

This is a Discontinued Product

Contact Kollmorgen Customer Support at
1-540-633-3545 or email us at support.kollmorgen.com
if assistance is required.

APPLICATION NOTE 804

PACIFIC SCIENTIFIC, 110 FORDHAM ROAD, WILMINGTON, MA 01887

Using the PC830 with an Exor Universal Operator Panel (UniOP) Interface Terminal (OIT)

Note: This application note only applies to PC830 drives with firmware version 2.00 and higher. It also only applies to Exor Designer for Windows software version 5.06.03 and higher.

The PC830 servo drive can be interfaced for RS-232/422/485 serial communications with an Exor UniOP Terminal. An Exor UniOP Terminal Model BKDC-16 and Designer for Windows version 5.06.03 software was used for the purpose of this application note.

Furthermore, Exor has installation reference manuals for their products which can be consulted for additional information. Additional support may also be obtained by contacting Exor Electronic R&D at (561) 753-2250 or at their website www.exor-rd.com.

Powering the UniOP Terminal

Pacific Scientific recommends that you purchase a ready made Exor Terminal power cable. Additional information concerning the power cable interface for the UniOp should be referenced in the UniOP Installation and Operation Guide product manual(s).

Communications Wiring for Configuring the Exor UniOP Terminal

Pacific Scientific recommends that you purchase a ready made PC to Exor Terminal cable from Exor Electronic R&D, Inc. To program a pane, connect the panel to a PC using a CA1 or CA2 programming cable and place the panel into Configuration Mode. Additional information concerning the communications cable interface can be referenced in the UniOP Installation and Operation Guide product manual(s).

Once the Exor UniOP Terminal has been powered and is wired up for communications, configure the Terminal using the Exor Designer for Windows software package.

Configuring the Exor UniOP Terminal using Designer for Windows

After the Designer software has been installed, refer to the Exor Designer for Windows manual pages 2-5 through 2-8 for additional information.

Procedure

1. Type Enable = ↵
2. Fill in the controller define field dialog box with the appropriate data. For example, Enable is PDID number 6. Therefore:

Address Offset = 6
Display Format = Numeric
Data Type = Integer Variable
Numeric Base = Decimal
Data Access = Read/Write
Field Width = 1
Field Height = 1
3. Click on **OK** to confirm the operation and create the field.
4. Select **Save** in the **File** menu to save the project to disk.
5. Power up the UniOP Terminal and put into Configuration Mode by holding down the ENTER key for a few seconds. **CONFIGURATION MODE** will appear on the Terminal.
6. Select **Transfers-Download** from the **Main** menu to send the project to the Terminal.
7. After a short pause, the text that was typed in step 1 should be visible along with the data value read from the controller. For example, "Enable = 1" or "Enable = 0" will appear depending upon if the motor is spinning or not.

Communications Wiring for the UniOP Terminal and the PC830

The proper communication cable must be chosen for the type of device being connected. The PLC port uses a D-sub 15 pin male connector with pin assignments as shown in the table below.

Pin	Description	Pin	Description
1	Frame Ground	9	TX- 20mA
2	RXD	10	RTS
3	TXD	11	CTS
4	+5V output (Max 100mA)	12	RX+ 20mA
5	GND	13	RX- 20mA
6	CHA-	14	CHA+
7	CHB-	15	CHB-
8	TX+ 20mA		

Different electrical standards are available for the signals in the PLC port connector: RS-232, RS-422, RS-485, or Current Loop 20mA. The cable used selects the appropriate signals. Contact Exor Electronic R&D, Inc. for a complete list of the available communication cables.

Demo Package for the UniOP Terminal and the PC830:

This example describes the steps necessary in order to set up the PC830/UniOP Terminal Demonstration package.

1. Put the UniOP Terminal into Configuration Mode by holding down the ENTER key for a few seconds. When the Terminal Command Menu appears, use the cursor keys to highlight the **Configuration Option** and then press ↵.
2. Connect the UniOP Terminal to the PC using a CA1 or CA2 cable.
3. Connect a ready made cable from the UniOP Terminal PLC port to J1 serial port of a PC830. (model no: PC833-001-N)
4. Configure the drive/motor combination of the PC830 by downloading the 830test.cfg configuration file. This file configures a PC833/PMA22B motor combination. If a different drive/motor combination is being used, modify the configuration as necessary.
5. The motor should now be ready to go.
6. An Enable/Disable command can now be made through the Exor UniOP Terminal. Press the UniOP Terminal keypad number 0 to highlight the digit read from the controller and press keypad 0 or 1 ↵ to Enable/Disable the PC830 brushless servo drive.

Keywords

The following table contains a listing of keywords for PC830/Exor Terminal setup. Please consult the 830Tools on-line help system for additional information.

Name	PDID	R/W	Data Format	Display Format	Data Type
AccelLmt	45	W	Real	Numeric	Float
ActiveAccelRate	120	R	Real	Numeric	Float
ActiveDecelRate	121	R	Real	Numeric	Float
ActiveDistance	149	R	Integer	Numeric	Integer
ActiveDistOffset	152	R	Integer	Numeric	Integer
ActiveDwell	151	R	Integer	Numeric	Integer
ActiveHomeDir	153	R	Integer	Numeric	Integer
ActiveHomeMode	154	R	Integer	Numeric	Integer
ActiveMove	155	W	Integer	Numeric	Integer
ActiveMoveType	148	R	Integer	Numeric	Integer
ActiveRegSelect	176	R	Integer	Numeric	Integer
ActiveRunSpeed	119	R	Real	Numeric	Float
ADF0	12	W	Real	Numeric	Float
ADOffset	13	W	Real	Numeric	Float
AIn1Map	87	W	Integer	Numeric	Integer
AIn2Map	88	W	Integer	Numeric	Integer
AIn3Map	89	W	Integer	Numeric	Integer
AInNull	83	W	Integer	Numeric	Integer
AnalogILmt	126	R	Real	Numeric	Float
AnalogILmtFilt	128	W	Real	Numeric	Float
AnalogILmtGain	127	W	Real	Numeric	Float
AnalogILmtOffset	129	W	Real	Numeric	Float
AnalogIn	0	R	Real	Numeric	Float
AnalogOut1	1	W	Real	Numeric	Float
AnalogOut2	30	W	Real	Numeric	Float
ARF0	4	W	Real	Numeric	Float
ARF1	5	W	Real	Numeric	Float
ARZ0	54	W	Real	Numeric	Float

ARZ1	55	W	Real	Numeric	Float
AxisAddr	42	W	Integer	Numeric	Integer
Baudrate	40	W	Integer	Numeric	Integer
BlkType	45	W	Integer	Numeric	Integer
Brake	77	R	Integer	Numeric	Integer
CcwInh	54	W	Integer	Numeric	Integer
CmdGain	16	W	Real	Numeric	Float
CmdGain2	90	W	Real	Numeric	Float
CommEnbl	50	W	Integer	Numeric	Integer
CommOff	17	W	Real	Numeric	Float
CommSrc	63	W	Integer	Numeric	Integer
CwInh	53	W	Integer	Numeric	Integer
DecellLmt	46	W	Real	Numeric	Float
DigitalCmd	91	R	Integer	Numeric	Integer
DigitalCmdFreq	94	R	Real	Numeric	Float
DM1F0	11	W	Real	Numeric	Float
DM1Gain	15	W	Real	Numeric	Float
DM1Map	5	W	Integer	Numeric	Integer
DM1Out	24	R	Real	Numeric	Float
DM2F0	35	W	Real	Numeric	Float
DM2Gain	32	W	Real	Numeric	Float
DM2Map	57	W	Integer	Numeric	Integer
DM2Out	26	R	Real	Numeric	Float
ElecAngTau	22	W	Integer	Numeric	Integer
Enable	6	W	Integer	Numeric	Integer
Enable2	64	W	Integer	Numeric	Integer
Enabled	7	R	Integer	Numeric	Integer
EncAlignRampICmd	25	W	Integer	Numeric	Integer
EncAlignTestDist	24	W	Integer	Numeric	Integer
EncAlignTime	23	W	Integer	Numeric	Integer
EncFreq	2	R	Real	Numeric	Float
EncIn	8	W	Integer	Numeric	Integer
EncInF0	56	W	Real	Numeric	Float

EncMode	38	W	Integer	Numeric	Integer
EncOut	37	W	Integer	Numeric	Integer
EncPos	9	R	Integer	Numeric	Integer
ExtFault	51	R	Integer	Numeric	Integer
Fault	85	R	Integer	Numeric	Integer
FaultCode	10	R	Integer	Numeric	Integer
FaultReset	61	W	Integer	Numeric	Integer
FVelErr	23	R	Real	Numeric	Float
FwV	44	R	Integer	Numeric	Integer
GearingOn	161	W	Integer	Numeric	Integer
HallState	178	R	Integer	Numeric	Integer
HomeSwitch	177	R	Integer	Numeric	Integer
HSTemp	38	R	Real	Numeric	Float
HwV	49	R	Integer	Numeric	Integer
I2tF0	124	W	Real	Numeric	Float
I2tFilt	125	R	Real	Numeric	Float
I2tThresh	123	W	Real	Numeric	Float
Icmd	21	R	Real	Numeric	Float
IFB	22	R	Real	Numeric	Float
ILmtMinus	2	W	Integer	Numeric	Integer
ILmtMode	86	W	Integer	Numeric	Integer
ILmtPlus	3	W	Integer	Numeric	Integer
Inp1	12	R	Integer	Numeric	Integer
Inp2	13	R	Integer	Numeric	Integer
Inp3	14	R	Integer	Numeric	Integer
Inp4	15	R	Integer	Numeric	Integer
Inp5	16	R	Integer	Numeric	Integer
Inp6	17	R	Integer	Numeric	Integer
InpMap1	66	W	Integer	Numeric	Integer
InpMap2	67	W	Integer	Numeric	Integer
InpMap3	68	W	Integer	Numeric	Integer
InpMap4	69	W	Integer	Numeric	Integer
InpMap5	70	W	Integer	Numeric	Integer

InpMap6	71	W	Integer	Numeric	Integer
InPosLmt	150	W	Integer	Numeric	Integer
Inputs	18	R	Integer	Numeric	Integer
IntgStopThresh	122	W	Real	Numeric	Float
Ipeak	14	R	Real	Numeric	Float
ItF0	7	W	Real	Numeric	Float
ItFilt	18	R	Real	Numeric	Float
ItThresh	43	W	Integer	Numeric	Integer
ItTreshA	83	R	Real	Numeric	Float
IU	39	R	Real	Numeric	Float
IV	40	R	Real	Numeric	Float
IW	41	R	Real	Numeric	Float
KdEnc	21	W	Integer	Numeric	Integer
KiEnc	20	W	Integer	Numeric	Integer
Kii	33	W	Real	Numeric	Float
Kip	27	W	Real	Numeric	Float
KpEnc	19	W	Integer	Numeric	Integer
Kpp	8	W	Real	Numeric	Float
Kvff	10	W	Real	Numeric	Float
Kvi	6	W	Real	Numeric	Float
Kvp	9	W	Real	Numeric	Float
Model	41	R	Integer	Numeric	Integer
Move0AccelRate	96	W	Real	Numeric	Float
Move0DecelRate	97	W	Real	Numeric	Float
Move0Dist	93	W	Integer	Numeric	Integer
Move0DistOffset	96	W	Integer	Numeric	Integer
Move0Dwell	95	W	Integer	Numeric	Integer
Move0HomeDir	97	W	Integer	Numeric	Integer
Move0HomeMode	98	W	Integer	Numeric	Integer
Move0RegSelect	162	W	Integer	Numeric	Integer
Move0RunSpeed	95	W	Real	Numeric	Float
Move0Type	92	W	Integer	Numeric	Integer
Move1AccelRate	99	W	Real	Numeric	Float

Move1DecelRate	100	W	Real	Numeric	Float
Move1Dist	100	W	Integer	Numeric	Integer
Move1DistOffset	103	W	Integer	Numeric	Integer
Move1Dwell	102	W	Integer	Numeric	Integer
Move1HomeDir	104	W	Integer	Numeric	Integer
Move1HomeMode	105	W	Integer	Numeric	Integer
Move1RegSelect	163	W	Integer	Numeric	Integer
Move1RunSpeed	98	W	Real	Numeric	Float
Move1Type	99	W	Integer	Numeric	Integer
Move2AccelRate	102	W	Real	Numeric	Float
Move2DecelRate	103	W	Real	Numeric	Float
Move2DistOffset	110	W	Integer	Numeric	Integer
Move2Dwell	109	W	Integer	Numeric	Integer
Move2HomeDir	111	W	Integer	Numeric	Integer
Move2HomeMode	112	W	Integer	Numeric	Integer
Move2RegSelect	164	W	Integer	Numeric	Integer
Move2RunSpeed	101	W	Real	Numeric	Float
Move2Type	106	W	Integer	Numeric	Integer
Move3AccelRate	105	W	Real	Numeric	Float
Move3DecelRate	106	W	Real	Numeric	Float
Move3Dist	114	W	Integer	Numeric	Integer
Move3DistOffset	117	W	Integer	Numeric	Integer
Move3Dwell	116	W	Integer	Numeric	Integer
Move3HomeDir	118	W	Integer	Numeric	Integer
Move3HomeMode	119	W	Integer	Numeric	Integer
Move3RegSelect	165	W	Integer	Numeric	Integer
Move3RunSpeed	104	W	Real	Numeric	Float
Move3Type	113	W	Integer	Numeric	Integer
Move4AccelRate	108	W	Real	Numeric	Float
Move4DecelRate	109	W	Real	Numeric	Float
Move4Distance	121	W	Integer	Numeric	Integer
Move4DistOffset	124	W	Integer	Numeric	Integer
Move4Dwell	123	W	Integer	Numeric	Integer

Move4HomeDir	125	W	Integer	Numeric	Integer
Move4HomeMode	126	W	Integer	Numeric	Integer
Move4RegSelect	166	W	Integer	Numeric	Integer
Move4RunSpeed	107	W	Real	Numeric	Float
Move4Type	120	W	Integer	Numeric	Integer
Move5AccelRate	111	W	Real	Numeric	Float
Move5DecelRate	112	W	Real	Numeric	Float
Move5Distance	128	W	Integer	Numeric	Integer
Move5DistOffset	131	W	Integer	Numeric	Integer
Move5Dwell	130	W	Integer	Numeric	Integer
Move5HomeDir	132	W	Integer	Numeric	Integer
Move5HomeMode	133	W	Integer	Numeric	Integer
Move5RegSelect	167	W	Integer	Numeric	Integer
Move5RunSpeed	110	W	Real	Numeric	Float
Move5Type	127	W	Integer	Numeric	Integer
Move6AccelRate	114	W	Real	Numeric	Float
Move6DecelRate	115	W	Real	Numeric	Float
Move6Distance	135	W	Integer	Numeric	Integer
Move6DistOffset	138	W	Integer	Numeric	Integer
Move6Dwell	137	W	Integer	Numeric	Integer
Move6HomeDir	139	W	Integer	Numeric	Integer
Move6HomeMode	140	W	Integer	Numeric	Integer
Move6RegSelect	168	W	Integer	Numeric	Integer
Move6RunSpeed	113	W	Real	Numeric	Float
Move6Type	134	W	Integer	Numeric	Integer
Move7AccelRate	117	W	Real	Numeric	Float
Move7DecelRate	118	W	Real	Numeric	Float
Move7Distance	142	W	Integer	Numeric	Integer
MoveDistOffset	145	W	Integer	Numeric	Integer
Move7Dwell	144	W	Integer	Numeric	Integer
Move7HomeDir	146	W	Integer	Numeric	Integer
Move7HomeMode	147	W	Integer	Numeric	Integer
Move7RegSelect	169	W	Integer	Numeric	Integer

Move7RunSpeed	116	W	Real	Numeric	Float
Move7Type	141	W	Integer	Numeric	Integer
MoveDist	107	W	Integer	Numeric	Integer
MoveDone	157	R	Integer	Numeric	Integer
MoveSelectBit0	158	W	Integer	Numeric	Integer
MoveSelectBit1	159	W	Integer	Numeric	Integer
MoveSelectBit2	160	W	Integer	Numeric	Integer
Out1	26	W	Integer	Numeric	Integer
Out2	27	W	Integer	Numeric	Integer
Out3	28	W	Integer	Numeric	Integer
Out4	29	W	Integer	Numeric	Integer
OutMap1	72	W	Integer	Numeric	Integer
OutMap2	73	W	Integer	Numeric	Integer
OutMap3	74	W	Integer	Numeric	Integer
OutMap4	75	W	Integer	Numeric	Integer
Outputs	30	W	Integer	Numeric	Integer
PoleCount	39	W	Integer	Numeric	Integer
PosCmdSet	56	W	Integer	Numeric	Integer
PosCommand	31	R	Integer	Numeric	Integer
PosError	32	R	Integer	Numeric	Integer
PosErrorMax	84	W	Integer	Numeric	Integer
Position	34	R	Integer	Numeric	Integer
PulsesFOut	55	W	Integer	Numeric	Integer
PulsesIn	35	W	Integer	Numeric	Integer
PulsesOut	36	W	Integer	Numeric	Integer
Reg1ActiveEdge	174	W	Integer	Numeric	Integer
Reg1EncoderPosition	171	R	Integer	Numeric	Integer
Reg1ResolverPosition	170	R	Integer	Numeric	Integer
Reg2ActiveEdge	175	W	Integer	Numeric	Integer
Reg2EncoderPosition	173	R	Integer	Numeric	Integer
Reg2ResolverPosition	172	R	Integer	Numeric	Integer
RemoteFB	65	W	Integer	Numeric	Integer
ResPos	33	R	Integer	Numeric	Integer

RunStop	62	W	Integer	Numeric	Integer
StartMove	156	W	Integer	Numeric	Integer
StopTime	31	W	Real	Numeric	Float
Vbus	44	R	Real	Numeric	Float
VBusTresh	43	W	Real	Numeric	Float
VelCmd	19	W	Real	Numeric	Float
VelCmd2	36	W	Real	Numeric	Float
VelCmdA	37	R	Real	Numeric	Float
VelCmdSrc	76	W	Integer	Numeric	Integer
VelErr	20	R	Real	Numeric	Float
VelFB	25	R	Real	Numeric	Float
VelLmtHi	48	W	Real	Numeric	Float
VelLmtLo	49	W	Real	Numeric	Float
Velocity	3	R	Real	Numeric	Float
ZeroSpeedThresh	91	W	Real	Numeric	Float

The following is an Alphabetical listing of Real Time Calls with PDID

Name	PDID	R/W	Data Format	Display Format	Data Type
LoadCalTable	3	n/a	Real	ASCII	Command
NVLoad	0	n/a	Real	ASCII	Command
NVSave	1	n/a	Real	ASCII	Command
SaveCalTable	4	n/a	Real	ASCII	Command
TableGen	5	n/a	Real	ASCII	Command
Unconfigure	2	n/a	Real	ASCII	Command



HIGH PERFORMANCE MOTORS & DRIVES

4301 Kishwaukee Street
P.O. Box 106
Rockford, Illinois 61105-0106
(815) 226-3100 Fax (815) 226-3080

Locations:
Rockford, IL
Wilmington, MA
Juarez, Mexico

www.pacsci.com