



Artisan Technology Group is your source for quality new and certified-used/pre-owned equipment

- FAST SHIPPING AND DELIVERY
- TENS OF THOUSANDS OF IN-STOCK ITEMS
- EQUIPMENT DEMOS
- HUNDREDS OF MANUFACTURERS SUPPORTED
- LEASING/MONTHLY RENTALS
- ITAR CERTIFIED SECURE ASSET SOLUTIONS

SERVICE CENTER REPAIRS

Experienced engineers and technicians on staff at our full-service, in-house repair center

*InstraView*SM REMOTE INSPECTION

Remotely inspect equipment before purchasing with our interactive website at www.instraview.com ↗

WE BUY USED EQUIPMENT

Sell your excess, underutilized, and idle used equipment. We also offer credit for buy-backs and trade-ins. www.artisanng.com/WeBuyEquipment ↗

LOOKING FOR MORE INFORMATION?

Visit us on the web at www.artisanng.com ↗ for more information on price quotations, drivers, technical specifications, manuals, and documentation

Contact us: (888) 88-SOURCE | sales@artisanng.com | www.artisanng.com

USER MANUAL

Accessory 36P

16 Channel 12-Bit ADC Board

3Ax-602363-xUxx

October 16, 2003



DELTA TAU
Data Systems, Inc.

NEW IDEAS IN MOTION ...

Single Source Machine Control *Power // Flexibility // Ease of Use*
21314 Lassen Street Chatsworth, CA 91311 // Tel. (818) 998-2095 Fax. (818) 998-7807 // www.deltatau.com

Copyright Information

© 2003 Delta Tau Data Systems, Inc. All rights reserved.

This document is furnished for the customers of Delta Tau Data Systems, Inc. Other uses are unauthorized without written permission of Delta Tau Data Systems, Inc. Information contained in this manual may be updated from time-to-time due to product improvements, etc., and may not conform in every respect to former issues.

To report errors or inconsistencies, call or email:

Delta Tau Data Systems, Inc. Technical Support

Phone: (818) 717-5656

Fax: (818) 998-7807

Email: support@deltatau.com

Website: <http://www.deltatau.com>

Operating Conditions

All Delta Tau Data Systems, Inc. motion controller products, accessories, and amplifiers contain static sensitive components that can be damaged by incorrect handling. When installing or handling Delta Tau Data Systems, Inc. products, avoid contact with highly insulated materials. Only qualified personnel should be allowed to handle this equipment.

In the case of industrial applications, we expect our products to be protected from hazardous or conductive materials and/or environments that could cause harm to the controller by damaging components or causing electrical shorts. When our products are used in an industrial environment, install them into an industrial electrical cabinet or industrial PC to protect them from excessive or corrosive moisture, abnormal ambient temperatures, and conductive materials. If Delta Tau Data Systems, Inc. products are directly exposed to hazardous or conductive materials and/or environments, we cannot guarantee their operation.

Table of Contents

INTRODUCTION	1
ACC-36P HARDWARE SPECIFICS.....	3
PMAC Address Select Jumpers.....	3
<i>Non-Turbo PMAC Addresses.....</i>	<i>4</i>
<i>Turbo PMAC Addressing.....</i>	<i>4</i>
E-Point Jumpers	4
Power Supply Connection	4
Adjustment Potentiometers	4
Power Supply Requirements	5
PMAC’s JEXP Limitation.....	5
Acc-36P Connections.....	5
<i>Connection to PMAC of First Acc-36P</i>	<i>5</i>
<i>Connecting Acc-36 to PMAC with Battery-Backed CPUs.....</i>	<i>6</i>
<i>Connecting Accessory 36 to PMACs with Flash CPUs.....</i>	<i>6</i>
<i>Connecting Multiple Acc-36Ps to PMAC</i>	<i>6</i>
<i>Connecting Acc-36P to PMAC via Acc-14</i>	<i>6</i>
<i>Connecting Acc-36P to PMAC via Acc-24 or Acc-29</i>	<i>6</i>
<i>Connection to Analog Signal.....</i>	<i>7</i>
ACC-36 THEORY OF OPERATION	9
Using PLCs to Read the Analog Inputs.....	9
M-Variable Definitions	9
Data Acquisition.....	10
Reading Data Through PLC Programs.....	11
NON TURBO PMAC AUTOMATIC READ FEATURE.....	13
Using the Automatic Read Feature for Data Acquisition	13
Using the Automatic Read Feature for Servo Feedback	14
Non-Turbo PMAC Encoder Conversion Table Setup.....	14
Motor I-Variable Setup	15
Absolute Power-On Position for Non-Turbo PMAC	15
<i>Example using Acc-36 for both Data Acquisition and Servo Feedback</i>	<i>16</i>
TURBO PMAC AUTOMATIC READ FEATURE.....	19
Turbo PMAC Data Acquisition Example:.....	20
Acc-36P Servo Feedback Use Example for Turbo PMAC	20
Acc-36P Power-On Position for Turbo PMAC.....	21
ACC-36 PINOUTS	23
TB1 (4-pin Terminal Block).....	23
J4 (JAPAN DB37).....	24
DIAGRAMS.....	25
APPENDIX A.....	29
I60: Auto-Converted ADC Register Address Range.....	29
I61: Number of Auto-Converted ADC Registers	29

INTRODUCTION

PMAC's Accessory 36P (Acc-36P) is an analog data acquisition board capable of converting 16 analog input signals. The basic Acc-36P board is populated for sixteen (16) channels of analog input. The Analog-to-Digital Converter (ADC) units used in Acc-36P are the MAX180 monolithic devices manufactured by Maxim Integrated Products. These devices have 12-bit resolution with $\pm 1/2$ LSB linearity specification. For more details of the ADC chips please refer to the data sheet published by the manufacturer:

Document 19-3950; Rev 0, 6/91
"Complete, 8-Channel, 12-Bit Data Acquisition Systems"
Maxim Integrated Products
120 San Gabriel Drive
Sunnyvale, CA 94086
Phone: (408) 737-7600

The Acc-36P's design features make it an ideal analog data acquisition board for monitoring and collection of signals from a variety of sensors and transducers. Up to 24 Acc-36Ps may be connected to PMAC providing up to 384 possible analog input channels. By using simple M-variable assignments, the converted data may be used in PLC programs for monitoring and data collection purposes.

For PMAC firmware V1.15 and above, 16 channels may be read to specified PMAC memory locations automatically. These registers can be monitored using M-Variables or read into PMAC's encoder conversion table for servo feedback control. Only one Acc-36 may be used in this fashion per PMAC. If one chooses to use this automatic feature, all channels on the card must be read in this manner. See the Using Acc-36 Data for Servo Feedback Use section of this manual for more details.

ACC-36P HARDWARE SPECIFICS

Refer to the layout diagram of Acc-36P for the location of the connectors on the board. The pin definition listings for J4 and TB1 are provided at the end of this manual.

J1

This connector is used for interface to PMAC's processor bus via the JEXP connector through the supplied 50 pin flat cable. The signals which are brought in through this connector are buffered on board. This connector should be used to connect an Acc-36P directly to a PMAC that does not have a buffered JEXP port. (PMACs with battery-backed memory do not have buffered JEXP ports). When connecting to a PMAC with buffered JEXP port (PMAC with flash backed RAM have buffered JEXP ports) or to another Acc-36 or Acc-14 board, use J2 instead. When using J1, jumper E11 must be off. (See PMAC's JEXP Connector Limitation.)

J2

This connector is used for interface to PMAC's processor bus via the JEXP connector through the supplied 50 pin flat cable. The signals which are brought in through this connector are not buffered on board. This connector should be used to connect an Acc-36P directly to a PMAC with a buffered JEXP port (PMAC with flash backed RAM have buffered JEXP ports) or to another Acc-36 or Acc-14 board. When using J2, jumper E11 must be on. (See PMAC's JEXP Connector Limitation.)

J3

This connector takes out the signals brought in via the J1 or the J2 connector to the J2 connector of another Acc-36P which is further away from PMAC. (See connection diagrams.)

J4

Through this connector the analog signals are brought into Acc-36P. In addition the +/- 12 to 15V power supplies are brought out. These power supplies may be used in situations where a separate supply unit is not available for the analog transducers. Note that the two fuses limit the current drawn to 0.5 A on each supply line.

J5

This is a 20-pin header which is used for factory calibration. Do not use this connector.

TB1

This is a 4-pin terminal block which provides the connection for power supply inputs to Acc-36P when it is used in a standalone configuration.

PMAC Address Select Jumpers

Each Acc-36P occupies two Y-memory locations on the address space of the PMAC's processor bus. On board provisions are made for address decoding such that up to 24 Acc-36Ps may be connected to a single PMAC. For an individual Acc-36P card, jumpers E1 through E10 determine its particular address. Table 1 shows the jumper selection scheme for the base address specification.

Non-Turbo PMAC Addresses

Jumpers ¹	E7	E8	E9	E10
E1	Y:\$FFD0 ²	Y:\$FFD2	Y:\$FFD4	Y:\$FFD6
E2	Y:\$FFD8	Y:\$FFDA	Y:\$FFDC	Y:\$FFDE
E3	Y:\$FFE0	Y:\$FFE2	Y:\$FFE4	Y:\$FFE6
E4	Y:\$FFE8	Y:\$FFEA	Y:\$FFEC	Y:\$FFEE
E5	Y:\$FFF0	Y:\$FFF2	Y:\$FFF4	Y:\$FFF6
E6	Y:\$FFF8	Y:\$FFFA	Y:\$FFFC	Y:\$FFFE

¹Note that only one row jumper (E1 to E6) and only one column jumper (E7 to E10) may be installed at any time. Otherwise, the communication between this accessory board and PMAC will be not work properly.

² Factory Default. Note that the odd addresses are decoded for the Status Register of the ADC chips. If the recommended ADC conversion procedure is used (see below), this register is not read.

Turbo PMAC Addressing

Jumpers ¹	E7	E8	E9	E10
E1	Y:\$78A00 ²	Y:\$78A02	Y:\$ 78A04	Y:\$ 78A06
E2	Y:\$78B00	Y:\$78B02	Y:\$ 78B04	Y:\$ 78B06
E3	Y:\$78C00	Y:\$78C02	Y:\$ 78C04	Y:\$ 78C06
E4	Y:\$78D00	Y:\$78D02	Y:\$ 78D04	Y:\$ 78D06
E5	Y:\$78E00	Y:\$78E02	Y:\$ 78E04	Y:\$ 78E06
E6	Y:\$78F00	Y:\$78F02	Y:\$ 78F04	Y:\$ 78F06

¹Note that only one row jumper (E1 to E6) and only one column jumper (E7 to E10) may be installed at any time. Otherwise, the communication between this accessory board and PMAC will be not work properly.

² Factory Default. Note that the odd addresses are decoded for the Status Register of the ADC chip(s). If the recommended ADC conversion procedure is used (see below), the user does not need to read this register.

E-Point Jumpers

The function of the address jumpers E1 to E10 is described above (see table 1). The jumper on E11 should only be installed if more than one Acc-36P are used with a single PMAC. In this case, the second and subsequent Acc-36Ps must have E11 installed. E11 must not be installed on the first or the only Acc-36P connected to PMAC. E12 is for factory use only and must not have a jumper installed at any time. E11 should be OFF when the board is connected to PMAC via J1 and E11 should be ON when the board is connected via J2.

Power Supply Connection

If Acc-36P is installed on the PC-bus, both the +5 V supply and the ± 12 V supplies are brought in through the bus. For standalone operations the terminal block TB1 should be used. The power supply requirements are approximately 100 mA for each of the three supplies.

Adjustment Potentiometers

There are 16 analog offset adjustment pots. These 12 turn pots are located at the top edge of the printed circuit board. From left to right R4 is for Channel 1, R8 is for channel 2 etc., and R36 is for channel 9, R40 is for channel 10 etc. (See the schematic.)

R65 and R67 pots are the voltage reference adjustment pots for the two ADC chips (these are factory preset and should not be readjusted). Also R66 and R68 are the digital offset pots for the two ADC chips.

Power Supply Requirements

Acc-36P draws approximately 100 mA for each of its three supply voltages (+12 V, -12V and +5 V).

5V	12V	-12V	Other 24V etc.
100 mA	100 mA	100 mA	N/A

Note:

Since the analog inputs are not optically isolated on this board, the $\pm 12V$ supply to this board should not be from the same supply that is used for the PMACs optically isolated analog outputs (DACs).

PMAC's JEXP Limitation

JEXP (expansion) is the 50-pin cable connector located on PMAC's CPU board. There are limitations to the amount of unbuffered boards connected to one PMAC. These limitations vary with the type of PMAC ordered. Currently PMAC has several options which can be ordered to enhance PMAC's processing speed:

- Standard 20 MHz CPU, One-Wait State RAM, Battery backup
- Option 4A 20 MHz CPU, Zero-Wait State RAM, Flash backup
- Option 5 30 MHz CPU, Zero-Wait State RAM, Battery backup
- Option 5A 40 MHz CPU, Zero-Wait State RAM, Flash Backup
- Option 5B 60 MHz CPU, Zero-Wait State RAM, Flash Backup

The number of unbuffered accessory boards which can be used with one Standard PMAC, PMAC with Option 5, and PMAC with Option 4A, 5A, or 5B is listed below.

PMAC with Flash CPU vs PMAC with Battery backed CPU:

- Standard PMAC (20 MHz): Two unbuffered boards
- Option 5 (30 MHz) One unbuffered board
- Option 4A, 5A, 5B No limit (buffers are included on PMAC)

Unbuffered Accessory Boards:

- Option 2 Dual-ported RAM
- Acc-24 Axis Expansion Board
- Acc-29 MLDT Interface Board

Accessory Boards for which only the first board is unbuffered:

- Acc-14 Digital I/O boards
- Acc-36 A/D Conversion Boards

If PMAC has flash memory, the on board buffers on the accessory boards are bypassed. However, for PMACs with battery backed CPUs, the accessory board which is connected to PMAC's JEXP connector uses its on board buffers. See connection diagrams on pages 17 through 19.

The maximum length of the cable between boards is 6 in. (150 mm).

Acc-36P Connections

In order to use Acc-36P in conjunction with PMAC and other related accessories several connections are required. In this section these connections will be explained.

Connection to PMAC of First Acc-36P

Note that a 5.25 inches long 50-pin flat cable is supplied with each Acc-36P. This cable should be used to connect PMAC's JEXP to Acc-36P's J1 or J2 connector. (See PMAC's JEXP Limitation Section.) In addition the jumper E11 must be removed for battery backed PMACs or E11 should be connected if using PMAC with flash memory (the OFF position is the default factory setting for E11).

Connecting Acc-36 to PMAC with Battery-Backed CPUs

When connecting Acc-36 to a PMAC with battery-backed memory, connect the J1 connector to PMAC's JEXP connector and make sure jumper E11 is open (off) for the first Acc-36. When connecting several Acc-36s to a PMAC with battery backed memory, connect the first Acc-36 (J1) to PMAC (JEXP), and then connect the next Acc-36 from J3 to the previous Acc-36 at its J2 connector. The table below summarizes the connection to PMAC with battery backed CPU.

From	To	Jumper E11
Acc-36 #1 J1	PMAC (JEXP)	Open
Acc-36 #2 J2	Acc-36 #1 J3	Closed
Acc-36 #3 J2	Acc-36 #2 J3	Closed

Connecting Accessory 36 to PMACs with Flash CPUs

If connecting only one Acc-36 to a PMAC with flash CPU, then connect the 50 pin ribbon from the J2 connector to PMAC's JEXP connector and also close the E11 jumper*. When connecting several Acc-36s to a PMAC with Flash Memory, connect the first Acc-36 (J2) to PMAC (JEXP), and then connect the next Acc-36 from J3 to the previous Acc-36 at its J2 connector. The table below summarizes the connection to PMAC with flash CPU:

From	To	Jumper E11
Acc-36 #1 J2	PMAC (JEXP)	Closed*
Acc-36 #2 J2	Acc-36 #1 J3	Closed
Acc-36 #3 J2	Acc-36 #2 J3	Closed

The reason J2 is used instead of J1 is to bypass the buffers on the accessory board since this PMAC option has on-board buffers. Connecting to PMAC in this manner makes the data transfer a much faster process.

* If bypassing the Acc-36 onboard buffers (connecting J2 to JEXP), failure to jumper E11 on a PMAC with flash memory results in a PMAC watchdog error.

Connecting Multiple Acc-36Ps to PMAC

As mentioned above, up to 24 Acc-36Ps may be connected to a single PMAC card. The address select jumpers (E1 to E10) must be correctly set on each Acc-36P board (see table 1). In addition, the Acc-36P board that connects directly to PMAC must do so through its J1 or J2 connector and have its E11 jumper removed for battery backed CPUs or have E11 jumpered for flash memory (this OFF position is the default factory setting). For additional Acc-36Ps that daisy chain through the first Acc-36P, E11 should be ON. Also the J1 connector should not be used (see the connection diagram for multiple Acc-36Ps). Instead, use J2 to bring in the signals and J3 to take them out to another Acc-36P.

Connecting Acc-36P to PMAC via Acc-14

When using one or more Acc-36Ps in conjunction with the I/O Expansion Board Accessory (Acc-14), it is recommended to connect the Acc-14D directly to PMAC. Acc-36Ps should then be connected to PMAC via the Acc-14D (see the connection diagram). The direct link between Acc-14D and PMAC should be use the J8 connector. On the first and the subsequent Acc-36Ps use the J2 connectors and install E11 jumpers (see the relevant connection diagram).

Connecting Acc-36P to PMAC via Acc-24 or Acc-29

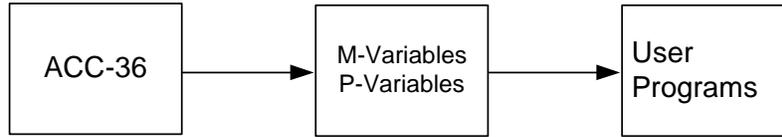
If one or more Acc-36Ps are used in conjunction with the Axis Expansion Board (Acc-24) or the MLDT Interface Board (Acc-29), then Acc-24 Option 2 is required. This is a daisy chain cable which connects PMAC's JEXP to Acc-24 or Acc-29's J1 and the first Acc-36P's J1. For the first Acc-36P E11 must be OFF. For the second and subsequent Acc-34Ps use J2 as input and J3 as output and install E11 (see the relevant connection diagram).

Connection to Analog Signal

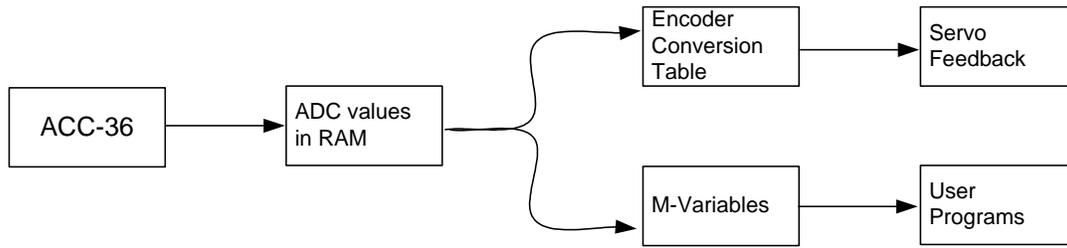
The analog signals are brought in through J4. For a single-ended connection (using ADCx and GND), the voltage range should be from 0 to 10 volts for unipolar signals and -5V to +5V for bipolar signals. For a differential connection (using ADCx and ADCx/) the voltage range should be between 0 to 5V for unipolar signals and -2.5V to 2.5V for bipolar signals. The $\pm 12V$ power supply is also brought out through this connector. Note that the two fuses limit the current drawn to 0.5 A on each supply line. For single ended inputs, ground the complimentary signals.

ACC-36 THEORY OF OPERATION

Reading the analog data through Acc-36P is a simple procedure. There are two methods the user may use to read the analog inputs. The user may write PLCs which monitor the Acc-36 board or use PMAC's automatic ADC register read feature. The PLC method will be discussed first followed by a detailed description of PMAC's automatic reading feature. The block diagram below shows the information flow from Acc-36 to the user programs.



The automatic read feature in PMAC is a simple process which enables the user to read the analog signals as feedback devices or for normal data acquisition by having M-variables pointing to memory locations which contain the information received by the automatic read feature. Remember, the signals used for this feature may only be read as bipolar inputs. If there is only one Acc-36 and the inputs range from -2.5 to 2.5V differential or -5V to 5V single ended, this method is recommended. The block diagram below shows the analog data flow for servo feedback and user programs.



Using PLCs to Read the Analog Inputs

If a PLC program were written to monitor the data, first define up to three M-variables for each Acc-36P board. Once the M-variables have been defined, PMAC's PLC programs may be used to initialize the analog-to-digital conversion process. One advantage of using PLC programs for data acquisition is the data to be read can be defined as either unipolar or bipolar. If PMAC's automatic read feature is used, it is limited to reading the data as bipolar only.

M-Variable Definitions

Two M-variables must be defined for an eight channel Acc-36P and three M-variables for an Acc-36P with its Option 1. A 24-bit wide unsigned integer M-variable must be pointed to the base address of the board (see table 1 for the base address E-point jumper setting). Next must define two more M-variables each 12-bits wide and pointed to the same base address. For bipolar signals (-5 to +5V single-ended and +2.5V to -2.5V differential) these M-variables must be defined as signed integers. For unipolar signals (0 to +10V single-ended or 0 to +5V differential) they should be defined as unsigned. For example, if the base address is at Y:\$FFD0 for the non-Turbo PMAC or \$78A00 for Turbo PMAC (assuming E1 and E7 are ON), then the three M-variables can be defined as follows:

For bipolar signals:

Non Turbo	Turbo	Description
M100->Y:\$FFD0,24	M100->Y:\$78A00,24	Conversion Channel Select
M101->Y:\$FFD0,0,12,s	M101->Y:\$78A00,0,12,s	Read Data for channels 1-8
M102->Y:\$FFD0,12,12,s	M102->Y:\$78A00,12,12,s	Read Data for channels 9-16

For unipolar signals:

Non Turbo	Turbo	Description
M100->Y:\$FFD0,24	M100->Y:\$78A00,24	Conversion Channel Select
M101->Y:\$FFD0,0,12,u	M101->Y:\$78A00,0,12,u	Read Data for channels 1-8
M102->Y:\$FFD0,12,12,u	M102->Y:\$78A00,12,12,u	Read Data for channels 9-16

Note that the address Y:\$FFD0 (non-Turbo) and Y:\$78A00 (Turbo) is the same for all three M-variables. Also, the third M-variable (in this case M102) is needed only if the Acc-36P board has its Option 1 installed.

Data Acquisition

By writing into the ADC registers pointed to by Conversion Channel Select M-variable (M100 in the above example), the analog-to-digital conversion process is initialized. Afterwards, the converted data may be read through the Read Data M-variables (M101 and M102 in the above example). The data written into the Conversion Channel Select M-variable determines both the input channel and the conversion type (unipolar Vs bipolar). This is shown in the following table:

Base Address Input ¹	Selected Analog Input Channels ²	Polarity	Single-Ended Range ³ (Volts)	Differential Range ⁴ (Volts)
0	1 & 9	Unipolar	0 to 10	0 to 5
1	2 & 10	Unipolar	0 to 10	0 to 5
2	3 & 11	Unipolar	0 to 10	0 to 5
3	4 & 12	Unipolar	0 to 10	0 to 5
4	5 & 13	Unipolar	0 to 10	0 to 5
5	6 & 14	Unipolar	0 to 10	0 to 5
6	7 & 15	Unipolar	0 to 10	0 to 5
7	8 & 16	Unipolar	0 to 10	0 to 5
8	1 & 9	Bipolar	-5 to 5	-2.5 to 2.5
9	2 & 10	Bipolar	-5 to 5	-2.5 to 2.5
10	3 & 11	Bipolar	-5 to 5	-2.5 to 2.5
11	4 & 12	Bipolar	-5 to 5	-2.5 to 2.5
12	5 & 13	Bipolar	-5 to 5	-2.5 to 2.5
13	6 & 14	Bipolar	-5 to 5	-2.5 to 2.5
14	7 & 15	Bipolar	-5 to 5	-2.5 to 2.5
15	8 & 16	Bipolar	-5 to 5	-2.5 to 2.5

¹The base address is selected using jumpers E1 to E10 (see table 1). The value in this column would be the value given to M100 in the above example.
² Channels 9 to 16 are only applicable when Option 1 is installed.
³ For single-ended wiring use ADCx input and AGND return.
⁴ For differential wiring use ADCx and ADCx/ inputs.

Reading Data Through PLC Programs

For example, channels 4 and 10 must be converted as unipolar inputs and read channels 1 and 9 as bipolar in the PLC 10 program:

Non Turbo	Turbo	Description
M100->Y:\$FFD0,24	M100->Y:\$78A00,24	Conversion Channel Select
M101->Y:\$FFD0,0,12,s	M101->Y:\$78A00,0,12,s	Read Data for channels 1-8
M102->Y:\$FFD0,12,12,s	M102->Y:\$78A00,12,12,s	Read Data for channels 9-16

```

OPEN PLC 10
CLEAR
.
.
.
M100=3                ;convert channel 4 as unipolar(also converts channel 12)
P104=M101             ;P100 now contains converted channel 4 data
M100=1                ;convert channel 10 as unipolar (also converts
channel 2)
P110=M102             ;P100 now contains converted channel 10 data
M100=8                ;convert channels 1 and 9 as bipolar
P101=M101             ;P101 now contains converted channel 1 data
P109=M102             ;P109 now contains converted channel 9 data
.
.
CLOSE
    
```


NON TURBO PMAC AUTOMATIC READ FEATURE

Starting in V1.15 of PMAC firmware, it is possible to use analog-to-digital converter channels on a single Acc-36P/V for servo loop feedback or data acquisition. V1.15 adds automatic software routines which copy an ADC pair of values into PMAC's RAM and select a new channel on the card every phase update (default of 9 kHz). The PMAC encoder conversion table and servo loop routines can then treat these RAM registers as if they were the ADC registers themselves. To use the automatic read feature I60 must be set to the address location of the board which contains the channel which will utilize this feature. I61 must be set to the number of channels on the card which will use this feature. Detailed definitions of these variables are given in the appendix of the manual.

If the automatic read feature is enabled PMAC will automatically store the information from the Acc-36 card to PMAC memory locations \$0708 to \$070F. Channels 1 through 8 are read to the Y registers and channels 9 through 16 are read to the X registers.

*PMAC2 does not have I variables I60 and I61. PMAC2 with option 12 or 12A (8 or 16 analog to digital converters) does have the automatic read feature built into its firmware. PMAC2's automatic read feature operates similar to a regular PMAC and this information can be found in the PMAC2 Software Addendum. If Acc-36 is ordered to use with PMAC2, the automatic read feature would not be available to the inputs from this accessory. The analog inputs to PMAC2 can be read using PLCs as mentioned previously in this manual.

Using the Automatic Read Feature for Data Acquisition

If the user wanted to use the card for data acquisition only, M-variable pointers must be defined to the 16 memory locations PMAC writes to automatically. To enable the feature, I60 must be set to address space of the PMAC's processor bus (see Address Select Jumpers section in this manual). To copy all ADCs into memory automatically, I61 should be set to a value of 0 or 7.

```
m61->y:$0708,0,12,u ;channel 1 A to D as bipolar
m62->y:$0709,0,12,u ;channel 2 A to D as bipolar
m63->y:$070A,0,12,u ;channel 3 A to D as bipolar
m64->y:$070B,0,12,u ;channel 4 A to D as bipolar
m65->y:$070C,0,12,u ;channel 5 A to D as bipolar
m66->y:$070D,0,12,u ;channel 6 A to D as bipolar
m67->y:$070E,0,12,u ;channel 7 A to D as bipolar
m68->y:$070F,0,12,u ;channel 8 A to D as bipolar
m69->x:$0708,0,12,u ;channel 9 A to D as bipolar
m70->x:$0709,0,12,u ;channel 10 A to D as bipolar
m71->x:$070A,0,12,u ;channel 11 A to D as bipolar
m72->x:$070B,0,12,u ;channel 12 A to D as bipolar
m73->x:$070C,0,12,u ;channel 13 A to D as bipolar
m74->x:$070D,0,12,u ;channel 14 A to D as bipolar
m75->x:$070E,0,12,u ;channel 15 A to D as bipolar
m76->x:$070F,0,12,u ;channel 16 A to D as bipolar
```

Once these variables are defined in PMAC memory, the user may use the M-Variables in PMAC motion programs and background PLCs.

Using the Automatic Read Feature for Servo Feedback

The ADCs from only one Acc-36P/V can be used for servo loop feedback, even if there is more than one Acc-36P/V attached to the PMAC. Parameter I60 specifies the PMAC address of the Acc-36P/V to be used in this fashion. This value must match the address selected by jumpers E1-E10 on the Acc-36P/V to be used for servo loop feedback. If I60 is set to zero, this function is disabled.

Parameter I61 specifies the number of ADCs (or pairs of ADCs if sub-option 1 is ordered) to be copied into RAM in this cyclic fashion. I61 has a range of 0 to 7, and the number of ADCs, or pairs of ADCs in the copying cycle is equal to I61+1. As a special note, a cycle of 1 is not possible, so a value of I61=0 is the same as a value of I61=7, telling PMAC to copy all eight ADCs (or pairs) in RAM in the cycle.

Some thought must be given to setting the proper value of I61, given the following tradeoff. First, the more ADCs in the copying cycle, the lower the maximum servo update frequency that is possible. Second, any ADCs on this ADC that are not in the copying cycle cannot be used at all.

The servo loop for a motor should not be closed more often than new feedback data is available to it. If all ADCs are in the copying cycle, new data is available every 8th phase cycle. At the default settings on PMAC, a servo loop is closed every 4th phase cycle. There are several possible actions if the setup has the servo loop being closed faster than new data is available.

First, if the servo loop for all of the axes can be slowed down, change jumpers E3-E6 on PMAC to generate a servo interrupt less often (e.g. every 8th phase cycle instead of every 4th). The default servo update rate of 2.25 kHz is faster than is needed for most applications

Second, it is possible to slow down the servo update of selected axes only with Ix60 (not I60). For example, if there is a servo interrupt every 4th phase cycle, a motor x with Ix60=1 will close its servo loop every 2nd servo interrupt, which is every 8th phase cycle.

Third, it is possible to speed up the phase cycle by changing the setting of jumpers E29-E33 on PMAC, also changing E3-E6 on PMAC to increase the divide-down of the servo cycle from the phase cycle to keep its frequency the same. For example, change the phase cycle from the default of 9 kHz to 18 kHz, also change to a divide-by-8 for the servo cycle, keeping it at 2.25 kHz. If no motors are doing phase commutation in PMAC, this adds only a slight computational overhead, but if several motors are being commutated by PMAC, this can add significant overhead.

Non-Turbo PMAC Encoder Conversion Table Setup

With I60 and I61 set properly some or all of the ADCs on a single Acc-36P/V are copied into RAM registers automatically. ADC1 is copied into Y:\$0708 low bits, ADC9 is copied into X:\$0708 low 12 bits; ADC2 into Y:\$0709 low bits, ADC10 into X:\$710 low bits and so on; if all eight are copied, ADC16 is copied into X:\$070F.

If the value of I60 matches the address selected by jumpers E1 through E10 on Acc-36, then the A/D data will go to the following locations.

Channel	Source Address	*Entry Address	*Converted Address
ADC1	\$Y:0708	\$072A	\$072C
ADC9	\$X:0708	\$072D	\$072F
ADC2	\$Y:0709	\$0730	\$0732
ADC10	\$X:0709	\$0733	\$0735
* Assuming Default Encoder Conversion Table Used (User Selectable)			
** Encoder Conversion Table ends at location \$73F			

Before the values in these registers can be used for servo feedback, the conversion table must process them. The conversion table can process many types of feedback; for these registers it should use the parallel data format, and not the ADC format (which is used for Acc-28 data in the high 16 bits of a register).

The PMAC Executive program has an interactive menu for adding entries to the conversion table; for conversion method, select Parallel Y Data for ADCs 1-8, which appear in Y-memory; or Parallel X Data for ADCs 9-16, which appear in X-memory. Source Address is a value from \$0708 to \$070F, depending on the ADC number. In either case, the Bit Enable Mask should be set to \$000FFF, which specifies the use of the low 12 bits only of the register.

Using either X or Y parallel data, it is possible for the conversion table to process the data with filtering or without filtering. The use of the conversion table format with filtering is recommended as protection against noise. The filtering limits the maximum change in input value per servo cycle that will be passed through. This value, the third setup value in the entry, should be set to a value slightly higher than the maximum real velocity ever expected.

If creating these entries with direct memory writes, Parallel Y Data With Filtering is format \$30; Parallel X Data With Filtering is format \$70. For example, to add conversions for ADCs 1 and 9 to the end of the default conversion table, use the following on-line command:

```
WY:$072A, $300708, $000FFF, $000010, $700708, $000FFF, $000010
```

Y:\$072A is the first open register in the conversion table itself after the default entries. \$300708 tells the table to use the parallel data in Y:\$0708. \$000FFF tells it to use only the low 12 bits (each F represents 4 bits). \$000010 says that the maximum permitted velocity is 16 (\$10) bits per servo cycle. \$700708 tells the table to use the parallel data in X:\$0708; \$000FFF and \$000010, are the same as for the Y data.

The processed data for the first entry is put in X:\$072C -- the last line of the entry; similarly, the processed data for the second entry is put in X:\$072F.

Remember that conversion table entries must be stored to non-volatile memory with the **SAVE** command in order to be kept through a power-down cycle or reset.

Motor I-Variable Setup

For use in the servo loop, the feedback-address I-variables must be set to point to the proper registers. Ix03 and Ix04 specify the registers to be used for position loop feedback and velocity loop feedback, respectively. Usually, these specify an address in the conversion table containing a processed feedback value, and usually the same register.

Continuing the above example, if Motor 1 were to use ADC1 for position and velocity feedback, and Motor 2 were to use ADC9, I103 and I104 would be set to \$072C (this always specifies an X register); I203 and I204 would be set to \$072F.

Absolute Power-On Position for Non-Turbo PMAC

If absolute power-on position information is desired, as it usually is, Ix10 (new in V1.14 firmware) must be set to point to the proper register and to tell PMAC how to interpret that register. If Ix10 is set to 0, PMAC will always treat the power-on position as zero. Ix10 should point to the register containing the data before processing through the conversion table.

In our example, I110 would be set to \$8C0708, where the 8 specifies a signed value from a Y register, the C specifies a 12-bit value, and the \$0708 specifies the address. I210 would be set to \$CC0708, where the first C (8+4) specifies a signed value from an X register, the second C specifies a 12-bit value, and the \$0708 specifies the address.

Channel	Ix10 (signed)	Source Address
1	\$8C0708	Y:\$0708
2	\$8C0709	Y:\$0709
3	\$8C070A	Y:\$070A
4	\$8C070B	Y:\$070B
5	\$8C070C	Y:\$070C
6	\$8C070D	Y:\$070D
7	\$8C070E	Y:\$070E
8	\$8C070F	Y:\$070F
9	\$CC0708	X:\$0708
10	\$CC0709	X:\$0709
11	\$CC070A	X:\$070A
12	\$CC070B	X:\$070B
13	\$CC070C	X:\$070C
14	\$CC070D	X:\$070D
15	\$CC070E	X:\$070E
16	\$CC070F	X:\$070F

Remember that I-variables must be stored to non-volatile memory with the **SAVE** command in order to be retained through a power-down cycle or reset.

Example using Acc-36 for both Data Acquisition and Servo Feedback

For this example, use the first eight channels for servo feedback and the last eight channels for data acquisition. To accomplish this, download the following definitions to PMAC:

Encoder Conversion Table Entries

```
WY:$0720,$300708,$000FFF,$000010
WY:$0723,$300709,$000FFF,$000010
WY:$0726,$300708,$000FFF,$000010
WY:$0729,$300708,$000FFF,$000010
WY:$072C,$300708,$000FFF,$000010
WY:$072F,$300708,$000FFF,$000010
WY:$0732,$300708,$000FFF,$000010
WY:$0735,$300708,$000FFF,$000010
```

M-Variables used for Data Acquisition

```
m69->x:$708,0,12,u           ;channel 9 A to D
m70->x:$709,0,12,u           ;channel 10 A to D
m71->x:$70A,0,12,u           ;channel 11 A to D
m72->x:$70B,0,12,u           ;channel 12 A to D
m73->x:$70C,0,12,u           ;channel 13 A to D
m74->x:$70D,0,12,u           ;channel 14 A to D
m75->x:$70E,0,12,u           ;channel 15 A to D
m76->x:$70F,0,12,u           ;channel 16 A to D
```

Remember:

Since the data is being processed as a filtered parallel word, it uses three rows in the encoder conversion table.

Example: If encoder channel #1 is sourced from location Y:\$720, the processed encoder information be at location X:\$722. Therefore, to use a filtered parallel word as feedback, set Ix03 = \$722 and Ix04 = \$722.

Position and Velocity Feedback Variables	Encoder Conversion Table Entry
I103, I104	\$722
I203, I204	\$725
I303, I304	\$728
I403, I404	\$72B
I503, I504	\$72E
I603, I604	\$731
I703, I704	\$734
I803, I804	\$737

TURBO PMAC AUTOMATIC READ FEATURE

Just like the standard Turbo PMAC2, the 3U Turbo PMAC allows the user to use automatic copy feature to simplify the reading of the A/D converted data. Using this method to read the data is the preferred method and allows the user to use the data for both data acquisition and closing servo loops. To enable the feature, I5060, I5061-I5076, and I5081-I5096 must be set as specified by the Turbo PMAC Software Reference.

The data from the ADC will come back to the PMAC memory address as a 24-bit word. The lower 12-bits will contain ADC0 through ADC7 and the upper 12-bits of this word will contain the data from channels ADC8 through ADC15. The data is automatically copied as follows:

Channel	Location	Channel	Location
ADC0	Y:\$3400,12,12	ADC8	Y:\$3401,12,12
ADC1	Y:\$3402,12,12	ADC9	Y:\$3403,12,12
ADC2	Y:\$3404,12,12	ADC10	Y:\$3405,12,12
ADC3	Y:\$3406,12,12	ADC11	Y:\$3407,12,12
ADC4	Y:\$3408,12,12	ADC12	Y:\$3409,12,12
ADC5	Y:\$340A,12,12	ADC13	Y:\$340B,12,12
ADC6	Y:\$340C,12,12	ADC14	Y:\$340D,12,12
ADC7	Y:\$340E,12,12	ADC15	Y:\$340F,12,12

For the PMAC2 option 12 or its use with the Acc-36P/V, I5060 controls the number of pairs of multiplexed A/D converters that are processed and de-multiplexed into individual registers. If I5060 is set to 0, none of these A/D converters is processed automatically.

If I5060 is set to a value greater than 0, it specifies the number of pairs of ADCs in the automatic processing ring. Each phase clock cycle, one pair is processed, and the values copied into image registers in RAM.

I5061 through I5076 control the addresses of the multiplexed A/D converters read in the A/D ring table, as enabled by I5060. These I-variables contain offsets from the starting Turbo PMAC address \$078800 where these ADCs can reside. If there is an Acc-36P with a base address of \$78A00, then the offset value placed into this register will be \$200 (\$78A00-\$78800).

I5081 through I5096 contain the convert codes written to the multiplexed A/D converters that are read in the A/D ring table, as enabled by I5060. The convert codes control which of the multiplexed ADCs at the address is to be read, and the range of the analog input for that ADC. The ADCs can be on-board the Turbo PMAC with Option 12 and 12A, or off-board with an Acc-36P/V.

I5081-I5096 are 24-bit values, represented by six hexadecimal digits. Legitimate values are of the format \$00m00n, where m and n can take any hex value from 0 through F.

For the Turbo PMAC2, the m value determines which of the inputs ANAI08 to ANAI15 is to be read, and how it is to be converted, according to the following formulas:

$$m = ANAI\#\ -8 \quad ; 0 \text{ to } +5\text{V unipolar input}$$

$$m = ANAI\# \quad ; -2.5\text{V to } +2.5\text{V bipolar input}$$

For the Option 12 with a Turbo PMAC2, the n value determines which of the inputs ANAI00 to ANAI07 and how it is to be converted, according to the following formulas:

$$n = ANAI\# \quad ; 0\text{V to } +5\text{V unipolar input}$$

$$n = ANAI\#\ +8 \quad ; -2.5\text{V to } +2.5\text{V bipolar input}$$

For example, to read ANAI02 and ANAI10, both as +/-2.5V inputs, into the first slot in the ring, m would be set to A (10) and n would be set to A (10), so I5081 would be set to \$00A00A.

Turbo PMAC Data Acquisition Example:

Setup the Turbo PMAC to read Acc-36P channels 0, 1, 2, 3, 4, 5, 6, and 7 as unipolar converted signals and read channels 8, 9, 10, 11, 12, 13, 14, and 15 as bi-polar signals. Assume Acc-36P is setup for base address \$78A00 (\$78800+\$200).

```

I5060=8 ;copy 8 ADC pairs
I5061=$000200 ;ADC0 and ADC8 are referenced to $078800 + $000200
I5062=$000200 ;ADC1 and ADC9 are referenced to $078800 + $000200
I5063=$000200 ;ADC2 and ADC10 are referenced to $078800 + $000200
I5064=$000200 ;ADC3 and ADC11 are referenced to $078800 + $000200
I5065=$000200 ;ADC4 and ADC12 are referenced to $078800 + $000200
I5066=$000200 ;ADC5 and ADC13 are referenced to $078800 + $000200
I5067=$000200 ;ADC6 and ADC14 are referenced to $078800 + $000200
I5068=$000200 ;ADC7 and ADC15 are referenced to $078800 + $000200
I5081=$008000 ;ADC0 unipolar ADC8 is bi-polar
I5082=$009001 ;ADC1 unipolar ADC9 is bi-polar
I5083=$00A002 ;ADC2 unipolar ADC10 is bi-polar
I5084=$00B003 ;ADC3 unipolar ADC11 is bi-polar
I5085=$00C004 ;ADC4 unipolar ADC12 is bi-polar
I5086=$00D005 ;ADC5 unipolar ADC13 is bi-polar
I5087=$00E006 ;ADC6 unipolar ADC14 is bi-polar
I5088=$00F007 ;ADC7 unipolar ADC15 is bi-polar

M5061->Y:$003400,12,12,u ;channel 0 A to D as unipolar
M5062->Y:$003402,12,12,u ;channel 1 A to D as unipolar
M5063->Y:$003404,12,12,u ;channel 2 A to D as unipolar
M5064->Y:$003406,12,12,u ;channel 3 A to D as unipolar
M5065->Y:$003408,12,12,u ;channel 4 A to D as unipolar
M5066->Y:$00340A,12,12,u ;channel 5 A to D as unipolar
M5067->Y:$00340C,12,12,u ;channel 6 A to D as unipolar
M5068->Y:$00340E,12,12,u ;channel 7 A to D as unipolar

M5081->Y:$003401,12,12,s ;channel 8 A to D as bipolar
M5082->Y:$003403,12,12,s ;channel 9 A to D as bipolar
M5083->Y:$003405,12,12,s ;channel 10 A to D as bipolar
M5084->Y:$003407,12,12,s ;channel 11 A to D as bipolar
M5085->Y:$003409,12,12,s ;channel 12 A to D as bipolar
M5086->Y:$00340B,12,12,s ;channel 13 A to D as bipolar
M5087->Y:$00340D,12,12,s ;channel 14 A to D as bipolar
M5088->Y:$00340F,12,12,s ;channel 15 A to D as bipolar

```

To start the automatic data transfer process the user must save and restart the Turbo PMAC and then simply read the m-variables associated with the ADC channel.

Acc-36P Servo Feedback Use Example for Turbo PMAC

To process the A/D information in the encoder conversion table you need to do the following. For this example we will be processing ADC0 from location Y:\$3400,12,12 in the ECT as a parallel unsigned entry.

Assuming the 9th entry of the encoder conversion table (ECT) is available.

```

I8008=$203400 ;read location Y:$3400 (ECT location $3509)
I8009=$00C00C ;read 12 bits shifted 12 bits from Y:$3400 (ECT
;location $350A)

```

The axis to be used is specified by x:

```

Set Ix03=$350A ;position feedback address
Set Ix04=$350A ;velocity feedback address

```

Acc-36P Power-On Position for Turbo PMAC

As of 3/27/2000, Delta Tau firmware does not support upper 12-bit word power on position reads. However, the user can still force the position register the appropriate power on value using the position offset register.

```

M164->D:$00CC           ;motor 1 offset position register (1/(32*Ix08))
M264->D:$014C           ;motor 2 offset position register (1/(32*Ix08))
M364->D:$01CC           ;motor 3 offset position register (1/(32*Ix08))
M464->D:$024C           ;motor 4 offset position register (1/(32*Ix08))
M564->D:$02CC           ;motor 5 offset position register (1/(32*Ix08))
M664->D:$034C           ;motor 6 offset position register (1/(32*Ix08))
M764->D:$03CC           ;motor 7 offset position register (1/(32*Ix08))
M864->D:$044C           ;motor 8 offset position register (1/(32*Ix08))

M5061->Y:$003400,12,12,u ;channel 0 A to D as unipolar
M5062->Y:$003402,12,12,u ;channel 1 A to D as unipolar
M5063->Y:$003404,12,12,u ;channel 2 A to D as unipolar
M5064->Y:$003406,12,12,u ;channel 3 A to D as unipolar
M5065->Y:$003408,12,12,u ;channel 4 A to D as unipolar
M5066->Y:$00340A,12,12,u ;channel 5 A to D as unipolar
M5067->Y:$00340C,12,12,u ;channel 6 A to D as unipolar
M5068->Y:$00340E,12,12,u ;channel 7 A to D as unipolar

```

A PLC could be written to read the ADC into the position offset registers at power up:

```

OPEN PLC 25 CLEAR
I5111=1000*8388608/i10   ;1000 msec delay to ensure data is read properly
While (i5111>0) endwhile
M164=m5061*32*i108      ;set power on position offset to m5061 for mtr1
M264=m5062*32*i108      ;set power on position offset to m5062 for mtr2
M364=m5063*32*i108      ;set power on position offset to m5063 for mtr3
M464=m5064*32*i108      ;set power on position offset to m5064 for mtr4
M564=m5065*32*i108      ;set power on position offset to m5065 for mtr5
M664=m5066*32*i108      ;set power on position offset to m5066 for mtr6
M764=m5067*32*i108      ;set power on position offset to m5067 for mtr7
M864=m5068*32*i108      ;set power on position offset to m5068 for mtr8
Disable plc25
clear

```


ACC-36 PINOUTS

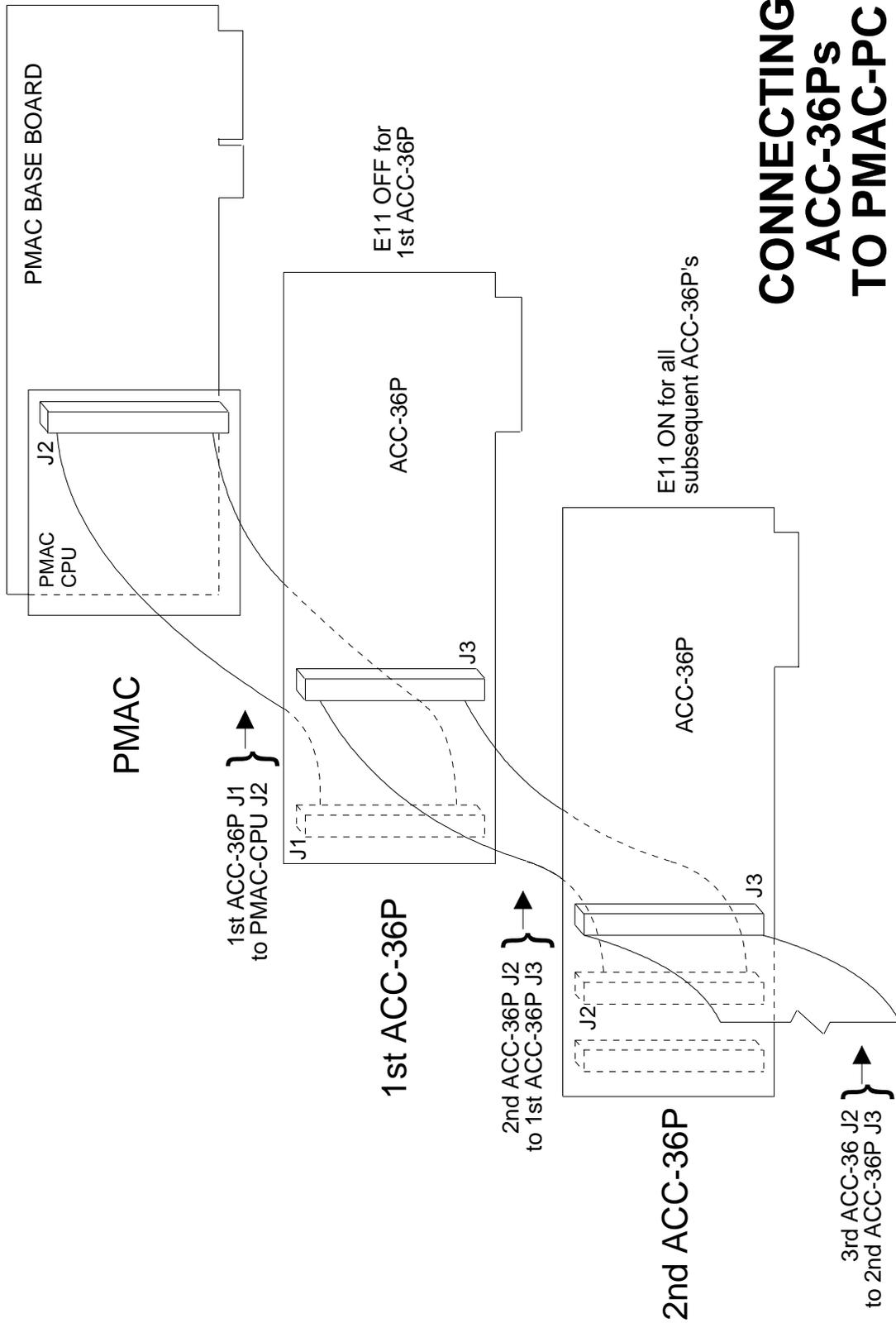
TB1 (4-pin Terminal Block)

Pin #	Symbol	Function	Description
1	GND	Common	Digital Ground
2	+5V	Input	External Supply
3	+12V	Input	External Supply
4	-12V	Input	External Supply

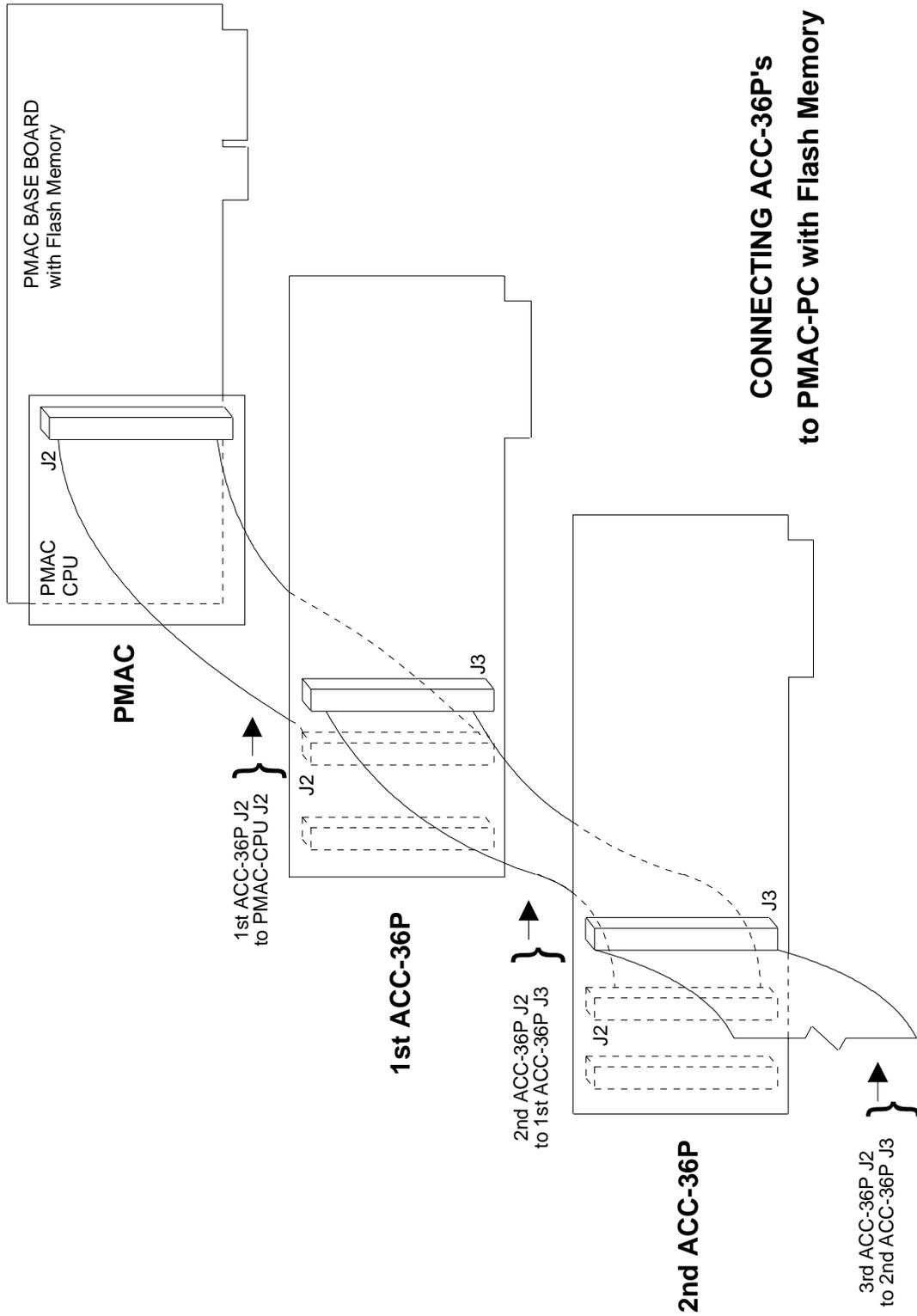
J4 (JPAN DB37)

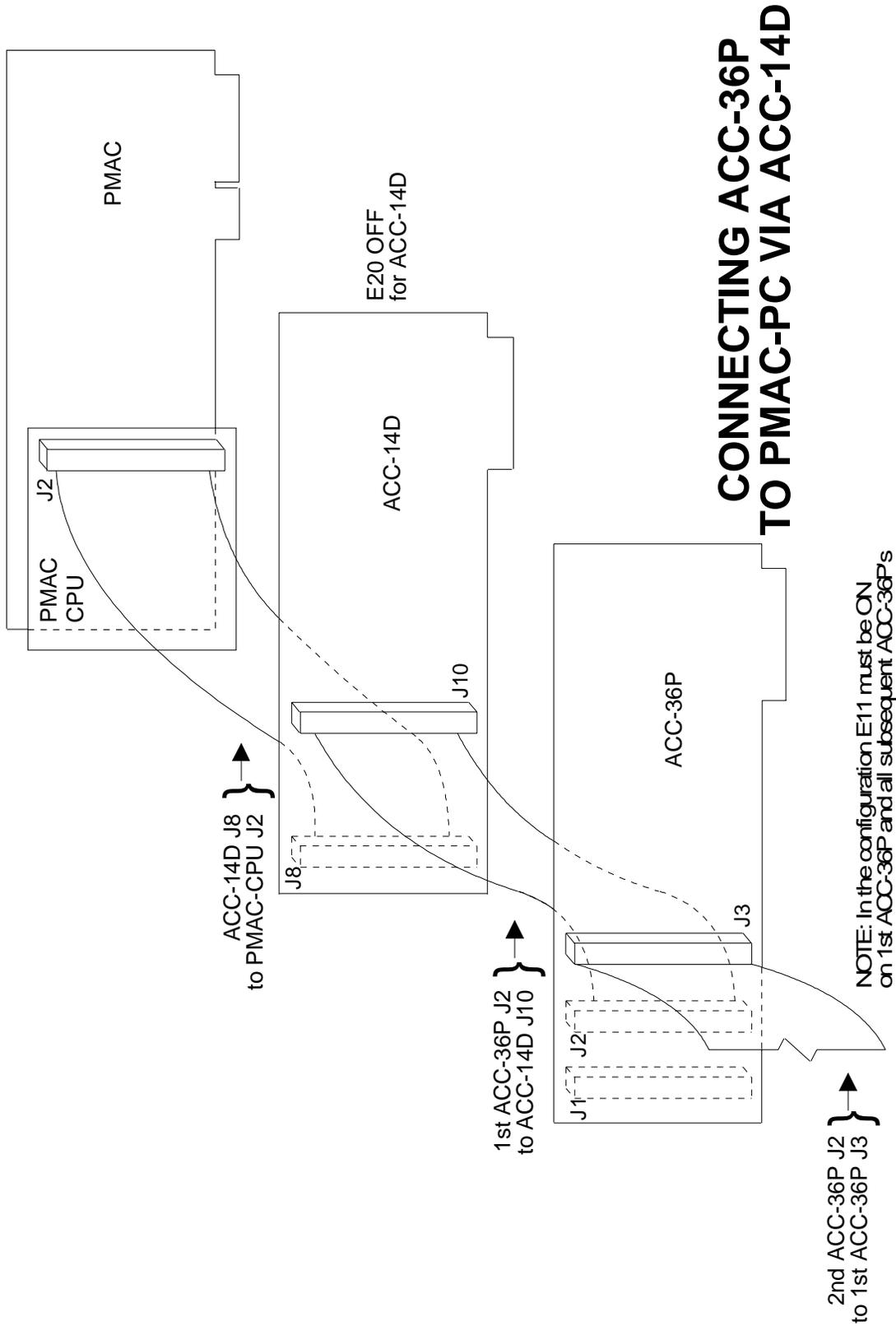
Pin #	Symbol	Function	Description
1	ADC1	Input	Analog Input #1
2	ADC2	Input	Analog Input #2
3	ADC3	Input	Analog Input #3
4	ADC4	Input	Analog Input #4
5	ADC5	Input	Analog Input #5
6	ADC6	Input	Analog Input #6
7	ADC7	Input	Analog Input #7
8	ADC8	Input	Analog Input #8
9	ADC9	Input	Analog Input #9
10	ADC10	Input	Analog Input #10
11	ADC11	Input	Analog Input #11
12	ADC12	Input	Analog Input #12
13	ADC13	Input	Analog Input #13
14	ADC14	Input	Analog Input #14
15	ADC15	Input	Analog Input #15
16	ADC16	Input	Analog Input #16
17	AGND	Common	Ground*
18	+12V	Output	Positive Supply**
19	AGND	Common	Ground
20	ADC1/	Input	Analog Input/ #1
21	ADC2/	Input	Analog Input/ #2
22	ADC3/	Input	Analog Input/ #3
23	ADC4/	Input	Analog Input/ #4
24	ADC5/	Input	Analog Input/ #5
25	ADC6/	Input	Analog Input/ #6
26	ADC7/	Input	Analog Input/ #7
27	ADC8/	Input	Analog Input/ #8
28	ADC9/	Input	Analog Input/ #9
29	ADC10/	Input	Analog Input/ #10
30	ADC11/	Input	Analog Input/ #11
31	ADC12/	Input	Analog Input/ #12
32	ADC13/	Input	Analog Input/ #13
33	ADC14/	Input	Analog Input/ #14
34	ADC15/	Input	Analog Input/ #15
35	ADC16/	Input	Analog Input/ #16
36	AGND	Common	Ground*
37	-12V	Output	Negative Supply**

* This common point is connected to the digital ground of the PMAC board.
**The supply voltages are for output from the board to supply the sensors connected to Acc-36P. The drawn current should not exceed 0.5 A.



CONNECTING ACC-36Ps TO PMAC-PC





APPENDIX A

I60: Auto-Converted ADC Register Address Range

Range: 0, \$FFD0 .. \$FFFE
Default: 0
Units: PMAC Y addresses

This parameter permits the user to specify the address of one Acc-36 analog-to-digital converter (ADC) board whose values will be copied automatically into PMAC's memory at a high rate so that they can be used as servo feedback. This can also be used to make user program access to these ADCs more convenient, but it is not required for this purpose.

There are 24 legal addresses for an Acc-36 in PMAC's memory and I/O space: even values from \$FFD0 to \$FFFE. If there is more than one Acc-36 connected to PMAC, only one board may be used in this manner. All other boards must be accessed in user programs. For the Acc-36 board automatically converted using I60 and I61, the board must never be accessed in user programs, but user programs may read the memory registers in PMAC to which the ADC values are copied.

If I60 is set to 0, no automatic conversion will take place. If the first two hex digits of I60 are set to anything except \$FF, PMAC will change them to \$FF automatically.

ADCs 1 to 8 are copied into the low twelve bits of registers Y:\$0708 to Y:\$070F, respectively. ADCs 9 to 16, if they exist on the addressed board, are copied into the low 12 bits of registers X:\$0708 to X:\$070F. These registers should be treated as signed registers.

Note:

It is easier to specify this parameter in hexadecimal form (\$ prefix). If I9 is set to 2 or 3, the value of this variable will be reported back to the host in hexadecimal form.

A PMAC system has an Acc-14D at address \$FFD0, and an Acc-36 at address \$FFD8. To convert all eight registers on the Acc-36 automatically, I60 is set to \$FFD8, and I61 is set to 7.

See Also:

Parallel Position Feedback Conversion (Setting Up a Motor)

I-variables I61, Ix10

Memory and I/O Map registers \$FFD0 to \$FFFE

Acc-36 User Manual

I61: Number of Auto-Converted ADC Registers

Range: 0 .. 7
Default: 0
Units: Number of registers minus 1

This parameter permits the user to specify the number of analog-to-digital converter (ADC) registers on the Acc-36 specified by I60 that will be converted automatically and copied into PMAC memory. On the base version of the Acc-36, there is one 12-bit converter per register; with Option 1, there are two 12-bit converters per register. The number of registers converted automatically is equal to I61 + 1.

Each phase cycle (9 kHz default), PMAC copies the contents of an Acc-36 register into RAM, then selects the next register, so the conversion can start and the results will be ready for the next phase cycle. PMAC will cycle through the first I61+1 registers on the Acc-36 in this fashion. If I61 is set to 0, PMAC will cycle through all eight registers on the Acc-36 (equivalent to I61=7).

If there is more than one Acc-36 connected to PMAC, only one board may be used in this manner. All other boards must be accessed in user programs. For the Acc-36 board automatically converted using I60 and I61, the board must never be accessed in user programs, but user programs may read the memory registers in PMAC to which the ADC values are copied.

ADCs 1 to 8 are copied into the low twelve bits of registers Y:\$0708 to Y:\$070F, respectively. ADCs 9 to 16, if they exist on the addressed board, are copied into the low twelve bits of registers X:\$0708 to X:\$070F. These registers should be treated as signed values.

Example:

The system has eight axes with analog feedback. There are 4 phase cycles per servo cycle, and it is important to have new feedback values every servo cycle. Therefore an Acc-36 with Option 1 is ordered, so there are two ADCs per register, and I61 is set to 3 to convert the first four registers in a cyclic fashion. ADCs 1 to 4 are copied into Y:\$0708 to Y:\$070B, respectively; ADCs 9 to 12 are copied into X:\$0708 to X:\$070B, respectively.

See Also:

Parallel Position Feedback Conversion (Setting Up a Motor)

I-variables I60, Ix10

Memory and I/O Map registers \$FFD0 to \$FFFE

Acc-36 User Manual



Artisan Technology Group is your source for quality new and certified-used/pre-owned equipment

- FAST SHIPPING AND DELIVERY
- TENS OF THOUSANDS OF IN-STOCK ITEMS
- EQUIPMENT DEMOS
- HUNDREDS OF MANUFACTURERS SUPPORTED
- LEASING/MONTHLY RENTALS
- ITAR CERTIFIED SECURE ASSET SOLUTIONS

SERVICE CENTER REPAIRS

Experienced engineers and technicians on staff at our full-service, in-house repair center

*InstraView*SM REMOTE INSPECTION

Remotely inspect equipment before purchasing with our interactive website at www.instraview.com ↗

WE BUY USED EQUIPMENT

Sell your excess, underutilized, and idle used equipment. We also offer credit for buy-backs and trade-ins. www.artisanng.com/WeBuyEquipment ↗

LOOKING FOR MORE INFORMATION?

Visit us on the web at www.artisanng.com ↗ for more information on price quotations, drivers, technical specifications, manuals, and documentation

Contact us: (888) 88-SOURCE | sales@artisanng.com | www.artisanng.com