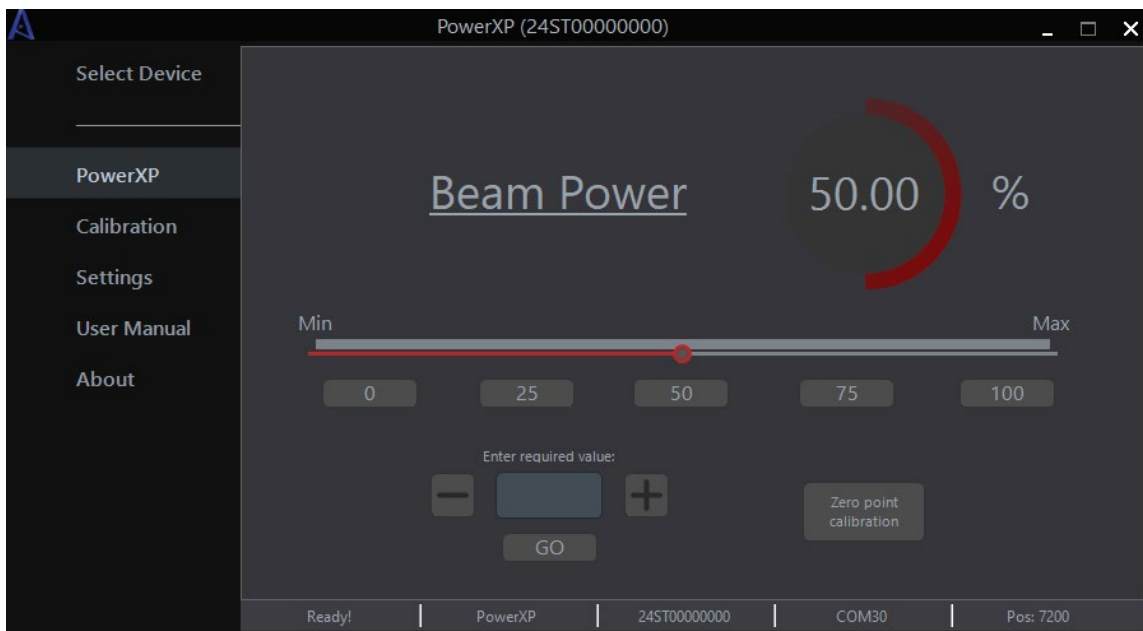


Figure 7. PowerXP tab.

Power meter panel always shows existing power after the attenuator. Percentage range is from 0% to 100% of transmission. Zero percent means that waveplate occurs at 45 deg angle, beam polarization is rotated by 90 deg and is maximally attenuated. Display reading “100%” means that waveplate is rotated at minimum attenuation – maximum transmission angle.

Slider changes power from 0% to 100% (or from minimum to maximum custom units of measure) with resolution of 0.1%. There is a “Min” and “Max” labels which when clicked drive attenuator to minimum or maximum power accordingly.

Enter required value field is used to manually enter value of PowerXP power. Decimal point symbol is “.” (dot), two decimal places can be used. Valid range is 0.00 % ... 100.00 % (when using different units of measure these values will be different). After entering desired value just hit “GO” button and value will be set.

Buttons “+” and “-” are used to adjust power continuously according to a button sign.

Preset buttons can be used to quickly set discrete values of output power. Button values could be changed in “**Settings**” tab. In order to use preset buttons, go to “**Settings**” tab to enable this function. This function is turned off by default.

Button “Zero point calibration” is used to recalibrate physical zero point of the device from which all movement of waveplate is calculated.

4.8. “Settings” tab

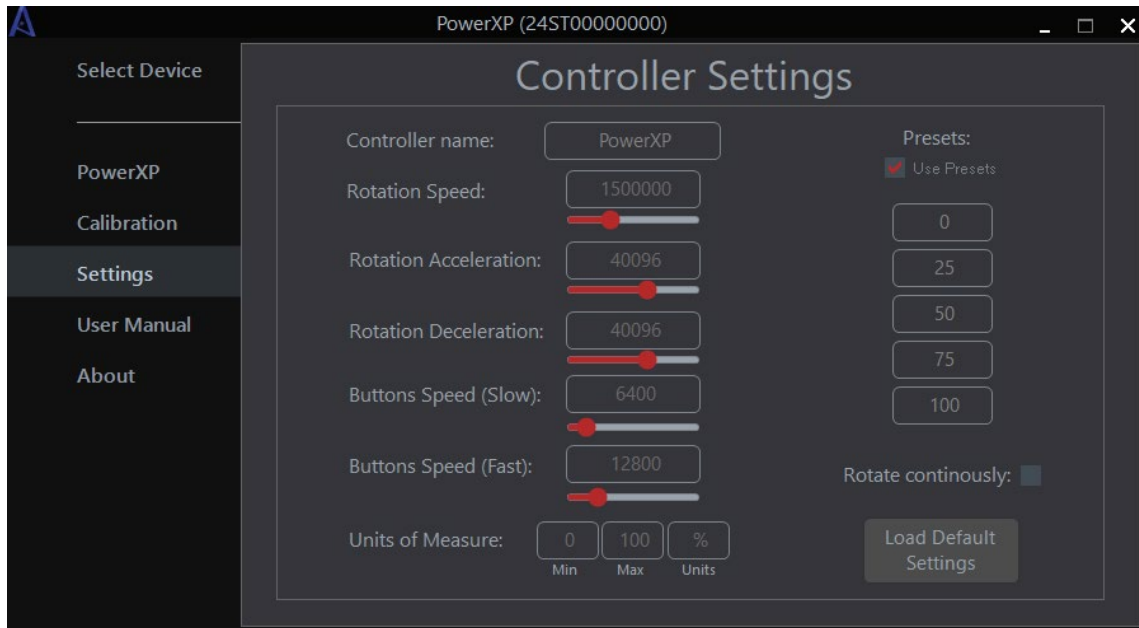


Figure 8. “Settings” tab.

In the “Settings” tab you can:

- Change **controller name** (up to 15 characters)
- Change **rotation speed** of the waveplate
- Change **acceleration** of the waveplate
- Change **deceleration** of the waveplate.
- Change rotation speed when using physical buttons (“first speed”*)
- Change rotation speed when using physical buttons (“second speed”*)
- Change units of measure and minimum and maximum values which will represent minimum and maximum attenuation
- Enable preset buttons
- Change preset buttons values
- Load default settings by clicking on “Load Default Settings”

* When using physical buttons to rotate waveplate firstly waveplate moves with “first speed“ and then after 2 seconds starts to move with “second speed”.

4.9. “User Manual” tab

In this tab you can comfortably read user manual of your PowerXP.



Figure 9. “User manual” tab.

When you enter this tab, you can maximize this window for greater readability. Also, you can zoom, print, change page and save document as if you were using “Adobe Acrobat Reader” software. This tab only works if you have “Adobe Acrobat Reader” software for reading PDF files.

4.10. “About” tab

In this tab you can see your product name, software version and controller’s firmware version.

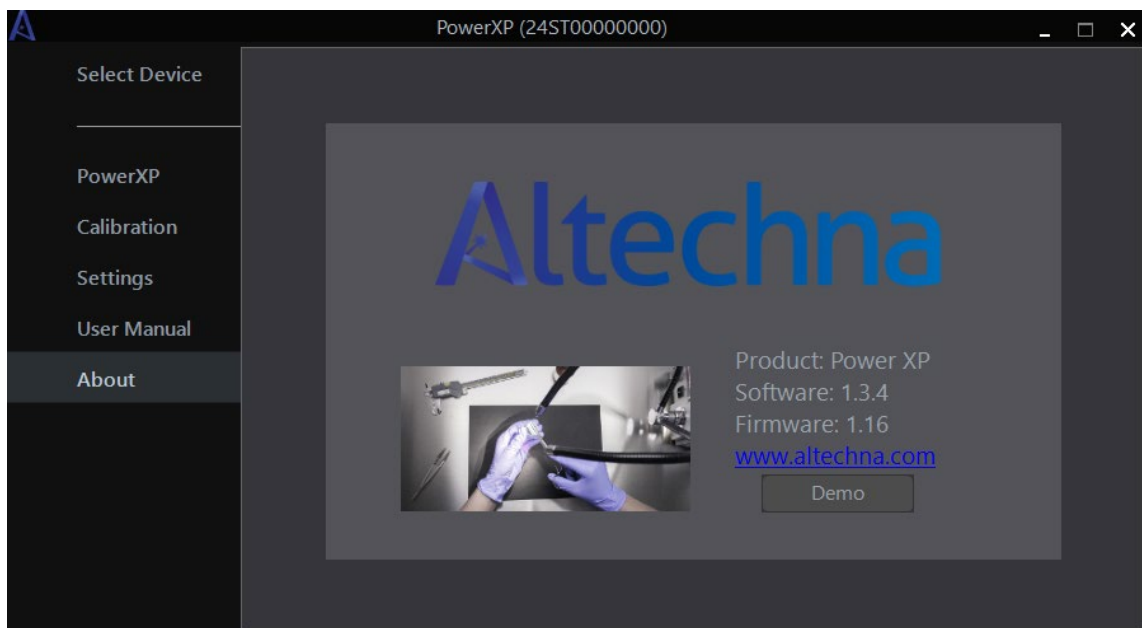


Figure 10. “About” tab.

5. PowerXP Controller Hardware

5.1. Controller specifications

PowerXP controller is bipolar stepper motor driver with specifications listed in Table 1 below.

Table 1. Controller specifications

Characteristic	Rating
Max output voltage	+12 V
Max output current	2 A
Current regulation type	Pulse Width Modulation
Microstepping capability	Steps 1/256
Position feedback	Open loop operation (no external position feedback encoder)
Controller protection	Driver have overheating and over current (2A) protection
Device can be operated by	USB, RS232 and Ethernet
Limit switch	One limit switch can be connected and used only for homing
Connector	26 pin D-SUB connector

5.2. Controller usage

FRONT VIEW

The following connections are situated in the front of PowerXP controller: power jack (12 V), 26 pin D-SUB for motor, Ethernet and RS232 connectors for PowerXP (Figure 11).



Figure 11. Front view of PowerXP Controller.

TOP VIEW

There are two buttons in the top of the PowerXP controller: “-” and “+”.

- “-” – while holding turns the waveplate to a minimum attenuation
- “+” – while holding turns the waveplate to a maximum attenuation

- **HOME** – holding both buttons **for 3 sec** moves attenuator to home position

There are 4 LED indicators:

- **Power** – indicates power from USB connection with PC
- **Status:**
 - If blinking slowly (1 blink in 1 sec) – need homing
 - If blinking fast (3 blinks in 1 sec) – the waveplate is rotating
 - If illuminating continuously – PowerXP is homed, the movement is stopped
- **MIN** – the PowerXP is at its minimum attenuation
- **MAX** – the PowerXP is at its maximum attenuation

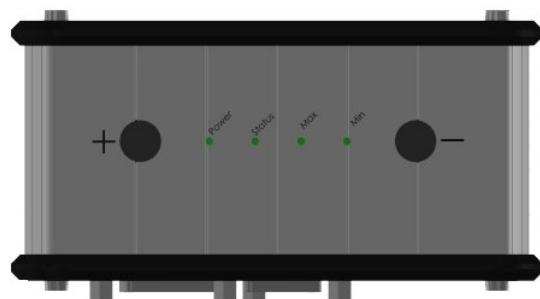


Figure 12. Top view of PowerXP Controller.

6. Serial and Ethernet Communication

6.1. Serial protocol description

When using “RS232” (or “USB”*) port all communication between controller and PC (or other device) is established by using RS232 communication protocol.

COM-port parameters are fixed on the controller-side:

Speed	115200 baud
Frame size	8 bits
Stop-bits	1 bit
Parity	none
Flow control	none
Byte receive timeout	400 ms

*USB port simulates COM port on a PC.

All the commands are the same for all communication protocols (see “6.4 Controller Commands”).

For more information please see “RS232 communication example” software and C# code (Visual Studio project).

6.2. Ethernet protocol description

When using “Ethernet” port all communication between controller and PC (or other device) is established by using TCP/IP communication protocol. When controller is connected to “Ethernet” port it constantly tries to connect to device (server) which IP address is specified in controller’s memory. To connect and communicate with controller you should create dedicated TCP server which will send and receive data to and from controller.

To achieve connection between server and the controller you must configure specific addresses and ports in the controller and the server (they must match in both devices). You can change controllers IP address and port using “TCP sample” software.

When creating server on your device do not use dynamic IP (DHCP), because your server IP address should be static for controller to maintain the connection.

All the commands are the same for all communication protocols (see “6.4 Controller Commands”).

For more information please see “TCP sample” software and C# code (Visual Studio project).

6.3. Command execution

All data transfers are initiated by the PC, meaning that the controller waits for incoming commands and replies accordingly. Each command is followed by the controller response, with rare exceptions of some service commands. One should not send another command without waiting for the previous command answer.

Command processing does not affect real-time engine control (PWM and etc).

Both controller and PC have an IO buffer. Received commands and command data are processed once and then removed from buffer. Each command consists of 6-byte identifier and optionally a data section followed by its 2-byte CRC. Data can be transmitted in both directions, from PC to the controller and vice versa. Command is scheduled for execution if it is a legitimate command and (in case of data) if its CRC matches. After processing a correct command controller replies with 1 byte – AA (for “OK”) or 01 (for “not OK”), followed by data and its 2-byte CRC, if the command is supposed to return data.

Device is a slave – it cannot initiate any data transfer. PC is a master. Each data transfer session looks like:

	Description	Note
PC	Send command	
Device	Confirm correct command receive (0xAA or 0x01)	If 0x01 (not correct command), PC should send command again
Device	Send additional data to PC	If needed to complete command

NOTE: OK - 0xAA, NOT OK - 0x01

In case of error in data from device to PC, PC initiate new session, starting from sending a command).

Each frame consists of **start symbol (@)**, data string **length (16 bits)**, ASCII **command (always 3 bytes)**, **data** (if need to transmit some data for command execution), **CRC (16 bits)** which is calculated for ASCII command and DATA fields:

Start	Length0, Length1		ASCII command			DATA		CRC0, CRC1	
@	0x00	0x00	A	B	C	0x00	0x00

Note for RS232 interface: device recognize end of the data frame if the pause after last transferred byte >50 ms.

For each correctly received command device answers - **0xAA**. If command is not correct it will answer - **0x01**.

Example command for homing:

Start (“@”)	Length0, Length1 (3 bytes)		ASCII command (“hom”)			DATA (none)		CRC0, CRC1	
0x40	0x03	0x00	0x68	0x6f	0x6d	0xd5	0x94

Example command for moving to “123456” absolute position:

Start (“@”)	Length0, Length1 (3 bytes)		ASCII command (“rad”)			DATA (“123456”)				CRC0, CRC1	
0x40	0x07	0x00	0x72	0x61	0x64	0x40	0xE2	0x01	0x00	0x1C	0xFD

Device calculate CRC, using this procedure (C):

```
uint16_t myCRC16_CCITT_CRC(uint16_t *crc, uint8_t c)
{
    uint8_t i;
    uint16_t cc;

    cc = ( (uint16_t) c ) << 8;

    for ( i=0; i<8; i++)
    {
        If (( (*crc) ^ cc ) & 0x8000)

        (*crc) = (( *crc ) << 1) ^ MTT;

    else (*crc) <<= 1;

    cc <<= 1;
    }
    return *crc;
}
```

Procedure for PC (C#):

```
//CRC16 XMODEM implementation
private static short calcCrc(byte[] data)
{
    unchecked
    {
        short crc = 0;

        for (int a = 0; a < data.Length; a++)
        {
            crc ^= (short)(data[a] << 8);
            for (int i = 0; i < 8; i++)
            {
                if ((crc & 0x8000) != 0)
                    crc = (short)((crc << 1) ^ 0x1021);
                else
                    crc = (short)(crc << 1);
            }
        }
        return crc;
    }
}
```

6.4. Controller Commands

Movement commands				
Command	Data	Return values	Description	Notes
“hom” (0x686F6D)	none	0xAA or 0x01 (OK or NOTOK)	Start the homing procedure.	
“rad” (0x726164)	4 bytes	0xAA or 0x01	Move to absolute micro step position x.	Works only when device is “Homed” .
“rgd” (0x726764)	4 bytes	0xAA or 0x01	Move x micro steps relative to the current position.	Works only when device is “Homed” .
“rgs” (0x726773)	4 bytes	0xAA or 0x01	Move x micro steps relative to the current position.	Works even if device is not “Homed” .
“stp” (0x737470)	none	0xAA or 0x01	Stop motor smoothly if it is currently running.	
Status and information commands				
Command	Data	Return values*	Description	Notes
“ost” (0x6F7374)	none	8 bytes (debug) + 4 bytes (flags) + 4 bytes (position) + 8 bytes (debug)	Get controller run state, position, limit switch states and etc.	“debug” bytes are just for debugging purposes. “flags” (logic 1 – active): bit0 – device is running bit1 – homing in progress bit2 – device is not homed bit3 – hardware error, can’t move bit4 – calibration corrupted bit5, bit6, bit7 – unused bit8 – motor driver reset detected bit9 – driver hi temp warning bit10 – left LS is pressed bit11 – error in load (motor) bit12 – motor driver error bit13 – stallguard is active bit14 – motor is standstill bit15 – target velocity reached bit16 – driver over temperature bit17 – target position reached bit18 – under voltage error bit19 – right LS is pressed bit20 – device is homed bit21 – calibration is done bit22 – load warning (open load detected on motor phase A or B) bit23 – device FRAM error
“pw ” (0x707720)	none	16 bytes (char)	Show serial number of a device.	
“n ” (0x6E2020)	none	17 bytes (char)	Show name of a device.	
“v ” (0x762020)	none	5 bytes (char)	Show firmware version of a device.	
“p ” (0x702020)	none	5 bytes (char)	Returns “pUSB:” if device is connected.	This command can be used to “ping” controller (to check if controller is attached to particular COM port).

“cd ” (0x6F7320)	none	4 bytes + 4 bytes + 4 bytes + 4 bytes + 1 byte + 4 bytes + 4 bytes + 4 bytes + 4 bytes + 4 bytes + 4 bytes + 4 bytes + 4 bytes + 5 bytes + 8 bytes + 8 bytes + 8 bytes + 8 bytes + 1 byte + 1 byte	Gets all parameters for the motor.	Data received: microsteps_per_degree (float) speed (Int32, usteps/sec) acceleration (int32, usteps/sec) deceleration (int32, usteps/sec) winding current (int32), mA limit_flags (uint8) timeout_speed_ms (int32) speed_btn_slow (int32) speed_btn_fast (int32) speed_home(int32) offset_microsteps (int32) min_power (float) max_power (float) measurement units (string 5 char) preset_but0 (double) preset_but1 (double) preset_but2 (double) preset_but3 (double) preset_but4 (double) flags_gui (uint8) flags_user (uint8)
-----------------------------	------	---	------------------------------------	--

Configuration commands

Command	Data	Return values	Description	Notes
“spd” (0x737064)	0x01 + 4 bytes of speed value	0xAA or 0x01 (OK or NOTOK)	Set motor speed. Value should be from 0 to 8000000.	microsteps/t** (Default value: 1500000)
“acl” (0x61636C)	0x01 + 4 bytes of speed value	0xAA or 0x01	Set motor acceleration. Value should be from 0 to 65535.	microsteps/ta*** (Default value: 40000)
“dcl” (0x64636C)	0x01 + 4 bytes of speed value	0xAA or 0x01	Set motor deceleration. Value should be from 0 to 65535.	microsteps/ta*** (Default value: 40000)
“wcr” (0x776372)	4 bytes	0xAA or 0x01	Set motor drive current. Value should be from 50 to 800 mA.	Higher current will generate more heat. (Default value: 350 mA)
“hcr” (0x686372)	4 bytes	0xAA or 0x01	Set motor hold in position current when motor is idle. Value should be from 50 to 800 mA.	Higher current will generate more heat. (Default value: 100 mA)
“sn ” (0x736E20)	<= 17 bytes (char)	0xAA or 0x01	Write maximum of 17-character long name to a device.	
“clr” (0x636C72)	none	0xAA or 0x01	Clear device calibration info and set it to default values.	
“sav” (0x736176)	**Data structure same as in “cd ” command**	0xAA or 0x01	Saves all parameters for the motor.	For sending - same data structure as in “cd ” command.

* If command returns data, in front of data will always be “0xAA (OK)” and data length (2 bytes). At the end it should have 2 byte CRC.

** “t” – time equivalent for speed. t = 1.39810.

*** “ta” – time equivalent for acceleration and deceleration. ta = 0.01527.

7. Main Dimensions

7.1. PowerXP

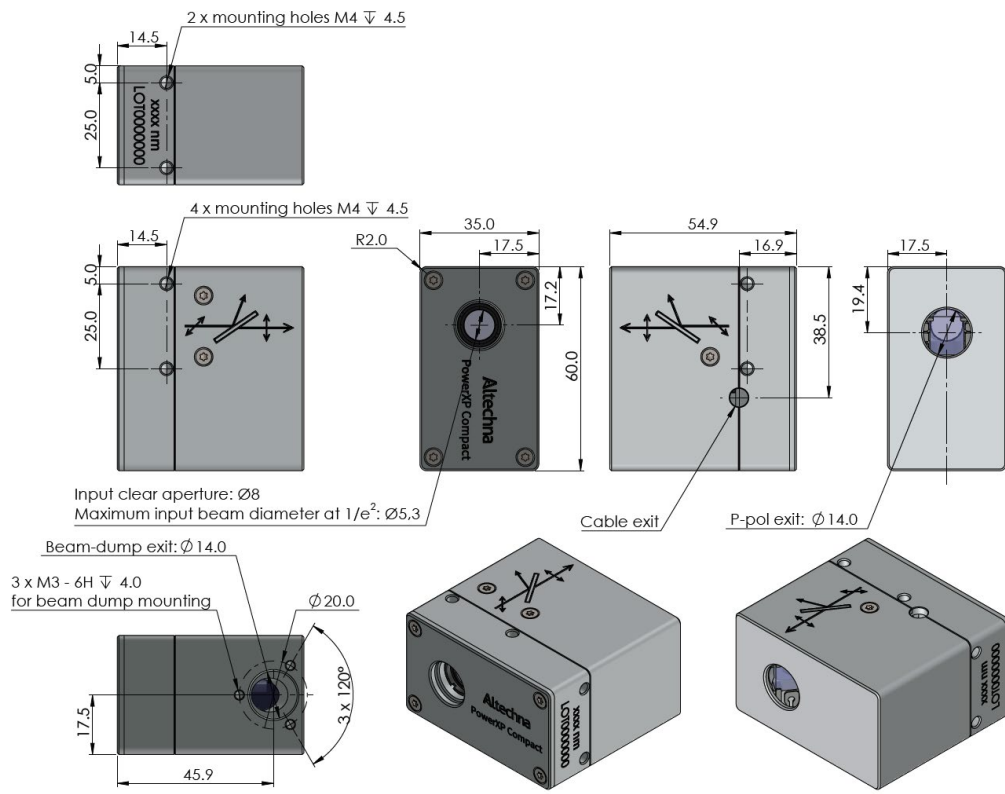


Figure 13. Main dimensions and mounting options of PowerXP.

7.2. PowerXP Controller

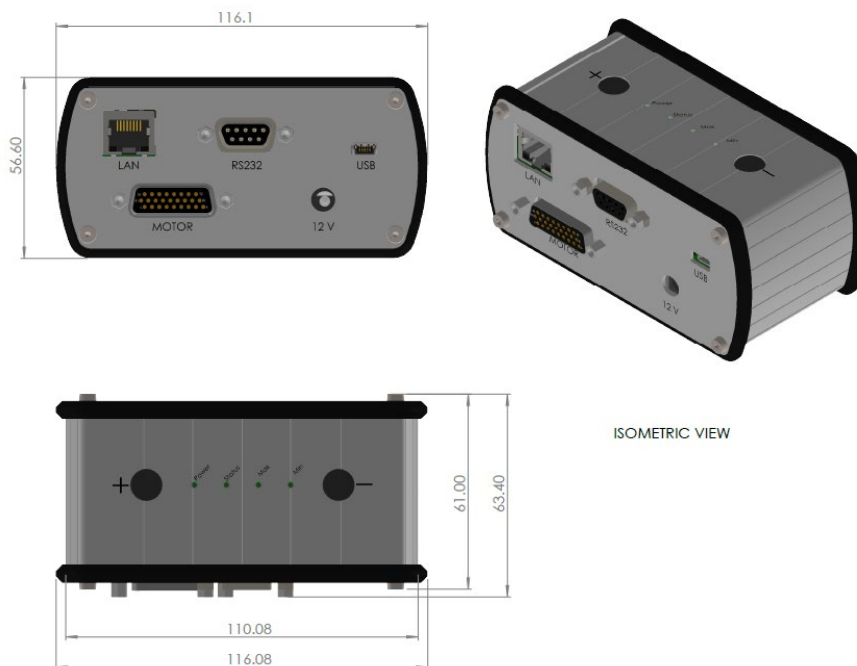


Figure 14. Main dimensions and mounting options of PowerXP Controller.

Tel. +370 5 272 5738
Fax +370 5 272 3704
info@altechna.com

Savanoriu ave. 176B
Vilnius, Lithuania

www.altechna.com