red lion

Tel +1 (717) 767-6511
Fax +1 (717) 764-0839
www.redlion.net

Applied Motion SCL Serial Driver

Information Sheet for Crimson v2.0

Compatible Devices

- Applied Motion Drives using SCL

Verified Devices

- STAC6-Q
- BLuDC4-Q
- Si3540

Device Options:

Drop Character – Select the address character. !, %, *, ?, @ are rejected.
Send Drop Char – Select as appropriate.
Device Type – Select BLU/STAC6 or Legacy as required.
Disable PM2 On Power Up – Allows the programmer to override the automatic default setting, by the driver, of the PM2 communications control.

NOTE: Whenever the driver connects to a BLU/STAC6 drive, it sets the drive to report all Immediate responses in decimal. The programmer is advised to clear the checksum-enabled bit, and set both the **Enable Address**, and **Acknowledge**, bits for those device types.

Accessible Data:

Commands are organized in groups according to usage, as defined by the SCL protocol document.

NOTE: Commands that apply to more than one command set appear in each list that the specification defines. E.g. **CT** (Continue) appears in both **Q-Program** and in **Immediate**. It is the same command in either case.

NOTE: Not all commands are available in all units. E.g. **PI** is valid only for the STAC, **PP** is valid only for the BLU. The programmer is responsible for verifying the command is valid for the type of drive connected. * represents commands available in both device types, # are commands only for Legacy devices. Unmarked and * commands will be sent to all BLU/STAC6 drives. * and # commands will be the only ones sent to Legacy drives.
Legacy products include: 1240i, 3540i, 7080i, Si2035, Si3540, Si5580

MOTION COMMANDS

Prefix	**Description**	**Notes**
AC	Acceleration	*
AM	Max Acceleration	*
CJ	Commence Jogging	*
DC	Change Distance	
DE	Deceleration	*
DI	Distance/Position	*
EG	Electronic Gearing	
EP	Encoder Position	
FC	Feed to Length, Spd Change	
FD	Feed Double Sensor (1=FD, 2=FDX)	1,8
FDA	FD(X) Input Number 1	1
FDB	FD(X) Condition 1	1,2
FDC	FD(X) Input Number 2	1
FDD	FD(X) Condition 2	1,2
FE	Follow Encoder	2
FEX	Follow Encoder, X	2
FLC	Feed/Length Command	*
FLV	Feed/Length with Value	
FM	Feed/Sensor, Mask Dist.	2
FMX	Feed/Sensor, Mask Dist. X	2
FO	Feed/Length + Set Out (1=L, 2=H)	8
FOY	Feed/Length + Set Out. Y (1=L, 2=H)	8
FPC	Feed/Position Command	*
FPV	Feed/Position with Value	
FS	Feed/Sensor	2*
FSX	Feed/Sensor, X	2
FY	Feed/Sensor, Safety Dist.	2
FYX	Feed/Sensor, Safety Dist.X	2
HW	Hand Wheel	2
HWX	Hand Wheel X	2
JA	Jog Acceleration	*
JC	Velocity mode second speed	
JD	Jog Disable	*
JE	Jog Enable	*
JL	Jog Deceleration	
JM	Jog Mode	
JS	Jog Speed	*
SH	Seek Home	2*
SHX	Seek Home, X	2
SJ	Stop Jogging	*
SM	Stop Move (1=SMD, 2=SMM)	8

SP	Set Position	*
ST	Stop (1=ST, 2=STD)	8*
VC	Velocity Change	
VE	Velocity	*
VM	Maximum Velocity	

CONFIGURATION COMMANDS

Prefix	Description	Notes
BD	Brake Disengage Delay	
BE	Brake Engage Delay	
CC	Change Current	*
CD	Idle Current Delay Time	
CF	Anti-resonance Filter Freq.	
CG	Anti-resonance Filter Gain	
CI	Change Idle Current	
CM	Control Mode (Command Mode)	
CP	Change Peak Current	
DA	Define Address	*
DL	Define Limits (X6, X7)	*
DR	Data Register (Capture)	
ED	Encoder Deadband	#
EF	Encoder Function	#
ER	Encoder Resolution	*
HG	4th Harmonic Filter Gain	
HP	4th Harmonic Filter Phase	
KC	Overall Servo Filter	
KD	Differential Constant	
KE	Differential Filter	
KF	Velocity FeedFwd Constant	
KI	Integrator Constant	
KK	Inertia FeedFwd Constant	
KP	Proportional Constant	
KV	Velocity Feedback Constant	
KW	Velocity Filter	
MO	Motion Output (Y2)	
MR	Microstep Resolution	#
PC	Power-up Current	*
PF	Position Fault/Torque %	
PI	Power-up Idle Current	
PL	Position Limit	
PM	Power-up Mode	*
PP	Power-up Peak current	
SA	Save Parameters	

SF	Step Filter Frequency	
SI	Enable Input Usage	
VI	Velocity Integrator Constant	
VP	Velocity Mode Prop. Constant	
ZC	Regen Resistor Cont. Wattage	
ZR	Regen Resistor Value	
ZT	Regen Resistor Peak Time	

I/O Commands

Prefix	**Description**	**Notes**
AD	Analog DeadBand	
AF	Analog Filter	
AG	Analog Velocity Gain	
AI	Alarm Input Usage	
AO	Alarm Output (Y3)	
AP	Analog Position Gain	
AS	Analog Scaling	
AT	Analog Threshold	
AV	Analog Offset Value	
AZ	Analog Zero	
BD	Brake Disengage Delay	
BE	Brake Engage Delay	
BO	Brake Output (Y1)	
DL	Define Limits (X6, X7)	*
FI	Filter Input	
FX	Filter select inputs	
IH	Immediate High Output	
IHY	Immediate High Output, Y	
IL	Immediate Low Output	
ILY	Immediate Low Output, Y	
IS	Input Status	*
ISX	Input Status, X	
MO	Motion Output (Y2)	
SO	Set Output (1=L, 2=H)	8*
SOY	Set Output, Y (1=L, 2=H)	8
SI	Enable Input Usage	

COMMUNICATION COMMANDS

Prefix	Description	Notes
TD	Transmit Delay	*
ZERR	Error Code Received	3*
G32D	Restart Comms after PC Disable	Appendix*
PC2D	PC to Drive (Enable=1, Disable=0)	Appendix*
IFD	Force Decimal Mode for Immed. Cmds	

Q Program Commands

Prefix	Description	Notes
CT	Continue	
MT	Multi-Tasking	
PS	Pause	*
QC	Queue Call (QC)	9
QCN	Queue Call, (QC<segment #>)	9
QD	Queue Delete	
QE	Queue Execute	11
QK	Queue Kill (1=QKD,2=QKM)	8
QL	Queue Load	11
QS	Queue Save	
QX	Queue Load & Execute	
SM	Stop Move	
WD	Wait Delay	
WT	Wait Time	*

DRIVE COMMANDS

Prefix	Description	Notes
AL	Alarm Code	
AX	Alarm Reset Buffered	
EP	Encoder Position	*
MD	Motor Disable	*
ME	Motor Enable	*
MN	Model Number	
SA	Save Parameters	
SC	Status Code	

REGISTER COMMANDS

Prefix	Description	Notes
DR	Data Register (Capture)	
RC	Reg. Counter	
RCX	Reg. Counter, X	
RD	Reg. Decrement (<data> = 0 - Z)	4
RI	Reg. Increment (<data> = 0 - Z)	4

RL	Reg. Load Immediate (a-z Read Only)	5
RM	Reg. Move (Selection -> <data 0-Z>)	6
RR	Reg. Read (<data> = NV memory)	7
RW	Reg. Write (<data> = NV memory)	7
RX	Reg. Load Buffered (a-z Read Only)	5
TS	Time Stamp	

IMMEDIATE COMMANDS

Prefix	Description	Notes
AL	Alarm Code	
AR	Alarm Reset immediate	
BS	Buffer Status	*
CS	Change Speed	
CT	Continue	
GC	Current Command	
IA	Immediate Analog (IA, IA1...IA3)	
IC	Immediate Current (Commanded)	
ID	Immediate Distance	*
IE	Immediate Encoder	*
IH	Immediate High Output	
IHY	Immediate High Output, Y	
IL	Immediate Low Output	
ILY	Immediate Low Output, Y	
IO	Output Status	
IOY	Output Status, Y	
IP	Immediate Position	*
IQ	Immediate Current (Actual)	
IS	Input Status	*
ISX	Input Status, X	
IT	Immediate Temperature	
IU	Immediate Voltage	
IVA	Immediate Actual Velocity	
IVT	Immediate Target Velocity	
IX	Immediate Position Error	
MN	Model Number	
QE	Queue Execute	
QL	Queue Load	
QS	Queue Save	
RE	Restart or Reset	
RL	Register Load Immediate	
RS	Request Status (4 chars/item)	10*
RV	Revision Level	*

SJ	Stop Jogging	
SK	Stop & Kill (1=SK, 2=SKD)	8*
ST	Stop (1=ST, 2=STD)	8*

IMPORTANT NOTE:

For BLU/STAC6, when a buffered command is received by the driver, it checks the Status Code (**SC)**. A buffered command is rejected, and the display will show ----- if any status bit is set other than those in the following list.
Status bits ignored when a buffered command is requested:
0 – Motor Enabled
2 – Drive Fault
3 – In Position
9 – Alarm Present

Accessibility Notes:

Note 1: FD – Feed Double Sensor has 4 parameters (Input numbers and Conditions) that must be stored before setting **FD** to a 1 (send ‘FD’), or a 2 (send ‘FDX’).

Note 2: These selections accept F, R, H, and L, for Falling, Rising, High, and Low, respectively, for the signal conditions.

Note 3: ZERR – Error Code Received. This is an internal value that will be set if a command cannot be executed. The upper 2 bytes are the command letters, and the lower 2 bytes are the error code. See the SCL Protocol documentation for error values.
The default value, or any write, will set **ZERR** to 0x20202030 (“   0”). If **ZERR** becomes 0x42555359 (“BUSY”), it means another command was buffered during motion. The display for buffered commands may blank during the motion. When motion is complete, the value will be restored, and **ZERR** will be restored to the correct value. If **ZERR** is made a String Tag, you **must** set “Length” to 4. Otherwise, configure it analogously to Request Status in Note 10 below.

Note 4: RD, RI – Register Decrement and Register Increment. The programmer arranges to send the ASCII value of the register to be operated upon. E.g. to decrement User Register 5, set the data to ‘5’ (decimal 53, hexadecimal 35).

Note 5: RL, RX – Register Load Immediate, Register Load Buffered. The programmer selects the desired ASCII symbol for the intended register. The valid values are any ASCII character from ‘0’ to ‘z’. Note that registers a – z cannot be written. However, all registers can be read.

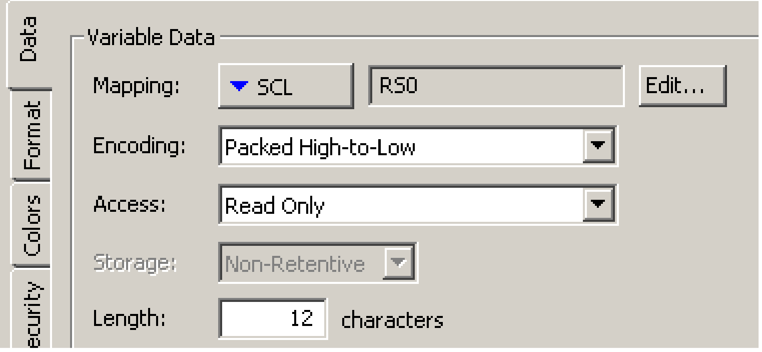
Note 6: RM – Register Move. The programmer selects the desired source register 0 - z in the configuration. When the command is to be executed, the programmer arranges to write the ASCII character for a destination register in the range '0' – 'Z'.
E.g. to move register 'a' to user register 3, the programmer selects RMa, and writes '3' (decimal 51, hexadecimal 0x33).

Note 7: RR, RW – Register Read and Register Write. The programmer selects the desired register '0' – 'Z' in the configuration. When the command is to be executed, the programmer writes the address of the non-volatile memory location to be accessed. **RR** transfers the NV to the selected register, and **RW** transfers the value in the selected register to the NV location. **RR** and **RW** items always return 0 when read.

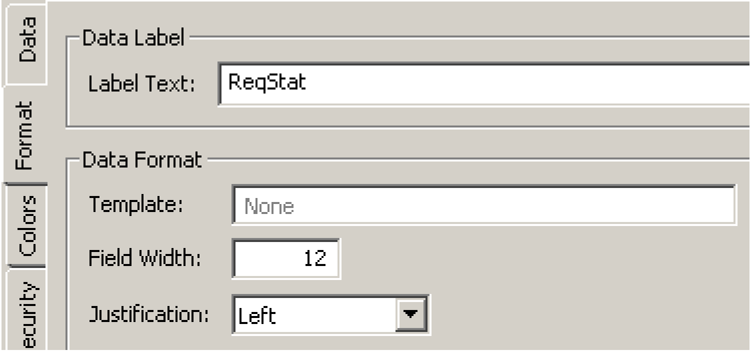
Note 8: Various commands. These items require either a 1 or a 2 as data in order to select the appropriate function. See the specification for details for the appended suffix.

Note 9: QC, QCN – Queue Call (**QC** = return to calling segment, **QCN** = call segment n). In order to permit calling segment 0, **QCN** returns the value 409600000 (0x186A0000) to provide 5 decimal 0's or 4 hexadecimal 0's when read. Configuring the item for 5 decimal digits or 4 hexadecimal digits, or fewer, is recommended. The 409600000 value can never be written.

Note 10: RS – Request Status. This command should be assigned to a String Tag, and configured as shown below:



Mapping: SCL RS0
Encoding: Packed High-to-Low
Access: Read Only
Storage: Non-Retentive
Length: 12 characters



Data Label

Label Text: ReqStat

Data Format

Template: None

Field Width: 12

Justification: Left

This will display the letter combinations of the status without the need to interpret the numerical values of the bytes. E.g. "PR" for In Position, and Ready. The characters are always shifted into the higher bytes, so the hex value of the display "PR" would be 0x50520000.

If a buffered command read cannot be executed because the drive is in motion, the driver executes an **RS** command prior to sending any further buffered command reads. If the drive is still in motion, those read commands are not sent. Once motion is concluded, the buffered command reads will be executed. Write commands will be sent, whether or not the drive is in motion.

Note 11: QL – Queue Load has two different forms:

Form 1: **QL1 – QL12** can be used to specify a segment load prior to selecting **QE**. The numeric value written will be stored in the driver until a **QE** is commanded. When the **QE** is received, the two commands will be sent. This is equivalent to doing **QXn**, but may have usefulness in some applications.

Form2: **QL** with no parameter will be sent immediately whenever the written data is greater than 12, the maximum number of segments. Since this is used to send buffered commands to the drive, the only immediate command sent can be **QSn**, where n is a valid segment number. Once the **QSn** is sent, the driver will re-enable the sending of immediate commands.
Should more than 62 buffered commands be sent before the **QS** is encountered, the driver will issue a "Stop and Kill" command, and resume normal operation.

APPENDIX

Applied Motion suggests powering the Drive only after any device that is communicating with it is active. The driver provides a way to avoid power cycling under most circumstances. Generally, the Applied Motion Software (AMSW) and the SCL driver cannot both be running at the same time.

PC2D: This selection, when set to 1, disables driver communications without the need to disable the device itself, allowing AMSW to run without interference.

G32D: This command issues a code that tells the Applied Motion drive to accept communications from the driver.

USING PC2D and G32D:

When using a G3 display, it is recommended that these two items be put on a page containing no items that read from the drive. Following the procedure below should allow easy switching of communications between the drive and AMSW and driver.

Step 1:

> Before connecting the AMSW, set **PC2D** to 1. External communication from the driver will cease, internal values will continue to be displayed. All writes, except to **PC2D** and **G32D**, will be disabled.

Step 2:

> Connect the AMSW. In some cases, it will be necessary to select the proper model number, enter SCL Terminal mode, and issue the command QT.

Step 3:

> Once finished with the AMSW, close it, or select the Terminal Mode, which will disable the SW's programming communication.

Step 4:

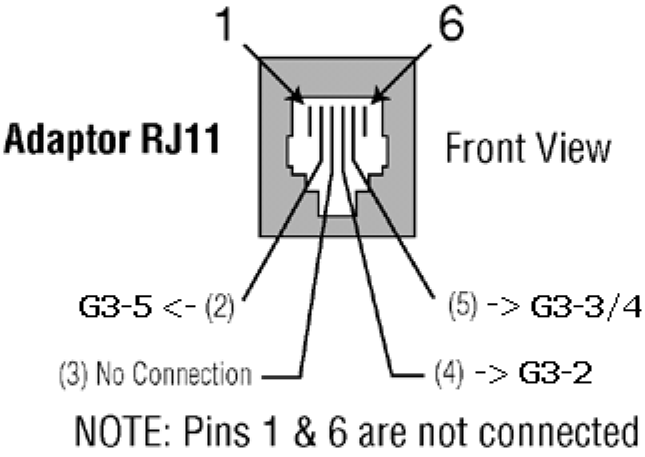
> Set **PC2D** to 0. If communications do not resume, set **G32D** to 1 in order to issue a command to the drive to resume SCL communications.

If the above steps don't result in communication, power cycling of the Drive will be required.

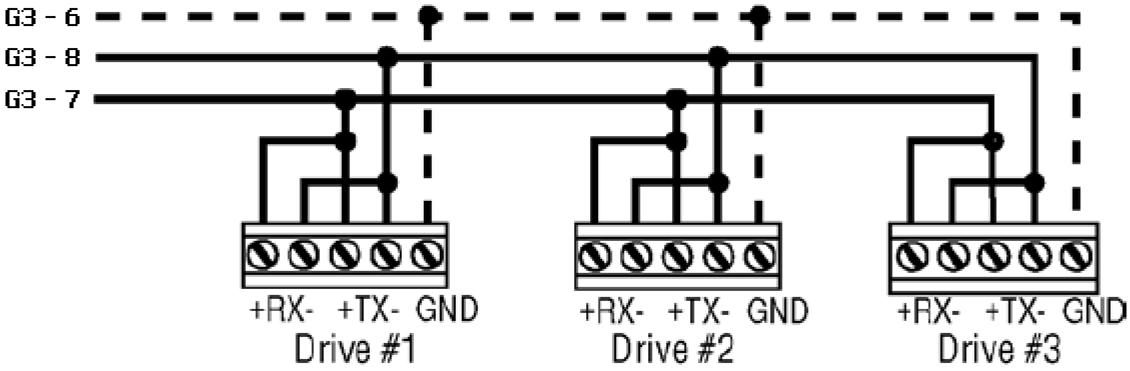
The driver has a feature that allows "On Startup" to set **PC2D** = 1. No driver communications will occur until **PC2D** is reset, yet **PC2D** and **G32D** will remain accessible. This allows AMSW to communicate properly when all units power simultaneously. **PC2D** will display 5678 until it is reset.

Serial Port Connections

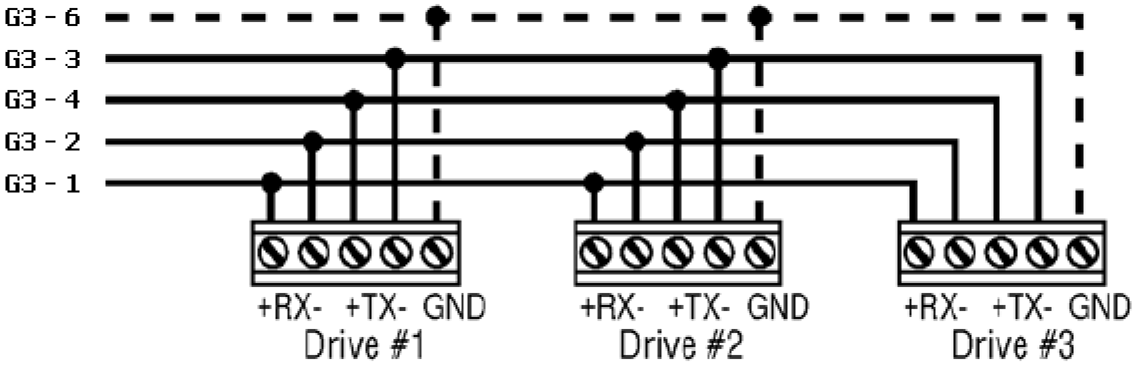
RS-232



RS-485 (2 Wire)



RS-485 (4 Wire)


G3 - 6
G3 - 3
G3 - 4
G3 - 2
G3 - 1
+RX- +TX- GND
Drive #1
+RX- +TX- GND
Drive #2
+RX- +TX- GND
Drive #3