DL Controller Series

Single-Axis Motion Controller for Delay Line Stages

Newport™

User's Manual

V1.0.x

Warranty

Newport Corporation warrants that this product will be free from defects in material and workmanship and will comply with Newport's 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 Newport's option.

To exercise this warranty, write or call your local Newport office or representative, or contact Newport headquarters in Irvine, California. 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 Newport's 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. NEWPORT CORPORATION SHALL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM THE PURCHASE OR USE OF ITS PRODUCTS.

Original instructions.

Table of Contents

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
☒ Machinery Directive – 2006/42/EC
☒ 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)
☒EN ISO 12100:2010 Safety of Machinery – General Principles of Design – Risk Assessment and Risk Reduction

Emissions:

☒EN 55011: 2016 +A1:2017 (4) Industrial, scientific and medical equipment - Radio-frequency disturbance characteristics - Limits and methods of measurement

Immunity:

☒EN 61000-4-2:2009 EMC/Electrostatic Discharge Immunity Test
☒EN 61000-4-3:2006+A2:2010 EMC/Radiated Radio Frequency Electromagnetic Field Immunity Test
☒EN 61000-4-4:2012 EMC/Electrical Fast Transient/Burst Immunity Test
☒EN 61000-4-5:2014+A1:2017 EMC/Surge Immunity Test

Manufacturers Name: MKS Instruments, Inc., 2 Tech Drive, Andover, MA 01810 USA

Authorized Representatives Name & Location: _____

Equipment Type/Description: **High-Performance Delay Line Stages with Single-Axis Motion Controller/Driver.**

Model Number(s) (6): **DL125/225/325.**

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: 11/6/2023

Le Cointe Hervé – Quality Director

4) Class A, Group 2
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
☒ Machinery Directive – 2006/42/EC
☒ 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)
☒BS EN ISO 12100:2010 Safety of Machinery – General Principles of Design – Risk Assessment and Risk Reduction

Emissions:
☒EN 55011: 2016 +A1:2017 (4) Industrial, scientific and medical equipment - Radio-frequency disturbance characteristics - Limits and methods of measurement

Immunity:
☒BS EN 61000-4-2:2009 EMC/Electrostatic Discharge Immunity Test
☒BS EN 61000-4-3:2006+A2:2010 EMC/Radiated Radio Frequency Electromagnetic Field Immunity Test
☒BS EN 61000-4-4:2012 EMC/Electrical Fast Transient/Burst Immunity Test
☒BS EN 61000-4-5:2014+A1:2017 EMC/Surge Immunity Test

Manufacturers Name: MKS Instruments, Inc., 2 Tech Drive, Andover, MA 01810 USA

Authorized Representatives Name & Location: _____/_____

Equipment Type/Description: **High-Performance Delay Line Stages with Single-Axis Motion Controller/Driver.**

Model Number(s) (6): **DL125/225/325.**

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: 11/6/2023

Le Cointe Hervé – Quality Director

4) Class A, Group 2
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.

MKS Instruments, Inc.
Andover, MA USA

Page **1** of **1**

Document Number: MKS-GPC-TM-20063

Preface

Confidentiality & Proprietary Rights

Reservation of Title

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

Newport 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 Newport. Customer shall not challenge, or cause any third party to challenge, the rights of Newport.

Preservation of Secrecy and Confidentiality and Restrictions to Access

Customer shall protect the Newport Programs and Related Materials as trade secrets of Newport, and shall devote its best efforts to ensure that all its personnel protect the Newport Programs as trade secrets of Newport Corporation. Customer shall not at any time disclose Newport's 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 Newport 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 Newport Programs; or (4) approved by Newport for release without restriction.

Sales, Tech Support & Service

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

Sales
Tel.: (800) 222-6440
e-mail: sales@newport.com

Technical Support
Tel.: (800) 222-6440
e-mail: tech@newport.com

Service, RMAs & Returns
Tel.: (800) 222-6440
e-mail: service@newport.com

Europe
MICRO-CONTROLE Spectra-Physics S.A.S
9 rue du bois sauvage
91055 Évry CEDEX
France

Sales France
Tel.: +33 (0)1.60.91.68.68
e-mail: france@newport.com

Sales Germany
Tel.: +49 (0) 61 51 /708 – 0
e-mail: germany@newport.com

Sales UK
Tel.: +44 (0)1635.521757
e-mail: uk@newport.com

Technical Support
e-mail: tech_europe@newport.com

Service & Returns
Tel.: +33 (0)2.38.40.51.55

Service Information

The user should not attempt any maintenance or service of the SMC100 Controller/Driver and its accessories beyond the procedures outlined in this manual. Any problem that cannot be resolved should be referred to Newport Corporation. When calling Newport regarding a problem, please provide the Tech Support representative with the following information:

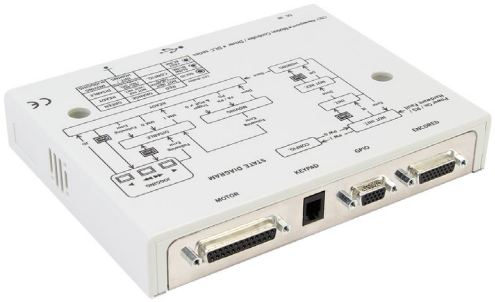
- Your contact information.
- System serial number or original order number.
- Description of problem.
- Environment in which the system is used.
- State of the system before the problem.
- Frequency and repeatability of problem.
- Can the product continue to operate with this problem?
- Can you identify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

Newport Corporation RMA Procedures

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

Packaging

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



Single-Axis Motion Controller for Delay Line Stages DL Controller

1.0 Introduction

1.1 Definitions and Symbols

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

1.1.1 General Warning or Caution

Figure 1: General Warning or Caution Symbol.

The Exclamation Symbol in Figure 1 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

Figure 2: Electrical Shock Symbol.

The Electrical Shock Symbol in Figure 2 may appear on labels affixed to the DL 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

Figure 3: 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.

1.2 Warnings and Cautions

The following are definitions of the Warnings, Cautions and Notes that may be used in this manual to call attention to important information regarding personal safety, safety and preservation of the equipment, or important tips.

WARNING

Situation has the potential to cause bodily harm or death.

CAUTION

Situation has the potential to cause damage to property or equipment.

NOTE

Additional information the user or operator should consider.

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 DL Controller/Driver. 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.

2.0 System Overview

2.1 General Description

The DL Controller is a single axis Controller /driver capable to drive stages with linear motors up to 80W. Its reduced command set and the auto configuration setup from the SmartStage ID and the embedded Stages DataBase makes it very easy to use.

At power up, the communication is set automatically with USB CDC Device Class Driver from TI and the Controller is seen by the PC as a virtual serial port.

To simplify programming, the DL Controller will read from its serial port until it gets a line terminator. Then, it will analyze the first characters to see if it is a known command. If so, it will try to execute it or it will memorize an error.

After the execution of the command, if some characters remain in the input string they will be thrown away.

Each command will handle properly the memorization of related errors, refer to the command set.

NOTE

If you have multiple DL products, make sure the name indicated on the sticker of the controller (DLxxx) matches with the one indicated on the sticker of the stage.

2.2 Part Numbers

Product	Description
DL Controller	Single-axis motion Controller/Driver for linear motors.
SMC-RC	Remote control keypad for DL Controller.
DL-PS	100 to 120 W power supply for DL Controller.

2.3 DL Controller

2.3.1 Contents of Delivery

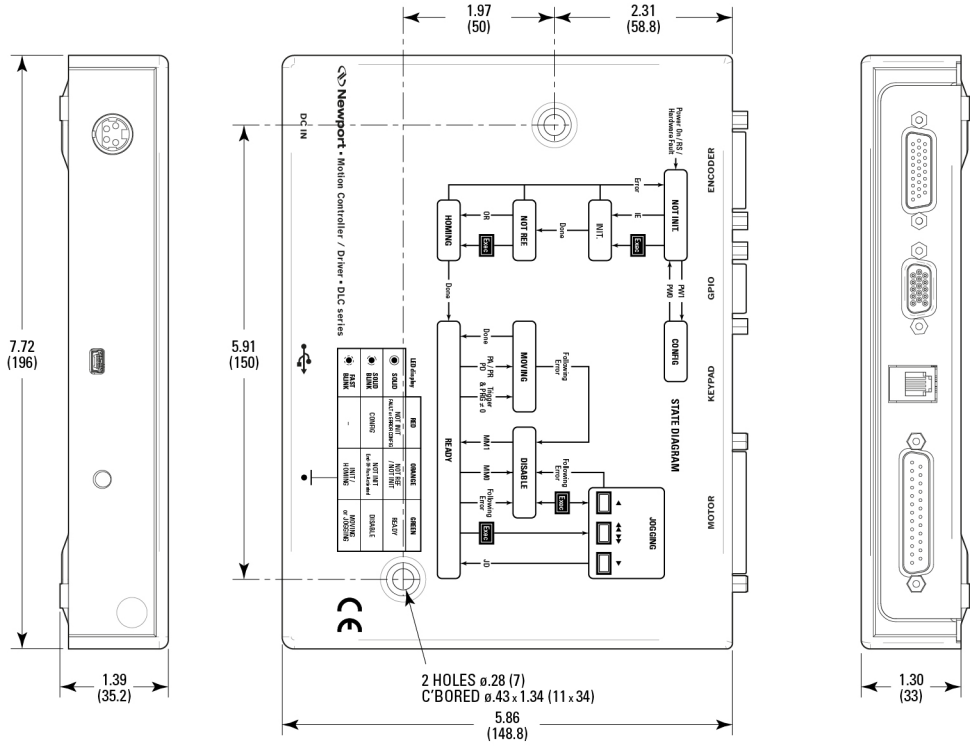
- DL Controller box

2.3.2 Specifications

General Description	Single-axis motion Controller/Driver for Delay Line stages
Control Capability	Linear motors, closed loop operation
Motor Output	3 Motor phases Maximum Voltage: 42 V Full Scale Current: 6.4 A (1) Max Peak Current: 6 A (1) Rms Current: 2.5 A Max Power: 80 W ESP interface Thermistor input (spare 78.125 kHz PWM switching frequency
Control loop	Floating point digital PID loop with acceleration feedforward 9.766 kHz servo rate Low pass filter 2nd order on PID output
Motion	Point-to-point motion with S-gamma profile and jerk time control
Computer interface	USB Virtual COM port
Programming	60+ intuitive, 2 or 3 letter ASCII commands
General purpose I/O	TTL Start Motion input trigger TTL Motion done output trigger (push-pull) TTL PCO output trigger (push-pull) RS-422 AquadB output 2 x Single ended ADC inputs (16bits, +/-5V, Z=47K)
Dedicated inputs	Sin/Cos 1Vpp + Index encoder inputs, max. 2 MHz rate Forward and reverse limits
Status display	Two color LED

(1) The Full-Scale Current is not a power specification but the maximum current measure capacity used to calculate the Scaling Force (FMS) parameter.

2.3.3 Dimensions



MODEL SHOWN: DL CONTROLLER

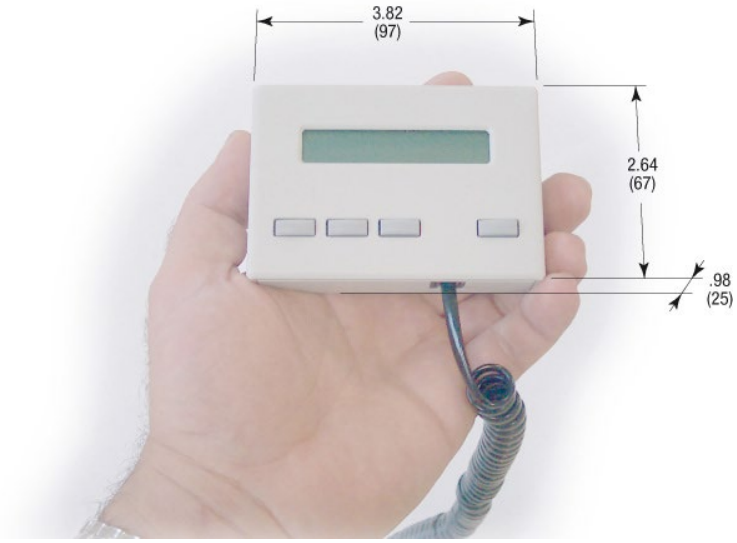
DIMENSIONS IN INCHES (AND MILLIMETERS)

2.4 SMC-RC

2.4.1 Specifications

General Description	Remote control keypad for DL Controller
Display	1 line x 16 characters LCD display for position and short action description of Exec. button depending on Controllers state
Function of push buttons (from left to right)	– Jog left – High jog velocity (when pressed together with left or jog right) – Jog right – Exec. (function as indicated in display depending on Controllers state)
Cable	0.5 m helix cable, both sides terminated with RJ11-4/4 connectors

2.4.2 Dimensions

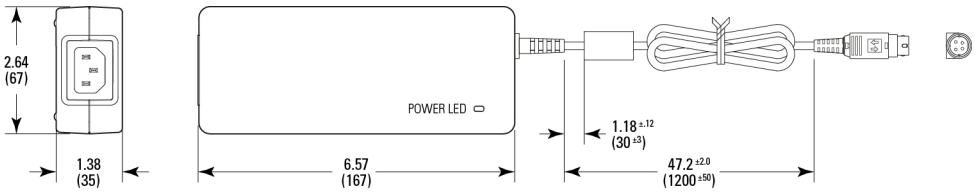


2.5 DL-PS

2.5.1 Specifications

AC Input	100–240 VAC, 47–63 Hz, 1.9 A
DC Output	48 V, 80 W max., 1.87A, <240 mVp-p ripple and noise
Load and line regulation	Better than 2%
Connector	Kycon 4PPX-4P or equivalent

2.5.2 Dimensions

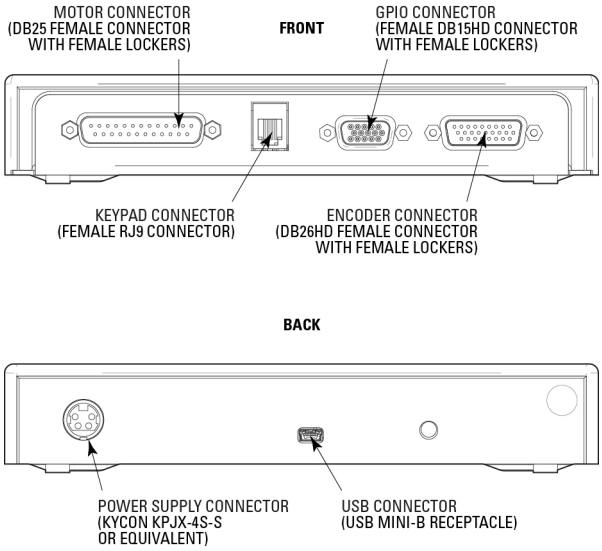


MODEL SHOWN: DL-PS
DIMENSIONS IN INCHES (AND MILLIMETERS)

2.6 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

2.7 Connector Identification



2.7.1 Front Side

POWER	4 pin female jack (Kycon KPJX-4S-S or equivalent)
USB	USB Mini-B Receptacle for computer communication
LED	LED: Status LED

2.7.2 Back Side

ENCODER	Sub-D15F: Encoder connection
GPIO	Sub-D15F: General purpose inputs/outputs
KEYPAD	RJ9F: For SMC-RC remote display and jog keypad.
MOTOR	Sub-D25F: Motor connection

2.8 USB Communication Settings

Communication parameters are preset in the DL Controller and do not require any configuration:

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

3.0 Getting Started

This section guides the user through the proper set-up of the DL Controller motion control system.

Carefully unpack and visually inspect the DL Stage, DL Controller, the SMC-RC keypad (if any) and DL-PS power supply for any damage. Place all components on a flat and clean surface.

NOTE

Default "Force & Masses" parameter (FM) of the DL Controller have been set in a factory without any load on the stage. In case of significant load added to the stage, these parameters must be adjusted prior to attempt any motion of the stage. To do so, the DL Controller must be connected to a Computer through USB connection.

Once this parameter is set, the DL Controller the DL Controller can be controlled eigher from a PC (USB) or ONLY in local mode through the SMC-RC keypad.

When using the SMC-RC keypad and NOT from a computer, you can skip this section and continue reading in chapter 0, DL Controller with SMC-RC keypad. If not already done, carefully unpack and visually inspect the Controllers and the stages for any damage. Place all components on a flat and clean surface.

3.1 Connections

To power the system, follow the sequence below:

1. Connect the DL Satge to the DL Controller (MOTOR and ENCODER connectors).
2. If any, connect the SMC-RC to the DL Controller (KEYPAD connector).
3. Connect the DL PC to the DL Controller (DC IN connector).
4. Connect the DL-PC to power.

The DL Controller is automatically powered on. Wait for the boot time (few seconds). DL Controller LED should turn solid orange

NOTE

if DL Controller LED turns red check Troubleshooting chapter below or I/O Errors chapter of the DLcontroller GUI manual

3.2 Communication Settings

3.2.1 Requirements

Provided DLS GUI and Dll are designed for Personal Computers running Windows 7, 8.1 and 10 platforms (32 bit & 64 bit).

Check 2.0 Host computer configuration chapter of the DL Controller GUI Manual to verify computer setting

3.2.2 PC USB Communication

Using the USB cable (provided) connect DL Controller (USB connector) to your PC.

If not already installed Windows will ask for the compatible driver. Follow the procedure below to install it. Otherwise skip the section and continue reading in chapter 3.2.3.

3.2.2.1 Driver Installation

Consult **Getting started** chapter of the DL Controller GUI Manual (chapter 3) to install and run the provided GUI

3.2.3 Configuring the Controller

Consult **Settings** chapter of the DL Controller GUI Manual (chapter 4.6) to set parameters

3.2.3.1 Modified payload

All DL Controller parameters have been set in the factory per the associated stage without any payload. In case of significant load added to the stage, 2 parameters must be adjusted depending on the payload placed on the stage:

- The Scaling Force parameter through the "FMP" command.
- The Acceleration parameter through the command "AC" using the formula:

$$\text{Accelaration} = \frac{\text{Force Limit (FML)}}{\text{Force Limit (FML)} + \text{Paylod (FMP)}}$$

Default value: 4000. See AC chapter below.

4.0 DL Controller with SMC-RC Keypad

The SMC-RC keypad allows basic use of the DL Controller without a computer. It features a 16 characters position display and four push buttons for configuration, jogging, homing, and enabling/disabling motors. It can be also used in parallel to a computer control.

If not already done, carefully unpack and visually inspect the DL Controller, the SMC-RC keypad, all stages and all accessories for any damage. Place all components on a flat and clean surface.

1. Connect the SMC-RC to the DL Controller (KEYPAD connector).
2. Connect your stage to the DL Controller (MOTOR and ENCODER connectors).
3. Connect the DL PC to the DL Controller (DC IN connector).
4. Connect the DL-PC to power.

During the initialization, the DL Controller checks if a SMC-RC keypad is connected. If so, it checks whether all buttons are open (not pressed). If not, an error message gets generated.

NOTE

The DL Controller does not recognize an SMC-RC after the initialization. Also, disconnecting the SMC-RC from the Controller and reconnecting without reinitializing the Controller does not work.

To reinitialize the DL Controller, temporarily disconnect from power and reconnect again, or send the RS command.

After successful initialization, the Controller is still in the NOT INITIALIZED state and the display displays **+0.00000 INI** (for more details about the DL Controller states, please refer to section 5.1).

Press the Exec. button to initialize the stage, when done, the display shows: **+0.00000 HOM**.

Press the Exec. button to home the stage. The stage starts moving to its home position. When done, the display shows: **+0.00000 JOG**

The digital value indicates the current position of the stage. The default units for Newport positioners are millimeters for linear stages and actuators, and degrees for rotation stages.

Pressing the Exec. button again gets the Controller to the JOGGING state and the display changes to **+0.00000 DIS**.

The jog buttons “<”, “<<>>”, and “>” are now enabled. Pressing the “<” (jog left) or “>” (Jog right) button starts a motion at slow velocity and with slow acceleration. Releasing the button stops the motion. These slow speed motion are ideal for precise adjustments.

Pressing the “<” (jog left) or “>” (Jog right) button and the “<<>>” (high speed) simultaneously starts a high speed motion. These high speed motion are ideal for coarse adjustments.

High velocity

and Low jog velocity are set through the JV Command:

High acceleration/decelartion

Low jog acceleration/decelartion are set through the JA Command:

NOTE

Any jog motion always respects the software limits (see settings in the software utility or with the SL and SR commands). When approaching a software limit, the Controller decelerates with the programmed acceleration even if the jog buttons are pressed.

Pressing the Exec. button when the three most right letters are DIS, gets the Controller to the DISABLE 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. 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 DISABLE state to the JOGGING state, press the Exec. button again.

The buttons of the keypad can get disabled by the JD command.

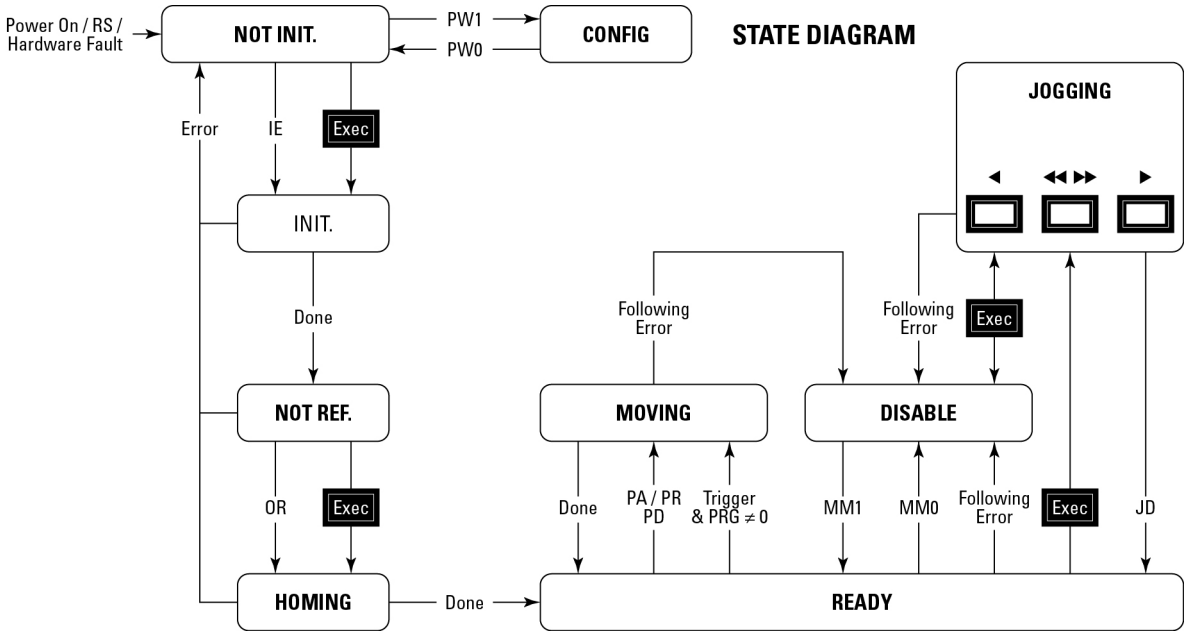
NOTE

The keypad does not allow stopping any motion started from a computer (all buttons are disabled when the Controller is in MOVING state). To take computer control when the Controller is in JOGGING state the Controller must first get to the READY state (change state from the software utility or by using the JD command).

5.0 Programming

5.1 State Diagram

For a safe and consistent operation, the DL Controller uses 8 different operation states: Not initialized, Not referenced, Configuration, Homing, Ready, Disable, Jogging and Moving. In each state, only specific commands are accepted by the DL Controller. Therefore, it is important to understand the state diagram below and which commands and actions cause transition between the different states. Also see chapter 5.5 for command/state information.



End of Runs encountered in the following state:

NOT INITIALIZED:	If hardware faults or wrong parameters then SOLID RED.
NOT INITIALIZED:	If everithing is OK then SOLID ORANGE.
NOT INITIALIZED:	If end of runs then SLOW BLINK ORANGE.
NOT REFERENCED:	SOLID ORANGE
CONFIGURATION:	No action.
HOMING:	Only check at end of HOMING and then change to NOT INITIALIZED state.
MOVING:	Abort motion and then change to NOT INITIALIZED state.
READY:	Change to NOT INITIALIZED state.
DISABLE:	Change to NOT INITIALIZED state.

LED display:

	RED	ORANGE	GREEN
SOLID	NOT INIT FAULT or ERROR CONFIG	NOT REF /NOT INIT	READY
SLOW BLINK	CONFIG	NOT INIT End-Of-Run Activated	DISABLE
FAST BLINK	-	INIT / HOMING	MOVING or JOGGING

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

When connecting the DL Controller to power, the gets to the NOT INITIALIZED state. From the NOT INITIALIZED state, the Controller can go to the CONFIGURATION state with the PW1 command. In CONFIGURATION stage, the DL Controller allows changing all stage 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 INITIALIZED state.

To execute any move commands (PA, PR), the Controller must be in READY state. To get from the NOT INITIALIZED state to the READY state, the positioner must be initialized first and then homed 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 INITIALIZED state.

In DISABLE state the motor is not energized and the control loop is open. But the encoder is still read and the current position gets updated. 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.

In JOGGING state the Controller allows computer independent motion from the SMC-RC keypad. The Controller can get to the JOGGING state ONLY by pressing the Exec. button on the SMC-RC when the Controller is in the READY or in the DISABLE state. To get from JOGGING state to READY state use the JD command.

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

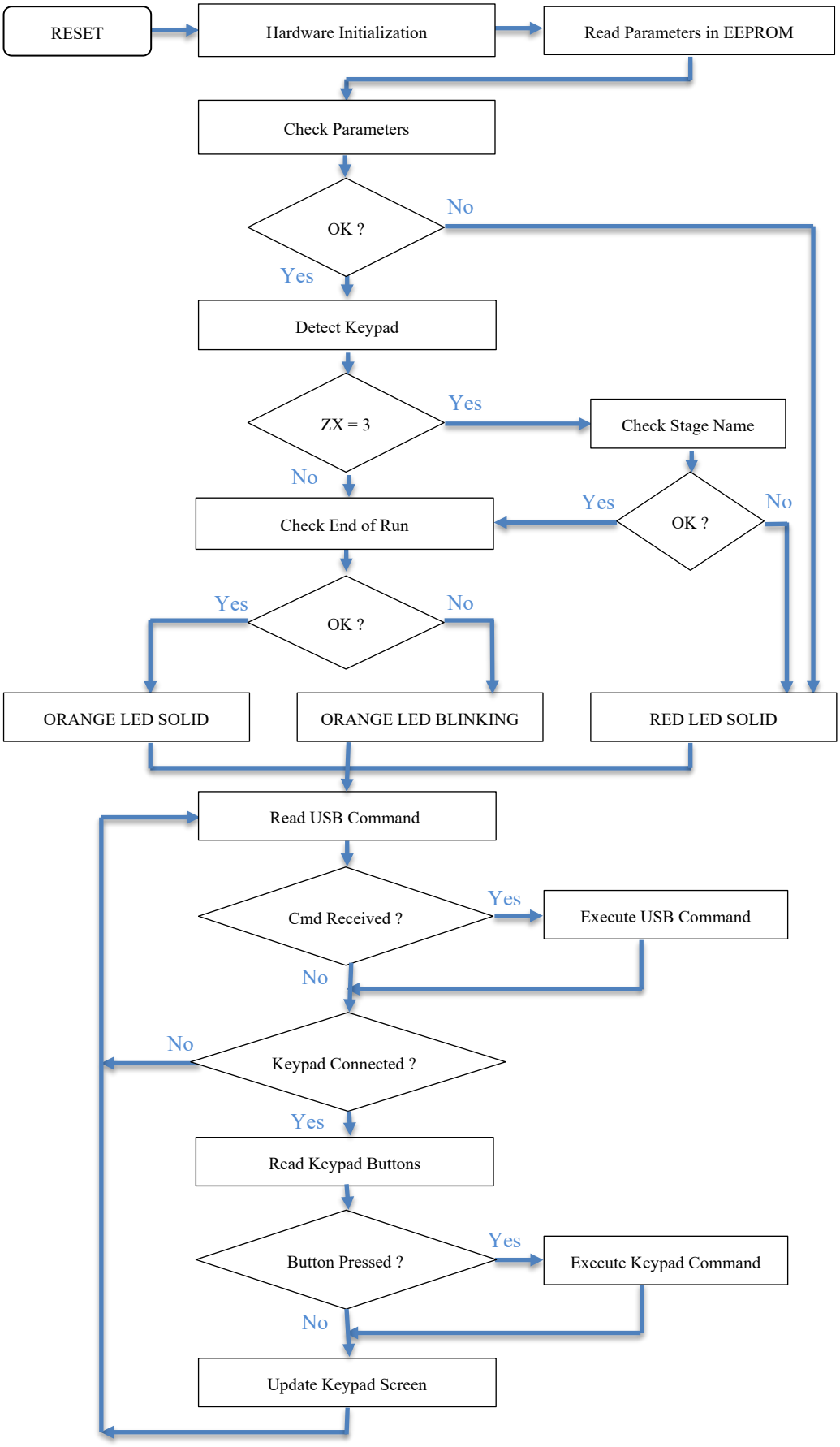
5.2 Initialization

NOTE

The DL Controller cannot initialize a stage when the carriage is at one of the travel limit. It is necessary to manually move the stage away from a limit before initialization

When connecting the DL Controller 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 chapter.

Initialization routine description:



5.3 Command Syntax

The DL Controller 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:

AA or AAA	xx

AA or AAA — 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 "**?**".

Note: xx parameter is converted in "float 32" format

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:

PA1.43 6

PA1.436

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 DL Controller.

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 5.5 for details.

5.4 Command Execution Time

The DL 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 last 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 DL Controller. The general command format is:

Command format:

AA or AAA	xx

AA — Command name.

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

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 DL Controller 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.

Param.	Not Init.	Not Ref.	Config.	Disable	Ready	Motion	Description
AC			○	□	□		Set/Get acceleration
AF			○	□			Set/Get acceleration feed-forward
DB			○	□			Set deadband
DC		●		●	●	●	Internal data capture
DV			○				Set/Get driver voltage
EN			○				Set/Get Encoder settings
EQ			○	□	□		Set/Get AquadB output settings
FD			○	□			Set/Get low pass filter for Kd
FE			○	□			Set/Get following error limit
FL			○	□			Set/Get 2nd order low pass filter for PID
FM			○	□			Set/Get Forces and Masses
FS			●				Factory Settings modification /restoration
GI			○	□	□		Set/Get input triggers settings
GO			○	□	□		Set/Get output triggers settings
GP				●	●		PCO
HO			○				Set/Get HOME offset
HT			○				Set/Get HOME search type
ID			○				Set/Get stage identifier
IE	●						Execute initialization sequence
IT			○				Set Initialization sequence type & setting
JA			○	□	□		Set/Get Jogging Acceleration time
JD						●	Leave JOGGING state
JM			○	□	□		Enable/disable keypad
JR			○	□	□		Set jerk time
JV			○	□	□		Set/Get Jogging velocity
KD			○	□			Set/Get derivative gain
KG			○	□			Set/Get PID variable gains
KI			○	□			Set/Get integral gain
KP			○	□			Set/Get proportional gain
KS			○	□			Set/Get integral saturation
LT			○				Set/Get Limits type
MD			○	□	□		Set/Get Motion Done parameters
MM				●	●		Enter/leave DISABLE state
MP			○				Set/Get Magnet Period
MT			○	□	□		Set/Get Motion Timeout for PD

Param.	Not Init.	Not Ref.	Config.	Disable	Ready	Motion	Description
NF			○	□			Set PID notch filter
OH			○				Set/Get HOME search velocity
OR		●					Execute HOME search
OT			○				Set HOME search time-out
PA					●		Move absolute
PI			○	□			Set/Get PID Integration time
PG			○	□	□		Set/Get Triggered move distance
PD					●		Move relative with response
PR					●		Move relative
PT				●	●	●	Get motion time for a relative move Get acceleration distance for a relative move
PW	●		●				Enter /Leave CONFIGURATION state
QC			○				Set/Get Current Loop Parameters
QI			○				Set/Get motor's current limits
RA	●	●	●	●	●	●	Get analog input value
RF	○	○	○	○	○	○	Set/Get Reference
RS	●	●	●	●	●		Reset Controller
SC			○				Set/Get control loop
SL			○	□	□		Set/Get negative software position limit
SN	●	●	●	●	●	●	Get Controller Serial Number
SR			○	□	□		Set/Get positive software position limit
ST				●	●	●	Stop motion
TB	●	●	●	●	●	●	Get error string
TE	●	●	●	●	●	●	Get last error
TH	●	●	●	●	●	●	Get set point position
TP	●	●	●	●	●	●	Get current position
TS	●	●	●	●	●	●	Get Controller status
VA			○	□	□		Set/Get positioner motion velocity
VE	●	●	●	●	●	●	Get Controller revision information
ZT	●	●	●	●	●	●	Get all axis parameters

Parameter
Controller functionality

Motion: Corresponds to HOMING and MOVING state (for details see state diagram chapter).

○: 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).

NOTE

Use ZT command to get all actual parameters value

AC — Set positioner acceleration

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **AC**nn, or **AC**?

Parameters

Description **nn** [float] — Acceleration value.

Range **nn** — >1E-6 and <1E12.

Units **nn** — Preset units/s².

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the maximum acceleration value which can 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands VA — Set positioner velocity.

Example AC500 | Set acceleration to 500 units/s².

AC? | Controller returns AC500.

AF — Set acceleration feed-forward

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **AF**nn, or **AF**?

Parameters

Description	**nn** [float]	—	Acceleration feed-forward value.
Range	**nn**	—	≥0 and <1E12.
Units	**nn**	—	Volt * second²/preset unit.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

Description In CONFIGURATION state, this command sets acceleration feed-forward of the PID control loop which can 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 acceleration feed-forward. 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

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands

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

Example AF0.015 | Set acceleration feed-forward to 0.015.

DB — Set deadband

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **DBH**mm, or **DBH**?

DBLnn, or **DBL**?

Parameters

Description

mm [float] — Deadband high limit.

nn [float] — Deadband low limit.

Range

mm — ≥0 and <1E12.

nn — ≥0 and <1E12.

Units

mm-nn — Preset unit.

Defaults

mm-nn Missing: Error.

Out of range: Error.

Description

In CONFIGURATION state, this command sets deadband of the PID control loop which can 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 deadband. This value is not saved in the controller's memory and will be lost after reboot.

To disable the deadband functionality set the limits to 0.

Returns

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

KI — Set integral gain.

KP — Set proportional gain.

KD — Set derivative gain.

AS — Set scaling acceleration.

Example

DBL10E-6 | Set deadband low limit to 10E-6 preset unit.

DBH15E-6 | Set deadband high limit to 15E-6 preset unit.

DC — Internal data capture

Usage ☐ Config ☐ NotInit ■ NotRef ■ Disable ■ Jogging
☐ Init ■ Homing ■ Ready ■ Moving

Syntax **DCD**mm, or **DCD**?
DCNnn, or **DCN**?
DCC?
DCVpp, or **DCV**?
DCMqq, or **DCM**?
DCSrr, or **DCS**?
DCAss.
DCT?

Parameters

Description
mm [int] — Frequency divisor value.
nn [int] — Number of data points to be gathered.
pp [int] — Data to be gathered.
qq [int] — Trigger mode.
rr [int] — Gathering enable state.
ss [int] — Number of the data line to get.

Range
mm [int] — 1 to 10000.
nn [int] — 1 to 2048.
pp [int] — 0 to 65535.
qq [int] — 0 or 1.
rr [int] — 0 or 1.
ss [int] — 1 to 2048.

Units **nn-ss** — None.

Defaults
nn-ss Missing: Error.
Out of range: Error.

Description **DCD**: Set se frequency divisor for the gathering.
The gathering frequency is “Controller frequency/divisor”.
DCN: Set the number of data lines to be gathered.
DCC: Get the current number of gathered data lines.
DCV: Set the data to be gathered with a 19-bits value. Each bit enables the gathering of a data when set to 1. The maximum number of data that can be gathered simultaneously is 7.

Bit	Data	Bit	Data
0	CurrentPosition	10	MeasuredCurrentB
1	SetpointPosition	11	SquaredRMSCurrent
2	SetpointVelocity	12	SquaredCurrent
3	SetpointAcceleration	13	EncoderPosition
4	OutputAcceleration	14	ISRtime
5	AdcGpio1	15	ISRratio
6	AdcGpio2	16	MotionDoneCheck & InMotion
7	Cmd_U	17	PositionErrorMeanValue
8	Cmd_V	18	VelocityMeanValue
9	MeasuredCurrentA		

DCM: Set the trigger mode for the gathering.

The command DCM0 configures the gathering to be started immediately when DCS1 command is received.

The command DCM1 configures the gathering to be started at the beginning of the next motion after the DCS1 command is received.

DCS: The DCS1 command enable the gathering, the DCS0 command disable the gathering.

DCA: The command DCAxx returns the gathered data line n°xx.

Returns an error if xx is superior to the current number of gathered lines (DCC?).

DCT: Get all gathered lines.

Controller returns DCT[nb_lines] and then all lines (data from lsb to msb, separated by a ";").

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

Errors

- Unknown message code
- Parameter out of limits
- Execution not allowed in CONFIG state.
- Execution not allowed in NOT INIT state.
- Execution not allowed in INIT state.
- No data selected for gathering.
- Gathering not completed.

Rel. Commands GC — Data capture on trigger.

Example

DCD150	|	Set the frequency divisor to 150.
DCN16	|	Set the number of data lines to be gathered to 16.
DCV33	|	Select CurrentPosition and ADC1 for the gathering (or DCV0x21).
DCM1	|	Configure the gathering to start at next move.
DCS?	|	Controller returns DCS0 (gathering not enabled).
DCS1	|	Enable the gathering.
DCS?	|	Controller returns DCS1 (gathering enabled).
DCC?	|	Controller returns DCC0 (no data gathered).
DCT?	|	Controller returns nothing (gathering not completed).
PR1	|	Relative move to start the gathering.
DCC?	|	Controller returns DCC1 (gathering completed).
DCA6	|	Controller returns DCA0.465249;-0.15213 (position=0.465249mm,adc1=-0.15213V).
DCT?	|	Controller returns all lines: DCT16 Time;CurrentPosition;AdcGpio1 0.0000000;-3.73647e-06;-0.151978 0.0153600;0.000193708;-0.151978 0.0307200;0.0184413;-0.152283 0.0460800; 0.12059;-0.152435 0.0614400; 0.27271;-0.152283 0.0768000;0.465249;-0.15213 0.0921600; 0.6576;-0.15213 0.1075200;0.829643;-0.152435 0.1228800; 0.94803;-0.151978 0.1382400;0.993404;-0.152283 0.1536000; 1.00135;-0.152435 0.1689600; 1.00135;-0.152435

0.1843200; 1.00121;-0.152435
0.1996800; 1.00107;-0.152283
0.2150400; 1.00093;-0.152283
0.23040001; 1.00079;-0.152588.

DCS0 | Disable the gathering.

DV — Set driver voltage

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **DV**nn, or **DV**?

Parameters

Description **nn** [float] — Driver voltage value.

Range **nn** — ≥12 & ≤48.

Units **nn** — Volts.

Defaults **nn** Missing: Error.

Out of range: Error.

Description This command sets the value of the voltage of the connected power supply.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands QI — Set current limit.

Example DV48 | Set Controller having 48 Volts power supply input.

EN — Set encoder settings

Usage ■ Config ☐ NotInit ☐ Init ☐ NotRef ☐ Homing ☐ Disable ☐ Ready ☐ Jogging ☐ Moving

Syntax **ENP**nn, or **ENP**?

ENFmm, or **ENF**?

Parameters

Description **nn** [float] — Encoder pitch.

mm [integer]— Encoder position filter frequency select.

Range **nn** — >1E-6 and <1E12.

mm — 0 : no filter.

1 : 600 Khz (210 ns delay).

2 : 230 Khz (540 ns delay).

3 : 110 Khz (1.14 µs delay).

4 : 49.74 Khz (2.5 µs delay).

5 : 24.74 Khz (5 µs delay).

6 : 12.43 Khz (10 µs delay).

7 : 6.22 Khz (20 µs delay).

8 : 3.11 Khz (40 µs delay).

9 : 1.55 Khz (80 µs delay).

10 : 0.78 Khz (160 µs delay).

Units **nn** — Preset unit.

mm — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Description **ENP**: Set the encoder pitch.

ENF: Set the Encoder position filter frequency.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands PW — Enter/Leave CONFIGURATION state.

Example ENP0.08 | Set encoder pitch to 0.08 units.

ENF8 | Set encoder position filter to 3.11 kHz.

EQ — Set AquadB output settings

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

SYNTAX **EQP**nn, or **EQP**?
EQFmm, or **EQF**?
EQRpp, or **EQR**?

Parameters

Description **nn** [integer]— AquadB output polarity.

mm [integer]— Position filter frequency select for AquadB output.

pp [integer]— Interpolation factor for AquadB output.

Range **nn** — 0 or 1.

mm — 0 : no filter.
1 : 600 Khz (210 ns delay).
2 : 230 Khz (540 ns delay).
3 : 110 Khz (1.14 µs delay).
4 : 49.74 Khz (2.5 µs delay).
5 : 24.74 Khz (5 µs delay).
6 : 12.43 Khz (10 µs delay).
7 : 6.22 Khz (20 µs delay).
8 : 3.11 Khz (40 µs delay).
9 : 1.55 Khz (80 µs delay).
10 : 0.78 Khz (160 µs delay).

pp — 2 : AquadB resolution = Encoder Pitch/4.
3 : AquadB resolution = Encoder Pitch/8.
4 : AquadB resolution = Encoder Pitch/16.
5 : AquadB resolution = Encoder Pitch/32.
6 : AquadB resolution = Encoder Pitch/64.
7 : AquadB resolution = Encoder Pitch/128.
8 : AquadB resolution = Encoder Pitch/256.
9 : AquadB resolution = Encoder Pitch/512.
10: AquadB resolution = Encoder Pitch/1024.
11: AquadB resolution = Encoder Pitch/2048.

Units **nn**-pp — None.

Defaults **nn**-ppMissing: Error.
Out of range: Error.

Description **EQP**: Set the AquadB output polarity.
EQF: Set the position filter frequency for AquadB output.
EQR: Set the position interpolation for AquadB output.

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

Errors		—	Unknown message code.
		—	Parameter missing or out of range.
		—	Execution not allowed.
		—	Execution not allowed in NOT INIT state.
		—	Execution not allowed in INIT state.
		—	Execution not allowed in NOT REFERENCED state.
		—	Execution not allowed in HOMING state.
		—	Execution not allowed in MOVING state.
Rel. Commands	PW	—	Enter/Leave CONFIGURATION state.
Example	EQP1	|	Set AquadB output polarity to 1.
	EQF8	|	Set position filter to 3.11 kHz for AquadB output.
	EQR32	|	Set the interpolation factor to 32 for AquadB output.

FD — Set low pass filter cut off frequency for Kd

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **FD**nn, or **FD**?

Parameters

Description **nn** [float] — Cut off frequency value.

Range **nn** — >1E-6 and <1/(2 * Tb).

Units **nn** — Hertz.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the value for the low pass filter cut-off frequency which can 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

PW — Enter/Leave CONFIGURATION state.

Example FD1500 | Set Kd cut off frequency to 1500 Hz.

FE — set following error limit

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **FE**nn, or **FE**?

Parameters

Description **nn** [float] — Following error limit value.

Range **nn** — >1E-6 and <1E12.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the value for the maximum allowed following error which can 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands SC — Set closed loop state.

PW — Enter/Leave CONFIGURATION state.

Example FE0.015 | Set following error limit to 0.015 units.

FL — Set low pass filter cut off frequency for PID

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **FL**nn, or **FL**?

Parameters

Description **nn** [float] — Cut off frequency value.

Range **nn** — >1E-6 and <1/(2 * Tb).

Units **nn** — Hertz.

Defaults **nn** Missing: Error.
Out of range: Error.

Description In CONFIGURATION state, this command sets the value for the 2nd order low pass filter cut-off frequency for PID output which can 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
— Unknown message.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands
SC — Set closed loop state.
PW — Enter/Leave CONFIGURATION state.

Example FL1500 | Set PID cut off frequency to 1500 Hz.

FM — Set forces and masses

Usage

■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **FMS**nn, or **FMS**?

FMLmm, or **FML**?

FMCpp, or **FMC**?

FMPqq, or **FMP**?

Parameters

Description

nn [float] — Scaling Force.

mm [float] — Force Limit.

pp [float] — Carriage Mass.

qq [float] — Payload Mass.

Range **nn**-qq — 0 to 1E12.

Units **nn**-mm — Newton.

pp-qq — Kg.

Defaults **nn**-qqMissing: Error.

Out of range: Error.

Description **FMS**: Set the Scaling Force.

FML: Set the Force Limit.

FMC: Set the Carriage Mass.

FMP: Set the Payload Mass.

ScalingAcceleration = ScalingForce/(CarriageMass + Payload).

AccelerationLimit = ForceLimit/(CarriageMass + Payload).

Scaling Parameters validity conditions:.

ForceLimit <= ScalingForce.

MaximumAcceleration <= ForceLimit/(CarriageMass + Payload).

CarriageMass + Payload >0.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Error in Scaling Parameters

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

AC — Set/get Acceleration.

PW — Enter/Leave CONFIGURATION state.

Example	FMS20000	|	Set Scaling Force to 20000 N.
	FML10000	|	Set Force Limit to 10000 N.
	FMC2.5	|	Set Carriage Mass to 2.5 kg.
	FMP1	|	Set Payload Mass to 1 kg.

FS — Factory settings modification/restoration

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	☐ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **FSM**nn, or **FSM**?

FSR.

Parameters

Description **nn** [string] — Password for factory settings modification.

Range **nn** — Unique password.

Units **nn** — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Description **FSM**: This command sends the password to allow factory settings or serial number modifications. If the password is correct, all parameters modification in CONFIGURATION state changes the factory settings of the Controller. The factory settings modifications rights are lost after leaving the CONFIGURATION state.

FSR: This command restores all parameters to factory settings.

Returns If the sign ″?″ takes place of nn, this command returns 1 if the Controller have the factory settings modifications rights, else it returns 0.

Errors

— Unknown message.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in DISABLE state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands PW — Enter/Leave CONFIGURATION state.

SN — Set/Get Serial Number.

Example FSR | Restore all parameters to factory settings.

GI — Set inputs triggers settings

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **GIM**nn, or **GIM**?

GICmm, or **GIC**? (not implemented).

GITpp, or **GIT**? (not implemented).

Parameters

Description

nn [integer]— Polarity for start motion trigger.

mm [integer]— Polarity for data capture trigger.

pp [integer]— Type of data capture trigger.

Range

nn	—	0 or 1.
mm	—	0 or 1.
pp	—	0 to 2.

Units

nn-pp	—	None.

Defaults

nn-ppMissing:	Error.
Out of range:	Error.

Description **GIM**: Set the polarity of input trigger 1(start motion trigger).
(0:falling edge, 1:rising edge).
GIC: Set the polarity of input trigger 2 (data capture).
(0:falling edge, 1:rising edge).
GIT: Set the type of input trigger 2.
(0: data capture/1: PGR direction/2: goto reference).

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

Errors

	—	Unknown message.
	—	Parameter missing or out of range.
	—	Execution not allowed.
	—	Execution not allowed in NOT INIT state.
	—	Execution not allowed in INIT state.
	—	Execution not allowed in NOT REFERENCED state.
	—	Execution not allowed in HOMING state.
	—	Execution not allowed in MOVING state.

Rel. Commands

PW	—	Enter/Leave CONFIGURATION state.
GC	—	Data capture on trigger.
PG	—	Set triggered move distance.

Example

GIM1	|	Set start motion trigger polarity to 1 (rising edge).
GIC0	|	Set data capture trigger polarity to 0 (falling edge).

GO — Set output triggers settings

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **GOT**nn, or **GOT**?
GOMmm, or **GOM**?
GOPpp, or **GOP**?
GOFqq, or **GOF**?
GOWrr, or **GOW**?

Parameters

Description
nn [integer]— Type of motion output trigger.
mm [integer]— Polarity of motion output trigger.
pp [integer]— Polarity of PCO output trigger.
qq [integer]— Position filter frequency of PCO output trigger.
rr [float] — Pulse width of PCO output trigger.

Range
nn — 0 : In Motion.
1 : Constant velocity.
mm — 0 or 1.
pp — 0 or 1.
qq — 0 : no filter.

1 : 600 Khz	(210 ns delay).
2 : 230 Khz	(540 ns delay).
3 : 110 Khz	(1.14 µs delay).
4 : 49.74 Khz	(2.5 µs delay).
5 : 24.74 Khz	(5 µs delay).
6 : 12.43 Khz	(10 µs delay).
7 : 6.22 Khz	(20 µs delay).
8 : 3.11 Khz	(40 µs delay).
9 : 1.55 Khz	(80 µs delay).
10 : 0.78 Khz	(160 µs delay).

rr — 15E-9 to 327.68E-6 s.

Units
nn-qq — None.
rr — Sec.

Defaults
nn-rrMissing: Error.
Out of range: Error.

Description
GOT: Set the type of output trigger 1 (motion trigger).
GOM: Set the polarity of output trigger 1 (motion trigger).
GOP: Set the polarity of output trigger 2 (PCO).
GOF: Set the position filter frequency for the PCO output.
GOW: Set the pulse width for PCO output trigger.
(rounded to nearest 5ns multiple).

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

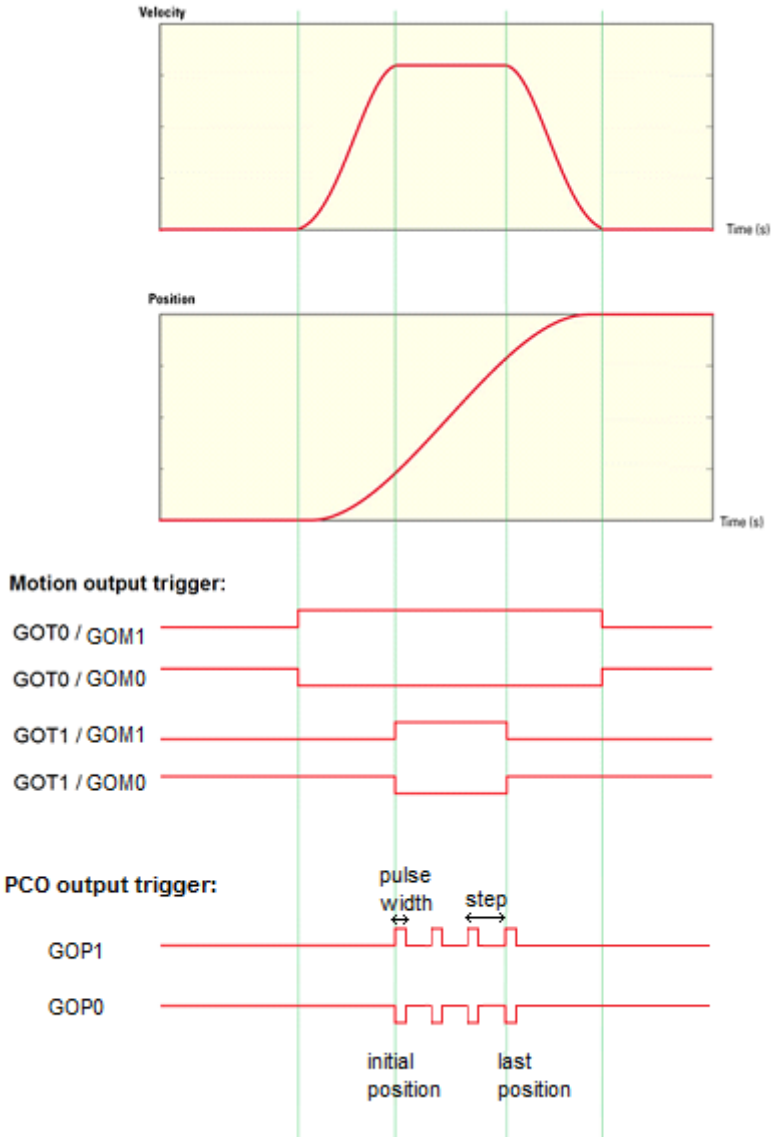
Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

.

Rel. Commands
- PW — Enter/Leave CONFIGURATION state.
- GP — PCO.

Example

GOT1	Set motion output trigger type to Constant Velocity.
GOM0	Set motion output trigger polarity to 0.
GOF3	Set PCO position filter to 110 kHz.
GOP0	Set PCO polarity to negative pulses.
GOW200e-9	Set PCO pulses duration to 200 ns.



GP — PCO (Position compare output trigger)

Usage	☐ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ■ Ready	☐ Jogging ☐ Moving

Syntax **GPI**nn, or **GPI**?
GPLmm, or **GPL**?
GPSpp, or **GPS**?
GPEqq, or **GPE**?

Parameters

Description
nn [float] — Initial position for PCO trigger.
mm [float] — Last position for PCO trigger.
pp [float] — Step distance for PCO trigger.
qq [integer]— Enable of PCO trigger.

Range
nn — SL to SR.
mm — SL to SR.
pp — MIM to ((SR-SL)/2).
qq — 0 or 1.

Units
nn — Preset unit.
mm — Preset unit.
pp — Preset unit.
qq — None.

Defaults
nn-qqMissing: Error.
Out of range: Error.

Description
GPI: Set the initial position.
GPL: Set the last position.
GPS: Set the distance between each pulse.
GPE: Set the enable of the PCO function.

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

Errors
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands GO — Output triggers settings.

Example
GPI10 | Set first position to 10 mm.
GPL20 | Set last position to 20 mm.
GPS10e-3 | Set step to 10 µm.
GPE1 | Enable PCO function.

NOTE

To restart triggering, make sure to move outside the defined window, disable and re-enable pulse triggering using GPE0 and GPE1 commands respectively.

HO — Set HOME search offset

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	☐ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax HOnn, or HO?

Parameters

Description **nn** [float] — Home offset.

Range **nn** — -1E12 to 1E12.

Defaults **nn** Missing: Error.

Out of range: Error.

Description This command sets the HOME offset that will be used when OR command will be executed.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

REL. COMMANDS OR — Search for HOME.

Example HO5 | Set Home offset to 5 units.

HT — Set HOME search type

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax HTnn, or HT?

Parameters

Description **nn** [int] — Home type value.

Range **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 **nn** — None.

Defaults **nn** Missing: Error.
Out of range: Error.

Description This command sets the HOME type that will be used when OR command will be executed.

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

Errors
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands OR — Search for HOME.

Example HT0 | Use MZ and encoder index for HOME.

ID — Set/Get stage identifier

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax IDnn, or ID?

Parameters

Description **nn** [string] — Stage identifier.

Range **nn** — 1 to 31 ASCII characters.

Defaults **nn** Missing: Error.

Out of range: Error.

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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands ZX — Set SmartStage configuration.

Example ID? | Get stage identifier.

| Controller returns DL Controller225.

IE — Execute initialization

Usage

☐ Config	■ NotInit	☐ NotRef	☐ Disable	☐ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax IE.

Parameters None.

Description This command starts the execution of the Initialization sequence as defined by the IT command.

Errors

— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in CONFIG state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands IT — Initialization settings.

Example IE| Execute initialization sequence.

IT — Set initialization sequence settings

Usage

■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **ITA**nn, or **ITA**?

ITDmm, or **ITD**?

Parameters

Description

nn [float] — Initialization acceleration level.

mm [integer]— Initialization cycle duration.

Range

nn — 0 to 100.

mm — >Tb and <1E12.

Units

nn — %.

mm — Sec.

Defaults

nn Missing: Error.

Out of range: Error.

Description **ITA**: Set InitializationAccelerationLevel.

(InitializationAccelerationLevel * ScalingAcceleration/100) must be smaller than MaxAcceleration.

ITD: Set InitializationCycleDuration.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

PW — Enter/Leave CONFIGURATION state.

IE — Execute initialization sequence.

Example

ITA15 | Set Initialization Acceleration level to 15%.

ITD2 | Set Initialization Cycle Duration to 2 s.

JA — Set jogging acceleration time

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

SYNTAX **JA**nn, or **JA**?

Parameters

Description **nn** [float] — Acceleration time value.

Range **nn** — >Tb and <1E12.

Units **nn** — Seconds.

Defaults **nn** Missing: Error.

Out of range: Error.

Description The jogging accelerarion time defines the time to reach the needed velocity in jogging mode (with a remote keypad).

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

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands JV — Set jogging velocity.

Example JA0.1 | Set acceleration time to 100 ms.

JD — Leave jogging state

Usage

☐ Config	☐ NotInit	☐ NotRef	☐ Disable	■ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax JD.

Parameters None.

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

Errors

— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in CONFIG state.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands JM — Enable/Disable keypad.

Example JD| Controller leave Jogging State.

JM — Enable/Disable keypad

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **JM**nn, **JM**?

Parameters

Description **nn** [integer]— Jog State.

Range **nn** — 0 or 1.

Units **nn** — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Description The JM1 command enables the SMC-RC keypad buttons (default setting). The JM0 command disables the SMC-RC 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).

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands JD — Leave JOGGING state.

ExampleJM1 | Enable keypad.

JR — Set jerk time

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **JR**nn, or **JR**?

Parameters

Description	**nn** [float]	—	Jerk time value.
Range	**nn**	—	>1E-6 and <1E12.
Units	**nn**	—	Seconds.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

Description 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.

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands	AC	—	Set positioner acceleration.
Example	JR0.05	|	Set jerk time to 0.05 second.

JV — Set jogging velocity

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **JV**nn, or **JV**?

Parameters

Description	**nn** [float]	—	Acceleration time value.
Range	**nn**	—	>1E-6 and ≤ maximum velocity.
Units	**nn**	—	Unit/s.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

Description The jogging velocity is the slow motion velocity when using a remote keypad.

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands	JA	—	Set jogging acceleration time.
Example	JV10	|	Set jogging velocity to 10 units/s.

KD — Set derivative gain

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KD**nn, or **KD**?

Parameters

Description **nn** [float] — Derivative gain value.

Range **nn** — ≥0 and <1E12.

Units **nn** — Volt * second/preset unit.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the derivative gain of the PID control loop which can 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

KI — Set integral gain.

KP — Set proportional gain.

AF — Set acceleration feed forward.

AS — Set scaling acceleration.

Example KD0.015 | Set derivative gain to 0.015.

KG — set variable PID gains

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KGP**nn, or **KGP**?
KGImm, or **KGI**?
KGDpp, or **KGD**?
KGFqq, or **KGF**?

Parameters

Description
nn [float] — Kp Gain.
mm [float] — Ki Gain.
pp [float] — Kd Gain.
qq [float] — KForm Gain.

Range **nn**-qq — 0 to 1.

Units **nn**-qq — None.

Defaults **nn**-qqMissing: Error.
Out of range: Error.

Description **KGP**: Set the Kp PID gain.
KGI: Set the Ki PID gain.
KGD: Set the Kd PID gain.
KGF: Set the KForm PID gain.

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

Errors
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands PW — Enter/Leave CONFIGURATION state.

Example
KGP0.8 | Set Kp gain to 0.8.
KGI0.9 | Set Ki gain to 0.9.
KGD0.7 | Set Kd gain to 0.7.
KGF1 | Set KForm gain to 1.

KI — Set integral gain

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KI**nn, or **KI**?

Parameters

Description	**nn** [float]	—	Integral gain value.
Range	**nn**	—	≥0 and <1E12.
Units	**nn**	—	Volt * preset unit/second.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

Description In CONFIGURATION state, this command sets the integral gain of the PID control loop which can 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

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands

SC	—	Set closed loop state.
KD	—	Set derivative gain.
KP	—	Set proportional gain.
AF	—	Set acceleration feed forward.
AS	—	Set scaling acceleration.

Example KI0.015 | Set integral gain to 0.015.

KP — Set proportional gain

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KP**nn, or **KP**?

Parameters

Description **nn** [float] — Proportional gain value.

Range **nn** — ≥0 and <1E12.

Units **nn** — Volt/preset unit.

Defaults **nn** Missing: Error.

Out of range: Error.

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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

KD — Set derivative gain.

KI — Set integral gain.

AF — Set acceleration feed forward.

AS — Set scaling acceleration.

Example KP0.015 | Set proportional gain to 0.015.

KS — Set integral saturation

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KS**nn, or **KS**?

Parameters

Description **nn** [float] — Integral saturation value.

Range **nn** — 0 to 1.

Units **nn** — Preset unit/second².

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the integral saturation level 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 integral sdaturation. 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

KD — Set derivative gain.

KI — Set integral gain.

AF — Set acceleration feed forward.

KP — Set proportional gain.

Example KS0.8 | Set the integral saturation to 0.8.

LT — Set limits type

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging	
		☐ Init	☐ Homing	☐ Ready	☐ Moving	

Syntax LTnn, or LT?

Parameters

Description **nn** [int] — Home type value.

Range **nn** — 0 Limit+ and Limit-.
1 Limit and Mechanical zero.

Defaults **nn** Missing: Error.
Out of range: Error.

Description This command sets the Limits type of the encoder plug.

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

Errors
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands OR — Search for HOME.

Example LT0 | Use Limit+ and Limit- limits type.

MD — Set Motion done parameters

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **MDM**nn, or **MDM**?

MDPmm, or **MDP**?

MDVpp, or **MDV**?

MDCqq, or **MDC**?

MDArr, or **MDA**?

MDTss, or **MDT**?

Parameters

Description

nn [integer]— Motion Done Mode.

mm [float] — Position Threshold.

pp [float] — Velocity Threshold.

qq [float] — Checking Time.

rr [float] — Mean Period.

ss [float] — Timeout.

Range

nn — 0 : Theoretical.
1 : Velocity and Position Window.

mm — 0 to 10^{12}.

pp — 0 to 10^{12}.

qq — 0 to 10^{3}.

rr — ISR period to 10^{3}.

ss — ISR period to 10^{3}.

Units

nn — None.

mm — Preset Unit.

pp — Preset Unit/second.

qq — Second.

rr — Second.

ss — Second.

Defaults

nn-ssMissing: Error.

Out of range: Error.

Description

MDM: Set MotionDone to Theoretical or VelocityandPositionWindow. In theory, MotionDone is completed as defined by the profiler with an additional configurable delay.

The VelocityAndPositionWindow MotionDone allows a more precise definition by specifying the end of the move with a number of parameters that take the settling of the positioner into account. In the VelocityAndPositionWindow MotionDone, the motion is completed when: | ErrorMeanValue | < | PositionThreshold |.

AND| VelocityMeanValue | < | VelocityThreshold |.

Is verified during the CheckingTime period.

MDP: Set the Position Threshold.

The position error has to be within ± of this value for a period of CheckingTime to validate the MotionDone.

MDV: Set the Velocity Threshold.

The velocity at the end of the motion has to be within ± of this value for a period of CheckingTime to validate the MotionDone.

MDC: Set the Checking Time. This parameter defines the period during which the conditions for the PositionThreshold and the VelocityThreshold must be true before setting the motion done.

In Theoretical mode this parameter define the delay after the end of the profiler to activate the motion done.

MDA: Set the Mean period. A sliding mean filter is used to attenuate the noise for the position and velocity parameters. The MeanPeriod defines the duration for calculating the sliding mean position and velocity.

MDT: Set the timeout which defines the maximum time the Controller will wait from the end of the theoretical move for the MotionDone condition, before sending a MotionDone time-out.

NOTE

MDC value must be lower than MDT otherwise, the Motion Done Time out error (40000) will appear systematically.

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

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands
PW — Enter/Leave CONFIGURATION state.

Example

MDM1	|	Set MotionDone to VelocityAndPositionWindow.
MDP4	|	Set PositionThreshold to 4 units.
MDV100	|	Set VelocityThreshold to 100 units.
MDC0.1	|	Set CheckingTime to 100 ms.
MDA0.001	|	Set MeanPeriod to 1 ms.
MDT0.5	|	Set Timeout to 500 ms.

MM — Enter/Leave DISABLE state

Usage	☐ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **MM**nn, or **MM**?

Parameters

Description **nn** [float] — Velocity feed forward value.

Range **nn** — 0 change from READY to DISABLE state.

1 change from DISABLE to READY state.

Units **nn** — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Description 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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in CONFIG state.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands PW — Enter/leave CONFIGURATION state.

Example MM0 | Controllers goes to DISABLE state.

MP — Set magnet period

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	☐ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **MP**nn, or **MP**?

Parameters

Description **nn** [float] — Magnet period.

Range **nn** — >0 and <1E12.

Units **nn** — Preset Unit.

Defaults **nn** Missing: Error.
Out of range: Error.

Description Set the Magnet Period value.

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

Errors
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands PW — Enter/leave CONFIGURATION state.

Example MP42 | Set the Magnet Period to 42 Preset Unit.

MT — Set motion timeout for PD

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **MT**nn, or **MT**?

Parameters

Description **nn** [float] — Timeout value.

Range **nn** — >Tb and <1E3.

Units **nn** — Seconds.

Defaults **nn** Missing: Error.

Out of range: Error.

Description Set the timeout value of the PD commands. If a PD motion is not finished after this time the Controller will send a timeout response.

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

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands PD — Move relative with response.

Example MT2 | Set PD timeout to 2 seconds.

NF — Set PID notch filter

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **NFF**nn, or **NFF**?

NFGpp, or **NFG**?

NFWmm, or **NFW**?

Parameters

Description

mm [float] — Notch filter frequency width.

nn [float] — Notch filter center frequency.

pp [float] — Notch filter gain.

Range

mm — ≥0 and <1E12.

nn — ≥0 and <1E12.

pp — ≥0 and ≤1.

Units

mm-nn — Hertz.

pp — None.

Defaults

mm-pp Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets notch filter parameters (frequency, width, gain) of the PID control loop which can 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 notch filter. This value is not saved in the controller's memory and will be lost after reboot.

To disable the notch functionality set the gain to 1.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

SC — Set closed loop state.

KI — Set integral gain.

KP — Set proportional gain.

KD — Set derivative gain.

Example

NFF400 | Set notch filter center frequency to 400 Hz.

NFW100 | Set notch filter frequency width to 100 Hz.

NFG0.1 | Set notch filter gain to 0.1.

OH — Set HOME search velocity

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **OH**nn, or **OH**?

Parameters

Description	**nn** [float]	—	HOME high velocity.
Range	**nn**	—	>1E-6 and ≤ maximum velocity.
Units	**nn**	—	Preset units/s.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

Description This command sets the maximum velocity that will be used by the motion profiler to move the positioner during the HOME search sequence.

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in DISABLE state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands

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

Example

OH50	|	Set HOME search velocity to 50 units/s.

OR — Execute HOME search

Usage	☐ Config	☐ NotInit	■ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **OR**.

Parameters None.

Description This command starts the execution of the HOME search sequence as defined by the HT command.

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in CONFIG state.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in DISABLE state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	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

OR	|	Execute HOME search sequence.

OT — Set HOME sequence time-out

Usage	■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
		☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **OT**nn, or **OT**?

Parameters

Description **nn** [float] — HOME search time-out.

Range **nn** — >1 and <1E^3.

Units **nn** — Seconds.

Defaults **nn** Missing: Error.

Out of range: Error.

Description This command sets the maximum time allowed to the HOME search sequence to find HOME. If the sequence do not finish successfully before this time elapses, it will be aborted and an error is recorded.

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

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in DISABLE state.
- — Execution not allowed in READY state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands
- HT — Set HOME search type.
- OH — Set HOME search velocity.
- OR — Execute HOME search.

Example OT2.2 | Set HOME search time-out to 2.2 seconds.

PA — Move absolute

Usage	☐ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	☐ Disable ■ Ready	☐ Jogging ☐ Moving

Syntax **PA**nn, or **PA**?

Parameters

Description	**nn** [float]	—	New target position.
Range	**nn**	—	>SL and <SR.
Units	**nn**	—	Preset units.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

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

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Target position out of limits.
—	Execution not allowed in CONFIG state.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in DISABLE state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands

PR	—	Move relative.
PD	—	Move relative with Controller response.
PG	—	Set triggered move distance.
TH	—	Tell set point position.
TP	—	Tell current position.

Example PA2.2 | Move to 2.2 units absolute position.

PI — Set PID integration time

Usage	■ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ☐ Ready	☐ Jogging ☐ Moving

Syntax **KI**nn, or **KI**?

Parameters

Description **nn** [float] — PID integration time.

Range **nn** — ≥0 and <1E12.

Units **nn** — Volt * second/preset unit.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the derivative gain of the PID control loop which can 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
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in READY state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands
- SC — Set closed loop state.
- KI — Set integral gain.
- KP — Set proportional gain.
- AF — Set acceleration feed forward.
- AS — Set scaling acceleration.

Example KD0.015 | Set derivative gain to 0.015.

PG — Set triggered move distance

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **PG**nn, or **PG**?

Parameters

Description **nn** [float] — Displacement.

Range **nn** — >(SL-SR) and <(SR-SL).

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description The PG command program a motion that will be executed each time a start motion trigger occurs in READY state.

Returns If the sign "?" takes place of nn, this command returns the programmed triggered move distance.

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands

GI	—	Set input triggers settings.
PA	—	Move absolute.
PD	—	Move relative with Controller response.
PR	—	Move relative.
TH	—	Tell set point position.
TP	—	Tell current position.

Example

PG2.2	|	Program a triggered motion of 2.2 units.
PG0	|	Disable triggered motion.

PD — Move relative with controller response

Usage

☐ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
	☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **PD**nn.

Parameters

Description **nn** [float] — Displacement.

Range **nn** — >SL and <SR.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description The PD 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 PD 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.

All others commands received by the Controller during PD command execution will be executed after the PD command response.

Returns When an error occurs the Controllers respond with PD0. Send TE command to get the error code.

When the Motion Done condition occurs the Controllers respond with PD1.

Errors

— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Target position out of limits.
— Displacement time (PTT) >timeout.
— Motion timeout.
— Motion error (Send TS command to see error).
— Execution not allowed in CONFIG state.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands

MT — Set Motion Time-out.
TE — Get last error.
PTT — Get Motion time for a relative move.

Example

PD2.2 | Move 2.2 units from target position.
PD1 | Controller returns motion done.

PR — Move relative

Usage	☐ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	☐ Disable ■ Ready	☐ Jogging ☐ Moving

Syntax **PR**nn, or **PR**?

Parameters

Description **nn** [float] — Displacement.

Range **nn** — >SL and <SR.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Target position out of limits.

— Execution not allowed in CONFIG state.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands

PA — Move absolute.

PD — Move relative with Controller response.

PG — Set triggered move distance.

TH — Tell set point position.

TP — Tell current position.

Example PR2.2 | Move 2.2 units from target position.

PT — Tell motion time or acceleration distance

Usage	☐ Config	☐ NotInit ☐ Init	☐ NotRef ☐ Homing	■ Disable ■ Ready	☐ Jogging ■ Moving

Syntax **PTT**nn, **PTA**.

Parameters

Description **nn** [float] — Displacement.

Range **nn** — >1E-6 and <1E12.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description The PT commands helps evaluating move times and acceleration distance for an efficient program flow.

PTT: When receiving the PTT 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, motion done delay, etc.). The Controller does not execute any motion.

PTA: When receiving the PTA command, the Controller returns the distance necessary to reach a constant velocity with the current working parameters (velocity, acceleration, etc.). The Controller does not execute any motion.

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in CONFIG state.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

Rel. Commands

PA — Move absolute.

PG — Triggered move.

PD — Move relative with Controller response.

PR — Move relative.

TH — Tell set point position.

TP — Tell current position.

Example

PTT2.2 | Tell move time to move for 2.2 units.

PTT0.25 | Controller returns value in seconds.

PTA | Tell acceleration distance.

PTA0.1 | Controller returns value in Preset Units.

PW — Enter/Leave configuration state

Usage

■ Config	■ NotInit	☐ NotRef	☐ Disable	☐ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax **PW**nn, or **PW**?

Parameters

Description **nn** [float] — Velocity feed forward value.

Range **nn** — 1 change from NOT INIT to CONFIGURATION.

0 change from CONFIGURATION to NOT INIT.

Units **nn** — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Description PW1 changes the Controller's state from NOT INIT 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, 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 INIT.

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

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in DISABLE state.

— Execution not allowed in READY state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands MM — Enter/Leave Disable state.

Example PW1 | Enters CONFIGURATION state.

QC — Set current loop parameters

Usage ■ Config ☐ NotInit ☐ Init ☐ NotRef ☐ Homing ☐ Disable ☐ Ready ☐ Jogging ☐ Moving

Syntax **QCR**nn, **QCR**?

QCLmm, **QCL**?

QCFpp, **QCF**?

Parameters

Description

nn [float]	—	Motor's resistance
mm [float]	—	Motor's inductance
pp [float]	—	Current loop CutOff Frequency.

Range

nn	—	>0 and <1E12.
mm	—	>0 and <1E12.
pp	—	>0 and <1E12.

Units

nn	—	Ohm
mm	—	Henry
pp	—	Hertz.

Defaults

nn-ppMissing:	Error.
Out of range:	Error.

Description **QCR**: Sets the motor's resistance.

QCL: Sets the motor's Inductance.

QCF: Sets the current loop Cutoff frequency.

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in DISABLE state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Example

QIR2.6	|	Set motor's resistance to 2.6 Ohm.
QIL2e-3	|	Set motor's inductance to 3mH.
QIF400	|	Set current loop CutOff Frequency to 400 Hz.

QI — Set motor's current limits

Usage

■ Config	☐ NotInit	☐ NotRef	☐ Disable	☐ Jogging
	☐ Init	☐ Homing	☐ Ready	☐ Moving

Syntax

QILnn, **QIL**?

QIRmm, **QIR**?

QITpp, **QIT**?

Parameters

Description

nn [float]	—	Motor's current limit
mm [float]	—	Motor's rms current
pp [float]	—	Motor's rms current average time.

Range

nn	—	>0.05 and <= 6 and nn >mm.
mm	—	>0.05 and <= 6 and mm <nn.
pp	—	>0.01 and <= 100.

Units

nn	—	Amperes
mm	—	Amperes
pp	—	Seconds.

Defaults

nn-ppMissing:	Error.
Out of range:	Error.

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 current limit.

Returns

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in DISABLE state.
—	Execution not allowed in READY state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands

DV	—	Set driver input voltage.

Example

QIL0.75	|	Set motor's current limit to 0.75 amps.
QIR0.25	|	Set motor's rms current to 0.25 amps.
QIT2.5	|	Set rms averaging period to 2.5 seconds.

RA — Tell analog input value

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **RAnn**

Parameters

Description **nn** [char] — ADC channel.

Range **nn** — A or B.

Units **nn** — None.

Defaults **nn** Missing: Error.

Out of range: Error.

Floating point: Error.

Description This command returns the analog input value in volts. This is the +/- 5 volts input located on the JXX connector. The converter is a +/- 15 bits analog to digital converter with XX volts of offset and XX % full scale linearity. The resolution is 152 µV.

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

Example

RAA	|	Tell ADC1 analog input.
RAA0.025	|	Controller returns 0.025 volts.
RAB	|	Tell ADC2 analog input.
RAB-0.5	|	Controller returns -0.5 volts.

RF — Set/Get reference

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **RF**nn, **RF**?

Parameters

Description **nn** [float] — Reference position.

Range **nn** — SL to SR.

Units **nn** — Preset Unit.

Defaults

nn Missing: Error.

Out of range: Error.

Floating point: Error.

Description Set the reference position.

The reference is saved in the Controller's memory and remains available after switching off the Controller.

Errors

— Unknown message code.

— Parameter missing or out of range.

— Position out of limits.

— Execution not allowed.

Example RF12 | Set the reference position to 12 units.

RS — Reset controller

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **RS**.

Parameters None.

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 INIT to CONFIGURATION.

Errors

— Unknown message code.

— Execution not allowed.

Example

RS | Reset Controller.

SC — Set control loop

Usage ■ Config ☐ NotInit ☐ NotRef ☐ Disable ☐ Jogging
☐ Init ☐ Homing ☐ Ready ☐ Moving

Syntax **SC**nn, or **SC**?

Parameters

Description **nn** [int] — Closed loop state.

Range **nn** — 1 change to CLOSED.
0 change to OPEN.

Units **nn** — None.

Defaults **nn** Missing: Error.
Out of range: Error.

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
— Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.
— Execution not allowed in NOT INIT state.
— Execution not allowed in INIT state.
— Execution not allowed in NOT REFERENCED state.
— Execution not allowed in DISABLE state.
— Execution not allowed in READY state.
— Execution not allowed in HOMING state.
— Execution not allowed in MOVING state.

Rel. Commands
KD — Set derivative gain
KI — Set integral gain
KP — Set proportional gain
AF — Set acceleration feed forward.

Example SC1 | Set Controller to closed loop control.

SL — Set negative software position limit

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **SL**nn, or **SL**?

Parameters

Description **nn** [float] — Negative software limit.

Range **nn** — >-1E12 and <= 0.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the negative software limit which can 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.

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

Errors
- — Unknown message code.
- — Parameter missing or out of range.
- — Execution not allowed.
- — Execution not allowed in NOT INIT state.
- — Execution not allowed in INIT state.
- — Execution not allowed in NOT REFERENCED state.
- — Execution not allowed in HOMING state.
- — Execution not allowed in MOVING state.

Rel. Commands SR — Set positive software position limit.

Example SL-100 | Set software negative limit to -100 units.

SN — Set/Get serial number

Usage	■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
		■ Init	■ Homing	■ Ready	■ Moving

Syntax **SN**nn, or **SN**?

Parameters

Description **nn** [integer]— Serial number.

Range **nn** — ≥0 and >1E12 and.

Units **nn** — Preset units.

Defaults **nn** Missing: Error.

Out of range: Error.

Description This command returns the serial number of the Controller.

The modification of the serial number is allowed only in CONFIGURATION state after sending the FSM command with a correct password.

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

Errors — Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

Rel. Commands FS — Factory settings Restoration/Modification.

Example SN? | Controller returns SNxxxxxxxxxx.

SR — Set positive software position limit

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **SR**nn, or **SR**?

Parameters

Description	**nn** [float]	—	Positive software limit.
Range	**nn**	—	≥0 and <1E12.
Units	**nn**	—	Preset units.
Defaults	**nn** Missing:		Error.
	Out of range:		Error.

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.

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

Errors

—	Unknown message code.
—	Parameter missing or out of range.
—	Execution not allowed.
—	Execution not allowed in NOT INIT state.
—	Execution not allowed in INIT state.
—	Execution not allowed in NOT REFERENCED state.
—	Execution not allowed in HOMING state.
—	Execution not allowed in MOVING state.

Rel. Commands	SL	—	Set negative software position limit.
Example	SR100	|	Set software positive limit to 100 units.

ST — Stop motion

Usage	☐ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	■ Moving

Syntax **ST**.

Parameters None.

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.

Errors

— Unknown message code.

— Execution not allowed.

— Execution not allowed in CONFIGURATION state.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

Example ST | Stop motion.

TB — Tell error string

Usage
■ Config ■ NotInit ■ NotRef ■ Disable ■ Jogging
■ Init ■ Homing ■ Ready ■ Moving

Syntax **TB**nn.

Parameters

Description **nn** [int] — Error code.

Range **nn** — 0 to ERROR CODE.

Units **nn** — None.

Defaults **nn** Missing: Error.
Out of range: Error.

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

Errors — Unknown message code.
— Parameter missing or out of range.
— Execution not allowed.

Rel. Commands TE — Tell error code.

Example TB@ | Tell explanation of error code @.
TB@ No error | Controller returns No error.

TE — Tell last error

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **TE**.

Parameters None.

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

— Unknown message code.

— Execution not allowed.

Rel. Commands TE — Tell error code.

Example

TE | Tell last error code.

TE@ | Controller returns @.

List of errors codes :.

A : Unknown Message Code.

B : Parameter out of Limits.

C : Scaling parameters dependance error.

D : Function Execution not Allowed.

E : Home sequence already started.

F : Function Execution not Allowed in NOT INITIALIZED mode.

G : Function Execution not Allowed in INITIALIZING mode.

H : Function Execution not Allowed in NOT REFERENCED mode.

I : Function Execution not Allowed in CONFIG mode.

J : Function Execution not Allowed in DISABLE mode.

K : Function Execution not Allowed in READY mode.

L : Function Execution not Allowed in HOMING mode.

M : Function Execution not Allowed in MOVING mode.

N : Function Execution not Allowed in JOGGING mode.

O : Target Position out of limit.

P : Current position out of software limit.

Q : Motion Timeout.

R : Motion Error.

S : USB Communication ERROR.

T : Gathering not completed.

U : Error during EEPROM access.

V : Estimated motion time >timeout.

TH — Tell set point position

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **TH**.

Parameters None.

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

— Unknown message code.

— Execution not allowed.

Rel. Commands

TP — Tell current position.

Example

TH | Tell set point position.

TH0 | Controller returns 0.

TP — Tell current position

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **TP**.

Parameters None.

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

— Unknown message code.

— Execution not allowed.

Rel. Commands

TH — Tell set point position.

Example

TP | Tell current position.

TP0 | Controller returns 0.

TS — Get positioner error and controller state

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **TS**.

Parameters None.

Description The TS command returns the error bits and the current Controller state.

Returns The TS command returns 8 characters (1TSabcdefgh).

The first character (a) represents the status bits in Hexadecimal.

The next 5 characters (bcdef) represent the error bits in Hexadecimal.

The last two characters (gh) represent the Controller state in Hexadecimal.

Status bits	
1	Bit end of run -
2	Bit end of run +
4	Bit ZM (not used)

Error bits	
00001	Bit end of run negative
00002	Bit end of run positive
00004	Bit current limit
00008	Bit rms current limit
00010	Bit Fuse Broken
00020	Bit following error
00040	Bit time out homing
00080	Bit bad SmartStage
00100	Bit Vin sense error (DC voltage too low)
00200	Bit Motor Driver Over Temperature Warning
00400	Bit Motor Driver Overcurrent shut-down or GVDD undervoltage protection occurred
00800	Bit Motor thermistance Error
01000	Bit Parameters EEPROM Error
02000	Bit Parameters Range Error
04000	Bit Sin/Cos Radius Error
08000	Bit Encoder quadrature error
10000	Bit AquadB output error
20000	Bit ISR Ratio error
40000	Bit Motion Done Timeout error
80000	Bit Power error

Controller state	
0A	NOT INITIALIZED: after reset
0B	NOT INITIALIZED: after CONFIG state
0C	NOT INITIALIZED: after INITIALAZING state
0D	NOT INITIALIZED: after NOT_REFERENCED state
0E	NOT INITIALIZED: after HOMING state
0F	NOT INITIALIZED: after MOVING state
10	NOT INITIALIZED: after READY state
11	NOT INITIALIZED: after DISABLE state
12	NOT INITIALIZED: after JOGGING state
13	NOT INITIALIZED: error, Stage type not valid
14	CONFIGURATION
1E	INITIALAZING: launch by USB
1F	INITIALAZING: launch by Remote Control
28	NOT_REFERENCED
32	HOMING: launch by USB
33	HOMING: launch by Remote Control
3C	MOVING
46	READY: after HOMING state
47	READY: after MOVING state
48	READY: after DISABLE state
49	READY: after JOGGING state
50	DISABLE: after READY state
51	DISABLE: after MOVING state
52	DISABLE: after JOGGING state
5A	JOGGING: after READY state
5B	JOGGING: after DISABLE state

Errors		—	Unknown message code.
Rel. Commands	TE	—	Tell last error.
Example	TS	|	Tell current status & errors.
	TS0040200F	|	Controller returns : No status bit. Sin/Cos Radius Error and following error. NOT INITIALIZED : after MOVING state.

VA — Set positioner motion velocity

Usage	■ Config	☐ NotInit	☐ NotRef	■ Disable	☐ Jogging
		☐ Init	☐ Homing	■ Ready	☐ Moving

Syntax **VA**nn, or **VA**?

VAM, or **VAM**?

Parameters

Description **nn** [float] — Velocity value.

Range **nn** — >1E-6 and <1E12.

Units **nn** — Preset units/s.

Defaults **nn** Missing: Error.

Out of range: Error.

Description In CONFIGURATION state, this command sets the maximum velocity value which can 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.

VAM command set the current velocity to the maximum velocity value.

VAM? Command returns the maximum velocity.

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

Errors

— Unknown message code.

— Parameter missing or out of range.

— Execution not allowed.

— Execution not allowed in NOT INIT state.

— Execution not allowed in INIT state.

— Execution not allowed in NOT REFERENCED state.

— Execution not allowed in HOMING state.

— Execution not allowed in MOVING state.

Rel. Commands AC — Set positioner acceleration.

Example VA50 | Set velocity to 50 units/s.

VE — Tell controller revision information

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **VE**.

Parameters None.

Description This command returns the Controller's revision information.

Errors —Unknown message code.

Rel. Commands TP — Tell current position.

Example

```
VE     | Get Controller revision information.
       | Controller returns VE DL Controller/Driver version 1.0.
```

ZT — List all configuration parameters

Usage

■ Config	■ NotInit	■ NotRef	■ Disable	■ Jogging
	■ Init	■ Homing	■ Ready	■ Moving

Syntax **ZT**.

Parameters None.

Description The ZT command returns the list of all current configuration parameters.

Errors —Unknown message code.

Rel. Commands TE — Tell error code.

Example

```
ZT |  Tell configuration parameters
PW1
…
AC500
…
…
PW0.
```

6.0 Connector Pinout

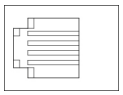
6.1 DC-IN (Kycon KPJX-4S-S or Equivalent)

Pin	Description
1	N.C.
2	+48 Vdc
3	Ground
4	N.C.
Shield	AC FG

6.2 USB Connector (USB Mini-B Receptacle)

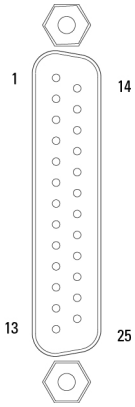
Pin	Description
1	N.C.
2	USBDM
3	USBDP
4	N.C.
5	Ground

6.3 Keypad (Female RJ9-4/4)



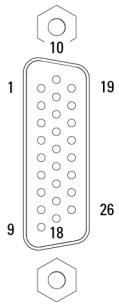
Pin	Description
1	+12 VDC
2	Tx
3	Rx
4	Ground

6.4 Stage Connector (DB25 Female with Female Lockers)



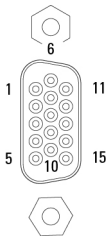
Pin	Description
1	U Motor
2	U Motor
3 & 4	N.C.
5	V Motor
6	V Motor
7	W Motor
8	W Motor
9	Reserved (Thermistor)
10 to 13	N.C.
14	Ground
15	N.C.
16	Ground
17	Reserved (SMDAT)
18	Reserved (SMCLK)
19 & 20	N.C.
21	+5 V
22	Ground
23 to 25	N.C.

6.5 Encoder Connector (DB26HD Female with Female Lockers)



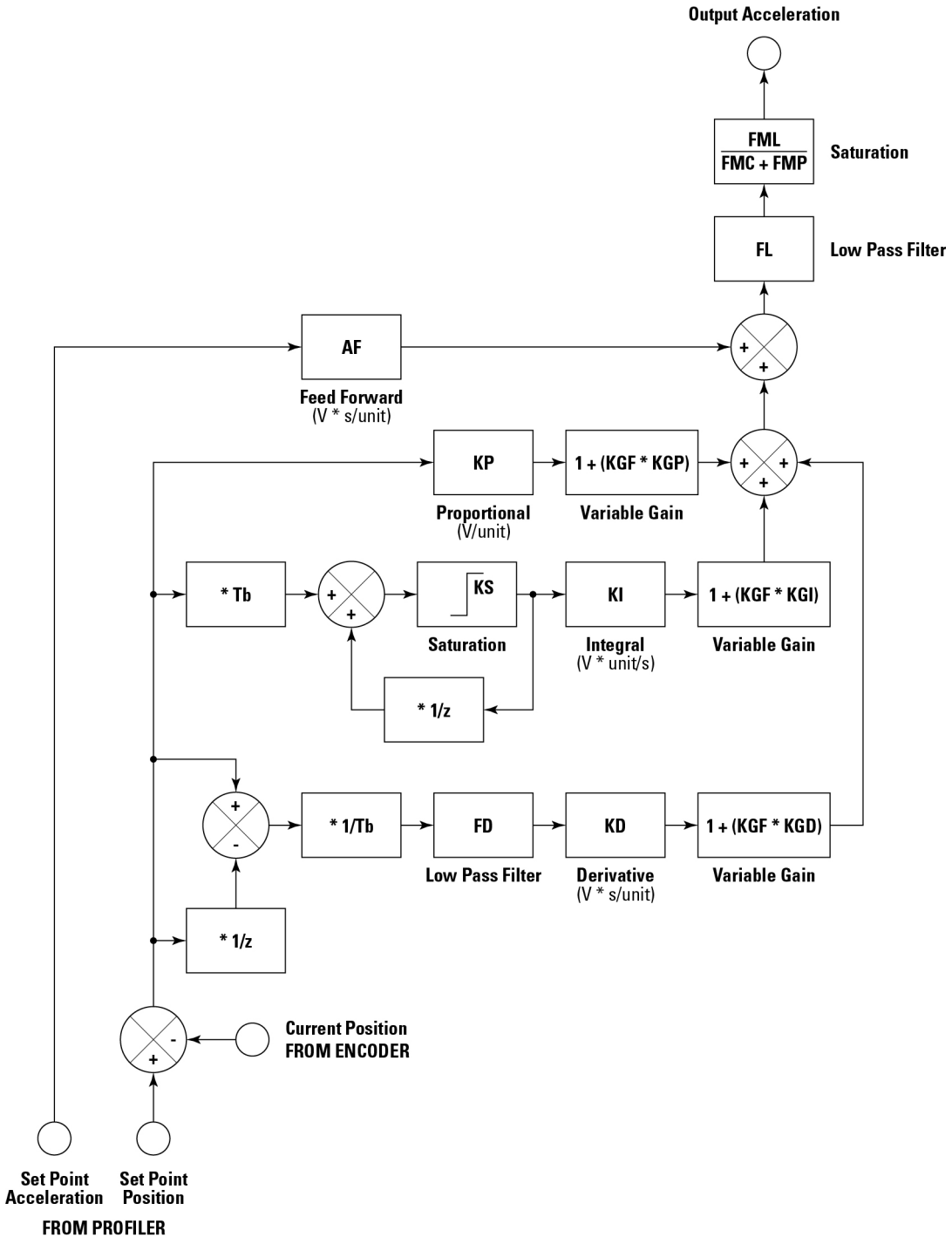
Pin	Description
1	+5 V
2 & 3	N.C.
4	Cosine
5	End-of-Run -
6	Sine
7	Ground
8	Index Pulse I
9 to 12	N.C.
13	/Cosine
14	End-of-Run +
15	/Sine
16	N.C.
17	Index Pulse /I
18	N.C.
19	+5V
20 to 24	N.C.
25	Ground
26	Ground

6.6 GPIO Connector (DB15HD Female with Female Lockers)



Pin	Description
1	Ground ADC
2	Quad A Out
3	Quad B Out
4	Reserved (Gathering)
5	Trigger Output 1 (PCO)
6	ADC1
7	ADC2
8	Quad A Out/
9	Quad B Out/
10	+5 V
11 & 12	N.C.
13	Trigger Input 2 (Start Motion)
14	Trigger Output 2 (In Motion)
15	Ground

7.0 PID Control Loop Structure



8.0 Troubleshooting

Here is a list of possible errors and the recommended corresponding actions to be taken prior to contact Newport.

GUI Error	"TS" Error	Description	Actions
End Of Run -	00001	Limit activated	• Check carriage position… • Check cables
End Of Run +	00002		
Current Limit	00004	The current set by the « QIL » command has been exceeded	• Wait current RMS time (QIT) • Check carriage "freedom"
RMS Current Limit	00008	The current set by the « QIR » command has been exceeded	• Check carriage "freedom"
Fuse broken	00010	Controller intrernal fuse (CMS) broken Or glitch on power suplly	• Cycle Power ON/OFF/ON
Following error	00020	Close loop (PID) or load (FMP) parameters not optimized Something interfers with the displacement.	• Check payload (FMP) parameter. Check carriage "freedom" • Restore Factory settings
Time out homing	00040	Home search sequence time exceeded time defined with « OT » parameter	• Check that Acceleration (AC) is not to low • Restore Factory settings
Bad smart stage	00080	Stage identifier is different from the one memorized in the controller	• Vérify stage (ID) parameter • Disable smart stage verification (ZX1)
Vin sense error….	00100	Power supply voltage lower than 42V caused by high peak current Or defective power supply	• Restore Factory settings
Bit Motor Driver Over Temp.Warning	00200	Motor driver over-temperature security warning stopping the controller	• Restore Factory settings
Motor driver overcurrent	00400	Motor driver over-current security error stopping the controller. Often due to a short circuit in the motor phases or wrong current loop (QC) parameters	• Restore Factory settings
Motor Thermistor error	00800	Not implemented in DLS series stages	
Param EEPROM error	01000	Wrong parameters checksum (EEPROM)	• Restore Factory settings
Param Range error	02000	One of the memorized parameters is out of range	• Restore Factory settings
Sin/Cos radius	04000	Incorrect encoder signals amplitude	• Check encoder cable
Encoder Quad	08000	Incoherent encoder signals	• Check encoder cable
Bit AquadB output error	10000	« AquadB » output activated but cannot follow encoder feedback.	• Set « AquadB » output interpolation to minimum (EQR2) • Lower « AquadB » output filter frequency (EQF) • Lower interpolation factor (EQR) • Lower displacement speed (VA) • Restore Factory settings
ISR Ratio	20000	CPU/Firmware Internal fatal error	• Contact Newport
Motion Done Timeout	40000	Stage not stabilized within expected motion time	• Check carriage "freedom" • Verify Close loop PID parameters • Check MD parameters • Restore Factory settings
Bit Power	80000	Maximum power reached stopping the controller	• Check carriage "freedom" • Restore Factory settings • Contact Newport

9.0 Maintenance and Service

9.1 Enclosure Cleaning

The DL 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.

9.2 Obtaining Service

The DL 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