

AUTOMATION MODULES, INC



*SD100*

TECHNICAL REFERENCE MANUAL

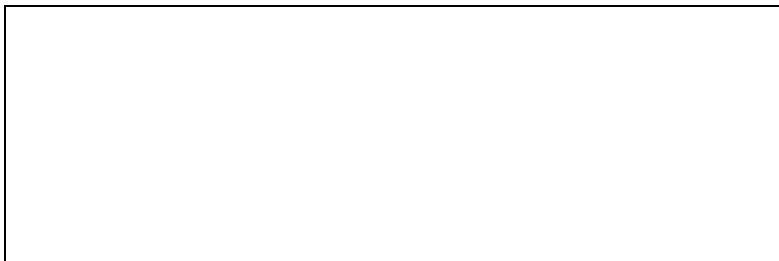
Revision 3.2, April 2002

**AUTOMATION MODULES, INC.**

**FOX ISLAND, WA 98333**

**PHONE: 1-253-549-4868 / FAX: 1-253-549-4866**

**FOR TECHNICAL ASSISTANCE CALL:**



**© COPYRIGHT 2002, AUTOMATION MODULES, INC.**

- 1. INTRODUCTION..... 4
- 1.1 SPECIFICATIONS..... 4
- 1.2 DEDICATED DIGITAL INPUTS..... 5
  - 1.2.1 Command Pulse Inputs..... 5
  - 1.2.2 Additional Inputs ..... 5
  - 1.2.3 Dedicated Digital Outputs..... 6
- 1.3 INCREMENTAL ENCODER INTERFACE ..... 7
- 1.4 OUTPUT DRIVER INTERFACE ..... 7
- 1.5 RS-232 SERIAL INTERFACE..... 7
- 2. MODES OF OPERATION ..... 8
  - 2.1 POSITION MODE ..... 8
  - 2.2 TORQUE MODE..... 8
  - 2.3 SWITCHING MODES ..... 8
  - 2.4 OUTPUT LIMITING ..... 9
- 3. ENTERING COMMANDS..... 10
  - 3.1 DOWNLOADING COMMANDS ..... 11
  - 3.2 COMMANDS..... 11
- 4. APPENDIX A, SD100 ERROR CODE DEFINITIONS..... 18
- 5. APPENDIX B, SUMMARY OF SD100 COMMANDS..... 19
- 6. APPENDIX C, SD100-R1 CONNECTOR PIN DEFINITIONS..... 20

## 1. Introduction

The SD100 is a device capable of allowing a motion controller with only step and direction (or count up / count down) type signals to control a brush-type DC servomotor or voice coil actuator thereby allowing those to be used in applications formerly assigned to stepping motors.

The SD100 implements a mnemonic type command instruction set via a standard RS-232 serial communications interface. These commands are used to set various operating parameters which are then stored in non-volatile memory for automatic retrieval at power-up.

The SD100 interfaces to the real world via an onboard PWM DC motor driver, a quadrature type encoder interface, 8 dedicated optoisolated digital inputs and 4 dedicated optoisolated digital outputs and an RS-232 serial communications link.

### 1.1 Specifications

Description	Stand-Alone 1 Axis Servo Motor Indexer / Driver
Operating Modes	Position, Torque
Filter Algorithm	PID
Servo Loop Rate	10 KHz
Servo Position Feedback	Incremental Quadrature Encoder
Output	PWM Motor Drive, 3 Amps Cont. and 6 Amps Peak at 48 VDC Max.
PWM Frequency	Approximately 19.531 KHz
Encoder Input	Single-ended or Differential
Encoder Output	Differential
Encoder Supply Voltage	5 VDC
Encoder Input Voltage	5.5 VDC Max., -0.1 VDC Min.
Encoder Count Rate	2 Million Quadrature Counts per Second
Command Pulse Count Rate	1 Million Quadrature Counts per Second (count and/or direction state must remain unchanged for 500 nS minimum)
Dedicated Digital Inputs	Reset, Clear, Output_Limit, Mode (x3), Count Up / Step (+5VDC only), Count Down / Direction (+5VDC only), all optoisolated.
Dedicated Digital Outputs	Ready, Over-Temp, Overflow, In_Position, all optoisolated.
Communication Interface	RS-232 Serial Interface, 9600 Baud, 8 Bits, 1 Stop Bit, No Parity, XON / XOFF Handshake
Supply Voltage	+11 To +48 VDC
Motor Voltage	+12 To +48 VDC
Dimensions	Approximately 4.1" Long by 4.0" Wide by 1.1" Thick
Weight	Approximately 10 oz.

**Table 1. Specifications.**

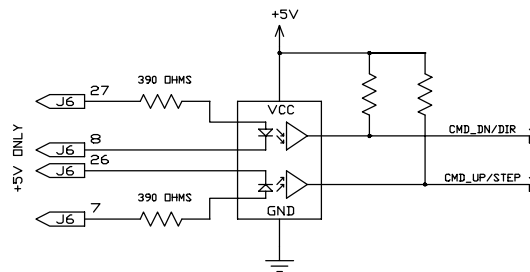
## 1.2 Dedicated Digital Inputs

### 1.2.1 Command Pulse Inputs

The SD100 provides several digital inputs to enhance its operation and to allow for more versatile modes of operation. The first of these are the Count Up / Step and Count Down / Direction inputs. Figure 1 shows a circuit example of these inputs.

Command pulses are accepted from the host motion controller via these inputs. Depending on the mode of operation set by the Phase (PH) command, the command pulse inputs will function as either independent Count Up and Count Down or paired as Step and Direction. In any case the actual count will take place at the **“on to off”** transition of the optoisolated input. **The state of any Command Pulse input must remain the same for at least 500 ns before changing** in order to be recognized by the SD100.

**!!! Please note that the command pulse inputs are designed !!!  
!!! to accept +5VDC signal levels only. !!!**



**Figure 1. Command Inputs**

### 1.2.2 Additional Inputs

Additional dedicated inputs include Reset, Clear, PM/QM, InitC\_Mode, INP\_Mode and Output\_Limit. Figure 2 shows a circuit example of these inputs.

The Reset input is used to reset the device and clear any errors. When the Reset input is held on the output will be disabled. As Reset is turned off, the encoder position as well as the commanded position are both set to zero.

The Clear input is used to place the SD100 in a known state. When the Clear input is turned on and then back off, the command counter, the encoder counter and all values associated with the PID loop are cleared to zero.

When turned on, the PM/QM input is used to put the SD100 into a constant current output mode of operation (known as Torque Mode or QM). When this input is then turned off, the unit will again return to the Position Mode of operation (PM). The InitC\_Mode input determines how the switch to QM will be initialized (see section 2.4 Switching Modes).

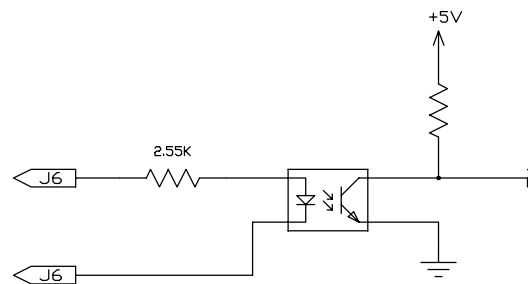
The INP\_Mode input is used to control the function of the In\_Position output. When this input is turned off, the In\_Position output will be turned on when the position following error (the TF command) is less than or equal to the value set by the Set Error (SE) command. When the INP\_Mode input is turned on, the In\_Position output will then be turned on when the output current value exceeds the value set by system parameter Current Threshold (see PI and DP commands).

The Output\_Limit input is used to limit the maximum output duty cycle of the SD100 to one of two levels:

Output_Limit	
Off	Use Output Limit 0
On	Use Output Limit 1

The Parameter Index (PI) and Define Parameter (DP) commands are used together to set the values for these different levels (see PI and DP commands). The default values for these settings are 100% duty cycle and 75% duty cycle.

**The logic state on any of these inputs should be maintained for at least 200 us before changing to guarantee that the input state is recognized.**



**Figure 2. Other Inputs**

### 1.2.3 Dedicated Digital Outputs

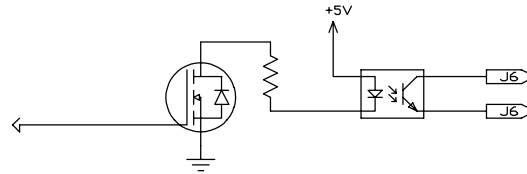
The dedicated outputs include Over-Temperature, Overflow, In\_Position and Ready. Figure 3 shows a circuit example of these outputs.

The Over-Temperature output will turn on when the temperature of the power driver of the SD100 exceeds 140° C. In this case the output of the SD100 will go inactive and will remain that way until the temperature of the driver falls below the 140° C threshold.

The Overflow signal will turn on when the deviation between the commanded input pulse count and the servo motor encoder pulse count exceeds  $\pm 2047$  counts. When this occurs the output of the SD100 will be disabled.

The In\_Position output will be turned on when the deviation between the commanded input pulse count and the servo motor encoder pulse count is less than the value programmed by the Set Error (SE) command. Otherwise this output will be turned off.

The Ready output is turned on when the SD100 is capable of normal operation. In the event of an Over-Temperature or Overflow condition, the Ready output will be turned off. Turning the Reset input on will also cause the Ready output to turn off.



**Figure 3. Dedicated Output**

### 1.3 Incremental Encoder Interface

The SD100 has a quadrature type encoder interface and the ability to supply +5 VDC at a minimum of 50 mA. The phase A+, phase B+, and Index+ inputs are pulled up to +5 VDC with 2.7K resistors. The phase A-, phase B- and Index- inputs are biased at +2.5 VDC with 2.7K resistors. This arrangement will accommodate both open collector and totem pole single-ended output encoders or line driver output encoders. The phasing of the channels (as interpreted by the SD100) can be changed via program command.

The encoder signals are received by a 2632 type line receiver. These signals are then passed through the SD100 to a 2631 type line driver for optional use by the hosting motion controller. The phasing command will have no effect on these signals.

### 1.4 Output Driver Interface

The SD100 onboard output driver is a PWM switching amplifier capable of supplying 3 Amps continuous and 6 Amps peak (for 200 mS minimum) at a switching frequency of approximately 19.531 KHz. This driver is intended for driving DC brush type motors or voice coil actuators. The driver uses the main power supply input and the peak voltage output to the motor will be nearly that of what is supplied. The output phasing of the driver can be changed via the Phase (PH) command.

The output driver includes an over-temperature sensor. If this sensor determines that the driver temperature is greater than 140° C, the driver will then be disabled, the Over-Temperature output will be turned on and the Ready output will be turned off.

### 1.5 RS-232 Serial Interface

The SD100 communicates with a host computer or a "dumb" terminal via an RS-232 serial interface. The baud rate is fixed at 9600 baud. Characters are fixed at 8 bits in length with 1 stop bit and no parity. Software XON / XOFF handshaking is supported. Hardware handshaking is not supported.

## 2. Modes of Operation

### 2.1 Position Mode

In normal operation, the SD100 will accept command pulses from the host motion controller. For each command pulse, the commanded position of the servo motor is either incremented or decremented. The SD100 does no trajectory planning as the frequency and rate of change of command pulses will ultimately determine the velocity profile. Unlike a normal open-loop stepping motor, if command pulses are given at a rate faster than the servomotor can respond, the SD100 will not lose count. The motor will simply lag behind until it is able to catch up.

### 2.2 Torque Mode

The SD100 is capable of operating in a constant current output mode known as Torque Mode. When Torque mode is entered the SD100 will no longer operate in a position holding mode. Instead it will operate in a constant current output mode with the commanded current being set by the command pulses from the motion controller.

The maximum usable range of commanded current is from 0 to  $\pm 1023$  counts representing from 0 to  $\pm 5$  Amps of output current (although the count is tracked to a command of up to  $\pm 32767$  counts to prevent overflows). As with any current source, the actual achievable output current will depend upon the load impedance and power supply voltage.

### 2.3 Switching Modes

When the PM/QM input is turned on, the SD100 will switch from Position mode (PM) to Torque mode (QM) of operation. When this occurs, the SD100 assumes an initial command of zero. Also, because the constant current loop error integrator is unknown it is set to zero.

This can present a problem in some applications in the form of a time lag in the output while the command is stepped to its required level and the error integrator ramps up sufficiently to create the necessary output. In order to reduce this output lag time in applications that require it, two system parameters are available to the user: "Initial Integral Term" (PI4) and "Initial Command" (PI5).

When the SD100 switches to QM, instead of assuming an initial output of zero, the command counter is loaded with the "Initial Command" (Init\_C) parameter. This allows the user to preset an output when switching modes. In a like manner, the error integrator is loaded with the "Initial Integrator Term" (Init\_I) parameter. This value is previously arrived at through empirical testing.

The InitC\_Mode input allows a further degree of flexibility by substituting the measured value of the actual output current for the Init\_C parameter when the unit enters QM. This allows the unit to automatically keep supplying the same output force when switching from PM to QM.

Because the command counter is used differently in Torque mode, when switching back to position mode by turning off the PM/QM input, the SD100 will have lost its command reference. Therefore, the command counter is loaded with the current value of the encoder counter and all PID loop

related variables are reset (i.e. following error, derivative term, etc.). The motor / actuator will need to be homed after operating in Torque mode.

## 2.4 Output Limiting

By using the Output\_Limit input, the maximum output duty cycle of the SD100 can be set to one of two different levels. The value for the two levels is set by using the Parameter Index (PI) and Define Parameter (DP) commands. The proper method for doing this is to first use the PI command to set the parameter index pointed to the desired system parameter, then to use the DP command to set the new value. For example:

```
PI0,DP32767      ; OL Input = 0 - max DC = 100%  
PI1,DP24576      ; OL Input = 1 - max DC = 75%
```

### 3. Entering Commands

Immediately after power-up, the SD100 is ready to accept commands. To verify this, you can hit the ESC key. If everything is working properly, this should cause a greater than sign (>) prompt to appear on your display. If not, you need to verify that the power and communications connections are correct and verify the compatibility of the communications protocol.

Commands are entered via a "dumb" terminal or host computer such as a PC compatible. Commands sent to the SD100 should consist of standard ASCII characters, and the command lines should be followed by a carriage return. Linefeeds are not necessary since they are used for formatting and therefore they are ignored. As characters are entered at the keyboard, they should be echoed on your display. If your display echoes its own transmitted characters, you will want to issue the Echo Off (EF) command; otherwise, the Echo On (EN) command (which is the default mode) should be issued. If you enter an invalid command, the SD100 will respond with a question mark "?" followed by a code indicating the type of error. These codes are listed on page 18 in the appendix.

If you make a mistake when entering a command, you can backspace to correct the error. If you are entering commands and change your mind, hitting the ESC key will cancel the line and give a new ">" prompt.

Once a command line has been entered and has finished executing, **hitting the RETURN key will cause the same command line to be re-executed.**

Command instructions are intended for use with the following syntax:

Command[Argument]<CR>

or...

Command[Argument],Command[Argument],...etc.

The numerical range of an argument will vary depending on the command with which it is used. The mathematical interpretation of the argument will depend on whether the Decimal Mode (DM) or Hexadecimal Mode (HM) was the last issued (DM is the power on default). Both decimal and hexadecimal numbers less than zero should be entered with a preceding minus "-" sign. If no argument is given, then it will be assumed as "0". It should be noted that commands can be strung together by using commas, up to a maximum line length of 127 characters.

If a command line contains the semicolon character ";", i.e...

```
>SG1000,SD5000 ; Set filter gains.<CR>
```

then the ";" and anything following it to the end of the line will be ignored. This feature is not particularly useful if you are entering commands manually, as comments are not retained by the SD100. However, if commands are downloaded to the SD100 from a host computer, the ability for line comments can make setup documentation possible and desirable.

### 3.1 Downloading Commands

In many cases (such as programming multiple units with the same values), it is more convenient to enter commands using a text editor on a host computer and then download that text file to the SD100 using a communications program such as ProComm® or the Microsoft® Windows™ Terminal program. Whatever communications software is used, it must have the ability to provide a short delay (approx. 100 mS) after transmitting each line to give the SD100 time to interpret and store the commands that were just sent.

### 3.2 Commands

---

**Command:**    **DM**    **-- Decimal Mode --**

Default:        Decimal Mode

This command causes all numerical input and output to be interpreted as decimal or base 10. Numbers will be output with a leading "-" if they are less than zero.

Related Commands: HM

---

**Command:**    **DP**    **-- Define Parameter --**

Argument:	Output Limit 0	(PI0) 0 <= n <= 32767
	Output Limit 1	(PI1) 0 <= n <= 32767
		(PI2) Reserved
		(PI3) Reserved
	Initial Integral Term	(PI4) 0 <= n <= 32767
	Initial Command	(PI5) -1023 <= n <= 1023
	Current Threshold	(PI6) 0 <= n <= 1023
	Current Gain	(PI7) 0 <= n <= 32767

Default:

- (PI0) 32767
- (PI1) 24576
- (PI2) n/a
- (PI3) n/a
- (PI4) 0
- (PI5) 0
- (PI6) 0
- (PI7) 0

This command allows the user to adjust a number of operating parameters of the SD100.

The "Output Limit" parameters allow the user to specify two output duty cycle limits of which one of these at a time is used as determined by the Output\_Limit input.

The "Initial Integrator Term" and "Initial Command" parameters are used as pre-loads to the current loop error integrator when switching to Torque Mode (QM).

The "Current Gain Threshold" parameter is used a comparator value for the In\_Position signal when the INP\_Mode input is turned on. When the output current (as reported by the TC command) is at a value equal to or greater than this parameter, the In\_Position output will turn on.

The "Current Gain" parameter sets the current loop error gain used when operating in Torque Mode. This allows the response of the current error integrator to be adjusted to suit different applications. A low value will provide slow response to load changes while a high value will provide quick response. If the value is set too low then inadequate control may occur. If the value is too high then the system may be unstable. A good "general" value is about 5000 to 8000.

Related Commands: PI

---

**Command: FRn -- Set Derivative Sampling Period --**

Argument:  $0 \leq n \leq 127$   
 Default: 0

This command allows for the adjustment of the derivative sampling interval for the servo. The period of this interval can be calculated by the following:

$$T = (n+1) * 0.000100$$

where "T" is the period in seconds and "n" is the FR command argument. For example, if the value set by the FR command were 9, then the derivative sample period would be:

$$(9+1) * 0.000100 = .001000 \text{ S or } 1 \text{ mS}$$

This command is useful in tuning the PID servo loop to the inertial properties of the system.

Related Commands: RI

---

**Command: HM -- Hexadecimal Mode --**

Default: Decimal Mode

This command causes all numerical input and output to be interpreted as hexadecimal or base 16. Numbers will be output as 2, 4 or 8 digits with leading 0's if they are positive and leading F's if they are negative.

Related Commands: DM

---

**Command: ILn -- Set Integration Limit --**

Argument:  $0 \leq n \leq 16,383$   
 Default: 0

This command clamps the level of influence that the PID digital filter integral term can use to reduce static position error thereby reducing integral "wind up". When properly adjusted, this can enhance loop stability and operation. The Integral Limit (IL) and Set Integral Gain (SI) must both be set to a non-zero value in order for the integral term to have any effect.

Related Commands: RI, SI

**Command: PL -- Parameter Load --**

This command causes all set parameters to be restored from non-volatile memory.

Related Commands: PS

**Command: PHn -- Set Servo Phasing --**

Argument:  $0 \leq n \leq 15$

Default: 0

This command is used to set the output polarity, the encoder phasing and the command counter mode and phasing. The polarity of the output will determine whether the servo is driven in a direction that reduces or increases position error. The encoder phase will determine whether the position count will increase or decrease for a valid encoder input sequence. The command counter mode will determine Count Up / Count Down or Step / Direction. The command counter phasing will determine the count direction of the command counter.

To determine the argument to be used with the PH command, use the follow table and add the required values together.

	Add to 'n'
Output Phase Normal	0
Output Phase Reversed	1
Encoder Phase Normal	0
Encoder Phase Reversed	2
Command Mode Step / Direction	0
Command Mode Count Up / Count Down	4
Command Phase Normal	0
Command Phase Reversed	8

The default phasing and sense is equivalent to issuing this command with a argument of 0.

**Command: PI -- Parameter Index --**

Argument:  $0 \leq n \leq 7$

Default: 0

This command sets the index pointer used by the Define Parameter (DP) command. Whichever of eight special parameters this command is used to point to is the one the DP command will operate on.

Related Commands: DP

**Command: PS -- Parameter Store --**

This command causes all set parameters to be saved in non-volatile memory.

Related Commands: PL

**Command:**    **RI n**    -- Sampling Rate of Integral --

Argument:    0 <= n <= 127  
Default:      0

This command allows for the adjustment of the PID digital filter integral sampling interval for the servo. The period of this interval can be calculated by the following:

$$T = (n+1) * 0.000100$$

where "T" is the period in seconds and "n" is the RI command argument. For example, if the value set by the RI command is 1, then the integral sample period will be:

$$(1+1) * 0.000100 = .000200 \text{ S or } 200 \text{ } \mu\text{S}$$

This command is useful in tuning the PID servo loop to the inertial properties of the system.

Related Commands: FR

---

**Command:**    **SD n**    -- Set Derivative Gain --

Argument:    0 <= n <= 32,767  
Default:      0

This command sets the derivative gain term of the PID digital filter loop for the servo.

Related Commands: SG, SI, IL

---

**Command:**    **SE n**    -- Set Maximum Following Error --

Argument:    0 <= n <= 16,383  
Default:      16,383

This command is used to set the maximum deviation of the encoder position from the commanded position for the servo that will allow activation of the In\_Position output signal. If the deviation exceeds the programmed value, the In\_Position output will be turned off.

Related Commands: TF

---

**Command:**    **SG n**    -- Set Proportional Gain --

Argument:    0 <= n <= 32,767  
Default:      0

This command sets the proportional gain term of the PID digital filter loop for the servo.

Related Commands: SI, SD, IL

**Command:**    **SIn**    -- Set Integral Gain --

Argument:    0 <= n <= 32,767

Default:     0

This command sets the integral gain term of the PID digital filter loop for the servo. The Set Integral Gain (SI) and Integral Limit (IL) must both be set to a non-zero value in order for the integral term to have any effect.

Related Commands: SG, SD, IL

---

**Command:**    **TC**    -- Tell Current Output --

This command reports the output current that is flowing through the load. The value will be from 0 to 1023 representing a current of from 0 to 5 amps.

---

**Command:**    **TD**    -- Tell Derivative Gain --

This command reports the derivative gain value of the PID digital filter for the servo.

Related Commands: TG, TI, TL

---

**Command:**    **TF**    -- Tell Following Error --

This command reports the following error for the servo. This value is the difference between the current command count position (or that which is reported by the Tell Optimal (TO) command) and the servo's current real position (or that which is reported by the Tell Position (TP) command).

Related Commands: SE

---

**Command:**    **TG**    -- Tell Proportional Gain --

This command reports the proportional gain value of the PID digital filter for the servo.

Related Commands: TI, TD, TL

---

**Command:**    **TI**    -- Tell Integral Gain --

This command reports the integral gain value of the PID digital filter for the servo.

Related Commands: TG, TD, IL

**Command:** TK -- Tell (K) Constants --

**Argument:** none

This command will display a number of internal settings.

Example display for the command "TK":

```
>TK

Parameter Values...

Proportional Gain ----- (SG) = 300
Integral Gain ----- (SI) = 200
Derivative Gain ----- (SD) = 5000
Integral Limit ----- (IL) = 10000
Maximum Following Error ---- (SE) = 2
Integral Sample Rate ----- (RI) = 0
Derivative Sample Rate ----- (FR) = 2
Phase and Sense Settings --- (PH) = 0
Output Limit 0 ----- (PI0,DP) = 32767
Output Limit 1 ----- (PI1,DP) = 24576
Output Limit 2 ----- (PI2,DP) = 16384
Output Limit 3 ----- (PI3,DP) = 8192
Initial Integral Term -- (PI4,DP) = 0
Initial Command ----- (PI5,DP) = 0
Current Threshold ----- (PI6,DP) = 0
Current Gain ----- (PI7,DP) = 5000
Firmware Revision ----- (VE) = 1.03 Mod516
>
```

Related Commands: VE

**Command:** TL -- Tell Integration Limit --

This command reports the integration limit value of the PID digital filter for the servo.

Related Commands: TG, TI, TD

**Command: TS -- Tell Status --**

This command reports the value of the status word variable whose bit are defined by the following table:

Bit	Name	Purpose
0	Reserved	Reserved
1	OLmt0_B	Output_Limit input 0 state, 1 = "on"
2	Mode2	InitC_Mode input state, 1 = "on"
3	Mode0	PM/QM mode input state, 1 = "on"
4	Mode1	INP_Mode input state, 1 = "on"
5	OMode0	Old INP_Mode input state, 1 = "on"
6	Reserved	Reserved
7	Reserved	Reserved
8	Enabled	Output enabled state, 1 = "enabled"
9	NewReset	Reset input state, 1 = "on"
10	OldReset	Old reset input state, 1 = "on"
11	NewClr	Clear input state, 1 = "on"
12	OldClr	Old clear input state, 1 = "on"
13	Reserved	Reserved
14	HexMod	1 = Hex output mode
15	Blink	LED blink state

**Command: VE -- Tell Version --**

This command reports the firmware revision level.

Related Commands: TK

## 4. Appendix A, SD100 Error Code Definitions

### 1 - ARGUMENT ERROR.

This error indicates that a command argument was not given or was specified out of the permissible numerical range.

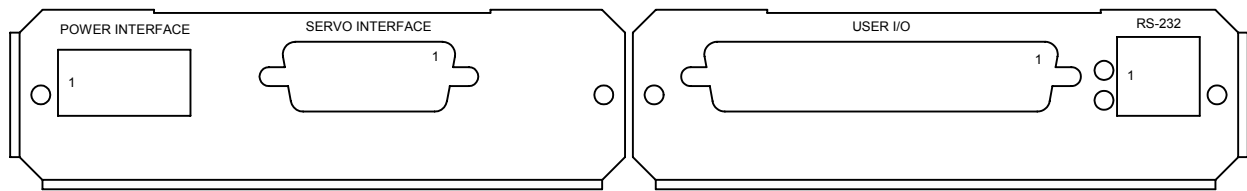
### 2 - INVALID COMMAND.

This error indicates that an invalid or unrecognized command was specified in the command line.

## 5. Appendix B, Summary of SD100 Commands

DM -- Decimal Mode  
DP -- Define Parameter  
FR -- Derivative Sampling Frequency  
HM -- Hexadecimal Mode  
IL -- Integration Limit  
PH -- Phase  
PI -- Parameter Index  
PL -- Parameter Load  
PS -- Parameter Store  
RI -- Sampling Rate of Integral  
RT -- Reset  
SD -- Set Derivative Gain  
SE -- Set Error  
SG -- Set Proportional Gain  
SI -- Set Integral Gain  
TC -- Tell Current  
TD -- Tell Derivative Gain  
TF -- Tell Following Error  
TG -- Tell Position Gain  
TI -- Tell Integral Gain  
TK -- Tell (K)Constants  
TL -- Tell Integration Limit  
TO -- Tell Optimal Position  
TP -- Tell Real Position  
TQ -- Tell Torque  
TS -- Tell Status  
VE -- Tell Version  
ZZ -- Dump Memory

## 6. Appendix C, SD100-R2 Connector Pin Definitions



**J1 - Power Interface :**  
**4-Pin 5.08mm Centers Terminal Block**  
**Mating Connector: OnShore# EDZ95004**  
**Digi-Key# ED1719**

1. Main power return
2. Main V+ power input
3. Motor+ output
4. Motor- output

**J2 - Servo Interface : 15-Pin Female D-Sub**  
**Mating Connector: Digi-Key# 215M**

1. Encoder phase A+
2. Encoder phase B+
3. Encoder Index+
4. +5 VDC
5. +5 VDC
6. Reserved, do not use
7. Reserved, do not use
8. Reserved, do not use
9. Encoder phase A-
10. Encoder phase B-
11. Encoder Index-
12. Common
13. Common
14. Reserved, do not use
15. Reserved, do not use

**J5 - RS-232 Comm. Interface:**  
**6-Pin Modular Jack**  
**Mating Connector: Digi-Key# A9093**

1. Handshake output
2. Handshake input
3. Receive data input
4. Transmit data output
5. Common
6. +5 VDC

**J6 - User I/O Interface : 37-Pin Female D-Sub**  
**Mating Connector: Digi-Key# 237M**

1. Over-temp. output (+)
2. Overflow output (+)
3. In\_Position output (+)
4. Ready output (+)
5. Reset input return (-)
6. Clear input return (-)
7. Count\_Up / Step input (+)
8. Count\_Down / Dir input return (-)
9. Reserved, do not use
10. Reserved, do not use
11. Reserved, do not use
12. Encoder Index- output
13. Encoder phase B+ output
14. Encoder phase A+ output
15. Common
16. INP\_Mode input (+)
17. PM/QM Mode input (+)
18. Output\_Limit input (+)
19. InitC\_Mode input (+)
20. Over-temp. output return (-)
21. Overflow output return (-)
22. In\_Position output return (-)
23. Ready output return (-)
24. Reset input (+)
25. Clear input (+)
26. Count\_Up / Step input return (-)
27. Count\_Down / Dir input (+)
28. Reserved, do not use
29. Reserved, do not use
30. Encoder Index+ output
31. Encoder phase B- output
32. Encoder phase A- output
33. +5V
34. INP\_Mode input return (-)
35. PM/QM Mode input return (-)
36. Output\_Limit input return (-)
37. InitC\_Mode input return (-)