Instruction Manual
Digital Piezo Amplifier
NV200/D NET

Please read carefully before switching on the power! Please see safety instructions for using piezoelectric actuators and power supplies!

(€

24.05.2023

Manual revision: 5.1

Table of contents

1 Introduction

This manual describes the digital piezo amplifier series NV200/D NET from piezosystem jena. You will also find additional information regarding piezoelectric products.

Definition: All systems from piezosystem jena such as electronics, actuators, and optical systems are called "units".

If you have any problems, please contact the manufacturer of the system: piezosystem jena, Stockholmer Str. 12, 07747 Jena. Phone: +49 36 41 66 88-0

2 Certification of piezosystem jena

The company piezosystem jena GmbH has worked according to a DIN EN ISO 9001 certified quality management system since 1999. Its effectiveness is verified and proven by periodic audits by the TÜV.

> This instruction manual includes important information for using piezo actuators and electronics. Please take the time to read this information. Piezo positioning systems are mechanical systems that offer the highest precision. Correct handling guarantees that this precision will be maintained over a long period of time.

3 Declaration of conformity

The CE Declaration of Conformity is available on our website www.piezosystem.com.

4   Purchased part package

Please check the completeness of the delivery after receiving the shipment:

- piezo amplifier NV200/D NET
- wide range power supply 24 VDC

For optional needed drivers or available software see our homepage.

5   Instructions for using piezo electrical elements and power supplies

- Piezoelectric actuators from piezosystem jena are controlled by voltages up to 180 V. These values can be quite hazardous. Therefore, read the installation instructions carefully and ensure that only authorized personnel handle the power supply.
- After transportation, piezoelectric actuators should be allowed to adapt to room temperature for approximately 2 hours before being switched on.
- Piezoelectric actuators are made from ceramic materials with and without metallic casings. The piezo-ceramic is a relatively brittle material. This should be noted when handling piezoelectric actuators. All piezo elements are sensitive to bending or shock forces.
- Due to the piezoelectric effect, piezo actuators can generate electrical charges by changing the mechanical load or the temperature, or by actions such as the ones described above.
- Piezoelectric actuators are able to work under high compressive forces. Only actuators with a pre-load can be used under tensile loads (these tensile forces must be less than the pre-load given in the data sheet). Please note that acceleration of the ceramic material (e.g., caused by fall down, discharging, or high dynamic application) will occur.
- After excitation of the actuators by a voltage in the upper control range, the ceramic will move and generate an opposite high voltage after disconnection.
- Heating of the ceramic material will occur during dynamic operation and is caused by structure conditional loss processes. This may cause failure if the temperature exceeds specified values cited below. With increasing temperature up to the Curie temperature TC, (usual values approx. 140 °C – 250 °C) the piezoelectric effect disappears. We recommend working in temperatures up to TC/2 (normally up to 80 °C).
- Piezoelectric actuators, such as stacks or other devices, work electrically as a capacitor. These elements are able to store electrical energy over a long period of time (up to some days) and the stored energy may be dangerous.
- If the actuator remains connected to the drive electronics, it will be unloaded within a second after shutdown and quickly reaches harmless voltage values.
- Piezo actuators can only generate voltages by warming or cooling (caused by the longitudinal change). The discharge potential should not be ignored due to the inner capacitance. This effect is insignificant at usual room temperature.
- Piezo actuators from piezosystem jena are adjusted and glued. Any opening of the unit will cause misalignment or possible malfunction and will result in the loss of the guarantee.
- Please only use original parts from piezosystem jena.

Please contact piezosystem jena or your local representative if there are any problems with your actuator or power supply.

Caution! Shock forces may damage the built-in ceramic elements. Please avoid such forces, and handle the units with care, otherwise the guarantee will be lost.

6 Safety instructions

6.1 Icons

⚠ RISK OF ELECTRIC SHOCK! Indicates that a risk of electric shock is present and the associated warning should be observed.

⚠ CAUTION! REFER TO OPERATOR´S MANUAL – Refer to your operator's manual for additional information, such as important operating and maintenance instructions.

RISK OF ELECTRIC SHOCK! ⚠

- Do not open the units! There are no user serviceable parts inside and opening or removing covers may expose you to dangerous shock hazards or other risks. Refer all servicing to qualified service personnel.
- Do not spill any liquids into the cabinet or use the units near water.

CAUTION! ⚠

- Allow adequate ventilation around the units so that heat can properly dissipate. Do not block ventilated openings or place the units near a radiator, oven, or other heat sources. Do not put anything on top of the units except those that are designed for that purpose (e.g. actuators).
- Only work with the units in a clean and dry environment! Only specially prepared units (e.g. actuators) can work under other conditions!
- Please only use original parts from piezosystem jena. piezosystem jena does not give any warranty for damages or malfunction caused by additional parts not supplied by piezosystem jena. Additional cables or connectors will change the calibration and other specified data. This can change the specified properties of the units and cause them to malfunction.
- Piezo elements are sensitive systems capable of the highest positioning accuracy. They will demonstrate their excellent properties only if they are handled correctly! Please mount them properly at the special mounting points.

Immediately unplug your unit from the wall outlet and refer servicing to qualified service personnel under the following conditions:

- when the cords or plugs are damaged
- if liquid has been spilled or objects have fallen into the unit
- if the unit has been exposed to rain or water
- if the unit has been dropped or the housing is damaged

5

6.2 Installation & power supply

RISK OF ELECTRIC SHOCK! ⚡

- Do not insert or unplug the power plug with wet hands, as this may result in electrical shock.
- Do not install in rooms where inflammable substances are stored. If flammable substances come into contact with electrical parts inside, it could result in fire or electrical shock.
- Do not damage or modify the power cord. Also, do not place heavy objects on the power cord, or pull on or excessively bend it, as this could cause electrical damage and result in a fire or electrical shock.
- Always grasp the plug portion when unplugging the power cord. Pulling on the power cord may expose or snap the core wire, or otherwise damage the power cord. If the cord is damaged, this could cause an electricity leak and result in a fire or electrical shock.

CAUTION! ⚠

- Do not use accessories other than the ones provided (e.g. power cord). Only plug the power cord into grounded power equipment and sockets.
- Do not place heavy objects on any cables (e.g. power cords, sensor cables, actuator cables, optical cables).
- Do not block ventilated openings or place the units near a radiator, oven, or other heat sources.
- Plug in the power cord completely so that it cannot loosen inadvertently.
- Leave sufficient space around the power plug so that it can be unplugged easily. If objects are placed around the power plug, you will be unable to unplug it in an emergency.
- Install the system so that the on/off-switch is easily accessible at all times.
- The power plug is the cut-off point to the main power supply.

6.3 Operation

RISK OF ELECTRIC SHOCK! ⚡

- Do not open the units! There are no user serviceable parts inside and opening or removing covers may expose you to dangerous shock hazards or other risks. Refer all servicing to qualified service personnel.
- Do not spill inflammable substances inside the voltage amplifier. If these items come into contact with an electrical component inside the voltage amplifier, this may result in a fire or electrical shock.

CAUTION! ⚠

- If the voltage amplifier emits smoke, high heat, or unusual smells, immediately turn off the power switch and unplug the power plug from the outlet. Then contact our technical service.

6

6.4   Maintenance and inspection

CAUTION! ⚠️

- Before cleaning the exterior box of the voltage amplifier, turn off the power switch and unplug the power plug. Failure to do so may result in a fire or electrical shock.
- Clean the exterior box using a damp cloth that has been firmly wrung-out. Do not use alcohols, benzene, paint thinner or other inflammable substances. If flammable substances come into contact with an electrical component inside the voltage amplifier, this may result in a fire or electrical shock.

6.5   Environmental conditions

The amplifier can be used:

- indoors only
- at an altitude of up to 2000 m
- at a temperature between: 5 ... 35 °C
- at a relative humidity between: 5 ... 95 % (non-condensing)

The recommended environmental conditions:

- indoors only
- at an altitude of up to 2000 m
- at a temperature between: 20 ... 22 °C
- at a relative humidity between: 5 ... 80 % (non-condensing)

7   Instructions for checking the function of the system / quick start

When you open the package, please check to make sure all the necessary parts are included (see packing list) and nothing is damaged. Check the electronics and the actuator for any visible damage:

- The top and bottom plates of the actuator (if it does not have another shape) should be parallel to each other and not have any scratches.
- If there is any damage to the system, please contact our local representative immediately!
- If the packaging material is damaged, please confirm this with the shipping company.
- Connect the piezo actuator by using the D-SUB 15pin connector. Be sure the cables are connected properly to the electronics.
- Connect the device to PC with a USB cable.
- Connect the power supply (24 VDC).
- Normally, the PC installs necessary drivers automatically.
- After initialization, the LED lights green (open loop mode) or orange (closed loop mode).
- In case of irregularities, please read the chapter on troubleshooting.
- The function check is complete.

8   How to operate the digital amplifier NV200/D NET

8.1   Introduction

The digital amplifier NV200/D NET is designed to control the D-actuators from piezosystem jena. It can be used with actuators with or without integrated position sensor. Actuators with integrated position sensor allows positioning without creeping or hysteresis by digital PID loop control. The amplifier can be controlled by USB, Ethernet, SPI or analog voltage input and reaches a resolution of 16 bits. piezosystem jena supports driver for Windows® operating systems.

The integrated digital loop controller comes as PID controller, additionally a slew rate limiter, a low pass filter of 4th order for set point filtering and a notch filter are available.

The digital amplifier **NV200/D NET** allows the interchangeability from D-actuators and amplifier. All needed information is stored on the ID-chip, located in actuator's connector, and will be read automatically by the piezo controller after system start.

The actuator is connected to the **NV200/D NET** using the "PIEZO" plug. After switch on, the device runs a self-test procedure, while the indicator LED is lightning red. This takes about 3 s. During this step the controller reads the system specific parameters from the actuators ID-chip and sets up the controller. If no actuator is connected, the LED stays red.

After the self-test, the actuator moves to its initial position. In case of open loop mode to the desired voltage, in case of closed loop mode it moves to the set position.

In OL mode, the actuator is directly controlled by a voltage between -20 V and +180 V, however it will be automatically limited to the admissible voltage range of the connected actuator. The reachable stroke (pos. 1 to pos. 4) in unloaded condition can be seen on the actuator's calibration protocol. The stroke vs. voltage relationship is non-linear, shows hysteresis and creep and is affected by the actuators load condition. The stroke decreases by higher stiffness loads. To reach a defined position, actuators with integrated position sensors are used. They will have a smaller stroke in CL mode, the remaining stroke, compared to OL mode is used as a reserve to reach the end positions (pos. 2 and pos. 3) also under load.
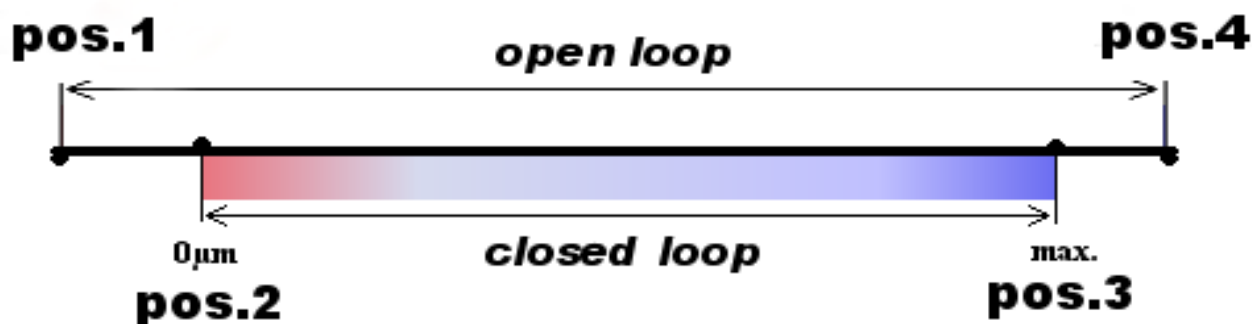
8.2 Technical data

Input voltage	$24\,V_{DC}$
Power supply connector	low voltage socket with 2.1 mm pin
Input current	max. 2 A (rms) / 7 A (peak) @ 24 V, idle current 110 mA
Power consumption	max. 48 W
Output voltages	-10 V ... +180 V or -20V … +130 V (automatically adapted to the actuator)
Output current	200 mA (continuous) 400 mA (peak for 1.2 ms) NanoX®-mode: 2 x 100 mA (continuous) 2 x 200 mA (peak for 1.2 ms)
Output voltage noise	$0.7\,mV_{RMS}$ @ 500 Hz bandwidth
Actuator/measurement connector "PIEZO":	15 pin D-Sub plug
Safety functions:	short circuit proof, over temperature protection
Display (LED):	green / orange / red
Interfaces	USB, Ethernet
Interface connectors	USB-C, RJ45
Cooling	passive
Dimensions (L * W * H):	165 x 120 x 65 mm³
Weight:	0.5 kg

Depending on the connected actuators voltage range and working principle, the following frequency response is feasible within the amplifier's current limitation.

Maximum Frequency Response for 2x100 mA (NanoX)

Maximum Frequency Response for 200 mA
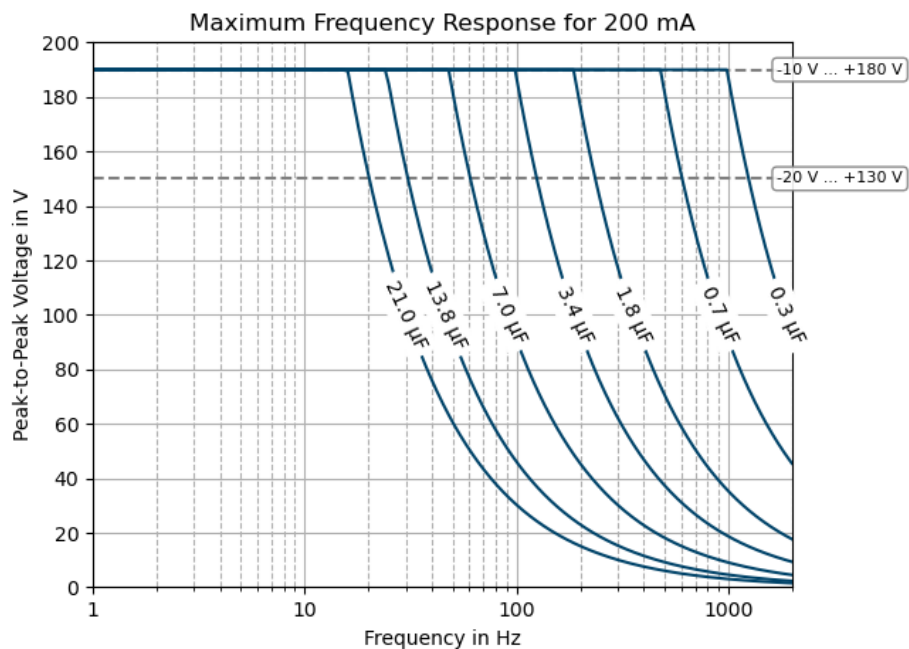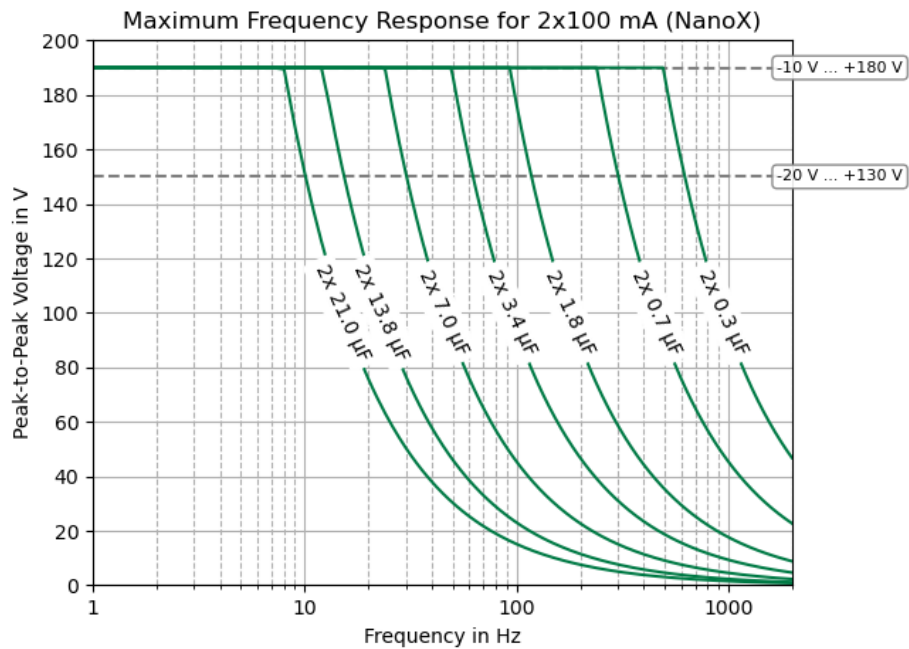
8.3   Connectors and Pin Assignment

8.3.1   Electrical Interface Control Drawing

I/O connector
(15 pin D-Sub HD jack)

Ethernet connector
(RJ45)

USB connector
(USB Type-C)

Piezo actuator connector
(15 pin D-Sub plug)

Status LED

Power supply connector
(low voltage socket
2.1mm x 6.3mm)
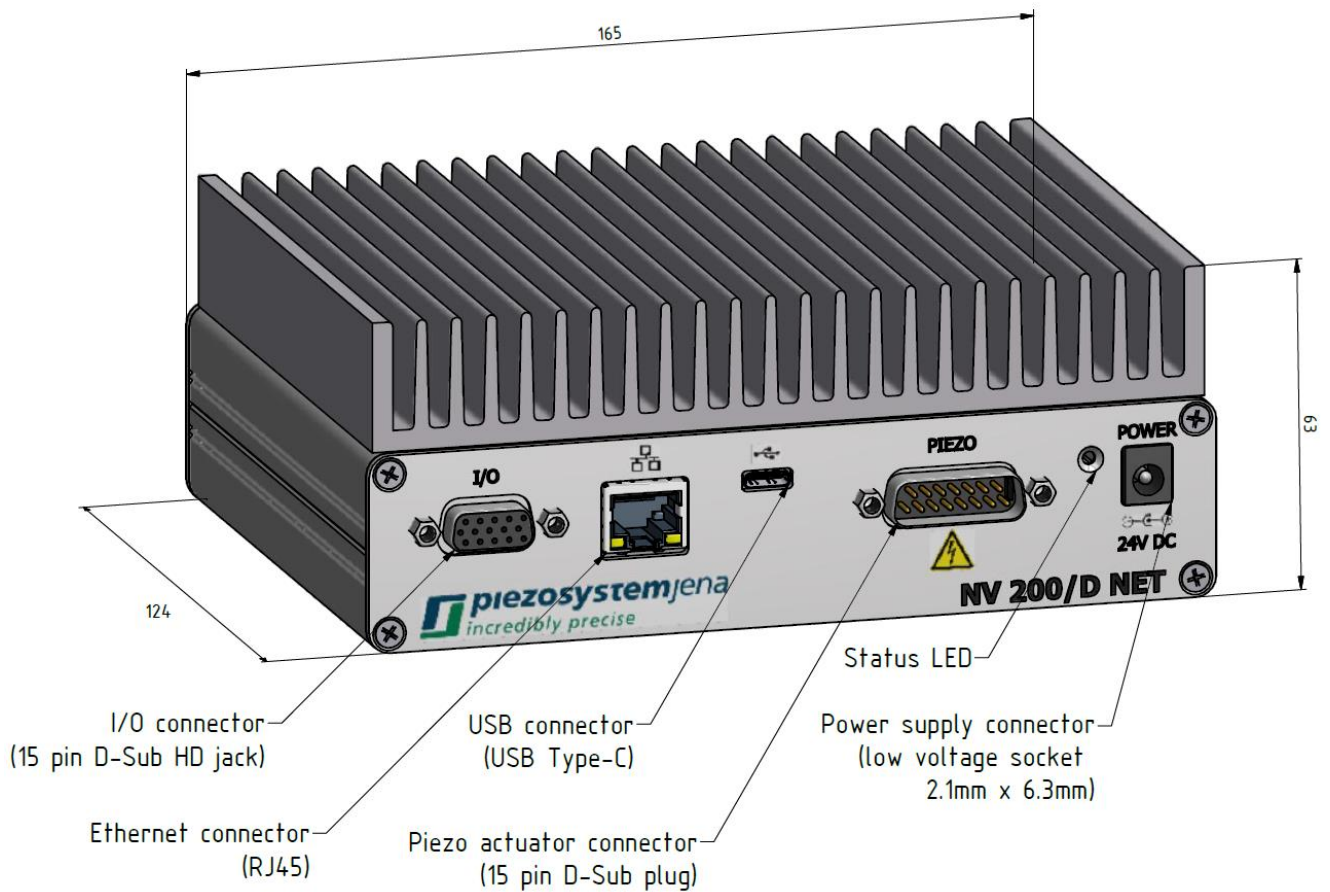
8.3.2 Piezo actuator connector

Type: 15 pin D-SUB plug

Pin	Label	Description
1, 2, 11	AGND	Analog ground
3	+15 V*	Operating voltage measurement +15 V
4, 14	GND	Digital ground
5	SDA	I²C-Bus SDA
6	3.3 V_{eeprom}*	Supply voltage ID-Chip
7	Vout2	Second voltage for NanoX® actuator +180 V … -20 V
8	Vout	Actuator voltage -20 V ... +180 V
9	+MEAS	Position signal -8 V ... +8 V
10	-15 V*	Supply voltage measurement -15 V
12	SCL	I²C-Bus SCL
13	DETECT	Actuator detection
15	Piezo ground	Actuator ground

[*] Not for external use

The actuator is connected to this plug. The actuator voltage goes to the piezo, the position signal from the measurement system goes to the DSP, and the integrated actuator information is stored on an ID-chip which goes to the main controller. It is important that the locking screws are screwed into the front panel. The dangerous voltages will be switched on only after connecting the actuator. When disconnecting the actuator, the dangerous voltages will be switched off by a safety relay.

8.3.3 I/O connector

Type: 15 pin D-Sub HD jack

Pin	Label	Description
1	MOD	Analog modulation input (0...10V), (Impedance 1 MΩ)
2	MON2	Not used
3	TRG in	Trigger input (TTL, 0 / 3,3V…5V)
4	SPI: CLK	SPI clock (3,3V logic, slave input)
5	DGND	Digital ground (for SPI & Trigger)
6	AGND	Analog ground (for MOD & MON)
7	DGND	Digital ground (for SPI & Trigger)
8	DGND	Digital ground (for SPI & Trigger)
9	SPI: MOSI	MOSI (3,3V logic, slave input)
10	TxD	Not used
11	MON	Analog monitor output (0 ...5V)
12	TRG out	Trigger output (TTL 3,3V)
13	SPI: /CS	Chip select (3,3V logic, low active, slave input)
14	SPI: MISO	MISO (3,3V logic, slave output)
15	RxD	Not used
shield wire	AGND	Analog ground

8.3.4 USB connector

The USB-C type connector supports USB 2.0 and allows the device to be controlled via a serial COM-Port.

8.3.5 Ethernet connector

The RJ45 Ethernet connector allows the device to be controlled via a Telnet connection.

8.3.6 Power supply connector

The power supply connector is a low voltage socket with a 2.1 mm pin. The supply voltage is 24 V.

8.4 Establishing communication

8.4.1 Serial communication via USB

The **NV200/D NET** can easily communicate with your PC with the use of a terminal program such as HTerm. A USB connection cable (included) is required to connect to your computer. The properties of the COM port are: 115200 baud, 8 bit, no parity, 1 stop bit, software handshake (XON / XOFF).
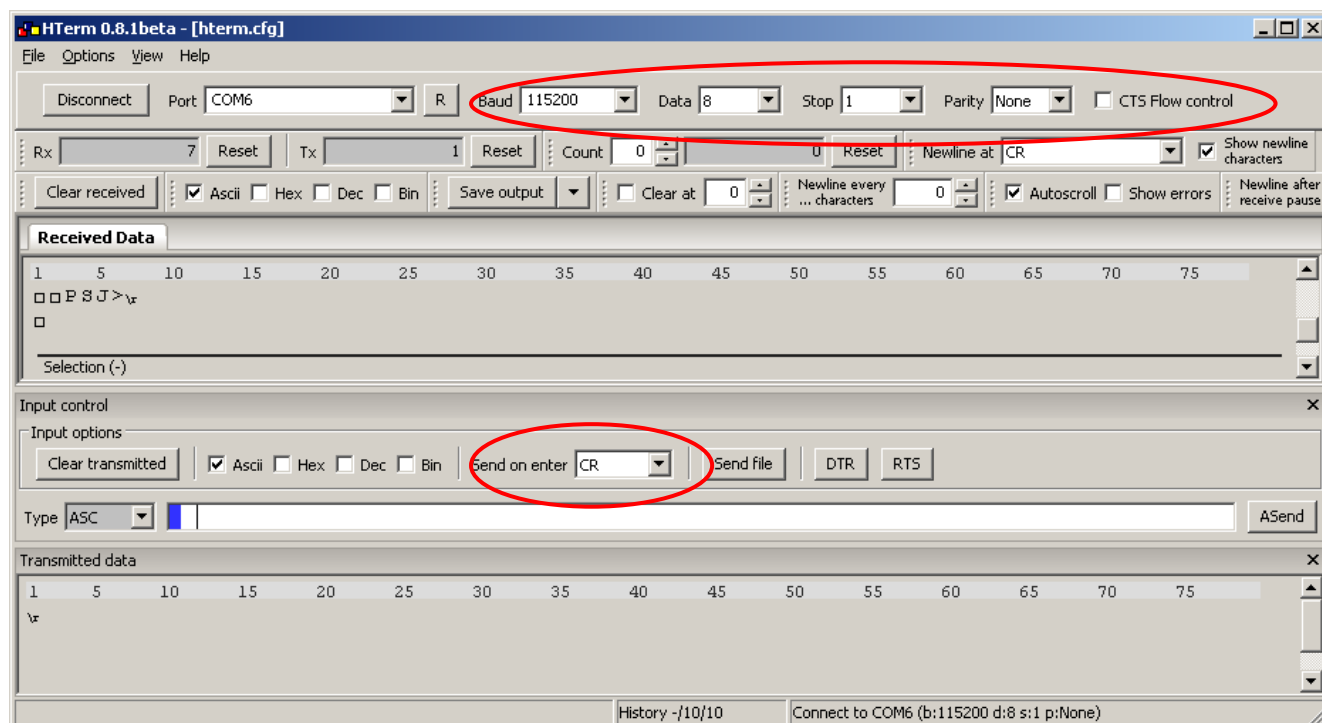

When the device is powered and connected, it answers after receiving "CR - carriage return" (Enter Key) with its prompt "NV200/D NET>"

8.4.2 Telnet communication via Ethernet

The communication with the **NV200/D NET** over Ethernet can be done with the Windows Telnet client. If not already enabled, this can be done via the "Turn Windows features on or off" functionality.

14
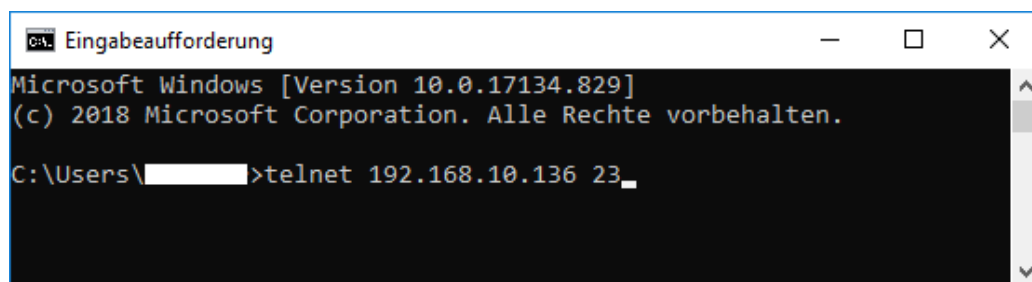
Alternatively, you can use other programs like Putty. After successful connection the full command set is available.
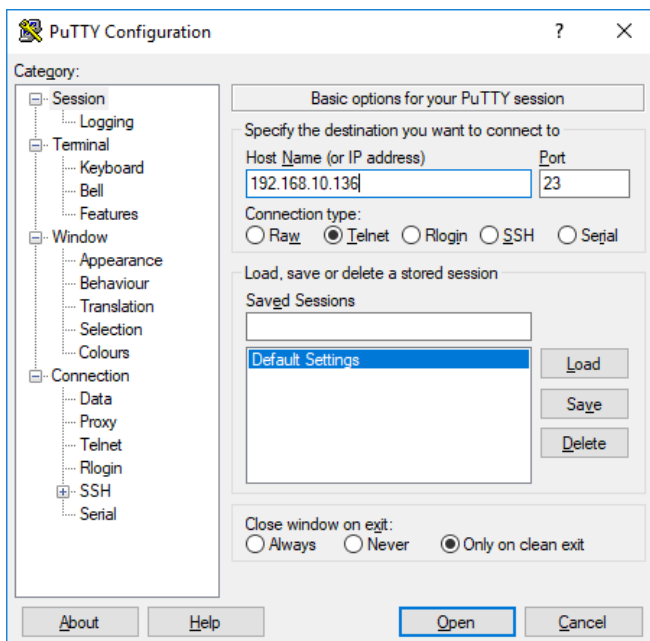
The Telnet connection can be established via the IP address of the controller and the correct port number. The default port number is 23.

By default, the IP address is set dynamically over DHCP. If necessary, it can be set static over the configuration. To get the IP address of the device ask you network administrator or use our tool "IP-Search" from the website.
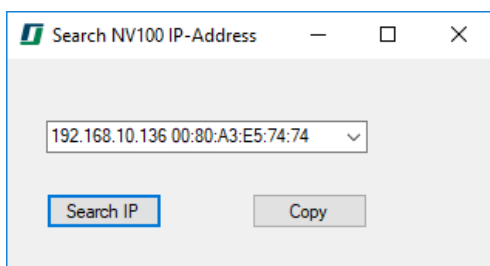
It is possible to use the host name of the device, if the "DHCP FQDN option" is enabled. By default, the host name is "C" + the last 6 digits of the MAC address. (Example: MAC=00:80:A3:E5:74:74 → host name: "CE57474"). The MAC address can be found on the type plate of the device.

8.4.3 Ethernet configuration

The network settings can be changed via Telnet connection (port: 9999) or via a web browser.

It is possible to change the IP address and the default port (23) for Telnet communication with the device. Under the item "Server" the IP address can be set to a static one or to DHCP. Under item "Channel 1" the port can be changed.

By accessing the IP in a web browser, the settings can be accessed graphically.
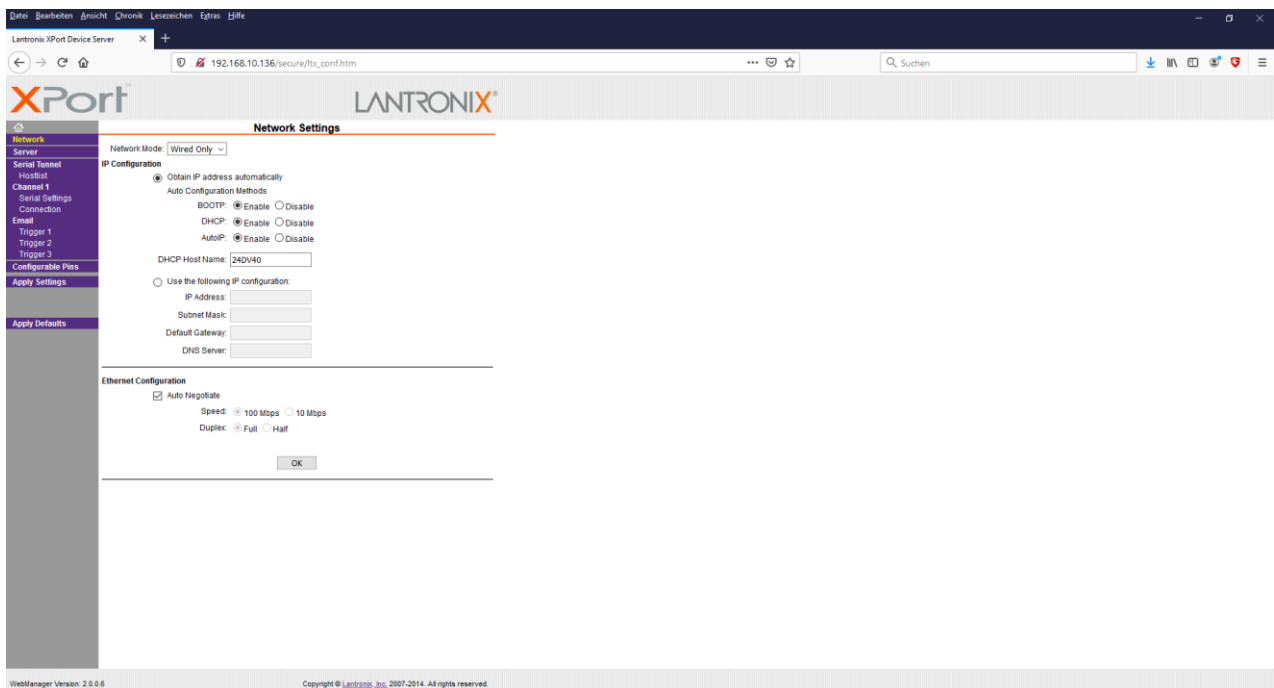
CAUTION: By default, the IP address is retrieved over DHCP. In networks without a DHCP server there is no access via Telnet possible without setting a static IP address.

8.4.4 SPI communication

Real-time communication can take place via the SPI interface. The device works as an SPI slave, i.e. the device receives setpoints from an external SPI master. The return value retrieved over the MISO channel can be set to different quantities (e.g. position values) via the USB or Ethernet interfaces. The SPI connection is exclusively to send setpoints to the device. The device has to be configured via the USB or Ethernet interfaces.

The interface requires the following configuration:

- 3.3 V logic voltage
- SPI clock: max. 6 MHz
- SPI mode: 3 (CPOL = 1, CPHA = 1)
- SPI CS: low active
- Word length: 16 bit, MSB First

Value range of the SPI-setpoint:

- in closed loop mode [0x0000 … 0xFFFE] corresponds to 0 … 100% of the closed loop travel of the connected actuator
- in open loop mode [0x0000 … 0xFFFE] corresponds to 0 … 100% of the admissible piezo voltage of the connected actuator
- The value [0xFFFF] is used for error detection and is not interpreted as a setpoint

Value range of the return value:

- The value selected by `spisrc` command is converted into a 16 bit integer value with the following scaling:

Configuration	Value	Scaling
`spisrc,0`	none	0x0000
`spisrc,1`	Measured position in closed loop	**posmin** ≙ 0x0000 **posmax** ≙ 0xFFFF
`spisrc,2`	Setpoint in open loop	**avmin** ≙ 0x0000 **avmax** ≙ 0xFFFF
`spisrc,3`	Piezo voltage (controller output)	**avmin** ≙ 0x2000 **avmax** ≙ 0xDFFF
`spisrc,4`	Position error	**-posmax** ≙ 0x2000 0 ≙ 0x8000 **+posmax** ≙ 0xE000
`spisrc,5`	abs(Position error)	0 ≙ 0x0000 **2*posmax** ≙ 0xFFFF
`spisrc,6`	Measured position in open loop	**posmin** ≙ 0x2000 **posmax** ≙ 0xDFFF
`spisrc,7`	Piezo current 1	-0.5A ≙ 0x0000 0 A ≙ 0x8000 +0.5A ≙ 0xFFFF
`spisrc,8`	Piezo current 2	-0.5A ≙ 0x0000 0 A ≙ 0x8000 +0.5A ≙ 0xFFFF
`spisrc,9`	Test value	0x5A5A

To synchronize the **NV200/D NET** PID controller loop execution to the SPI master the `spitrg` option can be used. In mode `spitrg,0` the internal 20 kHz clock is used to trigger the digital control loop interrupt. In mode `spitrg,1` the digital control loop interrupt is executed only when a new SPI setpoint is received. This allows precise synchronization of several NV200/D NET controllers. However, it is very important to make sure that SPI commands are sent with a 20 kHz update rate in order to get the right timing for the digital filters, since they are parameterized with respect to a constant sampling rate of 20 kHz.

8.5 Command list

All commands with one parameter to send are used in the following scheme:

`<command>,<value>\r`

Commands with indexing are used like this:

`<command>,<index>,<value>\r`

In case of read-only commands, commands without a value or to read back a value, the command is issued without a value:

`<command>\r`

Command	Description	Value range
General Commands		
`s`	Print full command list	-
`reset`	Hardware-reset of the controller	-
`fenable`	During controller power-up, cycle through the entire piezo voltage range once	0= disabled 1= enabled
`sinit`	Initial actuator position after power-up	0 … 100 %
`set`	Setpoint as position or voltage	In open loop: voltage in V In closed loop: position in µm or µrad Range is limited by the actuators stroke and voltage range
`setst`	Smooth Time Setpoint as position or voltage `setst,<value1>,<value2>`	In open loop: voltage in V In closed loop: position in µm or µrad <value1> = Setpoint <value2> = Jump Time Range is limited by the actuators stroke and voltage range
`meas`	For actuators with sensor: Position in actuator units (µm or mrad) For actuators without sensor: Piezo voltage in V	-
`imeas`	Measured piezo current	0 = piezo current 1 1 = piezo current 2
`ctrlmode`	Controller operation mode	0 = PID control 1 = ILC (identification mode) 2 = ILC (feedforward control mode) 3 = ILC (feedback control mode)
`temp`	Heat sink temperature	-
`stat`	Status register, see section 8.7 for details	-
`posmin`	Lower motion range limit	-
`posmax`	Upper motion range limit	-

avmin	Lower voltage range limit	-
avmax	Upper voltage range limit	-
modsrc	Signal source for setpoint	0 = set-Command via USB or Ethernet 1 = Analog In 2 = SPI 3 = Arbitrary waveform generator
monsrc	Source of data for analog output	0 = position (closed loop) 1 = setpoint 2 = piezo voltage (controller output) 3 = position error 4 = abs(position error) 5 = position (open loop) 6 = piezo current 1 7 = piezo current 2
PID and filters		
cl	PID open loop or closed loop mode	0 = open loop 1 = closed loop
sr	Slew rate limit	$0.0000008 ... 2000.0$ $\%/_{ms}$ (2000 = disabled)
kp	PID controller proportional term amplification	0 ... 10000 (0 = disabled)
ki	PID controller integral term amplification	0 ... 10000 (0 = disabled)
kd	PID controller differential term amplification	0 ... 10000 (0 = disabled)
tf	PID controller differential term	
pcf	PID controllers feed forward control amplification for position, velocity and acceleration: pcf,<pcf_x>,<pcf_v>,<pcf_a> The acceleration factor <pcf_a> is scaled by 1/1000000 internally, to reduce the number or required digits.	<pcf_x> = Factor for position <pcf_v> = Factor for velocity <pcf_a> = Factor for acceleration
setlpon	Enable or disable setpoint lowpass filter	0 = off 1 = on
setlpf	Setpoint low pass filter cut-off frequency	1 ... 10000 Hz
notchon	Enable or disable notch filter	0 = off 1 = on
notchf	Notch filter frequency	1 … 10000 Hz
notchb	Notch filter -3dB bandwidth	1 … 10000 Hz (max. 2 * notchf)
poslpon	Enable or disable measured position low pass filter	0 = off 1 = on
poslpf	Measured position low pass filter cut-off frequency	1 …10000 Hz
Arbitrary waveform generator		
grun	Start or stop the arbitrary waveform generator	0 = stop 1 = start
gsarb	Start index for arbitrary waveform generator output	0…1023
gearb	End index for arbitrary waveform generator output	0…1023
gcarb	Number of cycles to run	0 = infinitely 1…65535
goarb	Offset index when arbitrary waveform generator gets started	0…1023

giarb	Read current waveform generator index (e.g. after a stop command)	-
gtarb	Output sampling factor (<value> * 50µs)	1…65535
gbarb	Output values in the buffer in relative units	index: 0…1023 value: 0.0…100.0 (%)
gparb	Output values in the buffer in length units (µm or mrad)	index: 0…1023 value: **posmin** … **posmax**
gsave	Save the buffer in the EEPROM (End of process is acknowledged by CR LF)	-
gload	Loading the buffer from the EEPROM (End of process is acknowledged by CR LF)	-
Data recorder		
recsrc	Source of data to be stored in data recorder channel recsrc,<ch>,<src>	ch: 0 = buffer A 1 = buffer B src: 0 = piezo position in µm or mrad 1 = setpoint in µm or mrad 2 = piezo voltage in V (controller output, not amplifier output) 3 = position error 4 = abs(position error) 5 = piezo position in µm or mrad 6 = piezo current 1 in A 7 = piezo current 2 in A
recast	Autostart mode of the data recorder	0 = off 1 = start on **set**-command 2 = start on **grun**-command
recstr	Data recorder stride (store only every n^{th} value)	1….65535
reclen	Memory length for each of the two data recorder channels	0 …. 6144 0 = infinite loop over maximum length until recorder is stopped manually
recrun	Start or stop the data recorder	0 = stop 1 = start
recidx	Read current data recorder write index (e.g. after a stop command)	-
recout	Read the data recorder line by line: • ch: channel • Index: Distance to index 0 • Length: number of values (max. to reclen or rec_max) Response: recout, <ch>, <index>,<value>	ch: 0 = channel A 1 = channel B Index: 0… 6143 Length: 1… 6144
recoutf	Read the complete data recorder comma-seperated: Response: recoutf, <ch>, <value 1>,<value 2>,….,<value reclen>	ch: 0 = channel A 1 = channel B
Trigger In		
trgfkt	Function to run on trigger input	0 = none 1 = waveform generator start 2 = waveform generator step 3 = waveform generator sync 4 = ILC sync 5 = start data recorder

Trigger Out		
`trgedg`	Trigger on falling and/or rising edge	0 = off 1 = rising, not inverted 2 = falling, not inverted 3 = both, not inverted
`trgsrc`	Signal source for trigger out	0 = position 1 = setpoint (check: Example 8.12.2)
`trgss`	Trigger output start (lower position)	(**posmin**+0.001)...(**posmax**-0.001)
`trgse`	Trigger output stop (upper position)	(**posmin**+0.001)...(**posmax**-0.001)
`trgsi`	Trigger output step size	0.001…(**posmax**-0.001)
`trglen`	Trigger length in samples t=50µs*trglen (trglen=0→t=1µs)	0…255
SPI		
`spisrc`	SPI monitor/ Return value via MISO	0 = 0x0000 1 = position (closed loop) 2 = setpoint 3 = piezo voltage (controller output) 4 = position error 5 = abs(position error) 6 = position (open loop) 7 = piezo current 1 8 = piezo current 2 9 = test value (0x5A5A)
`spitrg`	Set the PID control loop interrupt source	0 = internal 1 = SPI
`spis`	Read last setpoint received via SPI	0 = setpoint as hex 1 = setpoint as decimal 2 = setpoint as stroke or voltage
ILC control		
`idata`	Read all relevant ILC parameters	-
`iemin`	Lower learning limit "emin"	0.0001 … 1.000 (default: 0.1)
`irho`	Learning rate "rho"	0.0001 … 1.000 (default: 0.1)
`in0`	Number of basic scanning n0 >= n1	2…65535
`in1`	Number of subsamples n1 < 1024 (=N1_max) n1 = 2^k	2,4,8,16,32,64,…,1024
`inx`	Number of frequency components to be learned nx < 128 (=NX_max) nx < ½ * n1	1…128
`iut`	Read the piezo voltage profile in time domain	Without index = output as array (up to `in1`) With index (up to `in1`) = output of the value at index
`iyt`	Read the measured position profile in time domain	Without index = output as array (up to `in1`) With index (up to `in1`) = output of the value at index
`ii1t`	Read the piezo current profile of amplifier channel 1 in time domain	Without index = output as array (up to `in1`) With index (up to `in1`) = output of the value at index
`ii2t`	Read the piezo current profile of amplifier channel 2 in time domain	Without index = output as array (up to `in1`) With index (up to `in1`) = output of the value at index

igc	Read of the learning function in the frequency domain	Without index = output as array (up to **in1**) With index (up to **in1**) = output of the value at index
iuc	Read the piezo voltage profile in frequency domain	Without index = output as array (up to **in1**) With index (up to **in1**) = output of the value at index
iwc	Desired position profile in frequency domain defined by real and imaginary part of Fourier coefficients	Set: iwc,<Index>,<real>,<imag> <Index> = 0… **inx** <real> = - infinite ... + infinite <imag> = - infinite ... + infinite Read: Without index = output as array (up to **in1**) With index (up to **in1**) = output of the value at index
iyc	Read the measured position profile in frequency domain	Without index = output as array (up to **in1**) With index (up to **in1**) = output of the value at index
igt	Read the correction type in frequency domain	0 = no learning 1 = learning based on offline identification 2 = learning based on online identification
isave	Save learned profiles in the actuator	-
iload	Load learned profiles from the actuator	-

8.6 Error messages

When the user sends wrong inputs to the device, the controller responds with an error message in the form "error,*x*". The error numbers x means:

Error Number	Description
1	Error not specified
2	Unknown command
3	Parameter missing
4	Admissible parameter range exceeded
5	Command's parameter count exceeded
6	Parameter is locked or read only
7	Underload
8	Overload
9	Parameter too low
10	Parameter too high

8.7   Status register

The status register is a 16bit register, in which each bit describes a certain setting of the amplifier or actuator. The decimal sum of all bits is the value of the status register. It can be requested by the `stat` command.

Bit	Description
0 (LSB)	0 = actuator not connected 1 = actuator connected
1,2	0, 0 ($\cong$ 0) = actuator without position sensor 0, 1 ($\cong$ 2) = strain gauge sensor 1, 0 ($\cong$ 4) = capacitive sensor
3	0 = open loop mode 1 = closed loop mode
4	0 = low pass filter off 1 = low pass filter on
5	0 = notch filter off 1 = notch filter on
6	reserved
7	0 = signal processing inactive 1 = signal processing active
8	0 = amplifier channels not bridged (single current) 1 = amplifier channels bridged (double current)
9	unused
10	0 = OK 1 = temperature too high
11	0 = OK 1 = actuator error or incompatible actuator type
12	0 = OK 1 = hardware error
13	0 = OK 1 = I²C-error
14	0 = OK 1 = lower control value limit reached*
15	0 = OK 1 = upper control value limit reached*

* If it is impossible to reach the setpoint in CL mode within 0.5 s and by means of the admissible piezo voltage a flag is set in the status register. This flag indicates too much load or a defect of the actuator. If the lower control value limit is reached and the actual position is still larger than the setpoint, Bit 14 is activated. If the upper control value limit is reached and the actual position is still smaller than the setpoint, Bit 15 is activated. As soon as the desired position is reached or a new setpoint is given, the flags will be cleared and the detection procedure starts again.

8.8 Feedback control modes

8.8.1 PID control

The following block diagram shows the functional groups of the controller in PID mode (`ctrlmode,0`):
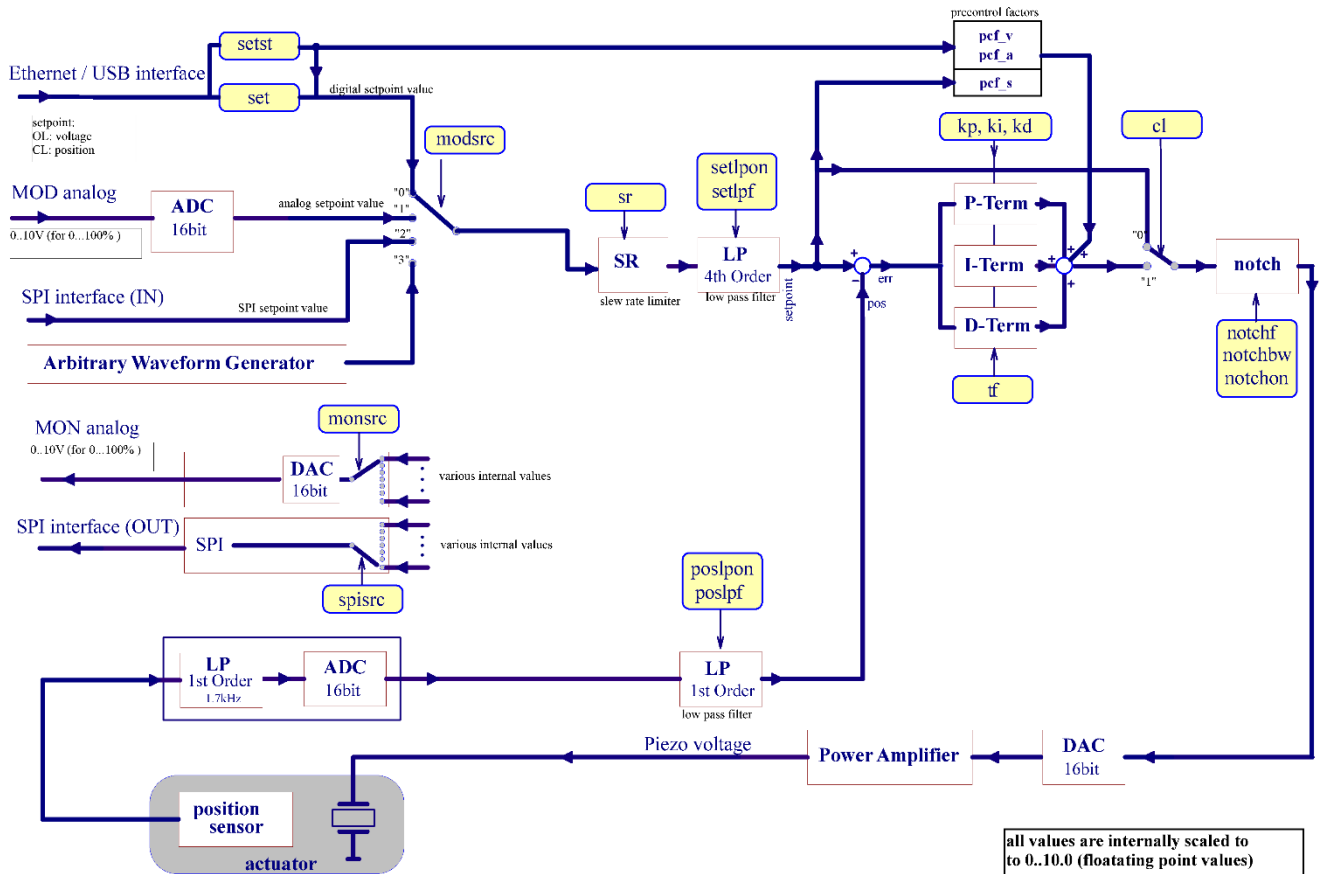
The PID-controller can work in two different modes: The open loop mode (OL) and the closed loop mode (CL). The setpoint can always be given by USB, Ethernet, arbitrary waveform generator or analog input. The slew rate, low pass and notch filters can always be used to avoid excitation of system resonances or to limit the setpoint dynamics in order to provide a trackable motion profile to the power amplifier and the actuator.

In open loop mode the PID-controller is bridged and the command input directly controls the amplifier, which can provide a voltage between -20 V and +180 V, however, limited to the admissible voltage range of the connected piezo actuator. The resulting piezo stroke then depends on the characteristic of the piezo actuator and is affected by piezo-typical creeping and hysteresis behavior.

In closed loop mode (CL), these effects will be compensated by the digital loop controller. The position sensor value can also be filtered with an adjustable 1st order low pass filter in order to reduce the measurement noise spectrum.

The PID-controller calculates the control error e between the (filtered and rate limited) reference input r and (filtered) position measurement $\hat{y}$. The PID algorithm with feed forward control is implemented in the form

$$u(s) = \underbrace{k_{\text{PCF}}\, r(s)}_{\text{Feed forward term}} + \underbrace{\left(k_{\text{P}} + k_{\text{I}}\frac{1}{s} + k_{\text{D}}\frac{s}{1 + sT_{\text{f}}}\right)e(s)}_{\text{PID term}}$$

with s being the Laplace variable. The PID algorithm yields u as the controller output prior to notch filtering.

The controller-specific parameters are stored in the actuator's ID-chip. The default parameters were determined in the **piezosystem jena** laboratory and ensure safe function of the actuator and will work for most applications. Depending on your application, additional load, setpoint trajectory and motion requirements, fine tuning of the control parameters will most likely improve your results.

When any actuator made by **piezosystem jena** is connected to the **NV200/D NET** its specific values are read from the actuator's ID-chip. The controller is then automatically configured with these values.

There are no PID parameters for actuators without a measurement system because the closed loop mode is disabled in this configuration.

To adapt the controller properties to your special application please start with the default values and use the commands `kp`, `ki`, `kd` and `tf`. In general, change the parameters in small steps and, depending on the actuator's reaction, slightly increase the step size. If the system begins to oscillate, switch to open loop mode (`cl,0`) immediately, then reset to stable values again.

For filtering the set point, a slew rate limiter is implemented, which can be changed via the `sr` command. The slew rate is defined in %/ms. For example, a value of 1 %/ms yields that a jump off the full motion range (100 %) is converted into a ramp with 100 ms duration. Independent from the slew rate value set in the controller, the limitation given by the high voltage amplifier cannot be exceeded, since the piezo voltage slew rate is limited by the amplifiers maximum output current and the actuators capacity.

25

8.8.2 ILC control

Iterative learning control is a method to iteratively determine the required piezo voltage profile to perfectly track a desired position profile. The only restriction is that the desired position profile is periodic. Therefore, it must be possible to define the position profile in the time window $t = [0; T)$ with T equal to the period duration.

The learning algorithm starts with selecting a suitable initial piezo voltage profile, which is output for a certain number of periods. At the same time the measured position is recorded for a complete period. Afterwards the actual control value calculation takes place. First, a complete period of the measured motion profile is transferred into the frequency domain by Fourier transformation. The difference between the desired position profile and the measured position profile in the frequency domain results in the control deviation. This is used to calculate an improved piezo voltage profile for the next iteration, which is the sum of the piezo profile of the current iteration and a correction component. The correction is calculated by filtering the control deviation in frequency domain with the inverse of the system's transfer function, also called learning function. The result is attenuated by the learning rate factor to increase the steady-state accuracy. Finally, the manipulated piezo voltage curve in the time domain for the next periods is determined by inverse Fourier transformation and will be output from now on. The system is then given a certain number of signal periods to react to the changed piezo voltage curve and to reach a steady state. Then the calculation starts again. This process is continued endlessly and, with a suitable choice of learning function and learning rate, leads to the convergence of the control deviation towards a minimum determined by disturbances, measurement noise and non-linearities.

The following block diagram shows the functional groups of the controller in ILC mode (`ctrlmode,3`):
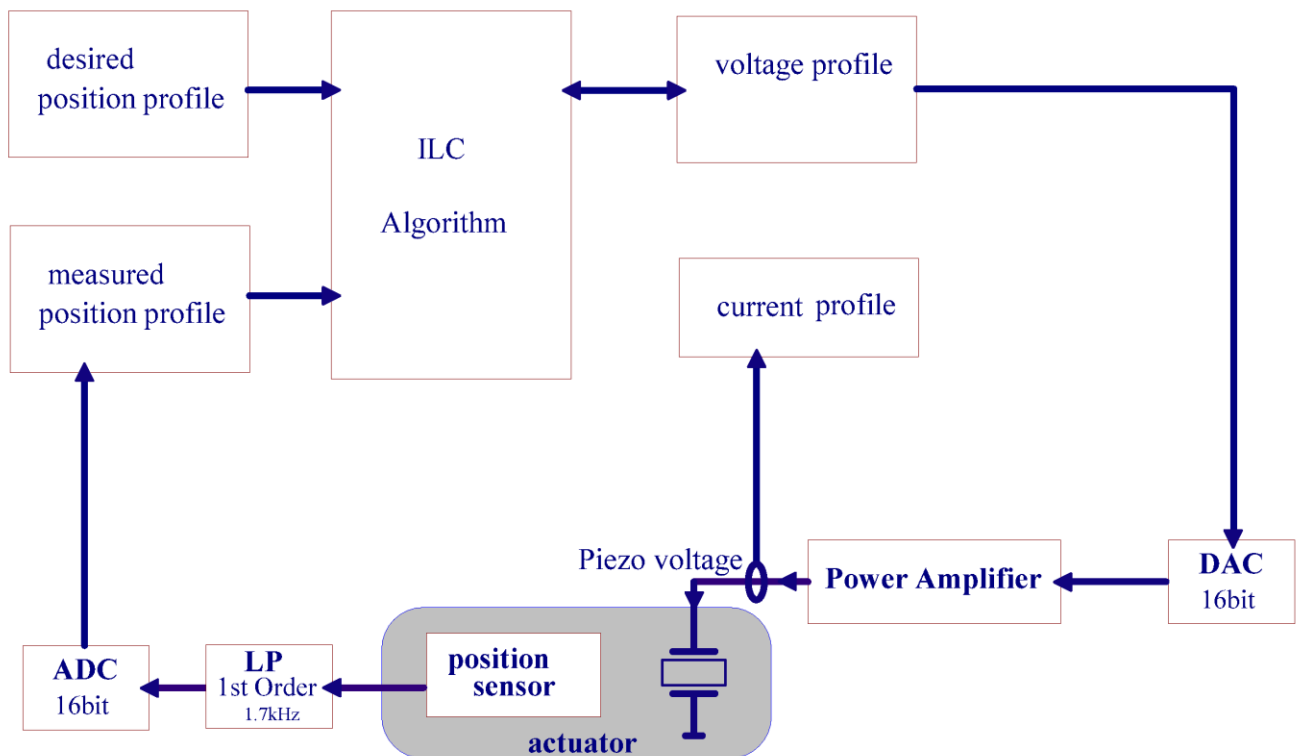
The following procedure is recommended in order to setup the ILC controller for a new signal. The required calculations can be easily scripted and automated in Python or similar or be guided by the PSJ controller software "NV200-Control". A basic Python script and the controller software are available on the device website.

First, the signal has to be defined in time domain. By means of a Fourier transformation, the Fourier coefficients of the signal need to be calculated. To be used in the controller, these coefficients have to be scaled to the NV200D/NET internal scaling. Finally, the scaled coefficients and some signal properties have to be sent to the controller. Therefore, the controller must be in `ctrlmode,0` which stops ILC mode. Next, the learning function has to be identified. To do so, the controller needs to be set into `ctrlmode,1`. After the identification, which takes several seconds and has to be done only once, the controller automatically switches to `ctrlmode,2` which is open loop ILC. To start the actual learning process the controller can now be set intro `ctrlmode,3` which is closed loop ILC mode. If a motion profile has been learnt to an acceptable quality, the whole configuration, including the learned voltage profile, can be stored in the actuators EEPROM, by the `isave` command. With the `iload` command, the stored values can be loaded, even after power cycling the controller.
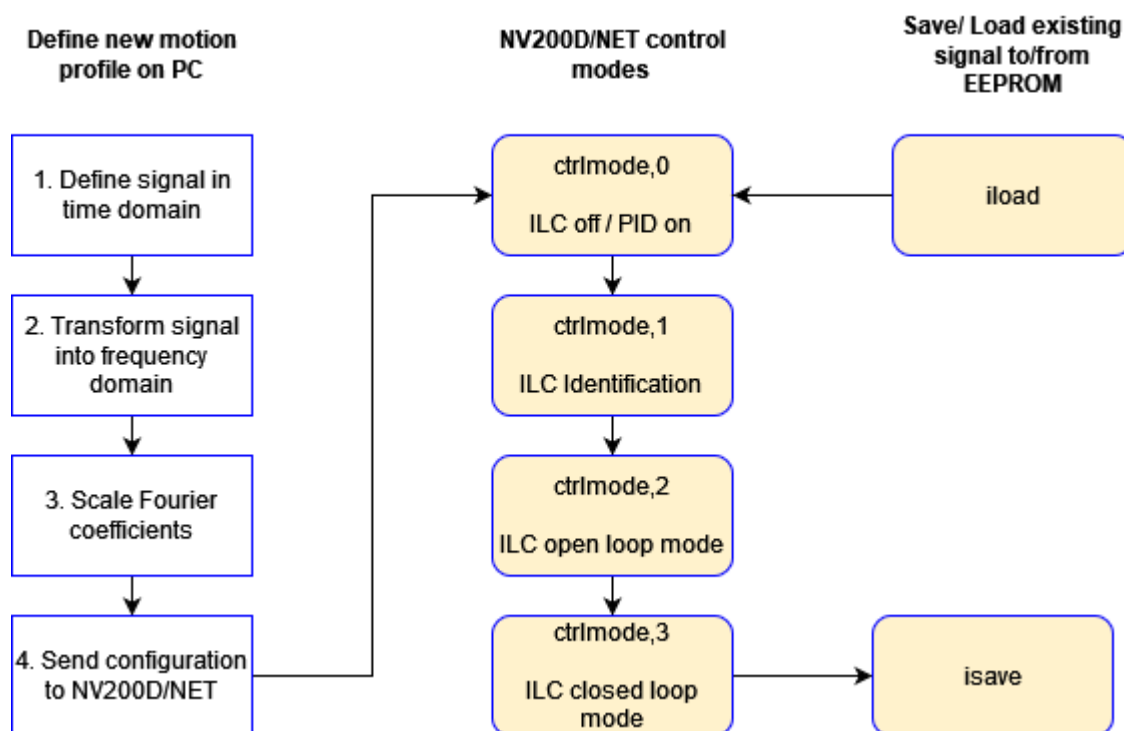
The ILC is tuning itself automatically by the identification step in order to adapt to the actuator's behavior. Therefore, no controller parameterization is required. However, the user can adjust the learning rate with the `irho` command. A value from zero to one determines the attenuation of the learning component in each iteration. Thus, a factor of zero means no learning and a factor of one means fastest learning. In order to have a good trade-off between learning speed, stability and noise suppression a learning rate in the range of 0.4 to 0.8 is recommended. Additionally, with the `iemin` command, a limit can be defined down to which each frequency component contributes to the learning process. If the difference between the desired signal's Fourier coefficient and the measured signal's Fourier coefficient is smaller than `iemin`, the controller will stop learning that frequency. This prevents the controller from learning random noise components.

27

8.9 Step smoothing

Basically, the `set` command can be used to set the actuator position in closed-loop operation. This generates a step change of the setpoint at the input of the PID controller. Unfiltered, however, this results in an unrealizable dynamic demand on the piezo actuator and the voltage amplifier. Furthermore, it usually results in a vibration excitation of the actuator. With the help of the integrated slewrate filter as well as a 4th order low pass filter, a jump by the `set` command, but also by voltage jumps at the analog modulation input can be filtered with respect to signal rate of change and frequency content.

Another possibility to specify a defined, physically realizable trajectory for the position change is offered by the command `setst`, which allows for the direct creation of a smoothened jump. The command generates a twofold continuously differentiable trajectory in the form of a cubic spline. This represents, with suitable parameterization, a setpoint trajectory that can be realized by the actuator and the amplifier. In conjunction with feedforward control, it is possible to perform dynamic position changes within a narrow tolerance band and, at the same time, to achieve steady-state accuracy and disturbance suppression by using a comparatively undynamically parameterized PID controller. When using the smoothened jumps by `setst`, the feedforward control can take into account both the influence of the required set position as well as that of the resulting velocity and acceleration profile on the actuator's voltage.

The command `setst,<pos>,<time>` ("set smooth time") can be used to set the target position `<pos>` and the jump duration `<time>` to be used for smoothing. In this way, position jumps can be performed in any amount of defined time, which can be useful for predictable timing of processes.

The jerk is defined as the third time derivative of the position profile. This has the consequence that the jump duration is a function of the jump distance. Jumps with a small distance thus occur in a shorter time than those with a longer distance. The relationship between jump duration T and jerk R is defined by

$$T = \sqrt[3]{\frac{32|\mathbf{pos}_1 - \mathbf{pos}_0|}{R}} \qquad .$$

By defining the jerk, a high degree of usage of the available system dynamics, limited by the current and voltage limitation of the amplifier, can be achieved for each jump distance. Furthermore, the limited jerk does also limit the shock or vibration excitation to the overall system.

When using the command `setst`, the slew rate as well as the input low-pass filter should be deactivated. In the block diagram of the controller (section 8.8.1), it becomes clear that the setpoint input would otherwise be unnecessarily delayed and filtered. Both filters can be deactivated via the commands `sr,2000` and `setlpon,0`.

In order to use the full dynamic potential of the step smoothing, the precontrol factors for position, velocity and acceleration must be set correctly with the `pcf` command. For the feedforward factor of the position a typical value of 0.8±0.1 can be selected for actuators calibrated with the PSJ standard. The velocity and acceleration feedforward factors are different for each actuator type. In addition, the load to be moved has a further influence on these factors. Their determination can be either empirical or derived from modelling of the overall system. It should be noted that the feedforward factors are calculated with the internal controller variables. In the controller, the setpoint position is represented as a float value, normalized to the value range [0.0 ... 10.0]. The minimum stroke of the actuator is mapped to the value 0, the maximum stroke to the value 10. The control variable is also normalized to the value range [0.0 ... 10.0]. The value 0 corresponds to the minimum actuator voltage of -20 V, the value 10 corresponds to the maximum actuator voltage of +130 V. Likewise, the velocity and acceleration resulting from the step rounding are calculated in the controller as unitless quantities from the derivatives of the normalized setpoint position. This must be taken into account when selecting the

velocity and acceleration feedforward factors. Furthermore, the acceleration factor is already scales by 1/1000000 internally in order to reduce the number of required digits. Typically the precontrol factors for velocity and acceleration are smaller than 1 by some magnitudes. The figure below shows the resulting magnitudes and should give an impression for the magnitude of the necessary feedforward factors.
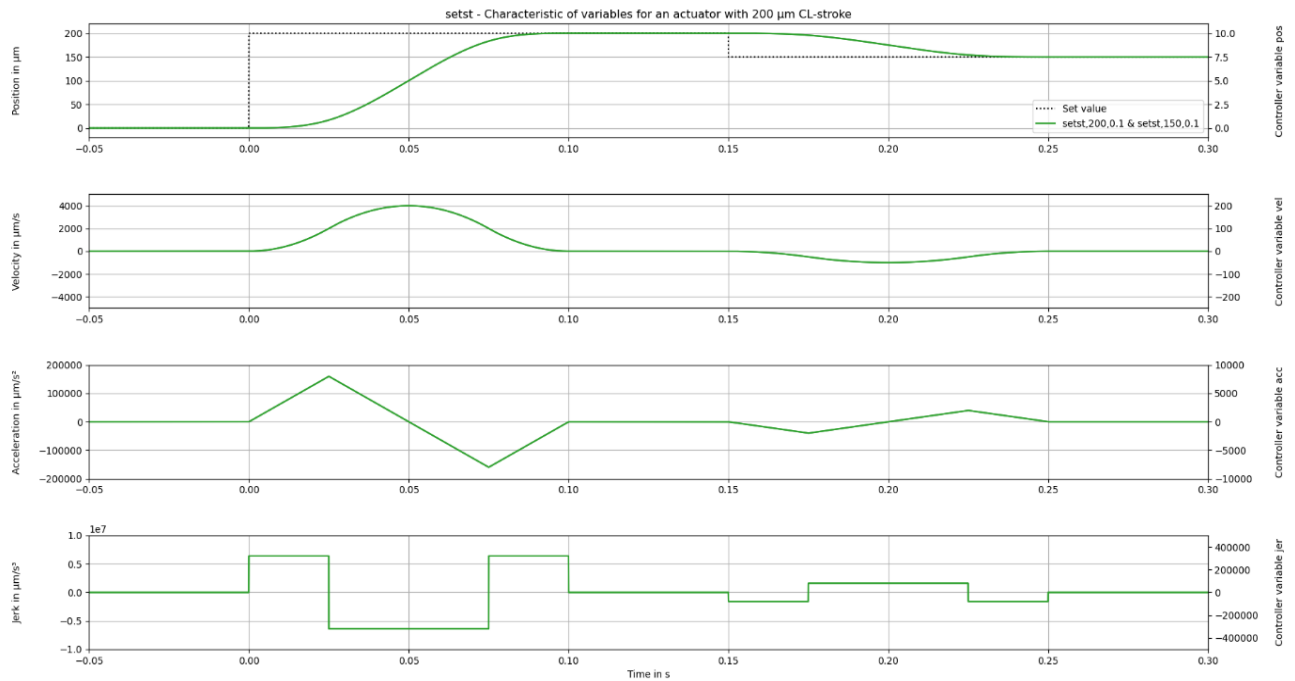
8.10 Arbitrary waveform generator

The arbitrary waveform generator can generate a single or repetitive setpoint signal. The curve shape can be freely defined by up to 1024 samples.
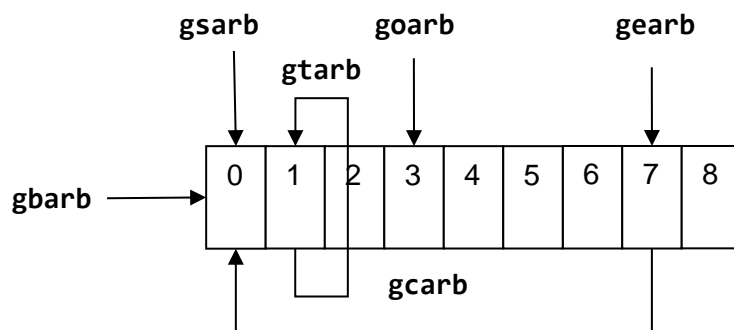
After the command **grun,1** the arbitrary waveform generator starts at the index defined by **goarb** and runs until the index defined by **gearb**. In all successive cycles, the arbitrary waveform generator starts at **gsarb**. This is repeated until the number of cycles reaches the value given by **gcarb**.

With **gtarb** the output sampling time of 50 µs can be extended. This means that the same index is output for the number of samples given by **gtarb**.

The arbitrary waveform generator can be stopped with **grun,0** at any time. The index of the sample at which the arbitrary waveform generator was stopped can be queried with the command **giarb**.

Example: Configure, fill and run the arbitrary waveform generator three cycles	
Command list	**Result**

`gsarb,0`
`gearb,4`
`goarb,0`
`gtarb,1`
`gcarb,3`
 	
`gparb,0,0.0`
`gparb,1,25.0`
`gparb,2,50.0`
`gparb,3,75.0`
`gparb,4,100.0`
 	
`modsrc,3`
`grun,1`	

8.11 Data recorder

The data recorder consists of two memory banks that are written to in parallel. In this way, two individual signals can be stored synchronously.

Example: A step input via the set command is used to trigger the data recorder and measure the piezo position and current of amplifier 1 for 25 ms.

Command list	Result
| ```
cl,1
modsrc,0
set,0

<pause to finish step>

recsrc,0,0
recsrc,1,6
reclen,500
recast,1
recstr,1
set,200

<pause to finish step>

recoutf,0
recoutf,1

<plot results>
``` |  |

8.12 Trigger

8.12.1 Trigger In

Various controller functions can be carried out with the trigger input. This can be used to synchronize the controller with several other *NV200/D NET* controllers or any other external device.

The trigger input uses a 5V logic and is activated by rising edges.

tgrfkt	Action on trigger pulse
0	No action
1	Start the arbitrary waveform generator (equal to **grun,1**)
2	Increment the arbitrary waveform generator index
3	Set the arbitrary waveform generator index to the start value (**goarb+gsarb**)
4	Synchronize the ILC counter/ Set the internal counter to the beginning of the setpoint curve
5	Start the data recorder (equal to **recrun,1**)

8.12.2 Trigger Out

In order to control external devices, the measured or desired (setpoint) position can be reported back via the trigger output. The input signal for the trigger can be chosen via `trgsrc` command.

The trigger signal is low-active, i.e. an H / L edge shows that a trigger point has been reached.

The signal range in which trigger events are generated is determined with `trgss` (lower limit) and `trgse` (upper limit). The interval between the trigger levels is determined with `trgsi`.

Triggering on a rising (`trgedg,1`), a falling (`trgedg,2`) or both edges (`trgedg,3`) is possible; the trigger is deactivated with `trgedg,0`.

The unit of measurement for `trgss`, `trgse`, `trgsi` is the actuator-specific unit in controlled operation (e.g. µm or mrad), the duration of the trigger pulses can be set as an integer multiple of 50µs (standard 1 * 50µs). It is important to ensure that trigger pulses do not overlap. If necessary, the duration of the pulses must be set to the minimum (`trglen,0`) or the travel speed of the actuator needs to be reduced.

Furthermore, it must be ensured that the motion range delimited by `trgss` and `trgse` is completely traversed (plus 0.2% of the total stroke); otherwise, no trigger points are generated as no change in direction of motion can be detected.

Example 8.12.2: An actuator has a closed loop stroke of 80 µm. With edge = rising (`trgedg,1`), `trgss,10` [µm] `ttrgse,30` [µm] `trgsi,5` [µm] the trigger events are generated at 10 µm, 15 µm, 20 µm, 25 µm & 30 µm. After reaching the position `trgss` (measured value> = 10µm) the trigger pulse is set and the next trigger point is calculated (15µm), the trigger is switched off after n * 50µs and it is waited for the position 15 µm to be reached. After reaching this position, the trigger is set again and the next position is calculated again (20 µm). After reaching 30 µm, the next trigger point is set to 10 µm, the trigger is only released when the position 10 µm is approached from "below" (i.e. from a position 10µm - (0.2% of 80µm)).

The edge detection works as follows:

- With a rising edge, the highest measured value (position) is always noted.
- If the measured value falls by more than 0.2% of the distance below the saved distance, a falling edge is recognized.
- Now the smallest measured value (position) is always noted.
- If the measured value now increases by more than 0.2% of the distance over the noted (smallest) distance, a rising edge is recognized again.

This functional principle requires that the area delimited by `trgss` and `trgse` is always passed beyond its limits (at least +/- 0.2% of the way).

8.13 Temperature measurement

To avoid damage to the electronics, the device is switched off at temperatures above 65 °C and can only be used again after power cycling or resetting by the `reset` command. To check the temperature of the heat sink in dynamic operation or under bad cooling conditions the temperature can be queried with the `temp` command.

9 Troubleshooting

Please check all cables and connections first if the system is not working properly.

Error	Possible cause and fix
LED lights red	No connection between actuator and amplifier. Check the actuator cable for damage.
LED blinks red	nanoX mode is not possible on this device.
Actuator oscillates in resonant frequency in closed loop	Unsuitable PID parameters. Try kp=0, ki=10, kd=0
No communication via USB possible	Unplug and re-plug the USB cable, restart or reconnect the terminal application. Check the device manager for a loaded USB driver. Check the interface settings: (115200 baud, 8 bits, 1 stop bit, no parity, software handshake (if applicable))
IP-Search did not find the device	Check the network connection cable Check the power supply (at powered device the led must light up) Check the firewall on PC and router (UDP 30718) In Ethernet configuration (see 8.4.3) under "6 Security" the setting "Port 77FE" must set to "enabled"
No connection via Telnet possible although the device is found in IP-Search	There is a connection with another program or from another computer to the device (only one connection at a time is possible via Telnet). The computer and the device are in different subnets. Set a static IP in the same subnet or change the subnet address on your computer. Check enabled firewalls TCP port 23 In Ethernet configuration (see 8.4.3) under "1 Channel 1" the "port" should be 23.