

# 6 Axis Driver Pack DPX01E16

## User's Guide



**ANAHEIM AUTOMATION**

910 East Orangefair Lane, Anaheim, CA 92801  
e-mail: [info@anaheimautomation.com](mailto:info@anaheimautomation.com)

(714) 992-6990 fax: (714) 992-0471  
website: [www.anaheimautomation.com](http://www.anaheimautomation.com)

# Communications

## Baud Rates

A term used frequently in serial data communications. A “baud” is defined as the reciprocal of the shortest pulse duration in a data word signal, including start, stop, and parity bits. This is often taken to mean the same as “bits per second”, a term that expresses only the number of “data” bits per second. Very often, the parity bit is included as an information or data bit. This unit accepts a baud rate of 115200 only and protocol settings of 1 start bit, 8 data bits, no parity and 1 stop bit.

## RS232 to RS485 Protocol

The controller can be connected to your PC serial port via an RS485 converter box. The RS232 converter will convert the RS232 communication format to the RS485 format. Only one converter box is needed per serial port.

## RS485 Protocol

The RS485 protocol is as follows, on board receivers will remain in active mode indefinitely. Transmitters must be turned off when the unit is not sending data to prevent the line from sending and receiving data at the same time. Therefore when the PC is transmitting data its driver will be turned on and each of the units connected will have their drivers off. If they are requested to send data back to the PC, the selected unit will turn it's driver on to send the data then turn it off after it has completed transmission. Note: The above protocol is done internally between the converter and the controller. The RS485 method of communication allows increased noise immunity and increased communication distance of up to 4000 feet without repeaters. RS485 repeaters allow an additional 4000 feet per repeater.

## Two Wire Configuration

The two wire configuration reduces cabling costs by requiring only three wires. A (-), B (+) and ground. The DPX01E16 is internally configured for the two wire configuration. RS485 devices may be internally or externally configured for two wire systems. Internally configured RS485 devices simply provide “A” and “B” connections (sometimes labeled “-” and “+”). To use the 2 wire configuration with a four wire converter, simply jumper wire the TX+ to RX+ to make the B(+) terminal, and the TX- to RX- to make the A(-) terminal on your converter box. Then run a wire from ground to ground, a wire from B(+) to B(+) and a wire from A(-) to A(-) to the first controller in the network. If multiple controllers are connected on the same network, simply wire from the first controller to the second to the third and so on keeping ground, A(-) and B(+) connected throughout the entire network. The tristate capabilities of RS485 allow a single pair of wires to share transmit and receive signals for half duplex communications. This “two wire” configuration (note that an additional ground conductor should be used) reduces cabling cost.

## Terminating Resistor

To eliminate noise on the transmission lines a terminating resistor may need to be used. If needed the termination resistor need only be added to the last (furthest from the converter box) controller in the network. A termination resistor with a value of 120 ohms is needed when using a 4000 ft. or longer cable. If needed this resistor needs to be placed across the + and - terminals of the RS485 connector (Pins 1 and 2 of P8).

## Electrical Specifications

### Power Requirements:

90-265 VAC Single Phase

### Operating Temperature:

0° to 60° C

### Pulse Output Range:

1 to 10,000 steps/sec

### Limit Inputs (TTL-CMOS):

Logic "0": 0 to 0.8VDC (1mA)

Logic "1": 3.5 to 5.0VDC (0mA)

Inputs are internally pulled high

### Baud Rate:

115200 BAUD

**Note:** For inductive loads a clamping diode must be used to protect from fly back voltage spikes.

### Driver Ratings:

Output Current 0.15 Amps peak

Item	Units	
Phase Output Current	0.11	A (RMS)
Phase Output Current	0.15	A (Peak)

### Data Format:

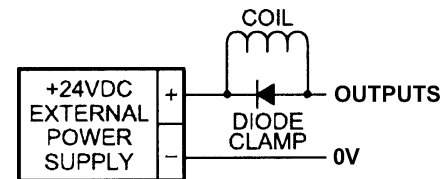
1 start bit, 8 data bits,  
no parity, 1 stop bit

### Outputs (8 programmable):

Open Drain Type

24V, 500mA

Vce(sat) = 300mV @ 100mA, 50mV @ 400mA



## Terminal Descriptions

### P1-P6: Motor Connections

Pin #	Description	Wire Color
1	Phase A	Orange
2	Phase A\	Black
3	Phase B	Red
4	Phase B\	Yellow

### P8: RS485

Pin #	Description	Wire Color
1	TD(B) (+)	Brown
2	TD(A) (-)	Brown/White
3	Ground	Black

### P7: Limit Inputs

Pin #	Description	Wire Color
1	Ground	Black
2	Axis 1 +	Brown
3	Axis 1 -	Red
4	Axis 2 +	Orange
5	Axis 2 -	Yellow
6	Axis 3	Green
7	Axis 4	Blue
8	Axis 5	Violet
9	Axis 6	Grey
10	Ground	Black

### P9: Outputs

Pin #	Description	Wire Color
1	Output 1	Brown
2	Output 2	Red
3	Output 3	Orange
4	Output 4	Yellow
5	Output 5	Green
6	Output 6	Blue
7	Output 7	Violet
8	Output 8	Grey

## Axis Selection

Each controller will be addressed using a two position dip switch allowing the PC to address up to 4 controllers from one port. The switch is located on top of the unit.

Axis Selected	SW1 Setting	SW2 Setting
0	OFF	OFF
1	ON	OFF
2	OFF	ON
3	ON	ON

## Reducing Output Current

Reducing the output current is accomplished automatically and occurs approximately 20mSec after a move is finished. The amount of current per phase in the reduction mode is approximately 70% of the set current. When the current reduction circuit is activated, the current reduction resistor is paralleled with the current adjustment potentiometer. This lowers the total resistance value, and thus lowers the per phase output current.

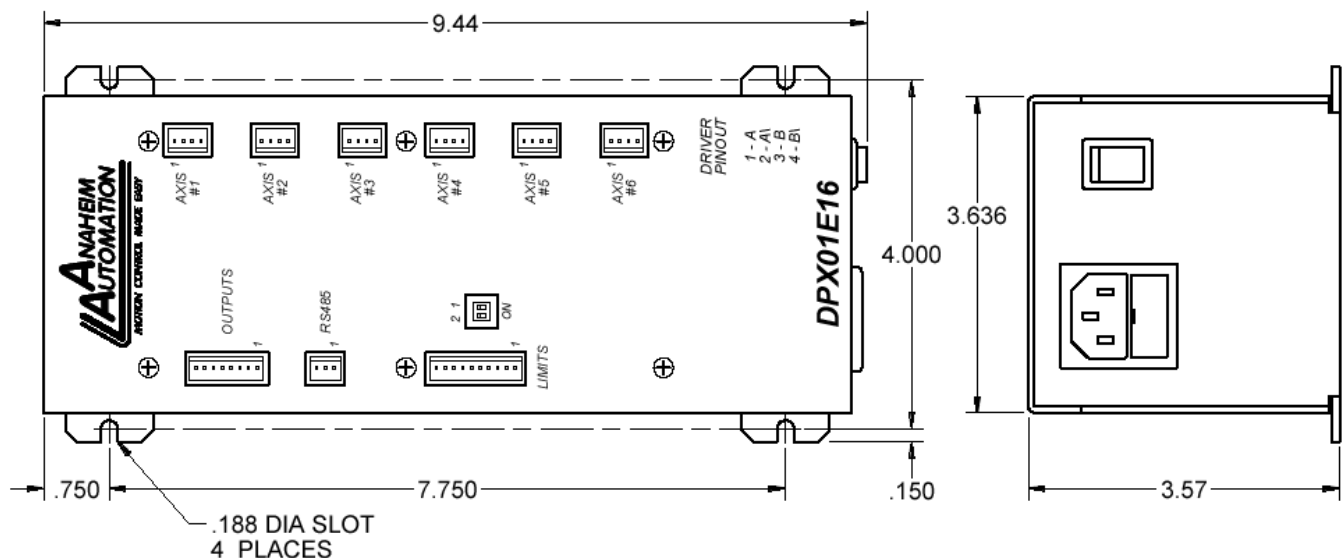
## Connecting the Step Motor

Phase 1 and Phase 3 of the step motor are connected to pins 1 and 2 on connectors P1 - P6. Phase 2 and Phase 4 of the step motor are connected to pins 3 and 4 on connectors P1 - P6.

**NOTE:** The physical direction of the motor with respect to the direction input will depend on the connection of the motor windings. To reverse the direction of the motor with respect to the direction input, switch the wires on Phase 1 and Phase 3.

**WARNING:** Do not connect or disconnect motor wires while power is applied!

## Dimensions



## COM Port Settings

**Baud Rate:** 115200  
**Data Bits:** 8  
**Parity:** None  
**Stop Bits:** 1  
**Flow Control:** Xon/Xoff

### Unit Selection

In order to select a unit the @ command followed by the address of the unit must be sent.  
NOTE: There should be no spaces between the @ and address select.

### How to select a unit:

@0 (Unit 0 is selected)  
@1 (Unit 1 is selected)  
@3 (Unit 3 is selected)

### How to get a response from a unit:

@0\$ (Carriage Return)

After the \$ command, the controller will return an ESS06 + the current version number.

Note: In direct talk mode each command is followed by a carriage return.

The unit communicates in half duplex mode, therefore proper setup of hyper terminal is necessary to view characters, if characters are to be echoed back to the screen.

## Instructions

All instructions require that no spaces be sent between the command and the parameter followed by a carriage return. After all instructions that set a register, the registers contents will be returned to the pc to verify that the correct value was received. A # indicates that a value between 1 - 6 needs to be entered to set the proper axis parameters. The # is the number of the axis.

(@0 not @ 0)

### Command Summary:

A - Acceleration/Deceleration	M - Max Speed
B - Base Speed	O - Set Outputs
D - Microstep Divisor	S - Stop Motion
E - Enable Motor	V - Verify
F - Poll if Busy	+ - Clockwise Direction
G - Go	- - Counterclockwise Direction
I - Set Index Distance	\$ - Version Number Register
L - Get Limit Register Status	! - Error Codes Register

## A - Acceleration/Deceleration

Format: A#\_ [value] ex:@0A1\_10000

Description: This command sets the acceleration profile based on the following function  
 $Speed\ 2 = Accel/Speed\ 1 + Speed\ 1$   
Each axis has its own accel value and needs to be entered in separately for each one. This value must be set before motion begins.

Range: 100 - 999,999 Default:1000

## B - Base Speed

Format: B#\_ [value] ex:@0B4\_100

Description: This command sets the base (start) speed for motion. This value must be set before motion begins and be less then the maximum speed. Each axis has its own base speed value and needs to be entered in separately for each one.

Range: 1 - 5000 Default: 1

## D - Microstep Divisor

Format: D[value] ex:@0D4

Description: This command sets the microstepping divisors for all axes. A 1 will set the drivers in full step, a 2 will set the drivers in half step, a 4 will set the drivers in quarter step, and an 8 will set the driver is eighth step. There is only one D value for all axes.

Range: 1, 2, 4 or 8 Default: 8

## E- Enable Motor

Format: E#\_[value] ex:@0E2\_1

Description: This command will enable or disable the given axis. A 1 will enable the driver and a 0 will disable the driver. Each axis has its own enable bit and must be entered separately for each one.

Range: 0 or 1 Default: 0

## F - Poll if Busy

Format: F ex:@0F

Description: This command will send back a 0 if a motor is not running, or a 1 if a motor is running

## **G - Go**

Format: G# ex:@0G5

Description: This command tells the given axis to move the number of steps set in the index register for that axis. All speed profile information must be set for the given axis before this command is used.

## **I - Index Distance**

Format: I#[value] ex:@0I6\_4000

Description: This command sets the number of steps for the motor to move following a G command. Each axis has its own index value and needs to be entered in separately for each one. This value must be set before motion begins.

Range: 0 - 65,535 Default: 0

## **L - Limit Register Status**

Format: L ex:@0L

Description: This command will send back the 8 bit binary value of the limit register. With no limits activated a value of 255 will be sent back, with all limits activated a value of 0 will be sent back. Limit 1+ is the LSB and Limit 6 is the MSB

## **M - Max Speed**

Format: M#\_ [value] ex:@0M1\_1500

Description: This command sets the max (running) speed for motion. This value must be set before motion begins and be greater than the base speed. Each axis has its own max speed value and needs to be entered in separately for each one.

Range: 1 - 10,000 Default: 1

## **O - Set Outputs**

Format: O[value] ex:@032

Description: This command sets the output register according to the binary value. To turn on Output 8 only send a value of 182, to turn on Output 1 and Output 2 send a value of 3. Output 1 is the LSB and output 8 is the MSB.

Range: 0 - 255 Default: 0

## **S - Stop Motion**

Format: S ex:@0S

Description: This command will stop the motor while it is running. This command is valid only when there is motion of one of the axes.

### + - Direction Clockwise

Format: + ex: @0+

Description: This command sets the direction for all of the axes to clockwise. A value of 1 will be sent back to the PC after this command is received. This is the default value for direction

### -- Direction Counter-Clockwise

Format: - ex: @0-

Description: This command sets the direction for all of the axes to counterclockwise. A value of 0 will be sent back to the PC after this command is received.

### \$ - Version Number Register

Format: \$ ex: @0\$

Description: This command requests the controller to return the version number of the internal firmware.

### ! - Error Codes Register

Format: ! ex: @0!

Description: This command requests the controller to get the current error code and print it to the screen. Refer to the troubleshooting section for definitions of error codes.

### V - Verify

Format: V [command] ex: @0VA1

Description: This command can be used with most commands to verify the register contents. The verify command causes the controller to send data back to the PC. The data is sent as an ASCII decimal string followed by a carriage return and a line feed. The permissible verify commands are shown below.

Command	Description
A#	Verify acceleration/deceleration
B#	Verify base speed
D	Verify microstep divisor
E#	Verify axis enabled
I#	Verify index distance
M#	Verify max speed
O	Verify output register
+	Verify direction (1 is CW, 0 is CCW)



## Troubleshooting

### Problem:

*Can not establish communications with the controller.*

### Possible Solutions:

- 1) Make sure the controller has power.
- 2) Check RS485 connections.
- 3) Check for loose cable connection either on the controller or COM Port.
- 4) Verify that the address is correct
- 9) If problems still exist, contact Anaheim Automation Tech Support...

### Problem:

*There is no power to the controller.*

### Possible Solutions:

- 1) Is the driver pack connected to the appropriate power supply?
- 2) Check the driver pack's fuse.
- 3) If problems still exist, contact Anaheim Automation Tech Support.

<b>Anaheim Automation Tech Support:</b>	<b>910 East Orangefair Lane Anaheim, CA, 92801-1195 phone: (714) 992-6990 fax: (714) 992-0471 www.anaheimautomation.com</b>
---	---

## Error Codes

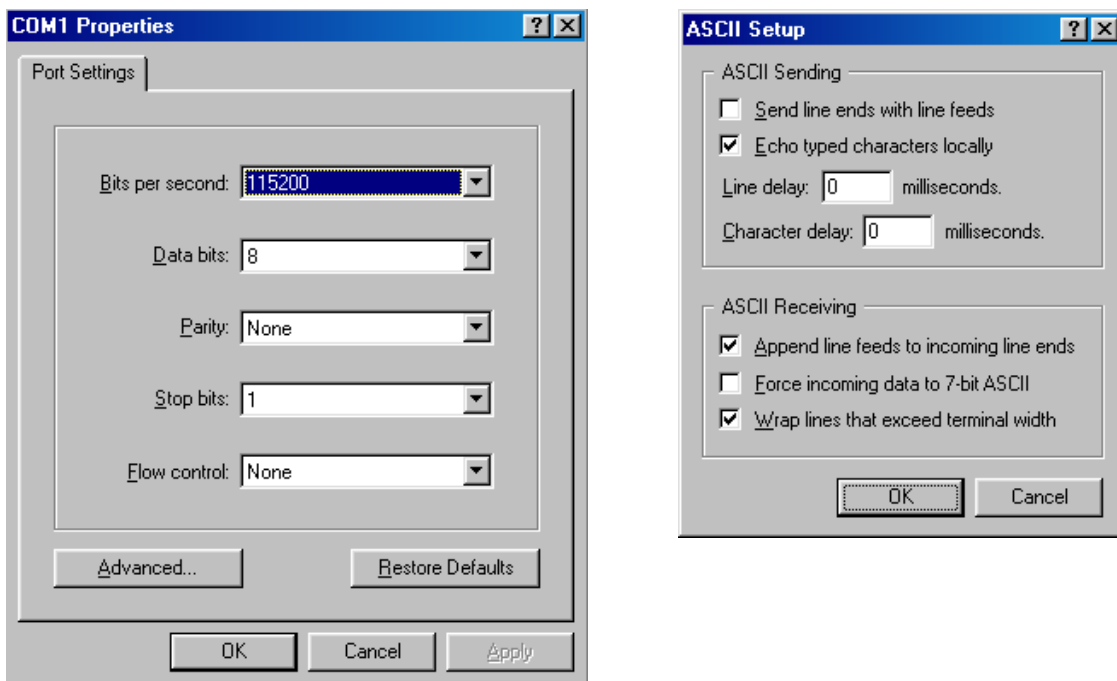
Error Code	Type	Description
1	Recieve Overflow Error	The serial communications had a receiving error. This is an internal error caused by the computer.
2	Transmitt Error	To many parameters sent back to the PC. This is an internal error caused by the eeprom.
4	Command Error	A bad command was sent to the controller. Please check to see that the command being sent is valid.
8	Zero Parameters Error	There were no parameters sent to the controller. A command was sent to the controller that expected to see parameters after the command.
16	Range Error	There was an invalid number of commands and characters sent to the controller. Check to see if the parameters are valid for the command.

## Appendix 1: Hyper Terminal Setup

To use Hyper Terminal with the DPX01E16, proper setup is required. Follow the procedures listed below for proper communication. Most versions of the Windows operating system includes Hyper Terminal.

- 1) Open HYPERTERM.EXE on the PC
- 2) Choose a name and icon to identify the file. The file will have the .ht extension when saved
- 3) Then select the correct com port to communicate with in the Connect To tab
- 4) The next screen is the Configure command button in the Connect To tab
  - a) Bits per second = 115200
  - b) Data bits = 8
  - c) Parity = None
  - d) Stop bits = 1
  - e) Flow control = None
- 5) In the File|Properties menu click on the Settings Tab and select the ASCII Setup button
  - a) Uncheck Send line ends with line feeds
  - b) Check Echo typed characters locally
  - c) Set both Line delay and Character delay to 0 milliseconds
  - d) Check Append line feeds to incoming line ends
  - e) Uncheck Force incoming data to 7-bit ASCII
  - f) Check Wrap lines that exceed terminal width
- 6) Save file so the parameters are set the next time you open the file. The file will have the .ht extension when saved
- 7) Set the DPX01E16 address to 0
- 8) Power on the DPX01E16 and connect the serial cable to the correct com port
- 9) Type @0\$ [enter] and the DPX01E16 should return the value ESS06 on one line and the version of the firmware on the next line
- 10) You now know Hyper Terminal is communicating with the unit. Make sure the Caps Lock key is activated before you begin sending more instructions to the unit

Note: The next time you want to open Hyper Terminal, click on the file name you created so the parameters will be saved. If you open HYPERTERM.EXE again, then you will have to setup the parameters again.



## Appendix 2: ASCII to Hex Conversion Table

ASCII Symbol	Hex Value	ASCII Symbol	Hex Value
0	30	F	46
1	31	G	47
2	32	I	49
3	33	L	4C
4	34	M	4D
5	35	O	4F
6	36	S	53
7	37	V	56
8	38	!	21
9	39	\$	24
A	41	+	2B
B	42	-	2D
D	44	@	40
E	45	_	5F
Carriage Return	0D		

### Copyright

Copyright 2004 by Anaheim Automation. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of Anaheim Automation, 910 East Orangefair Lane, CA, 92801. The only exception to this would be use of the program examples in this manual.

### Disclaimer

Though every effort has been made to supply complete and accurate information in this manual, the contents are subject to change without notice or obligation to inform the buyer. In no event will Anaheim Automation be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation.

### Limited Warranty

All Anaheim Automation products are warranted against defects in workmanship, materials and construction, when used under Normal Operating Conditions and when used in accordance with specifications. This warranty shall be in effect for a period of twelve months from the date of purchase or eighteen months from the date of manufacture, whichever comes first. Warranty provisions may be voided if the products are subjected to physical damage or abuse.

Anaheim Automation will repair or replace at its option, any of its products which have been found to be defective and are within the warranty period, provided that the item is shipped freight prepaid, with RMA (return material authorization), to Anaheim Automation's plant in Anaheim, California.

### Trademarks

Control Link and Driver Pack are registered trademarks of Anaheim Automation.  
Windows is a registered trademarks of Microsoft, Corp.

***ANAHEIM AUTOMATION***