Host Command Reference

Q and SCL commands for servo and stepper drives

APPLIED MOTION PRODUCTS, INC.

920-0002 Rev. D
7/15/2007

(this page intentionally left blank)

Contents

Introduction 9

What is SCL? 9
What is Q? 9
The SCL Communications Protocol 9

Getting Started 11

BLu Servo Drives 11
SV Servo Drives 11
STAC6 Servo Drives 11
ST-Q/Si Stepper Drives 11
ST-S Stepper Drives 12
STM Series Drive+Motors 12

Commands 13

Buffered Commands 13
Stored Programs in Q Drives 13
Multi-tasking in Q Drives 13
Immediate Commands 13
Using Commands 13
Commands in Q drives 14
SCL Setup Utility software 14
Command Summary 16
Motion Commands 17
Configuration Commands 18
I/O Commands 19
Communications Commands 19
Q Program Commands 20
Drive Commands 20
Register Commands 21
Immediate Commands 21
Command Listing 23
AC - Acceleration Rate 24
AD - Analog Deadband 25
AF - Analog Filter 26
AG - Analog Velocity Gain 27
AI - Alarm Reset Input 28
AL - Alarm Code 30
AM - Max Acceleration 31

AO - Alarm Output 32
AP - Analog Position Gain 34
AR - Alarm Reset (Immediate).....35
AS - Analog Scaling 36
AT - Analog Threshold.....37
AV - Analog Offset Value.....38
AX - Alarm Reset (Buffered) 39
AZ - Analog Zero.....40
BD - Brake Disengage Delay 41
BE - Brake Engage Delay 42
BO - Brake Output 43
BR - Baud Rate 45
BS - Buffer Status 46
CC - Change Current.....47
CD - Idle Current Delay Time.....49
CE - Communication Error.....50
CF - Anti-resonance Filter Frequency 51
CG - Anti-resonance Filter Gain.....52
CI - Change Idle Current.....53
CJ - Commence Jogging 55
CM - Command Mode (AKA Control Mode) 56
CP - Change Peak Current 58
CR - Compare Registers 59
CS - Change Speed.....60
CT - Continue.....61
DA - Define Address 62
DC - Change Distance 63
DE - Deceleration.....64
DI - Distance/Position 65
DL - Define Limits 66
DR - Data Register for Capture.....68
EF - Encoder Function 69
EG - Electronic Gearing.....70
EP - Encoder Position 71
ER - Encoder Resolution 72
FC - Feed to Length with Speed Change 73
FD - Feed to Double Sensor 74
FE - Follow Encoder 76
FI - Filter Input.....78

FL - Feed to Length 81
FM - Feed to Sensor with Mask Distance 82
FO - Feed to Length and Set Output 84
FP - Feed to Position 86
FS - Feed to Sensor.....87
FX - Filter select inputs 89
FY - Feed to Sensor with Safety Distance 90
GC - Current Command.....92
HG - 4th Harmonic Filter Gain 93
HP - 4th Harmonic Filter Phase 94
HW - Hand Wheel 95
Immediate Status Commands.....97
IA - Immediate Analog 98
IC - Immediate Current (Commanded) 99
ID - Immediate Distance 100
IE - Immediate Encoder 101
IF - Immediate Format 102
IH - Immediate High Output 103
IL - Immediate Low Output.....105
IO - Output Status 107
IP - Immediate Position 109
IQ - Immediate Current (Actual).....110
IS - Input Status 111
IT - Immediate Temperature.....114
IU - Immediate Voltage.....115
IV - Immediate Velocity 116
IX - Immediate Position Error.....117
JA - Jog Acceleration 118
JC - Velocity (Oscillator) mode second speed 119
JD - Jog Disable.....120
JE - Jog Enable.....121
JL - Jog Decel.....122
JM - Jog Mode 123
JS - Jog Speed 124
KC - Overall Servo Filter 125
KD - Differential Constant 126
KE - Differential Filter.....127
KF - Velocity Feedforward Constant.....128
KI - Integrator Constant.....129

KK - Inertia Feedforward Constant 130
KP - Proportional Constant 131
KV - Velocity Feedback Constant 132
KW - Velocity Filter 133
LV - Low Voltage threshold 134
MD - Motor Disable 135
ME - Motor Enable 136
MN - Model Number 137
MO - Motion Output 138
MT - Multi-Tasking 140
MV - Model & Revision 141
NO - No Operation 143
OF - On Fault 144
OI - On Input 145
OP - Option board 147
PB - Power-up Baud Rate 148
PC - Power-up Current 149
PF - Position Fault 150
PI - Power-up Idle Current 151
PL - Position Limit 152
PM - Power-up Mode 153
PP - Power-up Peak current 154
PR - Protocol 155
PS - Pause 156
PW - Password 157
QC - Queue Call 158
QD - Queue Delete 159
QE - Queue Execute 160
QG - Queue Goto 161
QJ - Queue Jump 162
QK - Queue Kill 163
QL - Queue Load 164
QR - Queue Repeat 165
QS - Queue Save 166
QU - Queue Upload 167
QX - Queue Load & Execute 168
RC - Register Counter 169
RD - Register Decrement 171
RE - Restart or Reset 172

RI - Register Increment....173
RL - Register Load - immediate....174
RM - Register Move....175
RR - Register Read....176
RS - Request Status....177
RU - Register Upload....178
RV - Revision Level....179
RW - Register Write....180
RX - Register Load - buffered....181
R+ - Register Add....182
R- - Register Subtract....183
R* - Register Multiply....184
R/ - Register Divide....185
R& - Register AND....186
RI - Register OR....187
SA - Save Parameters....188
SC - Status Code....189
SF - Step Filter Frequency....190
SH - Seek Home....191
SI - Enable Input Usage....193
SJ - Stop Jogging....195
SK - Stop & Kill....196
SM - Stop Move....197
SO - Set Output....198
SP - Set Position....200
SS - Send String....201
ST - Stop....202
TD - Transmit Delay....203
TI - Test Input....204
TR - Test Register....206
TS - Time Stamp....207
VC - Velocity Change....208
VE - Velocity....209
VI - Velocity Integrator Constant....210
VM - Maximum Velocity....211
VP - Velocity Mode Proportional Constant....212
WD - Wait Delay....213
WI - Wait for Input....214
WM - Wait on Move....216

WP - Wait Position 217
WT - Wait Time 218
ZC - Regen Resistor Continuous Wattage 219
ZR - Regen Resistor Value 220
ZT - Regen Resistor Peak Time 221

Data Registers 222

Read-Only data registers 222
Read/Write data registers 222
User-Defined data registers 222
Storage data registers 222

Using Data Registers 223

Loading (RL, RX) 223
Uploading (RL, RU) 223
Writing Storage registers (RW) *(Q drives only)* 224
Reading Storage registers (RR) *(Q drives only)* 224
Moving data registers (RM) *(Q drives only)* 224
Incrementing/Decrementing (RI, RD) *(Q drives only)* 224
Counting (RC, "I" register) *(Q drives only)* 224
Math & Logic (R+, R-, R*, R/, R&, RI) *(Q drives only)* 224
Conditional Testing (CR, TR) *(Q drives only)* 224

Data Register Assignments 225

Read-Only data registers: a - z 225
Read/Write data registers: A - Z 229
User-Defined data registers: 0 - 9, other characters 232

Appendices 234

Appendix A: Non-Volatile Memory in Q drives 235
Appendix B: Host Serial Communications 236
Appendix C: Host Serial Connections 240
Appendix D: The PR Command 244
Appendix E: Alarm and Status Codes 248
Appendix F: Working with Inputs and Outputs 255

Introduction

Thank you for your interest in Applied Motion Products. We hope you will find that performance, price, and ease of use make our products the best value for your application.

The Host Command Reference is a manual for Q and SCL commands available with DSP-based motor drives from Applied Motion. This includes the BLu and SV series servo drives, the STAC6, ST-Q/Si, and ST-S series stepper drives, and the STM series drive+motors.

What you'll find in this manual is a detailed listing of the Commands and Data Registers used by these drives. There are some helpful tips in getting started with these drives (see Getting Started section), as well as some important topics when working with these drives, which are contained in the Appendices at the end of the manual.

Our goal is to provide you with powerful yet easy-to-use products that will make your project a success. If you need assistance with your drive you can contact our technical support at 800-525-1609, or visit us on the web at www.applied-motion.com/contact/.

What is SCL?

SCL, or Serial Command Language, was developed by Applied Motion 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 control signals, like Pulse & Direction or +/-10V signals, to your step and servo motor drives. 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.

To use SCL in an application means that 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 available commands includes commands for motion of the motor, commands for controlling the inputs and outputs of the drive, as well as commands for configuring different aspects of the motor/drive system like motor current and resolution.

Currently, all DSP-based motor drives from Applied Motion have the ability to execute SCL commands sent from a host. There is also a large set of micro controller-based stepper drives from Applied Motion that can run in SCL mode - like the Si5580, 3540i, and Si2035 stepper drives - however SCL commands for these drives are covered in another manual. To download the SCL manual for micro-controller based stepper drives please visit www.applied-motion.com/software/scl.php.

What is Q?

The use of SCL commands with Applied Motion Products dates back many years. A few years ago a new control platform was created that expanded the use of SCL commands and allowed users to create stored programs with SCL commands. These programs could be saved in a drive's non-volatile memory, and the drive could run these programs stand-alone, or without a permanent connection to the host. This expansion of SCL's capabilities was called Q, and since that time Applied Motion has continued to expand the offering of drives with the Q motion controller built in. By combining the ability to run a sophisticated, single-axis motion control program stand-alone and the ability to communicate serially to a host device, Q drives offer a high level of flexibility and functionality to the machine designer and system integrator.

Q drives are always designated with a Q in their model number, such as BLuAC5-Q, STAC6-Q, or ST10-Q.

The SCL Communications Protocol

The term SCL not only designates a set of commands, but also a serial communications network protocol. SCL can be used in single-axis (point-to-point) applications, as well as multi-axis (multi-drop) applications. All of the drives discussed in this manual have RS-232 and/or RS-485 ports available for use with SCL. Some drives have both ports available in their standard configuration, others have one standard port and another optional port.

In a nutshell, host devices send command strings to a drive, and if the drive understands the string it will execute the command. Ack/Nack characters are available to enhance the protocol, along with a few other protocol settings that are set using the PR (Protocol) command.

There are three basic parts to SCL: the commands and data registers available for use by the drives, the drive configuration and protocol settings that are made to tailor the communications to your application, and the physical connection of serial ports between the host and drive(s). Commands and Data Registers are discussed in their respective sections of this manual. Drive configuration and protocol settings, as well as the physical connection of serial ports are discussed in the Appendices of this manual.

Keep this reference close at hand when using host commands with your Applied Motion servo or step motor drive. Whether you are sending SCL commands to an S drive from a PLC or PC, or programming a Q drive with the Q Programmer software application, this reference will be useful to you. It contains detailed descriptions of every host command available for DSP-based motor drives from Applied Motion.

This reference assumes that you are already familiar with wiring, installation, configuration and setup of your motor drive. If not, you should refer to the Hardware or User's Manuals for your particular drive. All of these manuals can be obtained from our web site, www.applied-motion.com, or by calling us at 800-525-1609.

Getting Started

The basic procedures for integrating an Applied Motion drive into your application are the same for every drive offered, with some small differences. For some helpful tips getting started with your specific drive model numbers review the sections below. Remember that all software applications needed for your drive are available for download from our web site at www.applied-motion.com/support/software.php.

BLu Servo Drives

- This series includes all BLuAC5, BLuDC9, and BLuDC4 drives.
- Use *Quick Tuner* software to tune and configure your drive. For SCL applications choose the SCL Operating Mode and for Q applications choose either the SCL or Q Program Operating Mode.
- For SCL applications use *SCL Setup Utility* to initially practice sending SCL commands to your drive.
- For Q applications use *Q Programmer* for both creating stored programs and for sending commands to your drive.
- All BLu series drives can be used in SCL applications, however BLu-Si drives are not recommended for multi-drop communications over the RS-485 port.
- BLu-Q, BLu-QE and BLu-Si drives can be used for Q applications, however for multi-drop communications over the RS-485 port use only BLu-Q and BLu-QE drives.

SV Servo Drives

- This series includes all SV7 drives.
- Use *Quick Tuner* software to tune and configure your drive. For SCL applications choose the SCL Operating Mode and for Q applications choose either the SCL or Q Program Operating Mode.
- For SCL applications use *SCL Setup Utility* to initially practice sending SCL commands to your drive.
- For Q applications use *Q Programmer* for both creating stored programs and for sending commands to your drive.
- All SV series drives can be used in SCL applications, however SV7-Si drives are not recommended for multi-drop communications over the RS-485 port.
- SV7-Q and SV7-Si drives can be used for Q applications, however for multi-drop communications over the RS-485 port use only SV7-Q drives.

STAC6 Servo Drives

- This series includes all STAC6 and STAC6-220 drives.
- Use *STAC6 Configurator* software to define your motor and configure the Motion Mode of the drive. For SCL applications choose the SCL Motion Mode and for Q applications choose either SCL or Q Program Motion Mode.
- For SCL applications use *SCL Setup Utility* to initially practice sending SCL commands to your drive.
- For Q applications use *Q Programmer* for both creating stored programs and for sending SCL commands to your drive.
- All STAC6 series drives can be used in SCL applications, however STAC6-Si drives are not recommended for multi-drop communications over the RS-485 port.
- STAC6-Q, STAC6-QE, and STAC6-Si drives can be used in Q applications, however for multi-drop communications over the RS-485 port use only STAC6-Q and STAC6-QE drives.

ST-Q/Si Stepper Drives

- This series includes all ST5-Q, ST10-Q, ST5-Si, and ST10-Si drives.
- Use *ST Configurator* software to define your motor and configure the Motion Mode of the drive. For SCL applications choose the SCL Motion Mode and for Q applications choose either SCL or Q Program Motion Mode.
- For SCL applications use *SCL Setup Utility* software to initially practice sending SCL commands to your drive.
- For Q applications use *Q Programmer* for both creating stored programs and for sending commands to your drive.
- All ST-Q and ST-Si drives can be used in SCL applications, however for multi-drop communications over the RS-485 port use only ST5-Q and ST10-Q drives.
- All ST-Q and ST-Si drives can be used in Q applications, however for multi-drop communications over the RS-485 port use only ST5-Q and ST10-Q drives.
- All ST-Q and ST-Si drives come standard with an RS-232 port.
- All ST-Q and ST-Si drives can be purchased with an optional RS-485 comm board (ST-485) for use in point-to-point or multi-drop communications, however for multi-drop communications use only the ST5-Q and ST10-Q drives.

ST-S Stepper Drives

- This series includes the ST5-S and ST10-S drives.
- Use *ST Configurator* software to define your motor and configure the Motion Mode of the drive.
- ST5-S and ST10-S drives can only be used in SCL applications.
- ST5-S and ST10-S drives are only available with RS-232 ports, and therefore are suitable for point-to-point communications with a host.
- ST5-S and ST10-S drives are suitable for multi-axis SCL communications with a host if a SiNet Hub is used as a serial router.

STM Series Drive+Motors

- This series includes all STM23x-xxx drives.
- Use *ST Configurator* software to configure the drive for SCL or Q Program Motion Mode.
- For SCL applications use *SCL Setup Utility* software to practice sending SCL commands to the drive.
- For Q applications use *Q Programmer* for both creating Q Programs and sending commands to the drive.
- All STM23 series drives can be used in SCL applications. Use STM23x-xAx for multi-axis applications with a SiNet Hub acting as an SCL router, and STM23x-xRx for multi-drop communications over the RS-485 port.
- STM23Q-xxx series drives can be used in Q applications. Use STM23Q-xAx for multi-axis applications with a SiNet Hub acting as an SCL router, and STM23Q-xRx for multi-drop communications over the RS-485 port.

Commands

There are two types of host commands available: buffered and immediate. Buffered commands are loaded into and executed out of the drive's volatile command buffer, also known as the *queue*. Immediate commands are not buffered: when received by the drive they are executed immediately.

Buffered Commands

After being loaded into the command buffer of a drive, buffered commands are executed one at a time. (See "Multi-tasking in Q Drives" below for an exception to this rule). If you send two buffered commands to the drive in succession, like an FL (Feed to Length) command followed by an SS (Send String) 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 PS (Pause) and CT (Continue), enable the buffer to be loaded and to pause execution until the desired time.

Stored Programs in Q Drives

Stored Q Programs, created with the *Q Programmer* application software, are created by using only buffered commands. The reason is simple. In a stored program, commands are sequenced together because of timing with external events in the application. An immediate command does not obey the timing of a sequence, and therefore cannot be used inside a stored program.

Multi-tasking in Q Drives

Multi-tasking allows for an exception to the "one at a time" rule of buffered commands. The multi-tasking feature of a Q drive allows you to initiate a move command (FL, FP, CJ, FS, etc.) and proceed to execute other commands without waiting for the move command to finish.

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 BS (Buffer Status) command, or the immediate status of digital inputs using the IS (Input Status) 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.

Using Commands

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. The carriage return denotes the end of transmission to the drive. Here is the basic syntax.

XXAB<cr>

In the syntax above, "XX" symbolizes the command itself, which is always composed of two capital letters. "A" symbolizes the first of two possible parameters, and "B" symbolizes the second. Parameters 1 and 2 vary in length, can be letters or numbers, and are often optional. The "<cr>" symbolizes the carriage return which terminates the command string. How the carriage return is generated in your application will depend on your host software.

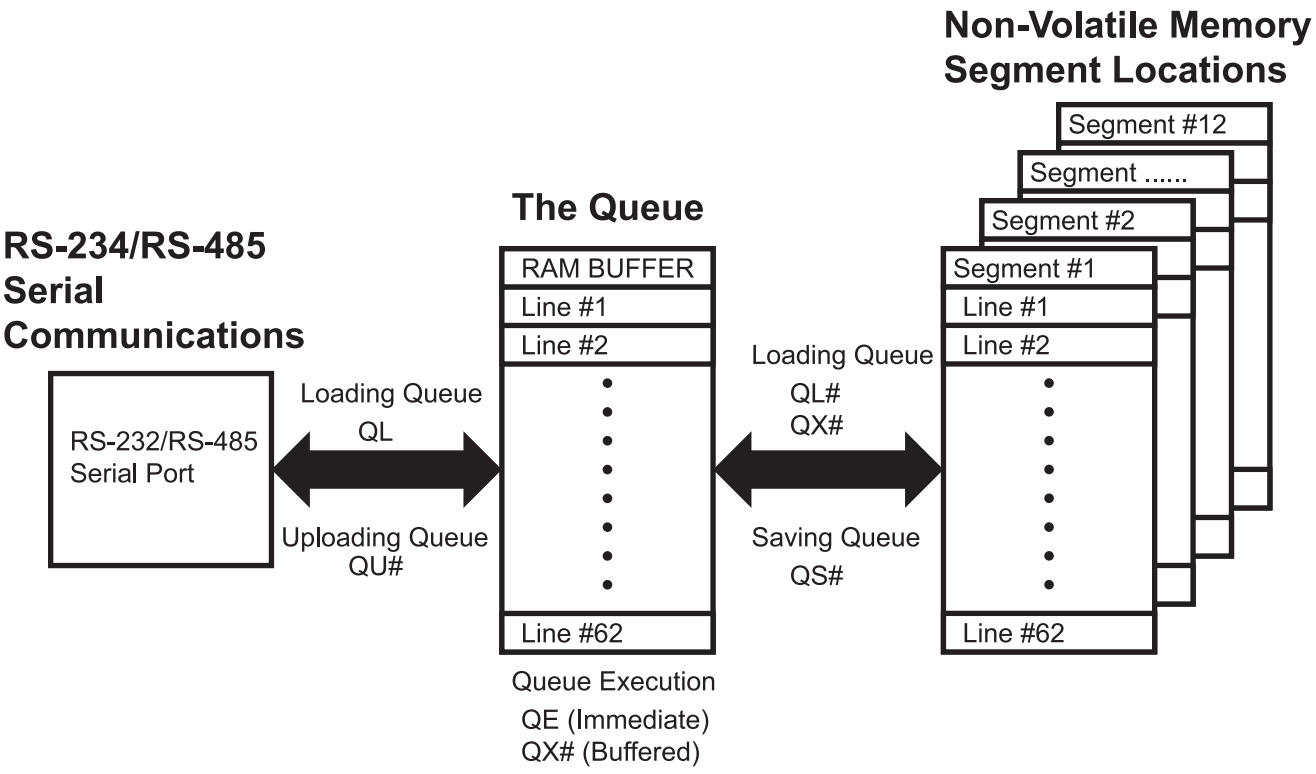
Once a drive receives the <cr> it will determine whether or not it understood the preceding characters as a valid command. If it did understand the command the drive will either execute or buffer the command. If Ack/Nack is turned on (see PR command), the drive will also send an Acknowledge character (Ack) back to the host. The Ack for an executed command is % (percent sign), and for a buffered command is * (asterisk). If the drive did not understand the command it will do nothing. If Ack/Nack is turned on a Nack will be sent, which is signified by a ?

(question mark). The Nack is usually accompanied by a numerical code that indicates a particular error. To see a list of these errors see the PR command details in the Appendix.

Commands in Q drives

Q drives have additional functionality because commands can also be composed into a stored program that the Q drive can run stand-alone. The syntax for commands stored in a Q program is the same as if the commands were being sent directly from the host, or "XXAB". *Q Programmer* software is used to create stored Q programs and can be downloaded for free from www.applied-motion.com/support/software.php.

The diagram below shows how commands sent from the host's serial port interact with the volatile command buffer (AKA the Queue), and the drive's non-volatile program memory storage. Loading and Uploading the Queue contents via the serial port are done with the QL and QU commands, respectively. Similarly, the Queue's contents can be Loaded from NV memory using the QL and QX commands, and can be saved to NV memory with the QS command. Finally, commands currently in the Queue can be executed with the QE or QX command.



The *Q Programmer* software automates many of the functions shown in the diagram above.

SCL Setup Utility software

The *SCL Setup Utility* software is an excellent application for familiarizing yourself with host commands. *SCL Setup Utility* can be downloaded for free from www.applied-motion.com/support/software.php.

To send commands to your drive from *SCL Setup Utility* simply type a command in the Command Line 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 *SCL Setup Utility* does two things: it terminates the command with a carriage return 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 carriage return.

IMPORTANT: 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/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. Then try the above sequence again.

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.

Command Summary

This section contains a set of tables that list all of the Host Commands available with your drive. In each table there are a number of columns that give information about each command.

- "Command" shows the command's two-letter Command Code.
- "Description" shows the name of each command.
- "NV" designates which commands are Non-volatile: that is, which commands are saved in non-volatile memory when the SA (Save) command is sent to the drive.
- "Write only" or "Read only" is checked when a command is not both Read/Write compatible.
- "Immediate" designates an immediate command (all other commands are buffered).
- "Compatibility" shows which drives use each of the commands.

The different categories for these tables - Motion, Configuration, Communications, I/O, Drive, Q Program, Register, and Immediate - are set up to aid you in finding particular commands quickly.

- "Motion Commands" have to do with the actual shaft rotation of the step or servo motor.
- "Configuration Commands" pertain to setting up the drive and motor for your application, including tuning parameters for your servo drive, step resolution and anti-resonance parameters for your step motor drive, etc.
- "Communications Commands" have to do with the configuration of the drive's serial ports.
- "I/O Commands" are used to control and configure the inputs and outputs of the drive.
- "Drive Commands" are generally status and enable/disable commands.
- "Q Program Commands" deal with programming functions when creating stored programs for your Q drive.
- "Register Commands" deal with data registers. Many of these commands are only compatible with Q drives.
- "Immediate Commands" lists all of the available immediate host commands.

Motion Commands

Com-mand	Description	NV	write only	read only	Imme-diate	Compatibility
AC	Accel Rate	•				All drives
AM	Accel Max	•				All drives
CJ	Commence Jogging		•			All drives
DC	Distance for FC, FM, FO, FY	•				All drives
DE	Decel Rate	•				All drives
DI	Distance or Position	•				All drives
EG	Electronic Gearing	•				All drives
EP	Encoder Position					Servos and steppers with encoders
FC	Feed to Length with Speed Change		•			All drives
FD	Feed to Double Sensor		•			All drives
FE	Follow Encoder		•			All drives
FL	Feed to Length		•			All drives
FM	Feed to Sensor with Mask Dist		•			All drives
FO	Feed to Length & Set Output		•			All drives
FP	Feed to Position		•			All drives
FS	Feed to Sensor		•			All drives
FY	Feed to Sensor with Safety Dist		•			All drives
HW	Hand Wheel		•			All drives
JA	Jog Accel/Decel rate	•				All drives
JC	Velocity mode second speed	•				All drives
JD	Jog Disable		•			All drives
JE	Jog Enable		•			All drives
JL	Jog Decel rate	•				All drives
JM	Jog Mode	•				AI drives (see JM command)
JS	Jog Speed	•				All drives
SH	Seek Home		•			All drives
SJ	Stop Jogging		•		•	All drives
SM	Stop the Move		•			Q drives only
SP	Set Absolute Position					All drives
ST	Stop Motion		•		•	All drives
VC	Velocity for Speed Change (FC)	•				All drives
VE	Velocity Setting (For Feed Com-mands)	•				All drives
VM	Velocity Max	•				All drives
WM	Wait on Move		•			Q drives only
WP	Wait on Position		•			Q drives only

Configuration Commands

Com-mand	Description	NV	write only	read only	Imme-diate	Compatibility
BD	Brake Disengage Delay time	•				All drives
BE	Brake Engage Delay time	•				All drives
CC	Change Current	•				All drives
CD	Idle Current Delay	•				Stepper drives only
CF	Anti-resonance Filter Frequency	•				Stepper drives only
CG	Anti-resonance Filter Gain	•				Stepper drives only
CI	Change Idle Current	•				Stepper drives only
CM	Control mode	•				All drives
CP	Change peak current	•				Servo drives only
DA	Define Address	•				All drives
DL	Define Limits	•				All drives
DR	Data Register for Capture		•			Q servo drives only
ER	Encoder or Resolution	•				Servos and steppers with encoders
HG	4th Harmonic Filter Gain	•				Stepper drives only
HP	4th Harmonic Filter Phase	•				Stepper drives only
KC	Servo Filter	•				Servo drives only
KD	Differential Constant	•				Servo drives only
KE	Differential Filter	•				Servo drives only
KF	Velocity Feed Forward Constant	•				Servo drives only
KI	Integrator Constant	•				Servo drives only
KK	Inertia Feed Forward Constant	•				Servo drives only
KP	Proportional Constant	•				Servo drives only
KV	Velocity Feed Back Constant	•				Servo drives only
KW	Velocity Feed Back Filter	•				Servo drives only
MO	Motion Output	•				All drives
OF	On Fault		•			Q drives only
OI	On Input		•			Q drives only
PC	Power up Current	•				All drives
PF	Position Fault	•				Servos and steppers with encoders
PI	Power up Idle Current	•				Stepper drives only
PL	In Position Limit	•				Servo drives only
PM	Power up Mode	•				All drives
PP	Power up peak current	•				Servo drives only
PW	Pass Word		•			Q drives only
SA	Save all NV Parameters		•			All drives
SF	Step Filter Frequency	•				Stepper drives only
SI	Enable Input usage	•				All drives
VI	Velocity Mode Integrator Constant	•				Servo drives only
VP	Velocity Mode Proportional Constant	•				Servo drives only

Configuration Commands (continued)

ZC	Regen Resistor Continuous Wattage	•				BLuAC5 and STAC6 drives only
ZR	Regen Resistor Value	•				BLuAC5 and STAC6 drives only
ZT	Regen Resistor Peak Time	•				BLuAC5 and STAC6 drives only

I/O Commands

Command	Description	NV	write only	read only	Immediate	Compatibility
AD	Analog Deadband	•				All stepper drives and SV servo drives
AF	Analog Filter	•				All drives
AG	Analog Velocity Gain	•				All stepper drives and SV servo drives
AI	Alarm Input usage	•				All drives
AO	Alarm Output usage	•				All drives
AP	Analog Position Gain	•				All drives
AS	Analog Scaling	•				All stepper drives and SV servo drives
AT	Analog Threshold	•				All drives
AV	Analog Offset	•				All drives
AZ	Analog Zero (Auto Zero)		•			All drives
BD	Brake Disengage Delay time	•				All drives
BE	Brake Engage Delay time	•				All drives
BO	Brake Output usage	•				All drives
DL	Define Limits	•				All drives
FI	Filter Input					All drives
IH	Immediate High Output		•		•	All drives
IL	Immediate Low Output		•		•	All drives
IS	Input Status request			•	•	All drives
MO	Motion Output	•				All drives
OI	On Input		•			Q drives only
SI	Enable Input usage	•				All drives
SO	Set Output		•			All drives
TI	Test Input		•			Q drives only
WI	Wait on Input		•			All drives

Communications Commands

Command	Description	NV	write only	read only	Immediate	Compatibility
BR	Baud Rate	•				All drives
PB	Power up Baud Rate	•				All drives
PR	Protocol	•				All drives
TD	Transmit Delay	•				All drives

Q Program Commands

Command	Description	NV	write only	read only	Immediate	Compatibility
MT	Multi-Tasking					Q drives only
OF	On Fault		•			Q drives only
OI	On Input		•			Q drives only
PS	Pause		•			All drives
QC	Queue Call		•			Q drives only
QD	Queue Delete		•			Q drives only
QG	Queue Goto		•			Q drives only
QJ	Queue Jump		•			Q drives only
QK	Queue Kill		•			Q drives only
QR	Queue Repeat		•			Q drives only
QX	Queue Load & Execute		•			Q drives only
NO	No Operation		•			Q drives only
SM	Stop Move		•			Q drives only
SS	Send String		•			All drives
TI	Test Input		•			Q drives only
WD	Wait Delay using Data Register		•			Q drives only
WI	Wait for Input		•			All drives
WM	Wait for Move to complete		•			Q drives only
WP	Wait for Position in complex move		•			Q drives only
WT	Wait Time		•			Q drives only

Drive Commands

Command	Description	NV	write only	read only	Immediate	Compatibility
AL	Alarm Code			•	•	All drives
AX	Alarm Reset (BUFFERED)	•				All drives
EP	Encoder Position					Servos and steppers with encoders
MD	Motor Disable		•			All drives
ME	Motor Enable		•			All drives
MN	Model Number			•	•	All drives
SA	Save Non-Volatile Parameters		•			All drives
SC	Status Code			•	•	All drives

Register Commands

Command	Description	NV	write only	read only	Immediate	Compatibility
CR	Compare Register		•			Q drives only
DR	Data Register for Capture		•			Q drives only
RC	Register Counter		•			Q drives only
RD	Register Decrement		•			Q drives only
RI	Register Increment		•			Q drives only
RM	Register Move		•			Q drives only
RR	Register Read		•			Q drives only
RW	Register Write		•			Q drives only
RX	Register Load (BUFFERED)					Q drives only
R+	Register Addition		•			Q drives only
R-	Register Subtraction		•			Q drives only
R*	Register Multiplication		•			Q drives only
R/	Register Division		•			Q drives only
R&	Register Logical AND		•			Q drives only
RI	Register Logical OR		•			Q drives only
TR	Test Register		•			Q drives only
TS	Time Stamp read		•			Q drives only

Immediate Commands

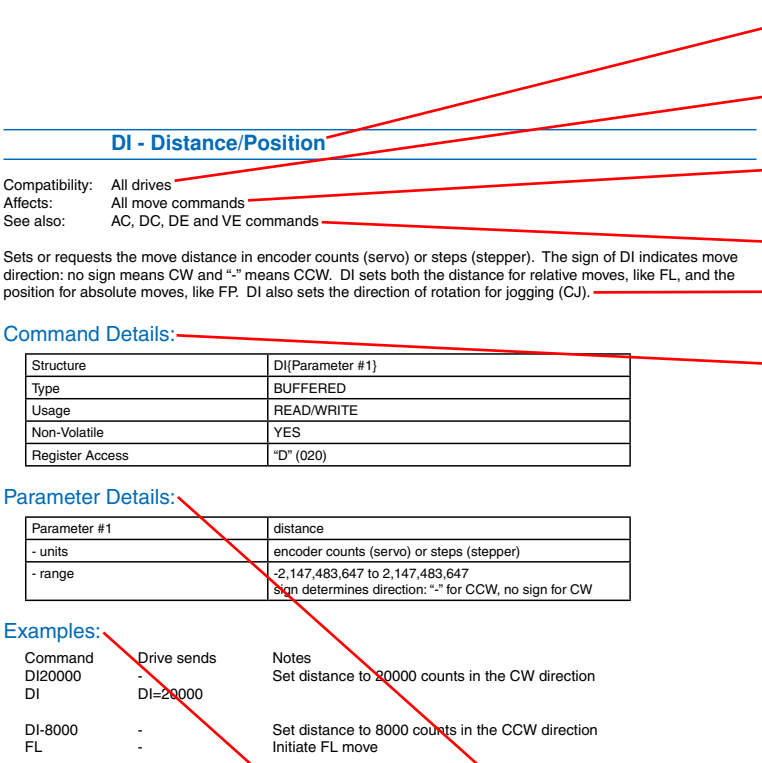
Command	Description	NV	write only	read only	Immediate	Compatibility
AL	Alarm Code			•	•	All drives
AR	Alarm Reset (IMMEDIATE)		•		•	All drives
BS	Buffer Status			•	•	All drives
CE	Comm Error			•	•	All drives
CS	Change Speed	•			•	All drives
CT	Continue		•		•	All drives
GC	Current Command	•			•	Servo drives only
IA	Immediate Analog value			•	•	All drives
IC	Immediate Commanded Current			•	•	Servo drives only
ID	Immediate Distance request, in hex			•	•	All drives
IE	Immediate Encoder request, in hex			•	•	Servos and steppers with encoders
IF	Immediate Format			•	•	All drives
IH	Immediate High output		•		•	All drives
IL	Immediate Low output		•		•	All drives
IO	Immediate output status request				•	All drives
IP	Immediate position request, in hex			•	•	All drives
IQ	Immediate Current value			•	•	Servo drives only

Immediate Commands (continued)

IS	Input Status request			•	•	All drives
IT	Immediate Temperature			•	•	All drives
IU	Immediate DC Bus Voltage			•	•	All drives
IV	Immediate Velocity			•	•	All drives
IX	Immediate Position Error			•	•	Servos and steppers with encoders
MN	Model Number			•	•	All drives
QE	Queue Execute		•		•	Q drives only
QL	Queue Load		•		•	Q drives only
QS	Queue Save		•		•	Q drives only
QU	Queue Upload (to serial port)			•	•	Q drives only
RE	Re-start or Reset		•		•	All drives
RL	Register Load (IMMEDIATE)				•	All drives
RS	Request Status			•	•	All drives
RU	Register Upload		•		•	Q drives only
RV	Revision Level request			•	•	All drives
SC	Request Status in Hexadecimal			•	•	All drives
SJ	Stop Jogging		•		•	All drives
SK	Stop & Kill Buffer / Queue		•		•	All drives
ST	Stop		•		•	All drives

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.



DI - Distance/Position

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

Sets or requests the move distance in encoder counts (servo) or steps (stepper). The sign of DI indicates move direction: no sign means CW and "-" means CCW. 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).

Command Details:

Structure	DI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"D" (020)

Parameter Details:

Parameter #1	distance
- units	encoder counts (servo) or steps (stepper)
- range	-2,147,483,647 to 2,147,483,647 sign determines direction: "-" for CCW, no sign for CW

Examples:

Command	Drive sends	Notes
DI20000	-	Set distance to 20000 counts in the CW direction
DI	DI=20000	
DI-8000	-	Set distance to 8000 counts in the CCW direction
FL	-	Initiate FL move

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

Compatibility - shows which drives use this command.

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

See Also - related commands

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

Command Details - shows the command's Structure, Type, Usage, Non-Volatile status, and Register Access. Structure always shows 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 ( ). Type can be BUFFERED or IMMEDIATE. Usage can be Read Only, Read/Write, or Write Only. Non-Volatile will show if the command can be saved (YES) or not (NO). Saving Non-Volatile commands to memory requires the SA (Save) command. Register Access shows any data registers associated with the command. If the command transfers data to a register that is accessible via the RL and RX commands, that register will be shown here.

Parameter Details - shows a description, the units, and the range of the parameter(s) available with a given command. Some commands will also have a Response Details section which shows how the drive's response to the given command is formatted.

Examples - shows what to expect when you use this command. Under "Command" are the command strings you would send from a host controller or write into a stored program. Under "Drive Sends" are the responses from the drive: no response from the drive is denoted by "-". "Notes" give additional information about the results of the command string.

AC - Acceleration Rate

Compatibility: All drives
Affects: FC, FD, FE, FL, FM, FS, FP, FY, SH commands
See also: AM, DE, DI, DC, VE commands

Sets or requests the acceleration rate used in point-to-point move commands in rev/sec/sec.

Command Details:

Structure	AC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“A” (017) Note: Units of AC command and “A” register are different. See Data Registers section for details of “A” register.

Parameter Details:

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 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 / counts

AD - Analog Deadband

Compatibility: All stepper drives and SV servo drives
Affects: Analog input
See also: CM command

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 “zero”. This zero point can be used as the zero velocity point in analog velocity mode, or as the zero position point in analog position mode (see CM command). The deadband is an absolute value that in usage is applied to either side of zero volts.

Command Details:

Structure	AD{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	Setting the AD command will affect the contents of the “a” (Analog Command) register

Parameter Details:

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

Compatibility: All drives
Affects: All commands using the analog inputs
See also: IA, CM commands

Applies a digital filter to the analog input(s). 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

Command Details:

Structure	AF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	Setting the AF command will affect the responsiveness of the “a”, “j”, and “k” registers to changes in analog voltage

Parameter Details:

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

Compatibility: All stepper drives and SV servo drives
Affects: Analog velocity modes
See also: CM command

Sets or requests the gain value used in analog velocity / oscillator 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 analog input is scaled to 0 - 5 volt input and 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.

Command Details:

Structure	AG{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

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

Compatibility: All drives, see below
Affects: Alarm Reset input usage
See also: AL, CM, DL, SI commands

BLu, SV, STAC6, ST-Q/Si

Defines the function of the X4 input. This input can be used to clear a drive fault and reset the Alarm Code (see AL command). When the Alarm Reset function is not needed at input X4, such as when operating with a host controller where faults and alarms can be cleared via serial commands, it may be useful to reconfigure X4 as a general purpose input, which allows it to be used by other types of input commands.

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

AI1: Alarm reset occurs when the X4 input is closed (energized). 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 (de-energized) and then closed.
AI2: Alarm reset occurs when the input is Open. 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.
AI3: Input is not used for Alarm Reset and can be used as a general purpose input.

ST-S, STM

Defines the EN input as an Alarm Reset Input. The EN input can be assigned to only one function in an application. If you want to use the EN input as an Alarm Reset input you can define it as such in two ways, with the *ST Configurator* software, or with the AI command. 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. Also, 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). In other words, the AI and SI commands, as well as Command Modes (CM) 13, 14, 17 and 18 each assign a usage to the EN input. Each of these must exclusively use the EN input. Command Modes are most dominant and will continually prevent AI and SI from using the input. AI and SI exclude each other by overwriting the usage of the EN input.

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

AI1: Alarm reset occurs when the EN input is closed (energized). 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 (de-energized) and then closed to reset the alarm.
AI2: Alarm reset occurs when the input is open (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.
AI3: The EN Input is not used for Alarm Reset and can be used as a general purpose input. AI will be automatically set to 3 if CM is set to 13, 14, 17, or 18, or if SI is set to 1 or 2 after the AI command is set.

Command Details:

Structure	AI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Input State (see above for details)
- units	integer number
- range	1, 2 or 3

Examples:

Command	Drive sends	Notes
AI1	-	Enables input to reset alarm when closed
AI	AI=1	

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

AL - Alarm Code

Compatibility: All drives
See also: AI, AR, AX commands, Appendix

Reads back an equivalent hexadecimal value of the Alarm Code's 16-bit binary word.

Command Details:

Structure	AL
Command Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	"f" (054) Note: response to AL command is a different format than the response to the RLf command. See Appendix F for details.
Units	Hexadecimal value of 16-bit binary word (see below)

Response Details:

Hex Value	BLu	SV	STAC6	ST	STM
0001	***Position Limit***				
0002	CCW Limit				
0004	CW Limit				
0008	***Over Temp***				
0010	***Excess Regen****	***Internal Voltage***	***Excess Regen***	***Internal Voltage***	***Internal Voltage***
0020	***Over Voltage***				
0040	***Under Voltage****	Under Voltage	***Under Voltage***	Under Voltage	Under Voltage
0080	***Over Current***				
0100	***Bad Hall Sensor***		***Open Motor Winding***		
0200	***Bad Encoder***				(not used)
0400	Comm Error				
0800	Bad Flash				
1000	Wizard Failed		No Move		
2000	Current Foldback		Motor Resistance Out of Range	(not used)	(not used)
4000	Blank Q Segment				
8000	No Move		(not used)		

* BLuAC drives only
NOTE: Items in ***bold italic*** represent Drive Faults, which automatically disable the motor. Use the OF command in a Q Program to branch on a Drive Fault.
NOTE: See Appendix for more detailed information on Alarm Codes.

Examples:

Command	Drive sends	Notes
AL	AL=0000	No alarms
AL	AL=0001	Position limit alarm
AL	AL=0201	Position limit and bad encoder signal alarms

AM - Max Acceleration

Compatibility: All drives
Affects: ST, SK , SM, QK commands; analog velocity and oscillator modes
See also: VM command

Sets or requests the maximum acceleration/deceleration allowed when using analog velocity and oscillator modes. Also sets the deceleration rate used when an end-of-travel limit is activated during a move or when an ST (Stop) or SK (Stop & Kill) command is sent.

Command Details:

Structure	AM{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Maximum acceleration/deceleration
- units	rev/sec/sec (rps/s)
- range	0.167 - 5461.167 (resolution is 0.167 rps/s)

Examples:

Command	Drive sends	Notes
AM2000	-	Set maximum acceleration/deceleration values to 2000 rev/sec/sec.
AM	AM=2000	

AO - Alarm Output

Compatibility: All drives
Affects: Alarm Output usage
See also: AI, BO, MO, SI commands

BLu, SV, STAC6, ST-Q/Si

Defines usage of digital output Y3. Normally this output is used to indicate an Alarm caused by a Drive Fault. This output can being reconfigured as a general purpose output for use with other types of output commands. There are three states that can be defined:

AO1: Output is closed (energized) when a Drive Fault is present.
AO2: Output is open (de-energized) when an Drive Fault is present.
AO3: Output is not used as an Alarm Output and can be used as a general purpose output.

ST-S, STM

Defines the drive's digital output as an Alarm Output. The output of a drive can be assigned to one of four functions: Alarm Output, Brake Output, Motion Output, or Tach 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 ST Configurator or via SCL commands. 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:

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.

AO1: Output is closed (energized) when a Drive Fault is present.
AO2: Output is open (de-energized) 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.

Command Details:

Structure	AO{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Output Usage (see above)
- units	integer code
- range	1, 2 or 3

Examples:

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

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, i.e. the circuit is energized, the logic state for that input/out-*

put 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

AP - Analog Position Gain

Compatibility: All drives
Affects: CM22 (Analog Positioning Command Mode)
See also: AF, AZ, CM commands

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. *Quick Tuner* (BLu, SV), *STAC6 Configurator* (STAC6), or *ST Configurator* (ST, STM) can be used to configure the analog inputs for the desired input type, scaling and offsetting.

Command Details:

Structure	AP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“X” (040)

Parameter Details:

Parameter #1	Analog position gain value
- units	encoder counts
- range	0 - 32767

Examples:

Command	Drive sends	Notes
AP8000	-	Position range over full scale of analog input is 8000 steps
AP	AP=8000	

AR - Alarm Reset (Immediate)

Compatibility: All drives
Affects: Alarm Code
See also: AL, ME, MD commands

Clears Alarms and Drive Faults. If an Alarm or Drive Fault condition persists after sending the AR command the Alarm is not cleared.

NOTE: Does not re-enable the drive. Use ME (Motor Enable) command to re-enable drive.

Command Details:

Structure	AR
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
AR	-	Reset Drive Fault and clear Alarm Code (if possible)

AS - Analog Scaling

Compatibility: All stepper drives and SV servo drives
Affects: Analog input
See also: CM command

Sets or requests the analog input scaling setting. This is a code that determines what type of analog input scaling is desired. The codes for selecting the various settings are in the Details table below.

Command Details:

Structure	AS{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Code
- units	integer number
- range	0 = single-ended +/- 10 volts 1 = single-ended 0 - 10 volts 2 = single-ended +/- 5 volts 3 = single-ended 0 - 5 volts 4 = differential +/- 10 volts 5 = differential 0 - 10 volts 6 = differential +/- 5 volts 7 = differential 0 - 5 volts

Examples:

Command	Drive sends	Notes
AS2	-	Analog input scaling set to single-ended +/- 5 volts
AS	AS=2	

AT - Analog Threshold

Compatibility: All drives
Affects: All “Feed to Sensor” type commands
See also: AF, AZ, FS, FY, FD commands

Sets or requests the Analog Input Threshold that is used by the “Feed to Sensor” command. The threshold value sets the Analog voltage that determines a sensor state or a trigger value.

Command Details:

Structure	AT{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“Y” (041)

Parameter Details:

Parameter #1	Analog threshold value
- units	volts
- range	BLu, SV, STAC6, ST-Q/Si: -10.000 to 10.000 ST-S, STM: 0.000 to 5.000

Examples:

Command	Drive sends	Notes
AT4.5	-	Analog input threshold set to 4.5 volts
AT	AT=4.5	

AV - Analog Offset Value

Compatibility: All drives
Affects: All Analog input functions
See also: AF, AP, AZ, CM & Feed commands

Sets or requests the analog offset value in volts.

Command Details:

Structure	AV{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“Z” (042) Note: Units of AV command are different than units of “Z” register; see Data Registers section for more details

Parameter Details:

Parameter #1	Analog offset value
- units	Volts
- range	BLu, SV, STAC6, ST-Q/Si: -10.000 to 10.000 ST-S, STM: -5.000 to 5.000

Examples:

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

AX - Alarm Reset (Buffered)

Compatibility: All drives
Affects: Alarm Code
See also: ME, AR Commands

Clears Alarms and Drive Faults. If an Alarm or Drive Fault condition persists after sending the AR command the Alarm is not cleared. This command functions the same as AR (Alarm Reset) but is a Buffered type command.

NOTE: Does not re-enable the drive. Use ME (Motor Enable) command to re-enable drive.

Command Details:

Structure	AX
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
AX	-	Reset Drive Fault and clear Alarm Code (if possible)

AZ - Analog Zero

Compatibility: All drives
Affects: All Analog input functions
See also: AF, AP, AV, CM & Feed commands

Activates the analog “auto offset” algorithm. It is useful in defining the current voltage present at the analog input as the zero reference point, or offset.

Command Details:

Structure	AZ
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
AZ	-	Start analog 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 will be very close to IA=3.00 (or 4 - 1 VDC).

BD - Brake Disengage Delay

Compatibility: All drives
Affects: All “F” (Feed) and Jog commands.
See also: BE command

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.

Command Details:

Structure	BD{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Delay time
- units	seconds
- range	0 - 32.767

Examples:

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

BE - Brake Engage Delay

Compatibility: All drives
Affects: All “F” (Feed) and Jog commands.
See also: BD command

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.

Command Details:

Structure	BE{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Delay time
- units	seconds
- range	0 - 32.767

Examples:

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

BO - Brake Output

Compatibility: All drives
Affects: Function of digital output
See also: AI, AO, ME, MD, MO, SI commands

NOTE: The digital output circuits available on Applied Motion 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 appropriate drive hardware manual for an example wiring diagram.

BLu, SV, STAC6, ST-Q/Si

Defines usage of digital output Y1. Normally this output is used to activate or deactivate a motor brake. This output can being reconfigured as a general purpose output for use with other types of output commands. There are three states that can be defined:

BO1: Output is closed (energized) when drive is enabled, and open when the drive is disabled.
BO2: Output is open (de-energized) when drive is enabled, and closed when the drive is disabled.
BO3: Output is not used for Brake control and can be used as a general purpose output.

ST-S, STM

Defines the drive's digital output as a Brake Output. The output of a drive can be assigned to one of four functions: alarm output, brake output, motion output, or tach 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 *ST Configurator* or via SCL commands. To set the output as a brake output, use the BO command and one of the codes below.

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.

There are three Brake Output states that can be defined with the BO command:

BO1: Output is closed (energized) when the drive is enabled, and open when the drives is disabled.
BO2: Output is open (de-energized) 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.

Command Details:

Structure	BO{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Output state (see above)
- units	integer number
- range	1, 2 or 3

Examples:

Command Drive sends Notes

BO1	-	Brake output (Y1) will be Closed when drive is enabled
BO	BO=1	

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

BR - Baud Rate

Compatibility: All drives
Affects: Serial communications
See also: TD, PB, PM, PR commands

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 host system (such as a software application from Applied Motion) is not detected after 1 second and the drive is configured for SCL or Q 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 anytime using this command.

NOTE: Setting the value takes effect immediately.

Command Details:

Structure	BR{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Baud rate (see above)
- units	integer code
- range	1 = 9600 bps 2 = 19200 3 = 38400 4 = 57600 5 = 115200

Examples:

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

BS - Buffer Status

Compatibility: All drives
See also: CT, PS commands

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 a command. If the drive responds with at least a “1”, a command can be sent.

If a drive responds to the BS command with the value “63” it means 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).

Command Details:

Structure	BS
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	Empty command spaces in buffer

Examples:

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

CC - Change Current

Compatibility: All drives
Affects: Motor current and torque
See also: CA, CI, CP, PC commands

BLu, SV

Sets or requests the continuous (RMS) current setting of the servo drive.

STAC6

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 Parameters table below based on the settings defined in the *STAC6 Configurator* software. Use *STAC6 Configurator* to select a motor and set the maximum current setting. Note that setting CC automatically sets CI to the same value if the new CC value is less than the starting CI value.

ST-Q/Si, ST-S, STM

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 Parameters table below based on the settings defined in the *ST Configurator* software. Use *ST Configurator* 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 always set CI after setting CC.

Command Details:

Structure	CC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"N" (030) Note: The CC command uses different units than the "N" register; see Data Registers section for details

Parameter Details:

BLu, SV

Parameter #1	Continuous current setting
- units	amps rms (resolution is 0.01 amps)
- range	BLuDC4: 0 - 4.5 BLuDC9: 0 - 9.0 BLuAC5: 0 - 5.0 SV: 0 - 7.0

STAC6, ST-Q/Si, ST-S, STM

Parameter #1	Running current
- units	amps (resolution is 0.01 amps)
- range*	STAC6: 0 - 6.0 ST5 : 0 - 5.0 ST10: 0 - 10.0 STM: 0 - 5.0

*Current setting in stepper drives depends on the selected motor. Use *Configurator* software to select a motor and set the maximum current setting.

Examples:

BLu, SV

Command	Drive sends	Notes
CC4.50	-	Set continuous current to 4.5 amps rms
CC	CC=4.5	

STAC6

Command	Drive sends	Notes
CC4.50	-	Set running current to 4.5 amps
CI2	-	Set idle current to 2.0 amps
CC1.8	-	Set idle current to 1.8 amps
CC	CC=1.8	
CI	CI=1.8	CI automatically set to 1.8 amps along with CC1.8 command

ST-Q/Si, ST-S, STM

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

Compatibility: Stepper drives only
Affects: Motor current at rest
See also: CC, CI commands

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 & direction mode the drive will reset the idle current delay timer each time a step pulse is received by the drive.

Command Details:

Structure	CD{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

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

Compatibility: All drives
See also: AL command

Requests the hexadecimal equivalent of the communication error's 8-bit binary word. The presence of a comm error will also be shown in the Alarm Code (AL command) as well as the status LEDs at the front of the drive (Appendix F). Bit assignments for the 8-bit word are shown in the Response Details table below.

Command Details:

Command Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None

Response Details:

Response	Communication error code
- units	hexadecimal code
- range	bit 0 = parity flag error bit 1 = framing error bit 2 = noise flag error bit 3 = overrun error bit 4 = Rx buffer full bit 5 = Tx buffer full bit 6 = bad SPI op-code bit 7 = Tx time-out

Examples:

Command	Drive sends	Notes
CE	CE=0010	Rx buffer full
CE	CE=0002	Framing error

CF - Anti-resonance Filter Frequency

Compatibility: Stepper drives only
Affects: Mid-range performance of step motors
See also: CG command

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

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

Command Details:

Structure	CF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Filter frequency
- units	Hz
- range	1 - 2000

Examples:

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

CG - Anti-resonance Filter Gain

Compatibility: Stepper drives only
Affects: Mid-range performance of step motors
See also: CF command

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 appropriate Configurator software application to set this value by entering as accurate a load inertia value as possible in the motor settings window.

Command Structure:

CG{Parameter #1}

Command Details:

Structure	CG{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

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

Compatibility: Stepper drives only
Affects: Motor current at standstill, holding torque
See also: CC, PI, CD commands

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.

STAC6

CI cannot be greater than CC. If you attempt to set CI higher than CC it will be automatically limited to the CC value. Furthermore, setting CC automatically sets CI to the same value if the new CC value is less than the starting CI value.

ST-Q/Si, ST-S, STM

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 be sure to always set CI after setting CC.

Command Details:

Structure	CI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	“O” (031) Note: The CI command uses different units than the “O” register; see Data Registers section for more details

Parameter Details:

STAC6

Parameter #1	Idle current
- units	amps
- range	0 - 100% of running current

ST-Q/Si, ST-S, STM

Parameter #1	Idle current
- units	amps
- range	0 - 90% of running current

Examples:

STAC6

Command	Drive sends	Notes
CI1.0	-	Set idle current to 1.0 amps
CI	CI=1	
CC0.5	-	Set running current to 0.5 amps
CI	CI=0.5	CI automatically set 0.5 amps along with CC0.5 command

ST-Q/Si, ST-S, STM

Command	Drive sends	Notes
CI2	-	Set idle current to 2 amps
CC2	-	Set running current to 2 amps
CI	CI=1	CI automatically set to 1 amp to match 50% of CC2 command
CI1.8	-	Set idle current to 1.8 amps, or 90% of last CC value

CJ - Commence Jogging

Compatibility: All drives
See also: JS, JA, JL, SJ, CS and DI commands.

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 SJ (Stop Jogging) 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, stall, or fault. The jogging direction is set by the last DI command. Use the CS command to change jog speed and direction while already jogging. CS does not affect JS.

Use in Q Programs (Q drives only)

Within a stored Q program jog moves are most commonly initiated with the CJ command. However, because the SJ and ST commands are immediate type they cannot be used within a Q program to stop the jog move. So the procedure to stop a jog move within a Q program involves both the MT (Multi-tasking) and SM (Stop Move) commands. See Examples below for a sample command sequence.

Command Details:

Structure	CJ
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
JA10	-	Set jog accel to 10 rps/s
JL25	-	Set jog decel to 25 rps/s
JS1	-	Set jog speed to 1 rps
CJ	-	Start jogging with speed set by last JS command
CS10	-	Change jog speed to 10 rps
SJ	-	Stop jogging using decel rate set by last JL command

Sample Q program sequence

MT1	Turn Multi-tasking ON
FI58	Filter input X5 for 8 processor ticks (2 msec)
WIX5L	Wait for input X5 low
CJ	Commence jogging
WIX5H	Wait for input X5 high
SMD	Stop Move using the decel ramp set by JL

CM - Command Mode (AKA Control Mode)

Compatibility: All drives
Affects: Drive mode of operation
See also: PM command

Sets or requests the Command Mode that the drive operates in. For more automated setup of command modes use the appropriate *Configurator* or *Quick Tuner* software application. The most common command mode is Point-to-Point (21), in which all move 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 when the move is finished. 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. For this reason it is suggested that the appropriate Configurator or Quick Tuner software application be used to test specific Command Modes first before changing them in the application using the CM command.

Command Details:

Structure	CM{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“m” (061) Note: Because a drive can change Command Mode on it’s own to complete certain moves, the CM command and the “m” register may not always match.

Parameter Details:

Parameter #1	Command mode
- units	integer code
- range	1 - Commanded Torque (servo only) 2 - Analog Torque (servo only) 7 - Step & Direction 10 - Commanded Velocity (jog mode) 11 - Analog velocity 12 to 18 - (see below) 21 - Point-to-Point 22 - Analog Position

Command Modes 12 to 18 are for stepper drives and SV servo drives only:
NOTE: It is recommended to use Configurator or Quick Tuner software for setting up velocity mode operation.
12 - Analog velocity mode with input X1 as run/stop input
13 - Analog velocity mode with input X5 as speed change input
14 - Analog velocity mode with input X1 as run/stop input and input X5 as speed change input
15 - Velocity mode (JS for speed)
16 - Velocity mode (JS for speed) with input X1 as run/stop input
17 - Velocity mode (JS for speed) with input X5 as speed change input
18 - Velocity mode (JS for speed) with input X1 as run/stop input and input X5 as speed change input

Examples:

Command	Drive sends	Notes
CM2	-	Sets the servo drive to Analog Torque mode, at which time there is a linear relationship between the voltage at the drive's analog input and the motor current.
CM7	-	Sets the drive to Step & Direction input mode, which is used for all digital positioning schemes like Step (Pulse) & Direction, CW/CCW Pulse, and A/B Quadrature. Use the appropriate Configurator or Quick Tuner application to set the proper scheme within this mode.
CM10	-	Sets the drive to Command Velocity, or jog mode, which in practice is very similar to Point-to-Point mode (CM21). When in CM21 and a jog command is issued, like CJ, the drive automatically switches to CM10 during the jog move and then back to CM21 when the jog move is stopped. Conversely, when in CM10 and a feed move is commanded, like FL, the drive automatically switches to CM21 during the move and then back to CM10 when the move is finished. CM10 is most useful with servo drives, and when the JM (Jog Mode) is set to 2. This puts the drive into a jog mode in which position error is ignored. Then, when the motor is at rest the drive acts somewhat like a constant friction device in that a certain amount of torque (set by CC and CP commands) is required to move the shaft.
CM11	-	Sets the drive to Analog Velocity mode. In servo drives this will be similar to the Analog Torque mode, where voltage level at the analog input relates to motor speed. In stepper drives this puts the drive into continuous oscillator mode, with speed set by the JS command.

CP - Change Peak Current

Compatibility: Servo drives only
Affects: Motor current, especially during acceleration and deceleration
See also: CC, PC, PP commands

Sets or requests the peak (RMS) current setting of the servo drive. Peak current sets the maximum current that should be used with a given motor. The servo drive will allow peak current for no more then 1 second. After one second of operation at peak current the current is reduced to the continuous current setting (see CC command). The peak current time calculation is done using I^2/T which integrates current values for more accurate modeling of drive and motor heating.

Command Details:

Structure	CP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"O" (031) Note: The CP command uses different units than the "O" register; see Data Registers section for more details

Parameter Details:

Parameter #1	Peak current limit
- units	amps RMS
- range	BLuDC4: 0 - 13.5 A BLuDC9: 0 - 18.0 A BLuAC5: 0 - 15.0 A SV7: 0 - 14.0 A

Examples:

Command	Drive sends	Notes
CP9.0	-	Peak current is set to 9.0 amps RMS
CP	CP=9.0	

CR - Compare Registers

Compatibility: Q drives only
Affects: Contents of condition code register “h”
See also: RI, RD, RM, RL, QJ commands

Compare the contents of two data registers. The first data register (Parameter #1) is tested by comparing it against the data value in the second data register (Parameter #2). The result is a condition code that can be used for program conditional processing (see QJ command). For Example, if the first data register is greater than the second the “greater than” flag is set and the QJGx command can be used to create a conditional jump.

Command Details:

Structure	CR(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	First data register assignment
- units	character
- range	All data register assignments
Parameter #2	Second data register assignment
- units	character
- range	All data register assignments

Examples:

Command	Drive sends	Notes
CRE1	-	Compare data register “E” to data register “1”
QJG5	-	If “E” register is greater than “1” register jump to line 5 of Q segment, otherwise proceed to next line.

CS - Change Speed

Compatibility: All drives
Affects: Jog speed while jogging
See also: CJ, JS, JA, JL commands

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

Command Details:

Structure	CS{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"J" (026) Note: The CS command uses different units than the "J" register; see Data Registers section for more details.

Parameter Details:

Parameter #1	Jog Speed
- units	rev/sec
- range	BLu, SV, STAC6, ST-Q/Si, ST-S: -133.3333 to 133.3333 (resolution is 0.0042) STM: -80.0000 to 80.0000 (resolution is 0.0042) sign determines direction: "-" for CCW, no sign for CW

Examples:

Command	Drive sends	Notes
JS1	-	Set base jog speed to 1 rev/sec
CJ	-	Commence jogging
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
SJ	-	Stop jogging

CT - Continue

Compatibility: All drives
See also: PS, ST, SK commands

Resume execution of buffered commands after a PS command has been sent. The PS (Pause) 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 resume execution in the order they were received, or until the command buffer is full.

Command Details:

Structure	CT
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
PS	-	Pause command buffer
FL2000	-	CW move, 2000 counts
WT.25	-	Wait 0.25 seconds
FL-2000	-	CCW move, 2000 counts
CT	-	Resume execution of buffered commands

DA - Define Address

Compatibility: All drives
Affects: Drive address for multi-drop communications

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

Command Details:

Structure	DA{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	RS-485 network address
- units	character
- range	Valid address characters are: ! " # $ % & ' ( ) * + , - . / 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

Compatibility: All drives
Affects: FC, FY, FO, FM commands.

Sets or requests the change distance. 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.

Command Details:

Structure	DC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“C” (019)

Parameter Details:

Parameter #1	distance
- units	encoder counts
- range	0 to 2,147,483,647 (the sign of negative values is ignored)

Examples:

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

DE - Deceleration

Compatibility: All drives
Affects: FC, FD, FE, FL, FM, FS, FP, FY, SH commands
See also: AM, DE, DI, DC, VE commands

Sets or requests the deceleration rate used in point-to-point move commands in rev/sec/sec.

Command Details:

Structure	DE{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	“B” (018) Note: The DE command uses different units than the “B” register; see Data Registers section for details

Parameter Details:

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	

DI - Distance/Position

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

Sets or requests the move distance in encoder counts (servo) or steps (stepper). The sign of DI indicates move direction: no sign means CW and "-" means CCW. 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).

Command Details:

Structure	DI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"D" (020)

Parameter Details:

Parameter #1	distance
- units	encoder counts (servo) or steps (stepper)
- range	-2,147,483,647 to 2,147,483,647 sign determines direction: "-" for CCW, no sign for CW

Examples:

Command	Drive sends	Notes
DI20000	-	Set distance to 20000 counts in the CW direction
DI	DI=20000	
DI-8000	-	Set distance to 8000 counts in the CCW direction
FL	-	Initiate FL move

DL - Define Limits

Compatibility: All drives
Affects: All move commands
See also: AM command

CW and CCW end-of-travel limits are available on all drives and can be used to define the boundaries of acceptable motion in a motor/drive system. 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. When defining these inputs as end-of-travel limits both inputs are defined together as either active low, active high, or not used. See below for details.

BLu, STAC6

Defines usage of inputs X6 and X7 as dedicated end-of-travel limits. X6 is the CCW limit input and X7 is the CW limit input. If not needed, X6 and X7 can be redefined as general purpose inputs.

ST-Q/Si, SV

Defines the usage of inputs X7 and X8 as dedicated end-of-travel limits. X7 is the CW limit input and X8 is the CCW limit input. If not needed, X7 and X8 can be redefined as general purpose inputs.

ST-S, STM

Defines the STEP and DIR inputs as CW end-of-travel and CCW end-of-travel limit inputs, 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 end-of-travel limit inputs you can define them as such in two ways, with the *ST Configurator* software, or with the DL command. DL takes no effect if the drive is set in Command Mode (CM) 7, 11, 12, 13, 14, 15, 16, 17 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, 11, 12, 13, 14, 15, 16, 17 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). Motion stops automatically at rate defined by AM command.
DL2: End-of-travel limit occurs when an input is open (de-energized). Motion stops automatically at rated defined by AM command.
DL3: Inputs are not used as end-of-travel limit inputs and can be used as a general purpose inputs. In the case of ST-S and STM drives, DL will be automatically set to 3 if CM is set to 7, 11, 12, 13, 14, 15, 16, 17, or 18, or if JE is executed after the DL command is set.

Command Details:

Structure	DL{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Limit input state (see above)
- units	integer number
- range	1, 2 or 3

Examples:

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

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

DR - Data Register for Capture

Compatibility: Q servo drives only (BLu-Q and SV-Q)
Affects: Quick Tuner Data Capture

Sets or requests the data register used in the register plot data source in Quick Tuner. Any data register can be selected for viewing when capturing data using Quick Tuner.

Command Details:

Command Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	All data register assignments

Examples:

Command	Drive sends	Notes
DRa	-	Set capture data register to “a” (Analog Command) register

EF - Encoder Function

Compatibility: Stepper drives with encoder feedback
Affects: Stall Detection and Stall Prevention
See also: ER, PF commands

STAC6, ST-Q/Si, STM

Sets or requests the decimal equivalent of the encoder function's 3-bit word. It is strongly recommended to use the *Configurator* software to select the desired encoder function, however the EF command can also be used. After setting the EF command a power cycle to the drive (OFF, ON) is required for the setting to take effect. Only stepper drives with encoder inputs (standard on STAC6 drives, optional on ST-Q/Si and STM drives) running a step motor with a shaft-mounted encoder can utilize the Stall Detection and Stall Prevention functions. Setting the encoder function involves turning on only one of the 3 bits in the encoder function's binary word. The bit assignments are below.

Bit 0 (EF1): Turn Stall Detection ON.
Bit 1 (EF2): Turn Stall Prevention ON.
Bit 2 (EF4): Turn Stall Prevention with time-out ON.

Since only one encoder function can be turned on at a time, the drive will only perform the last function set by the EF command. In other words, executing EF2 after EF1 will overwrite the Stall Detection setting by turning on Stall Prevention. Executing EF0 turns the enabled encoder function off.

NOTE: Remember that power cycling the drive (OFF, ON) is required for the EF setting to take place.

Command Details:

Structure	EF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Encoder function setting
- units	decimal equivalent of 3-bit binary word
- range	0 = Encoder function off 1 = Stall Detection 2 = Stall Prevention 4 = Stall Prevention with time-out

Examples:

Command	Drive sends	Notes
EF1	-	Turn ON Stall Detection function.
EF0	-	Turn OFF encoder function

EG - Electronic Gearing

Compatibility: All drives
Affects: Command Mode 7, FE and HW commands
See also: CM, ER, FE and HW commands.

BLu, SV
Sets or requests the pulses per revolution for electronic gearing. For example, with an EG value of 20000 the servo drive will require 20000 pulses from the master pulse source to move the servo motor 1 revolution.

STAC6, ST-Q/Si, ST-S, STM
Sets or requests the desired step/microstep resolution of the step motor.

Command Details:

Structure	EG{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	"R" (034) Note: With servo drives the EG command is equal to the "R" register. With stepper drives the EG command is equal to twice the "R" register.
Parameter #1	Servo = electronic gearing ratio Stepper = step resolution
- units	Servo = counts/rev Stepper = steps/rev
- range	Servo = 200 - 32000 Stepper = 200 - 51200

Examples:

Command	Drive sends	Notes
EG20000	-	Set electronic gearing resolution in servo drive to 20000 pulses/rev
EG	EG=20000	
RLR	RLR=20000	"R" register matches the EG setting in a servo drive
EG36000	-	Set microstep resolution to 36000 steps/rev in a stepper drive
EG	EG=36000	
RLR	RLR=18000	"R" register contains 1/2 the EG setting in a stepper drive, or 18000 steps/rev

EP - Encoder Position

Compatibility: Servo drives and stepper drives with encoder feedback
Affects: Encoder position value
See also: SP, MT, WM commands.

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

WARNING: When in Multi-tasking mode (see MT command), the EP command should not be issued while the drive is simultaneously executing a move command (CJ, FL, FP, FS, etc.). A drive fault may result.

Command Details:

Structure	EP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	NO
Register Access	“e” (053) read only

Parameter Details:

Parameter #1	Encoder position value
- units	Counts
- range	-2,147,483,647 to 2,147,483,647

Examples:

Command	Drive sends	Notes
EP0	-	Set encoder position to “0”
EP	EP=0	

ER - Encoder Resolution

Compatibility: Servo drives and stepper drives with encoder feedback
Affects: Motor Operation

Sets the encoder resolution in quadrature counts. For example, if the motor connected to the drive has an 8000 count (2000 line) per revolution encoder, set the encoder resolution to 8000.

WARNING: Changing this setting will affect motor commutation with servo drives. Use the Quick Tuner setup utility to change this setting, then run the "Timing Wizard" in Quick Tuner to properly set up the motor commutation.

Command Details:

Structure	ER{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Encoder resolution
- units	encoder counts/rev
- range	200 - 128000

Examples:

Command	Drive sends	Notes
ER8000	-	Set encoder resolution to 8000 counts/rev
ER	ER=8000	

FC - Feed to Length with Speed Change

Compatibility: All drives, though Q drives have added functionality (see below)
See also: VC, VE, DC, DI, WP commands

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, respectively. Initial speed is VE. After the motor has moved DC counts, the speed changes to VC. DC must not exceed DI.

Q drives only

With Q drives there may be multiple VCs and DCs per FC command, allowing for more complex, multi-velocity moves. To make multi-velocity moves with more than one speed change, the WP (Wait Position) command is also required. A sample sequence is shown below.

Command Details:

Structure	FC
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
DI50000	-	Set distance to 50000 counts/steps
VE5	-	Set velocity to 5 rps
DC40000	-	Set change distance to 40000 counts/steps
VC0.5	-	Set change velocity to 0.5 rps
FC	-	Initiate move
For Q drives only		
MT1	-	Turn multi-tasking ON*
DI50000	-	Set overall move distance to 50000 counts/steps
VE5	-	Set initial velocity to 5 rps
DC10000	-	Set 1st change distance to 10000 counts/steps
VC10	-	Set 1st change velocity to 10 rps
FC	-	Initiate move
WP	-	Wait position
DC20000	-	Set 2nd change distance to 20000 counts/steps
VC1	-	Set 2nd change velocity to 1 rps
WP	-	Wait position
DC30000	-	Set 3rd change distance to 30000 counts/steps
VC0.5	-	Set 3rd change velocity to 0.5 rps

* Because multi-tasking is required for the WP command to be used, only Q models can perform multi-segment moves.

FD - Feed to Double Sensor

Compatibility: All drives
See also: FM, FS, FY, VC commands; see AT command for using analog input as sensor input

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.

BLu, STAC6
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/OUT 1 or IN/OUT 2. In the case of BLuDC drives this means that both inputs must reside on the same connector, either the main driver board I/O connector (DB-25) or the top board connector (screw terminal).

Command Details:

Structure	FD(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, 1st Input number, Input condition NOTE: “X” designates that both inputs in this command reside on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge
Parameter #2	2nd Input number, Input condition
- units	integer, letter
- range	IN/OUT 1 or main board driver board (DB-25) - integer: 0 - 7, 8, 9, : (0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	1st Input number, Input condition

- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge
Parameter #2	2nd Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	1st Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge
Parameter #2	2nd Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
FDX2F4H	-	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

NOTE: When working with digital inputs and outputs in ST-S 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

FE - Follow Encoder

Compatibility: All drives
See also: EG, MT commands

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, or use the "R" register to dynamically adjust the following resolution while following. (Note that in stepper drives the "R" register is equal to 1/2 the EG command).

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 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 mode it was in before executing the FE command.

NOTE: You must use the appropriate configuration software - Quick Tuner for servos, Configurator for steppers - to set up the STEP/X1 and DIR/X2 inputs for encoder following. Do this by choosing A/B Quadrature in the Position (servo) or Pulse & Direction (stepper) mode settings.

NOTE: Take care when changing the "R" register while following because some move parameters will be scaled as well and therefore the move may change unexpectedly.

BLu, STAC6
An optional "X" in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	FE(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "X", 1st Input number, Input condition NOTE: "X" designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "X", integer, letter

- range	IN/OUT 1 or main driver board (DB-25) - integer: X0, X3 - X7, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	1st Input number, Input condition
- units	integer, letter
- range	integer: 0, 3 - 8 (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	1st Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

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

NOTE: When working with digital inputs and outputs in ST-S 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

FI - Filter Input

Compatibility: All drives
Affects: All commands using inputs
See also: FX, RC, WI and all feed to sensor commands

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 (one cycle = 250 usec) before high is updated as the input state. A filter value of 100 equals 25 msec. A filter value of “0” disables the filter.

BLu, STAC6

This command can be used to apply filters to low speed inputs X3 through X7 on the main driver board of all drives, and can also be used on top board inputs IN3 through IN7 of SE, QE, and Si drives. Reassigning the filters to top board inputs of SE, QE and Si drives is done with the FX command.

SV, ST-Q/Si

This command can be used to apply filters to low speed inputs X3 through X8.

ST-S, STM

This command can be used to apply filters to inputs STEP, DIR, and EN

Command Details:

Structure	FI(Parameter#1)(Parameter#2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Input number
- units	integer
- range	3 - 7 ... if FX=1, designates X3 - X7 of all drives ... if FX=0, designates IN3 - IN7 of SE, QE and Si drives.
Parameter #2	Filter value
- units	processor cycles
- range	0 - 32767 (0 disables the filter)

SV, ST-Q/Si

Parameter #1	Input number
- units	integer
- range	3 - 8
Parameter #2	Filter value
- units	processor cycles
- range	0 - 32767 (0 disables the filter)

ST-S, STM

Parameter #1	Input number
- units	integer
- range	1 = STEP, 2 = DIR, 3 = EN
Parameter #2	Filter value
- units	processor cycles
- range	0 - 32767 (0 disables the filter)

Examples:

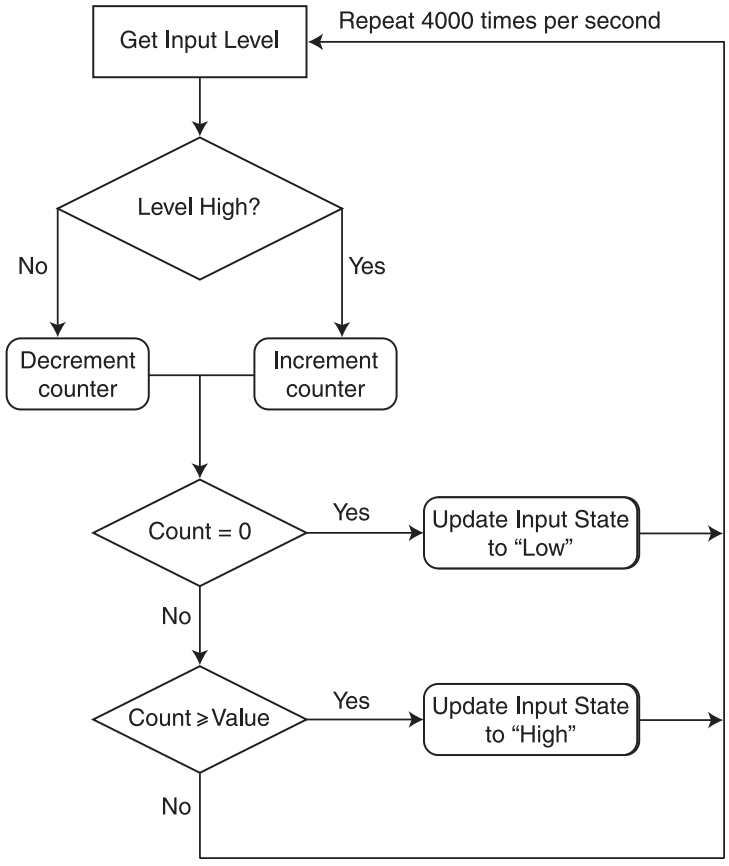
Command	Drive sends	Notes
FI4100	-	Requires that input X4 (if FX=1) maintain the same state (low or high) for 25 milliseconds before the drive registers that state
FI4	FI4=100	

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 flexibility by allowing the user to select the amount of filtering required to eliminate the effects of noise or bounce.

The digital filters implemented in ST-S drives work by continuously monitoring the level of the inputs to which filters have been applied using the FI command. During each processor cycle (every 250 microseconds), 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 is also known as the filter value, and is set by the FI command. The flow chart to the right shows how a digital filter works.



For example, if we apply a digital filter of 2 milliseconds to input 3, it means we'd like the level of input 3 (low or high) to be true for 2 milliseconds before the processor updates the state of input 3 to the state requested by the command currently being executed. If the command being executed is a WI3L command, which literally means "wait for input 3 low", it means the processor will wait until the level of input 3 has been low for 2 milliseconds before updating the state of the input as low and finishing the WI3L command. If by chance input 3 has already been low for the prerequisite 2 milliseconds when the WI3L command is initiated, there will be no delay in executing the command. On the other hand, if input 3 is high when the WI3L command is initiated, there will be an ad-

ditional minimum delay of 2 milliseconds after the input changes state from high to low. 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 250 microseconds. To filter the EN input for 2 milliseconds the value of the FI command would then be 2 msec divided by 250 usec, or 8. The correct syntax for the FI command would then be "FI38".

As can be seen from the example and flow chart above, the functioning of a digital input filter incorporates an averaging effect on the level of the input. This means that in the example above, if the level of the 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. 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.

Filter values are volatile, meaning that they are lost at power-down of a drive. Therefore filter values must be set each time the drive is powered on.

NOTE: When working with digital inputs and outputs in ST-S 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.*

FL - Feed to Length

Compatibility: All drives
See also: AC, DE, DI, VE commands

Executes a relative move command. Move distance and direction come from the last DI command. Speed, accel and decel are from the VE, AC and DE commands, respectively. Executing the FL command with no parameter initiates a feed to length move that uses the last DI command for direction and distance. Executing the FL command with a parameter uses the parameter settings for direction and distance without changing the DI command.

Command Details:

Structure	FL{Parameter #1}
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Relative distance
- units	counts or steps
- range	-2,147,483,647 to 2,147,483,647 sign determines direction: "-" for CCW, no sign for CW

Examples:

Command	Drive sends	Notes
DI20000	-	Set distance to 20000 counts in the CW direction
FL	-	Launch Feed to Length move
FL20000	-	Launch Feed to Length move of 20000 counts in the CW direction without affecting the DI command
FL-400	-	Launch Feed to Length move of 400 counts in the CCW direction without affecting the DI command

FM - Feed to Sensor with Mask Distance

Compatibility: All drives
See also: FS command

Executes a Feed to Sensor command (see FS command) except sensor is ignored for the first DC counts of the move. In other words the sensor is “masked” for a beginning portion of the move. This command is useful for ignoring noise from a mechanical switch or for clearing a part before sensing the next one.

BLu, STAC6
An optional “X” in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	FM(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, 1st Input number, Input condition NOTE: “X” designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	1st Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates encoder index channel) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	1st Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN 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 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 a 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 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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

FO - Feed to Length and Set Output

Compatibility: All drives
See Also: DC, AO, BO, MO commands

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 AO, BO, and MO commands.

BLu, STAC6
An optional "Y" in Parameter #1 is used to designate outputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structured	FO(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "Y", Output number, Output condition NOTE: "Y" designates that the output in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "Y", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: Y1 - Y3 - letter: L = low (closed), H = high (open) IN/OUT 2 or top board (screw terminal) - integer: 1 - 4 - letter: L = low (closed), H = high (open)

SV, ST-Q/Si

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 - 4 letter: L = low (closed), H = high (open)

ST-S, STM

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 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 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

FP - Feed to Position

Compatibility: All drives
See also: AC, DE, DI, VE commands

Executes an absolute move command. Move position comes from the last DI command. Speed, accel and decel are from VE, AC and DE commands, respectively. Executing the FP command with no parameter initiates a feed to position move that uses the last DI command for position. Executing the FP command with a parameter uses the parameter for position without changing the DI command.

Command Details:

Structure	FP{Parameter #1}
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Absolute position
- units	counts or steps
- range	-2,147,483,647 to 2,147,483,647 sign determines direction: "-" for CCW, no sign for CW

Examples:

Example: After homing the motor you want to zero the home position and move to an absolute position 8000 counts (or steps) from the new home position.

Command	Drive sends	Notes
SP0	-	Set current motor position as absolute zero
DI8000	-	Set move position to 8000 counts/steps
FP	-	Launch Feed to Position
FP8000	-	Launch Feed to Position to 8000 counts/steps without affecting the "D" register

FS - Feed to Sensor

Compatibility: All drives
See also: FD, FM and FY commands; see AT command for using AIN as sensor input

Executes a Feed to Sensor command. Requires input number and condition. The motor moves until a sensor triggers the specified input condition, 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 for DI 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 will equal the encoder resolution for a servo motor and the EG setting for a step motor.

$$\text{minimum decel distance} = \frac{(V)^2(R)}{2(D)}$$

BLu, STAC6
An optional "X" in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	FS(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "X", Input number, Input condition NOTE: "X" designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "X", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
FS1L	-	Launch move and decel to stop when sensor tied to input 1 is low
FS3R	-	Launch move and decel to stop when sensor tied to input 3 changes from low to high (rising edge)
FSX5L	-	Launch move and decel to stop when sensor tied to input X5 is low

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

FX - Filter select inputs

Compatibility: BLu servo and STAC6 stepper drives
Affects: FI command on SE, QE, and Si drives
See also: FI command

The FX command allows changing the target inputs of a drive's digital input filters from the main board X3 through X7 inputs to the top board IN3 through IN7 inputs. This can only be done on SE, QE, and Si drives with firmware 1.53U or later.

Command Details:

Structure	FX{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Digital inputs selector
- units	integer
- range	0 = top board inputs of SE, QE, and Si drives 1 = main board inputs of all drives

Examples:

Command	Drive sends	Notes
FX0	-	Cause digital input filters set by FI command to affect top board inputs IN3 through IN7 of SE, QE, and Si drives.
FX	FX=1	Digital filters are set to be applied to main driver board inputs X3 through X7.

FY - Feed to Sensor with Safety Distance

Compatibility: All drives
See also: DC, FD, FM and FS commands; see AT command for using AIN as sensor input

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 ("!") and adds a value of 1 to the Other Flags register ("F" register). DI defines the direction of rotation and the stop distance to move after the sensor triggers the stop input condition. 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 required 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.

BLu, STAC6

An optional "X" in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	FY(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	"F" (022) Executing the FY command will put a value of 2 in the "F" register when the sensor is successfully found, or a value of 1 in the "F" register if the safety distance is met. If you plan to use the "F" register for monitoring the success of the FY command you must zero the register before each FY command by executing RLF0.

Parameter Details:

BLu, STAC6

Parameter #1	Optional "X", Input number, Input condition NOTE: "X" designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "X", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter

- range	integer: 0 - 8, 9, :, ; (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
DI2000	-	Set distance to stop beyond sensor to 2000 counts/steps
DC60000	-	Set safety distance to 60000 counts/steps
FY2L	-	Launch Feed to Sensor: motor will stop when input 2 is low or when 60000 counts/steps are reached: whichever event comes first

When using the SE, QE, or Si drives and needing to access the main driver board inputs...

FYX2L	-	Launch Feed to Sensor: motor will stop when main driver board input 2 is low or when 60000 counts/steps are reached: whichever event comes first

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

GC - Current Command

Compatibility: Servo drives only
Affects: Commanded motor current
See also: CM command

Sets or requests the immediate current command for the servo motor and drive when the servo drive is set for Command Mode 1 (CM1).

NOTE: Setting this value may make the servo motor run to a very high speed, especially if there is no load on the motor. Take care when using this command.

Command Details:

Structure	GC{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	“G” (023)

Command Details:

Parameter #1	RMS Current
- units	0.01 amps rms
- range	-2000 to +2000 (+/- 20 amps rms)

Examples:

Command	Drive sends	Notes
CM1	-	Set servo drive to Commanded Current Command Mode
GC100	-	Set current to motor at 1 A rms
GC-100	-	Set current to motor at -1 A rms (opposite direction)

HG - 4th Harmonic Filter Gain

Compatibility: Stepper drives only
Affects: Low-speed performance of step motors
See also: HP command

Sets or requests the 4th harmonic filter gain setting. This setting works in conjunction with the 4th harmonic filter phase setting (HP) to reduce low-speed torque ripple in step motors.

NOTE: We strongly suggest you set this value in the ST Configurator software application only.

Command Details:

Structure	HG{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	YES only when set in Configurator software, otherwise NO
Register Access	None

Parameter Details:

Parameter #1	Filter gain
- units	integer number
- range	0 - 32767

Examples:

Command	Drive sends	Notes
HG8000	-	Set filter gain value to 8000
HG	HG=8000	

HP - 4th Harmonic Filter Phase

Compatibility: Stepper drives only
Affects: Low-speed performance of step motors
See also: HG command

Sets or requests the 4th harmonic filter phase setting. This setting works in conjunction with the 4th harmonic filter gain setting (HG) to reduce low-speed torque ripple in step motors.

NOTE: We strongly suggest you set this value in the ST Configurator software application only.

Command Details:

Structure	HP{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	YES only when set in Configurator software, otherwise NO
Register Access	None

Parameter Details:

Parameter #1	Filter phase
- units	integer number
- range	-125 to +125

Examples:

Command	Drive sends	Notes
HG105	-	Set 4th harmonic filter gain to 105
HG	HG=105	

HW - Hand Wheel

Compatibility: All drives
See also: EG, FE, and MT commands; see AT command for using analog input as sensor input

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. In other words, 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.

BLu, SV, STAC6, ST-Q/Si

Inputs X1 and X2 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 motor.

ST-S, STM

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.

Command Details:

Structure	HW(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, Input number, Input condition NOTE: “X” designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0, X3 - X7, X8, X9, X: (X0 designates the encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0, 3 - 8 (0 designates the encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter

- range	integer: 0 = Encoder Index (if present), 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

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

Compatibility: All drives
See Also: AD, AV, and AZ commands

BLu, SV, STAC6, ST-Q/Si

Requests present analog input value from the given source. There are three different analog values that can be accessed. With no parameter the IA command returns the Analog Command value which is derived from the analog inputs with gain and offset values applied as set in *Quick Tuner* or *Configurator* or via the AD, AV and/or AZ commands. When a parameter is given raw (unscaled) analog input values are returned.

ST-S, STM

Requests present analog input value. There are two different analog values that can be accessed. With no parameter the IA command returns the Analog Command value which is derived from the analog input with gain and offset values applied as set in *ST Configurator* or via the AD, AV and/or AZ commands. When a parameter is given raw (unscaled) analog input values are returned.

Command Details:

Structure	IA{Parameter #1}
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	All drives... “a” (049) Analog Command BLu, SV, STAC6, ST-Q/Si... “j” (058) Analog input 1 (unscaled) “k” (059) Analog input 2 (unscaled) ST-S, STM... “j” (058) AIN (unscaled)

Parameter Details:

BLu, SV, STAC6, ST-Q/Si

Parameter #1	Analog input
- units	integer
- range	No parameter or 0 = Analog command 1 = Analog input 1 (unscaled) 2 = Analog input 2 (unscaled) 3 = Expanded analog input (SE, QE, and Si models)

ST-S, STM

Parameter #1	Analog input
- units	integer
- range	No parameter or 0 = Analog command 1 = AIN (unscaled)

Examples:

Command	Drive sends	Notes
IA	IA=2.5	Analog Command is at mid range when drive is set to 0-5 volt input
IA1	IA=4.99	Analog input 1 is near 5 volts

IC - Immediate Current (Commanded)

Compatibility: All drives

Servo drives

Requests the present RMS current commanded by the servo loop. This may not be the actual current at the motor windings. Most AC servo motors are commutated using a sinusoidal current waveform that is a “peak” value and not directly represented by the commanded current. The commanded current is the average RMS current being asked of the driver. Typically with a well tuned current loop the RMS current in the servo motor is well represented by this value.

Stepper drives

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

Command Details:

Structure	IC
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“c” (051)
Units	0.01 amps

Examples:

Command	Drive sends	Notes
IC	IC=015E	3.5 amps
IC	IC=FEA2	-3.5 amps

If the IF command is set with Parameter #1=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

Compatibility: All drives

BLu, STAC6
Requests the total relative distance moved in the last completed move.

SV, ST-Q/Si, ST-S, STM
Requests the immediate relative distance traveled from the beginning of the last move. Once the move is finished the value will be equal to the relative distance of that last move until another move is initiated, at which time the value will zero and begin tracking the new relative distance moved.

Command Details:

Structure	ID
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“d” (052)
Units	encoder counts (servo) steps (stepper)

Examples:

Command	Drive sends	Notes
ID	ID=00002710	10000 (10000 counts into CW move)
ID	ID=FFFFD8F0	-10000 (10000 counts into CCW move)

If the IF command is set with Parameter #1=D

ID	ID=10000	10000 counts into CW move
ID	ID=-10000	10000 counts into CCW move

IE - Immediate Encoder

Compatibility: Servo drives and stepper drives with encoder feedback

Requests present encoder position.

Command Details:

Structure	IE
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“e” (053)
Units	encoder counts

Examples:

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

If the IF command 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

Compatibility: All drives
Affects: Immediate Commands IA, IC, ID, IE, IP, IT, IU, IV and IX

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 given baud rate (see BR and PB commands). 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.

Command Details:

Structure	IF{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	NO
Register Access	None

Parameter Details:

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

Compatibility: All drives
See also: IL, SO commands

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

BLu, STAC6
An optional "Y" in Parameter #1 is used to designate outputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	IH(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "Y", Output number, Output condition NOTE: "Y" designates that the output in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "Y", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: Y1 - Y3 - letter: L = low (closed), H = high (open) IN/OUT 2 or top board (screw terminal) - integer: 1 - 4 - letter: L = low (closed), H = high (open)

SV, ST-Q/Si

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 - 4 letter: L = low (closed), H = high (open)

ST-S, STM

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 letter: L = low (closed), H = high (open)

Examples:

Command	Drive sends	Notes
IH1	-	Output 1 goes high immediately
IH2	-	Output 2 goes high immediately

To force Outputs on main driver board when using an SE, QE or Si drive

IHY1	-	Output 1 of main driver board goes high immediately
IHY2	-	Output 2 of main driver board goes high immediately

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

IL - Immediate Low Output

Compatibility: All drives
See also: IH, SO commands.

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

BLu, STAC6
An optional "Y" in Parameter #1 is used to designate outputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	IL(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "Y", Output number, Output condition NOTE: "Y" designates that the output in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "Y", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: Y1 - Y3 - letter: L = low (closed), H = high (open) IN/OUT 2 or top board (screw terminal) - integer: 1 - 4 - letter: L = low (closed), H = high (open)

SV, ST-Q/Si

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 - 4 letter: L = low (closed), H = high (open)

ST-S, STM

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 letter: L = low (closed), H = high (open)

Examples:

Command	Drive sends	Notes
IL1	-	Output 1 goes low immediately
IL2	-	Output 2 goes low immediately

To force Outputs on main driver board when using an SE, QE, or Si drive

ILY1	-	Output 1 of main driver board goes low immediately
ILY2	-	Output 2 of main driver board goes low immediately

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

IO - Output Status

Compatibility: All drives

With no parameter this command requests the immediate status of the designated outputs. The status is displayed as an 8-bit binary number with output 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.

BLu, STAC6
An optional “Y” in Parameter #1 is used to designate outputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	IO{Parameter #1}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “Y”, Decimal equivalent of binary output pattern NOTE: “Y” designates that the outputs in this command reside on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “Y”, integer
- range	IN/OUT 1 or main driver board (DB-25): 0 - 7 IN/OUT 2 or top board (screw terminal): 0 - 15

SV, ST-Q/Si

Parameter #1	Decimal equivalent of binary output pattern
- units	integer
- range	0 - 7

ST-S, STM

Parameter #1	Decimal equivalent of binary output pattern
- units	integer
- range	0 - 1

Examples:

BLu and STAC6-S or -Q versions (optional “Y” character is not necessary)

Command	Drive sends	Notes
IO	IO=00000000	All 3 outputs of IN/OUT1 or main board are low (closed)
IO	IO=00000111	All 3 outputs of IN/OUT1 or main board are high (open)
IO0	-	Sets all 3 outputs low (closed)
IO7	-	Sets all 3 outputs high (open)

BLu and STAC6-QE or -Si versions

Command	Drive sends	Notes
IO	IO=00000000	All 4 outputs of IN/OUT2 or top board are low (closed)
IO	IO=00001111	All 4 outputs of IN/OUT2 or top board are high (open)
IO0	-	Sets all 4 outputs of IN/OUT2 or top board low (closed)
IO15	-	Sets all 4 outputs of IN/OUT2 or top board high (open)
IOY	IO=00000000	All 3 outputs of IN/OUT1 or main board are low (closed)
IOY	IO=00000111	All 3 outputs of IN/OUT1 or main board are high (open)
IOY0	-	Sets all 3 outputs of IN/OUT1 or main board low (closed)
IOY7	-	Sets all 3 outputs of IN/OUT1 or main board high (open)

SV, ST-Q/Si

Command	Drive sends	Notes
IO	IO=00000000	All 4 outputs are low (closed)
IO	IO=00001111	All 4 outputs are high (open)
IO0	-	Sets all 4 outputs low (closed)
IO7	-	Sets all 4 outputs high (open)

ST-S, STM

Command	Drive Sends	Notes
IO	IO=00000000	Output is low (closed)
IO	IO=00000001	Output is high (open)
IO0	-	Sets output low (closed)
IO1	-	Sets output high (open)

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, 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.*

IP - Immediate Position

Compatibility: All drives

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.

Command Details:

Structure	IP
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	encoder counts (servo) steps (stepper)

Examples:

Command	Drive sends	Notes
IP	IP=00002710	Absolute position is 10,000 counts (or steps)
IP	IP=FFFFD8F0	Absolute position is -10,000 counts (or steps)

If the IF command is set with Parameter #1=D

IP	IP=10000	Absolute position is 10000 counts (or steps)
IP	IP=-10000	Absolute position is -10000 counts (or steps)

IQ - Immediate Current (Actual)

Compatibility: Servo drives only

Requests present actual current. This current reading is the actual current measured by the drive. As with the Commanded Current this is an RMS value that represents the DC current in the motor windings.

Command Details:

Structure	IQ
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	0.01 Amps

Examples:

Command	Drive sends	Notes
IQ	IQ=015E	3.5 Amps
IQ	IQ=FEA2	-3.5 Amps

If the IF command is set with Parameter #1=D

IQ	IQ=350	3.5 Amps
IQ	IQ=-350	-3.5 Amps

IS - Input Status

Compatibility: All drives

Requests immediate status of all drive inputs. A closed input is represented by a “0” (zero), and an open input is represented by a “1” (one). Unused positions in the response are represented by “0” (zero).

BLu, STAC6

On S and Q drives the IS command requests the status of IN/OUT1 or main driver board (DB-25) inputs X1 through X7 plus the encoder index channel (if present). On SE, QE, and Si drives the ISX command (IS command with parameter character X) is required to request status of IN/OUT1 or main driver board (DB-25) inputs X1 through X7 plus the encoder index channel (if present), while IS requests IN/OUT2 or top board (screw terminal) inputs 1 through 8.

SV, ST-Q/Si

The IS command requests the status of inputs X1 through X8 plus the encoder index channel (if present).

ST-S, STM

The IS command requests the status of all three digital inputs, STEP, DIR, and EN, plus the encoder index channel (STM only, if present).

Command Details:

Structure	IS{Parameter #1}
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X” character used to access driver board inputs with SE, QE, and Si drives.

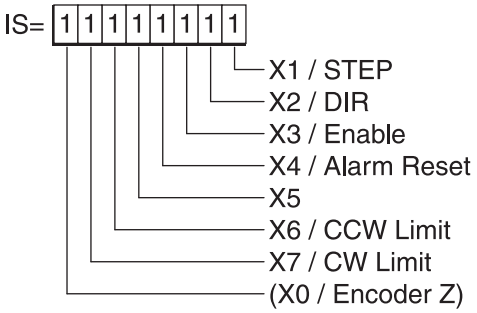
SV, ST-Q/Si, ST-S, STM

Parameter #1	“X” character ignored if used.

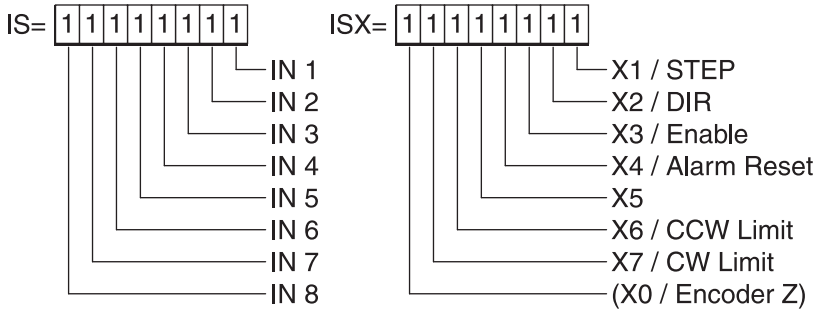
Response Details:

BLu, STAC6

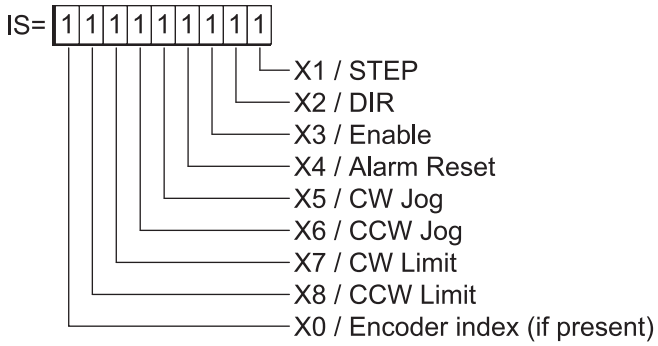
S and Q drives
(“X” character is not required to designate main board inputs)



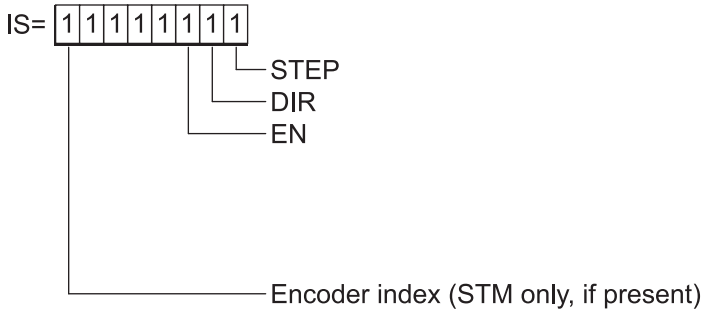
SE, QE, and Si drives
(“X” character is required to designate main board inputs)



SV, ST-Q/Si



ST-S, STM



Examples:

BLu and STAC6-S or -Q versions (optional "X" character is not necessary)

Command	Drive sends	Notes
IS	IS=00000000	All 8 inputs are low (closed)
IS	IS=11111111	All 8 inputs are high (open)
IS	IS=11101100	Inputs 1, 2, and 5 are closed
IS	IS=10000101	Inputs 2, 4, 5, 6, and 7 are closed

BLu and STAC6-SE, -QE, or -Si versions (optional "X" character necessary to access IN/OUT1 or main driver board (DB-25) inputs

Command	Drive sends	Notes
IS	IS=11010011	Inputs 3, 4, and 6 are closed
ISX	IS=10101110	Inputs X1, X5, and X7 are closed

SV, ST-Q/Si

Command	Drive sends	Notes
IS	IS=100110110	Inputs 1, 4, 7, and 8 are closed
IS	IS=011111111	Encoder index channel is closed

ST-S, STM

Command	Drive Sends	Notes
IS	IS=10000111	All inputs are open
IS	IS=00000111	Encoder index channel is closed
IS	IS=10000100	Inputs STEP and DIR are closed

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, 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 com-*

mands 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.

IT - Immediate Temperature

Compatibility: All drives

Requests present value of the temperature sensor on the main driver board. The temperature reads out in 0.1 degrees Centigrade resolution. The drive will fault when the temperature reaches a specified maximum value. (See hardware manuals for details).

Command Details:

Structure	IT
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“t” (068)
Units	0.1 deg C

Examples:

Command	Drive sends	Notes
IT	IT=275	Drive temperature is 27.5 C°
IT	IT=310	Drive temperature is 31.0 C°

IU - Immediate Voltage

Compatibility: All drives

Requests 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 then a minimum value. (See hardware manuals for details).

Command Details:

Structure	IU
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“u” (069)
Units	0.1 Volts DC, +/-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

Compatibility: All drives

Requests 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 Details:

Structure	IV(Parameter #1)
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	"v" (070) Actual velocity (servo drives and stepper drives with encoder) "w" (071) Target velocity

Parameter Details:

Parameter #1	Velocity selector
- units	integer
- range	0 = actual velocity request (servo drives and stepper drives with encoder) 1 = target velocity request

Examples:

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

IX - Immediate Position Error

Compatibility: Servo drives and stepper drives with encoder feedback

Requests present position error between motor and encoder.

Command Details:

Structure	IX
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	“x” (072)
Units	encoder counts

Examples:

Command	Drive sends	Notes
IX	IX=10	Position error is 10 counts

JA - Jog Acceleration

Compatibility: All drives
Affects: CJ, WI (jogging) commands
See also: CJ, CS, JD, JE, JL, JS, SJ

Sets or requests the accel/decel rate for Jog moves in rev/sec/sec. Sending JA with no parameter 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.

Command Details:

Structure	JA{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	"K" (027) Note: The JA command uses different units than the "K" register. See Data Registers section for details.

Parameter Details:

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
JA100	-	Set jog acceleration to 100 rev/sec/sec
JA	JA=100	

JC - Velocity (Oscillator) mode second speed

Compatibility: Stepper drives and SV servo drives
Affects: Analog velocity mode
See also: AD, AG, CM commands

Sets or requests the second speed used in velocity (oscillator) mode. This only applies to Command Modes (CM) 13, 14, 17, and 18.

SV, STAC6, ST-Q/Si

Input X5 is used to select the speed set by the JC command while in Command Mode 13, 14, 17 or 18.

ST-S, STM

The EN input is used to select the speed set by the JC command while in Command Mode 13, 14, 17 or 18.

Command Details:

Structure	JC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Analog velocity mode second speed
- units	rev/sec
- range	BLu, SV, STAC6, ST-Q/Si, ST-S: 0.0042 - 133.3333 (resolution is 0.0042) STM: 0.0042 - 80.0000 (resolution is 0.0042)

Examples:

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

JD - Jog Disable

Compatibility: All drives
Affects: Jogging during a WI command
See also: JA, JE, JS, WI commands

Disables jog inputs (which are active during a WI instruction if previously enabled by the JE command). Jog accel/decel and velocity are set using the JA and JS commands, respectively.

Command Details:

Structure	JD
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
JD	-	Disable jog inputs while executing the WI command

JE - Jog Enable

Compatibility: All drives
Affects: WI (jogging) command
See also: JA, JD, JS, WI commands

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

BLu, STAC6-S, SE, Q, QE

Inputs X1 and X2 are the designated jog inputs during a WI instruction.

BLu, STAC6-Si

Inputs 5 and 6 of IN/OUT2 or top board (screw terminal) connector are the designated jog inputs during a WI instruction.

SV, ST-Q/Si

Inputs X5 and X6 are the designated jog inputs during a WI instruction.

ST-S, STM

The STEP and DIR inputs are the designated jog inputs during a WI instruction. 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, 11, 12, 13, 14, 15, 16, 17 or 18, because these modes predefine these inputs and take precedence over the JE command. Also, setting the DL command (to 1 or 2) after setting the JE command reassigns the STEP and DIR inputs as end-of-travel limit inputs and turns off jogging functionality. In other words, the JE and DL commands, as well as Command Modes (CM) 7, 11, 12, 13, 14, 15, 16, 17 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 and DL from using the inputs. JE and DL exclude each other by overwriting the usage of the STEP and DIR inputs. To enable jogging with the STEP and DIR inputs simply execute the JE command with CM=21 or CM=22.

Command Details:

Structure	JE
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
JE	-	Enable jog inputs while executing the WI command
WIX4L	-	Wait for input X4 to close. While waiting jog inputs are active, which means the motor can be jogged in the CW and CCW directions by closing the jog inputs. After input X4 closes the jog function stops, at least until the next WI command executes.

JL - Jog Decel

Compatibility: All drives
Affects: Jogging during WI command, velocity (oscillator) modes, and CJ command
See also: JA command

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

Command Details:

Structure	JL{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Jog deceleration rate
- units	rev/sec/sec (rps/s)
- range	0.167 - 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	

JM - Jog Mode

Compatibility: All drives*, see below
Affects: CJ command, and jogging during a WI command
See also: CJ, JE, WI commands

Sets or requests the Jog mode. There are two Jog modes available:

*Stepper drives only utilize Jog Mode 1 (JM1), and therefore will ignore attempts to set JM2.

Jog Mode 1: Servo and stepper drives
For servo drives, Jog Mode 1 uses a “position-type” of servo control that moves the target position which causes the servo to move at the set velocity. Jog Mode 1 will cause the servo motor to always move the same distance over time. A drawback is that the servo can fault if the position error during the move exceeds the value set by the PF (Position Fault) command. For stepper drives, Jog Mode 1 causes the step motor to run at the set velocity (see JS and CS commands).

Jog Mode 2: Servo drives only
For servo drives only, Jog Mode 2 uses a “velocity-type” of servo control that applies torque to the motor to maintain velocity. This method functions better with high inertia loads because it ignores the value set by the PF (Position Fault) command. It also allows the drive to function in a “torque-limited velocity” mode or a “velocity-limited torque” mode. Jog Mode 2 also uses a different set of control parameters, VI and VP, for “tuning” the velocity mode. See VI & VP commands later in this guide.

Command Details:

Structure	JM{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Jog mode
- units	integer
- range	1 = position-type 2 = velocity-type

Examples:

Command	Drive sends	Notes
JM1	-	Put drive into position-type servo control when jogging
JM2	-	Put drive into velocity-type servo control when jogging
JM	JM=2	

JS - Jog Speed

Compatibility: All drives
Affects: Jogging during WI command, velocity (oscillator) modes, and CJ command
See also: CJ, CS, JA commands

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

Command Details:

Structure	JS{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	"J" (026) Note: The JS command uses different units than the "J" register. See Data Registers section for details.

Parameter Details:

Parameter #1	Move velocity
- units	rev/sec
- range	BLu, SV, STAC6, ST-Q/Si, ST-S: 0.0042 - 133.3333 (resolution is 0.0042) STM: 0.0042 - 80.0000 (resolution is 0.0042)

Examples:

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

KC - Overall Servo Filter

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control overall filter frequency. The filter is a simple one-pole, low-pass filter intended for attenuating high frequency oscillations. The value is a constant that must be calculated from the desired roll off frequency. See equation below.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

C = 72090 / (1400/F + 2.2)
where C = Filter Value, F = desired filter Frequency in Hz

Command Details:

Structure	KC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Filter Value
- units	integer
- range	0 - 32767 (see above for calculations)

Examples:

Command	Drive sends	Notes
KC7836	-	Set servo filter to 200 Hz
KC	KC=7836	

KD - Differential Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control differential gain. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KD is part of the Damping servo parameters in *Quick Tuner*. It works to damp low speed oscillations.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KD{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Differential Gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KD2000	-	Set differential gain to 2000
KD	KD=2000	

KE - Differential Filter

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the differential control parameter filter frequency. The filter is a simple one-pole, low-pass filter intended for attenuating high frequency oscillations. The value is a constant that must be calculated from the desired roll off frequency. See equation below.

C = 72090 / (1400/F + 2.2)
where C = Filter Value, K = desired filter Frequency in Hz

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KE{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Filter Value
- units	integer
- range	0 - 32767

Examples:

Command	Drive sends	Notes
KE7836	-	Set differential filter to 200 Hz

KF - Velocity Feedforward Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control velocity feedforward gain. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KF is part of the Damping servo parameters in *Quick Tuner*. It counters the effects of the KV parameter which can cause large following error. KF is usually the same value as KV.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Velocity feedforward gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KF4000	-	Set velocity feedforward gain to 4000
KF	KF=4000	

KI - Integrator Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control integrator gain term. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KI is part of the Stiffness servo parameters in *Quick Tuner*. It minimizes (or may even eliminate) position errors especially when holding position.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Integrator gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KI5000	-	Set integrator gain to 500
KI	KI=500	

KK - Inertia Feedforward Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control inertia feedforward gain. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KK is an Inertia servo parameter in *Quick Tuner*. KK improves acceleration control by compensating for the load inertia.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KK{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Inertia feedforward gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KK500	-	Set inertia feedforward gain to 500
KK	KK=500	

KP - Proportional Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control proportional gain term. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KP is part of the Stiffness servo parameters in *Quick Tuner*. This parameter is the primary gain term for minimizing the position error.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Proportional gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KP5000	-	Set proportional gain to 5000
KP	KP=5000	

KV - Velocity Feedback Constant

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the servo control velocity feedback gain term. Gain value is relative: “0” meaning no gain, “32767” meaning full gain. KV is part of the Damping servo parameters in *Quick Tuner*. It aids the KD command in damping system oscillation. This term helps to control larger inertial loads.

NOTE: The Velocity Feedback (KV) and Velocity Feedforward (KF) constants are typically set to similar values. The Feedforward value may at times be set larger depending on the frictional content of the motor load.

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KV{Paramter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Velocity feedback gain value
- units	integer
- range	0 - 32767 (0 = 0%, 32767 = 100%)

Examples:

Command	Drive sends	Notes
KV4000	-	Set velocity feedback gain to 4000
KV	KV=4000	

KW - Velocity Filter

Compatibility: Servo drives only
Affects: Servo tuning and performance

Sets or requests the velocity control parameter filter frequency. The filter is a simple one-pole, low-pass filter intended for attenuating high frequency oscillations. The value is a constant that must be calculated from the desired roll off frequency. See equation below.

C = 72090 / (1400/F + 2.2)
where C = Filter Value, K = desired filter Frequency in Hz

NOTE: It is recommended to use the Quick Tuner *software for tuning and configuring your servo system.*

Command Details:

Structure	KW{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Filter Value
- units	integer
- range	0 - 32767 (see calculation above)

Examples:

Command	Drive sends	Notes
KW7836	-	Set velocity filter to 200 Hz
KW	KW=7836	

LV - Low Voltage threshold

Compatibility: All drives
Affects: Under voltage alarm and fault

Sets or requests the low voltage threshold for under voltage alarm / fault conditions. In AC drives (e.g. BLuAC5 and STAC6) an under voltage condition causes a Drive Fault, which disables the motor outputs of the drive. In DC drives (SV, ST, and STM) 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.

Command Details:

Structure	LV{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Low voltage threshold
- units	All drives except BLuAC5: 0.1 volts DC BLuAC5: 1 volt DC
- range	BLuDC: 18 to 40 BLuAC: 90 to 300 STAC6: 90 to 160 ST5: 12 to 75 ST10: 12 to 75 SV7: 12 to 75 STM: 10 to 75

Examples:

Command	Drive sends	Notes
LV	LV=180	Low voltage threshold of ST5 set at 18 VDC
LV200	-	Set low voltage threshold of ST5 drive to 20 VDC
LV	LV=900	Low voltage threshold of STAC6 set at 90 VDC (bus voltage)
LV	LV=90	Low voltage threshold of BLuAC5 set at 90 VDC (bus voltage)

MD - Motor Disable

Compatibility: All drives
Affects: All motion commands
See also: BE, BO, ME commands

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.

Command Details:

Structure	MD
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	No
Register Access	None

Examples:

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

ME - Motor Enable

Compatibility: All drives
Affects: All motion commands
See also: BD, BO, MD commands

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.

Command Details:

Structure	ME
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
ME	-	Drive is enabled
ME	&	Drive is NOT enabled: check Servo Enable input (SI) for proper state

MN - Model Number

Compatibility: All drives

Requests the drive's Model Number. Drive returns a single character that is a code for the model number.

NOTE: Unlike most other commands that request data back from the drive, where the drive will send the original Command Code followed by an "=" and then a value, when the MN command is sent to a drive the drive only responds with the single character code. (See below).

Command Details:

Structure	MN
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	character code (see below)

Response Details:

Model Number	Character code	Model Number	Character Code
BLuDC4-S*	O	STAC6-220-S	\
BLuDC4-SE*	o	STAC6-220-SE	|
BLuDC4-Si*	P	STAC6-220-Q	]
BLuDC4-Q*	W	STAC6-220-QE	}
BLuDC4-QE*	w	STAC6-220-Si	^
BLuDC9-S*	R	ST5-S	D
BLuDC9-SE*	r	ST10-S	E
BLuDC9-Si*	S	ST5-Plus	J
BLuDC9-Q*	X	ST10-Plus	K
BLuDC9-QE*	x	ST5-Q	F
BLuAC5-S	T	ST10-Q	H
BLuAC5-SE	t	ST5-Si	G
BLuAC5-Q	U	ST10-Si	I
BLuAC5-QE	u	STM23S-xxx	a
BLuAC5-Si	V	STM23Q-xxx	b
STAC6-S	Y	SV7-S	;
STAC6-SE	y	SV7-Q	<
STAC6-Q	Z	SV7-Si	=
STAC6-QE	z		
STAC6-Si	[		

* BLu100 and BLu200 series drives have been replaced by BLuDC4 and BLuDC9 series drives, respectively. BLu100 and BLu200 drives are still supported, but part numbers have been changed.

Examples:

Command	Drive sends	Notes
MN	T	Connected drive is a BLuAC5-S

MO - Motion Output

Compatibility: All drives
See also: AO, BO commands

Defines the drive's Motion Output digital output function. There are three Motion Output states that can be defined for all drives 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.

For all stepper drives there are 5 additional states available:

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.

For SV servo drives there are 7 additional states available:

MO4: Output is used as a Tach Output at 64 pulses/rev with 8 pole motor (8 times number of poles)
MO5: Output is used as a Tach Output at 128 pulses/rev with 8 pole motor (16 times number of poles)
MO6: Output is used as a Tach Output at 256 pulses/rev with 8 pole motor (32 times number of poles)
MO7: Output is used as a Tach Output at 512 pulses/rev with 8 pole motor (64 times number of poles)
MO8: Output is used as a Tach Output at 1024 pulses/rev with 8 pole motor (128 times number of poles)
MO9: Output is closed when in position based on encoder error
MO10: Output is open when in position based on encoder error

BLu, SV, STAC6, ST-Q/Si

Output Y2 is the designated Motion Output.

ST-S, STM

The one output of these drives (OUT) can be assigned to one of the four available functions: alarm output, brake output, motion output, or tach 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 the *ST Configurator* software or via the MO command. 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, 2, or 4-8, usage of the output will be reassigned and MO will be automatically set to 3.

Command Details:

Structure	MO{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Condition code

- units	integer
- range	1 - 8 (see above)

Examples:

Command	Drive sends	Notes
MO1	-	Designated output will be closed when motor is not moving
MO	MO=1	

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

MT - Multi-Tasking

Compatibility: Q drives only
Affects: All move commands
See also: CJ, TI, TR, and QJ commands

Sets or request the status of the multi-tasking function (on or off). When multi-tasking is enabled (on), commands such as FL (Feed to Length) or HW (Hand Wheel) do not block execution of subsequent commands in the queue or program segment. This allows executing other type of operations, such as setting outputs (SO), while a move is taking place.

Command Details:

Structure	MT{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	No
Register Access	None

Parameter Details:

Parameter #1	Multi-tasking switch
- units	integer
- range	0 = multi-tasking disabled 1 = multi-tasking enabled

Examples:

Command	Drive sends	Notes
MT1	-	Enables multi-tasking
MT	MT=1	

MV - Model & Revision

Compatibility: All drives except BLu servo drives
See also: MN, RV commands

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 Response Details below.

Command Details:

Structure	MV
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	No
Register Access	None

Response Details:

Response will be in the format AAAABBBC, where AAAA is the firmware version, BBB is the model number code, and C is the sub-model number code. Model and sub-model number codes are listed below by drive, and Examples are given afterward.

Drive	Firmware	Model No. Code	Sub-Model No. Code
SV7-S	*	011	-
SV7-Q	*	012	-
SV7-Si	*	013	-
STAC6-S	*	041	-
STAC6-Q	*	042	-
STAC6-Si	*	043	-
STAC6-220-S	*	044	-
STAC6-220-Q	*	045	-
STAC6-220-Si	*	046	-
ST5-S	*	020	-
ST5-Q	*	022	-
ST5-Si	*	023	-
ST5-Plus	*	026	-
ST10-S	*	021	-
ST10-Q	*	024	-
ST10-Si	*	025	-
ST10-Plus	*	027	-
STM23S-2AN	*	049	A
STM23S-2AE	*	049	E
STM23S-2RN	*	049	C
STM23S-2RE	*	049	G
STM23S-3AN	*	049	B
STM23S-3AE	*	049	F
STM23S-3RN	*	049	D
STM23S-3RE	*	049	H
STM23Q-2AN	*	050	A
STM23Q-2AE	*	050	E
STM23Q-2RN	*	050	C
STM23Q-2RE	*	050	G

STM23Q-3AN	*	050	B
STM23Q-3AE	*	050	F
STM23Q-3RN	*	050	D
STM23Q-3RE	*	050	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	100Q012	Drive connected has DSP firmware version 1.00Q, and the drive model number is SV7-Q
MV	103F042	Drive connected has DSP firmware version 1.03F, and the drive model number is STAC6-Q
MV	102J049A	Drive connected has DSP firmware version 1.03F, and the drive model number is STM23S-2AN

NO - No Operation

Compatibility: Q drives only
Affects: Stored program flow

Q programs halt execution at blank lines. If a “no op” line is required in a program, for comments or other purposes, rather than leave the line blank the NO command is used. Think of the NO command as leaving a blank line in the middle of a sequence of commands. This is useful if after creating a sequence of commands you would like to delete a command without the line numbers of the remaining commands changing. Instead of deleting the line with the unwanted command, replace the unwanted command with a NO command and the remaining commands in the sequence will maintain their respective line numbers.

NOTE: NO commands are not required after the last command in a segment.

Command Details:

Structure	NO
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
NO	-	No operation takes place at this program line

OF - On Fault

Compatibility: Q drives only
Affects: Stored program flow
See also: AL, AX, AX, ME, OI commands

When a drive fault occurs, the OF command causes a given program segment to immediately load from non-volatile memory into the queue. The OF command acts as a kind of software switch in that when this function is turned on the drive's response to a drive fault (loading the designated program segment) is automatic. Once a fault occurs the fault must be cleared (AX) and the motor re-enabled (ME) before continuing normal program execution. Also, a drive fault will turn the OF function off, so after a fault the OF command must be executed again to reset the function. For this reason it is common to place the OF command in segment 1 of a Q program, and then load segment 1 (QX1) from the designated OF segment after the fault has been cleared and the motor re-enabled. A parameter value of "0" disables the On Fault function. See the AL (Alarm code) command for details of drive faults.

Command Details:

Structure	OF(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	No
Register Access	None

Parameter Details:

Parameter #1	program segment #
- units	integer
- range	1 - 12 = segment 1 - 12 0 = disable On Fault function

Examples:

Command	Drive sends	Notes
In segment 1 of a Q program...		
OF9	-	When a drive fault occurs load and execute program segment 9
In segment 9 of the same Q program...		
AX	-	Alarm reset
ME	-	Motor enable
QX1	-	Load and execute segment 1, which will also reset the OF function.
OF0	-	Disable the On Fault function

OI - On Input

Compatibility: Q drives only
Affects: Interrupt function and stored program flow
See also: OF command

When the given input condition is met the OI command causes program segment 10 to immediately load from non-volatile memory into the queue. The OI command operates as a kind of software switch. Executing the command turns the interrupt function on. Responding to the interrupt input (by loading segment 10) turns interrupt function off. Therefore after an interrupt condition is cleared in the system the OI command must be executed again to reset the interrupt function. One way to do this is place a copy of the OI command near the end of segment 10, before loading and executing another segment (QX command). Only one interrupt input can be defined at a time within a program. Executing the OI command with no parameter disables the interrupt function.

BLu, STAC6

An optional “X” in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	OI{Parameter #1}
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	No
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, Input number, Input condition NOTE: “X” designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates the encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge no parameter = disable interrupt function

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates the encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge no parameter = disable interrupt function

STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge no parameter = disable interrupt function

Examples:

Command	Drive sends	Notes
OIX5F	-	Load and execute program segment 10 when input X5 goes from high to low
OI	-	Disable interrupt function

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

OP - Option board

Compatibility: All drives
See also: IF, MV commands

Requests the decimal or hexadecimal equivalent (see IF command) of the option board's 5-bit binary word. Since some drives - like the SV, ST-Q/Si, and STM drives - are available with different option boards, it is useful for the host to be able to request this information from the drive. The 5 bits in the option board's binary word are shown below.

Bit 0: Encoder board
Bit 1: RS-485 board
Bit 2: CANopen board*
Bit 3: RESERVED
Bit 4: Resolver board*

*Contact Applied Motion at 800-525-1609 for release schedule.

Command Details:

Structure	OP
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	Yes
Register Access	None

Examples:

IF command set for decimal (IFD)...

Command	Drive sends	Notes
OP	OP3	Drive has both encoder and RS-485 option boards installed
OP	OP16	Drive has resolver board installed

IF command set for hexadecimal (IFH)...

Command	Drive sends	Notes
OP	OP0003	Drive has both encoder and RS-485 option boards installed
OP	OP0010	Drive has resolver board installed

PB - Power-up Baud Rate

Compatibility: All drives
See also: BR, PR, TD commands

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 an Applied Motion software application 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.

Command Details:

Structure	PB{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Baud rate code
- units	integer code
- range	1 = 9600 2 = 19200 3 = 38400 4 = 57600 5 = 115200

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

Compatibility: All drives
Affects: Motor current and torque
See also: CC, PI, PP commands

Servo drives

Sets or requests the continuous (RMS) current setting of the servo drive. This command differs from the CC command in that in addition to setting the continuous current of the drive, PC also immediately saves the setting to NV memory. In other words, PC = CC + SA.

Stepper drives

Sets or requests the current setting ("peak of sine") of the stepper drive, also known as the running current. This command differs from the CC command in that in addition to setting the running current of the drive, PC also immediately saves the setting to NV memory. In other words, PC = CC + SA.

Command Details:

Structure	PC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

BLu, SV

Parameter #1	Power-up continuous current setting
- units	amps rms (resolution is 0.01 amps)
- range	BLuDC4: 0 - 4.5 BLuDC9: 0 - 9.0 BLuAC5: 0 - 5.0 SV: 0 - 7.0

STAC6, ST-Q/Si, ST-S, STM

Parameter #1	Running current
- units	amps (resolution is 0.01 amps)
- range*	STAC6: 0 - 6.0 ST5 : 0 - 5.0 ST10: 0 - 10.0 STM: 0 - 5.0

*Current setting in stepper drives depends on the selected motor. Use *Configurator* software to select a motor and set the maximum current setting.

Examples:

Command	Drive sends	Notes
PC3.2	-	Set power-up continuous current to 3.2 amps RMS for servo drive or 3.2 amps running current for stepper drive
PC	PC=3.2	

PF - Position Fault

Compatibility: Servo drives and stepper drives with encoder feedback

Servo drives

Sets or requests the Position Fault limit in encoder counts. This value defines the limit threshold, in encoder counts, reached between actual position and commanded position before the system produces a position fault error.

Stepper drives:

Sets or requests the “percentage of torque” 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 *Configurator* software be used to make this setting.

Command Details:

Structure	PF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Servo: Position fault limit Stepper: Percentage of torque
- units	Servo: encoder counts Stepper: percentage of torque
- range	Servo: 1 - 32767 Stepper: 0 - 100 (percent)

Examples:

Command	Drive sends	Notes
PF2000	-	Set position fault limit to 2000 counts in servo drive
PF	PF=2000	
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

Compatibility: Stepper drives only
Affects: Motor current at standstill, holding torque
See also: CD, CI commands

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 differs from the CI command in that in addition to setting the idle current of the drive, PI also immediately saves the setting to NV memory. In other words, PI = CI + SA.

Command Details:

Structure	PI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

STAC6

Parameter #1	Idle current at power-up
- units	amps
- range	0 - 100% of running current

ST-Q/Si, ST-S, STM

Parameter #1	Idle current at power-up
- units	amps
- range	0 - 90% of running current

Examples:

Command	Drive sends	Notes
PC0.75	-	Set power-up idle current to 0.75 amps
PC	PC=0.75	

PL - Position Limit

Compatibility: Servo drives only
Affects: Motion Output function
See also: MO command

Sets or requests the count value used by the servo to determine if the motor is in position. This is used by the servo for determining the state of Motion Output (see MO command). When performing a move the Motion Output will be set to the designated condition until the servo is in position at the end of a move. The “In Position” status is set in the same way.

Command Details:

Structure	PL{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Positioning limit
- units	encoder counts
- range	0 - 32767

Examples:

Command	Drive sends	Notes
PL20	-	Set position limit to 20 counts
PL	PL=20	

PM - Power-up Mode

Compatibility: All drives
Affects: Serial communications
See also: CM command

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, for SCL applications set PM=2 or PM=5. The power-up mode is also set when configuring the drive with *Quick Tuner* or *Configurator*.

Q drives

When creating Q Programs for your Q drive, checking the "Execute "Q" at Power-up" box on the main screen of the Q Programmer software will change the power-up mode of the drive to 7 (PM7) with the next download. This will cause the drive to run its stored Q Program at power-up. You must download the program after checking this box for the change to take effect.

Command Details:

Structure	PM{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Power on mode
- units	integer code
- range	1 = Si Program (Si versions only) 2 = Host command (drive enabled) 3 = Quick Tuner (servos) or Configurator (steppers) 4 = SiNet Hub 5 = Host command (drive disabled) 6 = not used 7 = Q Program (Q versions only)

Examples:

Command	Drive sends	Notes
PM1	-	Si drive will power up in Si Program mode
PM2	-	Drive will power up in Host / SCL mode
PM	PM=2	

PP - Power-up Peak current

Compatibility: Servo drives only
Affects: Motor current, especially during acceleration and deceleration
See also: CC, CP, PC commands

Sets or requests the power-up peak (RMS) current setting of the servo drive. This command differs from the CP command in that in addition to setting the peak current of the drive, PP also immediately saves the setting to NV memory. In other words, PP = CP + SA.

Command Details:

Structure	PP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

Parameter #1	Power-up peak current limit
- units	amps RMS (resolution is 0.01 amps)
- range	BLuDC4: 0 - 13.5 BLuDC9: 0 - 18.0 BLuAC5: 0 - 15.0 SV7: 0 - 7.0

Examples:

Command	Drive sends	Notes
PP6	-	Set power-up peak current to 6.0 amps RMS
PP	PP=6	

PR - Protocol

Compatibility: All drives
Affects: RS-232 & RS-485 Serial Communications
See also: BR and PB commands

Sets or requests the serial communication protocol settings. 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 6-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.

Command Details:

Structure	PR{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Protocol code
- units	decimal (integer) value of binary word
- range	1 - 63 (000001 - 111111) bit 0 = Default (“Standard SCL”) bit 1 = Always use Address Character bit 2 = Ack/Nack bit 3 = Checksum (RESERVED) bit 4 = RS-485 Adaptor bit 5 = 3-digit numeric register addressing

*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.

Examples:

Command	Drive sends	Notes
PR1	-	Set to standard SCL protocol
PR4	-	Turn Ack/Nack on
PR	PR=4	

PS - Pause

Compatibility: All drives
See also: BS, CT command

Suspends execution of buffered commands until the next CT (Continue) command is executed. This can be useful for coordinating motion among axes by first pausing (PS) the drives, then loading the drives' command buffers with commands, and then resuming command execution (CT) in all drives at once. PS can also 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.

NOTE: It is possible to overflow the command buffer. Use the BS (Buffer Status) command to view how many command spaces are vacant in the buffer at any given time.

Command Details:

Structure	PS
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
PS	-	Pause execution of buffered commands

PW - Password

Compatibility: Q drives only

Normally the stored program of a Q drive can be uploaded and downloaded at will. This allows basically any user to access the stored program of a Q drive. To password-protect the stored program of a Q drive the PW command can be issued with a customized key code.

The factory default key code is “1234”, which allows uploading and downloading programs freely. To password-protect a stored program the user should enter the PW command with a new key code. This new key code can be any 4 character alpha-numeric code (characters A-Z, a-z, and 0-9 are acceptable). After entering the new key code the user must enter the SA (Save) command for the new key code to be saved in the drive. Then, the next time the drive is powered up password-protection will take effect, which means the user must first “unlock” the drive by sending the PW command with the customized key code before being able to upload (QU), save (QS), or delete (QD) any part of the Q drive’s stored program. (All other immediate commands function even if the drive is not “unlocked”). Furthermore, every subsequent power-up of the drive will require the same key code to be entered before uploading. To change the key code, enter the present key code at power up and then use the PW command to enter a new key code followed by the SA command. To return the drive to the default state of no-password protection, unlock the drive first by using the present key code, then enter the default key code of “1234” followed by the SA command.

NOTE: If the key code is forgotten or lost, re-entering the default code of “1234” will unlock the drive and ERASE THE CONTENTS OF THE DRIVE’S NON-VOLATILE MEMORY AT THE SAME TIME.

Command Details:

Structure	PW(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	4-digit alphanumeric key code
- units	upper and lower-case letters and numbers
- range	A-Z, a-z, 0-9
- default	Default key code is “1234”

Examples:

Command	Drive sends	Notes
PWak99	-	Password key code set to “ak99”
SA	-	New key code saved in drive
PWak99	-	Access to stored program unlocked at next power-up of drive

QC - Queue Call

Compatibility: Q drives only
See also: QX, QE, QL commands

Loads a program segment from non-volatile memory into the queue and begins executing at line #1. Loads the "calling" segment location and the line where the call occurred into a call "stack". When a QC command without a parameter is encountered in the segment a "return" to the calling segment is activated. The "calling" segment is loaded from non-volatile memory back into the queue and begins executing at the line immediately following the line number of the original "calling" QC command.

The call stack can go 5 calls deep which means you can nest up to 5 calls. If the number of calls before a "return" (QC with no parameter) is encountered exceeds 5 the "calling" QC command (with parameter) is ignored. If a "return" is encountered without a previous call, the return is ignored.

Command Details:

Structure	QC{Parameter #1}
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Program segment, no parameter means return to calling segment
- units	integer
- range	1 - 12 = segment 1 - 12 no parameter = return to calling segment

Examples:

Command	Drive sends	Notes
QC4	-	Call segment 4
QC	-	Return to calling segment

QD - Queue Delete

Compatibility: Q drives only
See also: QL, QS, PW commands

Deletes the contents of the non-volatile memory location associated with a particular program segment.

Command Details:

Structure	QD(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Segment number
- units	integer number
- range	1 - 12

Examples:

Command	Drive sends	Notes
QD5	-	Delete program segment 5 from the drive's non-volatile memory

QE - Queue Execute

Compatibility: Q drives only
See also: QL, QX

Begins executing a program segment previously loaded into the queue. Starts executing at line #1. A segment must have previously been loaded using the “QL” or “QX” commands.

Command Details:

Structure	QE
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
QE	-	Begin execution of loaded segment

QG - Queue Goto

Compatibility: Q drives only
See also: QJ, QR

Causes program segment execution to jump to the given line number in the queue. Gotos directed to the same line number as the QG command or past the end of the queue are ignored.

Command Details:

Structure	QG(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Segment line number
- units	integer
- range	1 - 62

Examples:

Command	Drive sends	Notes
QG10	-	Cause a jump to line 10 in the segment

QJ - Queue Jump

Compatibility: Q drives only
Affects: Program flow
See also: QG, TI, TR, CR and all Math commands (“R” commands)

Causes program segment execution to jump to the given line number in the queue based on a “condition code”. Jumps directed to the same line number as the QJ command or past the end of the queue are ignored. If the condition code is met the jump occurs, if not the program proceeds to the next line. Condition codes are set by previous commands such as the TI (Test Input) or TR (Test Register) commands. When using math commands (“R” commands) the condition code is set based on the result of the math operation.

Command Details:

Structure	QJ(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Condition code
- units	letter
- range	T = True F = False P = Positive G = Greater than L = Less than E = Equals U = Unequal Z = Zero
Parameter #2	Segment line number
- units	integer
- range	1 - 62

Examples:

Command	Drive sends	Notes
TI4L	-	Test input 4 to see if it’s low (active)
QJT15	-	Jump to line 15 if condition code is “True” (i.e. input 4 is low)

QK - Queue Kill

Compatibility: Q drives only
Affects: Queue execution and program flow
See also: SK

Halts execution of the queue. The queue contents are not affected and can be executed again using the “QE” command.

Command Details:

Structure	QK
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
QK	-	Stop execution of the queue/program

QL - Queue Load

Compatibility: Q drives only
Affects: Contents of command buffer
See also: QE, QS, QX commands

Initiates the loading of a command sequence into the queue. Loading can come from the serial port (host controller) or from non-volatile memory (stored program). When no parameter is sent with the command loading is done from the serial port. Loading is finished when a QS (Queue Save) or QE (Queue Execute) command is sent. When a parameter is sent with the command the parameter designates the non-volatile memory location of the desired program segment to be loaded into the queue. QL will cause an overwrite of any commands in the queue starting at line #1.

Command Details:

Structure	QL{Parameter #1}
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Program segment number
- units	integer
- range	1 - 12

Examples:

Command	Drive sends	Notes
QL	-	Initiates loading queue from serial port
QL3	-	Loads segment from non-volatile memory into the queue

QR - Queue Repeat

Compatibility: Q drives only
Affects: Selected data register
See also: QJ, QG, RL, RX commands

Causes program segment execution to jump to a previous line number in the queue for a given number of repeat counts. The repeat count is acquired from a selected Read/Write or User-Defined data register. Jumps past the end of the queue are ignored. Jumps to subsequent line numbers in the queue will not be repeated. If repeat count is set to "1" no Jump is performed. The data register selected for the repeat count must be set with the number of repeat counts prior to using the QR command: use the RX (Register Load - buffered) command to load the data register with the repeat count. The data register contents are affected by this command and must be re-loaded before each usage with the QR command.

NOTE: Although data registers A - Z can be used with the QR command it is not recommended. The QR command eventually destructs the data in a register by decrementing its value each time a jump is made in the repeat loop and could therefore lead to unexpected results in other parts of the program that make use of data registers A - Z.

Command Details:

Structure	QR(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Data register
- units	register character
- range	0 - 16, A - Z
Parameter #2	Segment line number
- units	integer
- range	1 - 62

Examples:

Command	Drive sends	Notes
RX120	-	Load user register #1 with the value 20
QR15	-	Cause a repeated jump to line 5 of the queue using the value (20) in data register #1 as the repeat count

QS - Queue Save

Compatibility: Q drives only
Affects: None
See also: QE, QL, QX, PW commands

Saves a segment currently in the queue to a non-volatile memory location. Ends a QL (Queue Load) if one is in progress. See Appendix B for more details on this command, including its limitations.

Command Details:

Structure	QS(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Program segment location
- units	integer
- range	1 - 12

Examples:

Command	Drive sends	Notes
QS2	-	Save contents of queue to non-volatile memory location #2

QU - Queue Upload

Compatibility: Q drives only
See also: QL, QE, QS, QX, PW commands

Uploads a stored program segment from the drive's non-volatile memory to the serial port. This command is used to retrieve stored program segments from the drive. When using this command the drive responds with either a "positive" or a "negative" response. A "positive" response consists of a normal acknowledge ("ack") followed by the contents of the requested program segment with each line number separated by a carriage return. Program segments 1 to 12 can be uploaded, as well as the contents of the queue. A "negative" response from the drive consists of a normal acknowledge ("ack") followed by one of four error codes: 0, -1, -2, or -3. *A negative acknowledge ("nack" / "?") will be sent from the drive if the command is not understood by the drive. See Appendix D for more information on acknowledge and negative acknowledge responses.*

Positive response format:

"ack" ("%")
First line = "QU##" where "##" = the number of lines in the segment + 1
Second line = command at line 1 of the segment
Next line = command at line 2 of the segment
...
Last line = command at last line of segment

Negative Responses:

"ack" ("%")
QU0 = No segment at specified location
QU-1 = Program Running (Cannot upload at this time)
QU-2 = Upload currently in process
QU-3 = Password Protected (Protection must be unlocked using PW command)

Command Details:

Structure	QU(Parameter #1)
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Program segment location
- units	integer
- range	1 - 12, or 0 to upload queue

Examples:

Command	Drive sends	Notes
QU0	(queue contents)	Uploads contents of queue to the serial port
QU3	(segment #3 contents)	Uploads contents of segment from non-volatile memory location #3

QX - Queue Load & Execute

Compatibility: Q drives only
Affects: Stored program flow
See also: QE, QL

Loads a program segment from non-volatile memory into the queue. Begins executing the segment at line #1. This is the similar to the combination of a QL (Queue Load) and a QE (Queue Execute) command with the difference being the QX command can be written into a stored program segment. Use this command to “jump” from segment to segment.

Command Details:

Structure	QX(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Program segment location
- units	integer
- range	1 - 12

Examples:

Command	Drive sends	Notes
QX2	-	Loads segment #2 and begins execution

RC - Register Counter

Compatibility: Q drives only
Affects: Data Register “I” (025)
See also: RL, RX , RI, RD, TS commands

This command enables a function that increments the “I” data register when the given input condition (determined by the RC command) is met. Typically the “R” or “F” input condition (see Details below) is used to trigger an increment. If however the “L” or “H” input condition is used the register will be incremented at a rate of 8000 times per second. In other words the “R” and “F” input conditions are used for true input counting while the “L” and “H” conditions act as input timers. Use the RL (Register Load - immediate) or RX (Register Load - buffered) commands to preset or set the “I” data register to a predetermined value.

This command is also used in conjunction with the TS (Time Stamp) command. See the TS command for more details.

BLu, STAC6

An optional “X” in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	RC(Parameter #1)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“I” (025)

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, Input number, Input condition NOTE: “X” designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates the encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates the encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

STM

Parameter #1	Input number, Input condition

- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
RC4F	-	Increment data register “I” each time input 4 changes from high to low (falling edge)

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, 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 low”, and SO1L means “set output 1 low”. A high state is represented by the “H” character.*

NOTE: 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.

RD - Register Decrement

Compatibility: Q drives only
Affects: All data registers
See also: RI, RM

Decrements by 1 the value of the designated data register.

Command Details:

Structure	RD(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All Read/Write and User-Defined data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	all Read/Write and User-Defined data registers

Examples:

Command	Drive sends	Notes
RDV	-	Decrements the value of the velocity data register “V”

RE - Restart or Reset

Compatibility: All drives

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.

Command Details:

Structure	RE
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
RE	-	Resets drive condition and parameters

RI - Register Increment

Compatibility: Q drives only
Affects: All data registers
See also: RD, RM commands

Increments by 1 the value of the designated data register.

Command Details:

Structure	RI(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All Read/Write and User-Defined data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	all Read/Write and User-Defined data registers

Examples:

Command	Drive sends	Notes
RIV	-	Increments the value of the velocity data register “V”

RL - Register Load - immediate

Compatibility: All drives
Affects: All data registers
See also: RX, RI, RD, RM commands

Sets a data register to the given immediate data value. The data value is checked and stored as a Long word. When setting a Short-word register with the given Long-word data value only the lower word of the Long value is used.

Command Details:

Structure	RL(Paramter #1){Parameter #2}
Type	IMMEDIATE
Usage	READ/WRITE
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	All Read/Write and User-Defined data registers; Read-Only data registers can be read back when Parameter #2 is not included
Parameter #2	Data register value
- units	integer
- range	+/- 2147483647 (long data registers) +/- 32767 (short data registers)

Examples:

Command	Drive sends	Notes
RLA100	-	Set acceleration register to 1000 rpm/s
RLA	RLA=100	Return acceleration register value
When PR command Bit 5 is set		
RL017100	-	Set Acceleration register to 1000 rpm/s
RL017	RLA=100	Return acceleration register value

NOTE: When setting a register no pre-processing of the data value is performed. Data is set to the internal raw value. For example, the internal raw acceleration value is in tens of rpm/s. See the “Data Register” section at the beginning of this manual for more details on data register assignments and units.

Units Example:
AC10 means 10 rps/s
RLA10 means 10 * 10 rpm/s = 1.667 rps/s
Multiply the desired rps/s value times 6 to convert to the “raw” acceleration value

RM - Register Move

Compatibility: Q drives only
Affects: All data registers
See also: RI, RD, RL, TR, RX commands

Move the contents of a first data register into a second data register.

Command Details:

Structure	RM(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All Read/Write and User-Defined data registers

Parameter Details:

Parameter #1	Source data register
- units	character
- range	all data register assignments
Parameter #2	Destination data register
- units	character
- range	all Read/Write and User-Defined data registers

Examples:

Command	Drive sends	Notes
RMAB	-	Move contents of acceleration register “A” into the deceleration register “B”

RR - Register Read

Compatibility: Q drives only
Affects: All data registers
See also: RI, RD, RL, RM, RW commands

Reads a data value from a non-volatile memory location into a data register. The data value is read as a Long word. If the value being read is too large for the destination data register, the value is truncated.

Command Details:

Structure	RR(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All Read/Write and User-Defined data registers

Parameter Details:

Parameter #1	Data register
- units	character
- range	all Read/Write and User-Defined data registers
Parameter #2	Non-volatile memory location
- units	integer
- range	1 - 100

Examples:

Command	Drive sends	Notes
RRV10	-	Read data from non-volatile memory location #10 and place it in data register “V”

RS - Request Status

Compatibility: All drives
See also: SC command

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.

Command Details:

Structure	RS
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
RS	RS=PR	Motor is in position, drive is enabled
RS	RS=AED	Alarm code is present, drive is faulted and disabled
RS	RS=JR	Motor is jogging, drive is enabled

Status character codes:
A = An Alarm code is present (use AL command to see code, AR command to clear code)
D = Disabled (the drive is disabled)
E = Drive Fault (drive must be reset by AR command to clear this fault)
H = Homing (SH in progress)
J = Jogging (CJ in progress)
M = Motion in progress (Feed & Jog Commands)
P = In position
R = Ready (Drive is enabled and ready)
S = Stopping a motion (ST or SK command executing)
T = Wait Time (WT command executing)
W = Wait Input (WI command executing)

RU - Register Upload

Compatibility: Q drives only
Affects: All data registers
See also: PR, RL, RX commands

Upload the contents of an array of data registers. Up to 16 registers can be read back with one RU command. Each reading is terminated with a carriage return.

Command Details:

Structure	RU(Parameter #1)(Parameter #2)
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	all data registers
Parameter #2	Number of consecutive data registers to upload
- units	integer
- range	1 - 16

Examples:

Command	Drive sends	Notes
RUA5	RUA=400	“A” The Acceleration value
	RUB=400	“B” The Deceleration value
	RUC=16000	“C” The Distance Change value
	RUD=8000	“D” The Distance value
	RUE=0	“E” The Encoder value
When PR command Bit 5 is set		
RU0175	RUA=400	“017” The Acceleration value
	RUB=400	“018” The Deceleration value
	RUC=16000	“019” The Distance Change value
	RUD=8000	“020” The Distance value
	RUE=0	“021” The Encoder value

NOTE: All Data values are “raw” meaning the data is not scaled to the drive user units. For example the velocity value (“V”) will be returned as 0.25 rpm instead of rps: raw value of 2400 = 10 rps.

RV - Revision Level

Compatibility: All drives
See also: MV command

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.

Command Details:

Structure	RV
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	Drive firmware version

Examples:

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

RW - Register Write

Compatibility: Q drives only
Affects: All data registers
See also: RI, RD, RL, RM, RW commands

Writes the data value of a given data register into non-volatile memory. The data value is written as a Long word. See Appendix B for more details on this command, including its limitations.

Command Details:

Structure	RW(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All Read/Write and User-Defined data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	all Read/Write and User-Defined data registers
Parameter #2	Non-volatile memory location
- units	integer
- range	1 - 100

Examples:

Command	Drive sends	Notes
RWV10	-	Write data from data register “V” into non-volatile memory location #10

RX - Register Load - buffered

Compatibility: Q drives only
Affects: All data registers
See also: RL, RU, RM commands

Sets a data register to the given immediate data value. The data value is checked and stored as a Long word. When loading a Short-word data register with the given Long-word data value only the lower word of the Long value is used. This command is the same as the RL command except it is a buffered command and therefore can be placed in a stored program.

Command Details:

Structure	RX(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	Data register assignment
- units	character
- range	All Read/Write and User-Defined data registers; Read-Only data registers can be used when Parameter #2 is not included (i.e. for reading back the contents of a Read-Only data register)
Parameter #2	Data register value
- units	integer
- range	+/- 2147483647 (long data registers) +/- 32767 (short data registers)

Examples:

Command	Drive sends	Notes
RXA100	-	Set acceleration register “A” to 1000 rpm/s
RXA	RXA=100	

R+ - Register Add

Compatibility: Q drives only
Affects: All data registers
See also: R-, R*, R/, R&, RD, RI, QJ commands

Adds the contents of a first data register to a second data register and places the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation: adding two Long word values can cause an overflow.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. R+ can set condition codes T, F, N, P, and Z

Command Details:

Structure	R+(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register assignment
- units	character
- range	all data registers
Parameter #2	Second data register assignment
- units	character
- range	all data registers

Examples:

Command	Drive sends	Notes
R+D1	-	Add contents of distance register “D” to user-defined register “1” and place the result in the accumulator register “0”

R- - Register Subtract

Compatibility: Q drives only
Affects: All data registers
See also: R+, R*, R/, R&, RD, RI, QJ commands

Subtracts the contents of the second data register from the first data register and places the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation: subtracting two Long word values can cause an underflow.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. Can set condition codes T, F, N, P, and Z.

Command Details:

Structure	R-(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register assignment
- units	character
- range	all data registers
Parameter #2	Second data register assignment
- units	character
- range	all data registers

Examples:

Command	Drive sends	Notes
R-D1	-	Subtract the contents of user-defined register “1” from the distance register “D” and place the result in the accumulator register “0”

R* - Register Multiply

Compatibility: Q drives only
Affects: All data registers
See also: R+, R-, R/, R&, RD, RI, QJ commands

Multiply the contents of the first data register by the second data register and place the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation: multiplying two Long word values can cause an overflow.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. Can set condition codes T, F, N, P, and Z.

Command Details:

Structure	R*(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register assignment
- units	character
- range	all data registers
Parameter #2	Second data register assignment
- units	character
- range	all data registers

Examples:

Command	Drive sends	Notes
R*D1	-	Multiply contents of distance register “D” by contents of user-defined register “1” and place result in accumulator register “0”

R/ - Register Divide

Compatibility: Q drives only
Affects: All data registers
See also: R+, R-, R*, R&, RD, RI, QJ commands

Divide the contents of the first data register by the second data register and place the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation. A value of “zero” in the second data register will cause an illegal “divide by zero”, in which case the divide operation is ignored.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. Can set condition codes T, F, N, P, and Z.

Command Details:

Structure	R/(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register
- units	data register assignment
- range	All data registers
Parameter #2	Second data register
- units	data register assignment
- range	All data registers

Examples:

Command	Drive sends	Notes
R/D1	-	Divide contents of distance register “D” by user-defined register “1” and place result in accumulator register “0”

R& - Register AND

Compatibility: Q drives only
Affects: All data registers
See also: R+, R-, R*, R/, RD, RI, QJ commands

Do a “bit-wise” AND of the contents of the first data register with the contents of the second data register and place the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation. This operation affects the “condition code” use by the QJ (Queue Jump) command.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. Can set condition codes T, F, N, P, and Z.

Command Details:

Structure	R&(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register
- units	data register assignment
- range	All data registers
Parameter #2	Second data register
- units	data register assignment
- range	All data registers

Examples:

Command	Drive sends	Notes
R&s1	-	AND the contents of status register “s” and user-defined register “1” and place the result in accumulator register “0”

R| - Register OR

Compatibility: Q drives only
Affects: All data registers
See also: R+, R-, R*, R/, R&, RD, RI, QJ commands

Do a “bit-wise” OR of the contents of the first data register with the contents of the second data register and place the result in the accumulator data register, User-Defined register “0”. This is a 32-bit operation.

All math operations affect the “condition code” used by the QJ (Queue Jump) command. Can set condition codes T, F, N, P, and Z.

Command Details:

Structure	RI(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“0” (000), Accumulator

Parameter Details:

Parameter #1	First data register
- units	data register assignment
- range	All data registers
Parameter #2	Second data register
- units	data register assignment
- range	All data registers

Examples:

Command	Drive sends	Notes
R|i1	-	OR the contents of inputs register “i” with user-defined register “1” and place the results in accumulator register “0”

SA - Save Parameters

Compatibility: All drives
See Also: RE command

Saves selected command parameters to non-volatile memory. This command is useful for setting up the drive configuration with the desired defaults at power-up. (See which commands are non-volatile in the Command Summary section.)

Command Details:

Structure	SA
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
SA	-	Save all Non-Volatile-designated data registers

SC - Status Code

Compatibility: All drives
See also: RS command

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 E for more details on the Status Code.

Command Details:

Structure	SC
Type	IMMEDIATE
Usage	READ ONLY
Non-Volatile	NO
Register Access	None
Units	Hexadecimal equivalent of the binary status code word (see bit assignments below)

Response Details:

Hex Value	Status Code bit definition
0001	Motor Enabled (Motor Disabled if this bit = 0)
0002	Sampling (for Quick Tuner)
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)
0100	Saving (parameter data is being saved)
0200	Alarm present (check Alarm Code)
0400	Homing (executing an SH command)
0800	Wait Time (executing a WT command)
1000	Wizard running (Timing Wizard is running)
2000	Checking encoder (Timing Wizard is running)
4000	Q Program is running
8000	Initializing (happens at power up)

Examples:

Command	Drive sends	Notes
SC	SC=0009	Drive is in position 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 enabled (hex values 0001, 0008, and 0200)

SF - Step Filter Frequency

Compatibility: Stepper drives only

Sets or requests the step filter frequency. The primary use of this filter is to introduce “microstep emulation” effects, which smooth out low resolution step pulses when the drive’s microstep/gearing resolution (EG command) is set to a low value. This command is exceptionally useful when using a low-resolution indexer with a STAC stepper drive and smooth motor shaft rotation is required.

Command Details:

Structure	SF{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Step filter frequency
- units	Hz
- range	0 - 16000

Examples:

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

SH - Seek Home

Compatibility: All drives
See Also: DL, FD, FS, FY commands

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 drive "sees" 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 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.

NOTE: This command is designed for use with three physical sensors or switches tied to three separate digital inputs of the drive: a home sensor, a CW end-of travel limit, and a CCW end-of-travel limit.

BLu, STAC6
An optional "X" in Parameter #1 is used to designate inputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	SH{Parameter #1}
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "X", Input number, Input condition NOTE: "X" designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "X", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X5, X8, X9, X: (X0 designates encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition

- units	integer, letter
- range	integer: 0 - 6, 9, :, ; (0 designates encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
SH1L	-	Seek home to input 1 low
SH3R	-	Seek home to input 3 rising edge
SHX5L	-	Seek home to input X5 low (main driver board input)

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

SI - Enable Input Usage

Compatibility: All drives
Affects: Enable input usage
See also: AI, CM commands

BLu, SV, STAC6, ST-Q/Si

Sets or requests the usage of the Enable input. Input X3 is the default Enable input on all drives, however IN3 on SE, QE and Si drives may also be designated as the Enable input (see below). If an external Enable function is not needed input X3 and/or IN3 can be defined solely as a general purpose input. In this scenario only the ME and MD commands will enable and disable the drive, respectively. When using the brake output (see BO, BD and BE commands) the disabling of the drive is delayed by the time value set using the BD command.

There are five possible usage states for the Enable function:

SI1: Drive is enabled when X3 is open (inactive, high).
SI2: Drive is enabled when X3 is closed (active, low).
SI3: Neither X3 nor IN3 is used for enabling/disabling the drive, but as general purpose inputs.
SI4: Drive is enabled when IN3 is open (inactive, high). (SE, QE, and Si drives only).
SI5: Drive is enabled when IN3 is closed (active, low). (SE, QE, and Si drives only).

ST-S, STM

Defines the EN input as an Enable Input. The EN input can be assigned to only one function in an application. If you want to use the EN input as an Enable input you can define it as such in two ways, with the *ST Configurator* 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 drive enable usage and turns off any alarm reset usage (SI3). In other words, the SI and AI commands, as well as Command Modes (CM) 13, 14, 17 and 18 each assign a usage to the EN input. Each of these must exclusively use the EN input. Command Modes are most dominant and will continually prevent SI and AI from using the input. SI and AI exclude each other by overwriting the usage of the EN input.

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.

Command Details:

Structure	SI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Enable input usage state
- units	integer
- range	1, 2, 3, 4 or 5 (see above)

Examples:

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

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

SJ - Stop Jogging

Compatibility: All drives
Affects: CJ command
See Also: JA, CJ, ST, SK, SM commands

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

Command Details:

Structure	SJ
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

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

SK - Stop & Kill

Compatibility: All drives
See Also: AM, DE, JA, ST, SM commands

Halts any buffered command in progress and removes any other buffered commands from the queue. 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).

Command Details:

Structure	SK{Parameter #1}
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Deceleration rate
- units	letter
- range	D = deceleration rate set by DE or JA command 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 queue
SKD	-	Stop motion immediately using the deceleration rate set by the DE command (or JA if jogging) and erase the contents of the queue

SM - Stop Move

Compatibility: Q drives only
See also: AM, DE, JL, SK, ST, QK commands

Stops any type of move in progress such as FL or CJ. This command acts like the ST (Stop) command except it will not stop a wait operation (like WD, WI, WP, or WT) and it can be part of a stored Q program. The contents of the queue are not affected by the SM command

Command Details:

Structure	SM(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Deceleration rate
- units	letter
- range	D = deceleration rate set by DE command or JL command (if jogging) M = deceleration rate set by AM command

Examples:

Command	Drive sends	Notes
SMD	-	Stop motion immediately using the deceleration rate set by the DE command or the JL command (if jogging)
SMM	-	Stop motion immediately using the deceleration rate set by the AM command

SO - Set Output

Compatibility: All drives
See Also: IL, IH, IO commands

Sets an output to the given condition. Care must be taken when using outputs on the main driver board because those outputs are by default programmed for dedicated purposes (Alarm, Brake, Motion). Use the AO, BO and MO commands to reconfigure main driver board output usage to general purpose before using the SO command with those outputs.

BLu, STAC6
An optional "Y" in Parameter #1 is used to designate outputs on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	SO(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "Y", Output number, Output condition NOTE: "Y" designates that the output in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "Y", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: Y1 - Y3 - letter: L = low (closed), H = high (open) IN/OUT 2 or top board (screw terminal) - integer: 1 - 4 - letter: L = low (closed), H = high (open)

SV, ST-Q/Si

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 - 4 letter: L = low (closed), H = high (open)

ST-S, STM

Parameter #1	Output number, Output condition
- units	integer, letter
- range	integer: 1 letter: L = low (closed), H = high (open)

Examples:

Command	Drive sends	Notes
SO1L	-	Set output 1 low (closed)

SO2H	-	Set output 2 high (open)
SOY1L	-	Set main driver board output 1 low (closed)
SOY2H	-	Set main driver board output 2 high (open)

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

SP - Set Position

Compatibility: All drives
Affects: FP commands
See Also: EP, FP commands

Sets or requests the motor's absolute position.

Command Details:

Structure	SP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Absolute position
- units	encoder counts
- range	+/- 2,147,483,647

Examples:

Command	Drive sends	Notes
SP100	-	Set absolute position offset to 100 encoder counts
SP	SP=100	

SS - Send String

Compatibility: All drives

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 queue 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 Details:

Structure	SS(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	String of characters
- units	any printable characters
- range	up to 4 characters

Examples:

Command	Drive sends	Notes
SSdone	done	String “done” sent when SS command is executed

ST - Stop

Compatibility: All drives
See Also: SK, SM commands

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.

Command Details:

Structure	ST{Parameter #1}
Type	IMMEDIATE
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Deceleration rate
- units	letter
- range	D = deceleration rate set by DE or JA command no parameter = deceleration rate set by AM command

Examples:

Command	Drive sends	Notes
ST	-	Stop motion immediately using the deceleration rate set by the AM command
STD	-	Stop motion immediately using the deceleration rate set by the DE or JA command

TD - Transmit Delay

Compatibility: All drives
Affects: RS-232 & RS-485 Serial Communications
See Also: BR, PB & PR commands

Sets or requests the time delay used by the drive when responding to a command that requests a response. Typically this is needed when using the 2-wire RS-485 interface (Half-duplex). Because the same wires are used for both receive and transmit a time delay is usually needed to allow transition time. The Host device's RS-485 specification must be understood to determine the time delay needed.

Command Details:

Structure	TD{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Time value
- units	milliseconds
- range	0 - 32767
- default	10

Examples:

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

TI - Test Input

Compatibility: Q drives only
Affects: Condition Code
See also: QJ, TR commands

Tests a digital or analog input against the given input state. If the input is in the state the condition code is set to "T" (true). If not the condition code is set to "F" (false).

BLu, STAC6
An "X" in Parameter #1 is used to designate an input on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	TI(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional "X", Input number, Input condition NOTE: "X" designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional "X", integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates the encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates the encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
TI4L	-	Test input 4 for a low state

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

TR - Test Register

Compatibility: Q drives only
Affects: All data registers
See also: CR, TI, RI, RD, RM, RL, QJ commands

Tests a data register against a given data value. The result of the test is the setting of the condition code, which can be used for conditional programming (see QJ command).

All conditions codes can be set by this command. See “QJ” command for more details.

Command Details:

Structure	TR(Parameter #1)(Parameter #2)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	All data registers

Parameter Details:

Parameter #1	Data register
- units	data register assignment
- range	All data registers
Parameter #2	Test value
- units	integer
- range	+/- 2,147,483,647 (long data registers) +/- 32,767 (short data registers)

Examples:

Command	Drive sends	Notes
TR15	-	Test user-defined register “1” against the value 5

TS - Time Stamp

Compatibility: Q drives only
Affects: Data Register “W”
See also: RC, WD, All register commands

Transfers the oldest Time Stamp value from the time stamp buffer into the “W” data register. The time stamp value is a time value in milliseconds, recorded between two input triggers when using the RC command. Each time a defined input condition is “True” (triggered), the elapsed time from the previous input is store in the time stamp buffer. The time stamp buffer is 8 words deep and acts as a FIFO buffer. The “I” data register, used by the RC command, records when an input trigger has occurred. Sending the RC command clears the time stamp buffer. Executing the TS command removes the oldest time value from the time stamp buffer and places it in the “W” Data Register where it can be used. With each execution of the TS command a time value is transferred, until the end of the time stamp buffer is reached. If a TS is sent with no time values in the time stamp buffer a “0” is placed in the “W” data register.

Command Details:

Structure	TS
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	“W” (039)
Units	milliseconds

Examples:

Command	Drive sends	Notes
TS	-	Move time stamp value from buffer into read/write data register “W”

VC - Velocity Change

Compatibility: All drives
Affects: FC, FD commands

Sets or requests the “change speed” for FC and FD moves..

Command Details:

Structure	VC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	“U” (037)

Parameter Details:

Parameter #1	Move velocity
- units	rev/sec
- range	BLu, SV, STAC6, ST-Q/Si, ST-S: 0.0042 - 133.3333 (resolution is 0.0042) STM: 0.0042 - 80.0000 (resolution is 0.0042)

Examples:

Command	Drive sends	Notes
VC5	-	Set change velocity to 5 rev/sec
VC	VC=5	

VE - Velocity

Compatibility: All drives
Affects: FC, FD, FE, FL, FM, FS, FP, FY, SH commands

Sets or requests shaft speed for point-to-point move commands like FL, FP, FS, FD, SH, etc.

Command Details:

Structure	VE{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	“V” (038)

Parameter Details:

Parameter #1	Move velocity
- units	rev/sec
- range	BLu, SV, STAC6, ST-Q/Si, ST-S: 0.0042 - 133.3333 (resolution is 0.0042) STM: 0.0042 - 80.0000 (resolution is 0.0042)

Examples:

Command	Drive sends	Notes
VE2.525	-	Set move velocity to 2.525 rev/sec
VE	VE=2.525	

VI - Velocity Integrator Constant

Compatibility: Servo drives only
Affects: Jog commands
See also: VP & JM commands

Sets or requests the velocity-mode (“JM2”) servo control integrator gain term. Gain value is relative: 0 = no gain, 32767 = full gain. VI minimizes steady state velocity errors.

Command Details:

Structure	VI{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Velocity integrator gain value
- units	integer (no specific units)
- range	0 - 32767 (0% - 100%)

Examples:

Command	Drive sends	Notes
VI5000	-	Set velocity integrator gain to 5000
VI	VI=5000	

VM - Maximum Velocity

Compatibility: All drives
Affects: Analog Velocity mode
See Also: AM, VC, VE commands

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

Command Details:

Structure	VM{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	YES
Register Access	None

Parameter Details:

BLu, SV, STAC6, ST-Q/Si

Parameter #1	Move velocity
- units	rev/sec (rps)
- range	0.0042 - 133.3333 (resolution is 0.0042 rev/sec)

ST-S, STM

Parameter #1	Move velocity
- units	rev/sec (rps)
- range	0.0042 - 80.0000 (resolution is 0.0042 rev/sec)

Examples:

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

VP - Velocity Mode Proportional Constant

Compatibility: Servo drives only
Affects: Jog commands
See also: VI & JM commands

Sets or requests the velocity-mode servo control Proportional gain term. Gain value is relative: 0 = no gain, 32767 = full gain. VP minimizes velocity error when in velocity mode 2 (see JM command).

Command Details:

Structure	VP{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Velocity mode proportional gain
- units	integer
- range	0 - 32767 (0% - 100%)

Examples:

Command	Drive sends	Notes
VP5000	-	Set velocity mode proportional gain to 5000
VP	VP=5000	

WD - Wait Delay

Compatibility: Q drives only
Affects: None
See also: WI, RX commands

Causes a time delay to occur using a time value from a given data register. The resolution is in milliseconds. Only up to 15 bits of the data register are used, giving a maximum wait time of 32 seconds.

Command Details:

Structure	WD(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Data register
- units	data register assignment
- range	All Read/Write and User-Defined data registers

Examples:

Command	Drive sends	Notes
WD5	-	Wait the number of milliseconds indicated by the value in user-defined data register “5”

WI - Wait for Input

Compatibility: All drives
Affects: Use of “Jog” Inputs
See Also: FI, JE, JD, WD, WM, TI commands

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) is active the drive’s “jog” inputs can be used to jog the motor. JD disables jogging using inputs. (See your drive’s User’s Manual for designation of “jog” inputs).

BLu, STAC6
An “X” in Parameter #1 is used to designate an input on the IN/OUT 1 connector of BLuAC or STAC drives, or the main driver board connector (DB-25) of BLuDC drives.

Command Details:

Structure	WI(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

BLu, STAC6

Parameter #1	Optional “X”, Input number, Input condition NOTE: “X” designates that the input in this command resides on the IN/OUT1 connector of BLuAC and STAC6 drives or the main driver board connector (DB-25) of BLuDC drives.
- units	optional “X”, integer, letter
- range	IN/OUT 1 or main driver board (DB-25) - integer: X0 - X7, X8, X9, X: (X0 designates the encoder index channel if present) - letter: L = low, H = high, F = falling edge, R = rising edge IN/OUT 2 or top board (screw terminal) - integer: 1 - 8 - letter: L = low, H = high, F = falling edge, R = rising edge

SV, ST-Q/Si

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 - 8, 9, :, ; (0 designates the encoder index channel if present) letter: L = low, H = high, F = falling edge, R = rising edge

ST-S, STM

Parameter #1	Input number, Input condition
- units	integer, letter
- range	integer: 0 = Encoder Index (if present), 1 = STEP, 2 = DIR, 3 = EN, 4 = AIN letter: L = low, H = high, F = falling edge, R = rising edge

Examples:

Command	Drive sends	Notes
WI3R	-	Wait for input 3 to go high (rising edge) before proceeding to the next command in the queue

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, 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 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.*

NOTE: 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.

WM - Wait on Move

Compatibility: Q drives only
Affects: Queue execution
See also: MT

When in multi-tasking is turned on (see MT command) this command will block execution of subsequent commands until the previously initiated move is complete. This can be any type of move such as “Feeds”, “Jogging” or the “Hand Wheel” (encoder following).

Command Details:

Structure	WM
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Examples:

Command	Drive sends	Notes
WM	-	Causes queue execution to wait until any move in progress is complete

WP - Wait Position

Compatibility: Q drives only
Affects: Multi-velocity, or complex, move profiles
See Also: FC, DC, VC, MT commands

When multi-tasking is turned on (“MT1”), this command is used in conjunction with the DC command to block program execution until a specific position(s) is reached during a move. When the position(s) specified by the DC command is reached program execution continues.

Common example:
This command is used as a “separator” in changing the motor speed of multi-velocity move profiles created using the FC command. The normal FC command provides for one speed change using values determined by DC and VC commands executed prior to the FC command. Additional speed changes can be added after an FC command is initiated by using the WP command to separate additional DC and VC commands. See the example below.

NOTE: This command, along with the ability to create multi-velocity move profiles with the FC command, is only available in BLu servo drive firmware revisions 1.53C or later. This command is available in all firmware revisions of STAC stepper drives.

Command Details:

Structure	WP
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	No
Register Access	None

Examples:

Command	Drive sends	Notes
DI100000	-	Overall move distance set to 100,000 counts
VE10	-	Initial move speed set to 10 rps
DC80000	-	1st change distance set to 80,000 counts
VC15	-	1st change speed set to 15 rps
FC	-	Initiate FC command (complex move)
WP	-	
DC60000	-	2nd change distance set to 60,000 counts
VC9	-	2nd change speed set to 9 rps
WP	-	
DC40000	-	3rd change distance set to 40,000 counts
VC1	-	3rd change speed set to 1 rps
WP	-	
DC20000	-	4th change distance set to 20,000 counts
VC19	-	4th change speed set to 19 rps

WT - Wait Time

Compatibility: All drives

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

Command Details:

Structure	WT(Parameter #1)
Type	BUFFERED
Usage	WRITE ONLY
Non-Volatile	NO
Register Access	None

Parameter Details:

Parameter #1	Time
- units	seconds
- range	0.00 - 320.00 (resolution is 0.01 seconds)

Examples:

Command	Drive sends	Notes
WT2.25	-	Causes time delay of 2.25 seconds

ZC - Regen Resistor Continuous Wattage

Compatibility: BLuAC5 and STAC6 drives only

Sets or requests the regeneration resistor wattage value. BLuAC and STAC drives dynamically calculate the continuous wattage induced into an external regeneration resistor and must know the continuous wattage rating of the regen resistor to do this effectively.

Command Details:

Structure	ZC{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Continuous wattage value of regen resistor
- units	Watts
- range	1 - 1000

Examples:

Command	Drive sends	Notes
ZC250	-	External regen resistor with value of 250 continuous watts is connected to the drive

ZR - Regen Resistor Value

Compatibility: BLuAC5 and STAC6 drives only

Sets or requests the regeneration resistor value. BLuAC and STAC drives dynamically calculate the continuous wattage induced into an external regeneration resistor and must know the value of the regen resistor to do this effectively.

Command Details:

Structure	ZR{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Value of regen resistor
- units	Ohms
- range	25 - 100

Examples:

Command	Drive sends	Notes
ZR50	-	50 ohm external regen resistor connected to drive

ZT - Regen Resistor Peak Time

Compatibility: BLuAC5 and STAC6 drives only

Sets or requests the regeneration resistor time constant. Decides the peak time that the resistor can tolerate full regeneration voltage. When regeneration occurs the full regeneration voltage of 400 volts is applied across the resistor. The peak wattage is typically very high, for example with the built-in 40 ohm resistor the peak wattage is 4000 Watts. Power resistors will tolerate this for only a brief period of time. In the case of the built-in 40 ohm/ 50 Watt regen resistor it is only 0.3125 seconds. The ZT value provides the resistor time constant used to create the “filter” for calculating average wattage in the regen resistor.

Command Details:

Structure	ZT{Parameter #1}
Type	BUFFERED
Usage	READ/WRITE
Non-Volatile	Yes
Register Access	None

Parameter Details:

Parameter #1	Maximum time for peak regen
- units	0.25 milliseconds
- range	1 - 32000

Examples:

Command	Drive sends	Notes
ZT1250	-	Regen resistor peak time set to 0.3125 seconds
ZT	ZT=1250	

Data Registers

Many of the commands listed in this reference function by transferring data to a drive for later use. These data values are stored in data registers within the drive and remain there until new commands change the values or power is removed from the drive. For example, if you send the Velocity command “VE10”, a maximum move speed of 10 rev/sec is placed in the data register for velocity. You can then execute as many FL (Feed to Length), FP (Feed to Position) or FS (Feed to Sensor) move commands as you’d like without sending another VE command: the move speed of 10 rev/sec will remain in the velocity data register until you change it.

In addition to the data register for velocity, there are registers for move acceleration (AC command, “A” register), deceleration (DE command, “B” register) and move distance (DI command, “D” register). There are also registers for limit sensors (DL command), motor current (CC command), encoder resolution (ER command), motor position (SP command) and encoder position (EP command). There are 75 data registers in all. See the following Data Register Assignments section for a complete listing of data registers available in your drive.

Not all commands function by transferring a data value into a register. Conversely, not all data registers are associated with a command. To access data registers that are not associated with a command, you can use a register’s unique character assignment. See the Data Register Assignments on the following pages for a listing of data registers and their character assignments. When accessing a data register using its character assignment you use the RL (Register Load Immediate) or RX (Register Load Buffered) commands. These commands allow you to load data values into a register as well as read back the contents of a data register. For example, we set the move speed to 10 rev/sec in the first paragraph of this page by using the velocity command “VE10”. You can accomplish the same thing by using the RL command and the character assignment for the velocity data register, “V”. By sending “RLV2400” to the drive (see units of “V” register in Data Register Assignments section) you set the move speed to 10 rev/sec.

There are four categories of data registers available with your drive: Read-Only, Read/Write, User-Defined, and Storage. The last two categories, User-Defined and Storage, are only for use with Q drives.

Read-Only data registers

Read-Only data registers are predefined registers that contain information about drive parameters, settings, and states. These include registers for commanded current, encoder position, analog input levels, drive temperature, internal bus voltage, and more. You cannot transfer data values to a Read-Only data register; you can only read the contents of them (see RL and RX commands). Read-Only registers are assigned to lower-case letters.

Read/Write data registers

Read/Write data registers are predefined registers that contain drive and move parameters that can be set by the user. These parameters include acceleration rate, velocity, move distance, continuous current setting, peak current setting, and more. Many of the Read/Write registers are associated with a particular command, so you can read their contents or load data into them with RL, RX, or that parameter’s particular command. Read/Write registers are assigned to upper-case letters.

User-Defined data registers

User-Defined data registers are read/write registers that are not predefined. These registers are only used with Q drives. They allow you to create more flexible and powerful Q programs through math functions, incrementing and decrementing, conditional processing, and more. These registers are assigned to single-digit numbers and other ASCII characters.

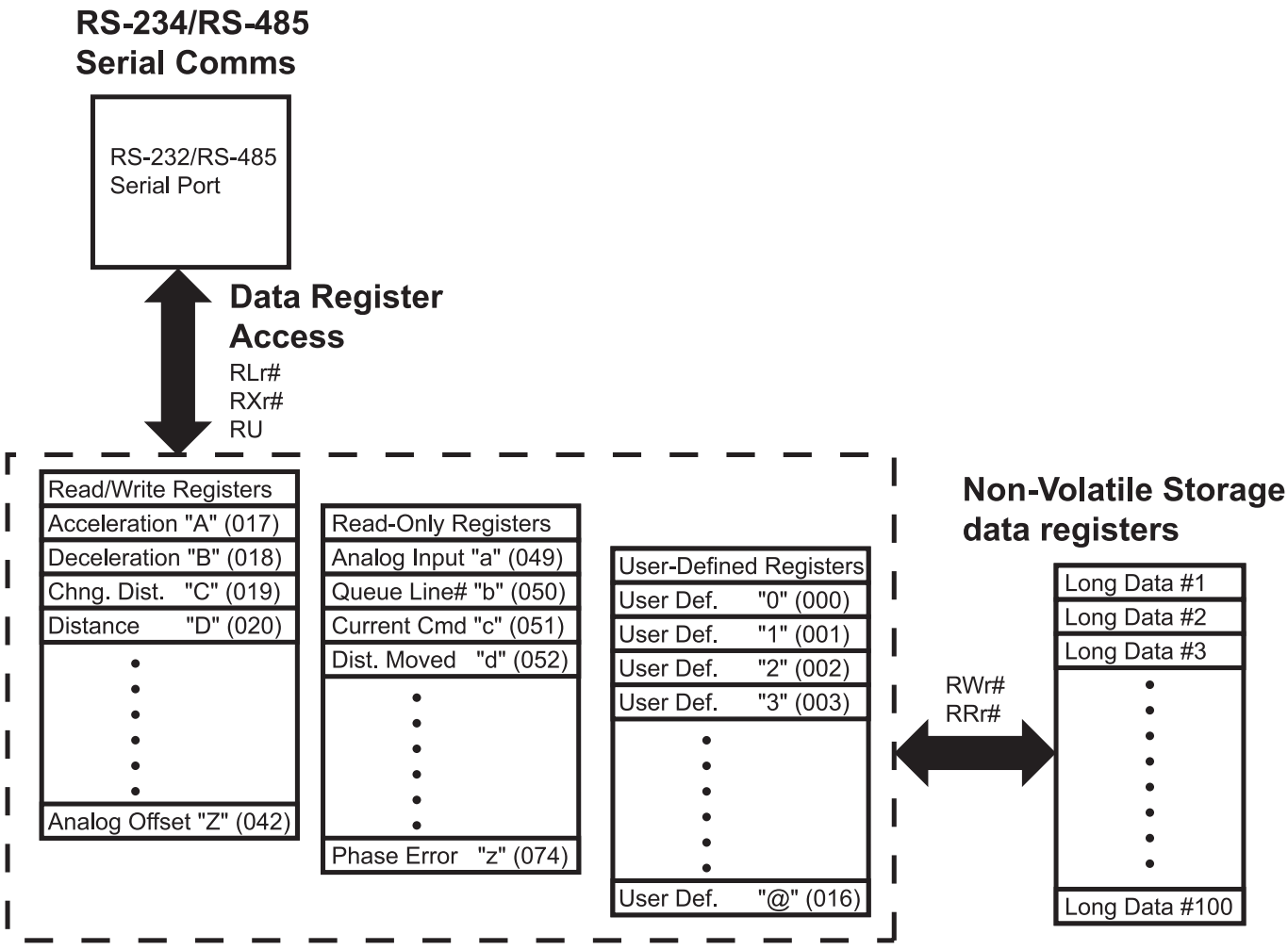
Storage data registers

Each Q drive comes with 100 non-volatile Storage data registers, which can be used to save the contents of other data registers to non-volatile memory. For example, since none of the User-Defined data registers are non-volatile, a user may want to save the values of some of these registers to memory. This can be done by transferring their values to Storage registers (called Writing) before power down of the drive. Then at the next power up, these values can be loaded back into the User-Defined registers from the Storage registers (called Reading).

Each Storage register can save one data register value, and the Storage registers are numbered 1 to 100. See the RR, RW, and SA commands as well as the Appendix for more information on accessing this section of memory.

Using Data Registers

The diagram below shows how a drive's serial port accesses the different volatile (Read-Only, Read/Write, User-Defined) and non-volatile (Storage) data registers within a drive. The user can Load and Upload data register values using the RL, RX, and RU commands via the drive's serial port(s). Read-Only data registers can be uploaded but not loaded. For Q drives only, non-volatile memory is available for data registers in the form of Storage registers. Moving the contents of the volatile data registers back and forth between the non-volatile Storage registers is done with the RW and RR commands. See below for more details.



Loading (RL, RX)

Accessing data registers is done by Loading data into a register, and Uploading data from a register. Loading a data register can be done from a host command line or from a line in a program. To load a register from a host command line use the RL (Register Load) command. This command can be executed at any time, even while a drive is running a program. The RL command is an immediate command. To load a register within a Q program use the RX command, which is a buffered version of Register Load.

Uploading (RL, RU)

Uploading data registers can only be done from a host command line, not within a program. There are two commands available for uploading register values. RL is used to upload one register value at a time, while RU can be used to upload a single register value or an array of register values. Both RL and RU are immediate commands, and therefore can be executed while a program is running. The RU command can request up to 10 data register values in sequence back from the drive. This is great when an array of information is required at one time.

Writing Storage registers (RW) *(Q drives only)*

Writing a data register allows the user to store data register values in non-volatile memory. To write a data register we use the RW (Register Write) command. There are 100 storage locations for data registers in NV memory. Note that the user must keep track of where data registers are stored because the NV memory locations are not associated with any specific data register.

Reading Storage registers (RR) *(Q drives only)*

Reading a data register allows the user to move data previously saved in NV memory into a data register. To read a data register we use the RR (Register Read) command. Reading is typically done in the midst of a Q program.

The following sub-sections describe additional usage of data registers within Q drives only.

Moving data registers (RM) *(Q drives only)*

Data register values can be moved from one register to another. This is done with the RM (Register Move) command. When executing an RM command, the contents of the originating data register are retained. Contents of read-only registers can be moved into read/write registers and user-defined registers. However, as implied by its label, no register values can be moved into read-only registers. Attempting to do so will have no effect and no error code is generated.

Incrementing/Decrementing (RI, RD) *(Q drives only)*

Read/write and user-defined registers can be incremented and decrmented by “1”. Two commands are used for these functions: the RI (Register Increment) and RD (Register Decrement) command. NOTE: Incrementing past the range of a data register will cause the value to wrap around.

Counting (RC, “I” register) *(Q drives only)*

A special data register, the “I” register (Input Counter), is designated for counting input transitions and input state times of a selected digital input. The “I” register is a read/write register that can be used with all other register functions including math and conditional testing.

The RC (Register Counter) command is used to assign digital inputs to register counting. There are four different input states that can be chosen and that have different effects on input counting. When using the “high” or “low” level states the counter acts as a “timer” with a resolution of 100 microseconds (SV servo drives and all stepper drives) or 125 microseconds (BLu servo drives). Edge type states like “falling” or “’rising” are used for input counting. (See details of the RC command in the Q Command Reference).

Math & Logic (R+, R-, R*, R/, R&, R|) *(Q drives only)*

Math and logic functions can be performed on data registers. Math is limited to integer values. Some of the math functions are also limited to 16-bit values. When doing math only one operation can be done per instruction. Math and logic results are stored in the Accumulator register, “0”. This register is part of the user-defined register set. Math functions include Add, Subtract, Multiply and Divide. Logic functions include Logical AND and Logical OR.

Conditional Testing (CR, TR) *(Q drives only)*

When constructing complex programs it is usually necessary to do some conditional processing to affect program flow. Two commands are available for evaluating a data register for conditional processing, the TR (Test Register) and CR (Compare Register) commands. The TR command will compare the “First” value of a given

data register against a “Second” immediate value. The CR command compares the “First” value of a given data register against the “Second” value of another data register. When using the TR and CR commands an internal “Condition” register is set with the result. The result can be:

“True”	the “First” value is either positive or negative
“False”	the “First” value is not a value (it’s zero)
“Zero”	the “First” value equals “0”
“Positive”	the “First” value is “positive”
“Negative”	the “First” value is “negative”
“Greater Than”	the “First” value is more positive than the “Second” value
“Less Than”	the “First” value is more negative than the “Second” value
“Equal to”	the “First” and “Second” values are equal
“Unequal to”	the “First” and “Second’ values are not equal

NOTE: The QJ (Queue Jump) command is designed to use the “Condition Codes” above for jumping. The Condition Code can also be accessed via the “h” register.

Data Register Assignments

What follows is a listing of all the data registers available with Applied Motion drives. In the tables below, “Ch.” denotes the data register’s character assignment, and “Description” gives the name of the data register. The column “3-digit” denotes the register’s 3-digit equivalent numerical assignment (see PR command, bit 5); “Data Type” designates whether the data register is a 16-bit word (Short) or a 32-bit word (Long); “Units” shows how a data register’s contents are used by the drive; and, “Compatibility” shows which drives can make use of the given register.

NOTE: When programming a Q drive with the Q Programmer software only the character assignment of the register can be used. When communicating to a Q drive via one of its serial ports, either the character assignment or the 3-digit numerical assignment can be used.

Read-Only data registers: a - z

Many of the Read-Only data registers can be read with a specific command. In the tables below, associated commands are shown in parentheses in the “Description” column.

Ch.	Description	3-digit	Data Type	Units	Compatibility
a	Analog Command value (IA)	049	Short	BLu, SV, STAC6, ST-Q/Si: 32760 = +10V; -32760 = -10V ST-S, STM: 16383 = +5V; 0 = 0V*	All drives

*Note that the “a” register is affected by the AV (Analog Offset) command, so the range may vary beyond 0 to 16383.

b	Queue Line Number	050	Short	Line # 1 - 62	Q drives only
c	Current Command (IC)	051	Short	Servo: 0.01 amps RMS Stepper: 0.01 amps, peak-of-sine	All drives
d	Relative Distance (ID)	052	Long	Servo: encoder counts Stepper: steps	All drives

BLu, STAC6

The “d” register (as well as the ID command) contains the relative move distance used in the last move. This means that the “d” register is only updated at the end of every relative move.

SV, ST-Q/Si, ST-S, STM

The “d” register (as well as the ID command) contains the immediate relative distance moved since the start of the last or current relative move. This means the “d” register is updated during relative moves and can therefore be polled during a move to see where the motor is with respect to the overall relative move distance.

e	Encoder Position (IE, EP)	053	Long	encoder counts	Servo drives and stepper drives with encoders

The “e” register can be zeroed by sending the command EP0.

f	Alarm Code (AL)	054	Long	hexadecimal equivalent of binary Alarm Code word (See AL command for details)	All drives
g	Sensor Position	055	Short	Servo: encoder counts Stepper: steps	All drives

The “g” register contains the absolute position of the point at which the input condition is met during moves like FS, FE, SH, and other “sensor-type” moves. It is common practice to use the EP and SP commands to establish known absolute positions within an application or program, which will make the value of the “g” register most meaningful. Otherwise, the absolute position of the motor is zeroed at every power-up of the drive.

h	Condition Code	056	Short	decimal equivalent of binary word (see below)	Q drives only

The response to the “RLh” command will be the decimal equivalent of the condition code’s binary word. Bit assignments and examples are shown below.

Description	Bit #	Decimal Value
TRUE	0	1
ZERO	1	2
POSITIVE	2	4
NEGATIVE	3	8
GREATER THAN	4	16
LESS THAN	5	32
EQUAL TO	6	64
UNEQUAL TO	7	128

Example:

Command	Drive Sends	Notes
RLh	RLh=149	Bits 7 (UNEQUAL TO), 4 (GREATER THAN), 2 (POSITIVE) and 0 (TRUE) are set. Within a Q program the programmer will often have more than one condition to choose from when using the QJ command. The condition FALSE in Q Programmer is represented by bit 0 = 0 (opposite of TRUE).

i	Driver Board Inputs (IS)	057	Short	decimal equivalent of binary bit pattern (see below)	All drives

Details when executing the “RLi” command:

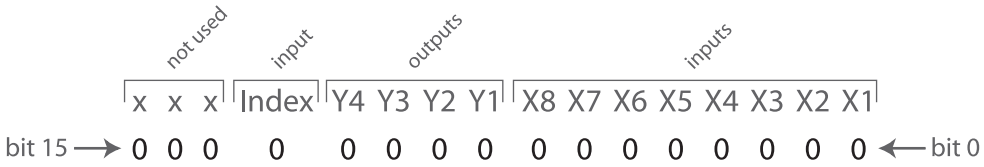
BLu, STAC6

The bit pattern of the “i” register breaks down as follows: bit 0 is the state of the encoder’s index (Z) channel, also known as input X0; bits 1 - 7 represent the states of driver board inputs X1 - X7, respectively; bits 8 - 10 represent the states of driver board outputs Y1 - Y3, respectively; and, bits 11 - 15 are not used. For bits 0 - 7 (inputs X0 - X7), a state of “1” means the optically isolated input is open, and a state of “0” means the input is closed. It is the exact opposite for bits 8 - 10 (outputs Y1 - Y3), for which a state of “1” means the optically isolated output is closed, and a state of “0” means the output is open.

For example, if driver board inputs X1 and X4 and driver board output Y2 were closed, the response of the drive to the command “RLi” would be “RLi=749” (0000 0010 1110 1101).

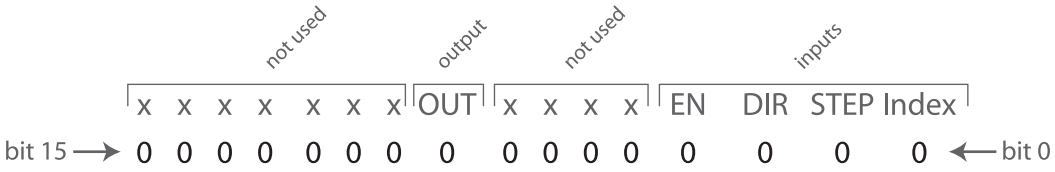
SV, ST-Q/Si

The bit pattern of the “i” register breaks down as follows: bits 0 - 7 represent inputs X1 - X8, respectively; bits 8 - 11 represent outputs Y1 - Y4, respectively; and, bit 12 is the encoder index channel (if present). For bits 0 - 7 and 12 (inputs X1 - X8 and the Index), a state of “1” means the optically isolated input is open, and a state of “0” means the input is closed. It is the exact opposite for bits 8 - 11 (outputs Y1 - Y4), for which a state of “1” means the optically isolated output is closed, and a state of “0” means the output is open.



ST-S, STM

The bit pattern of the “i” register breaks down as follows: bit 0 represents the encoder index channel (if present), bit 1 represents the STEP input, bit 2 the DIR input, and bit 3 the EN input. Bit 8 represents the drive’s single output, OUT. For bits 0 - 3 (Index, STEP, DIR, and EN inputs), a state of “1” means the optically isolated input is open, and a state of “0” means the input is closed.



j	Analog Input 1 (IA1)	058	Short	raw ADC counts, 0 - 32760 16383 = 0 volts for BLu, SV, STAC6, ST-Q/Si drives	All drives
k	Analog Input 2 (IA2)	059	Short	raw ADC counts, 0 - 32760 16383 = 0 volts	BLu, SV, STAC6, ST-Q/Si only
l	Absolute Position	060	Short	bit pattern	All drives
m	Command Mode (CM)	061	Short	Mode #	All drives

n	Velocity Move State	062	Short	State # (see below)	All drives

Response details to the “RLn” command:

Description	Decimal Value	Comment
WAITING	0	In velocity mode waiting for a command
RUNNING	1	Doing a velocity move (jogging)
FAST STOPPING	2	Stopping a velocity move (ST or SK with no parameter)
STOPPING	3	Stopping a velocity move (SJ, STD, or SKD)
ENDING	4	Clean up at end of move (1 PWM cycle, 62 usec)

o	Point-to-Point Move State	063	Short	State # (see below)	All drives

Details when using “RLo” command:

Description	Decimal Value	Comment
WAITING	0	In position mode waiting for command
WAITING ON BRAKE	1	Waiting for brake to release
CALCULATING	2	Doing the calculations for the move
ACCELERATION	3	Accelerating up to speed
CHANGE VELOCITY	4	Changing the speed (accel or decel)
AT_VELOCITY	5	At the desired speed
DECELERATION	6	Decelerating to a stop
FAST DECELERATION	7	Doing a fast deceleration (ST or SK)
POSITIONING	8	Clean up at end of move (1 PWM cycle, 62 usec)

p	Segment Number	064	Short	Segment # 1 - 12	Q drives only
q	Actual Motor Current (IQ)	065	Short	0.01 Amps	Servo drives only
r	Average Clamp Power	066	Short	Watts	BLuAC5, STAC6
s	Status Code (SC)	067	Short	hexadecimal equivalent of binary Status Code word (See SC command for details)	All drives
t	Drive Temperature (IT)	068	Short	0.1 °C	All drives
u	Bus Voltage (IU)	069	Short	0.1 Volts	All drives
v	Actual Velocity (IV0)	070	Short	0.25 rpm	Servo drives and stepper drives with encoder
w	Target Velocity (IV1)	071	Short	0.25 rpm	All drives*

*For stepper drives, the “w” register is only updated when Stall Detection or Stall Prevention is turned on.

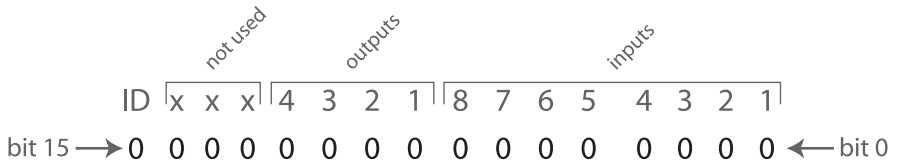
x	Position Error (IX)	072	Long	encoder counts	Servo drives and stepper drives with encoder
y	Expanded Inputs (IS)	073	Short	bit pattern	BLu, STAC6

Details when executing the “RLy” command:

BLu, STAC6 drives only

The bit pattern of the “y” register breaks down as follows: bits 0 - 7 represent the states of top board inputs 1

- 8, respectively; bits 8 - 11 represent the states of driver board outputs 1 - 4, respectively; and, bits 12 - 15 are not used. For all I/O bits 0 - 11 (inputs 1 - 8 and outputs 1 - 4), a state of “1” means the optically isolated input or output is open, and a state of “0” means the input or output is closed. Bit 15 represents the ID bit, which simply holds a 1 if the IN/OUT2 or screw terminal I/O board is present and a 0 of it’s not. In other words, for SE, QE and Si drives the ID bit will equal 1. For S and Q drives the ID bit will equal 0.



For example, if top board inputs 3 and 5 and top board outputs 1 and 2 were all closed, the response of the drive to the command “RLi” would be “RLi=-29461” (1000 1100 1110 1011). For a more efficient use of the “y” register it is recommended to mask off the ID bit and the other three not used bits. This can be done by using the R& (Register AND) command with the “y” register and a User Defined register set with the value 4095. For example, the command sequence would look like this.

RL14095	Load User Defined register “1” with the value 4095
R&y1	Register AND the “y” and “1” registers
RL0	Request the value stored in the Accumulator register “0” to which the drive’s response would be RL0=3307.

z	Phase Error	074	Short	encoder counts	Servo drives only

Read/Write data registers: A - Z

Many of the Read/Write data registers are associated with a specific command. In the tables below, associated commands are shown in parentheses in the “Description” column.

NOTE: When using registers pay attention to units. In the case of some Read/Write registers, the units of the register when using the RL and RX command are different than when using the same register’s associated command. For example, the “V” register uses units of 0.25 rpm, but its associated command, VE, uses revs/sec (rps). The reason for this difference is that all registers operate with integer math. On the other hand, when using commands it is often possible to include decimal places which allow for more user-friendly units.

Ch.	Description	3-digit	Data Type	Units	Compatibility
A	Acceleration (AC)	017	Short	10 rpm/sec	All drives

The “A” register units are 10 rpm/sec, which means that the value of the “A” register is equal to 6 times the AC command value. In other words, to achieve an acceleration value of 100 rev/sec/sec send the command RLA600.

B	Deceleration (DE)	018	Short	10 rpm/sec	All drives

The “B” register units are 10 rpm/sec, which means that the value of the “B” register is equal to 6 times the DE command value. In other words, to achieve a deceleration value of 100 rev/sec/sec send the command RLB600.

C	Change Distance (DC)	019	Long	counts	All drives
D	Distance (DI)	020	Long	counts	All drives

E	Position Offset	021	Long	counts	All drives

The “E” register contains the difference between the encoder count and the motor position. This value is most useful with BLu servo drives where the resolution of the motor and encoder are the same, and this offset can be useful when working with absolute positions. The register contains the difference in counts between the “e” register and the value set by the “SP” command.

F	Other Flags	022	Long	bit pattern (see below)	All drives

BLu

The value of the “F” register is a hexadecimal sum of various drive states, as shown below.

Description	Hex Value	Decimal Value
DISTANCE LIMIT FLAG	0x0001	1
SENSOR FOUND FLAG	0x0002	2
LOWSIDE OVERCURRENT	0x0004	4
HIGHSIDE OVERCURRENT	0x0008	8

Clear flags by sending “RLF0” to the drive.

SV

The value of the “F” register is a hexadecimal sum of various drive states, as shown below.

Description	Hex Value	Decimal Value
DISTANCE LIMIT FLAG	0x0001	1
SENSOR FOUND FLAG	0x0002	2
LOWSIDE OVERCURRENT	0x0004	4
HIGHSIDE OVERCURRENT	0x0008	8
OVER CURRENT READING	0x0010	16
BAD CURRENT OFFSET - Phase A	0x0020	32
BAD CURRENT OFFSET - Phase B	0x0040	64
BAD FLASH ERASE	0x4000	16384
BAD FLASH SAVE	0x8000	32768

Clear flags by sending “RLF0” to the drive.

STAC6

The value of the “F” register is a hexadecimal sum of various drive states, as shown below.

Description	Hex Value	Decimal Value
DISTANCE LIMIT FLAG	0x0001	1
SENSOR FOUND FLAG	0x0002	2
HARDWARE OVERCURRENT	0x0004	4
SOFTWARE OVERCURRENT	0x0008	8
BAD CURRENT OFFSET - Phase A	0x0010	16
BAD CURRENT OFFSET - Phase B	0x0020	32
OPEN WINDING - Phase A	0x0040	64
OPEN WINDING - Phase B	0x0080	128

Clear flags by sending “RLF0” to the drive.

ST-Q/Si, ST-S, STM

The value of the “F” register is a hexadecimal sum of various drive states, as shown below.

Description	Hex Value	Decimal Value
DISTANCE LIMIT FLAG	0x0001	1
SENSOR FOUND FLAG	0x0002	2
LOWSIDE OVERCURRENT	0x0004	4
HIGHSIDE OVERCURRENT	0x0008	8
OVER CURRENT READING	0x0010	16
BAD CURRENT OFFSET - Phase A	0x0020	32

BAD CURRENT OFFSET - Phase B	0x0040	64
OPEN WINDING - Phase A	0x0080	128
OPEN WINDING - Phase B	0x0100	256
LOGIC SUPPLY	0x0200	512
GATE SUPPLY	0x0400	1024
BAD FLASH ERASE	0x4000	16384
BAD FLASH SAVE	0x8000	32768

Clear flags by sending “RLF0” to the drive.

G	Current Command (GC)	023	Short	0.01 Amps	Servo drives only
H	Analog Velocity Gain	024	Short	+/- 32767 ADC counts	BLu servo drives only

The “H” register in BLu servo drives is similar to the AG command in all other drives. The “H” register is used to set the motor speed at a given DC voltage in analog velocity mode. It is recommended to make this setting in *Quick Tuner*, where it is labeled Speed in rev/sec at xx Volts, under the Velocity > Analog Operating Mode.

I	Input Counter	025	Long	counts per edge	Q drives only
J	Jog Velocity (JS)	026	Short	0.25 rpm	All drives

The “J” register units are 0.25 rpm, which means that the value of the “J” register is equal to 240 times the JS command value. In other words, to achieve a jog speed value of 7 rev/sec send the command RLJ1680.

K	Jog Acceleration (JA)	027	Short	10 rpm/sec	All drives

The “K” register units are 10 rpm/sec, which means that the value of the “K” register is equal to 6 times the JA command value. In other words, to achieve a jog acceleration value of 100 rev/sec/sec send the command RLK600.

L	Jog Deceleration (JL)	028	Short	10 rpm/sec	All drives

The “L” register units are 10 rpm/sec, which means that the value of the “L” register is equal to 6 times the JL command value. In other words, to achieve a jog deceleration value of 100 rev/sec/sec send the command RLL600.

M	Max Velocity (VM)	029	Short	0.25 rpm	Servo drives only
N	Continuous Current (CC, servo) Running Current (CC, stepper)	030	Short	Servo: 0.01 amps RMS Stepper: 0.01 amps, peak-of-sine	All drives
O	Peak Current (CP, servo) Idle Current (CI, stepper)	031	Short	Servo: 0.01 Amps RMS Stepper: 0.01 amps, peak-of-sine	All drives
P	Absolute Position Command	032	Long	counts	All drives
Q	RESERVED	033	-	-	-
R	Steps per Rev*	034	Short	counts	All drives

* Note: R = EG for servo drives. R = EG/2 for stepper drives.

S	Pulse Counter	035	Long	counts	All drives

The “S” register counts pulses coming into the STEP/X1 and DIR/X2 inputs of the drive. This is particularly useful when in Command Mode 7 (see CM command) or executing an FE (Follow Encoder) command. To zero the “S” register send the command RLS0.

T	Total Count	036	Long	(see below)	Q drives only

The “T” register is automatically saved at power down and restored at power up.

U	Change Velocity (VC)	037	Short	0.25 rpm	All drives

The “U” register units are 0.25 rpm, which means that the value of the “U” register is equal to 240 times the VC command value. In other words, to achieve a change velocity value of 7 rev/sec send the command RLU1680.

V	Velocity (VE)	038	Short	0.25 rpm	All drives

The “V” register units are 0.25 rpm, which means that the value of the “V” register is equal to 240 times the VE command value. In other words, to achieve a velocity value of 7 rev/sec send the command RLV1680.

W	Time Stamp	039	Short	0.001 sec	Q drives only
X	Analog Position Gain (AP)	040	Short	Servo: ADC counts/encoder count Stepper: ADC counts/step	All drives
Y	Analog Threshold (AT)	041	Short	raw ADC counts	All drives
Z	Analog Offset (AV)	042	Short	raw ADC counts	All drives

User-Defined data registers: 0 - 9, other characters

Ch.	Description	3-digit	Data Type	Units	Compatibility
0	Accumulator	000	Long	integer	Q drives only

The Accumulator register “0” is, aside from being a User-defined data register, the register in which the result of every register math function is placed. For example, if the drive executes the register addition command “R+D1” the result of this operation (i.e. the sum of the values in data registers “D” and “1”) will be placed in the Accumulator “0” register.

1	User-defined	001	Long	integer	Q drives only
2	User-defined	002	Long	integer	Q drives only
3	User-defined	003	Long	integer	Q drives only
4	User-defined	004	Long	integer	Q drives only

5	User-defined	005	Long	integer	Q drives only
6	User-defined	006	Long	integer	Q drives only
7	User-defined	007	Long	integer	Q drives only
8	User-defined	008	Long	integer	Q drives only
9	User-defined	009	Long	integer	Q drives only
:	User-defined	010	Long	integer	Q drives only
;	User-defined	011	Long	integer	Q drives only
<	User-defined	012	Long	integer	Q drives only
=	User-defined	013	Long	integer	Q drives only
>	User-defined	014	Long	integer	Q drives only
?	User-defined	015	Long	integer	Q drives only
@	User-defined	016	Long	integer	Q drives only
[	User-defined	043	Long	integer	Q drives only
\	User-defined	044	Long	integer	Q drives only
]	User-defined	045	Long	integer	Q drives only
^	User-defined	046	Long	integer	Q drives only
_	User-defined	047	Long	integer	Q drives only
‘	User-defined	048	Long	integer	Q drives only

Appendices

The following appendices detail various special topics in working with Applied Motion motor drives.

Appendix A: Non-Volatile Memory in Q drives

Appendix B: Host Serial Communications

Appendix C: Host Serial Connections

Appendix D: The PR Command

Appendix E: Alarm and Status Codes

Appendix F: Working with Inputs and Outputs

Appendix A: Non-Volatile Memory in Q drives

The non-volatile memory in Q drives is partitioned into 16 sections. The partitions are dedicated to various elements of a Q drive's data, and are designated as follows:

Partition 1..... Q Program Segment 1
Partition 2..... Segment 2
Partition 3..... Segment 3
Partition 4..... Segment 4
Partition 5..... Segment 5
Partition 6..... Segment 6
Partition 7..... Segment 7
Partition 8..... Segment 8
Partition 9..... Segment 9
Partition 10..... Segment 10
Partition 11..... Segment 11
Partition 12..... Segment 12
Partition 13..... Drive Parameters
Partition 14..... Alarm History
Partition 15..... NV Data Register Storage Locations 1-100
Partition 16..... RESERVED

The separation of these partitions is important in understanding how the drive writes to non-volatile memory. For example, each time the SA command is executed by the drive, all of the Drive Parameters are re-written to non-volatile memory partition 13. Similarly, each time an RW command is executed by the drive, all of the one hundred NV Data Register Storage Locations are re-written in partition 15, even if only one of the locations is being updated with a new data register value.

The significance of these operations becomes clear when we consider that the physical non-volatile memory of the Q drive is limited to approximately 10,000 write cycles. This means that after writing to any one of the 16 partitions 10,000 times, the integrity of the data stored in that memory partition cannot be insured.

For this reason, it is not recommended to use the RW or SA commands in stored Q programs. For example, it might be tempting for a user to include an RW command or two in a stored program in such a manner that allows for various data register values to be written to non-volatile memory on a regular basis. The temptation of this is that there won't be a need to reload register values manually in the case of a power down/up cycle: the register values can simply be loaded back into the program (using RR commands) from non-volatile memory. This is to be avoided, though, because using the RW command (or SA command) in this manner could result in the early failure of the non-volatile memory of the drive. The intended use of the RW command therefore is to be used in the early stages of an application, during startup and programming, to set up a series of non-volatile register locations that can be read into a stored program using the RR command.

The partitions designated for Q Program Segment storage are typically not going to be re-written in a manner similar to the RW and SA commands, as they are only accessed during program/segment downloads during startup and programming of an application.

Appendix B: Host Serial Communications

When a drive is operating in “host mode”, it means that a host device sends commands to the drive (or drives) over a serial connection (or network) and the drive executes the incoming commands. Here are some examples of typical host devices:

- A Windows-based PC running Applied Motion 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 Applied Motion Products motor drive in host mode.

- General structure of host serial communications.
- Hardware – wiring and connecting a host device to the serial ports of an Applied Motion drive. (Covered in detail in Appendix D).
- COM Port Settings – UART settings and Bit Rate (Baud) settings.
- Communications Protocol
- Communication Details
- Communication Errors

General structure of host serial communications

Applied Motion’s host serial communications are based on the common ASCII character set transmitted using standard UARTs over an RS-232 or RS-485 hardware interface.

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 and RS-485 hardware connections are commonly used with UARTs and also provide the easiest and most common form of connectivity.

Hardware

Details on drive terminals and connectors for wiring each of the available hardware configurations are shown in Appendix D. Below is an overview of the three available configurations.

RS-232: This is the easiest method for drive serial communications. Using an Applied Motion supplied adapter/programming cable (one supplied with each Applied Motion drive) a single drive can be connected directly to any PC with a standard 9-pin RS-232 serial port. Here are some RS-232 highlights:

- Easiest to use
- Configuration of choice for using Applied Motion software applications such as Q Programmer, Quick Tuner and STAC6 Configurator
- Short Cable Lengths
- Serial cable provided with each Applied Motion drive
- Susceptible to EMI

RS-422 (4-wire RS-485): RS-422 was originally designed for high reliability communications in point-to-point configurations. It 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). Here are some RS-422 highlights:

- Relatively easy to use
- NOT supported by Applied Motion software applications such as Quick Tuner or STAC6 Configurator. (Q Programmer does support RS-422 in a limited fashion).

- Permits longer cable Lengths
- May require special adaptor
- Immune 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:

- More difficult to use
- NOT supported by Applied Motion software applications such as Quick Tuner or Configurator. (Q Programmer does support RS-485 in a limited fashion).
- Permits longest cable lengths: up to 1000 feet at low baud rates
- May require special adaptor
- Fewest wires, smaller cables
- Immune 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 AMP drives default to 9600 baud from the factory. In most cases this speed is adequate for setup, configuring, 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" below). 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. The 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 (if necessary) by sending a response packet back to the host.

Command Transmission (host to drive): The transmission of 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 time-out feature is to allow the drive to purge its buffers automatically when a bad transmission occurs.

NOTE: This time-out feature limits the usage of host devices such as the Windows application HyperTerminal. We recommend using Applied Motion's SCL Setup Utility instead. This utility sends out an entire command packet with the minimum delay between characters, and includes the packet's terminating character (carriage return).

Command packets are terminated by a Carriage Return (ASCII 13).

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

(see below).

Response packets are terminated by a Carriage Return (ASCII 13).

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 understand) 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: the three states are 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. Applied Motion drives de-assert their transmitters very quickly. Typically it is done within 100 microseconds (.0001 second) after the end of a packet transmission. However it is possible that the host device won't be this fast, and so the TD command allows users to set the time delay that an Applied Motion 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. This can vary depending on the settings of the PR command. See Appendix E for more on the PR command. All packets are terminated by a Carriage Return (ASCII 13).

Drive Startup: At power-up, all Applied Motion 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 Applied Motion software applications such as Quick Tuner and STAC6 Configurator. 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, a BLuAC5-Si with f/w version 1.53 firmware 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 for the "BLuAC5-Si"

The power-up packet is always sent at 9600 baud, regardless of the bit rate set by the BR or PB command.

If an Applied Motion software application is present it will respond to the power-up packet and communications will continue at 9600 baud. If an Applied Motion 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.

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 a drive (via either LEDs or a 7-segment display on the drive, see Appendix F), 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 F 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 C: 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 Q drive, so that you may be communicating to and familiarizing yourself with your drive as quickly as possible. All software from Applied Motion communicate to a drive via the supplied RS-232 programming cable. These software include:

Quick Tuner ----------used for tuning and configuring servo drives
Configurator----------used for configuring your stepper drives
Q Programmer ----------create and edit stored Q programs, emulate a host
SCL Setup Utility ----------basic host terminal for host emulation

If your project calls for a Q drive (or drives) running stored programs, you will use the supplied RS-232 programming cable along with Quick Tuner or Configurator and Q Programmer to setup, configure, and program your drive(s). If your project calls for your drive(s) only running stored programs, you can read up on the RS-232 subsection in this section and not read any more about the other serial connections. However, 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. RS-422 was not designed for a true multi-node network.

2-wire interfaces require one more significant feature. A network node, 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 Applied Motion 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”. This is because unlike the standard RS-422, which is designed for single-node connections, the 4-wire RS-485 used by

Applied Motion drives allows multiple nodes.

NOTE: In general we recommend using half-duplex communications with the drives. Even though the 4-wire RS-485 network can support full-duplex, there is now the capability to have multiple nodes and therefore data collisions might occur. For this reason we recommend limiting communications to half-duplex, even with the 4-wire RS-485 connections.

Connecting to your Q drive's serial port(s)

Each drive comes with one or two physical connectors for connecting to a PC or other serial host controller device. One connector is an RJ11 connector (same as a 4-wire phone jack) that is used strictly for RS-232 communications. The second connector is a removable 5-position terminal block for use with 2-wire and 4-wire RS-485 connections.

COM Port Settings

When using software from Applied Motion Products 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 either a USB to Serial adapter or a PCMCIA Serial adapter. 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 your Applied Motion software. Also, if you are having troubles with your adapter, contact Applied Motion for help with recommended adapters.

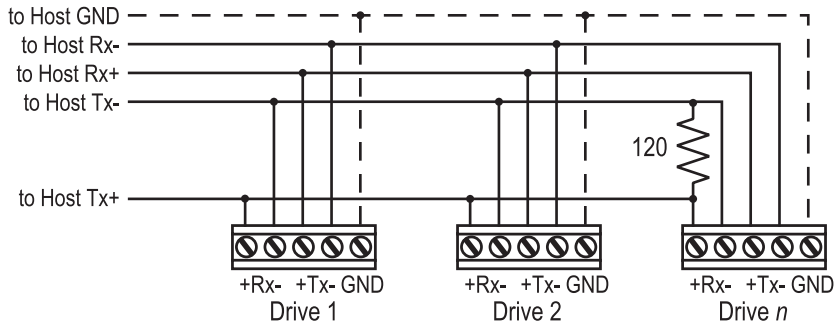
- Now take the 7 foot cable and plug one end into the adapter you just attached to your PCs COM port, and plug the other end into the RS-232 (RJ11) jack on the drive. If you need to locate your drive farther from the PC, you can replace the 7 foot cable with any 4-wire telephone cord. Do not exceed 50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

WARNING: Never connect an Applied Motion Products drive to a telephone circuit. It uses the same connectors and cords as telephones and modems, but the voltages are not compatible.

Connecting to a host using 4-wire RS-485

An Applied Motion drive's 4-wire RS-485 implementation is a multi-drop network with separate transmit and receive wires. One pair of wires connects the host's TX+ and TX- signals to each drive's RX+ and RX- terminals. Another pair connects the RX+ and RX- signals of the host 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 host's signal ground. Before wiring the entire system you'll need to connect each drive individually to the host 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.



NOTE: Proper cable shielding is a must. High voltage, high frequency, high current signals that are present on the servo 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.

Getting and Connecting an RS-485 4-wire adapter to your PC.

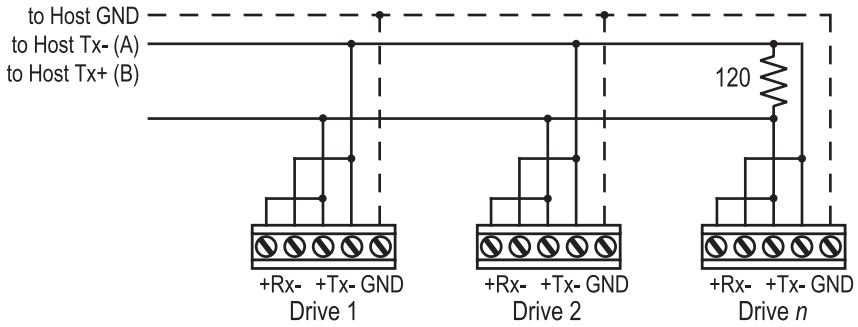
If you are using your computer to communicate to the drive(s) and therefore need an RS-485 adapter, model 117701 from Jameco Electronics (800-831-4242) works well. This adaptor is for a 25-pin serial port. If you are like most people and have a 9-pin serial port on your PC, you will also need to purchase Jameco cable 31721. Connect as follows:

Adaptor Terminal	Drive Terminal
1	RX+
2	RX-
3	TX-
4	TX+

Set the switches on the Jameco adaptor for DCE and TxON, RxON. Don't forget to plug in the DC power adapter that comes with the unit.

Connecting to a host using 2-wire RS-485

An Applied Motion 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 the host, you will need to connect the TX+/RX+ terminals of the drive to the host's TX+/RX+ terminal, and then the TX-/RX- terminals of the drive to the host'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.



Getting and Connecting an RS-485 2-wire adapter to your PC.

If you are using your computer to communicate to the drive(s) and therefore need an RS-485 adaptor, model 485-25E from Integrity Instruments (800-450-2001) works well. It comes with everything you need. Connect as follows:

Adaptor Terminal	Drive Terminals
A	TX+/RX+
B	TX-/RX-

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 the *SCL Setup Utility* on your PC. If you don't have the *SCL Setup Utility* installed, you can get it either from the CD-ROM that came with your drive or from Applied Motion's web site, www.applied-motion.com/support/software.php.

Once the *SCL Setup Utility* 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 D: The PR Command

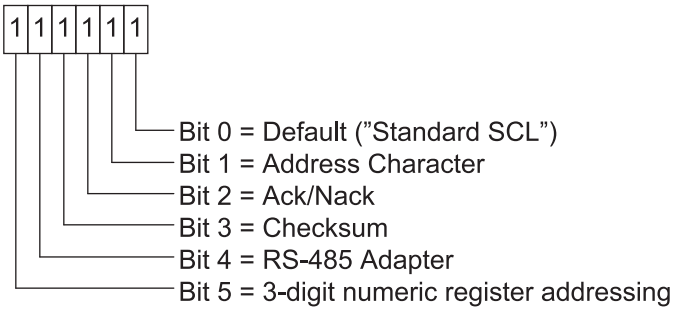
Because of the intense nature of serial communications required in host mode applications, you are allowed to adjust a drive's serial communications protocol to best fit your application. This adjusting of a drive's serial communications protocol is done using the PR command.

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 6-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 6-bit word as a bank of 6 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

The diagram to the right shows the assignments of each of the 6 bits in the protocol word. 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

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 (use 8-bit checksum)

Not implemented at this time. Call factory for schedule.

Bit 4 - RS-485 Adapter mode

Allows using a drive as an RS-232 to RS-485 adapter by letting the host communicate on an RS-485 network through the first drive’s RS-232 port. When the host sends commands with a “~” (tilde) at the beginning of the command to the first drive’s RS-232 port, the command is echoed out of both of that drive’s RS-232 and RS-485 ports. Drives connected on the RS-485 network will receive the same command with the “~” stripped off.

Without the Bit 4 option (Bit 4=0), a drive will normally echo any addressed command out of the RS-232 port only, whether the command was received from the drive’s RS-232 or RS-485 port. What the Bit 4 setting does (Bit 4=1), is force the drive to echo commands out the RS-485 port as well, allowing a host that is connected to a drive through its RS-232 port, to communicate to an RS-485 network of drives.

NOTE: When both Bits 4 and 2 are set (Bit 4=1, Bit 2=1), the host will receive back both the echoed packet and the acknowledge packet. For example, two drives are connected in an RS-485 network, and they both have PR command Bits 4 and 2 set. The first drive, which is also connected to the host via its RS-232 port, is addressed “1”, and the second drive is addressed “2”. Here is what you will see:

Send data Example:

Command	Drive Sends	Notes
~2DI8000	2DI8000	Drive at address “1” echoes original command over both serial ports
	2%	Drive at address “2” responds with ack.

Request data Example:

Command	Drive Sends	Notes
~2DI	2DI	Drive at address “1” echoes original command over both serial ports
	2DI8000	Drive at address “2” responds with distance

Bit 5 - 3-digit numeric register addressing

Each data register in a drive is normally accessed using its single letter, number, or other ascii character. With Bit 5 set (Bit 5=1), each of the data registers is instead accessed with a 3-digit number: 000 to 074. (See the Data Registers section for character and 3-digit numerical assignments). The Bit 5 option implements this specific usage for the RL (Register Load) and RU (Register Upload) commands.

NOTE: When data is returned from a drive (whether Bit 5 is set or not set), the data register is always represented by its single character designation.

RL Command Example:

Command	Drive Sends	Notes
RL017100	-	Load register 017 (“A”) with the value 100
RL017	RLA=100	Drive sends contents of acceleration register

RU command Example:

Command	Drive Sends	Notes
RU0174	RUA=100 RUB=150 RUC=140 RUD=210	Drive responds to register upload command by sending contents of 4 sequential data registers, starting with register 017 (“A”)

PR Command Examples

Now that you know what the bits in the PR command’s 6-bit binary word mean, here are a couple examples showing how you would set the serial communications protocol of your Q drive.

Example: Turn on Ack/Nack (Bit 2) and 3-digit numeric register addressing function (Bit 5)

The 6-bit word for this combination is - 100100 - and it’s decimal equivalent is 36. Therefore, to set your drive with this serial protocol, you would send the command “PR36” to your drive.

Example: Turn on RS-485 adaptor function (Bit 4)

The 6-bit word for this combination is - 010000 - and it’s decimal equivalent is 16. Therefore, to set your drive with this serial protocol, you would send the command “PR16” to your drive.

Appendix E: 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 servo system into your machine.

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 a code word can actually show information about more than one alarm or status condition.

Alarm Code Definitions

AL command

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

Hex Value	BLu	SV	STAC6	ST	STM
0001	***Position Limit***				
0002	CCW Limit				
0004	CW Limit				
0008	***Over Temp***				
0010	***Excess Regen****	***Internal Voltage***	***Excess Regen***	***Internal Voltage***	***Internal Voltage***
0020	***Over Voltage***				
0040	***Under Voltage****	Under Voltage	***Under Voltage***	Under Voltage	Under Voltage
0080	***Over Current***				
0100	***Bad Hall Sensor***		***Open Motor Winding***		
0200	***Bad Encoder***				(not used)
0400	Comm Error				
0800	Bad Flash				
1000	Wizard Failed		No Move		
2000	Current Foldback		Motor Resistance Out of Range	(not used)	(not used)
4000	Blank Q Segment				
8000	No Move		(not used)		

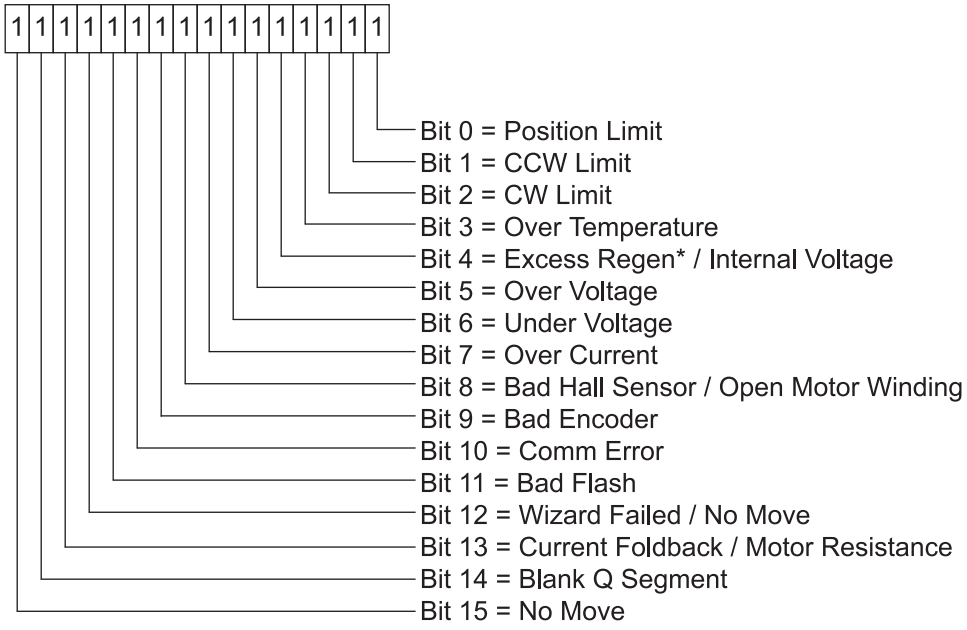
* BLuAC drives only
NOTE: Items in ***bold italic*** represent Drive Faults, which automatically disable the motor. Use the OF command in a Q Program to branch on a Drive Fault.

Example: The drive has hit the CW limit (0004), there is an under voltage condition (0040), and an encoder wiring connection has been lost resulting in an encoder fault (0200). The resulting Alarm Code is 0244, and when the host sends the "AL" command the drive will respond with "AL=244".

"f" data register

Another way to retrieve the Alarm Code is to use the "f" data register. If the host sends the RLf command, the response from the drive will be the decimal equivalent of the 16-bit Alarm Code word. The diagram below

shows the 16 bit assignments for the Alarm Code (which of course match the hexadecimal values above).



Alarm Code 16-bit word

Example: The drive has hit the CW limit (bit 2), there is an under voltage condition (bit 6), and an encoder wiring connection has been lost resulting in an encoder fault (bit 9). The resulting Alarm Code binary word is 0000 0010 0100 0100. The decimal equivalent of this word is 580, so the response from the drive to the RLf command will be “RLf=580”.

Status Code Definitions

SC command

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

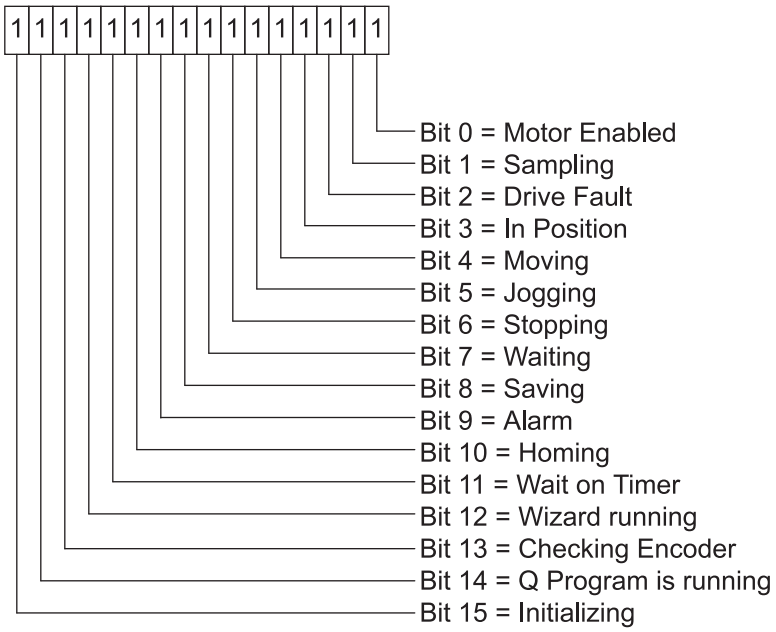
Hex Value	Status Code bit definition
0001	Motor Enabled (Motor Disabled if this bit = 0)
0002	Sampling (for Quick Tuner)
0004	Drive Fault (check Alarm Code)
0008	In Position (motor is in position)
0010	Moving (motor is moving)
0020	Jogging (currently in jog mode)
0040	Stopping (in the process of stopping from a stop command)
0080	Waiting for an input (executing WI command)
0100	Saving (parameter data is being saved)
0200	Alarm present (check Alarm Code)
0400	Homing (executing an SH command)
0800	Wait Time (executing a WT command)

Hex Value	Status Code bit definition
1000	Wizard running (Timing Wizard is running)
2000	Checking encoder (Timing Wizard is running)
4000	Q Program is running
8000	Initializing (happens at power up)

Example: The drive is running a stored Q program (hex value 4000), it's in position (hex value 0008), and it's waiting for the input specified by the WI command (hex value 0080). The Status Code for this condition is 4088, and when the host sends the "SC" command the drive will respond with "SC=4088".

"s" data register

Another way to retrieve the Status Code is to use the "s" data register. If the host sends the RLs command, the response from the drive will be the decimal equivalent of the 16-bit Status Code word. The diagram below shows the 16 bit assignments for the Status Code (which of course match the hexadecimal values above).



Status Code 16-bit word

Example: The drive is running a stored Q program (bit 14), it's in position (bit 3), and it's waiting for the input specified by the WI command (bit 7). The resulting Status Code binary word is 0100 0000 1000 1000. The decimal equivalent of this word 16,520, so the response from the drive to the RLs command will be "RLs=16520".

A useful tool for converting between binary, decimal, and hexadecimal.

If you're using a Windows-based PC as a host with your drive (which you'll definitely be doing at some point during the project), you can use the Calculator utility that comes with Windows to convert Alarm and Status Codes between binary, decimal, and hexadecimal values. This utility is usually found in Start Menu, Programs, Accessories. Once open, make sure to choose Scientific view from the View menu of Calculator. This view provides radio buttons for Hex, Dec, and Bin.

To figure out what your Alarm or Status Code is telling you, first select the appropriate radio button (Hex for the AL or SC commands, Dec for the "f" and "s" registers), then enter the response from the drive. Now you can toggle between Hex, Dec, and Bin to compare the values to the tables and diagrams above. Note: Calculator does not show leading zeros in a binary number, so you may see less than 16 bits when you select Bin. That's OK, just start counting from the right with Bit 0 and you'll be able to determine the conditions set in the codes.

LED and 7-Segment Display codes

In addition to the Alarm and Status codes, most drive alarms and faults as well as some status codes are displayed at the front of the drives, via either two-color flashing LED codes or 7-segment LED codes. The following tables show the various codes available with Applied Motion drives.

BLuDC LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
-	flashing quickly	Q program running (Q drives only)
1	1	***position limit***
1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit
3	1	***over temperature (> 85 deg C)***
3	2	bad flash
4	1	***over voltage (> 55 Vdc)***
4	2	under voltage (< 18 Vdc)
5	1	***over current / short circuit***
5	2	current limit
6	1	***bad hall***
6	2	***bad encoder***
7	1	serial communication error

BLuAC 7-Segment codes

Items in ***bold italic*** are Drive Faults.

	Position Mode		***Over Temp***		Comm Error
	Velocity Mode		***Over Voltage***		Move attempted while disabled
	Torque Mode		***Under Voltage***		Drive Start-up
	Step Mode		***Over Current***		Bad Flash
	Si Mode		Current Limit		Comm Time-out

	Drive Disabled		***Hall Bad***		Stack Overflow
	Position Limit		***Bad Encoder***		Stack Underflow
	CCW Limit		Memory Failed		Q Program Running
	CW Limit		***Excess Regen***		Drive enabled when flashing

SV LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
-	flashing quickly	Q program running (Q drives only)
1	1	***position limit***
1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit
3	1	***over temp***
3	2	***internal voltage out of range***
3	3	attempt to load blank Q segment
4	1	***over voltage***
4	2	under voltage
5	1	***over current / short circuit***
5	2	current limit
6	1	***bad hall***
6	2	***bad encoder***
7	1	serial communication error
7	2	bad flash

STAC6 LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
-	flashing quickly	Q program running (Q drives only)
1	1	***motor stall (w/ optional encoder only)***

1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit
3	1	***over temp***
3	2	***excess regen***
4	1	***over voltage***
4	2	***under voltage***
5	1	***over current / short circuit***
5	2	motor resistance out of range
6	1	***open motor winding***
6	2	***bad encoder signal (w/ optional encoder only)***
7	1	serial communication error

ST-Q/Si LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
-	flashing quickly	Q program running (Q drives only)
1	1	***motor stall (w/ optional encoder only)***
1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit
3	1	***over temp***
3	2	***internal voltage out of range***
4	1	***over voltage***
4	2	under voltage
5	1	***over current / short circuit***
6	1	***open motor winding***
6	2	***bad encoder signal (w/ optional encoder only)***
7	1	serial communication error

ST-S LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit

3	1	***over temp***
3	2	***internal voltage out of range***
4	1	***over voltage***
4	2	under voltage
5	1	***over current / short circuit***
6	1	***open motor winding***
7	1	serial communication error

STM LED codes

Items in ***bold italic*** are Drive Faults.

		DESCRIPTION
-	solid	Motor disabled
-	flashing slowly	Motor enabled
-	flashing quickly	Q program running (Q drives only)
1	1	***motor stall (w/ optional encoder only)***
1	2	move attempted while disabled
2	1	CCW limit
2	2	CW limit
3	1	***over temp***
3	2	***internal voltage out of range***
4	1	***over voltage***
4	2	under voltage
5	1	***over current / short circuit***
6	1	***open motor winding***
7	1	serial communication error

Appendix F: Working with Inputs and Outputs

This Appendix overviews a few key concepts regarding logic-level inputs and outputs:

- Low v. High
- I/O Connectors on BLu and STAC drives
- Configuring specific inputs and outputs
- Digital Input Filters

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, WIX4L means "wait for input X4 low", and SO1L means "set output 1 low". A high state is represented by the "H" character.

I/O Connectors on BLu and STAC drives

"Main driver board" I/O connector

All BLu and STAC drives have a "main driver board" inputs/outputs (I/O) connector. On BLuAC and STAC drives this connector is a female DB-25 connector labeled "IN/OUT 1". On BLuDC drives this connector is a female DB-25 connector labeled simply "I/O".

The "main driver board" connector houses the following inputs and outputs:

Type	Connector ID	Function
Analog Inputs	Analog IN+	Analog Input 1 (X9)
	Analog IN-	Analog Input 2 (X:)
Digital Inputs	X1/STEP	Digital Input X1
	X2/DIR	Digital Input X2
	X3/Enable	Digital Input X3
	X4/Alarm reset	Digital Input X4
	X5	Digital Input X5
	X6/CCW Limit	Digital Input X6
	X7/CW Limit	Digital Input X7
Digital Outputs	Y1/BRAKE	Digital Output Y1
	Y2/MOTION	Digital Output Y2
	Y3/FAULT	Digital Output Y3

Analog Inputs

Analog Inputs 1 and 2 can be interpreted by a drive as individual inputs. When issuing a host command that acts upon an input, Analog Input 1 (physical connection Analog IN+) is designated by "X9", and Analog Input 2 (physical connection Analog IN-) is designated by "X:". It is also possible to issue a command that acts upon the differential signal between Analog Inputs 1 and 2, and this is known as the Analog Command. When issuing a host command that acts upon an input, the Analog Command is designated by "X8".

The Analog Command (X8) is affected by the following commands: AD (Analog Deadband - STAC only), AF (Analog Filter), AS (Analog Scaling - STAC only), AV (Analog Offset Value), and AZ (Analog Zero). Analog Inputs 1 (X9) and 2 (X:) are affected by the AF (Analog Filter) command only.

Digital Inputs

When issuing commands that act upon an input, designate inputs on the main driver board with an “X” first, as they are shown in the table above: X1 through X7. This is not necessary for drives that only have a main driver board I/O connector and not a “top board” or “expanded” I/O connector, because these drives only have one connector of I/O and will therefore act upon input X3 whether it is designated as “X3” or simply “3” within a command. However, in practice it is a good idea to use the “X” character whenever accessing inputs on the main driver board I/O connector.

Digital Outputs

Similar to digital inputs on the main driver board I/O connector, outputs on this connector are designated with a “Y” character. This avoids confusion when issuing commands that act on an output in drives that have two I/O connectors - the main driver board I/O connector and the “top board” or “expanded” I/O connector.

“Top board” or “expanded” I/O connector

In addition to the “main driver board” I/O connector, the following drives have a “top board” I/O connector as well: drive model numbers ending in SE, QE, and Si. On BLuAC and STAC drives this connector is a male DB-25 connector labeled “IN/OUT 2”. On BLuDC drives this connector is a screw terminal type.

The “top board” connector houses the following inputs and outputs:

Type	Connector ID	Function
Analog Inputs	AIN	Analog Input*
Digital Inputs	IN1	Digital Input 1
	IN2	Digital Input 2
	IN3	Digital Input 3
	IN4	Digital Input 4
	IN5	Digital Input 5
	IN6	Digital Input 6
	IN7	Digital Input 7
	IN8	Digital Input 8
Digital Outputs	Out 1	Digital Output 1
	Out 2	Digital Output 2
	Out 3	Digital Output 3
	Out 4	Digital Output 4

* The analog input of the “top board” I/O connector is not implemented at this time.

Digital Inputs and Outputs

When issuing commands that act upon inputs or outputs of the “top board” I/O connector, designate those inputs and outputs simply by their number assignments: digital inputs 1 - 8, and digital outputs 1 - 4.

Configuring specific inputs and outputs

There exist host commands to access and define all of the inputs (both digital and analog) and outputs of a drive. In many cases, especially in the case of the analog inputs, it is much easier to use one of the setup software utilities available - Quick Tuner for servo drives, and Configurator for stepper drives - to define the parameters that affect the various inputs and outputs.

For example, rather than use the AD, AF, AS, AT, AV, and AZ commands to define the parameters of the Analog Command (analog inputs 1 and 2 of the “main driver board” I/O connector), these parameters can be set in the Configurator software which makes it much simpler. Similarly, some of these same Analog Command parameters can be set in the Quick Tuner software when using a servo drive.

As another example, the DL (Define Limits) command determines how the end of travel limit (digital) inputs will be used in a drive. Rather than using the DL command, the end of travel limit inputs can also be set in Configurator or Quick Tuner, which is usually a little easier.

Digital Input Filters

See FI and FX commands in the Command Listing section.

(this page intentionally left blank)

(this page intentionally left blank)

Applied Motion Products

404 Westridge Drive

Watsonville, CA 95076

USA

tel / 831-761-6555

fax / 831-761-6544

www.applied-motion.com

920-0002 Rev. D
7/15/2007