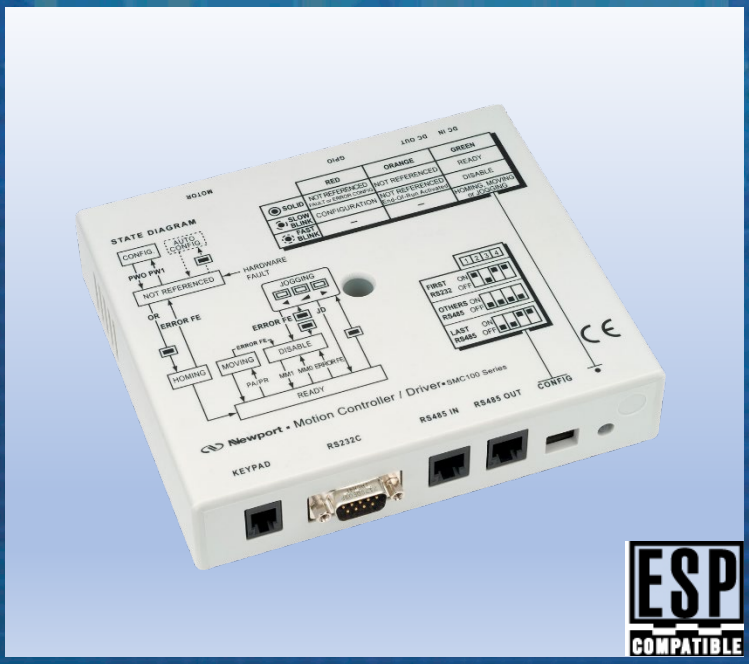
User's Manual

SMC100CC & SMC100PP

Single-Axis Motion Controller/Driver for DC or Stepper Motor



1771 Deere Avenue
Irvine, CA 92606 United States
+1 (949)-863-3144

FAX: +1 (949)-253-1680

WWW.NEWPORT.COM

Table of Contents

Warranty

MKS Instruments, Inc. warrants that this product will be free from defects in material and workmanship and will comply with MKS published specifications at the time of sale for a period of one year from date of shipment. If found to be defective during the warranty period, the product will either be repaired or replaced at MKS option.

To exercise this warranty, write or call your local MKS office or representative. You will be given prompt assistance and return instructions. Send the product, freight prepaid, to the indicated service facility. Repairs will be made, and the instrument returned freight prepaid. Repaired products are warranted for the remainder of the original warranty period or 90 days, whichever occurs last.

Limitation of Warranty

The above warranties do not apply to products which have been repaired or modified without MKS written approval, or products subjected to unusual physical, thermal or electrical stress, improper installation, misuse, abuse, accident or negligence in use, storage, transportation or handling.

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR USE. MKS INSTRUMENTS, Inc. SHALL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM THE PURCHASE OR USE OF ITS PRODUCTS.

EU Declaration of Conformity

2 Tech Drive
Andover, MA 01810
www.mksinst.com

EU27 Declaration of Conformity

Application of Council Directive(s):
☒ Electromagnetic Compatibility Directive (EMCD) – 2014/30/EU
☒ Restriction of Hazardous Substances Directive (RoHS3) – (EU) 2015/863(7)
☒ Waste Electrical and Electronic Equipment – Directive 2012/19/EU

Standard(s) to which conformity is declared:
☒ EN 61326-1:2013 – (EMC)

Emissions:
☒ CISPR 11:2015 Industrial, Scientific and Medical Equipment Radio-Frequency Disturbance Characteristics - Limits and Methods of Measurement

Immunity:
☒ IEC 61000-4-2:2008 EMC/Electrostatic Discharge Immunity Test
☒ IEC 61000-4-3:2006 2006+AMD1:2007+AMD2:2010 EMC/Radiated Radio - Frequency Electromagnetic Field Immunity Test
☒ IEC 61000-4-4:2012 EMC/Electrical Fast Transient/Burst Immunity Test
☒ IEC 61000-4-6:2013 EMC/Conducted Disturbances induced by Radio Frequency Fields Immunity Test
☒ IEC 61000-4-11:2004 + AMD 1:2017 EMC/Voltage Dips, Short Interruptions and Variations Immunity Test (5)

Manufacturers Name: MKS Instruments, Inc. Andover, MA, USA

Importer's Name & Location: /

Equipment Type/Description: **Motion Controller, single axis.**

Model Number(s) (6): **SMC100CC/PP; SMC-232/-USB/-PS80/-CB1/-CB3**

The object of the declaration described above is in conformity with the relevant Community harmonization legislation. MKS product conforms to the above Directive(s) and Standard(s) only when installed in accordance with manufacturer's specifications. This declaration has been issued under the sole responsibility of the manufacturer.

Date: 12/20/2021

Le Cointe Hervé
Quality Director

5) Applicable to AC powered product only.
6) Compliance of the above model numbers requires the use of a braided shielded cable properly terminated at both ends – if so noted in the MKS Instruction Manual.
7) RoHS Directive has to be checked for in scope products; cannot CE mark without compliance to RoHS. RoHS Directive can be unchecked only for systems which MKS sells which qualify for "Large Scale Industrial Tool" exclusion.

UK Declaration of Conformity

2 Tech Drive
Andover, MA 01810
www.mksinst.com

UK Declaration of Conformity

Application of Council Directive(s):
☒ Electromagnetic Compatibility Directive (EMCD) – 2014/30/EU
☒ Restriction of Hazardous Substances Directive (RoHS3) – (EU) 2015/863(7)
☒ Waste Electrical and Electronic Equipment – Directive 2012/19/EU

Standard(s) to which conformity is declared:
☒ BS EN 61326-1:2013 – (EMC)

Emissions:
☒ CISPR 11:2015 Industrial, Scientific and Medical Equipment Radio-Frequency Disturbance Characteristics - Limits and Methods of Measurement

Immunity:
☒ IEC 61000-4-2:2008 EMC/Electrostatic Discharge Immunity Test
☒ IEC 61000-4-3:2006 2006+AMD1:2007+AMD2:2010 EMC/Radiated Radio - Frequency Electromagnetic Field Immunity Test
☒ IEC 61000-4-4:2012 EMC/Electrical Fast Transient/Burst Immunity Test
☒ IEC 61000-4-6:2013 EMC/Conducted Disturbances induced by Radio Frequency Fields Immunity Test
☒ IEC 61000-4-11:2004 + AMD 1:2017 EMC/Voltage Dips, Short Interruptions and Variations Immunity Test (5)

Manufacturers Name: MKS Instruments, Inc. Andover, MA, USA

Importer's Name & Location: /

Equipment Type/Description: **Motion Controller, single axis.**

Model Number(s) (6): **SMC100CC/PP; SMC-232/-USB/-PS80/-CB1/-CB3**

The object of the declaration described above is in conformity with the relevant Community harmonization legislation. MKS product conforms to the above Directive(s) and Standard(s) only when installed in accordance with manufacturer's specifications. This declaration has been issued under the sole responsibility of the manufacturer.

Date: 12/20/2021

Le Cointe Hervé
Quality Director

5) Applicable to AC powered product only.
6) Compliance of the above model numbers requires the use of a braided shielded cable properly terminated at both ends – if so noted in the MKS Instruction Manual.
7) RoHS Directive has to be checked for in scope products; cannot CE mark without compliance to RoHS. RoHS Directive can be unchecked only for systems which MKS sells which qualify for "Large Scale Industrial Tool" exclusion.

Preface

CONFIDENTIALITY & PROPRIETARY RIGHTS

Reservation of Title

The MKS Instruments, Inc. Programs and all materials furnished or produced in connection with them ("Related Materials") contain trade secrets of MKS and are for use only in the manner expressly permitted. MKS claims and reserves all rights and benefits afforded under law in the Programs provided by MKS.

MKS shall retain full ownership of Intellectual Property Rights in and to all development, process, align or assembly technologies developed and other derivative work that may be developed by MKS. Customer shall not challenge, or cause any third party to challenge, the rights of MKS.

Preservation of Secrecy and Confidentiality and Restrictions to Access

Customer shall protect the MKS Instruments, Inc. Programs and Related Materials as trade secrets of MKS, and shall devote its best efforts to ensure that all its personnel protect the MKS Programs as trade secrets of MKS. Customer shall not at any time disclose MKS trade secrets to any other person, firm, organization, or employee that does not need (consistent with Customer's right of use hereunder) to obtain access to the MKS Programs and Related Materials. These restrictions shall not apply to information (1) generally known to the public or obtainable from public sources; (2) readily apparent from the keyboard operations, visual display, or output reports of the Programs; (3) previously in the possession of Customer or subsequently developed or acquired without reliance on the MKS Programs; or (4) approved by MKS for release without restriction.

SERVICE INFORMATION

The user should not attempt any maintenance or service of the present product and its accessories beyond the procedures outlined in this manual. Any problem that cannot be resolved should be referred to MKS I Newport. When calling MKS I Newport regarding a problem, please provide the Tech Support representative with the following information:

- Your contact information.
- System serial number or original order number.
- Description of problem.
- Environment in which the system is used.
- State of the system before the problem.
- Frequency and repeatability of problem.
- Can the product continue to operate with this problem?
- Can you identify anything that may have caused the problem?

MKS | NEWPORT RMA PROCEDURES

Any product being returned to MKS I Newport must have been assigned an RMA number by Newport. Assignment of the RMA requires the item serial number.

PACKAGING

Materials being returned under an RMA must be securely packaged for shipment. If possible, reuse the original factory packaging.

1 Safety Information

1.1 Definitions and Symbols

The following terms and symbols are used in this documentation and also appear on the SMC100 Controller/Driver where safety-related issues occur.

1.1.1 General Warning or Caution

The Exclamation Symbol may appear in Warning and Caution tables in this document. This symbol designates an area where personal injury or damage to the equipment is possible.

1.1.2 Electric Shock

The Electrical Shock Symbol may appear on labels affixed to the SMC100 Controller/Driver. This symbol indicates a hazard arising from dangerous voltage. Any mishandling could result in irreparable damage to the equipment, in personal injury, or death.

1.1.3 European Union CE Mark

The presence of the CE Mark on Newport Corporation equipment means that it has been designed, tested and certified as complying with all applicable European Union (CE) regulations and recommendations.

1.1.4 United Kingdom Conformity Assessed Mark

The presence of the UKCA Mark on Newport Corporation equipment means that it has been designed, tested and certified as complying with all applicable United Kingdom's regulations and recommendations.

1.2 Warnings and Cautions

Definitions of, NOTE, CAUTION, WARNING and DANGER messages used throughout the manual.

NOTE	**The NOTE sign denotes important information. It calls attention to a procedure, practice, condition, or the like, which is essential to highlight.**
CAUTION	**The CAUTION sign denotes a hazard. It calls attention to an operating procedure, practice, or the like, which, if not correctly performed or adhered to, could result in damage to or destruction of all or part of the product.**
WARNING	**The WARNING sign denotes a hazard. It calls attention to a procedure, practice, condition, on the like, which, if not correctly performed or adhered to, could result in injury to personnel.**
DANGER	**The DANGER sign Indicates an imminently hazardous situation that, if not avoided, will result in death or serious injury.**

1.3 General Warnings and Cautions

The following general safety precautions must be observed during all phases of operation of this equipment.

Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment.

- Heed all warnings on the unit and in the operating instructions.
- To prevent damage to the equipment, read the instructions in this manual.
- Only plug the power supply to a grounded power outlet.
- Assure that the power supply is properly grounded to earth ground through the grounding lead of the AC power connector
- Route power cords and cables where they are not likely to be damaged.
- Disconnect or do not plug in the AC power cord in the following circumstances:
    - If the AC power cord or any other attached cables are frayed or damaged.
    - If the power plug or receptacle is damaged.
    - If the unit is exposed to rain or excessive moisture, or liquids are spilled on it.
    - If the unit has been dropped or the case is damaged.
    - If the user suspects service or repair is required.
- Keep air vents free of dirt and dust.
- Keep liquids away from unit.
- Do not expose equipment to excessive moisture (>85% humidity).
- Do not operate this equipment in an explosive atmosphere.
- Disconnect power before cleaning the Controller/Driver unit. Do not use liquid or aerosol cleaners.
- Do not open the CONEX-SAG controller. There are no user-serviceable parts inside.
- Return equipment to Newport Corporation for service and repair.
- Dangerous voltages associated with the 100-240 VAC power supply are present inside the power supply. To avoid injury, do not touch exposed connections or components while power is on.
- Follow precautions for static-sensitive devices when handling electronic circuits.

2 System Overview

2.1 General Description

The SMC100CC/PP is a single axis motion controller/driver for DC servo or stepper motors up to 48 VDC at 1.5 A rms. It provides a very compact and low-cost solution for driving a variety of Newport and other manufacturers motorized positioners from a PC.

Communication with the SMC100CC/PP is achieved via a RS-232-C, or from a USB port using the external adapter SMC-USB (requires Windows™ operating system). A Windows™ based software (to be downloaded from Newport website) supports all configurations and enables basic motion. Advanced application programming is simplified by an ASCII command interface and a set of two letter mnemonic commands.

When used with Newport ESP enhanced positioners, the SMC100CC/PP will detect the connected product automatically and provides easy configuration using the supplied Windows-based utility software. This exclusive Newport feature reduces configuration time and provides the best protection of your equipment from any accidental damages.

Up to 31 controllers can be networked through the internal RS-485 communication link. This internal multi-drop full-duplex serial link simplifies communication to several units, without the need for sending "address selection commands". This results in enhanced multi-axes management with improved program readability and faster communication compared to alternative systems based on a RS-232-C chain. The typical execution time for a tell position command is only about 10 ms for the first controller and only about 16 ms for the other controllers. The SMC100CC/PP also features advanced "multi-axes" commands such as "Stop all" or "start a motion of all axes" and performs at a 57600 bauds rate communication speed. Furthermore, for an efficient process control, the SMC100CC/PP features dedicated digital outputs for "In Motion" and for "Not referenced".

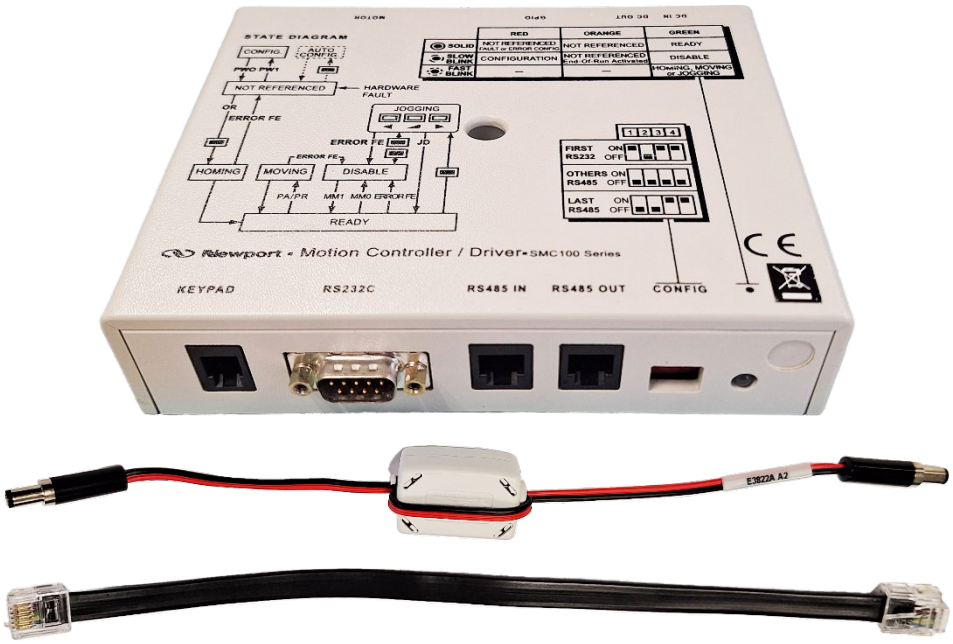
2.2 Part Numbers

Product	Description
SMC100CC	Single-axis motion controller/driver for DC servo motors. Includes 0.2 m long power and RS-485 cable.
SMC100PP	Single-axis motion controller/driver for stepper motors. Includes 0.2 m long power and RS-485 cable.
SMC-PS80	80 W power supply for SMC100CC/PP.
SMC-232	RS-232-C cable, 3 m length (DB9F to DB9F).
SMC-USB	USB interface, Includes one USB to COM port adapter and one RS-232-C cable. Requires Windows™ operating system.
SMC-CB1	1 m RS-485 cable (only required when RS-485 cable supplied with SMC100CC/PP is too short).
SMC-CB3	3 m RS-485 cable (only required when RS-485 cable supplied with SMC100CC/PP is too short).

2.3 SMC100CC/PP

2.3.1 Contents of Delivery

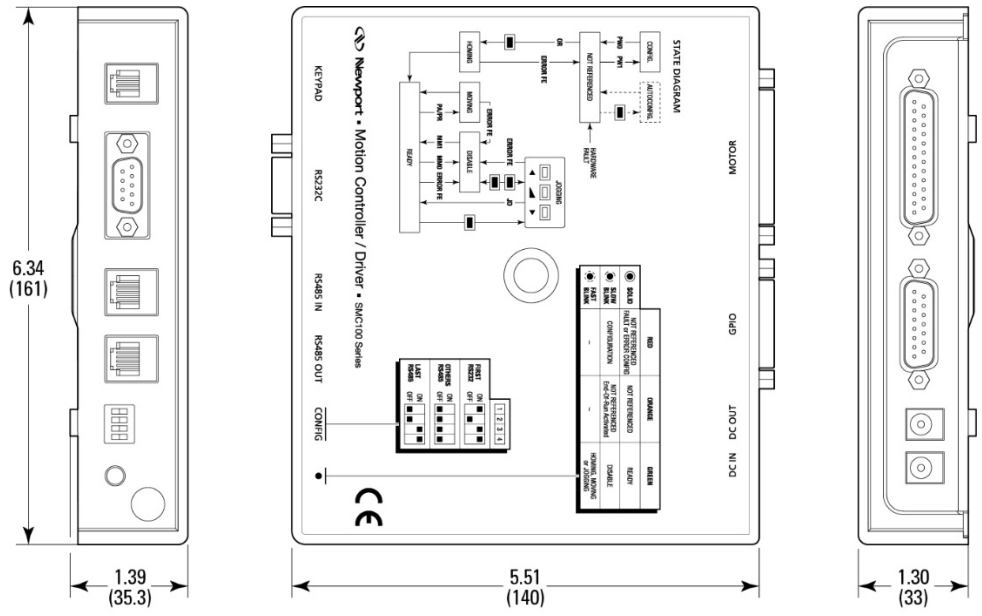
- SMC100CC/PP Controller box
- SMC-PSC0.2 Power cable, 0.2 m length
- SMC-CB0.2RS-485 network cable, 0.2 m length



2.3.2 Specifications

General Description	Single-axis motion controller/driver for DC servo motors (DC version) and for stepper motors (stepper version)
Control Capability	DC servo motors, open or closed loop operation (DC version) Stepper motors control, open loop operation only (stepper version)
Motor Output Power	– 48 VDC at 1.5 A rms, 3 A peak (DC version) – 48 VDC at 1.1 A rms per phase (stepper version) – 100 kHz PWM switching frequency
Control loop	– Floating point digital PID loop with velocity and friction feedforward – 2 kHz servo rate – Backlash compensation
Motion	Point-to-point motion with S-gamma profile and jerk time control
Computer interface	– RS-232-C with 57,600 baud rate – USB compatible with external adapter SMC-USB (requires Windows™ operating system) – RS-485 internal link for chaining up to 31 controllers from the same COM port
Programming	– 40+ intuitive, 2 letter ASCII commands – Command set includes software limits, user units, synchronized motion start, stop all
General purpose I/O	– 4 TTL out (Open collector, 30 V/40 mA Max.) – 4 TTL in (2.21 kΩ pull up to 5 V) – 1 analog input, ±10 V, 8-Bit
Dedicated inputs	– RS-422 differential encoder inputs for A, B, and I, max. 2 MHz rate – Forward and reverse limit, home switch and index pulse
Dedicated outputs	– 1 open-collector output for "In Motion" – 1 open collector output for "Not Referenced"
Status display	Two color LED
Internal safety feature	Watchdog timer

2.3.3 Dimensions



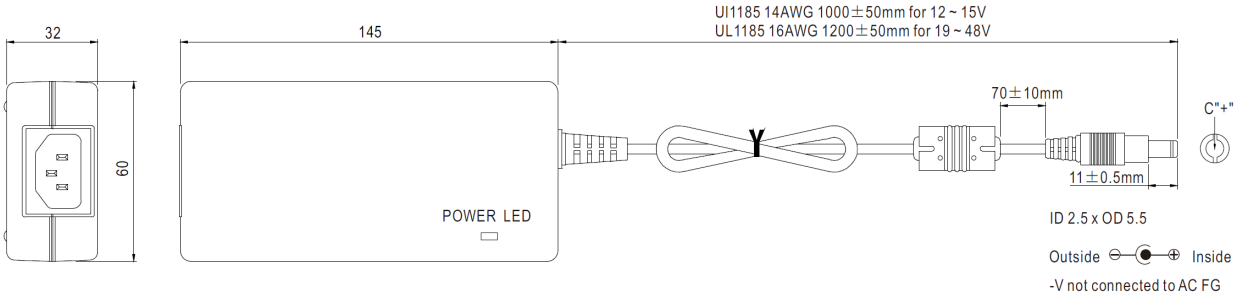
MODEL SHOWN: SMC100
DIMENSIONS IN INCHES (AND MILLIMETERS)

2.4 SMC-PS80

2.4.1 Specifications

AC Input	100–240 VAC, 47–63 Hz, 1.3 A
DC Output	48 V, 80 W max., 1.87A, < 240mVp-p ripple and noise
Load and line regulation	Better than ±2.5%
Connector	(male Ø 2.5 x Ø 5.5 x 11 mm)

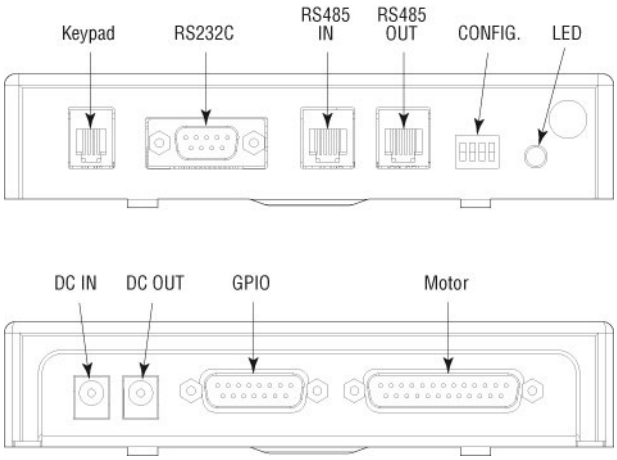
2.4.2 Dimensions



2.5 System Environmental Specifications

Operating temperature	5 °C to 40 °C
Operating humidity	< 85% relative humidity, non-condensing
Storage temperature	0 °C to 60 °C RH < 85% relative humidity, non-condensing
Installation category	II
Pollution degree	2
Use location	Indoor use only

2.6 Connector Identification



2.6.1 Front side

KEYPAD	RJ9F: For remote display and jog keypad.
RS-232-C	Sub-D9M: RS-232-C communication port for computer communication
RS-485 IN	RJ11F: RS-485 input for chaining several SMC100CC/PP in a multi-drop configuration
RS-485 OUT	RJ11F: RS-485 output for chaining several SMC100CC/PP in a multi-drop configuration
CONFIG.	4 switches: Dip switches for communication setup
LED	LED: Status LED

2.6.2 Back side

DC IN	Ø 2.1 x Ø 5.5 x 11 mm: Power supply input (connect to SMC80-PS)
DC OUT	Ø 2.1 x Ø 5.5 x 11 mm: Power supply repeater for connecting several SMC100CC/PP to the same power supply
GPIO	Sub-D15F: General purpose inputs/outputs
MOTOR	Sub-D25F: Motor connection

2.7 Serial Communication Settings

Communication parameters are preset in the SMC100CC/PP controller and do not require any configuration:

Bits per second	57,600
Data bits	8
Parity	None
Stop bits	1
Flow control	Xon/Xoff
Terminator	CRLF

3 Getting Started

This section guides the user through the proper set-up of the SMC100CC/PP motion control system. If not already done, carefully unpack and visually inspect the controllers and the positioners for any damage. Place all components on a flat and clean surface.

CAUTION

No cables should be connected to the controller at this point!

First, the controller must be configured properly. When using several SMC100CC/PP controllers from the same COM port through the internal RS485 communication link, an individual address must be set for each controller. Then, each controller must be configured to the connected positioner. For both steps, the software downloaded from Newport website on SMC100CC or SMC100PP page is used.

3.1 Communication Settings

3.1.1 RS232C Communication (Using SMC-232 Cable)

Apply the following settings to the COM port of your PC:

Bits per second	57,600
Data bits	8
Parity	None
Stop bits	1
Flow control	Xon/Xoff
Terminator	CRLF

3.1.2 USB Communication (Using SMC-USB Interface)

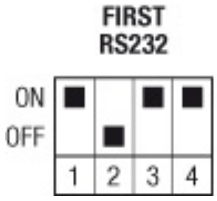
Install the software downloaded from Newport website on your PC. Follow the instructions supplied with the SMC-USB.

Apply the following settings to the COM port of your PC:

Bits per second	57,600
Data bits	8
Parity	None
Stop bits	1
Flow control	Xon/Xoff
Terminator	CRLF

3.2 Communication to a Single SMC100CC/PP

Set the dip switches on the SMC100CC/PP to FIRST:

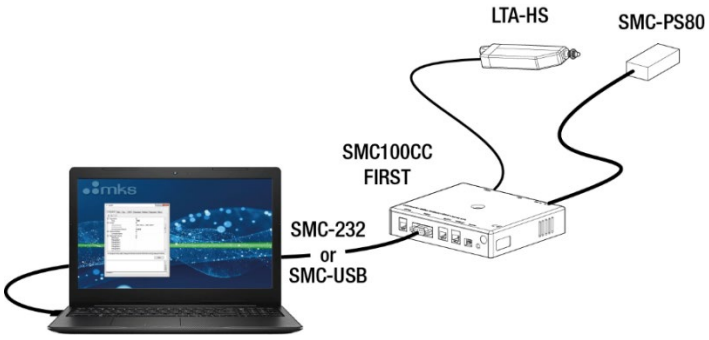


Connect the SMC100CC/PP to the RS232C or to the USB port of the PC by using a RS232 to USB converter (SMC-USB).

Connect the positioner to the SMC100CC/PP (MOTOR connector).

Connect the power supply.

The LED on the SMC100CC/PP turns RED.

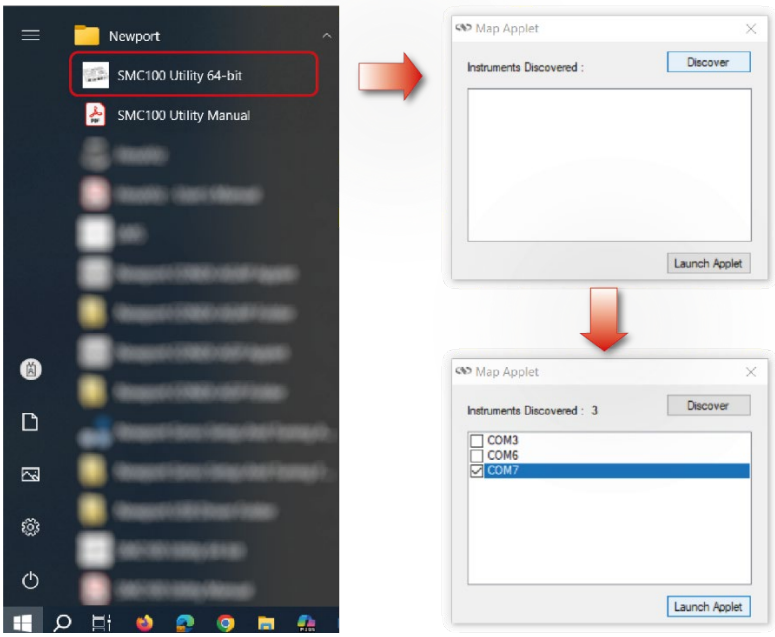


From Windows "START" menu , select "Newport\SMC100 Utility 32 bit" or "Newport\SMC100 Utility 64 bit" to open the SMC100 GUI Applet.

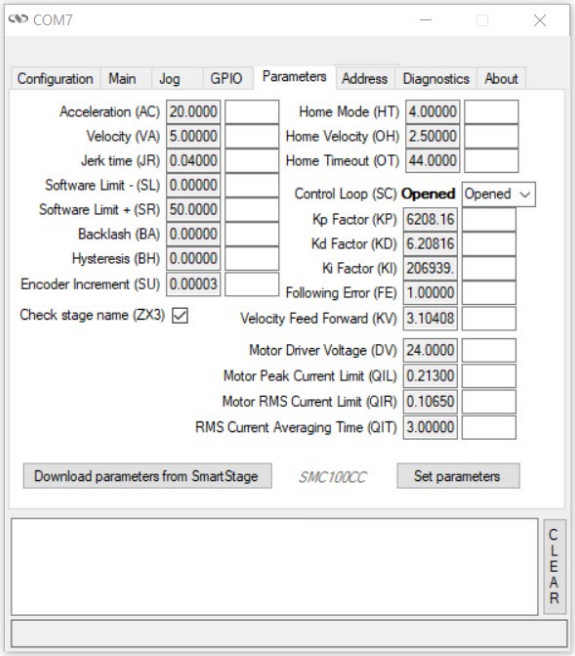
First select "Discover" button to display available COM port device.

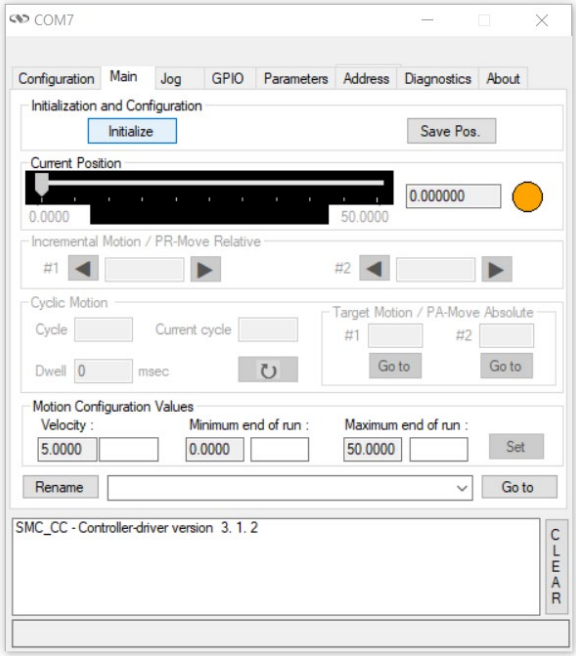
Then select the COM port device that the SMC100 controller is connected to.

Select "Launch Applet" button, the SMC100 Applet starts on "Configuration" tab.



Controller's status is displayed using LED display, refer to "SMC100 Controller GUI Manual" to set parameters and initialize positioners using **Download parameters from SmartStage** button in the "Parameters" tab and the **Initialize** button in the "Main" or "Jog" tab.





3.3 Communication to Several SMC100CC/PP

The SMC100 controller is equipped with a RS485 internal link for chaining up to 31 controllers from the same RS232C COM port.

Setting up a chain of several SMC100 controllers requires to:

- Configure RS485 address of all controllers,
- Configure the dip switches of all controllers according to the chaining process,
- Link the controllers to RS485 internal link,
- Configure the chain with the SMC100 applet,
- Configure and set parameters of each controller according to the connected positioner with the SMC100 applet.

NOTE **Command read is accepted in all states.**

SMC100CC and SMC100PP can be mixed in a daisy-chain

3.3.1 Controller RS485 Address Setting

The first thing to do is applying an individual RS485 address to each SMC100CC/PP controller.

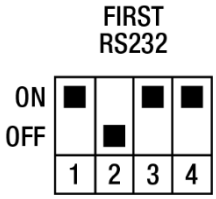
By default SMS100CC/PP are delivered in FIRST RS232 configuration with RS485 address number 2.

The SMC100 controller in the chain connected to the PC via the RS232 link is automatically identified as the first element in the chain and its RS485 address is 1, so there is no need to configure its RS485 address.

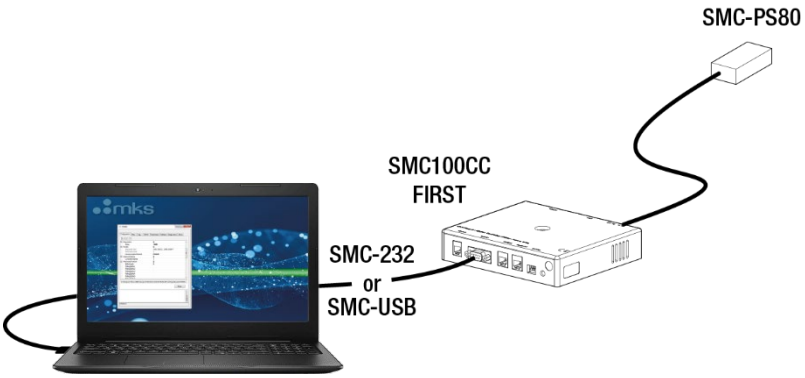
All other SMC100CC/PP controllers connected through the internal RS485 communication link have to be configured with a unique RS485 address from 2 to 31.

Each controller of the chain must be configured separately using RS232 link and must be configure in FIRST RS232:

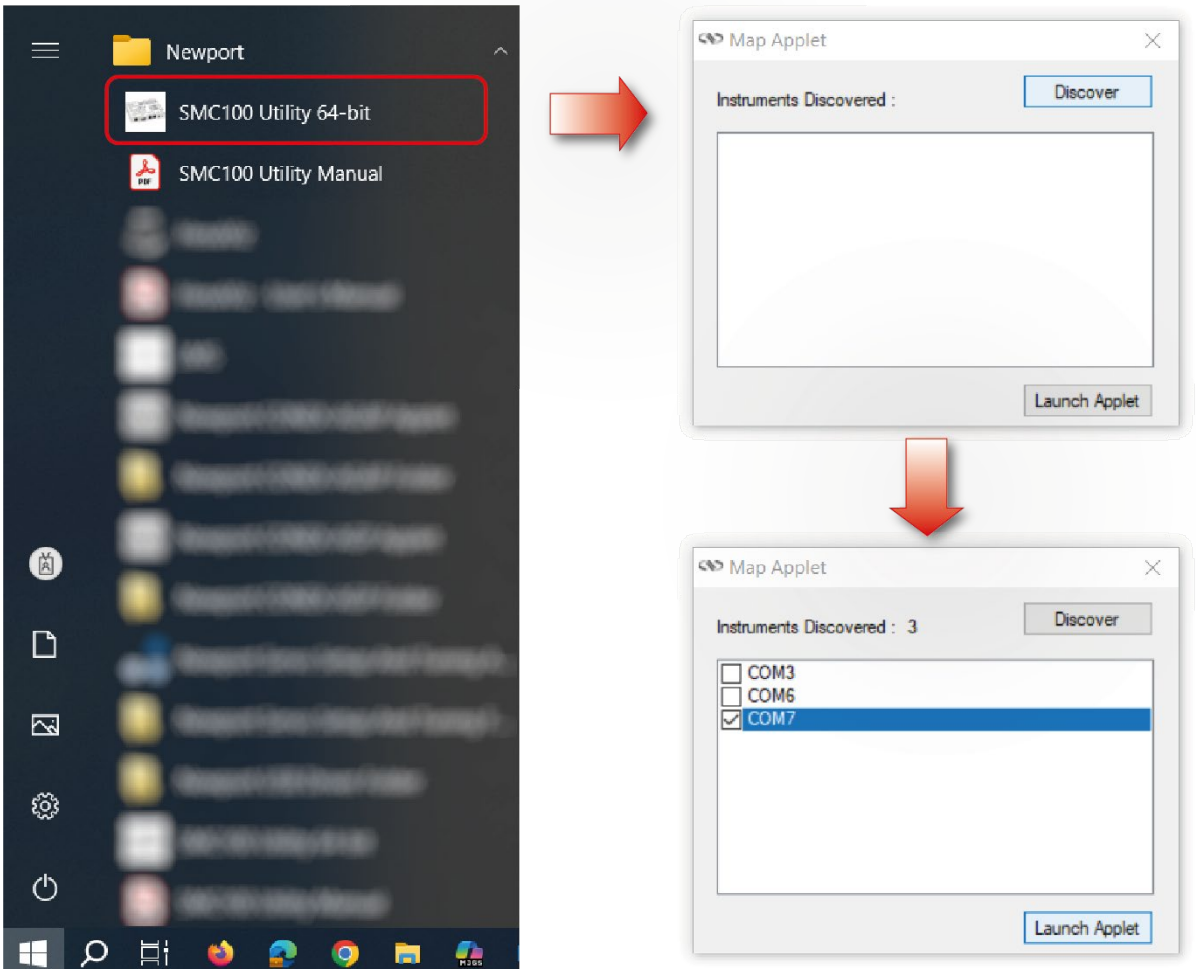
1. Set the dip switches of the SMC100CC/PP controller to FIRST RS232 (see graphic below).



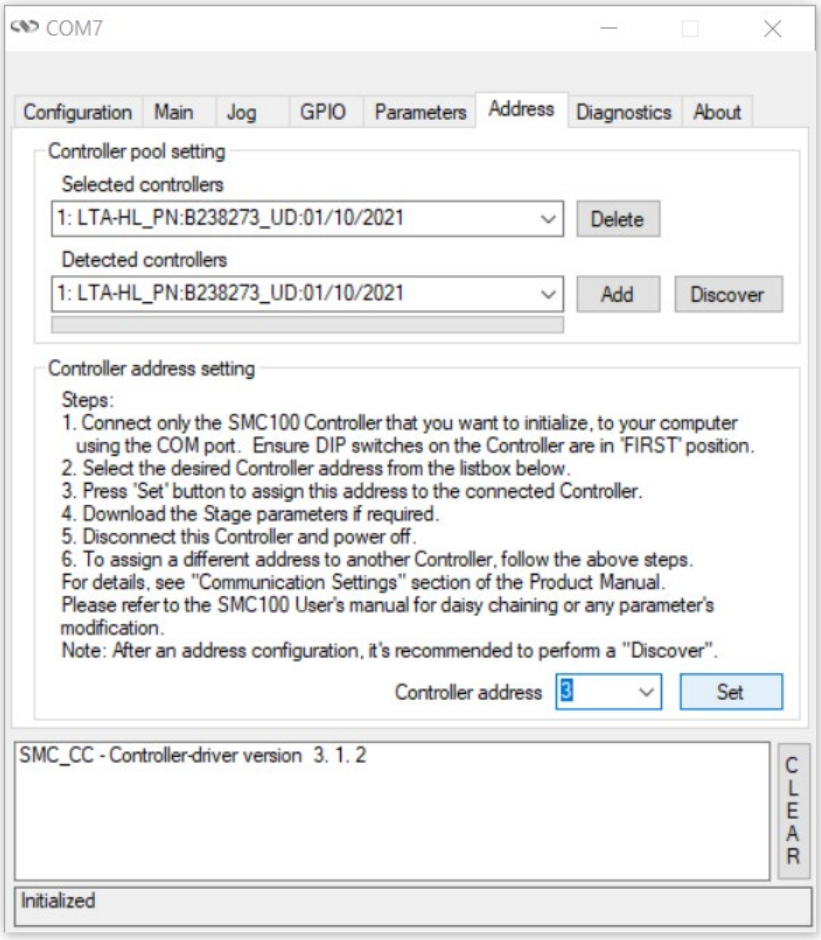
2. Connect the SMC100CC/PP to the RS232C or to the USB port of the PC by using a RS232 to USB converter (SMC-USB).
   It is not needed to connect any positioner to the controller.
   Connect the power supply.
   The LED turns RED.



3. From Windows "START" menu , select "Newport\SMC100 Utility 32 bit" or "Newport\SMC100 Utility 64 bit" to open the SMC100 GUI Applet.
   First select "Discover" button to display available COM port device.
   Then select the COM port device that the SMC100 controller is connected to.
   Select "Launch Applet" button, the SMC100 Applet starts on "Configuration" tab.



4. In the Newport SMC100 GUI, click on the "Address" tab, enter the desired controller address (in the SMC100 daisy-chain order) in the **Controller address** field and click on the **Set** button to save the controller address.



NOTE **Setting an address can also be performed by sending a SA command through GUI Diagnostics tab.**

This method requires to set the controller in CONFIGURATION state using PW command.

NOTE **It is recommended to note down the RS485 address of the controller, stickers are supplied with the SMC100CC/PP for this purpose**

5. Disconnect the controller from the PC.
6. Repeat the procedure for all SMC100 controller of the chain.

3.3.2 Building the System

Once all controller addresses have been defined, the system can be built.

Pull out all cables from all controllers.

Set the dip switches of the controller with the address number 1 as FIRST RS232, this controller will be connected to the PC via RS232 link.

Set the dip switches of the other controllers, except one, as OTHERS RS485, and set the dip switches of one controller as LAST RS485. See below graphic for illustration.

NOTE **When only two controllers are involved, one has to be set as FIRST RS232 (the one connected to the PC via RS232), and the other one as LAST RS485.**

Connect the SMC100CC/PP configured as FIRST RS232 to the RS232C or to the USB port of the PC by using a RS232 to USB converter (SMC-USB).

Connect a RS485 network cable from the RS485 OUT of the FIRST RS232 controller to the RS485 IN of the next controller (OTHERS 485).

Connect all controllers together from RS485 OUT of one to RS485 IN of the following controller.

The last controller in the chain is simply connected via its RS485 IN connector to the previous controller and is configured as LAST RS485.

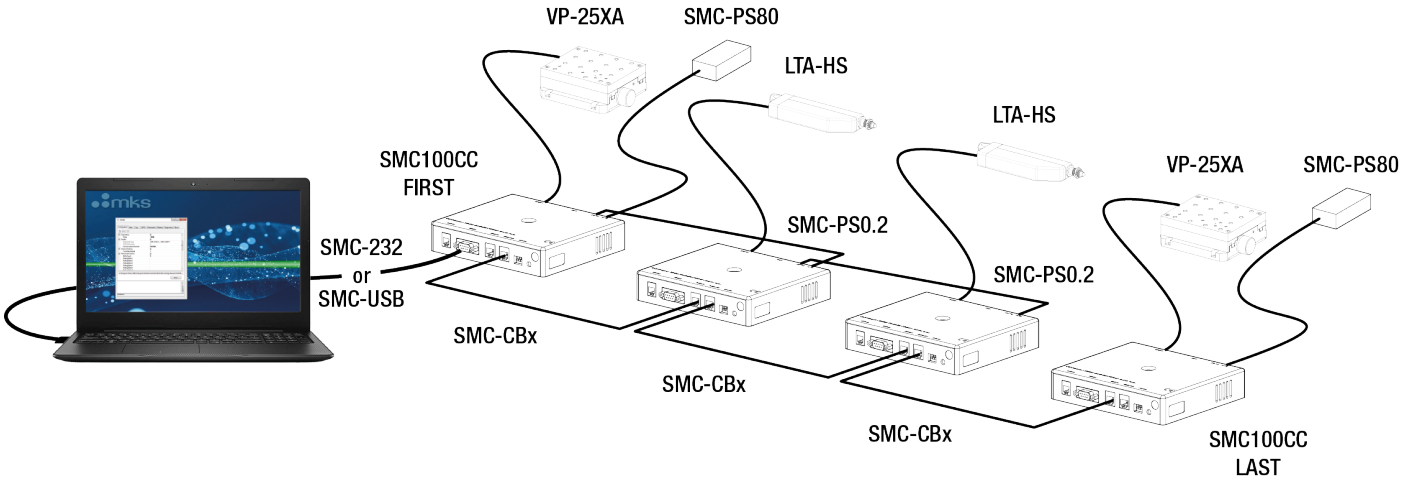
Connect the positioners to the SMC100CC/PP's (MOTOR connector).

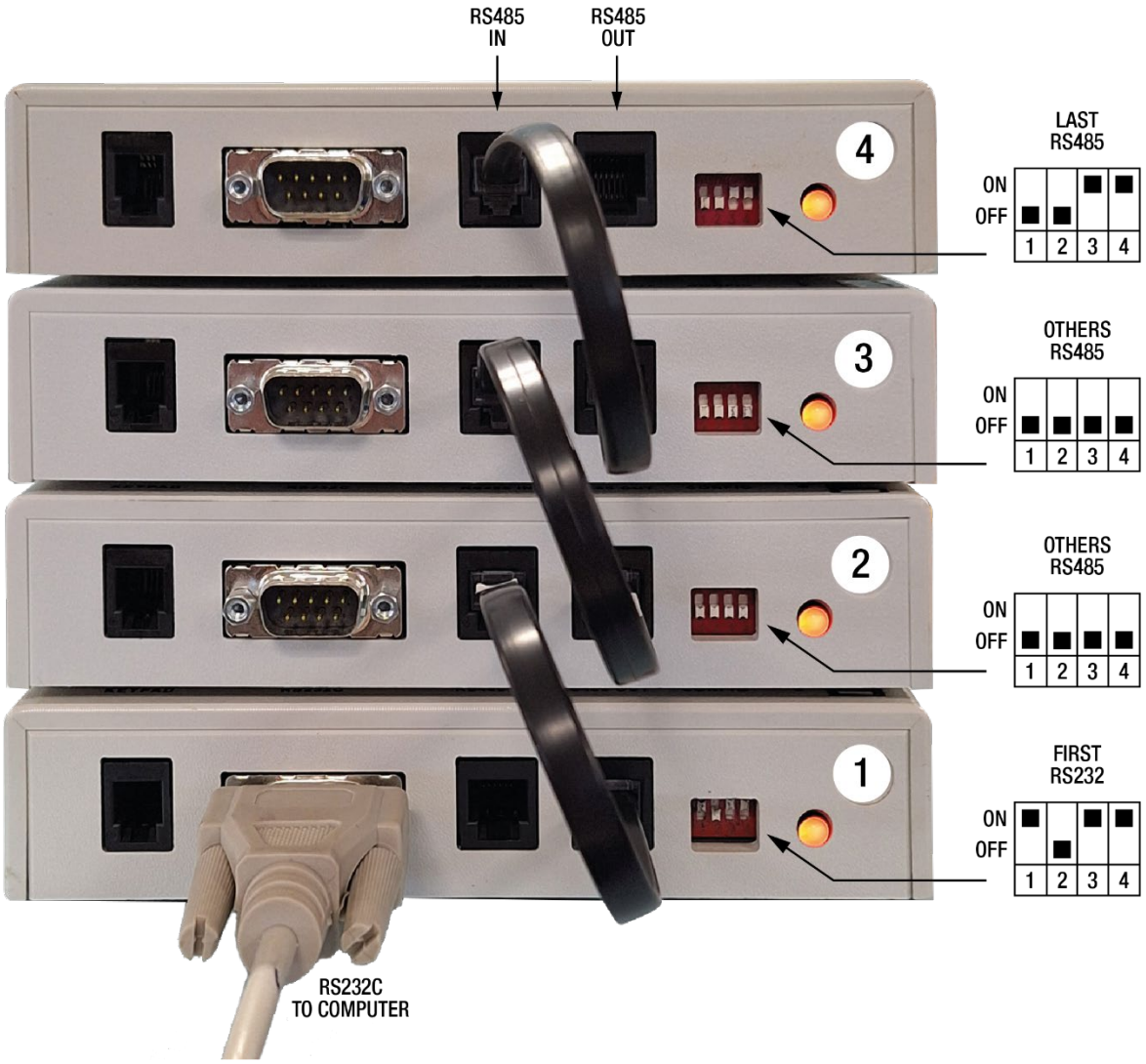
Connect the SMC100CC/PP's to power.

NOTE **The SMC100CC/PP allows chaining power from one SMC100CC/PP to another one using the SMC-PSC0.2 cable supplied with the controller. But the total power consumption of all positioners connected to the same power supply should not exceed 80 W. The maximum power consumption of each Newport positioner is listed in the Newport catalog and on the Newport web site. In case of questions, contact Newport.**

Example:
The maximum power consumption of a VP-25XA is 48 W. The maximum power consumption of an LTA-HS is 6 W. So it is possible to connect one VP-25XA and up to 5 LTA-HS to the same power supply. But it is not possible to connect two VP-25XA to the same power supply.

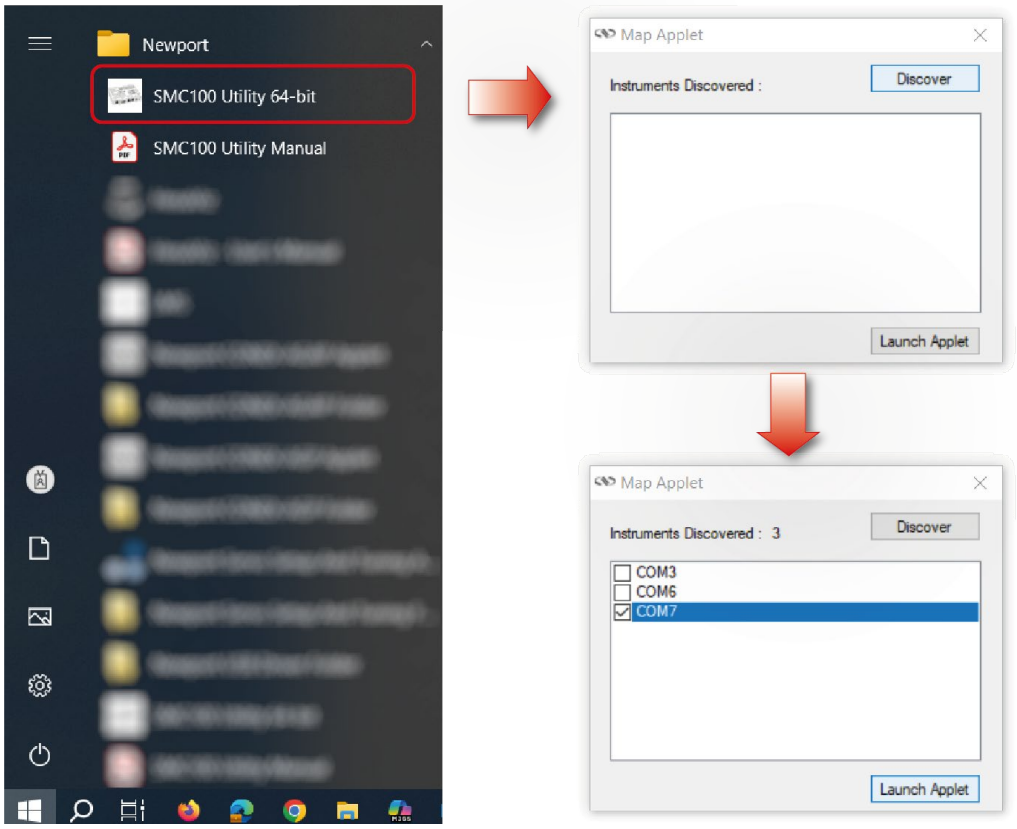
When done, the configuration should look as follow:



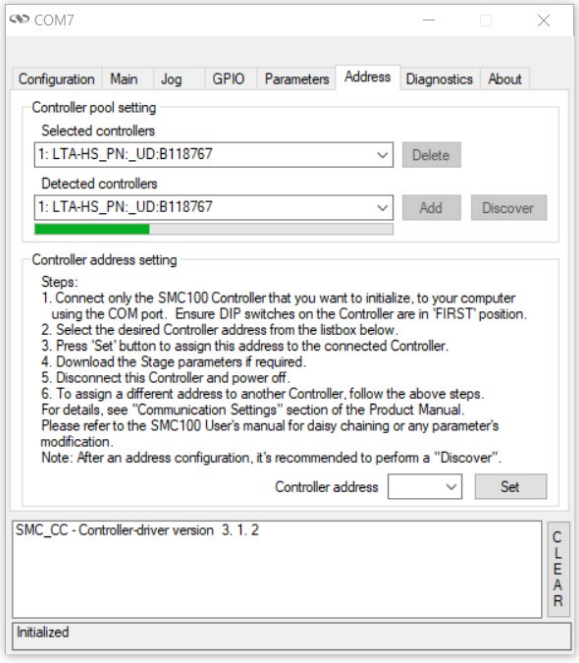


3.3.3 Configuring the Controller

Launch the SMC100 Utility to open the SMC100 GUI Applet.

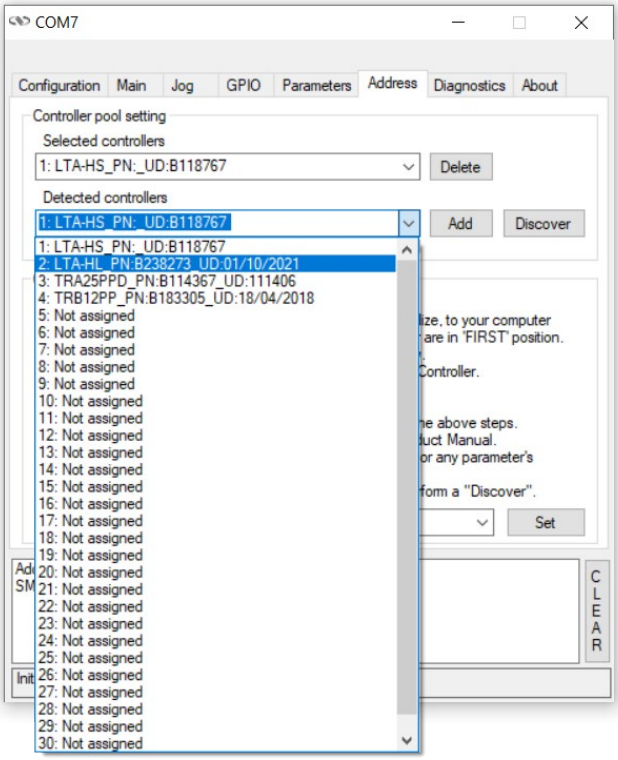


When using the SMC100CC/PP with Newport ESP compatible positioners (see label on the positioner), select "Address" tab and press **Discover** button.

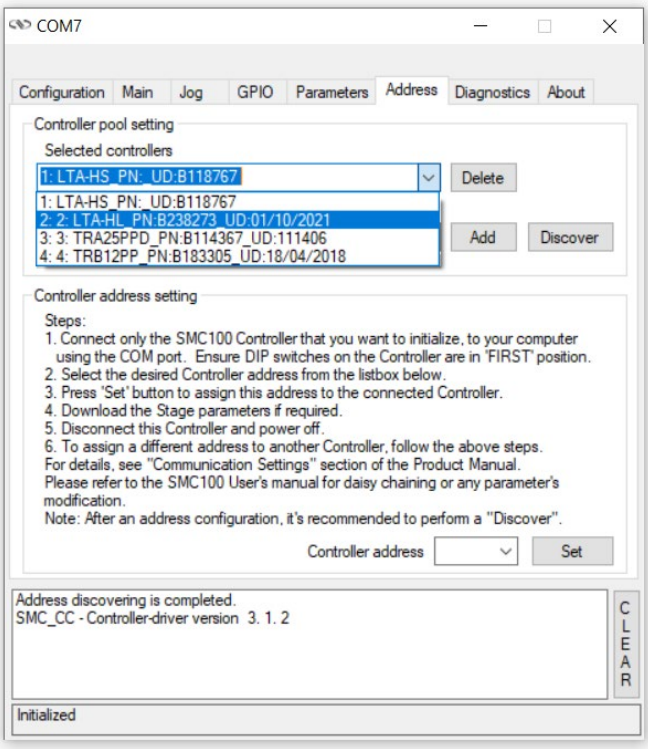


The SMC100 applet checks all the connected SMC100 and identifies the connected positioners.

Once the discovering process is completed, select the positioners to add in the SMC100 daisy-chain in the **Detected Controllers** list and add them pressing **Add** button.

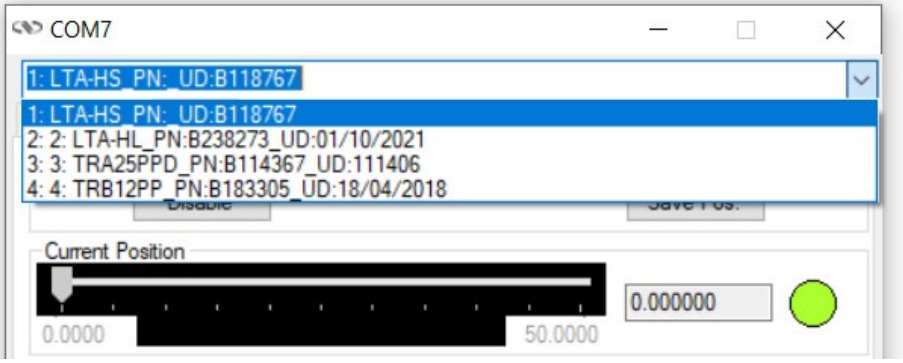


The list of the daisy-chained controllers is updated in the **Selected Controllers** list and can be managed using the **Delete** button.

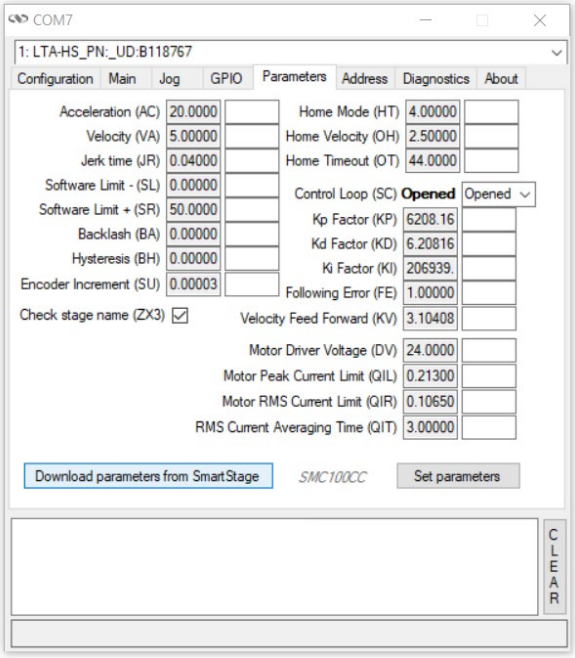


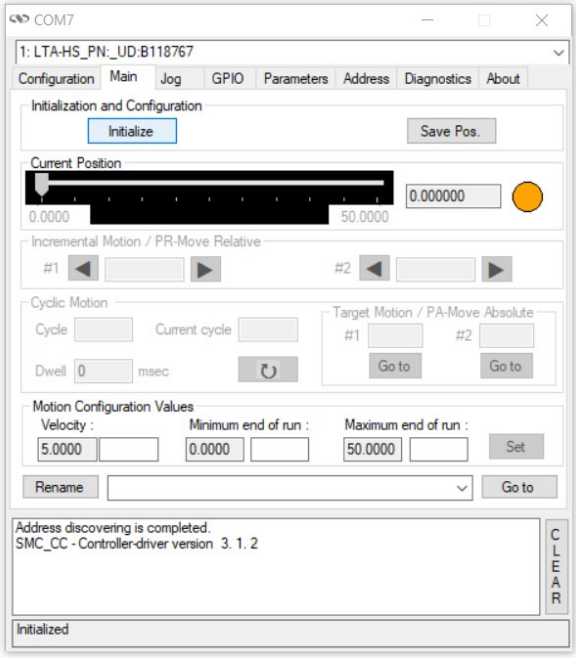
The daisy-chain is created, controller's status are displayed using LED display.

Once the daisy-chain is created, a selection list is displayed above the function tabs and allows to select the controller to drive.



Set parameters and initialize positioners using **Download parameters from SmartStage** button in the "Parameters" tab and the **Initialize** button in the "Main" or "Jog" tab.





Using the SMC100CC/PP with non Newport ESP compatible positioners or changing the default values

When using the SMC100CC/PP with non Newport ESP compatible positioners, enter the positioner parameters manually in the "Parameters" tab.

In the "Parameters" tab edit the configuration parameters stored in the controller.

NOTE **This method is not recommended unless for an experienced user.**

For further information about the meaning of the different parameters, refer to the explanations at the corresponding two letter commands (see command names in brackets) in section 5.5 Command Set.

4 Default Speed Setting Control for Newport Stepper Stages (only available for SMC100PP controller)

Due to some technical reasons, all Newport stepper positioners will be set to be driven at reduced speed with the SMC100PP controller (Reduced speed = Nominal speed / 2.5).

In order to check which positioners can be driven at reduced speed or full speed, please refer to the Newport web site (SMC100PP web page).

For example, an URSPP stage with a max speed of 40 °/s will be driven with a max speed of 16 °/s when controlled by the SMC100PP controller

For positioners that can be driven at full speed (please refer to the Newport web site to get the list), the default speed setting can be increased by the user to get the full nominal speed.

4.1 Irms Current Setting for SMC100PP Controller

The connection type of a stepper motor can be bipolar (full winding) or unipolar (half winding), but the SMC100PP controller always controls the stepper motor in the full winding control mode.

So the Irms current in each case must be different each from other.

In the case of a unipolar motor, if the motor resistance (controlled in half winding) is R, so the same motor resistance controlled in full winding is 2R.

For the same power (and the same thermal dissipation) in all two cases, we must have:

$$R.I_{half}^2 = 2R.I_{full}^2 \quad (1)$$

Where:

- I_{half} is the motor current in the case of half winding control
  (this is also Asmart: value found in the stage smart EPROM memory).
- I_{full} is the motor current in the case of full winding control.

From (1) we have:

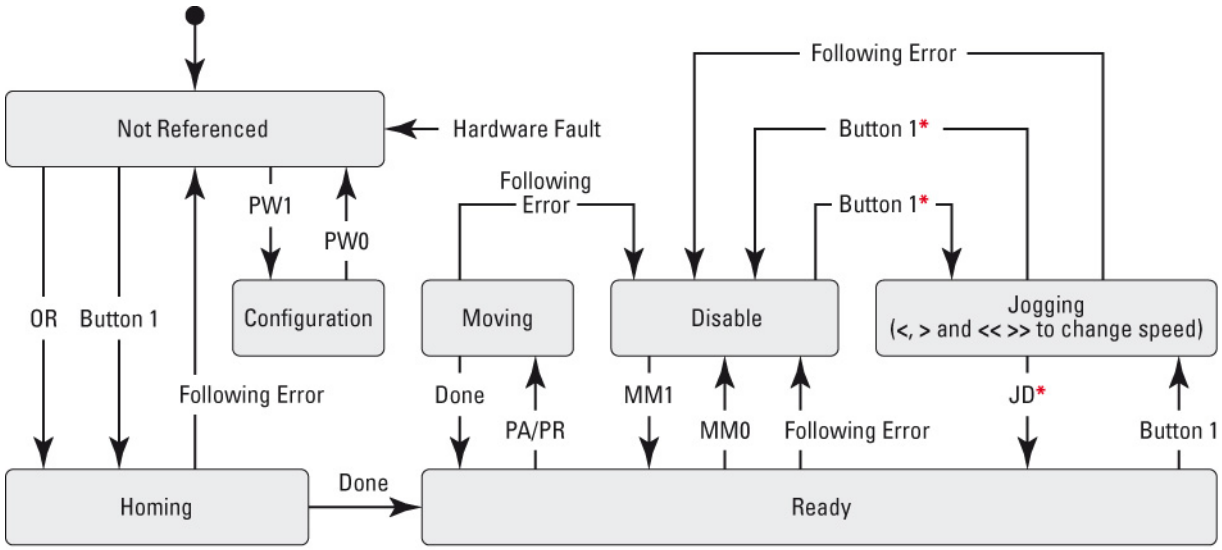
- $I_{full} = I_{half} \div \sqrt{2}$ *(2)*

So in the case of a unipolar motor controlled in full winding mode (SMC100PP), the motor must not be controlled with the Asmart value, but Asmart ÷ √2.

5 Programming

5.1 State Diagram

For a safe and consistent operation, the SCM100CC uses 7 different operation states: Not referenced, Configuration, Homing, Ready, Disable, Jogging and Moving. In each state, only specific commands are accepted by the SMC100CC/PP. Therefore, it is important to understand the state diagram below and which commands and actions cause transition between the different states. Also see section 5.3 for command/state information:



* No action, when jogging speed is different than zero, e.g. one of the keys "<", ">" or "<< >>" is pressed.

End of Runs encountered in the following state:

NOT REFERENCED:	No action.
CONFIGURATION:	No action.
HOMING:	Only check at end of HOMING and then change to NOT REFERENCED state.
MOVING:	Abort motion and then change to NOT REFERENCED state.
READY:	Change to NOT REFERENCED state.
DISABLE:	Change to NOT REFERENCED state.

LED display:

	RED	ORANGE	GREEN
SOLID	NOT REFERENCED: hardware faults or wrong parameters	NOT REFERENCED: everything is OK	READY
SLOW BLINK	CONFIGURATION	NOT REFERENCED: end of runs	DISABLE
FAST BLINK			HOMING MOVING JOGGING

When connecting the SMC100CC/PP to power, the controller initializes (see section 5.2 Initialization).

When the initialization is successful, the controller gets to the NOT REFERENCED state.

From the NOT REFERENCED state, the controller can go to the CONFIGURATION state with the PW1 command.

In CONFIGURATION stage, the SMC100CC/PP allows changing all positioner and motor configuration parameters like maximum motor current or travel limits.

The PW0 command saves all changes to the controller's memory and returns the controller back to the NOT REFERNCED state.

To execute any move commands (PA, PR), the controller must be in READY state.

To get from the NOT REFERENCED state to the READY state, the positioner must be homed first with the OR command.

During homing (OR command execution), the controller is in HOMING state. When the homing is successful, the controller automatically gets to the READY state. The process for homing, and which signals are looked for during homing, can be defined with the HT command.

In READY state the motor is energized and the control loop is closed (when control loop state is closed, SC1).

During a move execution (PA/PR), the controller is in MOVING state and gets automatically back to the READY state when the move is completed successfully.

A following error during a move changes the controller to DISABLE state. Other errors, for instance a loss of the encoder signals, may change the controller to the NOT REFERENCED state.

In DISABLE state the motor is not energized and the control loop is open (for DC version). But the encoder is still read and the current position gets updated (on the SMC100CC only).

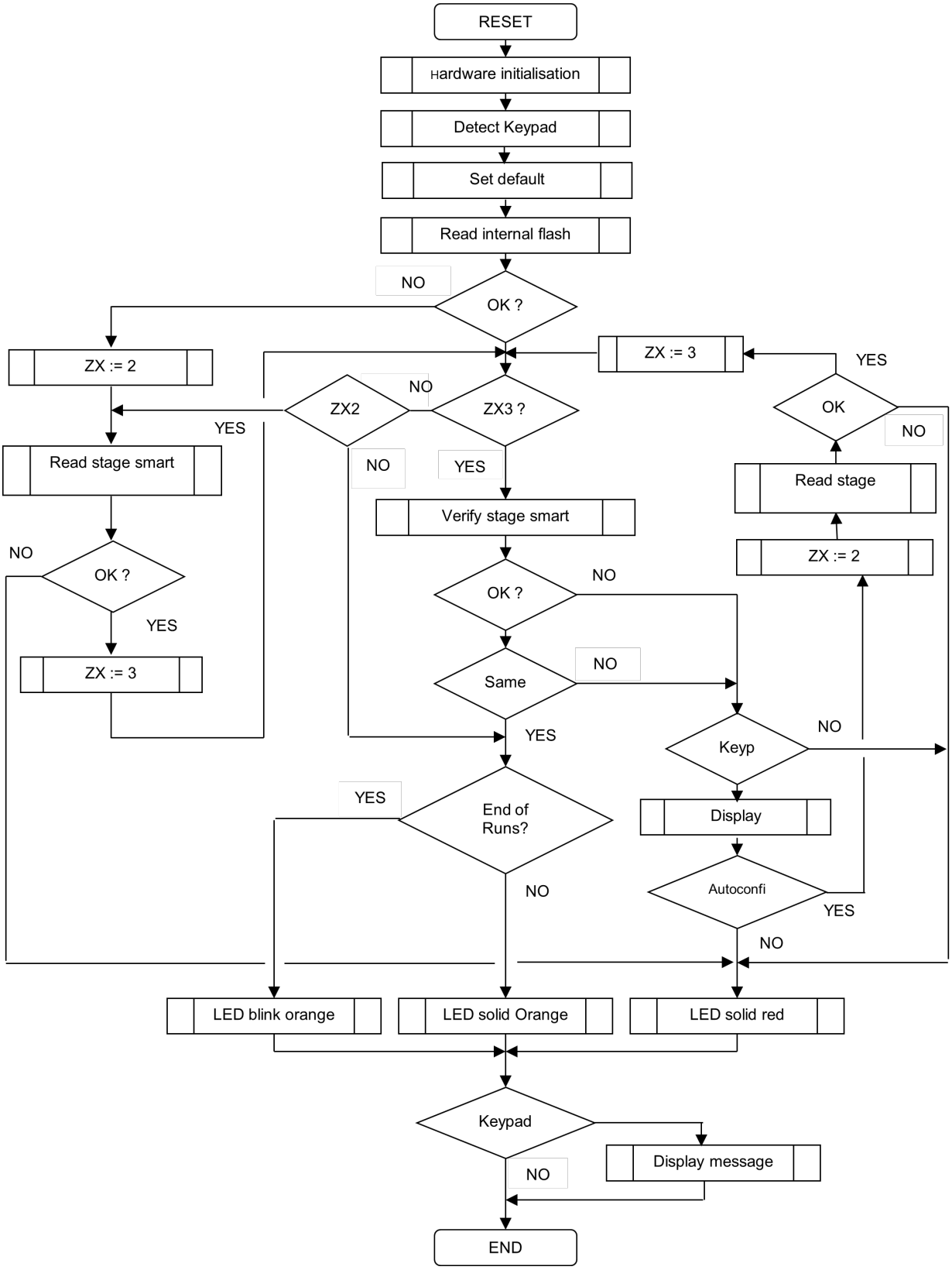
The DISABLE state can be used for instance for manual adjustments or to make sure that no energy goes to the motor.

To go from READY state to DISABLE state and vice versa, use the MM command.

To get from READY state or DISABLE state back to the NOT REFERENCED state, for instance to make some further parameter change in CONFIGURATION state, you need to reboot the controller with the RS command.

5.2 Initialization

When connecting the SMC100CC/PP to power, the following initialization routine gets executed. The initialization lasts less than 5 s. For more information about system errors during initialization, refer to the TS command in section 5.3.



5.3 Command Syntax

The SMC100CC/PP 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 SMC100.

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 5.3 for details.

5.4 Command Execution Time

The SMC100CC/PP controller interprets commands continuously as received. The typical execution time for a "tell position command" (nTP?) is about 10 ms for the first controller (controller address number 1) and about 16 ms for the other controllers. Here, command execution time means the time from sending the command until receive of the answer.

It is important to note that a move command, that may lasts for several seconds, will not suspend the controller from further command execution. So for an efficient process flow with many move commands it is recommended to use the PT command (get time for a relative move), and to query the controller status (TS command) or the current position (TP command) before any further motion command is sent. Alternative, the dedicated outputs "In Motion" and "Not Referenced" can be used for similar purposes. These will provide an even more timely accurate information of the controller state.

5.5 Command Set

This section describes the supported two-letter ASCII commands used to configure and operate the SMC100CC/PP. The general command format is:

Command format:

nn	AA	xx

nn — Optional or required controller address.

AA — Command name.

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

Since multiple SMC100CC/PP may be chained through the internal RS-485 Bus, each controller uses a predetermined address (**nn**), and by decoding the address field of the incoming commands, it can determine if the command is intended for it. Some command though, can be passed without a controller address. In that case the command applies to all concerned controllers.

For example: ST0 stops the motion on all controllers, 1ST0 stops the motion only on controller #1.

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 1VA10 sets the velocity of the controller #1 to 10 units/second. A 1VA? sends the response 1VA10.

Not every command can be executed in all states of the SMC100CC/PP and some commands have different meaning in different states. It is therefore important to understand the state diagram of the controller, see section 5.1.

	Not Ref.	Config.	Disable	Ready	Motion	Jogging	Description	SMC100CC/PP
AC		❍	❑	❑			Set/Get acceleration	□
BA		❍					Set/Get backlash compensation	□
BH		❍					Set/Get hysteresis compensation	□
DV		❍					Set/Get driver voltage	Not for PP
FD		❍	❑				Set/Get low pass filter for Kd	Not for PP
FE		❍	❑				Set/Get following error limit	Not for PP
FF		❍	❑				Set/Get friction compensation	Not for PP
FR		❍					Set/Get stepper motor configuration	Not for CC
HT		❍					Set/Get HOME search type	□
ID		❍					Set/Get stage identifier	□
JD						•	Leave JOGGING state	□
JM		❍	❑	❑			Enable/disable keypad	□
JR		❍	❑	❑			Set/Get jerk time	□
KD		❍	❑				Set/Get derivative gain	Not for PP
KI		❍	❑				Set/Get integral gain	Not for PP
KP		❍	❑				Set/Get proportional gain	Not for PP
KV		❍	❑				Set/Get velocity feed forward	Not for PP

	Not Ref.	Config.	Disable	Ready	Motion	Jogging	Description	SMC100CC/PP
MM			●	●			Enter/Leave DISABLE state	□
OH		❍					Set/Get HOME search velocity	□
OR	●						Execute HOME search	□
OT		❍					Set/Get HOME search time-out	□
PA				●			Move absolute	□
PR				●			Move relative	□
PT			●	●	●		Get motion time for a relative move	□
PW	●	●					Enter/Leave CONFIGURATION state	□
QI		❍					Set/Get motor's current limits	□
RA	●	●	●	●	●	●	Get analog input value	□
RB	●	●	●	●	●	●	Get TTL input value	□
RS	●		●	●			Reset controller	□
SA		❍					Set/Get controller's RS-485 address	□
SB			●	●	●	●	Set/Get TTL output value	□
SC		❍	❍				Set/Get control loop state	Not for PP
SE				●			Configure/Execute simultaneous started move	□
SL		❍	❑	❑			Set/Get negative software limit	□
SR		❍	❑	❑			Set/Get positive software limit	□
ST			●	●	●		Stop motion	□
SU		❍					Set/Get encoder increment value	Not for PP
TB	●	●	●	●	●	●	Get command error string	□
TE	●	●	●	●	●		Get last command error	□
TH	●	●	●	●	●	●	Get set-point position	□
TP	●	●	●	●	●	●	Get current position	□
TS	●	●	●	●	●	●	Get positioner error and controller state	□
VA		❍	❑	❑			Set/Get velocity	□
VB		❍	❑	❑			Set/Get base velocity	Not for CC
VE	●	●	●	●	●	●	Get controller revision information	□
ZT	●	●	●	●	●		Get all axis parameters	□
ZX		❍					Set/Get SmartStage configuration	□

Motion: Corresponds to HOMING and MOVING state (for details see state diagram, section 5.1).

❍ 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.

Blank: Not accepted command (will return an error).

Command: Command passed without preceding controller number applies to all controllers (e.g. MM0 disables all controllers).

Not for PP: The controller will return an error indicating that the command is not allowed for SMC100PP version.

Not for CC: The controller will return an error indicating that the command is not allowed for SMC100CC version.

AC — Set/Get acceleration

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxACnn** or **xxAC?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Acceleration value.

Range

xx	—	**1** to **31**
nn	—	**> 10^{-6}** and **< 10^{12}**

Units

xx	—	None
nn	—	Preset units/s^2

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 sets the maximum acceleration value which can than be saved in the controller's nonvolatile memory using the PW command. This is the maximum acceleration that can be applied to the mechanical system. It is also the default acceleration that will be used for all moves unless a lower value is set in DISABLE or READY state.
In DISABLE or READY state, this command sets the acceleration used for the following moves. Its value can be up to the programmed value in CONFIGURATION state. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

VA	—	Set velocity.

Example

1AC500	*|Set controller #1 acceleration to 500 units/s^2.*
1AC?	*|Controller returns 1AC500.*

BA — Set/Get backlash compensation

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxBAnn** or **xxBA?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Backlash value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 1E^{12}**

Units

xx	—	None
nn	—	Preset units

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 BA command sets the backlash compensation value. This is the value that the controller moves the motor in addition to the commanded distance with any move that reverses the direction of motion without changing the current position value (TP command).
The BA command helps compensating for repeatable mechanical defects that appear when reversing the direction of motion, for instance mechanical play. The value 0 disables this function. This feature can be only used when the hysteresis compensation (BH) is disabled.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **BH** — Set hysteresis compensation.

Example 1BA0.005 |*Set controller #1 backlash compensation to 0.005 units.*

BH — Set/Get hysteresis compensation

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxBHnn** or **xxBH?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Hysteresis value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None
nn	—	Preset units

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 BH command sets the hysteresis compensation value. When set to a value different than zero, the controller will issue for each move in the positive direction a move of the commanded distance plus the hysteresis compensation value, and then a second move of the hysteresis compensation value in the negative direction. This motion ensures that a final position gets always approached from the same direction and distance and helps compensating for non-repeatable mechanical defects like hysteresis or mechanical stiffness variations.
The value 0 disables this function. The BH command can not be used when the backlash compensation is enabled (BA command).

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **BA** — Set backlash compensation.

Example 1BH0.015 |*Set controller #1 backlash compensation to 0.015 units.*

DV — Set/Get driver voltage

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxDVnn** or **xxDV?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Driver voltage value.

Range

xx	—	**1** to **31**
nn	—	**≥ 12** and **≤ 48**

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 sets the max. output voltage of the driver to the motor.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

QI	—	Set current limit.

Example 1DV48 |*Set controller #1 maximum output voltage to 48 V.*

FD — Set/Get low pass filter cut off frequency for Kd

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxFDnn** or **xxFD?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Cut off frequency value.

Range

xx	—	**1** to **31**
nn	—	**> 10^{-6}** and **< 2000**

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 sets the value for the low pass filter cut-off frequency which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the low pass filter cut-off frequency. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands **SC** — Set closed loop state.

Example 1FD1500 |*Set controller #1 Kd cut-off frequency to 1500 Hz.*

FE — Set/Get following error limit

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxFEnn** or **xxFE?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Following error limit value.

Range

xx	—	**1** to **31**
nn	—	**> 10^{-6}** and **< 10^{12}**

Units

xx	—	None.
nn	—	Preset units.

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 sets the value for the maximum allowed following error which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used for the closed-loop control unless a different value is set in DISABLE state.
The following error is the most important parameter to control motion. It is the difference between the set point (or theoretical) position and the current (or encoder) position. When the current following error exceeds the maximum allowed value, a following error is issued and the controller is set to DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the maximum allowed following error. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands **SC** — Set closed loop state.

Example 1FE0.015 |*Set controller #1 following error limit to 0.015 units.*

FF — Set/Get friction compensation

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxFFnn** or **xxFF?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Friction compensation value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< DV**

Units

xx	—	None.
nn	—	Volt * second/preset units.

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 sets the value for the friction compensation which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used for any move unless a different value is set in DISABLE state.
The FF command helps minimizing the following error with systems that have significant friction. The value for the friction compensation is the voltage that gets added to the output voltage whenever the set point (or theoretical) velocity is different from zero. The sign of this voltage is the same as the sign of the set point velocity.
In DISABLE state, this command allows setting a new working parameter for the friction compensation. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands **SC** — Set closed loop state.

Example 1FF0.15 |*Set controller #1 friction compensation to 0.15 V * s/units.*

FR — Set/Get stepper motor configuration

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxFRSnn**, **xxFRM?** or **xxFRS?**

Parameters

Description

	xx [int]	—	Axis number.
	Mmm [int]	—	Micro-step factor.
	Snn [float]	—	Full step value.
Range	**xx**	—	**1** to **31**
	mm	—	**> 0** and **≤ 2000**
	nn	—	**> 1E^{-6}** and **< 1E^{12}**
Units	**xx**	—	None.
	Mmm	—	None.
	Snn	—	None.

Defaults

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

Description

FRM: this command sets the micro-step per full step factor.

FRS : this command sets the motion distance per motor's full step.

Returns

If the sign "**?**" takes place of mm or **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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
X	—	Command not allowed for SMC100CC version.

Rel. Commands VB — Set base velocity.

Example 1FRS0.02 |*Set controller #1 full step value to 0.02 units.*

HT — Set/Get HOME search type

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxHTnn** or **xxHT?**

Parameters

Description

	xx [int]	—	Controller address.
	nn [int]	—	Home type value.
Range	**xx**	—	**1** to **31**
	nn	—	**0** use MZ switch and encoder Index.
			1 use current position as HOME.
			2 use MZ switch only.
			3 use EoR- switch and encoder Index.
			4 use EoR- switch only.
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 sets the type of HOME search used with the OR command.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **OR** — Execute HOME search.

Example 1HT0 |*Set controller #1 HOME sequence to use MZ and encoder index.*

ID — Set/Get stage identifier

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxIDnn** or **xxID?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Stage 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 return the stage identifier. When used with Newport ESP compatible stages (see blue label on the product), this is the identical to the Newport product name. In CONFIGURATION mode, this command allows changing the stage identifier. However, customer should never do this when the ESP stage configuration is enabled (ZX3).

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **ZX** — Set SmartStage configuration.

Example

1ID?	|*Get stage identifier for controller #1.*
	|*Controller returns URS100CC.*

JD — Leave JOGGING state

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	❑	❑	❑	■

Syntax **xxJD**

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 In JOGGING STATE, when no jog buttons are pressed and the stage velocity is 0 the xxJD command sets the controller's state to READY.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **JM** — Enable/Disable keypad.

Example 1JD |*Controller #1 leaves jogging state.*

JM — Enable/Disable keypad

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxJMnn** or **xxJM?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Jog state.

Range

xx	—	**1** to **31**
nn	—	**0** or **1**

Units

xx	—	None
nn	—	None

Defaults

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

Description The JM1 command enables the keypad buttons (default setting). The JM0 command disables the keypad buttons.
Sending the JM command when the controller is in DISABLE or READY state only temporarily applies the setting. With the next boot of the controller the default setting will get applied again. Whereas sending the JM command when the controller is in CONFIGURATION state saves the setting in the controller's non-volatile memory).

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **JD** — Leave JOGGING state.

Example 1JM1 |*Enable keypad for controller #1.*

JR — Set/Get jerk time

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxJRnn** or **xxJR?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Jerk time value.

Range

xx	—	1 to 31
nn	—	**> 0.001** and **< 10^{12}**

Units

xx	—	None.
nn	—	Seconds.

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 sets the value for the maximum jerk time which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE or READY state.
Jerk is the derivative of acceleration. The jerk time defines the time to reach the needed acceleration. A longer jerk time reduces stress to the mechanics and smoothes motion.
In DISABLE or READY state, this command allows setting a new working parameter for the maximum jerk time. This value is not saved in the controller's memory and will be lost after reboot.

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 impossible (axis in movement).
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

AC	—	Set positioner acceleration.

Example

1JR0.05 |*Set controller #1 jerk time to 0.05 seconds.*

KD — Set/Get derivative gain

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxKDnn** or **xxKD?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Derivative gain value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None.
nn	—	Volt * second/preset unit.

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 sets the derivative gain of the PID control loop which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the derivative gain. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands

SC	—	Set closed loop state.
KI	—	Set integral gain.
KP	—	Set proportional gain.
KV	—	Set velocity feed forward.

Example 1KD0.015 |*Set controller #1 derivative gain to 0.015.*

KI — Set/Get integral gain

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxKInn** or **xxKI?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Integral gain value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None.
nn	—	Volt * preset unit/second.

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 sets the integral gain of the PID control loop which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the derivative gain. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands

SC	—	Set closed loop state.
KD	—	Set derivative gain.
KP	—	Set proportional gain.
KV	—	Set velocity feed forward.

Example 1KI0.015 |*Set controller #1 integral gain to 0.015.*

KP — Set/Get proportional gain

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxKPnn** or **xxKP?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Proportional gain value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None.
nn	—	Volt/preset unit

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 sets the proportional gain of the PID control loop which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the derivative gain. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands

SC	—	Set closed loop state.
KD	—	Set derivative gain.
KI	—	Set integral gain.
KV	—	Set velocity feed forward.

Example 1KP0.015 |*Set controller #1 proportional gain to 0.015.*

KV — Set/Get velocity feed forward

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	❑	❑	❑

Syntax **xxKVnn** or **xxKV?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Velocity feed forward value.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None.
nn	—	Volt * second/preset unit

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 sets the velocity feed forward of the PID control loop which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE state.
In DISABLE state, this command allows setting a new working parameter for the derivative gain. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands

SC	—	Set closed loop state.
KD	—	Set derivative gain.
KI	—	Set integral gain.
KP	—	Set proportional gain.

Example 1KV0.015 |*Set controller #1 velocity feed forward to 0.015.*

MM — Enter/Leave DISABLE state

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	■	■	❑	❑

Syntax **xxMMnn** or **xxMM?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Velocity feed forward value.

Range

xx	—	**0** to **31**
nn	—	**0** changes state from READY to DISABLE.
		1 changes state from DISABLE to READY.

Units

xx	—	None.
nn	—	None.

Defaults

xx	Missing:	Change to 0.
	Out of range:	Error B.
	Floating point:	Error A.
nn	Missing:	Error C.
	Out of range:	Error C.

Description

When the MM command is sent without preceding controller number or the controller number is 0, the MM command gets executed on all controllers.
MM0 changes the controller's state from READY to DISABLE. In DISABLE state the control loop is open and the motor is not energized . The encoder, though, is still read and the current position gets updated (on the SMC100CC only).
MM1 changes the controller's state from DISABLE to READY. The controller's set point position is set equal to its current position and the control loop gets closed (depending on the closed-loop state). The residual following error gets cleared from the buffer and the motor gets energized.

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

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.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **PW** — Enter/leave CONFIGURATION state.

Example MM0 |*All controllers go to DISABLE state.*

OH — Set/Get HOME search velocity

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxOHnn** or **xxOH?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	HOME high velocity.

Range

xx	—	**1** to **31**
nn	—	**> 10^{-6}** and **< 10^{12}**

Units

xx	—	None.
nn	—	Preset units/s.

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 sets the maximum velocity used by the controller for the HOME search.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

OR	—	Execute HOME search.
OT	—	Set HOME search time-out.

Example 1OH50 |*Set controller #1 HOME search velocity to 50 units/s.*

OR — Execute HOME search

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	❑	❑	❑	❑	❑

Syntax **xxOR**

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.
nn	Missing:	Error C.
	Out of range:	Error C.

Description

This command starts the execution of the HOME search as defined by the HT command.
When in NOT REFERENCED state, for instance after system start, any positioner must first get homed with the OR command before further motion commands can get executed.
The OR command gets accepted only in NOT REFERENCED state and only with no present hardware errors, except for end-of-run maybe. Refer to the TS command to get more information on the possible hardware errors.

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.
E	—	home sequence already started.
I	—	Execution not allowed in CONFIGURATION state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

HT	—	Set HOME search type.
OH	—	Set HOME search velocity.
OT	—	Set HOME search time-out.

Example 1OR |*Execute HOME search with controller #1.*

OT — Set/Get HOME search time-out

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxOTnn** or **xxOT?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	HOME time-out.

Range

xx	—	**1** to **31**
nn	—	**> 1** and **< 10^3**

Units

xx	—	None.
nn	—	Seconds

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 sets the time-out value for the HOME search. When the HOME search does not finish successfully before this time elapses, the HOME search will be aborted and an error gets recorded.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

HT	—	Set HOME search type.
OH	—	Set HOME search velocity.
OR	—	Execute HOME search.

Example 1OT2.2 |*Set controller #1 HOME time-out to 2.2 seconds.*

PA — Move absolute

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	❑	■	❑	❑

Syntax **xxPAnn** or **xxPA?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	New target position.

Range

xx	—	**1** to **31**
nn	—	**> SL** and **< SR**

Units

xx	—	None.
nn	—	Preset units.

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 PA command initiates an absolute move. When received, the positioner will move, with the predefined acceleration and velocity, to the new target position specified by **nn**.
The PA command gets only accepted in READY state, AND when the new target position is higher or equal to the negative software limit (SL), AND lower or equal to the positive software limit (SR).
To avoid any mismatch, the controller always rounds the new target position to the closest encoder position.

Returns

If the sign "**?**" takes place of **nn**, this command returns the target position 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.
G	—	Target position out of limits.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
J	—	Execution not allowed in DISABLE state.

Rel. Commands

PR	—	Move relative.
TH	—	Get set-point position.
TP	—	Get current position.
SU	—	Set encoder increment value.

Example 1PA2.2 |*Move positioner on controller #1 to absolute position 2.2 units.*

PR — Move relative

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	❑	■	❑	❑

Syntax **xxPRnn** or **xxPR?**

Parameters

Description

xx [int] — Controller address.

nn [float] — Displacement.

Range

xx — **1** to **31**

nn — **> SL** and **< SR**

Units

xx — None.

nn — Preset units.

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 PR command initiates a relative move. When received, the positioner will move, with the predefined acceleration and velocity, to a new target position **nn** units away from the current target position.
The PR command gets only accepted in READY state, AND when the distance of the positioner to the end of runs is larger than the commanded displacement.
To avoid any mismatch, the controller always rounds the new target position to the closest encoder position.

Returns

If the sign "**?**" takes place of **nn**, this command returns the target position 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.
G	—	Displacement out of limits.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
J	—	Execution not allowed in DISABLE state.

Rel. Commands

PA	—	Move absolute.
TH	—	Get set-point position.
TP	—	Get current position.
SU	—	Set encoder increment value.

Example

1PR2.2 |*Move positioner on controller #1 to a new position 2.2 units away from the current target position.*

PT — Get motion time for a relative move

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	■	■	■	❑

Syntax **xxPTnn**

Parameters

Description

xx [int] — Controller address.

nn [float] — Displacement.

Range

xx — **1** to **31**

nn — > $\mathbf{10^{-6}}$ and < $\mathbf{10^{12}}$

Units

xx — None.

nn — Preset units.

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 PT commands helps evaluating move times for an efficient program flow. When receiving the PT command, the controller returns the time, in seconds, necessary to execute a relative move of the displacement **nn** with the current working parameters (velocity, acceleration, etc.). The controller does not execute any motion.

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.

H — Execution not allowed in NOT REFERENCED state.

I — Execution not allowed in CONFIGURATION state.

Rel. Commands

PA — Move absolute.

PR — Move relative.

TH — Get set-point position.

TP — Get current position.

SU — Set encoder increment value.

Example

1PT2.2 |*Get time to move positioner on controller #1 by 2.2 units.*

|*Controller returns: 1PT0.25, means 0.25 seconds.*

PW — Enter/Leave CONFIGURATION state

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	❑	❑	❑	❑

Syntax **xxPWnn** or **xxPW?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Configuration mode.

Range

xx	—	**1** to **31**
nn	—	**1**: Go from NOT REFERENCED state to CONFIGURATION state. **0**: Go from CONFIGURATION state to NOT REFERENCED 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 NOT REFERENCED 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. set drive voltage, set Backlash compensation, etc.).
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 NOT REFERENCED.
The execution of a PW0 command may take up to 10 seconds. During that time the controller will not respond to any other command.

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

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.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **MM** — Enter/Leave DISABLE state.

Example 1PW1 |*Changes controller #1 to CONFIGURATION state*.

QI — Set/Get motor's current limits

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxQILnn**, **xxQIRnn**, **xxQITnn**, **xxQIL?**, **xxQIR?** or **xxQIT?**

Parameters

Description

xx [int]	—	Controller address.
Lmm [float]	—	Motor's peak current limit.
Rnn [float]	—	Motor's rms current limit.
Tpp [float]	—	Motor's rms current averaging time.

Range

xx	—	**1** to **31**
mm	—	**≥ 0.05** and **≤ 3.0**
nn	—	**≥ 0.05** and **≤ 1.5** and **≤ mm**
pp	—	**> 0.01** and **≤ 100**

Units

xx	—	None.
mm	—	Amperes.
nn	—	Amperes.
pp	—	Seconds.

Defaults

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

Description

QIL: Sets the controller's maximum or peak output current limit to the motor. When the controller detects a higher current than the peak current limit, it will generate a hardware error and a fault will be recorded.
QIR: Sets the controller's rms output current limit to the motor. The rms current limit must be lower than the peak current limit. When the controller's output current exceeds the rms current limit, it will generate a hardware error and a fault will be recorded.
QIT: Sets the controller's averaging period for rms current calculation. In general, the QIT command defines for how long time the actual motor current is allowed to exceed the rms output current limit.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands	DV	—	Set driver input voltage.
Example	1QIL0.75	|	*Set controller #1 current limit to 0.75 A.*
	1QIR0.25	|	*Set controller #1 rms current limit to 0.25 A.*
	1QIT2.5	|	*Set controller #1 rms averaging period to 2.5 s.*

RA — Get analog input value

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

Syntax **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 ±10 volts analog input. The converter is a ±7 bits analog to digital converter with ±0.15 volts of maximum offset and 5% full scale linearity. The resolution is 0.078125 volts.

Errors
A — Unknown message code or floating point controller address.
B — Controller address not correct.
D — Execution not allowed.
H — Execution not allowed in NOT REFERENCED state.
I — Execution not allowed in CONFIGURATION state.

Rel. Commands **SB** — Get TTL inputs.

Example
1RA |*Get controller axis #1 analog input.*
|*Controller returns: 1RA7.8125, means 7.8125 V.*

RB — Get TTL input value

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

Syntax **xxRB**

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 RB command returns the value of the TTL inputs. The returned decimal number represents the binary word made of all 4 inputs, where bit 0 is input 1, bit 1 is input 2, bit 2 is input 3, and bit 3 is input 4.
The TTL input value is 1 when the corresponding voltage on the pin is larger than 2.4 volts, and it is 0 when the corresponding voltage is below 0.8 volt. When the voltage is between these two values, the result is unreliable and can be 1 or 0.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.

Rel. Commands **RA** — Get analog input value.

Example 1RB |*Get TTL input value for controller #1.*
|*Controller returns: 1RB5, means input 0 and 2 are high, all others are low.*

RS — Reset controller

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	❑

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 a hardware reset of the controller, equivalent to a power-up.
To go from DISABLE or READY state to CONFIGURATION state, it is also needed to first reset the controller with the RS command, and then to change the controller's state with the PW1 command from NOT REFERENCED to CONFIGURATION.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Example 1RS |*Reset controller #1.*

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

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

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.
This address is ONLY used when the controller is configured to be a part of a daisy-chain configuration.
The SA command can only be sent to a controller:
- using RS-232-C communication,
- set as FIRST (dip switches)
- set in CONFIGURATION state (see PW command).

By default, in standard configuration the RS232C controller's address is 1 and the RS485 address is set to 2.
Only one controller at a time can be configured for RS485 communication.
Newport recommends using the supplied utility software for all controller configurations.
The SA command is of practical use only when not using this software.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Example

1PW1	*|Set controller in CONFIGURATION state*
1SA?	*|Controller returns: 1SA2, means controller's RS485 address is 2.*
1SA3	*|Set controller's RS-485 address to 3.*
1SA?	*|Controller returns: 1SA3, means controller's RS485 address is 3.*
1PW0	*|Set controller in NOT REFERENCED state*

SB — Set/Get TTL output value

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	■	■	■	■

Syntax **xxSBnn** or **xxSB?**

Parameters

Description

	xx [int]	—	Controller address.
	nn [int]	—	TTL output value.
Range	**xx**	—	**1** to **31**
	nn	—	**0** to **15**
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 SB command sets the value of the TTL outputs. The decimal number **nn** represents thereby the binary word made of all 4 outputs, where bit 0 is output 1, bit 1 is output 2, bit 2 is output 3, and bit 3 is output 4.
A 1 closes the open collector output transistor of the output. A 0 blocks the open collector output transistor of the output.

Returns If the sign "**?**" takes place of **nn**, this command returns the current TTL outputs 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.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.

Rel. Commands **RB** — Get TTL input value.

Example 1SB3 |*Close controller #1 TTL outputs 1 & 2 and open outputs 3 & 4.*

SC — Set/Get control loop state

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxSCnn** or **xxSC?**

Parameters

Description

xx [int]	—	Controller address.
nn [int]	—	Closed loop state.

Range

xx	—	**1** to **31**
nn	—	**1**: CLOSED loop control.
		0: OPEN loop control.

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

SC1 sets the controller to CLOSED loop control. This is the default.
SC0 sets the controller to OPEN loop control. Open loop control might be useful for defining stage parameters like friction compensation or velocity feed forward.

Returns

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

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Rel. Commands

KD	—	Set derivative gain.
KI	—	Set integral gain.
KP	—	Set proportional gain.
KV	—	Set velocity feed forward.

Example 1SC1 |*Set controller #1 to closed loop control.*

SE — Configure/Execute simultaneous started move

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	❑	■	❑	❑

Syntax **xxSEnn**, **xxSE?** or **SE**

Parameters

Description

	xx [int]	—	Controller address.
	nn [float]	—	New target position.

Range

xx	—	**0** to **31**
nn	—	**> SL** and **< SR**

Units

xx	—	None.
nn	—	Preset units.

Defaults

xx	Missing:	Change to 0.
	Out of range:	Error B.
	Floating point:	Error A.
nn	Missing:	Error C.
	Out of range:	Error C.

Description

The SE command allows starting a move on different controllers at the same time. The command xxSEnn sets a new target position for the controller **nn**. But different than the PA command, the move does not get executed immediately, but only after receipt of an SE command without preceding controller number and without following position value. When receiving the SE command, all controllers start a move to their new target position.
The xxSEnn command gets only accepted in READY state, AND when the new target position is higher or equal to the negative software limit (SL), AND lower or equal to the positive software limit (SR). To avoid any mismatch, the controller always rounds the new target position to the closest encoder position.
The SE command should not be confused with a synchronized move. With a synchronized move, all positioners start their motion simultaneously and have velocities, accelerations and jerk times which are limited to a rate which make all positioners start and complete their moves at the same time. The emphasis here is that they all start AND stop at the same time. The SE command starts a move on all controllers at the same time, but each positioner moves with its individually defined velocity and acceleration. So naturally, the different positioners don't complete their motion at the same time.

Returns

If the sign "**?**" takes place of **nn**, this command returns the target position value set by the SE command, which is not necessarily the same as the target position set by the PA 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.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.
J	—	Execution not allowed in DISABLE state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands	**PR**	—	Move relative.
	TH	—	Get set-point position.
	TP	—	Get current position.
	SU	—	Set encoder increment value.
Example	1SE2.2		*|Prepare controller #1 to move to absolute position 2.2 units.*
	2SE3.3		*|Prepare controller #2 to move to absolute position 3.3 units.*
	SE		*|All controllers start their programmed move, if any.*

SL — Set/Get negative software limit

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxSLnn** or **xxSL?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Negative software limit.

Range

xx	—	**1** to **31**
nn	—	**> -10^{12}** and **≤ 0**

Units

xx	—	None.
nn	—	Preset units.

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 sets the negative software limit which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE or READY state.
In DISABLE or READY state, this command allows setting a new working parameter for the negative software limit. It must be lower or equal to the set-point position. This value is not saved in the controller's memory and will be lost after reboot.
The software limits are useful to limit the travel range of a positioner. There is no possibility to disable software limits. For an almost infinite motion, for instance with a rotation stage, set the lowest possible value, which is: -2147000000 * "encoder increment value" (see SU command). For instance if the encoder increment value is 0,0005, this limit is -1073500.

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

SR	—	Set positive software limit.

Example

1SL-100 |*Set controller #1 negative software limit to –100 units.*

SR — Set/Get positive software limit

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxSRnn** or **xxSR?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Positive software limit.

Range

xx	—	**1** to **31**
nn	—	**≥ 0** and **< 10^{12}**

Units

xx	—	None.
nn	—	Preset units.

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 sets the positive software limit which can than be saved in the controller's nonvolatile memory using the PW command. It is also the default value that will be used unless a different value is set in DISABLE or READY state.
In DISABLE or READY state, this command allows setting a new working parameter for the positive software limit. It must be larger or equal to the set-point position. This value is not saved in the controller's memory and will be lost after reboot.
The software limits are useful to limit the travel range of a positioner. There is no possibility to disable software limits. For an almost infinite motion, for instance with a rotation stage, set the largest possible value, which is: 2147000000 * "encoder increment value" (see SU command). For instance if the encoder increment value is 0,0005, this limit is 1073500.

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands

SL	—	Set negative software limit.

Example

1SR100 |*Set controller #1 positive software positive to 100 units.*

ST — Stop motion

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	❑	■	■	■	❑

Syntax **[xx]ST**

Parameters

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

Range **xx** — **0** to **31**

Units **xx** — None.

Defaults **xx** Missing: Change to 0.
Out of range: Error B.
Floating point: Error A.

Description The ST command is a safety feature. It stops a move in progress by decelerating the positioner immediately with the acceleration defined by the AC command until it stops.
The xxST command with preceding controller address stops a move in progress on controller xx. The ST command without preceding controller address stops the moves on ALL controllers.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.

Example ST |*Stop moves on all controllers.*

SU — Set/Get encoder increment value

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxSUnn** or **xxSU?**

Parameters

Description

	xx [int]	—	Controller address.
	nn [float]	—	Equivalent units to one encoder count.
Range	**xx**	—	**1** to **31**
	nn	—	**> 10^{-6}** and **< 10^{12}**
Units	**xx**	—	None.
	nn	—	Units.
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 SU command sets the value for one encoder count. It defines also the system of units for all other parameters like travel limits, velocities, accelerations, etc. Therefore, it is the first parameter to be defined for any positioner.

Example: For a positioner with an encoder resolution of 1 µm, the command xxSU0.001 sets 1 encoder count = 1 µm = 0.001 unit or 1 unit = 1 mm.

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.
H	—	Execution not allowed in NOT REFERENCED state.
J	—	Execution not allowed in DISABLE state.
K	—	Execution not allowed in READY state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
W	—	Command not allowed for SMC100PP version.

Example 1SU7.5e-6 |*Set controller #1 encoder increment to 7.5 * 10^{-6} units.*

TB — Get command error string

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

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 @.*

|*Controller returns: 1TB@ No error, @ means no error.*

TE — Get last command error

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	❑

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.
E	—	Home sequence already started.
F	—	ESP stage name unknown.
G	—	Displacement out of limits.
H	—	Command not allowed in NOT REFERENCED state.
I	—	Command not allowed in CONFIGURATION state.
J	—	Command not allowed in DISABLE state.
K	—	Command not allowed in READY state.
L	—	Command not allowed in HOMING state.
M	—	Command not allowed in MOVING state.
N	—	Current position out of software limit.
S	—	Communication Time Out.
U	—	Error during EEPROM access.
V	—	Error during command execution.
W	—	Command not allowed for PP version.
X	—	Command not allowed for CC version.

TH — Get set-point position

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

Syntax **xxTH**

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 TH command returns the value of the set-point or theoretical position. This is the position where the positioner should be. In MOVING state, the set-point position changes according to the calculation of the motion profiler. In READY state, the set-point position is equal to the target position.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed.
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.

Rel. Commands TP — Get current position.

Example 1TH |*Get set-point position of controller #1.*
|*Controller returns: 1TH0, set-point position = 0 units.*

TP — Get current position

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

Syntax **xxTP**

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 TP command returns the value of the current position. This is the position where the positioner actually is according to his encoder value. In MOVING state, this value always changes. In READY state, this value should be equal or very close to the set-point and target position.
Together with the TS command, the TP command helps evaluating whether a motion is completed.

Errors

A	—	Unknown message code or floating point controller address.
B	—	Controller address not correct.
D	—	Execution not allowed
H	—	Execution not allowed in NOT REFERENCED state.
I	—	Execution not allowed in CONFIGURATION state.

Rel. Commands **TH** — Get set-point position.

Example 1TP |*Get current position of controller #1.*
|*Controller returns: 1TP0, actual position = 0 units.*

TS — Get positioner error and controller state

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

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) represent the positioner error in Hexadecimal. The last two characters (ef) represent the controller state.

Error code (ABCD): Convert each hexadecimal to a binary:

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1111	1110	1101	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001	0000

Each bit represents one possible error:

A				B				C				D			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
• Not used	• Not used	• Not used	• Not used	• Not used	• Not used	• 80 W output power exceeded	• DC voltage too low	• Wrong ESP stage	• Homing time out	• Following error	• Short circuit detection	• RMS current limit	• Peak current limit	• Positive end of run	• Negative end of run

Examples:

- Error map 0000 = No errors
- Error map 0013 = Short circuit detection, Positive end of run, negative end of run
- Error map 004C = Homing time out, RMS current limit, Peak current limit

Controller states (ef):

- **0A**: NOT REFERENCED from reset.
- **0B**: NOT REFERENCED from HOMING.
- **0C**: NOT REFERENCED from CONFIGURATION.
- **0D**: NOT REFERENCED from DISABLE.
- **0E**: NOT REFERENCED from READY.
- **0F**: NOT REFERENCED from MOVING.
- **10**: NOT REFERENCED ESP stage error.
- **11**: NOT REFERENCED from JOGGING.
- **14**: CONFIGURATION.
- **1E**: HOMING commanded from RS-232-C.
- **1F**: HOMING commanded by Keypad.
- **28**: MOVING.
- **32**: READY from HOMING.
- **33**: READY from MOVING.
- **34**: READY from DISABLE.
- **35**: READY from JOGGING.
- **3C**: DISABLE from READY.
- **3D**: DISABLE from MOVING.
- **3E**: DISABLE from JOGGING.
- **46**: JOGGING from READY.
- **47**: JOGGING from DISABLE.

NOTE

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.

THE ERROR "WRONG ESP STAGE" GETS ONLY DETECTED DURING THE BOOTING OF THE CONTROLLER. WHEN READ THE ERROR IS CLEARED.

With no errors in the error buffer the color of the LED will change from red to either green or orange depending on the controller state.

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.*
			|Controller returns: 1TS00000A, no errors and NOT REFERENCED from reset.

VA — Set/Get velocity

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxVAnn** or **xxVA?**

Parameters

Description

xx [int]	—	Controller address.
nn [float]	—	Velocity value.

Range

xx	—	**1** to **31**
nn	—	> $\mathbf{10^{-6}}$ and < $\mathbf{10^{12}}$

Units

xx	—	None.
nn	—	Preset units/s.

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 sets the maximum velocity value which can than be saved in the controller's nonvolatile memory using the PW command. This is the maximum velocity that can be applied to the mechanical system. It is also the default velocity that will be used for all moves unless a lower value is set in DISABLE or READY state.
In DISABLE or READY state, this command sets the velocity used for the following moves. Its value can be up to the programmed value in CONFIGURATION state. This value is not saved in the controller's memory and will be lost after reboot.

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.

Rel. Commands **AC** — Set positioner acceleration.

Example 1VA50 |*Set controller #1 velocity to 50 units/s.*

VB — Set/Get base velocity

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	■	■	❑	❑

Syntax **xxVBnn** or **xxVB?**

Parameters

Description

xx [int]	—	Axis number.
nn [int]	—	Base velocity.

Range

xx	—	**1** to **31**
nn	—	**≤ 0** and **≥ value fixed by VA command**.

Units

xx	—	None.
nn	—	Units.

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 sets the profile generator base velocity.

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.
H	—	Execution not allowed in NOT REFERENCED state.
L	—	Execution not allowed in HOMING state.
M	—	Execution not allowed in MOVING state.
X	—	Command not allowed for SMC100CC version.

Rel. Commands **VA** — Set velocity.

EXAMPLE 1VB0.1 |*Set axis #1 base velocity to 0.1 units/s.*

VE — Get controller revision information

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	■

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 **TP** — Get current position.

Example
1VE |*Get controller #1 revision information.*
|*Controller returns 1VE SMC - Controller-driver version 1.00r.*

ZT — Get all configuration parameters

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	■	■	■	■	■	❑

Syntax **xxZT**

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 ZT command returns the list of all current configuration parameters.
The ZT command allows a quick review of all current stage parameter and simplifies the configuration of non Newport stages, for instance by using Hyper Terminal file transfer.

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

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

Example
```
1ZT          |Get controller #1 configuration data.
1PW1
1AC320.000000
1BA0.000000
…
1VA80.000000
1ZX3
1PW1
```

ZX — Set/Get ESP stage configuration

Usage	Not Ref.	Config.	Disable	Ready	Motion	Jogging
	❑	■	❑	❑	❑	❑

Syntax **xxZXnn** or **xxZX?**

Parameters

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

Range
- **xx** — **1** to **31**
- **nn** — **1** disable ESP stage check.
  2 update ESP stage information.
  3 enable ESP stage check.

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 ZX command allows loading ESP stage data to the controller's flash memory and enables/disables ESP stage check during power-up. ESP refers to Newport stages with an EEPROM (called ESP chip), that contains all stage information like motor type, travel limits, maximum velocity, maximum acceleration, etc.

The command ZX2 reads the parameters from the ESP stage and saves them to the controller's flash memory. When using the SMC100CC/PP controller with Newport ESP compatible stages this is the fastest way of doing the stage configuration. When not using the Newport supplied utility software, just send the ZX2 command, and you're done.

The command ZX3 enables the ESP stage check. When enabled, the controller checks at each power-up whether the connected stage is the same as the one recorded in the controller flash memory. If not, it memorizes an error. The ESP stage check is recommended with all Newport ESP compatible stages.

The command ZX1 disables the ESP stage check. When disabled, the controller will not check the connected stage and the stage reference is set to UNKNOWN.

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

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.
- H — Execution not allowed in NOT REFERENCED state.
- J — Execution not allowed in DISABLE state.
- K — Execution not allowed in READY state.
- L — Execution not allowed in HOMING state.
- M — Execution not allowed in MOVING state.

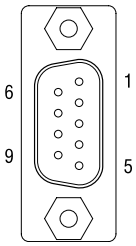
Example 1ZX? |*Controller returns: 1ZXURS100CC, means URS100CC stage.*

6 Connector Pinout

6.1 DC IN and DC OUT (Female Ø 2.1 x Ø 5.5 x 11 mm)

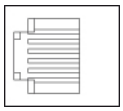
Pin #	Description
Center	+48 VDC
Outer	GND

6.2 RS-232-C (Male Sub-D9)



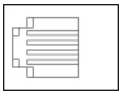
Pin #	Description
1	Shorted together with 4 and 6
2	TX
3	RX
4	Shorted together with 1 and 6
5	GND
6	Shorted together with 1 and 4
7	Shorted together with 8
8	Shorted together with 7
9	Not connected

6.3 RS-485 IN and RS-485 OUT (Female RJ11-6/6)



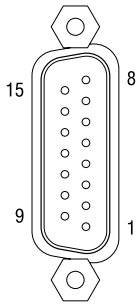
Pin #	Description
1	GND
2	RX+
3	RX-
4	TX-
5	TX+
6	GND

6.4 Keypad (Female RJ9-4/4)



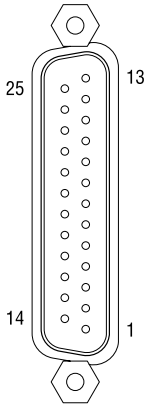
Pin #	Description
1	+12 VDC
2	Tx
3	Rx
4	GND

6.5 GPIO (Female Sub-D15)



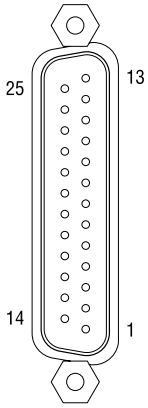
Pin #	Description
1	Analog in
2	GND
3	OUT1 (Open collector, 30 V/40 mA Max.)
4	OUT2 (Open collector, 30 V/40 mA Max.)
5	OUT3 (Open collector, 30 V/40 mA Max.)
6	OUT4 (Open collector, 30 V/40 mA Max.)
7	GND
8	IN1 (2.21 kΩ pull up to 5 V)
9	IN2 (2.21 kΩ pull up to 5 V)
10	IN3 (2.21 kΩ pull up to 5 V)
11	IN4 (2.21 kΩ pull up to 5 V)
12	GND
13	In Motion (Open collector)
14	Not Referenced (Open collector)
15	GND

6.6 DC Motor (Female Sub-D25)



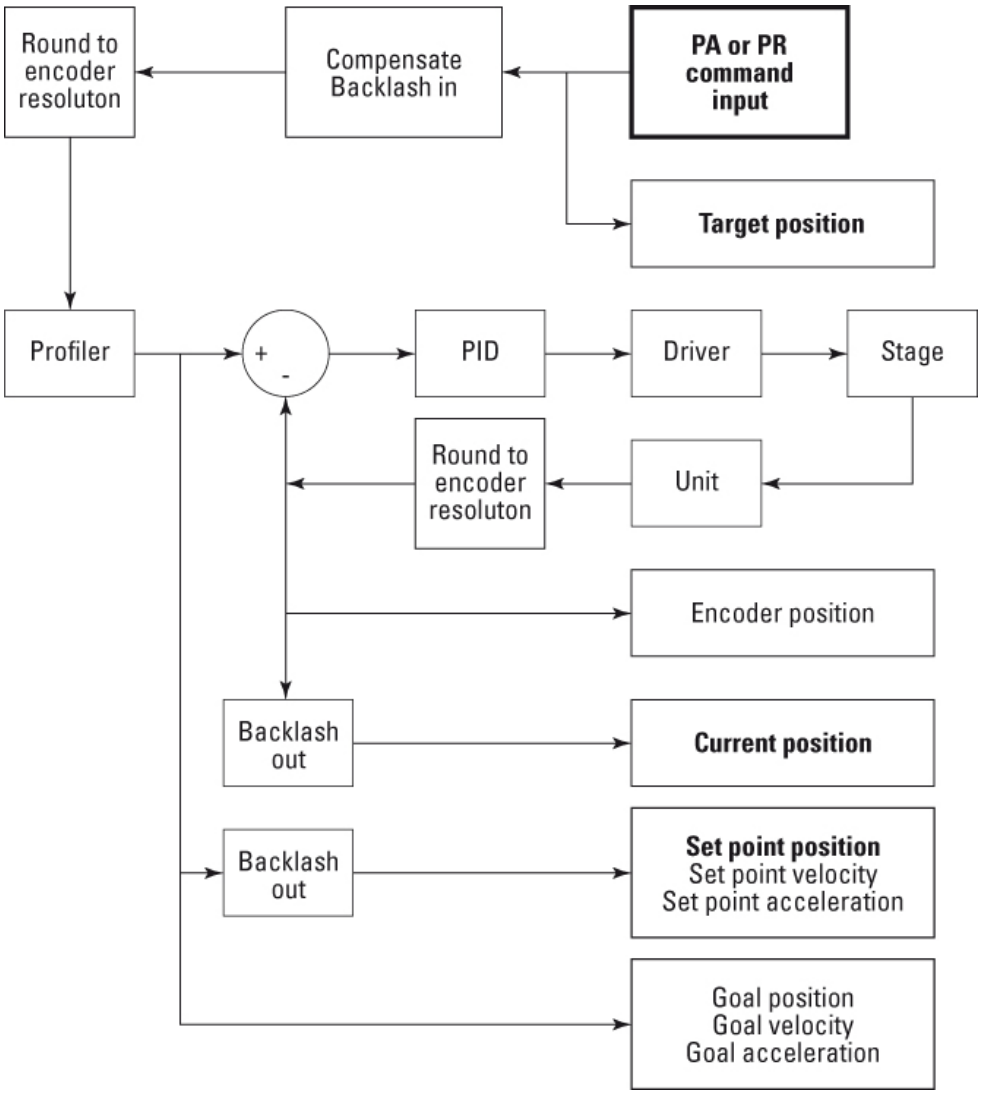
Pin #	Description
1	Not connected
2	Not connected
3	Not connected
4	Not connected
5	MOTOR+
6	MOTOR+
7	MOTOR-
8	MOTOR-
9	Not connected
10	Not connected
11	Not connected
12	Not connected
13	ZM
14	GND
15	VI
16	GVD
17	EoR+
18	EoR-
19	VA
20	VB
21	+5 V
22	GVD
23	/VA
24	/VB
25	/VI

6.7 Stepper Motor (Female Sub-D25)



Pin #	Description
1	Winding 1+
2	Winding 1+
3	Winding 1-
4	Winding 1-
5	Winding 2+
6	Winding 2+
7	Winding 2-
8	Winding 2-
9	Not connected
10	Not connected
11	Not connected
12	Not connected
13	ZM
14	GND
15	VI or N.C. if no encoder
16	GND
17	EoR+
18	EoR-
19	VA or N.C. if no encoder
20	VB or N.C. if no encoder
21	+5 V
22	GND
23	/VA or N.C. if no encoder
24	/VB or N.C. if no encoder
25	/VI or N.C. if no encoder

7 Backlash Compensation

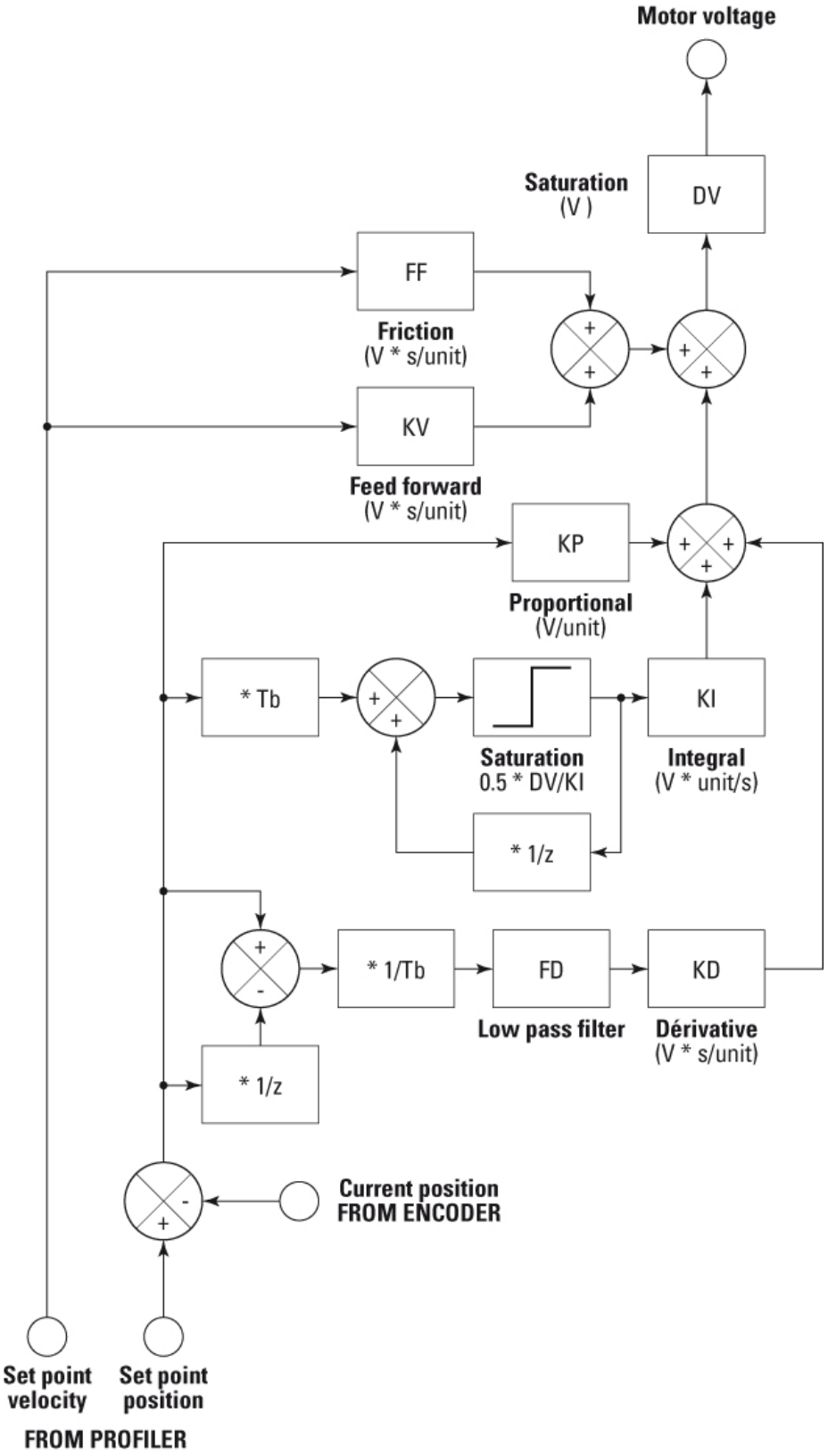


- Target position is read by PA command.
- Current position is read by TP command.
- Set-point position is read by TH command.
- Encoder resolution is set/read by the SU command.
- Backlash is set/read by the BA command.

8 ESP Stages

ESP refers to Newport stages with an EEPROM (ESP chip), that contains all stage information like motor type, travel limits, maximum speeds, etc. The SMC100CC/PP is capable reading this information from the stage and can save it to the controller's flash memory. This minimizes the stage configuration time and possible errors during configuration. The SMC100CC/PP can also be configured to confirm at each power-up that the connected stage is the same as the one recorded in the controller's memory, which is another safety feature.

9 PID Control Loop Structure



10 Maintenance and Service

10.1 Enclosure Cleaning

The SMC100CC/PP Controller/Driver should only be cleaned with a lightly damped cloth or sponge with a soapy water solution. Do not use an acetone or alcohol solution, this will damage the finish of the enclosure.

10.2 Obtaining Service

The SMC100CC/PP Controller/Driver contains no user serviceable parts. To obtain information regarding factory service, contact Newport Corporation or your Newport representative. Please have the following information available:

- Instrument model number (on front panel).
- Instrument serial number (on rear panel) or original order number.
- Description of the problem.

If the instrument is to be returned to Newport Corporation, you will be given a Return Number, which you should reference in your shipping documents.

Complete a copy of the Service Form as represented on the next page and include it with your shipment.

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 MKS I 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