# *SCL User Manual for STP-DRV and STP-MTRD*

***SCL Commands for STP-DRV-4850 and STP-DRV-80100 Step Motor Drives***

***and STP-MTRD Advanced Integrated Motor/Drives***


AUTOMATIONDIRECT.com

3rd Ed. Rev B
07/03/2019

BLANK
PAGE

# SureStep Advanced Microstepping Drives Serial Command Language User Manual

**Please include the Manual Number and the Manual Issue, both shown below, when communicating with Technical Support regarding this publication.**


***Manual Number:*** ***STP-DRV-SCL_UMW***

***Issue:*** ***Third Edition, Revision A***

***Issue Date:*** ***11/20/2018***


| Publication History | | |
|---|---|---|
| ***Issue*** | ***Date*** | ***Description of Changes*** |
| First Edition | 10/03/2008 | Original Issue (920-0019 Rev. B1) |
| Second Edition | 09/02/2014 | Completely rewritten |
| 2nd Edition, Revision A | 03/08/2018 | AM Command, Alarm Codes, LED Display Codes |
| Third Edition | 07/02/2018 | Addition of 40+ new SCL commands for SureStep Integrated Motors and SureMotion Pro release. |
| 3rd Edition, Revision A | 11/20/2018 | Added additional details for RS-485 queries and addressing, analog positioning, and some commands. |
| 3rd Edition, Revision B | 07/03/2019 | Fixed error in amperage range for stepper motors listed under CA command. |

TABLE OF CONTENTS

## Introduction

Thank you for purchasing an Automation Direct stepper drive. We hope you will find that performance, price, and ease of use make our products the best value for your application.

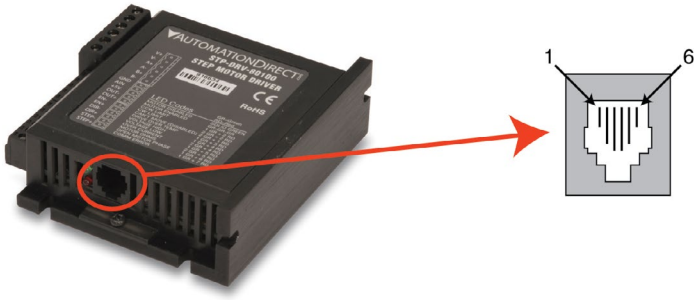
The Serial Command Language (SCL) can be used with Automation Direct advanced stepper drives and this manual focuses on using SCL with these drives only.

*NOTE: This manual only covers details related to using SCL with the advanced drives. For all other aspects of applying your advanced drive, including hardware configuration, I/O, and software settings, view the Help file contained in the SureMotion Pro software. This software can be downloaded for free from www.automationdirect.com.*

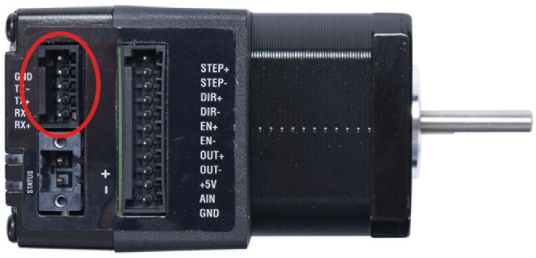
## What is SCL?

SCL was developed to give users a simple way to control a motor drive via a serial port. This eliminates the need for separate motion controllers or indexers to supply Pulse & Direction signals to your stepper drive. It also provides an easy way to interface to a variety of other industrial devices like PLCs and HMIs, which most often have standard or optional serial ports for communicating to other devices.

STP-DRV-4850 and -80100 drives come with one RS-232 serial port. This port is an RJ-11 jack (6P4C) as shown in the picture below.




The STP-MTRD drives have a 5-pin connector, as shown below:




To use SCL in an application means you will have a host device, such as a PC, a PLC, or an HMI, connected to the drive's serial port and using that connection to send commands to the drive. The set of commands defined by SCL includes commands for motion of the step motor, commands for using the three digital inputs, one analog input, and one digital output of the drive, as well as commands for configuring different aspects of the drive like motor current and microstep resolution.

When in SCL mode, the drive receives commands from the host into a command buffer, and then executes the received commands directly out of that buffer.

*NOTE: One thing you cannot do with the drive is create a stored program that the drive can run stand-alone.*

## SCL Details

There are two basic parts to the serial communications used in SCL: the physical connection between the drive and the host, and the ASCII communication language.

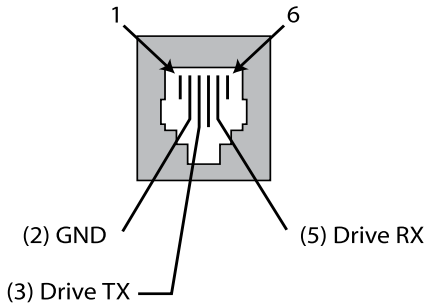
The physical connection between the drive and the host is based on either standard RS-232 connections or an RS-485 connection. Using a PC and a USB to serial adapter a connection can easily be established. With RS-232 or RS-485 (2-wire) there are only three connections to be made between the drive and the host: transmit (Tx), receive (Rx), and signal ground (GND). For configuring a STP-MTRD drive, a 4-wire RS-485 connection is required.

For the RS-232 drives, the STP-232RJ11-CBL configuration cable and the USB-RS232-1 USB adapter will allow PC communication. For the RS-485 drives, the STP-USB485-4W USB to serial adapter along with an STP-485DB9-CBL-2 will allow PC communication.

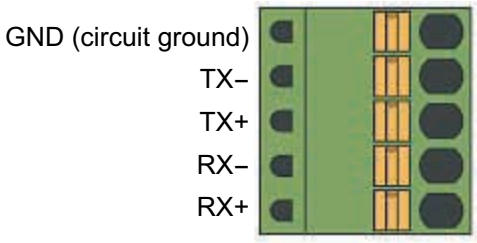
The STP-232RJ11-CBL can also be used with the STP-USB485-4W USB serial converter for communication to RS-232 drives. Please see the communications chapter of the SureStep User Manual for specific wiring examples.

The pin assignments of these connections on an STP-DRV drive are shown in the following diagram.

*WARNING: Power down the SureStep drive before plugging a communication cable in to the comm port of the drive. Failure to do so may result in damage to the drive comm port!*




*RS-232 Connection*




*RS-485 Connection*

To configure your host to properly communicate with the drive you'll need to configure your host's serial port as follows: 9600 bps, 8 data bits, 1 stop bit, no parity. These are the default COM port settings for a Windows-based PC.

The communications protocol of SCL is simple in that all communications are initiated by the host. The only communication the drive will ever initiate is at power-up of the drive. At power-up the drive sends a "power-up packet", which is simply an identifier that is used by Automation Direct software applications. This identifier tells our software which drive is connected and what its firmware version is. Other than that, all communications are initiated by the host.

The basic structure of a command packet from the host to the drive is always a text string followed by a carriage return (no line feed required). The text string is always composed of the command itself, followed by any parameters used by the command. If using an RS-485 drive then every command needs to be preceded by an address character. Refer to the DA command or the "Getting Started >> RS-485" topic in the next section for more details. The carriage return denotes the end of transmission to the drive. Here is the basic syntax.

**XXAB<cr>**

In the syntax above, "XX" designates the SCL command.

*NOTE: The letters use for "XX" in the SCL command MUST be entered in upper case.*

"A" designates the first of two possible parameters, and "B" designates the second. Parameters 1 and 2 vary in length, can be letters or numbers, and are often optional.

Example SCL command using set direction (SD):

**SD4I**

This will set the direction of I/O point 4 as an input on a variable I/O drive.

Once a drive receives the <cr> it will determine whether or not it understood the command. If it did understand the command, the drive will either execute or buffer the command. The drive will also send an Acknowledge character (ACK) back to the host.

| ***Acknowledge Character*** | ***Meaning*** |
|---|---|
| % | Executed command |
| * | Buffered command |
| ? | NACK - did not understand |

# Getting Started

To get up an running with your advanced drive and SCL as quickly as possible, follow the basic steps below.

## *Step 1: Install software*

The *Sure*Step advanced drives are configured using *SureMotion Pro*™ configuration software, which is available for download from the Automationdirect.com website.

Install *SureMotion Pro* on your Windows-based PC. *SureMotion Pro* will be used to configure your drive and put it into SCL mode.

*NOTE: Laptop computers without a serial port will require a USB-Serial adapter or PCMCIA-Serial adapter. Automation Direct offers a USB to serial adapter (part number USB-RS232-1 or STP-USB485-4W) that is suitable for use in these applications.*

## *Step 2: Configure your Advanced drive using SureMotion Pro*

If you haven't already done so, unpack your STP-DRV drive and step motor and collect them together near your PC. You're going to need the following items to begin developing your application.

- An advanced SureStep stepper drive
- A 2-phase step motor. Automation Direct offers a number of step motors specifically chosen for use with the SureStep series drives. A drop-down list of these motors is contained in the Motor window of *SureMotion Pro*. If you have a different step motor view the Help file in *SureMotion Pro* for details on setting up a custom motor.
- The programming cable supplied with your STP-DRV drive (9-pin D-sub at one end, RJ-11 modular connector at the other, or an STP-USB485-4W USB to serial adapter along with an STP-485DB9-CBL-2 cable for the STP-MTRD drives).
- A small, flat-blade screwdriver.
- A correctly sized DC power supply.

*WARNING: Never connect a step motor to your STP-DRV drive with power applied to the drive. Always make sure your DC power supply is either off or disconnected from the STP-DRV drive when connecting or disconnecting your step motor.*

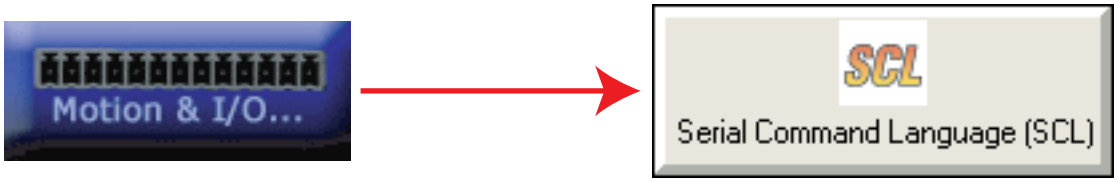
*WARNING: Power down the SureStep drive before plugging a communication cable in to the comm port of the drive. Failure to do so may result in damage to the drive comm port!*

Connect the 2-phase step motor to your STP-DRV drive. Then connect your STP-DRV drive to your PC using the programming cable. Launch *SureMotion Pro*. Power your drive ON. If power was ON to the drive when you launched *SureMotion Pro* power the drive OFF, then back ON.

*NOTE: As long as the drive is powered on after the correct port settings have been adjusted, SureMotion Pro will recognize the correct model of the drive connected.*

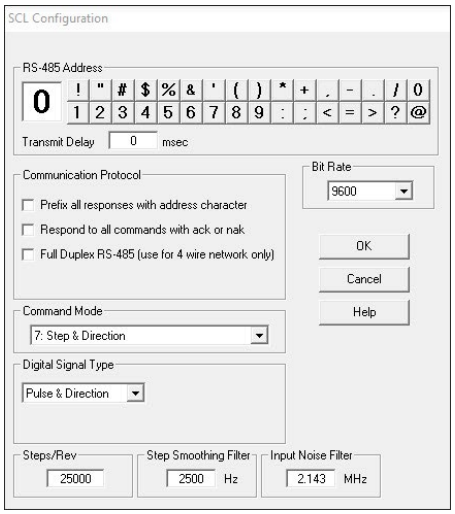
Configure your motor by clicking the "Motor" button on the main screen of *SureMotion Pro*. Use the Help file contained in *SureMotion Pro* for details on configuring your drive for the particular step motor you have. If you have a recommended Automation Direct step motor you may simply find that part number in the drop-down menu of the Motor window.

Each advanced drive can run in one of four Motion Control Modes: Pulse & Direction, Analog Velocity, Fixed Velocity, or Point to Point (SCL). Click the "Motion & I/O" button in SureMotion Pro, then select SCL. This will bring up the SCL Configuration window.

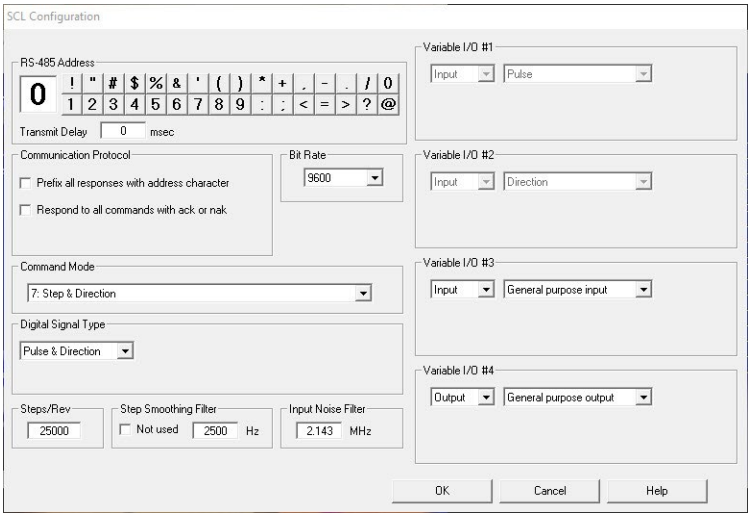


## GETTING STARTED – STEP 2 (CONTINUED)

In the SCL Configuration window there are a number of settings you can make that affect how the drive operates while in SCL mode. Which drive you have connected will determine which version of the SCL dialog window gets displayed. Below are the different variations of the SCL dialog window.

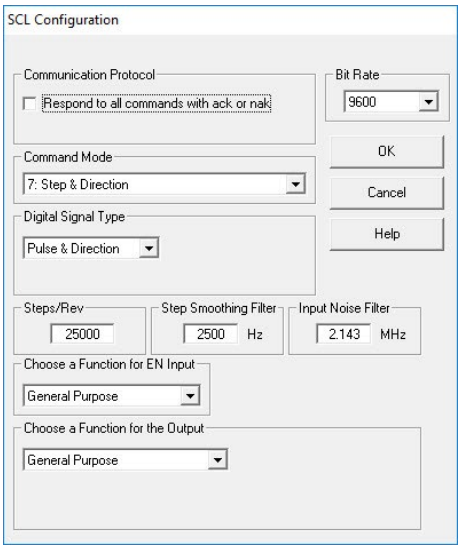



*SCL dialog window when connected to an RS-485 drive without Variable I/O*




*SCL dialog window when connected to an RS-485 drive with Variable I/O*

**RS-485 Address** contains all the characters you can use for defining an address to each drive. See also DA command. Transmit delay time can also be configured here. If a drive is assigned an address (1) then it will respond to commands prefixed by that address (for example: 1FL2000) but it will also respond to the same command without the address (example: FL2000). Commands without an address are known as global commands. The only difference is that the drive will not send an acknowledgement back to the host when given a global command.




*SCL dialog window when connected to an RS-232 only drive*

If you have 4 addressed drives on a network and you send FL2000 then all drives will respond to the command but none will send an acknowledgement.

All RS-485 queries require an address character. Commands that do not require a response can be sent globally. Example: you can send an AR, FL or MD to all axes at once by omitting the address char. But if you send a query like AL or SC, the drive will want to respond. Since you only want one drive responding to each query to avoid network collisions, for queries the address is required.

**Communication Protocol** contains settings to turn on ACK/NACK and the ability to set all responses to include the address character. ACK/NACK is a useful setting because the drive will respond to each command it receives with an ACK (% or * sign) or NACK (? sign).

**Command Mode** sets which mode the drive will power up in. There are a number of different modes the drive can operate in and still communicate with SCL commands. Most applications will at least start out in Command Mode 21: Point-to-point Positioning. See details on the Command Mode (CM) command for more information.

**Digital Signal Type** only appears in command mode 7. This allows you to configure what type of pulse commands you plan to use.

Once you've set up the motor in *SureMotion Pro* and have chosen SCL mode for the drive, click the Download to Drive button on the main screen.


### *Step 3: Get familiar with SCL commands*

After downloading, click the Drive menu, then SCL Terminal. This opens the Host Terminal window. To send commands to your drive simply type a command in the Command Line of the Host Terminal and press the ENTER key to send it. (Remember that all commands are capital letters so pressing the Caps Lock key first is a good tip). Pressing the ENTER key while in the Host Terminal does two things: it terminates the command with a <cr> and automatically sends the entire string. Try the example sequence below. In this example, note that <ENTER> means press the ENTER key on your keyboard, which is the same as terminating the command with a <cr>.

*WARNING: We recommend practicing with SCL commands with no load attached to the motor shaft. You want the motor shaft to spin freely during startup to avoid damaging mechanical components in your system.*

| | |
|---|---|
| AC25<ENTER> | Set accel rate to 25 rev/sec/sec |
| DE25<ENTER> | Set decel rate to 25 rev/sec/sec |
| VE5<ENTER> | Set velocity to 5 rev/sec |
| FL20000<ENTER> | Move the motor 20000 steps in the CW direction. |

If your motor didn't move after sending the FL20000 check the LEDs on your drive to see if there is an error present. If so send the AR command (AR<ENTER>) to clear the alarm. If after clearing the alarm you see a solid green LED it means the drive is disabled. Enable the drive by sending the ME command (ME<ENTER>) and verify that the you see a steady, flashing green LED. You might also want to review your settings in SureMotion Pro to make sure the motor current is set properly. Then try the above sequence again. Make sure you are using uppercase commands (FL20000, not fl20000).

Here is another sample sequence you can try.

| | |
|---|---|
| JA10<ENTER> | Set jog accel rate to 10 rev/sec/sec |
| JL10<ENTER> | Set jog decel rate to 10 rev/sec/sec |
| JS1<ENTER> | Set jog speed to 1 rev/sec |
| CJ<ENTER> | Commence jogging |
| CS-1<ENTER> | Change jog speed to 1 rev/sec in CCW direction |
| SJ<ENTER> | Stop jogging |

In the above sequence notice that the motor ramps to the new speed set by CS. This ramp is affected by the JA and JL commands. Try the same sequence above with different JA, JL, JS, and CS values to see how the motion of the motor shaft is affected.

### *STEP 4: DEVELOP YOUR APPLICATION*

This step will involve different things for different users. You'll probably want to spend sufficient time getting familiar with SCL commands using the SCL Terminal window before getting to this step, but once you have consider the following:

If your host is a PC you've already done a lot of the hardware configuration necessary for your application. The rest of your application will involve developing your PC applications to properly send SCL commands to your drive. Which application or language you use, whether it be VisualBasic, C+, LabView, a proprietary vision system application, or something else, is up to you.

If your host is a PLC you'll have to connect and configure the ASCII module, RS-232 port or Aux serial port on your PLC according to the pin assignments and COM port settings listed in the Introduction section. From there you'll have to be able to send text strings followed by carriage returns from the PLC.

*NOTE: Once the application is ready to be started, the software must be disconnected and the drive powered down for at least 10 seconds.*

## Commands

There are two basic types of SCL commands: buffered and immediate. Buffered commands are loaded into and executed out of your drive's command buffer. Immediate commands are not buffered: when received by the drive they are executed immediately.

### *Buffered Commands*

After being loaded into the command buffer, buffered commands are executed one at a time. If you send two buffered commands to the drive in succession, like a Feed to Length (FL) command followed by a Send String (SS) command, the SS command sits in the command buffer and waits to execute until the FL command is completed. The command buffer can be filled up with commands for sequential execution without the host controller needing to wait for a specific command to execute before sending the next command. Special buffer commands, like Pause (PS) and Continue (CT), enable the buffer to be loaded and to pause execution until the desired time.

### *Immediate Commands*

Immediate commands are executed right away, running in parallel with a buffered command if necessary. For example, this allows you to check the remaining space in the buffer using the Buffer Status (BS) command, or the immediate status of digital inputs using the Input Status (IS) command, while the drive is processing other commands. Immediate commands are designed to access the drive at any time and can be sent as often as needed. This allows a host controller to get information from the drive at a high rate, most often for checking drive status or motor position.

## Command Device Compatibility

The table below details the compatibility of each command with supported ADC drives, as well as noting whether each command is non-volatile, read/write, write only, or read only, and if the command is buffered or executed immediately. If a command is not marked as immediate, then it is considered to be a buffered command. All volatile commands can be saved.

| Command | Description | Non-volatile | Read/Write | Write Only | Read Only | Immediate | STP-DRV-4850 STP-DRV-80100 (All Models) | STP-MTRD-xR (All Models) | Any Drive with Variable I/O (V Models) | Any Drive with Encoder Feedback (E Models) |
|---|---|---|---|---|---|---|---|---|---|---|
| **AC** | Accel Rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AD** | Analog Deadband | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AF** | Analog Filter | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AG** | Analog Velocity Gain | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AI** | Alarm Reset Input | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AL** | Alarm Code | | | | ✓ | ✓ | ✓ | ✓ | | |
| **AM** | Accel Max | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AO** | Alarm Output | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AP** | Analog Position Gain | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AR** | Alarm Reset | | | ✓ | | ✓ | ✓ | ✓ | | |
| **AT** | Analog Threshold | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AV** | Analog Offset | ✓ | ✓ | | | | ✓ | ✓ | | |
| **AZ** | Analog Zero (Auto Zero) | | | ✓ | | | ✓ | ✓ | | |
| **BD** | Brake Disengage Delay Time | ✓ | ✓ | | | | ✓ | ✓ | | |
| **BE** | Brake Engage Delay Time | ✓ | ✓ | | | | ✓ | ✓ | | |
| **BO** | Brake Output | ✓ | ✓ | | | | ✓ | ✓ | | |
| **BR** | Baud Rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **BS** | Buffer Status | | | | ✓ | ✓ | ✓ | ✓ | | |
| **CA** | Change Acceleration Current | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CC** | Change Current | ✓ | ✓ | | | | | ✓ | | |
| **CD** | Idle Current Delay | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CE** | Communications Error | | | | ✓ | ✓ | ✓ | ✓ | | |
| **CF** | Anti-resonance Filter Frequency | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CG** | Anti-resonance Filter Gain | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CI** | Change Idle Current | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CJ** | Commence Jogging | | | ✓ | | | ✓ | ✓ | | |
| **CM** | Control Mode | ✓ | ✓ | | | | ✓ | ✓ | | |
| **CS** | Change Speed | | ✓ | | | ✓ | ✓ | ✓ | | |
| **CT** | Continue | | | ✓ | | ✓ | ✓ | ✓ | | |
| **DA** | Define Address | ✓ | ✓ | | | | | ✓ | | |
| **DC** | Distance for FC, FM, FO, FY | ✓ | ✓ | | | | ✓ | ✓ | | |

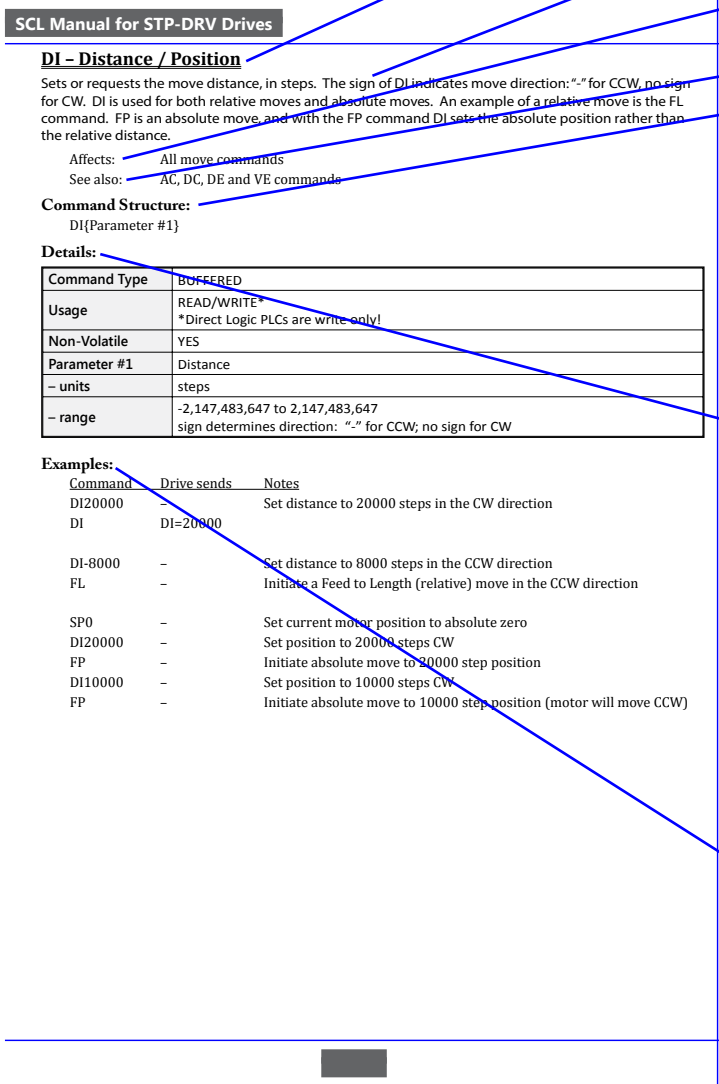
| Command | Description | Non-volatile | Read/Write | Write Only | Read Only | Immediate | STP-DRV-4850 STP-DRV-80100 (All Models) | STP-MTRD-xR (All Models) | Any Drive with Variable I/O (V Models) | Any Drive with Encoder Feedback (E Models) |
|---|---|---|---|---|---|---|---|---|---|---|
| **DE** | Decel Rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **DI** | Distance or Position | ✓ | ✓ | | | | ✓ | ✓ | | |
| **DL** | Define Limits | ✓ | ✓ | | | | ✓ | ✓ | | |
| **ED** | Encoder Direction | ✓ | ✓ | | | | | | | ✓ |
| **EF** | Encoder Function | ✓ | ✓ | | | | | | | ✓ |
| **EG** | Electronic Gearing | ✓ | ✓ | | | | ✓ | ✓ | | |
| **EI** | Input Noise Filter | ✓ | ✓ | | | | ✓ | ✓ | | |
| **EP** | Encoder Position | | ✓ | | | | | | | ✓ |
| **FC** | Feed to Length with Speed Change | | | ✓ | | | ✓ | ✓ | | |
| **FD** | Feed to Double Sensor | | | ✓ | | | ✓ | ✓ | | |
| **FE** | Follow Encoder | | | ✓ | | | ✓ | ✓ | | |
| **FI** | Filter Input | ✓ | ✓ | | | | ✓ | ✓ | | |
| **FL** | Feed to Length | | | ✓ | | | ✓ | ✓ | | |
| **FM** | Feed to Sensor with Mask Dist | | | ✓ | | | ✓ | ✓ | | |
| **FO** | Feed to Length & Set Output | | | ✓ | | | ✓ | ✓ | | |
| **FP** | Feed to Position | | | ✓ | | | ✓ | ✓ | | |
| **FS** | Feed to Sensor | | | ✓ | | | ✓ | ✓ | | |
| **FY** | Feed to Sensor with Safety Dist | | | ✓ | | | ✓ | ✓ | | |
| **HW** | Hand Wheel | | | ✓ | | | ✓ | ✓ | | |
| **IA** | Immediate Analog | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IC** | Immediate Current | | | | ✓ | ✓ | ✓ | ✓ | | |
| **ID** | Immediate Distance | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IE** | Immediate Encoder | | | | ✓ | ✓ | | | | ✓ |
| **IF** | Immediate Format | | ✓ | | | ✓ | ✓ | ✓ | | |
| **IH** | Immediate High Output | | | ✓ | | ✓ | ✓ | ✓ | | |
| **IL** | Immediate Low Output | | | ✓ | | ✓ | ✓ | ✓ | | |
| **IO** | Output Status | | ✓ | | | ✓ | ✓ | ✓ | | |
| **IP** | Immediate Position | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IS** | Input Status request | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IT** | Immediate Temperature | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IU** | Immediate Voltage | | | | ✓ | ✓ | ✓ | ✓ | | |
| **IV** | Immediate Velocity | | | | ✓ | ✓ | ✓ | ✓ | | |
| **JA** | Jog Accel/Decel rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **JC** | Velocity mode second speed | ✓ | ✓ | | | | ✓ | ✓ | | |
| **JD** | Jog Disable | | | ✓ | | | ✓ | ✓ | | |
| **JE** | Jog Enable | | | ✓ | | | ✓ | ✓ | | |

| Command | Description | Non-volatile | Read/Write | Write Only | Read Only | Immediate | STP-DRV-4850 STP-DRV-80100 (All Models) | STP-MTRD-xR (All Models) | Any Drive with Variable I/O (V Models) | Any Drive with Encoder Feedback (E Models) |
|---|---|---|---|---|---|---|---|---|---|---|
| **JL** | Jog Decel rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **JS** | Jog Speed | ✓ | ✓ | | | | ✓ | ✓ | | |
| **LV** | Low Voltage Threshold | ✓ | ✓ | | | | ✓ | ✓ | | |
| **MC** | Motor Current, Rate | ✓ | ✓ | | | | ✓ | | | |
| **MD** | Motor Disable | | | ✓ | | | ✓ | ✓ | | |
| **ME** | Motor Enable | | | ✓ | | | ✓ | ✓ | | |
| **MO** | Motion Output | ✓ | ✓ | | | | ✓ | ✓ | | |
| **MV** | Model & Revision | | | | ✓ | ✓ | ✓ | ✓ | | |
| **PA** | Power-up Acceleration Current | ✓ | ✓ | | | | | ✓ | | |
| **PB** | Power up Baud Rate | ✓ | ✓ | | | | ✓ | ✓ | | |
| **PC** | Power up Current | ✓ | ✓ | | | | ✓ | ✓ | | |
| **PF** | Position Fault | ✓ | ✓ | | | | | | | ✓ |
| **PI** | Power up Idle Current | ✓ | ✓ | | | | ✓ | ✓ | | |
| **PM** | Power up Mode | ✓ | ✓ | | | | ✓ | ✓ | | |
| **PN** | Probe On Demand | | | ✓ | | | ✓ | ✓ | | |
| **PR** | Protocol | ✓ | ✓ | | | | ✓ | ✓ | | |
| **PS** | Pause | | | ✓ | | | ✓ | ✓ | | |
| **RE** | Restart / Reset | | | ✓ | | ✓ | ✓ | ✓ | | |
| **RO** | Anti-Resonance ON | ✓ | ✓ | | | | ✓ | ✓ | | |
| **RS** | Request Status | | | | ✓ | ✓ | ✓ | ✓ | | |
| **RV** | Revision Level | | | | ✓ | ✓ | ✓ | ✓ | | |
| **SA** | Save all NV Parameters | | | ✓ | | ✓ | ✓ | ✓ | | |
| **SC** | Status Code | | | | ✓ | ✓ | ✓ | ✓ | | |
| **SD** | Set Direction | ✓ | ✓ | | | | | | ✓ | |
| **SF** | Step Filter Frequency | ✓ | ✓ | | | | ✓ | ✓ | | |
| **SH** | Seek Home | | | ✓ | | | ✓ | ✓ | | |
| **SI** | Enable Input usage | ✓ | ✓ | | | | ✓ | ✓ | | |
| **SJ** | Stop Jogging | | | ✓ | | ✓ | ✓ | ✓ | | |
| **SK** | Stop & Kill Buffer | | | ✓ | | ✓ | ✓ | ✓ | | |
| **SO** | Set Output | | | ✓ | | | ✓ | ✓ | | |
| **SP** | Set Absolute Position | | ✓ | | | | ✓ | ✓ | | |
| **SS** | Send String | | | ✓ | | | ✓ | ✓ | | |
| **ST** | Stop Motion | | | ✓ | | ✓ | ✓ | ✓ | | |
| **TD** | Transmit Delay | ✓ | ✓ | | | | ✓ | ✓ | | |
| **VC** | Velocity for Speed Change (FC) | ✓ | ✓ | | | | ✓ | ✓ | | |
| **VE** | Velocity Setting (For Feed Commands) | ✓ | ✓ | | | | ✓ | ✓ | | |
| **VM** | Velocity Max | ✓ | ✓ | | | | ✓ | ✓ | | |

| Command | Description | Non-volatile | Read/Write | Write Only | Read Only | Immediate | STP-DRV-4850 STP-DRV-80100 (All Models) | STP-MTRD-xR (All Models) | Any Drive with Variable I/O (V Models) | Any Drive with Encoder Feedback (E Models) |
|---|---|---|---|---|---|---|---|---|---|---|
| ***WI*** | Wait for Input | | | ✓ | | | ✓ | ✓ | | |
| ***WT*** | Wait Time | | | ✓ | | | ✓ | ✓ | | |

## *COMMAND LISTING*

This section is an alphabetical listing of all the commands available with your drive. Each page in this section contains the details of one available command. Below is a sample of what these pages look like, with an explanation of the information you will find on each page.



SCL Manual for STP-DRV Drives

**DI – Distance / Position**

Sets or requests the move distance, in steps. The sign of DI indicates move direction: "-" for CCW, no sign for CW. DI is used for both relative moves and absolute moves. An example of a relative move is the FL command. FP is an absolute move, and with the FP command DI sets the absolute position rather than the relative distance.

Affects: All move commands
See also: AC, DC, DE and VE commands

**Command Structure:**

DI{Parameter #1}

**Details:**

| | |
|---|---|
| Command Type | BUFFERED |
| Usage | READ/WRITE*<br>*Direct Logic PLCs are write-only! |
| Non-Volatile | YES |
| Parameter #1 | Distance |
| – units | steps |
| – range | -2,147,483,647 to 2,147,483,647<br>sign determines direction: "-" for CCW; no sign for CW |

**Examples:**

| Command | Drive sends | Notes |
|---|---|---|
| DI20000 | – | Set distance to 20000 steps in the CW direction |
| DI | DI=20000 | |
| | | |
| DI-8000 | – | Set distance to 8000 steps in the CCW direction |
| FL | – | Initiate a Feed to Length (relative) move in the CCW direction |
| | | |
| SP0 | – | Set current motor position to absolute zero |
| DI20000 | – | Set position to 20000 steps CW |
| FP | – | Initiate absolute move to 20000 step position |
| DI10000 | – | Set position to 10000 steps CW |
| FP | – | Initiate absolute move to 10000 step position (motor will move CCW) |

Title – shows the command's two-letter command code followed by the command's name.

Description – an explanation of what the command does and how it works.

Affects – a summary of parameters or other commands the command affects.

See Also – related commands

Command Structure – shows the command's syntax. The format for this line is always the two-letter command code, followed by the number of parameters it uses. Not all commands have parameters, some commands have optional parameters, and other commands always have a parameter. Optional parameters are designated by { }, and required parameters are designated by ( ).

Details – shows the "Command Type" (buffered or immediate), the command's "Usage" (Read Only, Read/Write, or Write Only), and whether the command is "Non-Volatile" or not. Non-Volatile commands are saved when the Save (SA) command is sent. Also, the details of the command's parameter(s) are shown. Parameter #1 or #2 gives a brief description of the parameter, "– units" shows how the parameter is interpreted by the drive, "– range" gives the acceptable range of values for the parameter, and "– default" shows the default value of the parameter.

Examples – shows what to expect when you use this command. Under "Command" are the command strings you would send from a host controller. Note that <cr> is not shown after each command string in these examples but is still necessary to terminate the string. Under "Drive Sends" are the responses from the drive: no response from the drive is denoted by "–", although if Ack/Nack is turned on there will always be a response to every command sent. "Comments" gives additional information about the results of the command string.

## AC – Acceleration Rate

Sets or requests the acceleration rate used in all "F" (point-to-point) moves in rev/sec/sec.

Affects: FC, FL, FM, FP, FS, FY, SH Commands

See also: DE, DI, DC, VE Commands

***COMMAND STRUCTURE:***

AC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Acceleration rate |
| ***– units*** | rev/sec/sec (rps/s) |
| ***– range*** | 0.167 to 5461.167 (resolution is 0.167 rps/s) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC100 | – | Set Acceleration rate to 100 rev/sec/sec |
| AC | AC=100 | |
| AC25 | – | Set Acceleration rate to 25 rev/sec/sec |
| DE25 | – | Set Deceleration rate to 25 rev/sec/sec |
| VE1.5 | – | Set Velocity to 1.5 rev/sec |
| FL20000 | – | Execute Feed to Length move of 20000 steps in CW direction |

## AD – Analog Deadband

Sets or requests the analog deadband value in millivolts. The deadband value is the zone around the "zeroed" value of the analog input. This deadband defines the area of the analog input range that the drive should interpret as the zero velocity point in analog velocity modes. The deadband is an absolute value that in usage is applied to either side of the zero point.

Affects: Analog input

See also: CM command

***COMMAND STRUCTURE:***

AD{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog deadband value |
| ***– units*** | millivolts |
| ***– range*** | 0–255 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AD100 | – | Set analog deadband to 0.1 volts |
| AD | AD=100 | |

## AF – Analog Filter

Applies a digital filter to the analog input. This is a simple single pole filter that rolls off the analog input. The filter value of the AF command is related to the desired value of the analog filter in Hz by the following equation:

- Filter value = 72090 / [ (1400 / x ) + 2.2 ]
  where x = desired value of the analog filter in Hz

Affects: Analog input

See also: IA, CM commands

### *Command Structure*

AF{Parameter #1}

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Filter value |
| ***– units*** | integer (see formula above) |
| ***– range*** | 0–32767* (0 disables the filter) |
| ** An AF value of 28271 equates to 4000.425 Hz. Setting the AF command to anything higher than 28271 has a negligible effect on the analog filter. In other words, the maximum value of the filter is approximately 4000 Hz.* | |

### *Examples:*

| Command | Drive sends | Notes |
|---|---|---|
| AF5000 | – | Make the analog input bandwidth 114.585 Hz |
| AF | AF=5000 | |

## AG – Analog Velocity Gain

Sets or requests the gain value used in analog velocity modes. The gain value is used to establish the relationship between the analog input and the motor speed. The units are 0.25 rpm. For example, if the gain is set to 2400, when 5 Volts is read at the analog input the motor will spin at 10 rps. TIP: To set the analog velocity gain to the desired value, multiply the desired motor speed in rps by 240, or the desired motor speed in rpm by 4.

Affects: Analog velocity mode

See also: CM command

### *Command Structure:*

AG{Parameter #1}

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog velocity gain value |
| ***– units*** | 0.25 rpm |
| ***– range*** | -32767 to 32767 |

### *Examples:*

| Command | Drive sends | Notes |
|---|---|---|
| AG3000 | – | Set top speed of analog velocity mode to 12.5 rps |
| AG | AG=3000 | |

## AI – Alarm Reset Input

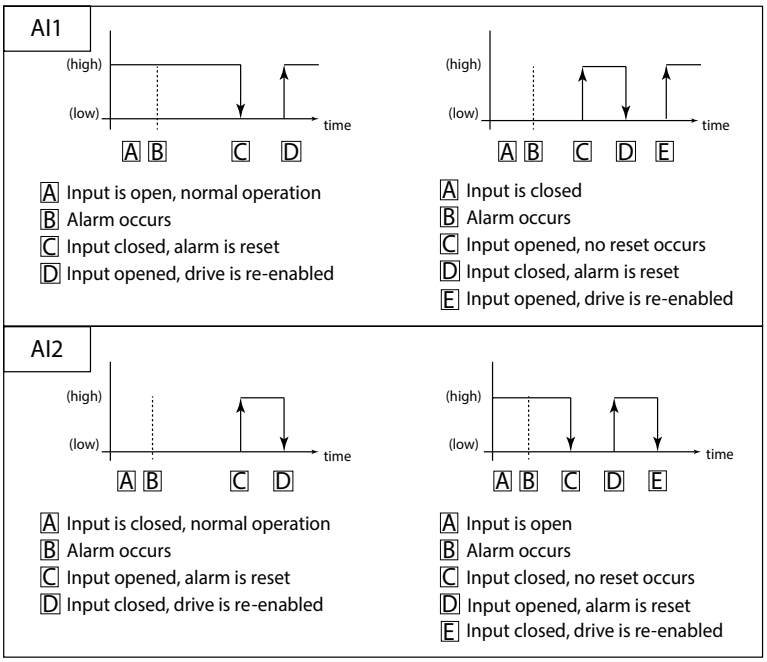
**For Standard Input Drives:**

Defines the EN input or a Variable I/O input as an Alarm Reset Input. AI takes no effect if the drive is set in Command Mode (CM) 13, 14, 17, or 18, because these modes use the EN input as a speed change input and take precedence over the AI command. Setting the SI command after setting the AI command reassigns the EN input to drive enable usage and turns off any alarm reset usage (AI3).

There are three Alarm Reset Input states that can be defined with the AI command:

AI1: For normal operation, the EN input must be open (inactive, high). Alarm reset occurs when the EN input is closed (active, low). This is an edge-triggered event. If the switch is closed when an alarm is activated, no reset will occur. The input must be opened and then closed to reset the alarm. After the alarm is cleared, the drive will be enabled when the input is opened again.

AI2: For normal operation, the EN input must be closed (active, low). Alarm reset occurs when the input is opened (inactive high). This is an edge-triggered event. If the switch is open when an alarm is activated, no reset will occur. The input must be closed and then opened to reset the alarm. After the alarm is cleared, the drive will be enabled when the input is closed again.




AI3: The EN input is not used for Alarm Reset and may be used as a general purpose input. AI will be automatically set if CM is set to 13, 14, 17, or 18, or if SI is set to either 1 or 2 after the AI command is set.

**For Variable I/O Drives:**

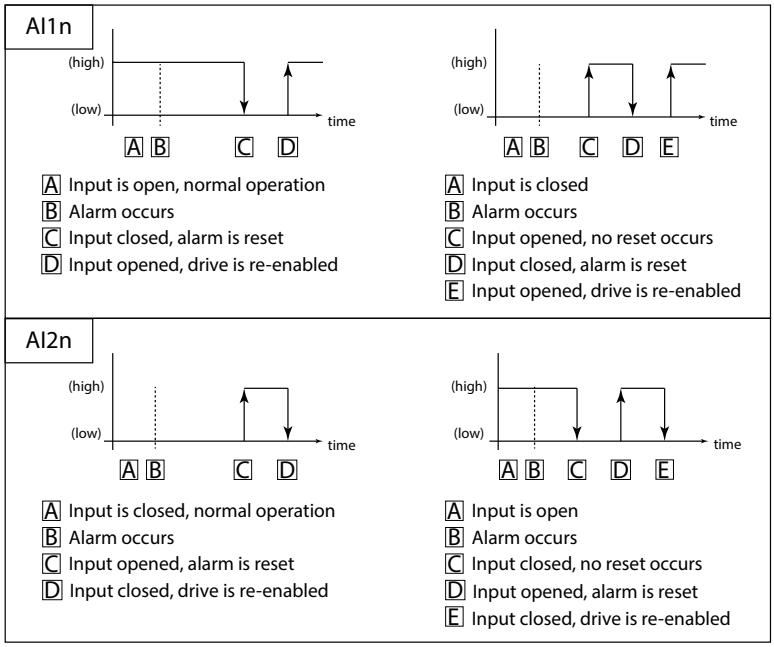
Drives with Variable I/O allow a second parameter which allows the user to specify the I/O point used as the Alarm Reset input. Before an I/O point can be used as an Alarm Reset input, it must be configured as an input with the SD command. See the SureStep User Manual for details of which inputs may be used as the Alarm Reset input.

There are three Alarm Reset Input states that can be defined with the AI command ("n" denotes the I/O point to be used):

AI1n: For normal operation, the designated input "n" must be open (inactive, high). Alarm reset occurs when the input is closed (active, low). This is an edge-triggered event. If the switch is closed when an alarm is activated, no reset will occur. The input must be opened (inactive, high) and then closed to reset the alarm. The drive will be enabled when the input is returned to the opened state (inactive, high), unless the SI command has been used to configure hardware enable functionality.

AI2n: For nomal operation, the designated input "n" must be closed (active, low). Alarm reset occurs when the input is opened (de-energized). This is an edge-triggered event. If the switch is open when an alarm is activated, no reset will occur. The input must be closed (energized) and then opened to reset the alarm. The drive will be enabled when the input is returned to the closed state (active, low), unless the SI command has been used to configure hardware enable functionality.

AI3n: The designated input "n" is not used for Alarm Reset and may be used as a general purpose input.




*NOTE: A rule of thumb when using the Alarm Reset function is to toggle the designated input twice whenever an alarm occurs. If the input is normally open, it should be closed and then opened again. If the input is normally closed, it should be opened and then closed again.*

Affects: Alarm Reset input range

See also: AL, CM, DL, SD, and SI commands

***COMMAND STRUCTURE:***

AI{Parameter #1}

AI{Parameter#1}{Parameter #2} (for Variable I/O only)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Input Usage |
| ***– units*** | Integer code |
| ***– range*** | 1, 2, or 3 |
| ***Parameter #2*** | I/O Point (if applicable)* |
| ***– units*** | Integer Code |
| ***– range*** | 2 or 4 (see SureStep User Manual for details) |

*Note: For drives equipped with Variable I/O, the SD command must be executed to set an I/O point as an input before it can be used as the Alarm Reset Input.*
**Parameter #2 only applies to drives equipped with Variable I/O. Parameter #2 is not defined for drives equipped with standard I/O.*

***EXAMPLES:***

- Standard Input Drives

| Command | Drive sends | Notes |
|---|---|---|
| AI1 | – | Enables input to reset alarm when closed (active, low) |
| AI2 | – | Enables input to reset alarm when opened (inactive, high) |
| AI3 | – | Enables input as general purpose input |
| AI | AI=1 | |

- Drives with Variable I/O

| Command | Drive sends | Notes |
|---|---|---|
| SD4I | – | Configures I/O 4 as input (see SD command for details) |
| AI14 | – | Assigns input 4 to reset the alarm when closed (active, low) |
| AI24 | – | Assigns input 4 to reset the alarm when opened (de-energized) |
| AI34 | – | Enables I/O 4 as a general purpose input |
| AI | AI=14 | |

*NOTE: When working with digital inputs and outputs it is important to remember the designations "low" and "high". If current is flowing into or out of an input or output (the circuit is energized), the logic state for that input/output is defined as "low" or closed. If no current is flowing (circuit is de-energized), or the input/output is not connected, the logic state is "high" or open. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 low", and SO1L means "set output 1 low." A high state is represented by the "H" character.*

## AL – Alarm Code

Reads back an equivalent hexadecimal value of the Alarm Code's 16-bit binary word. This command is useful for viewing over the serial port any alarms present at the drive.

See also: AR command, Appendix B

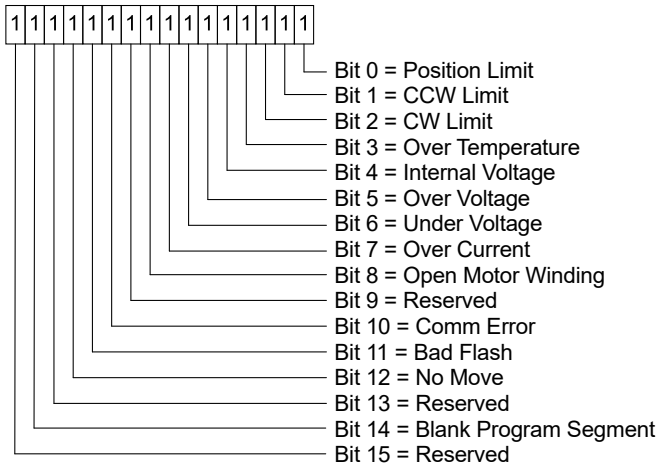
***COMMAND STRUCTURE:***

AL

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***Units*** | Hexadecimal value of 16-bit binary word (see below) |

| *Alarm Description* | *Bit #* | *Hex Value* |
|---|---|---|
| * Position Limit | 0 | 0001 |
| CCW Limit | 1 | 0002 |
| CW Limit | 2 | 0004 |
| * Over Temperature | 3 | 0008 |
| * Internal Voltage | 4 | 0010 |
| * Over Voltage | 5 | 0020 |
| Under Voltage | 6 | 0040 |
| * Over Current | 7 | 0080 |
| * Open Motor Winding | 8 | 0100 |
| Reserved | 9 | 0200 |
| Comm Error | 10 | 0400 |
| Bad Flash | 11 | 0800 |
| No Move | 12 | 1000 |
| Reserved | 13 | 2000 |
| Blank Program Segment | 14 | 4000 |
| Reserved | 15 | 8000 |




** The only alarm conditions that are categorized as "faults" are listed below.*
*(These are the only alarm conditions that set the drive Status Code (SC command) "Fault" bit #2 (Hex 0004).)*

- Position Limit (bit #0),
- Over Temperature (bit #3),
- Excess Regen / Internal Voltage (bit #4),
- Over Voltage (bit #5),
- Over Current (bit #7),
- Open Motor Winding (bit #8).

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AL | AL=0000 | No alarms (0000000000000000) |
| AL | AL=0002 | CCW end-of-travel Limit alarm (0000000000000010) |

## AM – Max Acceleration

Sets or requests the maximum acceleration/deceleration allowed when using analog velocity (oscillator) mode. Also sets the deceleration used when an End-of-Travel Limit is activated during any of the "Feed" moves, or when an ST (Stop) or SK (Stop & Kill) command is sent.

Affects: SK, SM, ST commands; Analog velocity (oscillator mode)

See also: VM command

***COMMAND STRUCTURE:***

AM{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Maximum Acceleration/Deceleration |
| ***– units*** | rev/sec/sec (rps/s) |
| ***– range*** | 0.167 to 5461.167 (resolution is 0.167 rps/s) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AM2000 | – | Sets maximum acceleration/deceleration rates to 2000 rev/sec/sec |
| AM | AM=2000 | |

## AO – Alarm Output

**For Standard Input Drives:**

Defines the drive's digital output as an Alarm Output. The output of a drive can be assigned to one of five functions: Alarm Output, Brake Output, Motion Output, Tach Output, or General Purpose Output. Each of these functions must exclusively use the output, so only one function is allowed. To set the output as an Alarm Output, use the AO command and one of the codes below.

There are three Alarm Output states that can be defined with the AO command:

AO1: Output is closed (active, low) when a Drive Fault is present.

AO2: Output is open (inactive, high) when a Drive Fault is present.

AO3: Output is not used as an Alarm Output and can be used for another automatic output function or as a general purpose output.

**For Variable I/O Drives:**

Drives with Variable I/O allow a second parameter which allows the user to specify the I/O point used as the Alarm Output. Before an I/O point can be used as an Alarm Output, it must be configured as an Output with the SD command.

There are three Alarm Output states that can be defined with the AO command ("n" denotes the I/O point to be used):

AO1n: Designated output "n" is closed (active, low) when a Drive Fault is present.

AO2n: Designated output "n" is open (inactive, high) when a Drive Fault is present.

AO3n: Designated output "n" is not used as an Alarm Output and can be used for another automatic output function or as a general purpose output.

*NOTE: Setting the AO command to 1 or 2 overrides previous assignments of this output's function. Similarly, if you use the BO or MO command to set the function of the output after setting the AO command to 1 or 2, usage of the output will be reassigned and AO will be automatically set to 3.*

Affects: Alarm Output usage

See also: AI, BO, MO, SD, SI commands

### *Command Structure:*

AO{Parameter #1}

AO{Parameter#1}{Parameter #2} (for Variable I/O only)

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Output Usage |
| ***– units*** | Integer code |
| ***– range*** | 1, 2, or 3 |
| ***Parameter #1*** | I/O Point (if applicable)* |
| ***– units*** | Integer code |
| ***– range*** | 1-4 |

*Note: For drives equipped with Variable I/O, the SD command must be executed to set an I/O point as an input or output before it can be used as the Alarm Output.*
**Parameter #2 only applies to drives equipped with Variable I/O. Parameter #2 is not defined for drives equipped with standard I/O.*

### *Examples:*

- Standard Output Drives

| Command | Drive sends | Notes |
|---|---|---|
| AO1 | – | Alarm Output will close when a Drive Fault occurs |
| AO2 | – | Alarm Output will open when a Drive Fault occurs |
| AO | AO=1 | |

- Drives with Variable I/O

| Command | Drive sends | Notes |
|---|---|---|
| SD4O | – | Configures I/O 4 as ouput (see SD command for details) |
| AO14 | – | Alarm Output is mapped to I/O 4 and will close for Drive Fault |
| AO24 | – | Alarm Output is mapped to I/O 4 and will open for Drive Fault |
| AO | AO=14 | |

*NOTE: When working with digital inputs and outputs it is important to remember the designations "low" and "high". If current is flowing into or out of an input or output (the circuit is energized), the logic state for that input/output is defined as "low" or closed. If no current is flowing (circuit is de-energized), or the input/output is not connected, the logic state is "high" or open. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 low", and SO1L means "set output 1 low." A high state is represented by the "H" character.*

## AP – Analog Position Gain

Sets or requests the analog input gain that relates to motor position when the drive is in analog position command mode (see CM command, parameter value 22). Gain value sets the commanded position when the analog input is at the configured full scale value. AP sets the distance or position, in steps, at the maximum extent of the analog input. The steps per rev setting is a global setting either set in SM-PRO or by using the EG command. SureMotion Pro can be used to configure the analog inputs for the desired input type, scaling, and offsetting.

Affects: CM22 (Analog Positioning Command Mode)

See also: AD, AF, AZ, CM, and SF commands

***COMMAND STRUCTURE:***

AP{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog position gain value |
| ***– units*** | Encoder counts |
| ***– range*** | 0 to 32767 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AP8000 | – | Position range over full scale analog input is 8000 steps |
| EG20000 | – | Sets steps/rev to 20,000 |
| SF10 | – | Sets step filter to 10 |
| AF1000 | – | Sets analog filter to 1000 |
| AP20000 | – | Sets full range of the analog input to 20,000 steps |

## AR – Alarm Reset

Resets the alarm and clears the fault (if faulted). If fault or alarm condition still persists the alarm is not cleared.

*NOTE: AR does NOT re-enable drive. Use Motor Enable (ME) command to re-enable drive.*

Affects: Alarm Code

See also: AL, ME, and MD commands

***COMMAND STRUCTURE:***

AR

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AR | – | Alarm code is cleared (if possible) |

## AT – Analog Threshold

Sets or requests the analog input threshold at the AIN input that is used by the Feed to Sensor (FS) command. The threshold value sets the analog voltage that determines a sensor state or a trigger value.

Affects: All "Feed to Sensor" type commands

See also: FM, FS, and FY commands

***COMMAND STRUCTURE:***

AT{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog threshold value |
| ***– units*** | volts |
| ***– range*** | -5.000 to 5.000 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AT5 | – | Analog input threshold set to 5 volts |
| AT | AT=5 | |

## AV – Analog Offset Value

Sets or requests the analog offset value of the analog input. The analog offset value can be set manually using the AV command, or automatically using the AZ command, which automatically sets the offset value to the current analog input value.

Affects: Analog input

See also: AZ, CM commands

***COMMAND STRUCTURE:***

AV{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog offset value |
| ***– units*** | volts |
| ***– range*** | -5.000 to 5.000 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AV0.25 | – | Set analog offset to 0.25 Volts |
| AV | AV=0.25 | |

## AZ – Analog Zero

Activates the analog input auto offset algorithm. This algorithm can also be accessed in SureMotion Pro, in the Advanced Settings button of the Velocity Control Mode dialog. It is useful in defining the current voltage present at the analog input as the zero point, or offset. AZ directly affects the AV command, which can be used to manually adjust the analog input offset value.

Affects: Analog input

See also: AV command

***COMMAND STRUCTURE:***

AZ

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AZ | – | Start analog auto offset algorithm |

*Example:* Apply 1 VDC across the AIN (+) and GND (-) terminals of the drive. Then send the AZ command to the drive. Next apply 4 VDC across the AIN and GND terminals. Send the IA command and the response should be very close to IA=3.00.

## BD – Brake Disengage Delay

This command only takes effect if the BO command is set to 1 or 2. After a drive is enabled, this is the time value that may delay a move waiting for the brake to disengage. When beginning a move the delay value must expire before a move can take place. The delay timer begins counting down immediately after the drive is enabled and the brake output is set. The BD command sets a time in milliseconds that a move may be delayed.

Affects: All "F" (Feed) and Jog commands

See also: BE and BO commands

***COMMAND STRUCTURE:***

BD{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Delay time |
| ***– units*** | Seconds |
| ***– range*** | 0 to 32.767 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| BD0.2 | – | Sets brake disengage delay to 200ms |
| BD | BD=0.2 | |

## BE – Brake Engage Delay

This command only takes effect if the BO command is set to 1 or 2. After a drive is commanded to be disabled, this is the time value that delays the actual disabling of the driver output. When using the dedicated brake output (see BO command) the output is activated immediately with the disable command, then the drive waits the delay time before turning off the motor current.

Affects: All "F" (Feed) and Jog commands

See also: BD and BO commands

***COMMAND STRUCTURE:***

BE{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Delay time |
| ***– units*** | Seconds |
| ***– range*** | 0 to 32.767 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| BE0.25 | – | Sets brake engage delay to 250ms |
| BE | BE=0.25 | |

## BO – Brake Output

*NOTE: The digital output circuits available on SureStep drives are not sized for directly driving a typical holding brake. An external relay must be wired in circuit between the digital output of the drive and the holding brake. See the SureStep User Manual for an example wiring diagram.*

**For Standard Input Drives:**

Defines the drive's digital output as Brake Output. The output of a drive can be assigned to one of five functions: Alarm Output, Brake Output, Motion Output, Tach Output, or General Purpose Output. Each of these functions must exclusively use the output, so only one function is allowed. There are two ways to define the function of this output: via SureMotion Pro or via SCL commands. To set the output as a Brake Output, use the BO command and one of the codes below:

BO1: Output is closed (active, low) when the drive is enabled, and open when the drive is disabled.

BO2: Output is open (inactive, high) when the drive is enabled, and closed when the drive is disabled.

BO3: Output is not used as an Brake Output and can be used for another automatic output function or as a general purpose output.

**For Variable I/O Drives:**

Drives with Variable I/O allow a second parameter which allows the user to specify the I/O point used as the Brake Output. Before an I/O point can be used as a Brake Output, it must be configured as an output with the SD command.

To set the output as a Brake Output, use the BO command and one of the codes below:

BO1n: Designated output "n" is closed (active, low) when the drive is enabled and open when the drive is disabled.

BO2n: Designated output "n" is open (inactive, high) when the drive is enabled and closed when the drive is disabled.

BO3n: Designated output "n" is not used as an Alarm Output and can be used for another automatic output function or as a general purpose output.

*NOTE: Setting the BO command to 1 or 2 overrides previous assignments of this output's function. Similarly, if you use the AO or MO command to set the function of the output after setting the BO command to 1 or 2, usage of the output will be reassigned and BO will be automatically set to 3.*

Affects: Function of digital output

See also: AI, AO, BD, ME, MD, MO, SD, SI commands

***COMMAND STRUCTURE:***

BO{Parameter #1}

BO{Parameter#1}{Parameter #2} (for Variable I/O only)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Output Usage |
| ***– units*** | Integer code |
| ***– range*** | 1, 2, or 3 |
| ***Parameter #1*** | I/O Point (if applicable)* |
| ***– units*** | Integer code |
| ***– range*** | 1-4 |

*Note: For drives equipped with Variable I/O, the SD command must be executed to set an I/O point as an output before it can be used as the Brake Output.*
**Parameter #2 only applies to drives equipped with Variable I/O. Parameter #2 is not defined for drives equipped with standard I/O.*

***EXAMPLES:***

- Standard Output Drives

| Command | Drive sends | Notes |
|---|---|---|
| BO1 | – | Brake Output will be closed when drive is enabled |
| BO | BO=1 | |

- Drives with Variable I/O

| Command | Drive sends | Notes |
|---|---|---|
| SD4O | – | Configures I/O 4 as ouput (see SD command for details) |
| BO14 | – | Brake Output is mapped to I/O 4 and will be closed when drive is enabled |
| BO | BO=14 | |

## BR – Baud Rate

Sets or requests the bit rate (baud) for serial communications. At power up a drive will send its power-up packet at 9600 baud. If a response from a host system (such as a software application) is not detected after 1 second and the drive is configured for SCL operation (see PM command), the drive will set the baud rate according to the value stored in the Baud Rate NV parameter. A Host system can set the baud rate at any time using this command. See Appendix B, "Host Serial

*NOTE: Setting the value takes effect immediately. Drive will immediately be disconnected from the host controller. Communications will need to be re-established with new baud rate.*

Affects: Serial communications

See also: TD, PB, PM, and PR commands

***COMMAND STRUCTURE:***

BR{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Baud rate |
| ***– units*** | Integer code |
| ***– range*** | 1 = 9600bps<br>2 = 19200bps<br>3 = 38400bps<br>4 = 57600bps<br>5 = 115200bps |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| BR2 | – | Baud rate is immediately set to 19200 |
| BR | BR=2 | |

## BS – Buffer Status

Requests from the drive the number of available command locations in the command buffer. This technique simplifies sending commands by eliminating the need to calculate if there is enough space in the buffer for additional commands. If the drive responds with at least a "1", a command can be sent.

If a drive responds to the BS command with "63", the buffer is empty. If a "0" is returned the buffer is full and no more buffered commands can be accepted (a buffer overflow will occur if another command is sent).

See also: CT and PS commands

***COMMAND STRUCTURE:***

BS

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | Empty command spaces in buffer |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| BS | BS=20 | There is room in the buffer for 20 more commands |

## CA – Change Acceleration Current

Sets or requests the accel/decel current setting ("peak of sine") of the stepper drive, also known as the peak current. CA will only accept parameter values equal to or larger than the current CC setting. CA is only applicable to STP-MTRD integrated motors.

Affects: Motor accel/decel current and torque

See also: CC, PA, and PC commands

*NOTE: CA has no effect in Command Mode 7 (CM7 - Step and Direction mode).*

***COMMAND STRUCTURE:***

CA{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Accel/Decel Current |
| ***– units*** | amps (resolution is 0.01 amps) |
| ***– range*** | STP-MTRD-17 series: 0 - 2.0<br>STP-MTRD-23 series: 0 - 5.0 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CA1.75 | – | Set accel/decel current to 1.75 amps (peak of sine) |
| CA | CA=1.75 | |

## CC – Change Current

Sets or requests the current setting ("peak of sine") of the stepper drive, also known as the running current. The range of the CC command may be limited from the ranges shown in the table below based on the settings defined in the SureMotion Pro software. Use SureMotion Pro to select a motor and set the maximum current setting. Note that setting CC automatically sets CI to 50% of CC. If a CI value different than 50% of CC is needed be sure to set CI after setting CC.

Affects: Motor current and torque

See also: CA, CI, CP, and PC commands

***COMMAND STRUCTURE:***

CC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Running Current |
| ***– units*** | amps (resolution is 0.01 amps) |
| ***– range*** | 0 - 5.0 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CC3 | – | Set running current to 3.0 amps |
| CI | CI=1.5 | CI automatically set to 1.5 amps along with CC3 command |
| CI1 | – | Set idle current to 1.0 amps |

## CD – Idle Current Delay Time

Sets or requests the amount of time the drive will delay before transitioning from full current (CC) to idle current (CI). This transition is made after a step motor takes the final step of a move. Operating in any form of pulse and direction mode the drive will reset the idle current delay timer each time a step pulse is received by the drive.

Affects: Motor current at rest

See also: CC and CI commands

***COMMAND STRUCTURE:***

CD{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Delay time |
| ***– units*** | Seconds |
| ***– range*** | 0.00 to 10.00 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CD0.4 | – | Idle current delay time set to 0.4 seconds |
| CD | CD=0.4 | |

## CE – Communication Error

Reads back the communication error code. This can be read back when the status code indicates a communication error is present. The value sent from the drive is the hexadecimal equivalent of the binary code. Bit assignments are shown in the Details table below.

***COMMAND STRUCTURE:***

CE

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Communication error code |
| ***– units*** | hex code |
| ***– range*** | bit 0 = not used<br>bit 1 = framing error<br>bit 2 = noise error<br>bit 3 = overrun error (too many characters in Rx buffer*)<br>bit 4 = Rx buffer full<br>bit 5 = Tx buffer full<br><br>*Rx buffer can hold up to 63 commands |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CE | CE=10 | Rx buffer full |

## CF – Anti-resonance Filter Frequency

Sets or requests the anti-resonance filter frequency setting. This setting is in Hz and works in conjunction with the anti-resonance setting (CG) to cancel instabilities due to mid-band resonance.

*NOTE: We strongly suggest using the SureMotion Pro software to set this value by entering as accurate a load inertia value as possible in the motor settings window.*

Affects: Mid-range performance of step motors

See also: CG command

***COMMAND STRUCTURE:***

CF{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Filter frequency |
| ***– units*** | Hz |
| ***– range*** | 1 - 2000 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CF1400 | – | Set anti-resonance filter frequency to 1400Hz |
| CF | CF=1400 | |

## CG – Anti-resonance Filter Gain

Sets or requests the anti-resonance filter gain setting. This setting is unit-less and works in conjunction with the anti-resonance filter frequency setting (CF) to cancel instabilities due to mid-band resonance.

*NOTE: We strongly suggest using the SureMotion Pro software to set this value by entering as accurate a load inertia value as possible in the motor settings window.*

Affects: Mid-range performance of step motors

See also: CF command

***COMMAND STRUCTURE:***

CG{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Filter gain |
| ***– units*** | Integer number |
| ***– range*** | 0 - 32767 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CG800 | – | Set anti-resonance filter gain to 800 |
| CG | CG=800 | |

## CI – Change Idle Current

Idle current is the level of current supplied to each motor phase when the motor is not moving. Using an idle current level lower than the running motor current level (see CC command) aids in motor cooling. A common level used for the idle current setting is 50% of the running current. After a motor move, there is a time delay after the motor takes its last step before the reduction to the idle current level takes place. This delay is set by the CD command.

CI cannot be greater than 90% of CC. If you attempt to set CI to a higher value than this, CI is automatically limited to 90% of CC. Furthermore, setting CC automatically sets CI to 50% of the CC value. If a CI value different than 50% of CC is needed, make sure to set CI after setting CC.

Affects: Motor current at standstill, holding torque

See also: CC, CD, and PI commands

***COMMAND STRUCTURE:***

CI{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Idle current |
| ***– units*** | amps |
| ***– range*** | 0 - 90% of running current |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CI2 | – | Set idle current to 2 amps |
| CC2 | – | Set running current to 2 amps |
| CI | CI=1 | |
| CI1.8 | – | Set idle current to 1.8 amps, or 90% of last CC value |

## CJ – Commence Jogging

Starts the motor jogging. The motor accelerates up to the jog speed (JS) at a rate defined by the jog accel (JA) command, then runs continuously until stopped. To stop jogging, use the Stop Jogging (SJ) command for a controlled decel rate (decel rate set by JL command). For a faster stop, use the ST command (decel rate set by AM command), but beware that if the speed or load inertia is high, the drive may miss steps or fault. The jogging direction is set by the last DI command. Use the CS command to change jog speed and direction while jogging. CS does not affect JS.

See also: CS, DI, JA, JL, JS, SJ, and ST commands.

***COMMAND STRUCTURE:***

CJ

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JS5 | – | Set jog speed to 5 rps |
| CJ | – | Initiate jogging at 5 rps |
| CS10 | – | Change jog speed to 10 rps |
| SJ | – | Stop jogging |

## CM – Command Mode

Sets or requests the Command Mode that the drive operates in. For more automated setup of command modes use *SureMotion Pro* software. The most common command mode is Point-to-Point (21), in which all SCL commands can be executed. Move commands (like FL, FP, FS, and CJ) can still be executed when the command mode is set to Step & Direction (7), because the drive will temporarily switch to command mode 21 to execute the move, then revert back to command mode 7. However move commands are either ignored or do not function properly when the command mode is set to any velocity mode (11-18) or the Analog Position mode (22).

*WARNING: CHANGING THE COMMAND MODE WITHOUT PROPER CARE MAY CAUSE THE MOTOR TO SPIN AT A HIGH RATE OF SPEED OR GIVE OTHER UNEXPECTED RESULTS. A POWER CYCLE SHOULD BE PERFORMED AFTER CHANGING THE COMMAND MODE.*

### COMMAND STRUCTURE:

CM{Parameter #1}

### DETAILS:

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Control mode |
| ***– units*** | integer code |
| ***– range*** | 7 – Step & Direction (Pulse & Direction)<br>11 – Analog velocity<br>12 – Analog velocity (Inputs: STEP=run/stop, DIR=direction change)<br>13 – Analog velocity (Inputs: DIR=direction change, EN=speed change)<br>14 – Analog velocity (Inputs: STEP=run/stop, DIR=direction change, EN=speed change)<br>15 – Constant Velocity<br>16 – Constant Velocity (Inputs: STEP=run/stop, DIR=direction change)<br>17 – Constant Velocity (Inputs: DIR=direction change, EN=speed change)<br>18 – Constant Velocity (Inputs: STEP=run/stop, DIR=direction change, EN=speed change)<br>21 – Point-to-Point (SCL Mode)<br>22 – Analog Position<br><br>Note: Refer to the following pages for more detailed information about the Command Mode settings. |

### EXAMPLES:

| Command | Drive sends | Notes |
|---|---|---|
| CM7 | – | Sets the drive to Step/Pulse & Direction mode |
| CM | CM=7 | |

*(CM – Command Mode continued next page.)*

## *CM – Command Mode (continued)*

### *Detail explanations of Command Mode selections:*

7 – Step & Direction (aka Pulse & Direction mode):

Step & Direction mode configuration does not require accel, decel and velocity settings. The behavior of the drive and motor is a function of the "EG" parameter, which determines the ratio of output pulses to input pulses. The SCL filter command "SF" can smooth the operation of the motor at low speed by introducing "micro-step emulation." The benefit of this filter decreases as the "EG" value increases.

- Pulse and Direction – Accepts a high speed pulse train output signal (from a PLC, motion controller, etc.). With this mode the frequency of the pulses fed into one input determines the speed; the direction of rotation is determined by a signal fed into another input. You can configure whether an ON or OFF signal represents clockwise motion.
- CW and CCW Pulse – The motor will move CW or CCW depending on which input the pulse is fed into. The drive has two inputs allocated to this feature; pulses fed into one input will generate CW motion, and pulses fed into the other input will generate CCW motion.
- A/B Quadrature – Sometimes called "Encoder Follower Mode" or "Slave Mode." The motor will move according to signals that are fed to the drive from a master encoder. This encoder can be mounted on a shaft on the machine or it can be another motor in the system. Using quadrature input mode it is possible for a number of motors to be "daisy chained" together with the encoder output signal from each drive being fed into the next.

11 – Analog velocity (Inputs: DIR= direction change):

Analog velocity mode has the speed controlled by an analog input signal on terminal "AIN." The drive will turn the motor at a velocity determined by the analog input and the "AG" command.

- The "DIR" (X2) input is used for forward/reverse control.

12 – Analog velocity (Inputs: STEP= run/stop, DIR= direction change, EN= speed change):

Analog velocity mode has the speed controlled by an analog input signal on terminal "AIN." The drive will turn the motor at a velocity determined by the analog input and the "AG" command.

- The "STEP" (X1) input is used for run/stop control.
- The "DIR" (X2) input is used for forward/reverse control.

13 – Analog velocity (Inputs: DIR= direction change, EN= speed change):

Analog velocity mode has the speed controlled by an analog input signal on terminal "AIN." The drive will turn the motor at a velocity determined by the analog input and the "AG" command.

- The "DIR" (X2) input is used for forward/reverse control.
- The "EN" (X3) input is used for setting the RPMs at a constant velocity (JC Command).

14 – Analog velocity (Inputs: STEP= run/stop, DIR= direction change, EN= speed change):

Analog velocity mode has the speed controlled by an analog input signal on terminal "AIN." The drive will turn the motor at a velocity determined by the analog input and the "AG" command.

- The "STEP" (X1) input is used for run/stop control.
- The "DIR" (X2) input is used for forward/reverse control.
- The "EN" (X3) input is used for setting the RPMs at a constant velocity (JC Command).

15 – Velocity (Inputs: DIR= direction change):

Motor will run at a constant velocity according to the JS command.

- The "DIR" (X2) input is used for forward/reverse control.

16 – Velocity (Inputs: STEP= run/stop, DIR= direction change):

Motor will run at a constant velocity according to the JS command.

- The "STEP" (X1) input is used for run/stop control. The "DIR" (X2) input is used for forward/reverse control.

17 – Velocity (Inputs: DIR= direction change, EN= speed change):

Motor will run at a constant velocity according to the JS command.

- The "DIR" (X2) input is used for forward/reverse control.
- The "EN" (X3) input is used for setting the RPMs at a constant velocity (JC Command).

18 – Velocity (Inputs: STEP= run/stop, DIR= direction change, EN= speed change):

Motor will run at a constant velocity according to the JS command.

- The "STEP" (X1) input is used for run/stop control.
- The "DIR" (X2) input is used for forward/reverse control.
- The "EN" (X3) input is used for setting the RPMs at a constant velocity (JC Command).

21 – Point-to-Point mode (SCL Mode):

Can also be referred to as position mode and is recommended for streaming command applications. SCL mode 21 provides for configuration of the SureStep drive to accomplish trapezoidal moves that would normally require a motion controller or programmable pulse output device such as the high speed outputs available in the Productivity PLC or BRX PLC. Using SCL commands, the SureStep drive can be configured to accomplish complex motion profiles. Either relative or absolute moves can be commanded utilizing this mode.

22 – Analog Position:

Sets the drive to Analog Positioning mode. In this mode it is also possible to control position through the use of an external encoder.

The analog signal going into the drive is going through an analog to digital converter with a range of 0-32,760. The 0-5V is a direct correlation to the position previously mentioned. The EG command will adjust the steps per rev. Depending on the steps per rev, the AF command will need to be played with a bit to get smooth motion. There is a simple example under the AP command for achieving smooth motion when bench testing. The application and load will dictate a lot of what kind of settings you will need. There is no drift or accumulated error using CM22.

Relatively long moves can often be somewhat rough, just due to the fact that a small change in the analog input would equate to a large change in rotor position. You can use AD and AF to set up a deadband and filter on the analog input to quiet some of the noise.

## CS – Change Speed

Sets or requests the jog speed in rev/sec while jogging. When Jogging using the CJ command the jog speed can be changed dynamically by using this command. The value of CS can be positive or negative allowing the direction of jogging to be changed also. Ramping between speeds while jogging is controlled by the JA and JL commands. Changing CS does not change either JS or DI.

Affects: Jog speed while jogging

See also: CJ and JS commands

***COMMAND STRUCTURE:***

CS{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Jog Speed |
| ***– units*** | rev/sec |
| ***– range*** | STP-DRV = -133.3333 to 133.3333<br>STP-MTRD = -80.0000 to 80.0000<br>Sign determines direction: "-" for CCW; no sign for CW. Resolution is 0.0042 for both. |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CS2.5 | – | Set jog speed to CW at 2.5 rev/sec |
| CS | CS=2.5 | Displays current Jog speed |
| CS-5 | – | Set jog speed to CCW at 5 rev/sec |

## CT – Continue

Resume execution of buffered commands after a PS command has been sent. The Pause (PS) command allows you to pause execution of commands in the command buffer. After sending the PS command, subsequent commands are buffered in the command buffer until either a CT command is sent, at which time the buffered commands will execute in the order they were received, or until the command buffer is full.

See also: PS, SK, and ST commands

***COMMAND STRUCTURE:***

CT

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| CT | – | Resumes execution of a paused command buffer |

## DA – Define Address

Sets or requests individual drive address character for multi-drop RS-485 communications. This command is not valid for single-axis (point-to-point) or RS-232 communications.

If a drive is assigned an address (1) then it will respond to commands prefixed by that address (for example: 1FL2000) but it will also respond to the same command without the address (example: FL2000). Commands without an address are known as global commands. The only difference is that the drive will not send an acknowledgement back to the host when given a global command. If you have 4 addressed drives on a network and you send FL2000 then all drives will respond to the command but none will send an acknowledgement.

All RS-485 queries require an address character. Commands that do not require a response can be sent globally. Example: you can send an AR, FL or MD to all axes at once by omitting the address char. But if you send a query like AL or SC, the drive will want to respond. Since you only want one drive responding to each query to avoid network collisions, for queries the address is required.

Affects: Drive address for multi-drop communications

***COMMAND STRUCTURE:***

DA{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | RS-485 network address |
| ***– units*** | character |
| ***– range*** | Valid address characters are:<br>! " # $ % & ' ( ) + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < > ? @ |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| DA1 | – | Set drive address to "1" |
| DA | DA=1 | |

## DC – Change Distance

Sets or requests the "change distance" or offset distance in steps. The change distance is used by various move commands to define more than one distance parameter. All move commands use the DI command at some level, and many require DC as well. Examples are FC, FM, FO, and FY. The moves executed by these commands change their behavior after the change distance (DC) has been traveled. For example, FM is similar to FS, but in an FM move the sensor input is ignored until the motor has moved the number of steps set by DC. This is useful for masking unwanted switch or sensor triggers. Since DI sets move direction (CW or CCW), the sign of DC is ignored.

Affects: FC, FM, FO, and FY commands

See also: VC command

***COMMAND STRUCTURE:***

DC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Distance |
| ***– units*** | steps |
| ***– range*** | 0 to 2,147,483,647 |
| ***– default*** | 0 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| DC80000 | – | Set change distance to 80000 counts |
| DC | DC=80000 | |
| | | |
| DI-100000 | – | Set overall distance to 100000 counts in CCW direction |
| DC50000 | – | Set change distance to 50000 counts |
| VE5 | – | Set velocity to 5 rev/sec |
| VC2 | – | Set change velocity to 2 rev/sec |
| FC | – | Initiate FC command |

## DE – Deceleration Rate

Sets or requests the deceleration rate used in all "F" (point-to-point) moves in rev/sec/sec.

Affects: FC, FL, FM, FP, FS, FY, SH commands

See also: AC, DE, DI, VE commands

### *Command Structure:*

DE{Parameter #1}

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Deceleration rate |
| ***– units*** | rev/sec/sec (rps/s) |
| ***– range*** | 0.167 to 5461.167 (resolution is 0.167 rps/s) |

### *Examples:*

| Command | Drive sends | Notes |
|---|---|---|
| DE125 | – | Set deceleration rate to 125 rev/sec/sec |
| DE | DE=125 | |
| | | |
| AC25 | – | Set Acceleration rate to 25 rev/sec/sec |
| DE10 | – | Set Deceleration rate to 10 rev/sec/sec |
| VE5 | – | Set Velocity to 5 rev/sec |
| FL200000 | – | Execute Feed to Length move of 200000 steps in CW direction |

## DI – Distance / Position

Sets or requests the move distance, in steps. The sign of DI indicates move direction: "-" for CCW, no sign for CW. DI sets both the distance for relative moves, like FL, and the position for absolute moves, like FP. DI also sets the direction of rotation for jogging (CJ).

Affects: All move commands

See also: AC, DC, DE and VE commands

***COMMAND STRUCTURE:***

DI{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Distance |
| ***– units*** | steps |
| ***– range*** | -2,147,483,647 to 2,147,483,647<br>sign determines direction: "-" for CCW; no sign for CW |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| DI20000 | – | Set distance to 20000 steps in the CW direction |
| DI | DI=20000 | |
| | | |
| DI-8000 | – | Set distance to 8000 steps in the CCW direction |
| FL | – | Initiate a Feed to Length (relative) move in the CCW direction |
| | | |
| SP0 | – | Set current motor position to absolute zero |
| DI20000 | – | Set position to 20000 steps CW |
| FP | – | Initiate absolute move to 20000 step position |
| DI10000 | – | Set position to 10000 steps CW |
| FP | – | Initiate absolute move to 10000 step position (motor will move CCW) |

## DL – Define Limits

Defines the STEP and DIR inputs as CW end-of-travel and CCW end-of-travel limit inputs, respectively. Both inputs are assigned together as end-of-travel limits, and for the same connection type (see definition of states below). If one of these inputs is activated while defined as an end-of-travel limit, motor rotation will stop in that direction, and an alarm code will show at the drive's status LEDs. The alarm code will be 2 Green + 2 Red for a CW end-of-travel, and 1 Green + 2 Red for a CCW end-of-travel.

The STEP and DIR inputs can each be assigned to only one function in an application. If you want to use the STEP and DIR inputs as end-of-travel limit inputs you can define them as such in two ways, with the SureMotion Pro software, or with the DL command. DL takes no effect if the drive is set in Command Mode (CM) 7, 14, or 18, because these modes predefine these inputs and take precedence over the DL command. Also, setting the JE command after setting the DL command reassigns the STEP and DIR inputs as jog inputs and turns off any limit input usage (DL3). In other words, the DL and JE commands, as well as Command Modes (CM) 7, 14, and 18 each assign a usage to the STEP and DIR inputs. Each of these must exclusively use the STEP and DIR inputs. Command Modes are most dominant and will continually prevent DL and JE from using the inputs. DL and JE exclude each other by overwriting the usage of the STEP and DIR inputs.

There are three end-of-travel limit input states that can be defined with the DL command:

- DL1: End-of-travel limit occurs when an input is closed (energized). Motor automatically decelerates using the AM command.
- DL2: End-of-travel limit occurs when an input is open (de-energized). Motor automatically decelerates using the AM command.
- DL3: The STEP and DIR inputs are not used as end-of-travel limit inputs and can be used as a general purpose inputs. DL will be automatically set to 3 if CM is set to 7, 14, or 18, or if JE is executed after the DL command is set.

Affects: All "F" commands, CJ, SH, WI (when jogging) commands

See also: AM command

### *Command Structure:*

DL{Parameter #1}

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Limit input state (see above) |
| ***– units*** | integer number |
| ***– range*** | 1, 2 or 3 |
| ***– default*** | 3 |

### *Examples:*

| Command | Drive sends | Notes |
|---|---|---|
| DL1 | – | Set limit inputs to act as normally open |
| DL | DL=1 | |
| DL3 | – | Set limit inputs to act as general purpose inputs |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## ED – Encoder Direction

Sets or requests the encoder count direction.

Affects: Encoder count direction

See also: EF and EI commands

***COMMAND STRUCTURE:***

ED{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Encoder Count Direction |
| ***– units*** | Binary flag (0 or 1) |
| ***– range*** | 0 = default behavior<br>1 = count in reverse |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| ED1 | – | Set encoder to count in reverse |
| ED | ED=1 | |

## EF – Encoder Function

Sets or requests the decimal equivalent of the encoder function's 3-bit word. The encoder function can be set through Configurator or by using the EF command. Only stepper drives with encoder inputs running a step motor with a shaft-mounted encoder can utilize the Stall Detection and Stall Prevention functions.

An encoder with differential outputs and a resolution of at least 1000 lines (4000 counts/rev) is recommended.

EF0: Disable Encoder Functionality

EF1: Turn Stall Detection ON.

EF2: Turn Stall Prevention ON.

EF6: Turn Stall Prevention with time-out ON.

The drive performs a full current probe for encoder alignment during power-up and after each EF command is sent. It is very important to raise the idle and continuous current settings to the maximum value and then execute the new EF setting after a 1 second delay. Once the EF command is completed, the current may be reset to its normal value.

Affects: Stall Detection and Stall Prevention

See also: CC, CI, ER, and PF commands

***COMMAND STRUCTURE:***

EF{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Encoder function setting |
| ***– units*** | Decimal equivalent of 3-bit binary word |
| ***– range*** | 0 = Encoder function off<br>1 = Stall Detection<br>2 = Stall Prevention<br>6 = Stall Prevention with time-out |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| EF1 | – | Turn ON Stall Detection function |
| EF | EF=1 | |
| EF6 | – | Enable Stall Prevention with time-out |
| EF | EF=6 | |

Example encoder alignment sequence (STP-MTRD-24)

| Command | Drive sends | Notes |
|---|---|---|
| CC6 | – | Raise current to 6A |
| CI5.4 | – | Raise idle current to 5.4A* |
| EF1 | – | Enable Stall Detection feature |
| CC3 | – | Lower current to normal running level (application dependent) |
| CI2.4 | – | Lower idle current to normal running level (application dependent) |

*90% of CC; see CI command for details

## EG - Electronic Gearing

Sets or requests the desired step resolution of the step motor. Units of the EG command, steps/rev, are based on 2-phase, 1.8 degree step motors, which provide 200 full steps per revolution.

*NOTE: Step resolution is always set in SureMotion Pro during startup and configuration of the drive. If no further adjustment to the step resolution is required the EG command is not needed.*

Affects: All Command modes

See also: CM, ER, FE, and HW commands

### *Command Structure:*

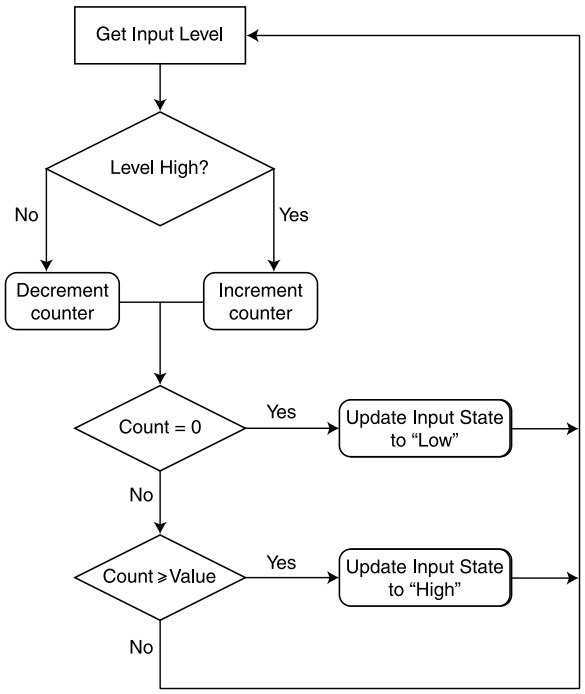
EG{Parameter #1}

### *Details:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Step resolution |
| ***– units*** | steps/rev |
| ***– range*** | 200 to 51,200 (in increments of 2) |

### *Examples:*

| Command | Drive sends | Notes |
|---|---|---|
| EG36000 | – | Set microstep resolution to 36000 steps/rev |
| EG | EG=36000 | |



## EI - Input Noise Filter

Sets or requests the Input Noise Filter parameter. This parameter acts as a low-pass filter, rejecting noise above the specified frequency.

For STP-MTRD-17 series drives, given a cutoff frequency an appropriate EI value may be calculated as follows (where "f" is the target cutoff frequency):

EI=9,000,000 / f

For all other drives, EI can be calculated as follows:

EI=15,000,000 / f

Affects: "Input Noise Filter" parameter

See also: CM, ER, FE, and HW commands

***COMMAND STRUCTURE:***

EI{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Encoder Noise Filter Constant |
| ***– units*** | n/a |
| ***– range*** | 0 - 255 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| EI128 | – | STP-MTRD-17 series, set encoder noise filter to 70.3kHz |
| EI128 | – | Non-STP-MTRD-17 series, set encoder noise filter to 117.2kHz |
| EI | EG=128 | |

## EP - Encoder Position

The EP command allows the host to define the present encoder position. For example, if the encoder is at 4500 counts, and you would like to refer to this position as 0, send EP0. To ensure that the internal position counter resets properly, use SP immediately following EP. For example, to set the position to zero after a homing routine, send EP0 then SP0.

Sending EP with no position parameter requests the present encoder position from the drive.

For best results when using stepper systems, we recommend setting both CC and CI to the motor's maximum ratings before issuing an EP command. This will avoid any position error caused by the motor's detent torque. Once EP has been changed, reset CC and Ci to their running levels.

Affects: Encoder position value

See also: SP command

***COMMAND STRUCTURE:***

EP{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Encoder position value |
| ***– units*** | Counts |
| ***– range*** | -2,147,483,647 to 2,147,483,647 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| EP0 | – | (Step 1) reset internal position counter |
| SP0 | – | (Step 2) reset internal position counter |

## FC - Feed to Length with Speed Change

Executes a Feed to Length (relative move) with a speed change. Overall move distance and direction come from the last DI command. Accel and decel are from AC and DE commands. Initial speed is VE. After the motor has moved DC counts, the speed is reduced to VC.

*NOTE: If DC is greater than DI minus the distance required to decelerate the motor (affected by DE and VE), no speed change will result.*

See also: DC and VC commands

***COMMAND STRUCTURE:***

FC

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| DI50000 | – | Set overall distance to 50000 steps |
| VE5 | – | Set initial velocity to 5 rev/sec |
| DC40000 | – | Set change distance to 40000 steps |
| VC0.5 | – | Set change velocity to 0.5 rev/sec |
| FC | – | Initiate move |

## FD - Feed to Double Sensor

Accelerates the motor at rate AC to speed VE. When the first sensor is reached (first input condition is made), the motor decelerates at rate DE to speed VC. When the second sensor is reached (second input condition is made), the motor decelerates over the distance DI to a stop at rate DE. The sign of the DI register is used to determine both the direction of the move (CW or CCW) and the distance past the second sensor. If DI is long, the motor may not begin decel immediately after the second sensor. If DI is short, the motor may decelerate using a faster decel rate than DE. Both analog and digital inputs can be used as sensor inputs.

Both sensor inputs must be from the same physical I/O connector of the drive. This means that both inputs used in this command must reside on the same I/O connector, either IN/OUT1 or IN/OUT2.

See also: FM, FS, FY, and VC commands. See AT command for using analog input as sensor input

***COMMAND STRUCTURE:***

FD{Parameter #1}{Parameter #2}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameters*** | See "Appendix C: Working with Inputs and Outputs" |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| FD2F4H | – | Launch Feed to Double Sensor move: decel from VE to VC when input 2 changes from high to low (falling), then decel to a stop when input 4 is high |
| AC50 | – | Set accel rate to 50 rev/sec/sec |
| DE50 | – | Set decel rate to 50 rev/sec/sec |
| DI-1 | – | Set move direction to CCW |
| VE5 | – | Set initial velocity to 5 rev/sec |
| VC1 | – | Set change velocity to 1 rev/sec |
| FD1F2H | – | Launch Feed to Double sensor move: decel from VE to VC when input 1 changes from high to low (falling), then decel to a stop when input 2 is high |

## FE - Follow Encoder

Puts drive in encoder following mode until the given digital or analog input condition is met. The master encoder channels A and B must be wired to the STEP/X1 and DIR/X2 inputs of the drive. Use the EG command before the FE command to set the following resolution. The Step Smoothing Filter is active in FE mode; see the SF command for details.

When the FE command is initiated, the acceleration rate AC is used to ramp the motor up to the following speed. (Doing this prevents extreme accelerations when the master encoder signal is already at its target velocity). The motor continues to follow the master encoder pulses until the input condition is met, at which time the motor decelerates at rate DE to a stop using the DI command as the overall decel distance. If DI is long, the motor may not begin to decel immediately after the input condition is met. If DI is short, the motor may have to decelerate at a rate faster than DE.

Before the input condition is met, the motor will follow the master encoder pulses in both CW and CCW directions, regardless of the sign of the DI command. However, once the input condition is met the motor will only stop properly if moving in the direction set by the DI command.

When done executing, the drive returns to the move it was in before executing the FE command.

*NOTE: You must use the SureMotion Pro software to set up the STEP/X1 and DIR/X2 inputs for encoder following. Do this by choosing A/B Quadrature in the Position mode settings.*

See also: EG and ST commands

***COMMAND STRUCTURE:***

FE{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameters*** | See "Appendix C: Working with Inputs and Outputs" |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC500 | – | Limit acceleration in encoder following to 500 rps/s |
| DI8000 | – | Set the stopping offset distance to 8000 counts |
| FE4L | – | Run in encoder following mode until input 4 is low |

## FI - Filter Input

Applies a digital filter to the given input. The digital input must be at the same level for the time period specified by the FI command before the input state is updated. For example, if the time value is set to 100 the input must remain high for 100 processor cycles before high is updated as the input state. One processor cycle is 100µsec. A value of "0" disables the filter. This command can be used to apply to filters to inputs STEP, DIR and EN. For drives with Variable I/O, the command can be used to apply filters to any input.

Affects: All commands using inputs

See also: SD, WI, and all feed to sensor commands.

***COMMAND STRUCTURE:***

FI{Parameter #1}{Parameter #2}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameters*** | See "Appendix C: Working with Inputs and Outputs" |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| FI4100 | – | Requires that input X4 maintain the same state (low or high) for 100 total processor cycles before the drive registers the change |
| FI4 | FI4=100 | |

***DIGITAL INPUT FILTERS IN DETAIL***

Drives have the capability to apply digital filters to selected digital inputs. With factory defaults, digital inputs are not filtered through any means other than the natural response time of the optical couplers used in the input circuits. Analog filtering has purposely not been implemented so as to not restrict the input circuit. However, digital filtering is available on select digital inputs to enhance the usage of those inputs.

On occasion, electrical noise at digital inputs may create a false trigger or even a double-trigger. This can often happen when using mechanical switches that "bounce" when activated or de-activated. For this reason, there may be a need to filter an input to eliminate the effects of these noise conditions. Digital filtering gives the greatest variability by allowing the user to select the amount of filtering required to eliminate the effects of noise or bounce.

The digital filters work by continuously monitoring the level of the inputs to which filters have been applied using the FI command. During each processor cycle (100µsec) internal counters associated with the filters are incremented or decremented depending on whether each input is high (open) or low (closed) respectively. When a command that accesses a digital input is executed, the state of the input requested by that command will be updated only after the internal counter for that input's filter reaches a threshold value. This threshold value, or filter value, is set by the FI command. The flow chart shows how a digital filter works.

For example, if we apply a digital filter of 2 milliseconds to input 3 on a STP-DRV-4850 stepper drive, it means we'd like the level of input 3 (low or high) to be true for a total of 2 milliseconds before the processor updates the state of the input to the state requested by the command currently being executed. If the command being executed is waiting for input 3 to be low, the processor will wait until the input has been low for a total of 2 milliseconds before updating the state of the input as low and finishing the command. If the input has already been at the low state for 2 or more milliseconds when the command is initiated, there will be no delay. If the input is in the high state when the command is initiated, there will be an additional 2 millisecond delay for the state change from high to low. It is important to understand that any fluctuation of the physical signal, by switch bounce or electrical noise, will contribute to a lag in the processed signal.

To turn filtering of input 3 on, we need to use the FI command. The FI command works in processor cycles, so a value of 1 equals 100 microseconds. To filter the input for 2 milliseconds the value of the FI command would be 2 msec divided by 100µsec, or 20. The correct syntax for the FI command would be "FI320".

The digital filter incorporates an averaging effect on the level of the input. In the example above, if the level of input 3 were fluctuating between low and high over a range of processor cycles (maybe due to electrical noise), the drive would not update the input state until the internal counter value went to zero (for a low state) or the filter value (for a high state). Another example of this averaging effect is if the input were connected to a pulse train from a signal generator with a duty cycle of 51% high and 49% low. The input state would eventually be set to a high state, depending on the time value used in the pulse train.

Filter values are non-volatile if followed by an SA command.

*NOTE: A side effect of the digital filter, which is true of any filter, is to cause a lag in the response to an input level. When an input changes state and is solid (no noise), the lag time will be the same as the filter value. When noise is present, the lag may be longer.*

## FL - Feed to Length

Executes a Feed to Length (relative move) command. Move distance and direction come from the last DI command. Speed, accel and decel are from VE, AC and DE commands respectively. Optional parameter allows using a local distance for the FL command rather than the last DI value.

*NOTE: The last DI command can be ignored/unaltered if a parameter is used with the FL command.*

See also: AC, DE, DI, VE commands

***COMMAND STRUCTURE:***

FL{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Relative distance |
| ***– units*** | steps |
| ***– range*** | -2,147,483,647 to 2,147,483,647<br>sign determines direction: "-" for CCW; no sign for CW |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC100 | – | Set accel rate to 100 rev/sec/sec |
| DE150 | – | Set decel rate to 150 rev/sec/sec |
| VE8 | – | Set velocity to 8 rev/sec |
| DI20000 | – | Set distance to 20000 steps in the CW direction |
| FL | – | Initiate Feed to Length move |
| FL10000 | – | Initiate Feed to Length move of 10000 steps in the CW direction without affecting the last DI command |
| FL-400 | – | Initiate Feed to Length move of 400 steps in the CCW direction without affecting the last DI command |

## FM - Feed to Sensor with Mask Distance

Executes a Feed to Sensor command, but sensor is ignored for the first DC counts of the move. Useful for masking a switch or clearing a part before sensing the correct stop input. DI sets the distance to move after the stop input is triggered. AC sets accel rate, VE sets velocity, and DE sets decel rate.

See also: DC and FS commands; see AT command for using AIN as sensor input

***COMMAND STRUCTURE:***

FM(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Input, Input condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = STEP; 2 = DIR; 3 = EN; 4 = AIN<br>letter: L = low; H = high; F = falling edge; R = rising edge |

***EXAMPLES:***

*Example*: Parts are feeding on a conveyor which is being driven by the step motor. A sensor detects the leading edge of the part and stops. If the part has a hole in it, which is common, when you attempt to feed the next part into position you may in fact stop after feeding the previous part only a short distance because the sensor will register the hole in the part rather than the leading edge of the next part. The solution is to use the FM command instead of the FS command, and to set the DC command for the size of the part (or greater).

Example continued: The parts on the conveyor are 6 inches long. Your mechanical linkage provides 2000 steps per inch. You want the leading edge of the part to stop moving 1 inch past the sensor, and therefore 5 inches of the part will not have gone past the sensor yet. To avoid holes in the part and see the next part properly, we need to mask 5 inches or more of the move. Here are the commands you could use.

| Command | Drive sends | Notes |
|---|---|---|
| DI2000 | – | Set distance to stop past sensor at 1 inch (2000 steps) |
| DC10200 | – | Set distance over which to ignore (mask) the sensor at 5.1 inches, enough to allow the previous part to completely clear the sensor |
| FM1F | – | Initiate FM move. Sensor is connected to input 1 and will close when it sees a part |

*NOTE: When working with inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

*NOTE: When working with the analog input of an STP-DRV drive (AIN terminal), "L" designates an analog value lower than the value set by the AT command. Similarly "H" designtates an analog value greater than the value set by the AT command.*

## FO - Feed to Length and Set Output

Same as Feed to Length (FL) but changes the state of an output during the move. Overall move distance is defined by the DI command. Accel rate, decel rate, and velocity are set by the AC, DE and VE commands, respectively. Distance within overall move at which output condition should be set is defined by the DC command.

*NOTE: Dedicated output functions - alarm output, brake output, motion output - must be configured as general purpose before the FO command can be used with the drive's output.*

See Also: DC command

***COMMAND STRUCTURE:***

FO(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Output, Output condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1<br>letter: L = low (closed); H = high (open) |

***EXAMPLES:***

*Example*: You're feeding parts to be cut to length. For maximum throughput, you want to trigger the cut-off knife as the part is nearing the final position.

| Command | Drive sends | Notes |
|---|---|---|
| AC100 | – | Set accel rate to 100 rev/sec/sec |
| DE100 | – | Set decel rate to 100 rev/sec/sec |
| VE2.5 | – | Set velocity to 2.5 rev/sec |
| DI20000 | – | Overall move distance set to 20000 steps |
| DC15000 | – | Set output distance set to 15000 steps |
| FO1L | – | Initiate move and set output low at 15000 steps |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## FP - Feed to Position

Executes a Feed to Position (absolute) move. Move position comes from the last DI command. Speed, accel and decel are from VE, AC and DE commands, respectively. Optional parameter allows using a local absolute position for the FP command rather than the last DI value.

*NOTE: The last DI command can be ignored/unaltered if a parameter is used with the FP command.*

See also: AC, DE, DI, VE commands

***COMMAND STRUCTURE:***

FP{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Absolute position |
| ***– units*** | steps |
| ***– range*** | -2,147,483,647 to 2,147,483,647 |

***EXAMPLES:***

*Example*: After homing the motor you want to zero the home position and move to an absolute position 80000 steps from the new home position.

| Command | Drive sends | Notes |
|---|---|---|
| SP0 | – | Set current motor position as absolute zero |
| VE5 | – | Set velocity to 5 rev/sec |
| DI80000 | – | Set move position to 80000 steps |
| FP | – | Initiate Feed to Position move |
| FP40000 | – | Initiate Feed to Position move to 40000 steps without affecting the last DI command |

## FS - Feed to Sensor

Executes a Feed to Sensor command. Requires input number and condition. The motor moves until a sensor triggers the stop input, then stops a precise distance beyond the sensor. The stop distance is defined by the DI command. The direction of rotation is defined by the sign of the DI command ("-" for CCW, no sign for CW). Speed, accel and decel are from the last VE, AC and DE commands, respectively.

A motor moving at a given speed, with a given decel rate, needs a certain distance to stop. If you specify too short a distance, the drive may overshoot the target. Use the following formula to compute the minimum decel distance, given a velocity V (in rev/sec) and decel rate D (in rev/sec/sec). R = steps/rev, which in this equation equals the value set by the EG command in steps/rev.

- minimum decel distance = $(V)^2(R) / 2(D)$

See also: FM and FY commands; see AT command for using AIN as sensor input

***COMMAND STRUCTURE:***

FS(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Input number, Input condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = STEP; 2 = DIR; 3 = EN; 4 = AIN<br>condition: L (low); H (high); F (falling edge); R (rising edge) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC100 | – | Set accel rate to 100 rev/sec/sec |
| DE100 | – | Set decel rate to 100 rev/sec/sec |
| DI1000 | – | Set stop distance to 1000 steps |
| VE0.5 | – | Set velocity to 0.5 rev/sec |
| FS1L | – | Initiate move and decel to stop when sensor tied to input 1 is low |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

*NOTE: When working with the analog input of an STP-DRV drive (AIN terminal), "L" designates an analog value lower than the value set by the AT command. Similarly "H" designtates an analog value greater than the value set by the AT command.*

## FY - Feed to Sensor with Safety Distance

Executes a Feed to Sensor move while monitoring a predefined safety distance DC. If sensor is not found before DC is reached the motor is stopped and the drive sends the host an exclamation point ("!"). DI defines the direction of rotation and the stop distance to move after the sensor triggers the stop input. Accel rate, decel rate, and velocity are set by the AC, DE, and VE commands, respectively. Note that the final motor position if the sensor is not found will be the safety distance plus the distance requied to decelerate the load, which is dependent on the decel rate DE. This command is useful for avoiding machine jams or detecting the end of a roll of labels. For example, you are feeding labels and you want to stop each label 2000 steps after the sensor detects the leading edge. The labels are 60,000 steps apart. Therefore, if you move the roll more than 60,000 steps without detecting a new label, you must be at the end of the roll.

See also: DC, DE, FM and FS commands; see AT command for using AIN as sensor input

***COMMAND STRUCTURE:***

FY(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Input number, Input condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = STEP; 2 = DIR; 3 = EN<br>condition: L (low); H (high); F (falling edge); R (rising edge) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC50 | – | Set accel rate to 50 rev/sec |
| DE50 | – | Set decel rate to 50 rev/sec |
| VE2.5 | – | Set velocity to 2.5 rev/sec |
| DI2000 | – | Set distance to stop beyond sensor to 2000 steps, and set move direction to CW |
| DC60000 | – | Set safety distance to 60000 steps |
| FY2L | – | Launch Feed to Sensor: motor will stop when input 2 is low or when 60000 steps are reached: whichever event comes first |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

*NOTE: When working with the analog input of an STP-DRV drive (AIN terminal), "L" designates an analog value lower than the value set by the AT command. Similarly "H" designtates an analog value greater than the value set by the AT command.*

## HW - Hand Wheel

Puts drive in “hand wheel” mode until the given digital or analog input condition is met. Hand wheel mode is a kind of low speed following mode, where the motor follows master encoder signals as a hand wheel is manually turned. This command differs from the FE command in that the AC, DE, and DI commands are not used in any way. This means the motor will attempt to follow the master encoder signals without injecting any ramps to smoothly approach high frequency target speeds or to come to a stop when the stop input condition is met.

Inputs STEP and DIR are used for connecting the A and B signals of the encoder-based handwheel. The EG (Electronic Gearing) command defines the following resolution of the step motor.

See also: EG and FE commands. See AT command for using analog input as sensor input.

***COMMAND STRUCTURE:***

HW{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameters*** | See “Appendix C: Working with Inputs and Outputs” |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| HWX4L | – | Run in hand wheel mode until input X4 low |

### *Immediate Status Commands*

The following section describes commands that return "Immediate" results when sent. These selected commands provide useful information for monitoring internal values from the drive.

Data can be sent out in two different formats, Hexadecimal or Decimal. By default the data is returned in Hexadecimal because of its speed and efficiency. Conversion to ascii in the Decimal format is slower and causes a slight delay that varies in length. Hexadecimal minimizes the overhead required to convert the internal binary data to ascii form. This speeds up the process of sending out the requested data thus giving the most recent value. Typically, applications written on more powerful Host computers can easily convert a hexadecimal value to an integer value.

The Immediate Format (IF) command sets the format of the returned data to hexadecimal or decimal. For cases where a slight delay is acceptable the data can be sent out in decimal form. Setting the format affects all of the "I" commands (except IH and IL). See IF command in the following pages.

All the "I" commands can be used at any time and at the fastest rate possible limited only by the given Baud Rate (See BR and PB commands). As with any immediate type command it is acted upon as soon as it's received. Regardless of format (hex or dec) there will be a slight delay in processing the command. "Real time" usage of the data must be carefully analyzed.

## IA - Immediate Analog

Requests present analog input value. The IA command returns the "analog command" value which is derived from the analog input with gain and offset applied in SureMotion Pro.

***Command Structure:***

IA{Parameter #1}

***Details:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Analog input |
| ***– units*** | integer |
| ***– range*** | No parameter or 0 = Analog command;<br>1 = Analog input (raw AD counts) |

***Examples:***

| Command | Drive sends | Notes |
|---|---|---|
| IA | IA=2.5 | Analog command value is at mid range |
| IA1 | IA=4.99 | Analog input is near 5 volts |

Example: Send the command AV1 to the drive to set the Analog Offset Value to 1 VDC. Then apply 4 VDC across the AIN (+) and GND (-) terminals of the drive. The response to the IA command will then be very close to IA=3.00.

## IC - Immediate Current (Commanded)

Requests the present (peak-of-sine) current applied to each motor phase. This value will change depending on what themotor is doing at the moment the command is processed. If the motor is moving this value will equal the CA or CC value. If the motor is not moving, this value will equal the CI value.

***COMMAND STRUCTURE:***

IC

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | 0.01 amps |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IC | IC=015E | 3.5 amps |
| IC | IC=FEA2 | -3.5 amps |
| | | |
| With IF=D... | | |
| IFD | – | Set values to be read back in decimal |
| IC | IC=350 | 3.5 amps |
| IC | IC=-350 | -3.5 amps |

## ID - Immediate Distance

Requests present relative distance from the beginning of the last move.

***COMMAND STRUCTURE:***

ID

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | steps |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| With IF=H... | | |
| ID | ID=00002710 | 10000 (10000 counts into CW move) |
| ID | ID=FFFFD8F0 | -10000 (10000 counts into CCW move) |
| | | |
| With IF=D... | | |
| ID | ID=10000 | 10000 counts into CW move |
| ID | ID=-10000 | 10000 counts into CCW move |

## IE - Immediate Encoder

Requests present encoder position in hex.

See also: IF command

***COMMAND STRUCTURE:***

IE

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | Encoder counts (default is hex value) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IE | IE=00002710 | Encoder position is +10000 counts |
| IE | IE=FFFFD8F0 | Encoder position is -10000 counts |

If the IF (Immediate Format) comand is set with Parameter #1=D...

| | | |
|---|---|---|
| IE | IE=10000 | Encoder position is +10000 counts |
| IE | IE=-10000 | Encoder position is -10000 counts |

## IF - Immediate Format

Sets the data format, hexadecimal or decimal, for data returned using all "I" commands (except IH, IL, IO and IS).

Data can be requested from the drive in two formats: hexadecimal or decimal. By default data is returned in hexadecimal because of its speed and efficiency. Conversion to ascii in the decimal format is slower and causes a slight delay that varies in length. Hexadecimal minimizes the overhead required to convert the internal binary data to ascii form. This speeds up the process of sending out the requested data thus giving the most recent value. Typically, applications written on more powerful host computers can easily convert a hexadecimal value into a decimal value.

All "I" commands can be used at any time and at the fastest rate possible limited only by the baud rate. Immediate commands are executed as they are received, regardless of what is in the drive's command buffer. Regardless of format (hex or dec) there will be a slight delay in processing the response to an "I" command. "Real time" usage of the data must be carefully analyzed.

Affects: IA, ID, and IP commands

***COMMAND STRUCTURE:***

IF{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Return format |
| ***– units*** | letter |
| ***– range*** | H (hexadecimal) or D (decimal) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IFH | – | Sets format to Hexadecimal |
| ID | ID=00002710 | Distance is 10000 counts |
| IF | IF=H | |
| | | |
| IFD | – | Sets format to Decimal |
| ID | ID=10000 | Distance is 10000 counts |
| IF | IF=D | |

## IH - Immediate High Output

Sets the output high (open) immediately. Use SO if you don't want the output to change until a buffered command (like a move) is complete.

See also: IL, SO commands

***COMMAND STRUCTURE:***

IH(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Output number |
| ***– units*** | integer |
| ***– range*** | 1 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IH1 | – | Output set high immediately, regardless of what commands are in the command buffer |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## IL - Immediate Low Output

Sets the output low (closed) immediately. Use SO if you don't want the output to change until a buffered command (like a move) is complete.

See also: IH, SO commands

***COMMAND STRUCTURE:***

IL(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Output number |
| ***– units*** | integer |
| ***– range*** | 1 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IL1 | – | Output set low immediately |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## IO - Output Status

With no parameter this command requests the immediate status of the designated outputs. The status is displayed as an 8-bit binary number with input 1 in the far right position (bit 0). With a parameter this command sets the outputs high or low using the decimal equivalent of the same binary pattern. Logic zero ("0") turns an output on by closing it.

*NOTE: Since the STP-DRV drives only have one digital output, this command will only return values of 0 or 1.*

See also: IS command

***COMMAND STRUCTURE:***

IO{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Decimal equivalent of binary output pattern |
| ***– units*** | integer |
| ***– range*** | 0, 1 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IO | IO=00000000 | Output is low (closed) |
| IO | IO=00000001 | Output is high (open) |
| | | |
| IO0 | – | Set output low (closed) |
| IO1 | – | Set output high (open) |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## IP - Immediate Position

Requests present absolute position. The position data is assigned a 32-bit value. When sent out in Hexadecimal it will be 8 characters long. When sent out in decimal it will range from 2147483647 to -2147483648.

***COMAND STRUCTURE:***

IP

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | steps (default is hex value) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| With IF=H... | | |
| IP | IP=00002710 | Absolute position is 10,000 steps |
| IP | IP=FFFFD8F0 | Absolute position is -10,000 steps |
| | | |
| With IF=D... | | |
| IP | IP=10000 | Absolute position is 10000 steps |
| IP | IP=-10000 | Absolute position is -10000 steps |

## IS - Input Status

Requests immediate status of all three digital inputs, STEP, DIR, and EN.

***COMMAND STRUCTURE:***

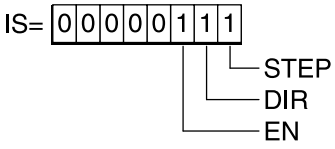
IS

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IS | IS=00000000 | All 3 inputs are closed |
| IS | IS=00000111 | All 3 inputs are open |
| IS | IS=00000001 | STEP input is open, DIR and EN inputs are closed |
| IS | IS=00000101 | STEP and EN inputs are open, DIR input is closed |




*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## IT - Immediate Temperature

Requests drive temperature, as measured by either an on-chip or board-mounted sensor.

The temperature reads out in decivolts, or units of 0.1 degrees celcius. The drive will fault when the temperature reaches a specified maximum value. Overtemp occurs at 85°C.

***COMMAND STRUCTURE:***

IT

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– range*** | 0 - 1 |
| ***– units*** | 0.1 degrees C |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IT | IT=275 | Drive temperature is 27.5°C |

## IU - Immediate Voltage

Requests the present value of the DC bus voltage, +/- 5%. The voltage reads out in 0.1 volts resolution. The drive will fault when the DC bus voltage reaches a specified maximum value. An alarm will be set when the DC bus voltage is less than a minimum value (see hardware manuals for details).

See also: IF command

***COMMAND STRUCTURE:***

IU

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | 0.1 VDC, +/- 5% |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| If the IF command is set with Parameter #1=H | | |
| IU | IU=01E2 | DC supply voltage is 48.2 volts |
| IU | IU=067E | DC bus voltage is 166.2 volts |
| | | |
| If the IF command is set with Parameter #1=D | | |
| IU | IU=482 | DC supply voltage is 48.2 volts |
| IU | IU=1662 | DC bus voltage is 166.2 volts |

## IV - Immediate Velocity

Requests the present velocity of the motor in rpm. There are two different velocities that can be read back: the motor's actual velocity and the motor's target velocity.

***COMMAND STRUCTURE:***

IV{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Velocity selector |
| ***– units*** | Integer |
| ***– range*** | 0 = actual velocity request (drives with encoder)<br>1 = target velocity request |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| IV0 | IV=1000 | Motor is running at 1000 rpm |
| IV1 | IV=1000 | Target motor velocity is 1000 rpm |

## JA - Jog Acceleration

Sets or requests the accel/decel rate for Jog moves in rev/sec/sec. Sending JA with no number causes drive to respond with present jog accel/decel rate. Setting JA overwrites the both the last JA and JL values. This means that to have different jog accel and jog decel values, you should first send JA to set the jog accel and then send JL to set the jog decel. The JA value cannot be changed while jogging. To change jog speed while jogging use the CS command.

Affects: CJ, WI (jogging) commands

See also: CS, JD, JE, JL, JS, and SJ commands

***COMMAND STRUCTURE:***

JA{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Jog acceleration value |
| ***– units*** | rev/sec/sec (rps/s) |
| ***– range*** | 0.167 to 5461.167 (resolution is 0.167 rps/s) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JA10 | – | Set jog acceleration to 10 rev/sec/sec |
| JA | JA=10 | |

## JC - Velocity (Oscillator) Mode Second Speed

Sets or requests the second speed used in velocity (oscillator) mode. The EN input is used to select the speed set by the JC command. This only applies to Command Mode (CM) 18.

Affects: Velocity (oscillator) mode

See also: CM command

***COMMAND STRUCTURE:***

JC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Analog velocity mode second speed |
| ***– units*** | rev/sec (rps) |
| ***– range*** | 0.0042 to 133.0 rps |
| ***– default*** | 5 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JC11 | – | Set second speed in velocity mode to 11 rps |
| JC | JC=11 | |

## JD - Jog Disable

Disables the STEP and DIR inputs as jog inputs during a WI instruction.

Affects: Jogging during WI command

See also: JE and WI commands

***COMMAND STRUCTURE:***

JD

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JD | – | Disable jogging while executing a WI command |

## JE - Jog Enable

Enables the STEP and DIR inputs as jog inputs during a WI instruction. Jog accel, decel and velocity are set using the JA, JL, and JS commands, respectively.

The STEP and DIR inputs can each be assigned to only one function in an application. If you want to use the STEP and DIR inputs as jog inputs you can define them as such with the JE command. JE takes no effect if the drive is set in Command Mode (CM) 7, 14 or 18, because these modes predefine these inputs and take precedence over the JE command. In other words, the JE command, as well as Command Modes (CM) 7, 14 and 18 each assign a usage to the STEP and DIR inputs. Each of these must exclusively use the STEP and DIR inputs. Command Modes are most dominant and will continually prevent JE from using the inputs.

To enable jogging with the STEP and DIR inputs simply execute the JE command with CM=21.

Affects: Jogging during WI command

See also: JD and WI commands

***COMMAND STRUCTURE:***

JE

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JE | – | Enable jogging while executing the WI command |

## JL - Jog Decel

Sets or requests the decel rate for Jog moves and velocity (oscillator) mode in rev/sec/sec. The JL value cannot be changed while jogging. JA sets both the JA and JL values, so when a different JL value is required set JA first, then set JL.

Affects: Jogging during WI command, velocity (oscillator) mode, and CJ command

See also: CS, JA, JE, JS, and SJ commands

***COMMAND STRUCTURE:***

JL{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Jog deceleration rate |
| ***– units*** | rev/sec/sec (rps/s) |
| ***– range*** | 0.167 to 5461.167 rps/s (resolution is 0.167 rps/s) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JL25 | – | Sets jog deceleration rate to 25 rps/s |
| JL | JL=25 | |

## JS - Jog Speed

Sets or requests the speed for Jog moves in rev/sec. Sending JS with no number causes drive to respond with present jog speed.

Affects: Jogging during WI command, oscillator (velocity) mode, and CJ command

See also: CJ, CS, DI, and JE commands

***COMMAND STRUCTURE:***

JS{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | Yes |
| ***Parameter #1*** | Jog speed |
| ***– units*** | rev/sec (rps) |
| ***– range*** | 0.0042 to 133.3333 (resolution is 0.0042 rps) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| JS10.35 | – | Set jog speed to 10.35 rps |
| JS | JS=10.35 | |

## LV - Low Voltage Threshold

Sets or requests the low voltage threshold for under voltage alarm/fault conditions. In DC drives, an under voltage condition causes an alarm. If desired, the user can change the low voltage threshold of the drive, however in most applications it is neither necessary nor recommended. The factory default for low voltage threshold is set to both protect the drive from damage and work with the widest range of supply voltages possible.

Affects: Under voltage alarm and fault

***COMMAND STRUCTURE:***

LV{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Low voltage threshold |
| ***– units*** | 0.1 volts DC |
| ***– range*** | STP-DRV = 12 to 75<br>STP-MTRD = 10 to 75 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| LV | LV=180 | Low voltage threshold of STP-DRV set at 18VDC |
| LV200 | – | Set low voltage threshold of STP-DRV to 20VDC |

## MC - Motor Current, Rated

Specifies the maximum current that can be sent to the motor. This is the same value set in SureMotion Pro's custom motor screen for rated current.

This value serves as the upper ceiling for the CC command, preventing excessive current from being sent to the motor, potentially damaging it. It is also used when the motor is probed to determine its electrical characteristics (see PN command for details).

See also: CC and PN commands

***COMMAND STRUCTURE:***

MC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Motor rated current |
| ***– units*** | Amps |
| ***– range*** | 0 - 10.00 amps |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| MC2.5 | – | Motor maximum current set to 2.5A |
| MC | MC=2.5 | |

## MD - Motor Disable

Disables motor outputs (reduces motor current to zero). Disabling the motor also activates the Brake Output function (see BO command). Motor current is not reduced to zero until the Brake Engage (BE command) time has expired.

Affects: All move commands

See also: BE, BO, and ME commands

***COMMAND STRUCTURE:***

MD

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| MD | – | Drive turns off current to the motor |

## ME - Motor Enable

Restores drive current to motor. If the drive cannot be enabled due to the Enable Input (SI) state, the drive will respond with a "&" which indicates that the drive could not be enabled. Enabling the drive also deactivates the Brake Output function (see BO command). Enabling of the motor is delayed by the BD (Brake Disengage) time delay.

*WARNING: THIS COMMAND RESTORES THE PREVIOUS MODE OF OPERATION. IF, FOR EXAMPLE, THE DRIVE IS OPERATING IN ANALOG VELOCITY MODE, THE MOTOR MAY IMMEDIATELY START MOVING. EXTERNAL INPUTS TO THE DRIVE MUST BE SEQUENCED PROPERLY TO AVOID UNPREDICTABLE OPERATION.*

Affects: All move commands

See also: BD, BO, and MD commands

### *COMMAND STRUCTURE:*

ME

### *DETAILS:*

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

### *EXAMPLES:*

| Command | Drive sends | Notes |
|---|---|---|
| ME | – | Drive is enabled |
| ME | & | Drive is NOT enabled; check Enable Input (SI) for proper state |

## MO – Motion Output

**For Standard Input Drives:**

Defines the drive's Motion Output digital output function. For standard input stepper drives there are eight Motion Output states that can be defined with the MO command:

MO1: Output is closed (energized) when motor is not moving.

MO2: Output is open (de-energized) when motor is not moving.

MO3: Output is not used as a Motion Output and can be used for another automatic output function or as a general purpose output.

MO4: Output is used as a Tach Output at 100 pulses/rev with 1.8 degree step motor.

MO5: Output is used as a Tach Output at 200 pulses/rev with 1.8 degree step motor.

MO6: Output is used as a Tach Output at 400 pulses/rev with 1.8 degree step motor.

MO7: Output is used as a Tach Output at 800 pulses/rev with 1.8 degree step motor.

MO8: Output is used as a Tach Output at 1600 pulses/rev with 1.8 degree step motor.

Each of the five available output functions (alarm output, brake output, motion output, tach output, or general purpose output) must exclusively use the output, so only one function is allowed. The output can be defined using the SureMotion Pro software or via the MO command.

**For Variable I/O Drives:**

Drives with Variable I/O allow a second parameter which allows the user to specify the I/O point used as the Motion Output. Before an I/O point can be used as a Motion Output, it must be configured as an output with the SD command.

Possible uses for the MO command with Variable I/O are as follows ("n" denotes the I/O point to be used):

MO1: Output is open (de-energized) when static position error is less than set value.

MO2: Output is closed (energized) when static position error is less than set value.

MO3: Output is not used as a Motion Output and can be used for another automatic output function or as a general purpose output.

MO4n: Output is used as a Tach Output at 100 pulses/rev with 1.8 degree step motor.

MO5n: Output is used as a Tach Output at 200 pulses/rev with 1.8 degree step motor.

MO6n: Output is used as a Tach Output at 400 pulses/rev with 1.8 degree step motor.

MO7n: Output is used as a Tach Output at 800 pulses/rev with 1.8 degree step motor.

MO8n: Output is used as a Tach Output at 1600 pulses/rev with 1.8 degree step motor.

*NOTE: Setting the MO command to 1, 2, or 4-8 overrides previous assignments of this output's function. Similarly, if you use the AO or BO command to set the function of the output after setting the MO command to 1 or 2, usage of the output will be reassigned and AO will be automatically set to 3.*

Affects: Motion Output usage

See also: AO, BO, PL, and SD commands

***COMMAND STRUCTURE:***

MO{Parameter #1}

MO{Parameter#1}{Parameter #2} (for Variable I/O only)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Output Usage |
| ***– units*** | Integer code |
| ***– range*** | 1-8 |
| ***Parameter #2*** | I/O Point (if applicable)* |
| ***– units*** | Integer code |
| ***– range*** | 1-4 |

*Note: For drives equipped with Variable I/O, the SD command must be executed to set an I/O point as an output before it can be used as the Motion Output.*
**Parameter #2 only applies to drives equipped with Variable I/O. Parameter #2 is not defined for drives equipped with standard I/O.*

***EXAMPLES:***

- Standard Output Drives

| Command | Drive sends | Notes |
|---|---|---|
| MO1 | – | Motion output will close when the motor is not moving |
| MO | MO=1 | |

- Drives with Variable I/O

| Command | Drive sends | Notes |
|---|---|---|
| SD4O | – | Configures I/O 4 as output (see SD command for details) |
| MO14 | – | Motion Output is mapped to I/O 4 and will close when the motor is not moving |
| MO | AO=14 | |

*NOTE: When working with digital inputs and outputs it is important to remember the designations "low" and "high". If current is flowing into or out of an input or output (the circuit is energized), the logic state for that input/output is defined as "low" or closed. If no current is flowing (circuit is de-energized), or the input/output is not connected, the logic state is "high" or open. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 low", and SO1L means "set output 1 low." A high state is represented by the "H" character.*

## MV - Model & Revision

Requests the connected drive's DSP firmware version, model number code, and sub-model number code (if applicable). Not all drive series utilize the sub-model number code. The response from the drive is a single string of characters with no breaks or delimiters. The sequence of characters is firmware revision (3 numbers and 1 letter), model number code (3 numbers), sub-model number code (1 letter). See below for details.

See also: RV command

***COMMAND STRUCTURE:***

MV

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | No |

| ***Drive*** | ***Firmware*** | ***Model No. Code*** | ***Sub-Model No. Code*** |
|---|---|---|---|
| ***STP-DRV-4850*** | * | 020 | - |
| ***STP-DRV-80100*** | * | 021 | - |
| ***STP-MTRD-17*** | n/a | n/a | n/a |
| ***STP-MTRD-23*** | n/a | n/a | n/a |
| ***STP-MTRD-17030R*** | * | 055 | E |
| ***STP-MTRD-17030RE*** | * | 055 | K |
| ***STP-MTRD-17038R*** | * | 055 | F |
| ***STP-MTRD-17038RE*** | * | 055 | L |
| ***STP-MTRD-23042R*** | * | 049 | C |
| ***STP-MTRD-23042RE*** | * | 049 | G |
| ***STP-MTRD-23065R*** | * | 049 | D |
| ***STP-MTRD-23065RE*** | * | 049 | H |
| ***STP-MTRD-24075RV*** | * | 051 | D |
| ***STP-MTRD-24075RVE*** | * | 051 | H |

*See example below for format of firmware version.

- Denotes no sub-model number code is used for this drive.

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| MV | 100Q055K | Drive connected has DSP firmware version 1.00Q, and the drive model number is STP-MTRD-17030RE |
| MV | 103F051D | Drive connected has DSP firmware version 1.03F, and the drive model number is STP-MTRD-24075RV |
| MV | 100Q020 | Drive connected has DSP firmware version 1.00Q, and the drive model number is STP-DRV-80100 |

## PA - Power-up Acceleration Current

Sets or requests the power-up accel/decel current setting ("peak of sine") of the stepper drive, also known as the peak current. PA is similar to the CA command in that a change to PA affects the current value of the accel/decel current. However, PA differs from CA in that a change to PA is automatically written to non-volatile memory at the time of the change. For a change in CA to be written to non-volatile memory an SA command must be executed afterwards. PA will only accept parameter values equal or larger than the current PC setting.

Relationship of PA and CA:

- A change to PA affects the current accel/decel current value and is automatically stored in non-volatile memory.
- A change to PA automatically changes CA and saves the value.
- A change to CA only affects the current accel/decel current value, but does not automatically change PA to the same value.
- A change to CA is stored in non-volatile memory only after an SA command is executed. When this occurs, the PA command is also automatically changed to the new value.

*NOTE: PA has no effect in Command Mode 7 (CM7 - Step and Direction mode).*

Affects: Motor accel/decel current and torque

See also: CA, CC, PC, and SA commands

***COMMAND STRUCTURE:***

PA{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Power-up accel/decel current |
| ***– units*** | Amps (resolution is 0.01 amps) |
| ***– range*** | STP-MTRD-17 = 0 - 2.0<br>STP-MTRD-23 = 0 - 5.0<br>STP-MTRD-24 = 0 - 6.0 |

SureMotion Pro software may also be used to set all current levels.

*NOTE: This data is saved to non-volatile memory immediately upon execution. It is not required to execute the SA command to save to non-volatile memory.*

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PA1.2 | – | Set power-up accel/decel current to 1.2 amps (peak of sine) |
| PA | PA=1.2 | |

## PB - Power-up Baud Rate

Sets or requests the power-up baud rate for serial communications. When executed, this command sets the baud rate and immediately saves it to non-volatile memory. At power-up the drive defaults to 9600 baud. If configuration software such as SureMotion Pro is not detected after 1 second and the drive is configured for host operation, the drive will set the baud rate according to the value stored in the Power-up Baud Rate non-volatile parameter. A host system can change the baud rate at any time.

*NOTE: Setting the baud rate takes effect immediately.*

See also: BR, PR, and TD commands

***COMMAND STRUCTURE:***

PB{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES (see note below) |
| ***Parameter #1*** | Baud rate code |
| ***– units*** | Integer code |
| ***– range*** | 1 = 9600<br>2 = 19200<br>3 = 38400<br>4 = 57600<br>5 = 115200 |

*NOTE: This data is saved to non-volatile memory immediately upon execution. It is not required to execute the SA command to save to non-volatile memory.*

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PB2 | – | Power-up baud rate is set to 19200 and this value is immediately saved to non-volatile memory |
| PB | PB=2 | |

## PC - Power-up Current

If using a stepper drive, PC sets or requests the current setting ("peak of sine") of the stepper drive, also known as the running current.

*NOTE: This command is similar to CC. It differs only in that in addition to setting the continuous current of the drive, PC also immediatley saves the setting to NV memory. See CC command for further details.*

Affects: Motor current and torque

See also: CC and PI commands

***COMMAND STRUCTURE:***

PC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES (see note below) |
| ***Parameter #1*** | Running current |
| ***– units*** | Amps (resolution is 0.01 amps) |
| ***– range*** | STP-DRV-4850 = 0 - 5.0<br>STP-DRV-80100 = 0 - 10.0<br>STP-MTRD-17 = 0 - 2.0<br>STP-MTRD-23 = 0 - 5.0<br>STP-MTRD-24 = 0 - 6.0 |

*NOTE: We recommend using SureMotion Pro software to select a motor and set the maximum current.*

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PC3.2 | – | Set power-up to 3.2 amps running current |
| PC | PC=3.2 | |

## PF - Position Fault

Sets or requests the "percentage of torque" (torque utilization) used in the Stall Prevention function for systems with an encoder installed on the motor. Making this setting with the PF command requires that an SA (Save) command be sent afterwards, then a power-down/power-up cycle before the change will take effect. It is recommended that the SureMotion Pro software be used to make this setting.

***COMMAND STRUCTURE:***

PF{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Percentage of torque |
| ***– units*** | Percentage |
| ***– range*** | 0 - 100 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PF50 | – | Set percentage of torque to 50% in stepper drive fitted with encoder and with the Stall Prevention function turned on |
| PF | PF=50 | |

## PI - Power-up Idle Current

Idle current is the level of current supplied to each motor phase when the motor is not moving. Using an idle current level lower than the running motor current (see CC and PC commands) aids in motor cooling. A common level used for the idle current setting is 50% of the running current. After a motor move, there is a time delay after the motor takes its last step before the reduction to the idle current takes place. This delay is set by the CD command.

This command is similar to the CI command. It differs only in that in addition to setting the idle current of the drive, PI also immediately saves the setting to NV memory. See the CI command page for details.

Affects: Motor current at standstill, holding torque

See also: CC, CD, and CI commands

***COMMAND STRUCTURE:***

PI{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES (see note below) |
| ***Parameter #1*** | Idle current at power-up |
| ***– units*** | Amps |
| ***– range*** | 0 - 90% of running current |

*NOTE: This data is saved to non-volatile memory immediately upon execution. It is not required to execute the SA command to save to non-volatile memory.*

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PI0.75 | – | Set power-up idle current to 0.75 amps |
| PI | PI=0.75 | |

## PM - Power-up Mode

Sets or requests the power-up mode of the drive. PM determines how the drive is configured for serial communications at power-up. For example, the SCL applications set PM=2 or PM=5. The power-up mode is also set when configuring the drive with SureMotion Pro software.

*NOTE: If the drive is configured for power-up mode 3, it will not respond to SCL commands issued by a host device. If SCL communications are required in this scenario, the host device must recognize the drive's power-up packet and issue the response "00" (double-zero, no carriage return) within two seconds to force the drive into SCL mode without altering the PM setting. See Appendix A for further information.*

See also: CM command

***COMMAND STRUCTURE:***

PM{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES (see note below) |
| ***Parameter #1*** | Power on mode |
| ***– units*** | Integer code |
| ***– range*** | 1 = not used<br>2 = SCL (drive enabled)<br>3 = SureMotion Pro<br>4 = not used<br>5 = SCL (drive disabled)<br>6 = not used<br>7 = not used |

*NOTE: This data is saved to non-volatile memory immediately upon execution. It is not required to execute the SA command to save to non-volatile memory.*

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PM2 | – | Drive will power-up in SCL mode (drive enabled) |
| PM | PM=2 | |

## PN - Probe On Demand

Perform a full-current probe of the motor. The motor's maximum rated current is used as defined by the MC command. This allows the drive to dynamically measure electrical parameters such as inductance and resistance, which is used to optimize the drive's control over the motor.

This probe is automatically done on power-up and after an EF command is issued, but may be performed at any time using the PN command.

*NOTE: This operation will briefly energize the motor with full current. Use caution when executing the PN command as this may cause slight movement of the motor shaft.*

See also: EF and MC commands

***COMMAND STRUCTURE:***

PN

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PN | – | Perform a full-current probe of the motor |

## PR - Protocol

Sets or requests the serial communication protocol settings. There are a number of settings that can be turned on or off in the PR command. Each setting is assigned a bit in a 9-bit binary word. The parameter of the PR command is the decimal equivalent of this word. If you send the PR command without a parameter, the drive will respond with the decimal equivalent of the word as well. The different protocol settings and their bit assignments are shown below. For more information, please refer to Appendix E, "The PR Command".

Affects: RS-232 and RS-485 serial communications

See also: BR and PB commands

***COMMAND STRUCTURE:***

PR{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Protocol code |
| ***– units*** | Decimal (integer) value of binary word |
| ***– range*** | 1 - 63 (00001 - 111111)<br><br>bit 0 = Default ("Standard SCL")<br>bit 1 = Always use Address Character<br>bit 2 = Ack/Nack<br>bit 3 = Checksum<br>bit 4 = (reserved)<br>bit 5 = (reserved)<br>bit 6 = (reserved)<br>bit 7 = (reserved)<br>bit 8 = Full Duplex in RS-485 (4-wire) |

*NOTE: Bit 0 is only required when all other bits are set to 0. If any other bit in the word is set to 1, Bit 0 is ignored. For example, PR4 and PR5 provide the same protocol settings.*

Bit 3 = 1: has checksum / 0: No checksum

Bit 8 = 1: Full Duplex / 0: Half Duplex

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PR1 | – | Set to standard SCL protocol |
| PR3 | – | Turn Checksum on |
| PR | PR=3 | |

## PS - Pause

Suspends execution of buffered commands until the next Continue (CT) command is executed. This can be useful for holding a sequence of commands in the drive's command buffer to time with an external event. Use the PS command to pause the command buffer, then send each (buffered type) command in the desired sequence to the drive. When the timing with the external event occurs, simply send the CT command which will trigger the execution of the already buffered sequence of commands.

See also: CT command

***COMMAND STRUCTURE:***

PS

Details:

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| PS | – | Pause execution of commands in the command buffer |

## RE - Restart or Reset

Restarts the drive by resetting fault conditions and re-initializing the drive with the startup parameters. Leaves the drive in a disabled state to prevent any movement after the restart is complete. RE is not a factory reset. All volatile parameters will be lost unless saved using the SA command.

***COMMAND STRUCTURE:***

RE

Details:

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| RE | – | Resets drive condition and parameters |

## RO - Anti-Resonance ON

Enables or disables the Anti-Resonance algorithm. This command has the same effect as the "Anti-Resonance off" check box in SureMotion Pro's motor configuration dialog.

***COMMAND STRUCTURE:***

RO{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Anti-resonance Algorithm Status |
| ***– units*** | Integer |
| ***– range*** | 0 = Anti-resonance OFF<br>1 = Anti-resonance ON |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| RO1 | – | Enable Anti-resonance algorithm |
| RO | RO=1 | |

## RS - Request Status

Requests the immediate status of the drive. This basically asks the drive to respond with what it's doing. The drive has a number of different states of operation that are represented by character codes. The drive can send more than one code at a time to define its current status.

See also: SC command

***COMMAND STRUCTURE:***

RS

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | character code |
| ***– range*** | A = Alarm Code is present (use AL command to view alarm code)<br>D = Drive Disabled<br>E = Drive Faulted (use AR command to clear fault)<br>H = Homing (SH in progress)<br>J = Jogging (CJ in progress)<br>F = Motion in progress (Feed & Jog commands)<br>M = Motion in progress (from any kind of input)<br>R = Ready (drive is enabled and ready)<br>S = Stopping a motion (ST or SK command executing)<br>T = Wait Time (WT command executing)<br>W = Wait Input (WI command executing) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| RS | RS=R | Drive is enabled and ready |
| RS | RS=ADE | Alarm code is present, drive is faulted and disabled |
| RS | RS=JR | Motor is jogging, drive is enabled |

## RV - Revision Level

Requests the drive's firmware version. Data is returned as a three digit value. To see the firmware version's sub-letter as well (if applicable), use the MV command.

See also: MV command

***COMMAND STRUCTURE:***

RV

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | Drive firmware version |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| RV | RV=150 | Drive is running firmware version 1.50 |

## SA - Save Parameters

Saves selected command parameters to non-volatile memory. All non-retentive (volatile) commands are saved when the SA command is sent. This command is useful for saving the desired defaults for subsequent power-ups.

***COMMAND STRUCTURE:***

SA

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SA | – | Save all data registers and commands |

## SC - Status Code

Requests the current drive status as the Hexadecimal equivalent of a binary word. Each bit in the binary word relates to a status condition (see assignments below). The representation of this binary word as a hexadecimal value is called the Status Code. Drives can have multiple status conditions at one time, and host systems can typically interpret a Hexadecimal code very quickly. See Appendix B for more details on the Status Code.

See also: RS command

***COMMAND STRUCTURE:***

SC

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | READ ONLY |
| ***Non-Volatile*** | NO |
| ***– units*** | Hexadecimal equivalent of the binary status code word (see bit assignments below) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SC | SC=0009 | Drive is in motion and enabled (hex values 0001 and 0008) |
| SC | SC=0004 | Drive is faulted and disabled (hex value 0004) |
| SC | SC=0209 | Drive has an alarm, is in position, and is enabled (hex values 0001, 0008, and 0200) |

Status Code bit assignments:

| *Hex Value* | *Satus Code Bit Definition* |
|---|---|
| 0001 | Motor enabled (motor disabled if this bit = 0) |
| 0002 | Reserved |
| 0004 | Drive fault (check Alarm Code) |
| 0008 | In position (motor is in position) |
| 0010 | Moving (motor is moving) |
| 0020 | Jogging (currently in jog mode) |
| 0040 | Stopping (in the process of stopping from a stop command) |
| 0080 | Waiting (for an input; executing a WI command) |
| 0100 | Saving (parameter data is being saved) |
| 0200 | Alarm present (check Alarm Code) |
| 0400 | Homing (executing an SH command) |
| 0800 | Waiting (for time; executing a WD or WT command) |
| 1000 | Reserved |
| 2000 | Checking encoder |
| 4000 | Reserved |
| 8000 | Initializing (happens at power up) |

## SD - Set Direction

The Variable I/O feature allows the user to specify the direction of each I/O point - i.e. to configure each as either an input or output. SD may be used as a query if issued without a parameter. The drive will then report the direction of each I/O point.

*WARNING: The SD command allows dynamic changes to I/O behavior of the drive, and may cause unintended interactions with other machine components if not implemented properly. Extreme caution should be used. The SD command is documented here only for completeness; we strongly recommend that SureMotion Pro software be used to make changes to drive I/O behavior.*

Affects: All input and output commands

See also: AI, AO, BO, DL, MO, and SI commands

***Command Structure:***

SD{Parameter #1}{Parameter #2}

***Details:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | I/O point to configure |
| ***– units*** | Integer |
| ***– range*** | 1 - 4 |
| ***Parameter #2*** | Direction (input or output) |
| ***– units*** | Single character |
| ***– range*** | I = input<br>O = output<br>Note: Both characters are letters, not numbers |

*Note: This command requires either the letter "I" or "O" as parameter #2. The drive's response, however, is composed of the numbers 1 (one = input) or 0 (zero = output).*

***Examples:***

| Command | Drive sends | Notes |
|---|---|---|
| SD2O | – | Set I/O point 2 as an output |
| SD4I | – | Set I/O point 4 as an input |
| SD | SD=00000111 | Drive reports that I/O points 1, 2, and 3 are inputs, 4 is an output |

## SF - Step Filter Frequency

Sets or requests the step filter frequency which affects motor smoothness in Step & Direction mode (CM7). The primary use of this filter is to introduce microstep emulation into the motion of the step motor, which smooths motion when the drive's microstep resolution (EG command) is set to a low value. This command is exceptionally useful when using a low-resolution indexer or encoder to send pulses to the stepper drive and smooth motor shaft rotation is required.

*Note: The step filter frequency is labeled Step Smoothing Filter in the Pulse & Direction Control dialog of SureMotion Pro.*

***COMMAND STRUCTURE:***

SF{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Step filter frequency |
| ***– units*** | Hz |
| ***– range*** | 0–2500 |
| ***– default*** | 2500 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SF500 | – | Set step filter frequency to 500Hz |
| SF | SF=500 | |

## SH - Seek Home

Executes the Seek Home command. Requires input number and condition for the home sensor. Speed is set by the last VE command. Accel and decel are set by AC and DE. Direction comes from the sign of the last DI command ("-" is CCW; no sign is CW).

The SH command performs a number of operations all combined into one command. The basic operation acts like a combination of the FS (Feed to Sensor) and FP (Feed to Position) commands. First, an "FS-like" move is made that runs the motor until the motor reaches the home sensor. When the drive sees this home sensor it does two things: it records the absolute position of the home sensor and it immediately starts decelerating the motor to a stop. After the motor has come to a stop the drive then does an "FP-like" move to move the motor back to the absolute position recorded for the home sensor. Another function of the SH command is that if an end-of-travel limit switch or sensor is encountered before the home sensor condition is met, the move direction is reversed until the opposite limit is found. After the opposite limit is found the move then returns to the original direction and again attempts to find the home sensor. This always ensures that the motor is moving in the desired direction when the drive sees the home sensor.

This command is designed to use three physical sensors or switches tied to three digital inputs of the stepper drive: a home sensor, a CW end-of-travel limit switch, and a CCW end-of-travel limit switch. With STP-DRV drives tie the home sensor to the EN input, the CW end-of-travel limit switch to the STEP input, and the CCW end-of-travel limit switch to the DIR input. Use the DL command to define the usage of STEP and DIR as end-of-travel limits.

If end-of-travel limits are not used in the application, tie the home sensor to any one of the three digital inputs and use the DL command to define the STEP and DIR inputs as general purpose inputs (DL3).

See also: DL command

***COMMAND STRUCTURE:***

SH(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Input number, Input condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = STEP; 2 = DIR; 3 = EN<br>letter: L = low; H = high; F = falling edge; R = rising edge |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SH3L | – | Seek home to EN input low (STEP and DIR inputs wired to end-of-travel limit switches) |

*NOTE: When working with digital inputs and outputs in STP-DRV drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

*NOTE: When working with the analog input of an STP-DRV drive (AIN terminal), "L" designates an analog value lower than the value set by the AT command. Similarly "H" designtates an analog value greater than the value set by the AT command.*

## SI – Enable Input Usage

**For Standard Input Drives:**

Defines the EN input as an Enable Input. If you want to use the EN input as an Enable input you can define it as such in two ways - with the SureMotion Pro software or with the SI command. SI takes no effect if the drive is set in Command Mode (CM) 13, 14, 17, or 18, because these modes use the EN input as a speed change input and take precedence over the SI command. Also, setting the AI command after setting the SI command reassigns the EN input to Alarm Reset usage and turns off any drive enable usage (SI3). The AI and SI commands, as well as Command Modes (CM) 13, 14, 17, and 18 each exclusively use the EN input so only one of them can be active at a time.

There are three Enable input states that can be defined with the SI command:

SI1: Drive is enabled when the EN input is open (inactive, high).

SI2: Drive is enabled when the EN input is closed (active, low).

SI3: The EN input is not used for Enable and can be used as a general purpose input. SI will be automatically set to 3 if CM is set to 13, 14, 17, or 18, or if AI is set to 1 or 2 after the SI command is set.

**For Variable I/O Drives:**

Drives with Variable I/O allow a second parameter which allows the user to specify the I/O point used as the Enable input. Before an I/O point can be used as the Drive Enable input, it must be configured as an input with the SD command. See the hardware manual for details of which inputs may be used as the Drive Enable input.

There are three Enable input states that can be defined with the SI command ("n" denotes the I/O point to be used):

SI1n: Drive is enabled when the designated input is open (inactive, high).

SI2n: Drive is enabled when the designated input is closed (active, low).

SI3n: Designated input "n" is not used for Drive Enable and can be used as a general purpose input.

Affects: Enable input usage

See also: AI, CM, and SD commands

***COMMAND STRUCTURE:***

SI{Parameter #1}

SI{Parameter#1}{Parameter #2} (for Variable I/O only)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Input usage |
| ***– units*** | Integer code |
| ***– range*** | 1, 2, or 3 |
| ***Parameter #2*** | I/O Point (if applicable)* |
| ***– units*** | Integer code |
| ***– range*** | 1 or 3 |

*NOTE: For drives equipped with Variable I/O, the SD command must be executed to set an I/O point as an input or output before it can have a dedicated function assigned to it.*

**Parameter #2 only applies to drives equipped with Variable I/O. Parameter #2 is not defined for drives equipped with standard I/O.*

***EXAMPLES:***

- Standard Output Drives

| Command | Drive sends | Notes |
|---|---|---|
| SI1 | – | Cause drive to be enabled when X3/EN input is open |
| SI | SI=1 | |

- Drives with Variable I/O

| Command | Drive sends | Notes |
|---|---|---|
| SD3I | – | Configures I/O 3 as input (see SD command for details) |
| SI13 | – | Cause drive to be enabled when I/O 3 is open |
| SI | SI=13 | |

*NOTE: When working with digital inputs and outputs it is important to remember the designations "low" and "high". If current is flowing into or out of an input or output (the circuit is energized), the logic state for that input/output is defined as "low" or closed. If no current is flowing (circuit is de-energized), or the input/output is not connected, the logic state is "high" or open. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 low", and SO1L means "set output 1 low." A high state is represented by the "H" character.*

## SJ - Stop Jogging

Stops the motor when jogging (CJ starts jogging). Jog decel rate is defined by the JA command.

Affects: CJ command

See Also: JA, JL, SK, and ST commands

***COMMAND STRUCTURE:***

SJ

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SJ | – | Stops jogging immediately using the deceleration rate set by the JA command |

## SK - Stop & Kill Buffer

Halts any buffered command in progress and erases all buffered commands in the command buffer. When used to stop a move deceleration rate is controlled by the AM (Max Acceleration) parameter. If the "D" parameter is used deceleration rate is controlled by either DE (with "Feed" moves like FL, FP, SH) or JA (when jogging).

Affects: Motion and command buffer contents

See Also: ST command

***COMMAND STRUCTURE:***

SK{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Deceleration rate |
| ***– units*** | letter |
| ***– range*** | D = deceleration rate set by DE or JA command<br>no parameter = deceleration rate set by AM command |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SK | – | Stop motion immediately using the deceleration rate set by the AM command and erase the contents of the command buffer |
| SKD | – | Stop motion immediately using the deceleration rate set by the DE command (or JA if jogging) and erase the contents of the command buffer |

## SO - Set Output

Sets drive's digital output to the given condition, low or high. This can only be done if the digital output is not being used for a dedicated function such as Alarm Output (AO), Brake Output (BO) or Motion Output (MO).

See Also: IH, IL, IO commands

***COMMAND STRUCTURE:***

SO(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Output number, Output state |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = Output<br>letter: L = Low; H = High |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SO1L | – | Set Output low (closed) |
| SO1H | – | Set Output high (open) |

*NOTE: When working with digital inputs and outputs in SureStep drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

## SP - Set Position

Sets or requests the motor's absolute position in steps.

See Also: EP, FP command

***COMMAND STRUCTURE:***

SP{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Absolute position |
| ***– units*** | steps |
| ***– range*** | ± 2,147,483,647 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| SP100 | – | Set absolute position offset to 100 steps |
| SP | SP=100 | |

## SS - Send String

Instructs drive to respond with the desired character string (up to 4 characters). This command is useful for letting the host system know via the serial port when a sequence of commands has finished executing. Multiple SS commands can be placed into the command buffer at any time, though care should be taken when using this command to avoid serial data collisions. For example, the host system should avoid sending commands to the drive while expecting a character string (from a previously buffered SS command).

***COMMAND STRUCTURE:***

SS(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | String of characters |
| ***– units*** | any printable characters |
| ***– range*** | up to 4 characters |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| AC100 | – | Set accel rate to 100 rev/sec/sec |
| DE100 | – | Set decel rate to 100 rev/sec/sec |
| VE1 | – | Set velocity to 1 rev/sec |
| EG10000 | – | Set microstep resolution to 10000 steps/rev |
| DI100000 | – | Set move distance to 100000 steps |
| FL | – | Initiate Feed to Length move |
| SSdone | done | String "done" sent to host after FL command is completed |

## ST - Stop

Halts the current buffered command being executed, but does not affect other buffered commands in the command buffer. When used to stop a move deceleration rate is controlled by the AM (Max Acceleration) command. If a "D" parameter is used deceleration rate is controlled by either the DE command (with "Feed" moves like FL, FP, and SH) or the JL* command (when jogging).

**** Note that setting the JA command also sets the JL command.***
***If distinct JA and JL values are required, always set JL after setting JA.***

See Also: SK command

***COMMAND STRUCTURE:***

ST{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | IMMEDIATE |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Deceleration rate |
| ***– units*** | letter |
| ***– range*** | D = deceleration rate set by DE or JL command<br>no parameter = deceleration rate set by AM command |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| ST | – | Stop current command immediately and use the deceleration rate set by the AM command if motion is in progress |
| STD | – | Stop current command immediately and use the deceleration rate set by the DE or JL command if motion is in progress |

## TD - Transmit Delay

Sets or requests the time delay used by the drive when responding to a command that requests a response. Most RS-232 hosts will not require a non-zero TD value because separate Rx and Tx lines are used.

Affects: RS-232 Serial Communications

***COMMAND STRUCTURE:***

TD{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Time value |
| ***– units*** | milliseconds |
| ***– range*** | 0–32767 |
| ***– default*** | 0 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| TD10 | – | Set drive Tx time delay to 10 milliseconds |
| TD | TD=10 | |

## VC - Change Velocity

Sets or requests the "change speed" for FC moves.

Affects: FC command

***COMMAND STRUCTURE:***

VC{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Change speed value |
| ***– units*** | rev/sec (rps) |
| ***– range*** | 0.0042 to 133.3333 (resolution is 0.0042 rev/sec) |
| ***– default*** | 5 |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| VC5 | – | Set change velocity to 5 rev/sec |
| VC | VC=5 | |
| AC100 | – | Set accel rate to 100 rev/sec/sec |
| DE100 | – | Set decel rate to 100 rev/sec/sec |
| DI100000 | – | Set overall move distance to 100000 steps |
| DC75000 | – | Set change distance to 75000 steps |
| VE5 | – | Set initial velocity to 5 rev/sec |
| VC1 | – | Set change velocity to 1 rev/sec |
| FC | – | Initiate Feed to Length with Speed Change command |

## VE - Velocity

Sets or requests move speed for moves like FL, FP, FS, FD, SH, etc.

Affects: All move commands except jogging

***COMMAND STRUCTURE:***

VE{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE*<br>*Direct Logic PLCs are write only! |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Move velocity |
| ***– units*** | rev/sec (rps) |
| ***– range*** | 0.0042 to 80.0000 (resolution is 0.0042 rev/sec) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| VE2.525 | – | Set move velocity to 2.525 rev/sec |
| VE | VE=2.525 | |
| DI-20000 | – | Set move distance to 20000 steps in CCW direction |
| VE2.5 | – | Set move velocity to 2.5 rev/sec |
| FL | – | Initiate Feed to Length command |

## VM - Maximum Velocity

Sets or requests the maximum motor velocity in rev/sec. Used in analog velocity mode to limit the maximum speed of the drive.

Affects: Analog Velocity mode

See Also: AM, VC, and VE commands

***COMMAND STRUCTURE:***

VM{Parameter #1}

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | READ/WRITE |
| ***Non-Volatile*** | YES |
| ***Parameter #1*** | Move velocity |
| ***– units*** | rev/sec (rps) |
| ***– range*** | 0.0042 to 133.3333 (resolution is 0.0042 rev/sec) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| VM50 | – | Set maximum move velocity to 50 rev/sec |
| VM | VM=50 | |

## WI - Wait for Input

Waits for an input to reach the given condition. Allows very precise triggering of moves if a WI command is followed by a move command. When JE (Jog Enable) has been executed and the drive is in CM21, the STEP and DIR inputs act as jog inputs during WI commands. JD (Jog Disable) disables jogging using these inputs during WI commands.

Affects: Use of STEP and DIR inputs as jog inputs

See Also: JD, and JE commands

***COMMAND STRUCTURE:***

WI(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Input number, Input condition |
| ***– units*** | integer, letter |
| ***– range*** | integer: 1 = STEP; 2 = DIR; 3 = EN<br>letter: L = low; H = high; F = falling edge; R = rising edge |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| WI3R | – | Wait for EN input to go high (rising edge) before proceeding to the next command in the command buffer |

*NOTE: When working with digital inputs and outputs in SureStep drives it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output, i.e. the circuit is energized, the logic state for that input/output is defined as **low** or **closed**. If no current is flowing, i.e. the circuit is de-energized, or the input/output is not connected, the logic state is **high** or **open**. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input 3 (EN input) low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

*NOTE: When working with the analog input of a SureStep drive (AIN terminal), "L" designates an analog value lower than the value set by the AT command. Similarly "H" designtates an analog value greater than the value set by the AT command.*

## WT - Wait Time

Causes a time delay in seconds. The resolution is 0.01 seconds with the largest value being 320.00 seconds.

***COMMAND STRUCTURE:***

WT(Parameter #1)

***DETAILS:***

| | |
|---|---|
| ***Command Type*** | BUFFERED |
| ***Usage*** | WRITE ONLY |
| ***Non-Volatile*** | NO |
| ***Parameter #1*** | Time |
| ***– units*** | seconds |
| ***– range*** | 0.00 to 320.00 (resolution is 0.01 seconds) |

***EXAMPLES:***

| Command | Drive sends | Notes |
|---|---|---|
| WT2.25 | – | Causes time delay of 2.25 seconds |
| PS | – | Pause command buffer |
| WT1 | – | Time delay 1 second |
| FL | – | Initiate Feed to Length instruction |
| CT | – | Continue execution of commands in command buffer |

# Appendix A: Host Serial Communications

When an advanced SureStep stepper drive from Automation Direct is operating in host mode (AKA SCL mode), it means that a host device sends commands to the drive (or drives) over a serial connection and the drive executes the incoming commands. Here are some examples of typical host devices:

- A Windows-based PC running Automation Direct software
- An industrial PC running a custom-built or other proprietary software application
- A PLC with an ASCII module/serial port for sending text strings
- An HMI with a serial connection for sending text strings

The aim of this appendix is to describe the following aspects of operating an Automation Direct SureStep drive in **host mode**.

- General structure of host serial communications (host mode)
- Hardware – wiring and connecting a host device to the serial ports of an Automation Direct drive
- COM Port Settings – UART settings
- Communications Protocol
- Communication Details
- Communication Errors

## *General Structure of Host Serial Communications*

The SureStep host serial communications are based on the common ASCII character set transmitted using standard UARTs over an RS-232 hardware interface or RS-485 depending on the drive.

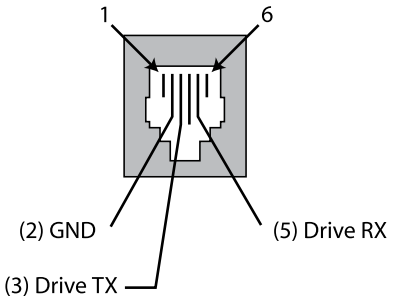
The ASCII character set is used because it is common and well-understood, as well as easy to read. UART (Universal Asynchronous Receiver Transmitter) serial transceivers are available on many types of equipment, including most PCs, and provide a common form of serial communications interface. RS-232 hardware connections are commonly used with UARTs and also provide the easiest and most common form of connectivity.

## *Hardware*

Each STP-DRV stepper drive ships with an RS-232 programming cable. A separate cable and adapter must be purchased when using the RS-485 compatible drives. This adapter can also be used with the RS-232 drives (Adapter: STP-USB485-4W, Cable: STP-485DB9-CBL-2). This cable should be used for configuring your drive with the *SureMotion Pro* software. Please see the communications chapter of the SureStep User Manual for more detailed information on setting up communications.

***RS-232:***

For RS-232 applications where the host device is a PLC or HMI, you will need to make your own communications cable. The pinouts of the RS-232 connector (RJ-11, 6P4C) on STP-DRV drives are shown in the diagram to the right. Pin 2 of the connector should be connected to your host's signal ground pin. Pin 3 of the connector should be connected to your host's Rx pin. And Pin 5 of the connector should be connected to your host's Tx pin.



*RS-422 (4-wire RS-485):*

RS-422 usually requires a special adapter to work with a PC, but is common on many types of controllers such as PLCs and HMIs. Our implementation allows for multi-drop communications with a single master (serial network). RS-485 4-wire is required for communications to SureMotion Pro. Here are some RS-422 highlights:

- Permits longer cable lengths
- May require special adapter
- Highly resistant to EMI (when wired properly)

*RS-485 (2-wire RS-485):*

Designed for multi-drop serial networks, provides simple wiring, high reliability, and long cable lengths. Here are some RS-485 highlights:

- Permits longest cable lengths: up to 1000 feet at low baud rates
- May require special adapter
- Fewest wires, smaller cables
- Highly resistant to EMI (when wired properly)

### COM Port Settings

**UART Settings**: We operate our UARTs with the following settings: 1 start bit, 8 data bits, 0 (no) parity bits, and 1 stop bit.

**Bit rate (baud) Settings (BR and PB commands)**: All AutomationDirect drives default to 9600 baud from the factory. In most cases, this speed is adequate for setup, configuring, and programming, as well as host mode communications. If higher baud rates are required, the drives can be configured to operate with a different rate using the BR (Bit rate) or PB (Power-up Bit rate) command. In all cases, the drive starts up at the factory rate, 9600, and will remain there if the "power-up packet" is acknowledged by the host (see "Drive Startup" on page page 108). When the power-up cycle is complete and if the drive has not received the power-up packet, the drive will activate the new baud rate.

Selecting a baud rate higher than the default 9600 is dependent on the application. If there is a host device operating a number of drives on a network, a higher speed may be required in order to process all the communication needs.

### Communications Protocol

In general, the protocol for communications between a host device and a drive is quite simple. SureStep series drives do not initiate communications on their own, so drives are normally in a state to receive packets from the host. A communications packet, or packet for short, includes all the characters required to complete a command (host to drive) or response (drive to host) transmission. In other words, a host initiates communication by sending a command packet, and the drive responds to that command by sending a response packet back to the host.

**Command Transmission (host to drive):** The philosophy of sending characters to the drive requires the host to send all the required characters that form a packet in a limited time frame. At the start of receiving a packet, the drive begins timing the space between characters. Each time a character is received an internal timer is reset to 200 milliseconds. If the timer reaches zero before the next character in the packet is received the drive will terminate its packet parsing (characters will still go into the receive buffer) and may send out an error response packet depending on the protocol setting. The purpose of the timeout feature is to allow the drive to purge its buffers automatically when a bad transmission occurs.

*NOTE: This timeout feature limits the usage of host devices such as the Windows application HyperTerminal, in which characters are sent as soon as they are typed. For this reason HyperTerminal is not recommended for use with these drives.*

**Response Transmission (drive to host):** In response to a command packet from the host a drive will usually send a response packet. The drive sends out its entire response packet with very limited space between characters. At 9600 baud rate the space between characters is less then 1 bit space (0.0001 seconds). The host system must be able to handle this speed.

Response packets are terminated by a Carriage Return (ASCII Dec 13 or Hex D).

**Protocol Settings (PR Command):** The PR (Protocol) command offers users the ability to add various features to the overall communications protocol, i.e. tailor the structure of command and response packets to best fit the needs of the application. In general, when a host device sends a command packet to a drive, the drive will either understand the command or not. If the drive understands the command the drive executes the command. If the drive doesn't understand the command it cannot execute the command. In most cases the host device will want to know whether the drive has understood the command or not, and so the drive can be set to automatically send an Acknowledge (understood) or Negative Acknowledge (not understood) response packet to the host for every command packet received.

Along with Acknowledge/Negative Acknowledge (Ack/Nack), the PR command controls a number of other protocol settings. See Appendix E for details on the PR command. Also, the PR command controls whether or not the drive will respond with error codes in the response packet when communications errors occur.

### *Communication Details*

**Transmit Delay (TD Command):** The TD command allows users to define a dwell time in a drive, which is used by the drive to delay the start of transmission of a response packet after the end of reception of a command packet.

When using 2-wire RS-485 networks there are times when a drive's response packet must be delayed until the network is ready for the drive to transmit. Why is this necessary? The answer is because RS-485 networks are by nature "half-duplex", which means you cannot transmit and receive at the same time. Rather, a host must first transmit, stop, then wait to receive. This is because the host and drive transmitters share the same pair of wires. When transmitting, the device that has the transmission rights must assert its transmitter outputs and therefore take control of the pair. At the same time, all other devices on the network must de-assert, or open, their transmitters so as not to interfere with the device that has the rights. Transmitters in this scenario have tri-state outputs: transmit, open, and receive.

Some devices are not as quick in opening their transmitters as others. For this reason it may be necessary for other, faster, devices on the network to dwell some time while the slower devices open their transmitters. AutomationDirect drives de-assert their transmitters very quickly. Typically it is done within 100 microseconds (.0001 seconds) after the end of a packet transmission. However, it is possible that the host device won't be this fast--the TD command allows users to set the time delay that a drive will delay after receiving a command packet before sending a response packet.

**Communications Packet:** A Communications Packet, or packet for short, includes all the characters required to complete a command or response transmission.

**Drive Startup:** At power-up, all Automation Direct drives send out what is called the "power-up packet". This packet notifies a host of the drive's presence. After sending the power-up packet the drive waits for a response from the host. This is one of the rare instances in which a drive will initiate communications with the host. This process is necessary for a number of Automation Direct software applications such as *SureMotion Pro*™. The power-up packet is an exception to the ASCII character rule in that all the characters in the packet are binary value. Even if the character is printable its binary value is what is important. The power-up packet consists of three binary characters with the first character being a binary 255 (255 is not a printable ASCII character). This character designates to the software application that the packet is a power-up packet. The following two characters are the firmware version number and the model number of the drive, respectively.

Power-Up Packet = (255)(F/W Version)(Model No.)

As an example, an STP-DRV drive with f/w version 1.53 will send out a power-up packet that looks like this: (255)(53)(38). To an ASCII terminal this packet may look like "ÿ5&". The (255) is the power-up packet designator, the (53) actually stands for f/w version 1.53 (the "1" is implied), and the (38) is an internal model number.

The power-up packet is always sent at 9600 baud, regardless of the bit rate set by the BR or PB command. If an Automation Direct software application is present it will respond to the power-up packet and communications will continue at 9600 baud. If an Automation Direct software application is not present, the drive's request made by the power-up packet will time-out and the drive will begin communicating at the saved bit rate (BR or PB command), 9600 or otherwise.

**Interaction with PM parameter (Power-up Mode):** If the drive is currently in power-up modes 1 or 3 (PM1 or PM3), it will be unable to respond to standard SCL commands. In these modes, the drive is using a proprietary communication protocol using by the SureMotion Pro configuration software. Standard SCL commands will not be recognized or acted upon by the drive in these modes. If the application requires it, the drive may be temporarily forced into SCL mode through the use of the "double zero".

**Double Zero:** When the drive initializes, it will send the power-up packet as detailed above. Typically this packet is used only by AutomationDirect software, but a host device may also use it to force SCL communication in a drive otherwise not configured to do so.

The host device must recognize the power-up packet and respond with a simple double zero (00). No carriage return is required. Note that this response must occur within 2 seconds of the power-up packet being sent, but must delay at least 2 milliseconds (0.002 seconds). This will force the drive into standard SCL mode and enable serial communication without altering the PM setting of the drive.

### *Communication Errors*

During the process of sending communication packets between the host and drive(s), two different types of communication errors can occur.

**Hardware errors:** Hardware errors are displayed physically by an STP-DRV drive via the red and green LEDs on the drive, (see Appendix B), but no response packet is automatically generated from the drive to the host. Therefore it is the responsibility of the host to check for hardware comm errors using the AL, RS, and/or SC commands. See Appendix B for more details on the AL and SC commands. Once the host has determined the presence of a hardware comm error, the nature of the error can be retrieved using the CE command.

**Parsing errors:** Parsing errors happen when a drive receives a command packet but cannot properly interpret (parse) the command. Parsing errors can automatically generate a response packet from the drive to the host, depending on the settings of the PR command (see Appendix E, PR command, Bit 2).

# Appendix B: Alarm and Status Codes

One of a drive's diagnostic tools is its ability to send alarm and status codes back to a host. The AL (Alarm Code) and SC (Status Code) commands can be used by a host to query a drive at any time. If a drive faults or sets an alarm, the AL command allows the host to find out what alarm, or alarms, has been set. Similarly, the SC command allows a host to find out what the status code of a drive is at any time during drive operation. A status code provides information as to whether the drive is running, in position, disabled, homing, and other conditions. Both alarm and status codes can be very useful when initially setting up and integrating a drive into your application.

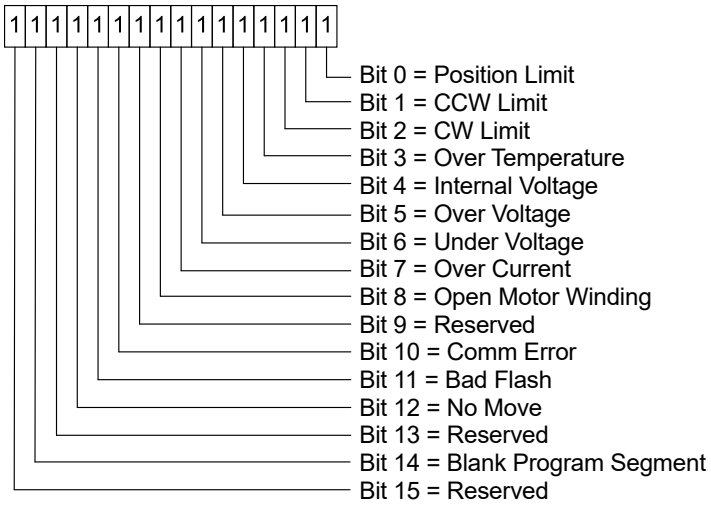
The Alarm and Status codes are hexadecimal equivalents of 16 bit binary "words". Each bit in each binary word is assigned a meaning, and therefore the responses to these two commands can actually show information about more than one alarm or status condition.

## Alarm Code Definitions

The diagram below shows the meaning assigned to each of the 16 bits in the Alarm Code's binary word. For example, if Bit 5 = 1, there is an Over Voltage condition at the drive. A drive will set any and all bits that pertain to its immediate alarm/fault status at the moment of receiving the AL command from the host.

When a host sends the AL command, the response from the drive will be the Hexadecimal equivalent of this 16-bit word. This hexadecimal value is considered the Alarm Code, and the equivalent hexadecimal value for each of the bits is given below.

| Alarm Description | Bit # | Hex Value |
|---|---|---|
| * Position Limit | 0 | 0001 |
| CCW Limit | 1 | 0002 |
| CW Limit | 2 | 0004 |
| * Over Temperature | 3 | 0008 |
| * Internal Voltage | 4 | 0010 |
| * Over Voltage | 5 | 0020 |
| Under Voltage | 6 | 0040 |
| * Over Current | 7 | 0080 |
| * Open Motor Winding | 8 | 0100 |
| Reserved | 9 | 0200 |
| Comm Error | 10 | 0400 |
| Bad Flash | 11 | 0800 |
| No Move | 12 | 1000 |
| Reserved | 13 | 2000 |
| Blank Program Segement | 14 | 4000 |
| Reserved | 15 | 8000 |




** The only alarm conditions that are categorized as "faults" (and automatically disable the motor) are listed below. These are the only alarm conditions that set the drive Status Code (SC command) "Fault" bit #2 (hex 0004).*

- Position Limit (bit #0),
- Over Temperature (bit #3),
- Internal Voltage (bit #4),
- Over Voltage (bit #5),
- Over Current (bit #7),
- Open Motor Winding (bit #8).

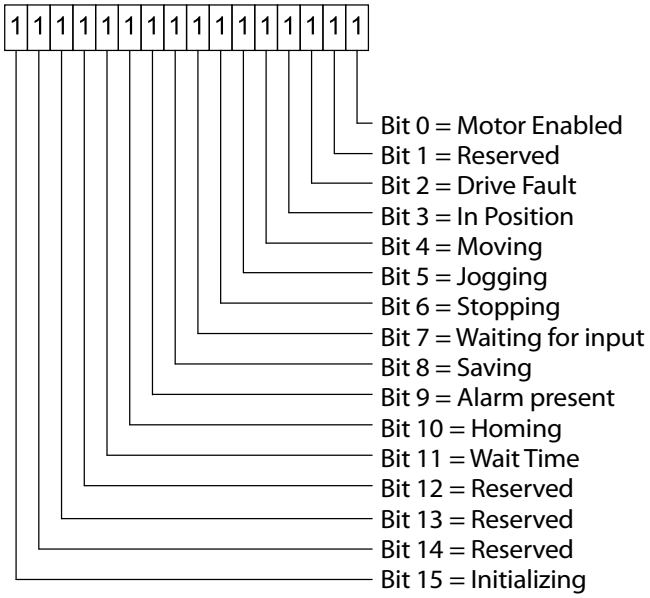
***Example:*** The drive has hit the CW limit (Bit 2, hex value 0004) and there is an under voltage condition (Bit 6, hex value 0040). The resulting 16-bit word is 0000 0000 0100 0100, and the equivalent hexadecimal value is 0044. Therefore, when the host sends "AL," the drive will respond with "AL=44."

### *Status Code Definitions*

Below is a diagram showing the meaning assigned to each of the 16 bits in the Status Code's binary word. For example, when Bit 1 = 1, the drive is disabled. Similarly, when Bit 10 = 1, the drive is seeking the home sensor (defined by the SH command). A drive will set any and all bits that pertain to its immediate status condition at the moment of receiving the SC command from the host.

When a host sends the SC command, the response from the drive will be the Hexadecimal equivalent of this 16-bit word. This hexadecimal value is considered the Status Code, and the equivalent hexadecimal value for each of the bits is given below.

| *Bit Assignment* | *Hex Value* | *Bit #* |
|---|---|---|
| Motor Enabled (Motor disabled if this bit = 0) | 0001 | 0 |
| Reserved | 0002 | 1 |
| Drive Fault (check Alarm code, AL) | 0004 | 2 |
| In Position (motor is in position) | 0008 | 3 |
| Moving (motor is moving) | 0010 | 4 |
| Jogging (currently in Jog mode) | 0020 | 5 |
| Stopping (in the process of stopping from a stop command) | 0040 | 6 |
| Waiting for an input (executing WI command) | 0080 | 7 |
| Saving (parameter data is being saved) | 0100 | 8 |
| Alarm Present (check Alarm Code) | 0200 | 9 |
| Homing (executing SH command) | 0400 | 10 |
| Wait Time (executing WT command) | 0800 | 11 |
| Reserved | 1000 | 12 |
| Reserved | 2000 | 13 |
| Reserved | 4000 | 14 |
| Initializing (happens at power up) | 8000 | 15 |




***Example:*** The drive is executing an FL command (Bit 4), and it's waiting for the input specified by the WI command (Bit 7). The 16-bit word for this condition is – 0000 0000 1001 0000 – and the hexadecimal equivalent is 90. Therefore, when the host sends "SC," the drive will respond with "SC=90."

### *Tool for Converting Alarm and Status Codes to Binary*

If you're using a Windows-based PC as a host with your SureStep drive (which you'll definitely be doing if you're using any of the AutomationDirect software supplied with your drive), you can use the Calculator utility that comes with Windows to convert hexadecimal values into binary values or "words." This utility is usually found in the Accessories folder of the Programs Folder in the Start menu. Once open, make sure the Scientific view is set by choosing it from the View menu of Calculator. This view provides some radio buttons for switching between Hex and Bin (as well as Dec and Oct).

To figure out what your Alarm or Status Code is telling you, check the Hex radio button and enter the hexadecimal code sent by the drive. Then check the Bin radio button and your code will automatically be converted to a binary word. Note that Calculator does not allow leading zeros in entries, so you may see less than 16 bits. That's OK, just start counting from the right with Bit 0, and you will be able to determine the conditions set in the codes.

## LED Display Codes

In addition to the AL and SC commands, some alarm and status codes are physically displayed at the red and green LEDs of STP-DRV drives and STP-MTRD motor/drives.

| LED Alarm/Status Display Codes | |
|---|---|
| ***Alarm Code*** | ***Error Description*** |
| solid green | no alarm, motor disabled |
| flashing green | no alarm, motor enabled |
| flashing red | configuration or memory error* |
| fast green | program running |
| 1 red, 1 green | motor stall (optional encoder only) |
| 1 red, 2 green | move attempted while drive disabled |
| 2 red, 1 green | ccw limit |
| 2 red, 2 green | cw limit |
| 3 red, 1 green | drive overheating |
| 3 red, 2 green | internal voltage out of range** |
| 3 red, 3 green | blank prog segment |
| 4 red, 1 green | power supply overvoltage** |
| 4 red, 2 green | power supply undervoltage* |
| 4 red, 3 green | flash memory backup error |
| 5 red, 1 green | over current / short circuit**† |
| 6 red, 1 green | open motor winding** |
| 6 red, 2 green | bad encoder signal (optional encoder only) |
| 7 red, 1 green | serial communication error |
| 7 red, 2 green | flash memory error |

** Does not disable the motor.*
*The alarm will clear about 30 seconds after the fault is corrected.*

*** Disables the motor. Cannot be cleared until power is cycled.*

*† The over-current/short-circuit alarm typically indicates that an electrical fault exists somewhere in the system external to the drive. This alarm does not serve as motor overload protection.*

### Alarm Code Definitions

| Error | Description | Corrective Action |
|---|---|---|
| No alarm, motor disabled | No faults active, Circuit is closed between EN+ and EN-. | N/A |
| No alarm, motor enabled | No faults active, Circuit is open between EN+ and EN-. | N/A |
| Configuration or memory error | Memory error detected when trying to load config from flash on powerup. | Restart device. No fix if restart doesn't work. |
| Program running | No faults active. | N/A |
| Motor stall (optional encoder only) | Motor torque demand exceeded capability and the motor skipped steps. This is configured in SureMotion Pro. | Increase torque utilization if it's not already maxed out, otherwise decrease the torque demand by modifying the move profile, or put in a larger motor. |
| Move attempted while drive disabled | Drive is disabled and move attempted. | Reset alarm, enable motor, and move again. |
| CCW limit | CCW limit is reached. The digital input that has been assigned CCW limit has been activated. | Unblock the CCW sensor (open the circuit) or redifine the input with SureMotion Pro. |
| CW limit | CW limit is reached. The digital input that has been assigned CW limit has been activated. | Unblock the CCW sensor (open the circuit) or redefine the input with SureMotion Pro. |

| Error | Description | Corrective Action |
|---|---|---|
| Drive overheating | The drive's internal temperature is too high. | If the drive is operating within its standard range (input voltage and output current are OK), more heat must be removed from the drive during operation. For Advanced drives (see "Mounting the Drive" on page 4-14), ensure the drive is mounted to a metal surface that can dissipate the drive's heat. For Integrated motor/drives, see "Mounting" on page 5-13. For both types of drives: If the mounting surface cannot pull enough heat away from the drive, forced airflow (from a fan) may be required to cool the drive. |
| Internal voltage out of range | Gate voltage, 5V rail, or 3V rail are out of spec. | Ensure adequate supply voltage (in very rare cases, low input voltages combined with fast accelerations can draw down the gate voltage) and try again. If persistant, RMA is required. |
| Blank prog segment | Attempt to execute a blank programming segment. | Ensure program is downloaded and try again. |
| Power supply overvoltage | The DC voltage feeding the drive is above the allowable level. | Decrease the input voltage.<br>Linear power supplies do not output a fixed voltage: the lighter the output current, the higher the output voltage will float. If a linear supply's voltage floats above the drive's max voltage, you can install a small power resistor across the linear power supply's output to provide some load that will help pull down the floating voltage.<br>Consider using a switching power supply such as the Rhino PSB power supply series.<br>Overvoltage can also be fed back into a system by regeneration (when an overhauling load pushes energy back into the motor). In an application with regen problems, install an STP-DRVA-RC-050 regen clamp to help dissipate the extra energy. (The regen clamp will not help with the floating linear power supply that floats too high, but it will help with excess voltage generated from an overhauling load.) |
| Power supply undervoltage | The DC voltage feeding the drive is below the allowable level. | Correct the power supply. If this error occurs during operation, the power supply is most likely undersized. A sudden high current demand can cause an undersized power supply to dip in output voltage. |
| Flash memory backup error | Memory error detected when trying to load config from flash on powerup. | Restart device. No fix if restart doesn't work. |
| Over current / short circuit | Motor leads shorted - only checked on powerup. | Check and fix motor wiring. |
| Open motor winding | Motor leads not connected - only checked on powerup. | Check and fix motor wiring. |
| Bad encoder signal (optional encoder only) | Noisy or otherwise incorrectly formatted encoder signal (lack of A or B, lack of differential signal). | Check encoder wiring, always use differential encoders (or use checkbox in SureMotion Pro to disable this error when using single ended). |
| Serial communication error | Catch-all error for something wrong with serial communications. See CE command in HCR for details. | If drive can communicate, CE can give a precise diagnosis. If not, refer to the Serial Communications part of the HCR for troubleshooting. |
| Flash memory error | Memory error detected when trying to load config from flash on powerup. | Restart device. No fix if restart doesn't work. |

# Appendix C: Working with Inputs and Outputs

This Appendix covers I/O usage on SureStep drives.

## *Low v. High*

When working with inputs and outputs it is important to remember the designations **low** and **high**. If current is flowing into or out of an input or output the logic state for that input/output is defined as **low** or closed. If no current is flowing, or the input/output is not connected, the logic state is **high** or open. A low state is represented by the "L" character in parameters of commands that affect inputs/outputs. For example, WI3L means "wait for input X3 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.

When working with the analog inputs, "L" designates an analog value lower than the value set by the AT command. Similarly "H" designates an analog value greater than the value set by the AT command.

## *Parameter Details*

The tables on the following pages show general I/O details for commands as they relate to specific drives. There are exceptions to these general rules, so be sure to check the command pages for the specific SCL commands you wish to implement, as well as the list of exceptions at the end of this section. For specific voltage or wiring questions, consult the user manual.

### *Input Parameter Details*

Applies to advanced drives:

- STP-DRV-4850
- STP-DRV-80100
- STP-MTRD-17R
- STP-MTRD-23R

| | |
|---|---|
| ***Parameter #1*** | Optional "X", input number, input condition<br>NOTE: Including/omitting the optional "X" has no effect on the execution of the command. |
| ***– units*** | Optional "X", integer, letter |
| ***– range*** | - Integer: 0 (encoder index, if present), 1 (STEP), 2 (DIR), 3 (EN), 4 (AIN)<br>- Letter: L = Low, H = High, F = Falling Edge, R = Rising Edge |
| ***Parameter #2*** | Input number, input condition |
| ***– units*** | Integer, letter |
| ***– range*** | - Integer: 0 (encoder index, if present), 1 (STEP), 2 (DIR), 3 (EN), 4 (AIN)<br>- Letter: L = Low, H = High, F = Falling Edge, R = Rising Edge |

Applies to advanced drive:

- STP-MTRD-24R

Drives with Variable I/O allow a user to configure I/O1 through I/O4 as either inputs or outputs by using the Set Direction (SD) command.

| | |
|---|---|
| ***Parameter #1*** | Optional "X", input number, input condition<br>NOTE: Including/omitting the optional "X" has no effect on the execution of the command. |
| ***– units*** | Optional "X", integer, letter |
| ***– range*** | - Integer: 0 (encoder index, if present), 1-4, 5 (AIN)<br>- Letter: L = Low, H = High, F = Falling Edge, R = Rising Edge |
| ***Parameter #2*** | Input number, input condition |
| ***– units*** | Integer, letter |
| ***– range*** | - Integer: 0 (encoder index, if present), 1-4, 5 (AIN)<br>- Letter: L = Low, H = High, F = Falling Edge, R = Rising Edge |

#### *Exceptions:*

When using the Follow Encoder or Hand Wheel commands (FE or HW, respectively), the master encoder channels A and B must be wired to drive inputs STEP/X1/IN1 and DIR/X2/IN2. In these modes, these inputs must not be used for sensor inputs.

Using the On Input (OI) command with no parameter will disable the interrupt function.

The Seek Home (SH) command makes use of the drive's CW and CCW limit functions. As such, the home sensor may not be wired to the following inputs:

| | |
|---|---|
| ***STP-DRV-4850*** | STEP, DIR |
| ***STP-DRV-80100*** | STEP, DIR |
| ***STP-MTRD-17*** | STEP, DIR |
| ***STP-MTRD-23*** | STEP, DIR |
| ***STP-MTRD-24*** | I/O 3, I/O 4 |

### *Output Parameter Details*

Applies to:

- STP-DRV-4850
- STP-DRV-80100
- STP-MTRD-17
- STP-MTRD-23

| | |
|---|---|
| ***Parameter #1*** | Optional "Y", output number, output condition<br>NOTE: Including/omitting the optional "Y" has no effect on the execution of the command. |
| ***– units*** | Optional "Y", integer, letter |
| ***– range*** | - Integer: 1<br>- Letter: L = Low, H = High |

Applies to:

- STP-MTRD-24

Drives with Variable I/O allow a user to configure I/O1 through I/O4 as either inputs or outputs by using the Set Direction (SD) command.

| | |
|---|---|
| ***Parameter #1*** | Optional "Y", output number, output condition<br>NOTE: Including/omitting the optional "Y" has no effect on the execution of the command. |
| ***– units*** | Optional "Y", integer, letter |
| ***– range*** | - Integer: 1 - 4<br>- Letter: L = Low, H = High, F - Falling Edge, R = Rising Edge |

# Appendix D: Host Serial Connections

## *Introduction*

When communicating to a drive over its serial port, you will always be using one of the following serial connections: RS-232, 2-wire RS-485, or 4-wire RS-485. Out of the box we suggest starting with RS-232 along with the programming cable and software that was supplied with your drive so that you may familiarize yourself and communicate with your drive as quickly as possible.

If your application calls for a serial host controller (PC, PLC, HMI, or other serial device that can act as a host) being able to communicate to the drive(s), you will need to choose one of the three available serial connections.

## *Available Host Serial Connections: RS-232, 2-wire RS-485, 4-wire RS-485*

When choosing the best serial connection for your project, the choice may be made for you based on the host controller you plan to use. For example, some devices only communicate via 2-wire RS-485. If you are not restricted by your host controller, here are two guidelines for choosing the best connection.

### *Single or Multi-axis*

If your project calls for communicating to only one drive you can consider any of the three options. If your project calls for communicating to more than one drive, you should use 2-wire or 4-wire RS-485.

### *Long Communication Cables*

In many applications, the limitation of 50 feet on RS-232 will be sufficient. In applications where the distance between drive and host controller will be more than 50 feet (up to 1000 feet), you will need to choose 2-wire or 4-wire RS-485.

## *A Quick Summary of 2-wire and 4-wire RS-485 Connections*

The 2-wire and 4-wire RS-485 protocols that the drives utilize are based on industry standard RS-485 and RS-422 protocols. Strictly defined, RS-485 is a 2-wire interface that allows multi-node connections limited to half-duplex serial communications. Up to 32 nodes that both transmit and receive can be connected to one network. On the other hand, RS-422 in the strictest definition is a 4-wire point-to-point connection that allows full-duplex serial communications when connected to a single node. RS-422 has one node that is the driver or transmitter and up to 10 nodes that are receivers.

2-wire interfaces require one more significant feature. A network mode, master or slave, must be able to tri-state its transmitter to allow other nodes to use the network when required. For high speed baud rates, this must be done very quickly to avoid communication collisions.

4-wire interfaces can go beyond simple point-to-point communications and be used in multi-node networks if the slave nodes are capable of tri-stating their transmitters as required in the 2-wire networks. Some RS-485 devices (like AutomationDirect drives) are set up to do this and can be used in a 4-wire, multi-node configuration.

The drives are designed to work in a multi-node environment, and so they use both the standard 2-wire RS-485 connection, and a modified RS-422 (4-wire) connection that has been termed “4-wire RS-485”.

*NOTE: We recommend using half-duplex communications with the drives. Although the 4-wire RS-485 network can support full-duplex, with multiple nodes data collisions might occur. Limiting communications to half-duplex avoids this. 4-wire is required when using SureMotion Pro.*

### COM Port Settings

When using SureMotion Pro software to communicate to a drive there is no need to worry about COM port settings because the software will take care of them. In applications where a host serial controller will be communicating to a drive via one of it's serial ports, the COM port settings should be set as follows: 8 data bits, no Parity, 1 stop bit. The default Baud rate is 9600, though this can be changed (see BR and PB commands).

### Connecting to a PC using RS-232

Each drive comes with a programming cable for use with the drive's RS-232 port. This cable is made up of two parts, a 7 foot 4-wire cable (looks just like a 7 foot telephone cord), and an RJ11 to 9-pin DSUB adapter. This adapter allows you to connect to the COM port (serial port) of your PC. Here are the general directions for connecting your drive to your computer.

- Locate your computer within 6 feet of the drive.
- Plug the 9-pin end of the adapter supplied with your drive to the COM1 serial port of your PC. Secure the adapter with adapter's two screws. If the COM1 port on your PC is already used by something else, you may use the COM2 port of your PC. On some PCs, COM2 will have a 25-pin connector rather than a 9-pin. If this is the case with your PC, and you must use COM2, you will have to purchase a 25 to 9 pin serial adapter at your local computer store.

*NOTE: If you are using a laptop computer that does not have any COM ports, you will have to use a USB to Serial adapter (ADC part number USB-RS232-1). There are a variety on the market, and some work better than others. But in general, once you've installed one of the adapters your PC will assign the adapter a COM port number. Remember this number when you go to use SureMotion Pro. Also, if you are having troubles with your adapter, contact AutomationDirect for help with recommended adapters.*

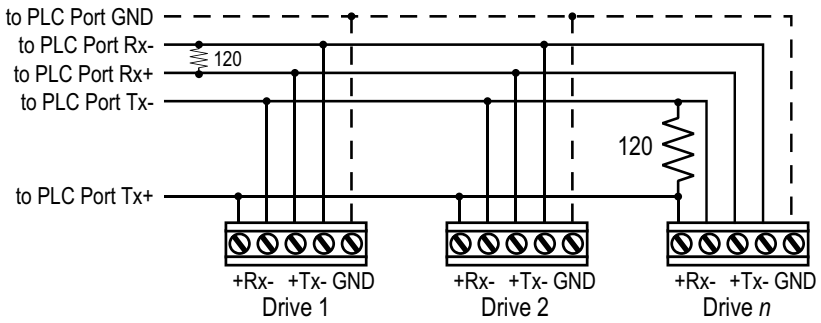
- Now take the 7 foot cable and plug one end into the adapter you just attached to your PCs COM port, and plug the other end into the RS-232 (RJ11) jack on the drive. If you need to locate your drive farther from the PC, you can replace the 7 foot cable with any 4-wire telephone cord. Do not exceed 50 feet.

*WARNING: Never connect an AutomationDirect drive to a telephone circuit. It uses the same connectors and cords as telephones and modems, but the voltages are not compatible and will damage the drive.*

### *Connecting to a Host using 4-wire RS-485*

An AutomationDirect drive's 4-wire RS-485 implementation is a multi-drop network with separate transmit and receive wires. One pair of wires connects the AutomationDirect PLC port's TX+ and TX- signals to each drive's RX+ and RX- terminals. Another pair connects the RX+ and RX- signals of the PLC port to the TX+ and TX- terminals of each drive. A common ground terminal is provided on each drive and can be used to keep all drives at the same ground potential. This terminal connects internally to a drive's ground connection, so if all the drives on the 4-wire network are powered form the same supply it is not necessary to connect the logic grounds. You should still connect one drive's GND terminal to the PLC port's signal ground. Before wiring the entire system you'll need to connect each drive individually to the PLC port so that a unique address can be assigned to each drive. (See following sub-section "Before you connect the drive to your system"). Proceed as follows, using the figure below.

1. Connect the drive TX+ to the host RX+.
2. Connect the drive TX- to the host RX-.
3. Connect the drive RX+ to the host TX+.
4. Connect the drive RX- to the host TX-.
5. Connect GND to the host signal ground.
6. We recommend a 120 ohm terminating resistor be connected between the Rx+ and Rx- terminals of the drive farthest from the host and at the PLC port's Rx+ and Rx- terminals.




*NOTE: Proper cable shielding is a must. High voltage, high frequency, high current signals that are present on the motor cables can emit a significant amount of electrical interference. Without proper shielding on the communications wiring this interference can disrupt even noise-tolerant differential line drivers.*

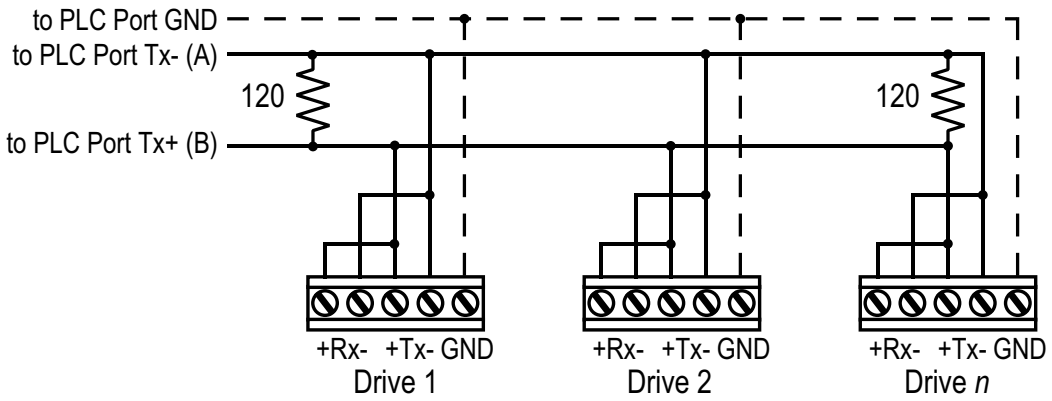
### *Connecting an RS-485 4-wire Adapter to your PC*

AutomationDirect recommends the STP-USB485-4W adapter for RS-485 communications between your drive and a PC. To connect it to PC, follow the steps below:

1. Connect the STP-USB485-4W to your computer's USB port using the 3 foot USB A to USB B cable that came with the STP-USB485-4W. If Windows recognizes the STP-USB485-4W and successfully installs the driver, skip Step 2. Otherwise, disconnect the adapter from your PC and proceed to Step 2.
2. Install driver software located here (www.automationdirect.com/pn/STP-USB485-4W) and follow the onscreen instructions.
3. After your PC has finished installing the STP-USB485-4W, use the Device Manager on your PC to set and/or note the COM port that your PC automatically assigned to the STP-USB485-4W. This is needed for port identification in Step 6. If you disconnect/disable all other COM ports on your PC, you can skip this step.
4. Disconnect the STP-USB485-4W from your PC. The STP-USB485-4W receives its power from your PC's USB port, and it is best practice to remove power from any hardware before configuring and wiring it.
5. Configure and wire the STP-USB485-4W for 4-wire RS-485 (recommended for STP-MTRD models) or 2-wire RS-485 communications. If connecting to an STP-DRV drive then the serial cable included with the drive will work.
6. Reconnect the STP-USB485-4W to your PC's USB port and connect the STP-USB485-4W to the drive using the STP-485DB9-CBL-2. Launch the SureMotion Pro software on your PC and select the COM port to which the STP-USB485-4W is connected. Finally, power up the drive and begin communicating.

### *Connecting to a Host Using 2-wire RS-485*

An AutomationDirect drive's 2-wire RS-485 implementation is a multi-drop network with one pair of wires that is used for both transmit and receive. To make this type of connection you will first need to jumper the TX+ terminal of a drive to it's own RX+ terminal, and then do the same with the TX- and RX- terminals. To then connect a drive to an AutomationDirect PLC port, you will need to connect the TX+/RX+ terminals of the drive to the PLC port's TX+/RX+ terminal, and then the TX-/RX- terminals of the drive to the PLC port's TX-/RX- terminal. We also recommend a 120 terminating resistor be connected between the Tx+ and Tx- terminals of the drive farthest from the host. Here is a diagram:



### *Before you connect the drive to your system*

If you plan to implement a 2-wire or 4-wire RS-485 network of drives, you will first need to address each drive individually. An easy way to do this is prior to hooking the drives up with one of the RS-485 implementations shown above, use the RS-232 cable that came with each drive and the SCL Setup Utility. If you've already connected your drive using one of the RS-485 implementations, completing this sub-section will allow you to test your connections.

First connect your PC and drive. (See preceding sub-sections on connecting to a PC or host for help with this). Then launch SureMotion Pro.

Once SureMotion Pro is launched, select the proper COM port of your PC, and then apply power to the drive. Press the Caps Lock key on your keyboard (because the drives only accept commands in **UPPERCASE**). Type RV then press Enter. If the drive has power and is properly wired, it will respond with "RV=x", where x is the firmware version of your drive. This confirms that communication has been established. If you don't see the "RV=x" response, check your wiring and follow the above procedures again.

Next, you must choose an address for each drive. Any of the "low ascii" characters (many of which appear above the number keys on a PC keyboard) are acceptable:

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < > ? @

To find out which address is already in your drive, type DA then press Enter. The drive will respond with "DA=x", where x is the address that was last stored. To change the address, type "DAy", where y is the new address character, then press Enter.

To test the new address, type "yRV" where y is the address you've just assigned to the drive, and then press Enter. For example, if you set the address to % and want to test the address, type "%RV" then press Enter. The drive should respond with "%RV=x" where x is the firmware version of the drive.

Once each drive in your network has been given a unique address, you can proceed with wiring the whole network together.

# Appendix E: The PR Command

Because of the intense nature of serial communications required in host mode applications, you are allowed to adjust a drive's serial communications protocol to best fit your application. This adjusting of a drive's serial communications protocol is done using the PR command.

Typically the PR command is used one time when configuring a drive and saved as part of the startup parameters (use SA command to save startup parameters). However, it can be changed at any time to dynamically alter the serial communications.

The PR command works by sending the decimal equivalent of a 9-bit binary "word". Each bit in the word represents a different setting of the serial communications protocol. These settings are additive, meaning when you set a bit to "1", or turn it on, you are adding the functionality of that setting to the serial protocol. Think of this 9-bit word as a bank of 9 dip switches. You can turn each dip switch on or off, and in doing so add or subtract a particular setting from the overall protocol.

### *The PR command in detail*

Remember that when you use the PR command the parameter that you send along with the command code (PR) is the decimal equivalent of this binary word. Below are the details of each of the bits and the settings they are assigned to.

### *Bit 0 - Default ("Standard SCL")*

PR cannot be set to 0, so if no other bits in the PR word are set to 1 then at least bit 0 must be set to 1. Setting Bit 0 to 1 when any other bits are also set to 1 has no effect on the communications protocol. For example, PR4 (bit 2 set to 1) is the same as PR5 (bits 0 and 2 set to one). With only bit 0 set to 1, when commands that do not request returned data are received by the drive no other response is sent from the drive. In other words, the drive will only send a response to commands that require a response.

*Send data Examples:*

| Command | Drive Sends | Notes |
|---|---|---|
| DI8000 | - | Global set distance to 8000 counts or steps |
| 1DI8000 | - | Drive with address "1" set distance to 8000 counts or steps |

*Request data Examples:*

| Command | Drive Sends | Notes |
|---|---|---|
| DI | DI=8000 | Global distance request |
| 1DI | 1DI=8000 | Drive with address "1" responds with distance |

### *Bit 1 - Address Character (always send address character)*

With this option set (Bit 1=1) a drive's address character will always be included in the response packet along with any requested data.

***Send data Examples:***

| Command | Drive Sends | Notes |
|---|---|---|
| VE50 | - | Global set velocity to 50 rps |
| 1VE50 | - | Drive with address "1" set velocity to 50 rps |

***Request data Examples:***

| Command | Drive Sends | Notes |
|---|---|---|
| VE | 1VE=50 | Drive responds with address "1" and velocity to global velocity request |
| 1VE | 1VE=50 | Drive responds with address "1" and velocity to specific velocity request from drive at address "1" |

### *Bit 2 - Ack/Nack (always send acknowledge character)*

This option causes the drive to acknowledge every transmission from a host, whether the command is requesting data or not. If a host requests data (for example a DI command with no parameter), the response is considered the acknowledgement. However, if the host sends commands that do not request data from the drive, the drive will still respond with one of the following characters:

"%" - The "percent" character is a Normal Acknowledge (Ack) character that means the drive accepted the command and executed it.

"*" - The "asterisk" character is an Exception Acknowledge (Ack) character that means the drive accepted the command and buffered it into the queue. Depending on the status of the queue, execution of the exception acknowledged command(s) can occur at any time after the acknowledge.

"?" - The "question mark" character is a Negative Acknowledge (Nack) character that means a parsing error occurred while the drive was receiving the command. A second character may follow the question mark, which provides an error code describing the type of parsing error. Here is the list of error codes:

***Negative Acknowledge Codes***

| | |
|---|---|
| 1 | Command timed out |
| 2 | Parameter is too long |
| 3 | Too few parameters |
| 4 | Too many parameters |
| 5 | Parameter out of range |
| 6 | Command buffer (queue) full |
| 7 | Cannot process command |
| 8 | Program running |
| 9 | Bad password |
| 10 | Comm port error |
| 11 | Bad character |

12 I/O point already used by current Command Mode, and cannot be changed (I/O drives only)

13 I/O point configured for incorrect use (i.e., input vs. output) (I/O drives only)

14 I/O point cannot be used for requested function - see User manual for possible I/O function assignments. (I/O drives only)

Acknowledge characters are always sent out of the RS-232 port. When operating on a 2-wire or 4-wire RS-485 network, the acknowledge characters are sent out under the following conditions:

1. An acknowledge character is sent when the received command has an address character at the beginning.
2. An acknowledge character is NOT sent when global commands (commands without addresses) that do not request data from the drive are used.
3. Global commands that request data will cause data to be returned from the drive(s). This can cause data collisions if there are more than one drive on a network. NOTE: Always use addresses with commands in multi-drop networks to avoid data collisions.

*NOTE: When possible avoid using Acknowledge characters (%, *, ?) as drive addresses in multi-drop networks to prevent confusion.*

***Good command Example:***

| Command | Drive Sends | Notes |
|---|---|---|
| DI8000 | % | Drive sends normal Ack (over RS-232 port only) in response to global set distance to 8000 |
| 1DI8000 | 1% | Drive at address “1” sends normal Ack (over both ports) in response to address-specific set distance to 8000 |

***Bad command Example:***

| Command | Drive Sends | Notes |
|---|---|---|
| VE200 | ?5 | Drive sends Nack (over RS-232 port only) in response to global set velocity to 200 rps; error code 5 is sent because parameter “200” is out of range |
| 1VE200 | 1?5 | Drive at address “1” sends Nack (over both ports) and error code in response to address-specific set velocity to 200 rps |

***Buffered command Example:***

| Command | Drive Sends | Notes |
|---|---|---|
| AC10 | * | Drive sends Exception Ack (over RS-232 port only) in response to global set acceleration to 10 rps/s |
| 1AC10 | 1* | Drive at address “1” sends Exception Ack and address (over both ports) in response to address-specific set acceleration |

***Bit 3 - Checksum***

When this bit is 1, checksum is implemented. When this bit is 0, checksum is not implemented.

***Bit 4 - Reserved***

***Bit 5 - Reserved***

***Bit 6 - Reserved***

***Bit 7 - Reserved***

***Bit 8 - Full Duplex in RS-485 (4-wire)***

When this bit is 1, Full Duplex is used. When this bit is 0, Half Duplex is used.

# Appendix F: Troubleshooting

This Appendix addresses potential issues that may occur while using AutomationDirect equipment.

*NOTE: Every drive must be configured with SureMotion Pro software prior to operation. It is never safe to assume that the configuration state of the drive is known when it is received. This step should not be considered optional.*

| Troubleshooting | | |
|---|---|---|
| ***Error Message / Indication*** | ***Explanation*** | ***Solution*** |
| While streaming commands to the drive, it behaves erratically or does not send legible ACK / NACK responses. | The drive's command buffer may be full, which may cause unpredictable behavior. | It is recommended that the user receive and process the drive's ACK / NACK character before sending the next command. This will ensure that the drive's command buffer never overflows and the drive behaves normally.<br><br>If this is not possible, a delay should be introduced between commands that are streamed to the drive. A delay of approximately 10ms should be sufficient for all commands that do not cause motion. |
| "The drive is not responding. Is it connected to the right port and turned on?" | The software is unable to communicate to the drive. There are four common causes for this error:<br><br>1 - The drive is not powered.<br><br>2 - The software is using the wrong COM port.<br><br>3 - The drive was already running before the software was launched. (wrong power-up sequence)<br><br>4 - The USB/Serial converter is faulty or not supported by ADC. If an onboard 9-pin COM port is not available, use a USB/Serial converter based on the FTDI chipset. The chipset used will be shown on the converter's documentation.<br><br>Hint: If communications have been established, ADC software will display the drive's firmware revision along with the model number. If this box is empty, communications have not been established. | 1 - Apply power to the drive.<br><br>2 - Physical 9-pin COM ports are typically assigned COM1 or COM2. USB adapters are often assigned arbitrary COM port identifiers. Check your computer's hardware settings in the Control Panel to verify which COM port your device is using.<br><br>3 - Ensure that the software is running and using the correct COM port. Then, cycle power on the drive. This will allow the software to intercept the drive's power-up packet (as detailed in Appendix B) and initiate communications. |
| "You have not set the load inertia in the Motor Settings. The electronic damping and anti-resonance will work better if you set the load inertia accurately. Do you want to download your settings anyway?" | The drive is missing important information used to properly configure the anti-resonance features. The motor will run without this information, but it may not be as smooth as it otherwise could be. This is generally acceptable only for initial testing, and should be addressed before normal operation. | Set the load inertia. Depending on the configuration software used, it is either possible to enter the actual calculated load inertia or a best-guess estimate of the inertia ratio (load : motor). For example, if the load inertia is five times that of the motor's rotor, the ratio would be entered as 5 : 1. |

| Troubleshooting | | |
|---|---|---|
| ***Error Message / Indication*** | ***Explanation*** | ***Solution*** |
| Drive's LED blinks red and green | An alarm or fault condition exists. The display consists of a specific number of red and green blinks, and will repeat continuously until resolved. | Fault codes are drive-dependent. Consult Appendix B and your drive's hardware manual for specific information. |
| Drive's LED shows solid red | A firmware download was interrupted, and the drive is unable to boot properly. | Cycle power on the drive and repeat the firmware download process. |

SCL User Manual
for STP-DRV & STP-MTRD Drives

SCL Commands for the STP-DRV-4850 and STP-DRV-80100 Step Motor Drives
and the STP-MTRD-17R, STP-MTRD-23R, STP-MTRD-24R Integrated Motor/Drives

AUTOMATIONDIRECT.com