CONEX-PSD

Two-Axis Position & Power Sensing device

Newport™ **Controller Documentation**

V1.0.x

Table of Contents

Two-Axis Position & Power Sensing Device CONEX-PSD

1.0 System Overview

1.1 General Description

The CONEX-PSD consists in a two-axis position and power-sensing device connected to a controller. It provides a very compact and low-cost solution for measuring position and power of a laser spot.

Communication with the CONEX-PSD is achieved via an USB port (requires Windows™ operating system). A Windows™ based software enables basic motion. Advanced application programming is simplified by an ASCII command interface and a set of two letter mnemonic commands.

1.2 CONEX-PSD

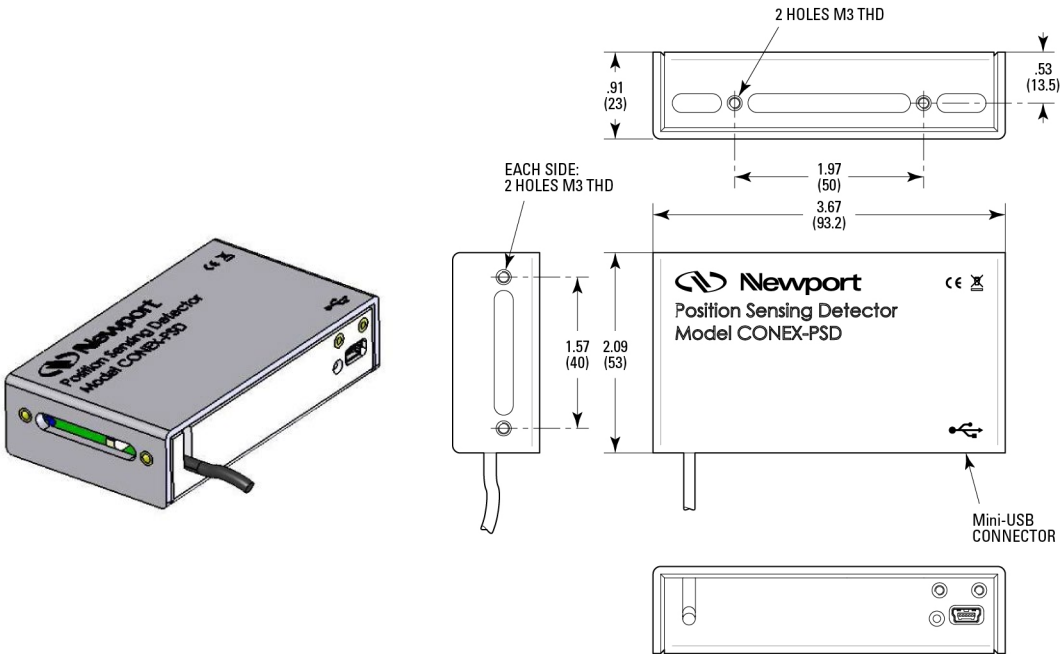
1.2.1 Contents of Delivery

- CONEX-PSD or CONEX-PSD10Ge
  Controller box with associated sensor (cable length: 0.55 m).
- CONEX-USB USB cable, 1.8 m length.
- CONEX-MOTION CD-ROM.

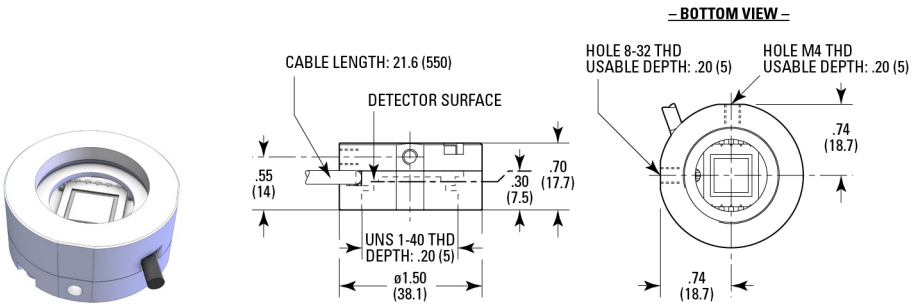
1.2.2 Specifications

General Description	2-axis position sensing device with power level indication
Position Capability	9 x 9 mm or 10 x 10 mm sensor with 12-bit resolution measurement
Power Display	0 to 100% of max power with 12 bits resolution measurement
Status display	Bi-color LED
Programming	20+ intuitive, 2-letter ASCII commands including position and power readings, offset and gain setting.
Computer interface	USB (requires Windows™ operating system)
Communication rate	50 Hz Max. (USB)
Internal safety feature	Watchdog timer
Consumption	+5 V (USB): < 0.5 A

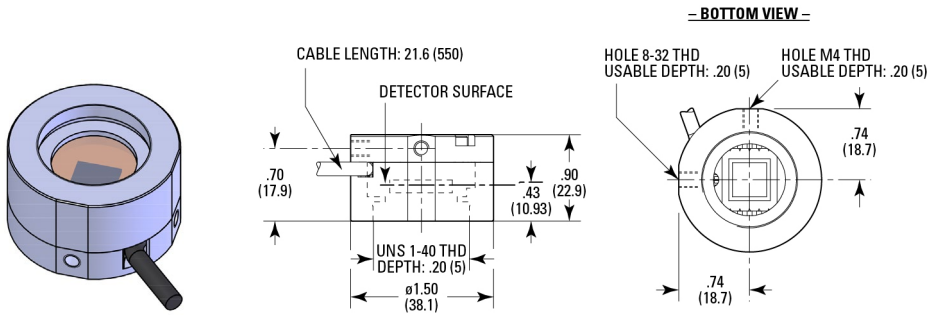
1.2.3 Controller Dimensions



1.2.4 Silicon Sensor Dimensions



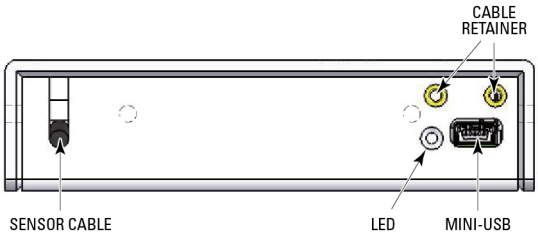
1.2.5 Germanium Sensor Dimensions



1.3 System Environmental Specifications

Operating temperature	5 °C to 40 °C
Operating humidity	20% to 85% relative humidity, non-condensing
Location	Indoor use only

1.4 Connector Identification



USB	Mini-USB connector
LED	Status LED
Sensor	Sensor cable entry
Cable retainer	2 x M3 threaded hole to attach cable retainer

1.5 USB Communication Settings

Communication parameters are preset in the CONEX-PSD controller and do not require any configuration:

Bits per second	921,600
Data bits	8
Parity	None
Stop bits	1
Flow control	None
Terminator	C_RL_F

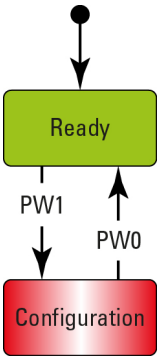
NOTE

USB communication allows Inputs reading and outputs setting up to 50 times/s.

2.0 Programming

2.1 State Diagram

For a consistent operation, the CONEX-PSD uses 2 different operation states: Ready and configuration. In each state, only specific commands are accepted by the CONEX-PSD. Therefore, it is important to understand the state diagram below and which commands and actions cause transition between the different states. Also see section 2.5 for command/state information:



LED display

CONFIGURATION: **SLOW BLINK RED**.

READY: **SOLID GREEN**.

When connecting the CONEX-PSD to power, the controller initializes (see section 2.2 Initialization). When the initialization is successful, the controller gets to the READY state. From the READY state, the controller can go to the CONFIGURATION state with the PW1 command. In CONFIGURATION stage, the CONEX-PSD allows changing configuration parameters like analog input offset and gain. The PW0 command saves all changes to the controller's memory and returns the controller back to the READY state.

2.2 Initialization

During initialization (< 1 sec) LED display is turned off.

For more information about system errors during initialization, refer to the TS command in section 2.5.

2.3 Command Syntax

The CONEX-PSD is a command driven controller. The general format of a command is a two-letter ASCII character preceded and followed by parameters specific to the command:

Command format

nn	AA	xx

nn — Optional or required controller address.

AA — Command name.

xx — Optional or required value or "**?**" to query current value.

Both, upper and lower case characters are accepted. Depending on the command, it can have an optional or required prefix (**nn**) for the controller address and/or a suffix (**xx**) value or a "**?**".

Blank spaces

Blanks are allowed and ignored in any position, including inside a numerical value. The following two commands are equivalent, but the first example might be confusing and uses more memory:

2P A1.43 6

2PA1.436

Decimal separator

A dot (".") is used as decimal separator for all numerical values.

Command terminator

Commands are executed as the command terminator C_RL_F (carriage-return line-feed, ASCII 13 and ASCII 10) is received. The controller will analyze the received string. If the command is valid and its parameters are in the specified range, it will be executed. Otherwise it will memorize an error.

After the execution of the command, all remaining characters in the input string, if any, will be ignored. In particular, it is not possible to concatenate several commands on a single string from the PC to the CONEX-PSD.

Each command will handle properly the memorization of related errors that can be accessed with the TE command. Please refer to the command set in section 2.5 for details.

2.4 Command Execution Time

The CONEX-PSD controller interprets commands continuously as received. The typical execution time for a "get position" (nGP?) is about 20 ms. Here, command execution time means the time from sending the command until receive of the answer.

2.5 Command Set

This section describes the supported two-letter ASCII commands used to configure and operate the CONEX-PSD. The general command format is:

Most commands can be used to set a value (in that case the command name is followed by the value "**xx**") or to query the current value (in that case the command name is followed by a "**?**"). When querying a value, the controller responds with the command it received followed by the queried value. For example, a 1IX0.1 sets the X offset of the controller #1 to 0.01. A 1IX? reads the response 1IX0.1.

Not every command can be executed in all states of the CONEX-PSD and some commands have different meaning in different states. It is therefore important to understand the state diagram of the controller, see section 2.1.

	Ready	Config.	Description
GP	●	●	Get X, Y positions and laser power
ID	□	○	Set/Get stage identifier
IS	□	○	Set/Get Set/Get offset on ADC input Sum (Si sensor) Not used (Ge sensor)
IX	□	○	Set/Get offset on ADC input X (Si sensor) Set/Get offsets on X position (Ge sensor)
IY	□	○	Set/Get offset on ADC input Y (Si sensor) Set/Get offsets on Y position. (Ge sensor)
LF	□	○	Set/Get low pass filter frequency
OF	□	○	Not used (Si sensor) Set/Get offset on ADC inputs 1 to 4 (Ge sensor)
PS	□	○	Set/Get gain on ADC input Summ (Si sensor) Not used (Ge sensor)
PX	□	○	Set/Get gain on ADC input X (Si sensor) Set/Get gain on X position (Ge sensor)
PY	□	○	Set/Get gain on ADC input Y (Si sensor) Set/Get gain on Y position. (Ge sensor)
PW	●	●	Enter/Leave CONFIGURATION state
RA	●	●	Get raw analog input values
RC	●	●	Get corrected analog input values
RS	●	●	Reset controller
RS##	□	□	Reset controller's address to 1
SA	–	○	Set/Get controller's address
TB	●	●	Get command error string
TE	●	●	Get last command error
TS	●	●	Get positioner error and controller state
VE	●	●	Get controller revision information

○ Changes configuration parameters. Those changes will be stored in the controller's memory with the PW1 command and remain available after switching off the controller.

□ Changes working parameters only. Those changes will get lost when switching off the controller.

● Accepted command.

– Write command not accepted (will return an error).

Command Command passed without preceding controller number applies to all controllers

GP — Get X, Y positions and laser power level

Usage	**Ready**	**Config.**	
	●	●	
Syntax	**xxGP** or **xxGP?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
Units	**xx**	—	None
Defaults	**xx**	Missing:	Error B.
		Out of range:	Error B.
		Floating point:	Error A.

Description This command returns the X, Y positions and the laser power level. The coordinates are equals to the X or Y corrected signals (as given by the RC command) divided by the also corrected SUM signal (see “CONEX-PSD-x Part5:Sensor Manual”) and then multiplied by half of the dimension of the sensor.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
Rel. Commands	**RA**	—	get raw analog inputs.

Example

1GP | *Get X,Y positions and laser power level.*

1GP3.125,-2.962,52 | *Controller returns X=3.125, Y=-2.962, LP=52%.*

ID — Set/Get sensor identifier

Usage	**Ready**	**Config.**	
	□	○	
Syntax	**xxIDnn** or **xxID?**		
Parameters			
Description	**xx** [int]	—	Controller address.
	nn [float]	—	Sensor model number.
Range	**xx**	—	**1** to **31**
	nn	—	**1** to **31** ASCII characters.
Units	**xx**	—	None
	nn	—	None
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
	nn Missing:		Error C.
	Out of range:		Error C.
Description	The ID command sets the sensor identifier string.		
Returns	If the sign "**?**" takes place of **nn**, this command returns the current programmed value.		
Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	C	—	Parameter missing or out of range.
	K	—	Execution not allowed in READY state.
Example	1ID? | *Get sensor identifier for controller #1.*		
	CONEX-PSD | *Controller returns CONEX-PSD.*		

IS — Set/Get offset on ADC input SUM

Usage	**Ready**	**Config.**	
	–	○	
Syntax	**xxISnn or xxIS?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
	nn [float]	—	**>-2.5** and **<2.5**
Units	**xx**	—	None
	nn	—	Volts
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
	nn Missing:		Error C.
	Out of range:		Error C.

Description **This command is only active with the Si sensor.**
In CONFIGURATION state, this command will set the offset for the ADC input SUM. The default value is 0.
When used with the Ge sensor, the IS? command returns the value 0 and must not be modified.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	C	—	Parameter missing or out of range.
	D	—	Execution not allowed.
Example	1IS0.010	|	*Sets the ADC input offset to 10mV for SUM channel of controller #1.*

IX — Set/Get offset on ADC input X

Usage	**Ready**	**Config.**	
	–	○	
Syntax	**xxIXnn or xxIX?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
	nn [float]	—	**>-2.5** and **<2.5**
Units	**xx**	—	None
	nn	—	Volts
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
	nn Missing:		Error C.
	Out of range:		Error C.

Description

Si sensor:
In CONFIGURATION state, this command will set the offset for the **ADC input X**.
The offset value is subtracted from the ADC input X value. The default value is 0.
Ge sensor:
In CONFIGURATION state, this command will set the offset for the **X position**.
The offset value is subtracted from the X position value. The default value is 0.

Returns

If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	C	—	Parameter missing or out of range.
	D	—	Execution not allowed.
Example	1IX0.01	|	*Sets the ADC input offset to 10mV for X channel of controller #1.*

IY — Set/Get offset on ADC input Y

Usage	Ready	Config.
	–	○

Syntax **xxIYnn or xxIY?**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

nn [float] — **>-2.5** and **<2.5**

Units **xx** — None

nn — Volts

Defaults

xx	Missing:	Error B.
	Out of range:	Error B.
	Floating point:	Error A.
nn	Missing:	Error C.
	Out of range:	Error C.

Description **Si sensor**:
In CONFIGURATION state, this command will set the offset for the **ADC input Y**.
The offset value is subtracted from the ADC input Y value. The default value is 0.
Ge sensor:
In CONFIGURATION state, this command will set the offset for the **Y position**.
The offset value is subtracted from the Y position value. The default value is 0.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.

Example 1IY0.01 | *Sets the ADC input offset to 10mV for Y channel of controller #1.*

LF — Set/Get low pass filter

Usage	Ready	Config.
	–	○

Syntax **xxLFnn or xxLF?**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

nn [float] — > 0 and < 1000.0

Units **xx** — None

nn — Hertz

Defaults **xx** Missing: Error B.

Out of range: Error B.

Floating point: Error A.

nn Missing: Error C.

Out of range: Error C.

Description In CONFIGURATION state, this command will set the frequency of the first order low pass filter applied on the ADC inputs.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors A — Unknown message code or floating point controller address.

B — Controller address not correct.

D — Execution not allowed.

Example 1LF50 | *Sets the low pass filter frequency to 50 Hz of controller #1.*

OF — Set/Get offset on ADC inputs 1 to 4

Usage	**Ready**		**Config.**
	☐		●
Syntax	**xxOFnn,nn,nn,nn or xxOF?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
	nn [float]	—	**> -1** and **< 1**
Units	**xx**	—	None
	nn	—	None
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
Description	**This command is only available with the Ge sensor**. In CONFIGURATION state, this command will set the offset for the ADC input 1 to 4. Offset values are subtracted from ADC input values. Default values are 0.		
Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	D	—	Controller address not correct.
Rel. Commands	**ZT**	—	Get all parameters.
Example			
1OF0.01221,0.01221,-0.02442,-0.01221			*Sets the ADC input offset of controller #1.*

PS — Set/Get gain on ADC input SUM

Usage	**Ready**	**Config.**	
	–	○	
Syntax	**xxPSnn or xxPS?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
	nn [float]	—	**>0.1** and **<10**
Units	**xx**	—	None
	nn	—	None
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
	nn Missing:		Error C.
	Out of range:		Error C.

Description

This command is only active with the Si sensor.
In CONFIGURATION state, this command will set the gain applied on ADC input SUM. The default value is 1.
When used with the Ge sensor, the PS? command returns the value 1 and must not be modified.

Returns

If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.

Example

1PS0.995 | *Sets the ADC input gain to 0.995 for SUM channel of controller #1.*

PX — Set/Get gain on ADC input X

Usage	Ready	Config.
	–	○

Syntax **xxPXnn or xxPX?**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

nn [float] — **>0.1** and **<10**

Units **xx** — None

nn — None

Defaults

xx	Missing:	Error B.
	Out of range:	Error B.
	Floating point:	Error A.
nn	Missing:	Error C.
	Out of range:	Error C.

Description **Si sensor**:
In CONFIGURATION state, this command will set the gain applied on **ADC input X**.
The default value is 1.
Ge sensor:
In CONFIGURATION state, this command will set the gain applied on **X position**.
The default value is 1.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.

Example 1PX0.995 | *Sets the ADC input gain to 0.995 for X channel of controller #1.*

PY — Set/Get gain on ADC input Y

Usage	Ready	Config.
	–	○

Syntax **xxPYnn or xxPY?**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

nn [float] — **>0.1** and **<10**

Units **xx** — None

nn — None

Defaults

xx	Missing:	Error B.
	Out of range:	Error B.
	Floating point:	Error A.
nn	Missing:	Error C.
	Out of range:	Error C.

Description
Si sensor:
In CONFIGURATION state, this command will set the gain applied on **ADC input Y**.
The default value is 1..
Ge sensor:
In CONFIGURATION state, this command will set the gain applied on **Y position**.
The default value is 1.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.

Example 1PY0.995 | *Sets the ADC input gain to 0.995 for Y channel of controller #1.*

PW — Enter/Leave CONFIGURATION state

Usage	**Ready**	**Config.**	
	●	●	
Syntax	**xxPWnn**		
Parameters			
Description	**xx** [int]	—	Controller address.
	nn [float]	—	State.
Range	**xx**	—	**1** to **31**
	nn	—	**1**: Go from READY state to CONFIGURATION state.
			0: Go from CONFIGURATION state to READY state.
Units	**xx**	—	None.
	nn	—	None.
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.
	nn Missing:		Error C.
	Out of range:		Error C.

Description

PW1 changes the controller's state from READY to CONFIGURATION. In Configuration state all parameter settings are saved in the controller's memory and remain available after switching off the controller. In addition, some settings are only possible in CONFIGURATION state (e.g. offsets or gains.).

PW0 checks all stage parameters, and if they are acceptable, saves them in the flash memory of the controller. After that, it changes the controller's state from CONFIGURATION to READY.

The execution of a PW0 command may take up to 10 seconds. During that time the controller will not respond to any other command.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	C	—	Parameter missing or out of range.
	D	—	Execution not allowed.
Example	1PW1	|	*Changes controller #1 to CONFIGURATION state.*

RA — Get analog input values

Usage	**Ready**	**Config.**	
	●	●	
Syntax	**xxRA or xxRA?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
Units	**xx**	—	None.
Defaults	**xx**	Missing:	Error B.
		Out of range:	Error B.
		Floating point:	Error A.

Description

The RA command returns the value of the analog inputs.
The converter is a ±11 bits analog to digital converter with ±0.15 volts of maximum offset and 5% full-scale linearity.
The resolution is 0.078125 volts.
The CONEX-PSD controller has analog input low pass filters with a cut-off frequency of 175Hz .

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	D	—	Execution not allowed.

Example for PSD9

1RA | *Get controller #1 analog inputs.*

1RA0.9,1.2,2.3 |Controller returns: X=0.9, Y=1.2 and SUM=2.3.

Example for PSD10GE

1RA | *Get controller #1 analog inputs.*

1RA11.66,8.75,10.28,10.28 |Controller returns: X1=11.66, X2=8.75, Y1=10.28 and Y2=10.28

RC — Get corrected analog input values

Usage	**Ready**	**Config.**	
	●	●	
Syntax	**xxRC or xxRC?**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
Units	**xx**	—	None.
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.

Description

The RC command returns the value of the corrected analog inputs.
The converter is a ±11 bits analog to digital converter with ±0.15 volts of maximum offset and 5% full-scale linearity. The resolution is 0.078125 volts.
For X input, the RC command will return the RA reading minus X offset (set by the commands IX, IY, IS) and then multiplied by X gain (set by the commands PX, PY, PS). This allows an easy scaling to a specific setup.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	D	—	Execution not allowed.

Example for PSD9

1RC | *Get controller #1 corrected analog inputs.*

1RC0.9,1.2,2.3 | Controller returns: X=0.9, Y=1.2 and SUM=2.3.

Example for PSD10GE

1RC | *Get controller #1 corrected analog inputs.*

1RC11.66,8.75,10.28,10.28 |Controller returns: X1=11.66, X2=8.75, Y1=10.28 and Y2=10.28

RS — Reset controller

Usage	Ready	Config.
	●	●

Syntax **xxRS**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

Units xx — None.

Defaults **xx** Missing: Error B.
Out of range: Error B.
Floating point: Error A.

Description The RS command issues hardware reset of the controller, equivalent to a power-up.

LED display is turned off, controller re-initializes itself and after a short period of time returns to the READY state with LED display solid green. During initialization CONEX-PSD is not communicating.

Errors A — Unknown message code or floating point controller address.
B — Controller address not correct.
D — Execution not allowed.

Example 1RS | *Reset controller #1.*

RS## — Reset controller's address

Usage	**Ready**	**Config.**	
	☐	☐	
Syntax	**xxRS## or RS##**		
Parameters			
Description	**xx** [int]	—	Axis number.
Range	**xx**	—	**0 to 31**
	##	—	**Always ##**
Units	**xx**	—	None.
Defaults	**xx** Missing:		Change to 0.
	Out of range:		Error B.
	Floating point:		Error A.
Description	The RS## command resets the controller's address to 1. This address needs to be different for each CONEX controller when connected to a RS-485 communication network.		
Returns			
Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
	D	—	Execution not allowed.
Example	2RS##	|	Res*et controller's #2 address to 1.*

SA — Set/Get controller's RS-485 address

Usage	Ready	Config.
	–	○

Syntax **xxSAnn** or **xxSA?**

Parameters

Description

xx [int] — Axis number.

nn [int] — Controller's axis number.

Range

xx — **1**

nn — **2** to **31**

Units

xx — None.

nn — None.

Defaults

xx Missing: Error B.

Out of range: Error B.

Floating point: Error A.

nn Missing: Error C.

Out of range: Error C.

Description The SA command sets the controller's RS-485 address.

Returns If the sign "**?**" takes place of **nn**, this command returns the current programmed value.

Errors

A — Unknown message code or floating point controller address.

B — Controller address not correct.

C — Parameter missing or out of range.

D — Execution not allowed.

Example 1SA3 | *Set controller's RS-485 address to 3.*

NOTE

Special function. Contact Newport to get additional support on this command.

TB — Get command error string

Usage	Ready	Config.
	●	●

Syntax **xxTBnn**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

nn [char] — Error code (refer to TE command).

Units **xx** — None.

Defaults **xx** Missing: Error B.

Out of range: Error B.

Floating point: Error A.

nn Missing: Returns explanation of current error.

Out of range: Error C.

Description The TB command returns a string that explains the meaning of the error code **nn** (see TE command for complete list).

Errors A — Unknown message code or floating point controller address.

B — Controller address not correct.

C — Parameter missing or out of range.

D — Execution not allowed.

Rel. Commands **TE** — Get error code.

Example 1TB@ | *Get explanation to error code @.*

1TB@ No error | Controller returns: @ meaning No error.

TE — Get last command error

Usage	Ready	Config.
	●	●

Syntax **xxTE**

Parameters

Description **xx** [int] — Controller address.

Range **xx** — **1** to **31**

Units **xx** — None.

Defaults **xx** Missing: Error B.
Out of range: Error B.
Floating point: Error A.

Description The TE command returns the currently memorized error. When a command is not executable, it memorizes an error. This error can be read with the TE command. After the execution of a TE command, the error buffer gets erased and another TE command will return @, means no error. When a new command error is generated before the previous command error is read, the new command error will overwrite the current memorized error.

For a safe program flow it is recommended to always query the command error after each command execution.

Errors
A — Unknown message code or floating point controller address.
B — Controller address not correct.
D — Execution not allowed.

Rel. Commands **TB** — Get error string.

Example
1TE | *Get last error memorized on controller #1.*
| *Controller returns: 1TE@, means no error.*

List of errors and corresponding strings (see TB command):
@ — No error.
A — Unknown message code or floating point controller address.
B — Controller address not correct.
C — Parameter missing or out of range.
D — Command not allowed.
I — Command not allowed in CONFIGURATION state.
K — Command not allowed in READY state.
S — Communication Time Out.
V — Error during command execution.

TS — Get positioner error and controller state

Usage	**Ready**	**Config.**	
	●	●	
Syntax	**xxTS**		
Parameters			
Description	**xx** [int]	—	Controller address.
Range	**xx**	—	**1** to **31**
Units	**xx**	—	None.
	nn	—	None.
Defaults	**xx** Missing:		Error B.
	Out of range:		Error B.
	Floating point:		Error A.

Description The TS command returns the positioner error and the current controller state.

Returns The TS command returns six characters (1TSabcdef).

The first 4 characters (abcd) are set to "0000".

The last two characters (ef) represent the controller state.

Controller states (ef):

– **14**: CONFIGURATION.

– **32**: READY.

NOTES

The error buffer gets updated periodically, approx. every 1 ms.

The TS command reads the error buffer and clears the error buffer at the same time (same as for commands TE, TB). So when launching the TS command, it is important to process the TS feedback accordingly.

Errors	A	—	Unknown message code or floating point controller address.
	B	—	Controller address not correct.
Rel. Commands	**TE**	—	Get last error.
Example	1TS	|	*Get error and state of controller #1.*
	1TS000032	|	*Controller returns: READY.*

VE — Get controller revision information

Usage	Ready	Config.
	●	●

Syntax **xxVE**

Parameters

Description **xx** [int] — Controller address.
nn [string] — Action.

Range **xx** — **1** to **31**

Units **xx** — None.

Defaults **xx** Missing: Error B.
Out of range: Error B.
Floating point: Error A.

Description This command returns the controller's revision information.

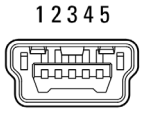
Errors A — Unknown message code or floating point controller address.
B — Controller address not correct.

Rel. Commands **GP** — Get current position.

Example 1VE | *Get controller #1 revision information.*
1VE CONEX-PSD revision 1.0.0. | *Controller returns the firmware revision*

3.0 Controller Interfaces

3.1 Mini-USB (Male) Connector Pinout



USB
MATING CONNECTOR:
PLUG MINI-USB B 5 CTS

PIN	DESCRIPTION
1	+5 VDC IN DO NOT CONNECT IF COMM CONNECTOR IS USED
2	DATA–
3	DATA+
4	N.C.
5	GND

Service Form

Your Local Representative

Tel.: _____

Fax:_____

Name: _____

Company:_____

Address:_____

Country:_____

P.O. Number:_____

Item(s) Being Returned:_____

Model#: _____

Return authorization #: _____
(Please obtain prior to return of item)

Date: _____

Phone Number: _____

Fax Number: _____

Serial #: _____

Description: _____

Reasons of return of goods (please list any specific problems): _____

Visit Newport Online at:

www.newport.com

North America & Asia
Newport Corporation
1791 Deere Ave.
Irvine, CA 92606, USA

Sales
Tel.: +1 (949)-863-3144
e-mail: sales@newport.com

Technical Support
Tel.: +1 (949)-863-3144
e-mail: tech@newport.com

Service, RMAs & Returns
Tel.: +1 (949)-863-3144
e-mail: service@newport.com

Europe
MICRO-CONTROLE Spectra-Physics S.A.S
7 rue des Plantes
45340 Beaune-la-Rolande
France

Sales Europe (EMEAI)
Tel.: +49 (0) 6151-708-0
e-mail: germany@newport.com

Sales France
Tel.: +33 (0)1 60 91 68 68
e-mail: france@newport.com

Sales UK
Tel.: +44 (0)1235 432 710
e-mail: uk@newport.com

Technical Support
e-mail: tech_europe@newport.com

Service & Returns
Tel.: +33 (0)2 38 40 51 55
DST-BEA-RMA-service@newport.com