USER MANUAL

16-Axis MACRO CPU

Remote MACRO Interface

3xx-603719-xUxx

February 1, 2005

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

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

The 16-Axis MACRO CPU provides a remote interface for encoders, flags, direct-PWM digital drives, analog drives, and/or digital I/O for a Turbo PMAC2 with MACRO interface. It communicates with the Turbo PMAC2 solely through the MACRO ring, but interfaces to standard drives, encoders, flags, and Opto-22 style I/O through on-board connectors. It is designed to run up to sixteen motors.

With the fiber optic MACRO interface, the 16-Axis MACRO CPU can be up to three kilometers (two miles) away from the Turbo PMAC2 controller or any other station on the ring. With the RJ-45 electrical interface, it can be up to 30 meters (100 feet) away.

With the 16-Axis MACRO CPU, the Turbo PMAC2 can control servo axes and I/O just as if they were connected directly to the Turbo PMAC2, even though they are a great distance away and the only interface from the Turbo PMAC2 is the MACRO ring.

This manual explains the setup of the 16-Axis MACRO CPU. It should be used in conjunction with the Hardware Reference manuals for the 3U MACRO CPU 16x board, the 3U-format accessories that are used, and the Software Reference Manual for the 16x MACRO Station.

16-Axis MACRO Station Differences from the 8-Axis MACRO Station

The new 16-Axis MACRO CPU has two MACRO ICs (0 and 1) and each MACRO IC can support servo control for two Servo ICs each (eight motors). It can be thought of as two of 3U 8-Axis MACRO Stations. Each UMAC MACRO IC is bound to separate MACRO Masters. Each MACRO Master addresses its own MACRO IC Slave and the Servo IC MInn variables on the Ring with the **MS<node>** commands. Each MACRO IC and Servo IC has their own setup MI variables MI900 – 969 and MI990 – 999. MACRO IC 0's MI996 (the ring binding MI variable) is determined by SW1 (Nodes enabled) and SW2 (Master #) or setup by the Ring Order Who are you? software configuring. MACRO IC 1 defaults to a binding of Master number equal to SW2+1 and node 11 enabled.

The 16-Axis MACRO CPU (DSP56309) is run at 100 MHz where as the 8-Axis MACRO CPU (DSP56303) is run at 60 MHz. The new CPU has additional program memory and now has the feature of running a simplified PLCC program locally in the station. This PLCC program has available 512 PMAC M and P-type variables called MM and MP variables. With the PLCC program, simple integer arithmetic and logic can be performed locally in the station. This feature comes standard with PEWINPro.

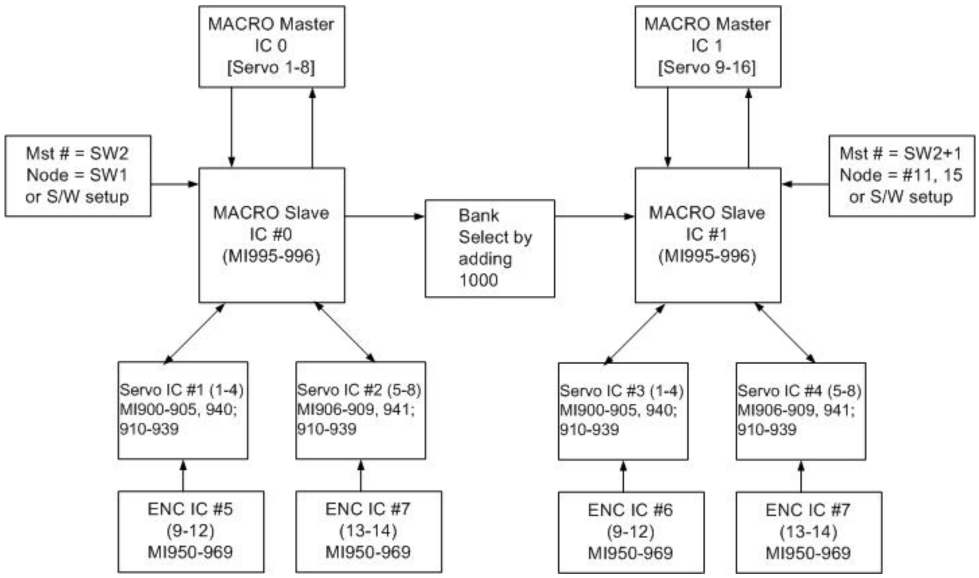
Another new feature includes Turbo PMAC type I4900 (MI200) variables for determining all UMAC cards in the UMAC rack. Also added in the MI200 variables, are measurements of phase and background times for duty-cycle measurements.

Hardware additions include two handwheel input channels and the standard PMAC type display output.

16-Axis MACRO Slave Station Binding to a MACRO Master

1. MACRO IC 0's MI996 binding setup rules are the same as the 8-Axis station where SW1 enables certain servo nodes and SW2 determines the master number. For software setup (SW1=14): Use Ring Order Who Are You? to setup MACRO IC 0's MI996.
2. MACRO IC 1's MI996 binding setup rules are the following:
   a. For modes where SW1 and SW2 are used to bind a MACRO Slave Station to a Master, use the value of SW2 +1 to set the Master number and enable node 11,15 in MACRO IC 1's MI996 on a **$$$***** initialization. Then access MACRO IC 1's MI-Variables from its Master's using node 11 (MS27, MI996, etc. type commands).
   b. For a mode using software setup (SW1=14): Use Ring Order Who Are You? to setup MACRO IC 0's MI996 and then add 1000 to the MInn variables to select MACRO IC 1's MI variables for setup. For example, MI996 accesses MACRO IC 0 and MI1996 accesses MACRO IC 1. The same for MI992 and MI1992.

Once MI996 of MACRO IC 1 is set, its associated MInn variables can be accessed from its Turbo PMAC MACRO Master with MS type commands.

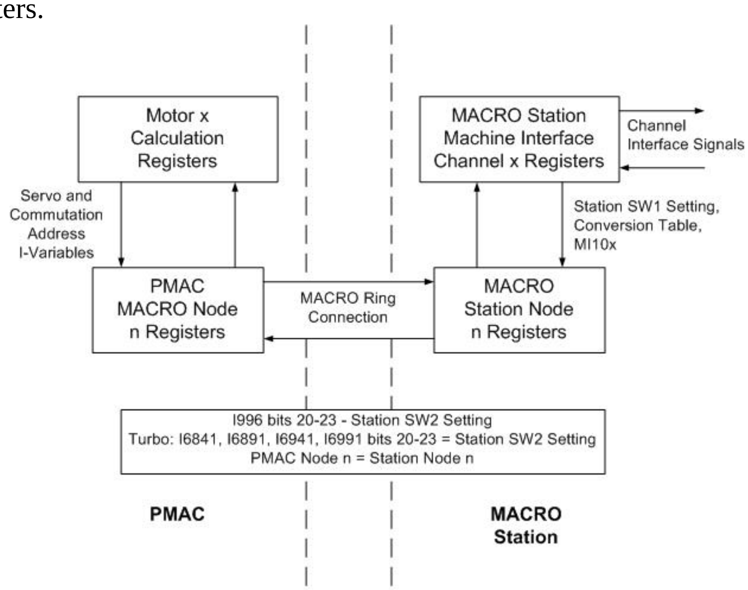


16-Axis MACRO CPU Setup Overview

Fundamentally, the setup of the 16-Axis MACRO CPU with a Turbo PMAC2 (usually an Ultralite version) involves several steps of mapping registers and connections.

For the axis control, the following mappings must occur:

1. The first mapping is the connection of physical devices (encoders, drives, and flags) to a particular machine channel on the MACRO 16x Station.
2. The second mapping is between the machine interface channel and a MACRO servo nodes on the 16x Station.
3. The third mapping is between the MACRO servo node of the 16x Station and a MACRO motor node on the Turbo PMAC2.
4. The fourth mapping is between the MACRO servo node on the Turbo PMAC2 and the motor calculation registers.



Once the basic mapping is set up, the operation of the MACRO ring becomes essentially invisible to the actual operation of the system and the system operates just as if devices were interfaced directly to the MACRO controller.

Mapping Servo Channels

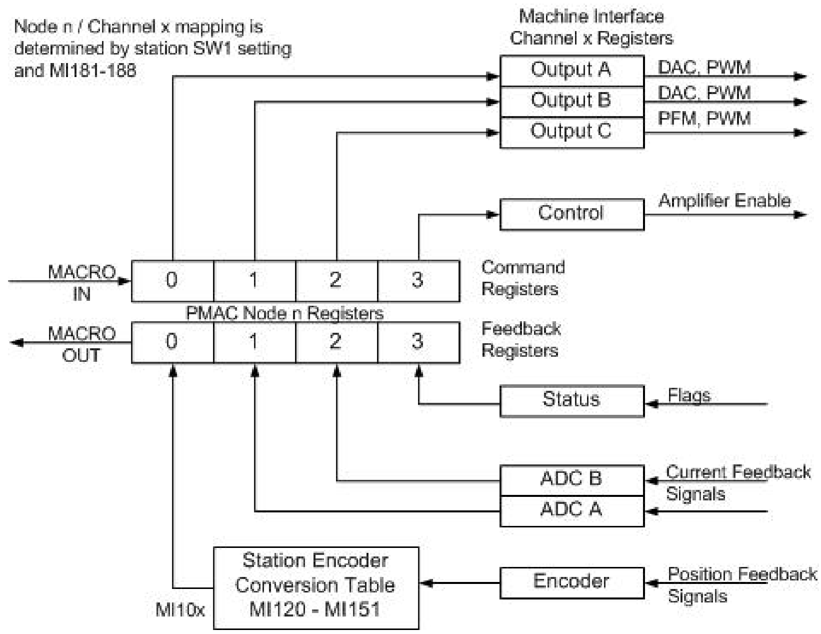
Mapping Physical Servo Channels to the 16-Axis MACRO Station's Servo Channels:
MI179 and MI180 provide the base addresses of the two Servo ICs attached to the MACRO IC and Y:MI181 – Y:MI188 provide the base address for the eight servo channels. Station servo node-specific variables MI910 - MI939 use Y:MI18n to configure the Servo IC's interface channels associated with specified MACRO node determined by SW1. Each MACRO IC has its own unique set of these MI variables.

UMAC (pack) cards provide breakout connectors to wire the physical machine interface to the 16-Axis servo channels. The dip switch S1 determines the base address of the UMAC pack cards like Acc-24E2A. This address will show up in MI179 and MI180.

Mapping Servo Channels to Servo Node

Mapping the 16-Axis MACRO Station's servo channels to the Station's MACRO servo node registers is set up automatically and cannot be changed.

- The mapping addresses are in MI181 – MI188. The X: part (upper six hex characters) is the MACRO node command/status flag address and the Y: part (lower six hex characters) is the Servo IC's status register address. Each MACRO IC has its own unique set of these MI variables.
- Station Encoder conversion table (MI120-MI151) and Motor x variables MI10x control mapping of feedback position from machine interface channels to Station MACRO servo nodes. Each MACRO IC has its own unique set of these MI variables.
- Station Motor x variables MI11x control mapping of power-on absolute feedback position to Station MACRO servo nodes. Each MACRO IC has its own unique set of these MI variables.

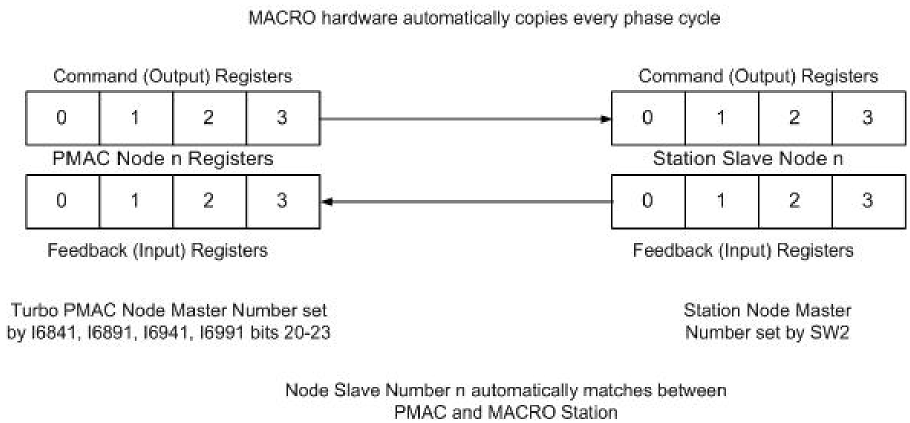


Mapping Motor Node Registers

Map the 16-Axis MACRO CPU's MACRO motor node registers to the Turbo PMAC2's MACRO motor node registers:

- Connection of the Turbo PMAC2 MACRO master and MACRO Slave Station in a common ring
- Turbo PMAC2 MACRO cycle frequency control
  - Turbo PMAC2 Ultralite I6800, I6801
- Turbo PMAC2 I6840, I6890, I6940, I6990
- Turbo PMAC2 MACRO Node Activation Control and Master number I6841, I6891, I6941, I6991

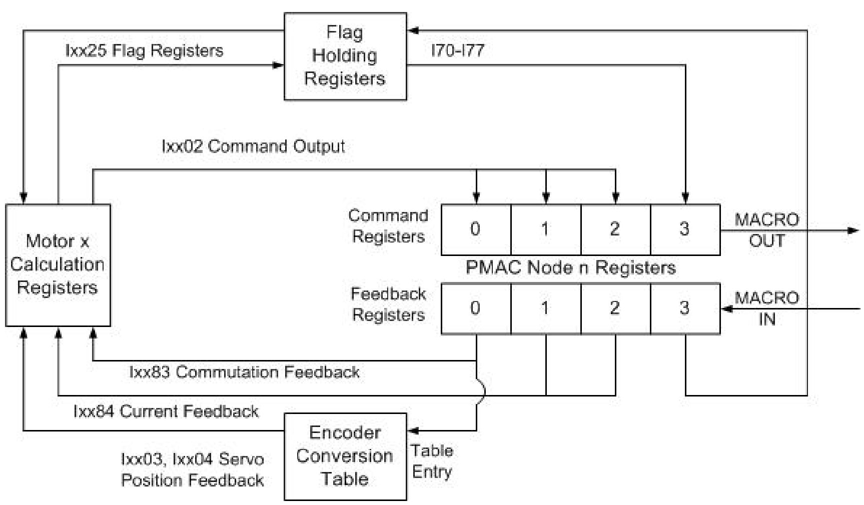
- Turbo PMAC2 MACRO Configuration I-variables
  - MACRO Node Flag Register Enable [I70, I72, I74, I76]
  - MACRO Node Flag Type Control [I71, I73, I75, I77]
  - MACRO Ring Check Period [I80]
  - MACRO Maximum Ring Error Count [I81]
  - MACRO Minimum Sync Packet Count [I82]
    - MACRO Master/Slave Auxiliary Communication Timeout [I78]
    - MACRO Type 1 Master/Master and Ring Order Communication Timeout [I79]
- Station SW2 setting for Master number
- Station SW1 setting and MI976 setting for active MACRO IC 0 servo nodes
- Station MACRO cycle frequency control with MI992 and MI997 for both MACRO ICs.



Mapping Motor Function Registers to Node Registers

Map the Turbo PMAC2 motor function registers to the Turbo PMAC2 MACRO node registers:

- Encoder Conversion Table Setup Addresses [I8000 – I8191]
- Ixx02 Command Output Address
- Ixx03 Position-Loop Feedback Address
- Ixx04 Velocity-Loop Feedback Address
- Ixx10, Ixx95 Power-On Position Feedback Address
- Ixx24, Ixx25 Flag Address
- Ixx81, Ixx91 Power-On Phase Feedback Address
- Ixx82 Current-Loop Feedback Address
- Ixx83 Phase Position Feedback Address



Mapping of General Purpose I/O

General-purpose I/O is processed through a similar set of mapping functions. Once the setup of the mapping has been done, PMAC software can access the I/O points on the 16-Axis MACRO CPU as if they were on the PMAC itself.

- Mapping physical devices to the 16-Axis MACRO CPU I/O circuitry
  - Wiring between the I/O connectors and the devices
- Mapping the Station I/O registers to 16-Axis MACRO CPU MACRO I/O nodes
  - Station MI19 setting for frequency of data copying
  - Station MI69 – MI71, MI169 – MI175 settings for matching MACRO I/O nodes to accessory boards with IOGATE ICs: Acc-9E – 12E, 14E UMAC I/O boards.
  - Station MI20-MI68 settings for matching MACRO I/O nodes to other I/O circuitry, e.g. Acc-36E/59E ADC/DAC boards
- Mapping 16-Axis MACRO CPU I/O nodes to PMAC MACRO I/O nodes
  - Connection of the PMAC and MACRO Station in a common ring
  - I6841, I6891, I6941, I6991 MACRO Node Activation Control of MACRO I/O Nodes
  - Station SW2 setting for Master number
  - Station MI975 setting for active MACRO I/O nodes (MI975 available on MACRO IC0 only)
- Mapping PMAC MACRO I/O nodes to PMAC M-variables
  - M-variable definitions to images of I/O in PMAC memory
  - M-variable definitions to MACRO node registers (entire register only)
  - PMAC commands (usually in PLC) to copy between image registers and MACRO I/O nodes

HARDWARE SETUP

The hardware setup of the 16-Axis MACRO CPU for the UMAC MACRO 16x is covered in the Hardware Reference manual for the 3U 16-Axis MACRO CPU Board and the manuals for each of the individual accessory boards in the station. A brief summary is given here.

The electronic hardware of a 16-Axis MACRO CPU consists of a 3U 16-Axis MACRO CPU Interface board that contains the MACRO link to the ring and the processor that governs the operation of the Station, plus some combination of axis interface boards and I/O interface boards.

Physical and Logical Configuration of the MACRO Station

This section briefly describes how the boards in a MACRO Station interface together and how they communicate – what addresses they occupy in the address space of the MACRO CPU. More details are given in the manuals for each specific board.

UMAC (Pack) Configuration

In the UMAC (Pack) configuration, the Axis-interface boards and the I/O-interface boards communicate to the MACRO CPU board via an Acc-U*x* Ubus backplane board. Each board can slide into a standard 3U rack with 4T (20mm, 0.8”) spacing between boards and connect physically to the backplane board.

Servo Accessory Boards

For servo interface, the 16-Axis MACRO CPU board can address up to four servo interface/breakout accessory boards on the UMAC backplane. The boards in this family that are presently available include:

- Acc-24E2 2/4-channel PWM servo interface/breakout board
- Acc-24E2A 2/4-channel analog servo interface/breakout board
- Acc-24E2S 4-channel stepper/encoder interface/breakout board
- Acc-51E high-resolution encoder-interpolator board

Note:

Option 1A or Option 1D on the Acc-24E2 or Acc-24E2A, while it adds an extra physical slot, does not count as an extra accessory board for addressing purposes.

The addresses and channel numbers on these boards are set by Dip switch S1.

S1-1	S1-2	S1-3	S1-4	S1-5	S1-6	Servo IC #	Board Base Address
ON	ON	ON	ON	ON	ON	1	$8000
OFF	ON	ON	ON	ON	ON	2	$8040
ON	ON	OFF	ON	ON	ON	3	$9000
OFF	ON	OFF	ON	ON	ON	4	$9040
ON	ON	ON	OFF	ON	ON	5	$A000
OFF	ON	ON	OFF	ON	ON	6	$A040
ON	ON	OFF	OFF	ON	ON	7	$B000
OFF	ON	OFF	OFF	ON	ON	8	$B040

I/O Accessory Boards

For I/O interface, the MACRO CPU board can address accessory boards at four different addresses on the backplane. The addresses on these boards are set by jumpers on some accessory boards and Dip switches on other boards.

The I/O boards whose addresses are set by jumpers are:

- Acc-9E isolated 48 input board
- Acc-10E isolated 48 output board
- Acc-11E isolated 24 In/24 out board
- Acc-12E isolated 24 In/24 high-power out board

For these boards, the jumper settings and the board addresses they select are:

Address Jumper On	Board Base Address
E1 (CS10)	$8800, $9800, $A800,$B800
E2 (CS12)	$8840, $9840, $A840,$B840
E3 (CS14)	$8880, $9880, $A880,$B880
E4 (CS16)	$88C0, $98C0, $A8C0,$B8C0

For these boards, up to three boards can share an address because each board only occupies one byte (eight bits) of the 24-bit data bus and each board can be set up as to which byte it occupies:

E6A- H Rows Connected	Byte Used on Data Bus
1 and 2	Low (Bits 0 – 7)
2 and 3	Middle (Bits 8 – 15)
3 and 4	Middle (Bits 8 – 15)
4 and 5	High (Bits 16 – 23)

The I/O boards available presently whose addresses are set by DIP-switches are:

- Acc-14E 48-TTL-I/O board
- Acc-28E 2/4-channel 16-bit ADC board
- Acc-36E 16-channel 12-bit ADC board
- Acc-53E SSI encoder interface board
- Acc-59E 8-channel 12-bit ADC/8-channel 12-bit DAC board
- Acc-65E self-protected sourcing 24 in/24 out board
- Acc-66E self-protected sourcing 48 input board
- Acc-67E self-protected sourcing 48 output board
- Acc-68E self-protected sinking 24 in/24 out board

For these boards, the switch settings and the board addresses they select are:

S1-1	S1-2	S1-3	S1-4	S1-5	S1-6	I/O Cards	Board Base Address
ON	ON	ON	ON	ON	ON	1	$8800-$883F
OFF	ON	ON	ON	ON	ON	2	$9800-$983F
ON	OFF	ON	ON	ON	ON	3	$A800-$A83F
OFF	OFF	ON	ON	ON	ON	4	$B800-$B83F
ON	ON	OFF	ON	ON	ON	5	$8840-$887F
OFF	ON	OFF	ON	ON	ON	6	$9840-$987F
ON	OFF	OFF	ON	ON	ON	7	$A840-$A87F
OFF	OFF	OFF	ON	ON	ON	8	$B840-$B87F
ON	ON	ON	OFF	ON	ON	9	$8880-$88BF
OFF	ON	ON	OFF	ON	ON	10	$9880-$98BF
ON	OFF	ON	OFF	ON	ON	11	$A880-$A8BF
OFF	OFF	ON	OFF	ON	ON	12	$B880-$B8BF
ON	ON	OFF	OFF	ON	ON	13	$88C0-$88C7
OFF	ON	OFF	OFF	ON	ON	14	$98C0-$98C7
ON	OFF	OFF	OFF	ON	ON	15	$A8C0-$A8C7
OFF	OFF	OFF	OFF	ON	ON	16	$B8C0-$B8C7

Generally, with these boards it is possible to put only one board at any given address. (The Acc-14E board always occupies the low byte only of the data bus, so it is possible to put Acc-9E, 10E, 11E, or 12E boards at the same address in the middle or high bytes.)

Auto Configuration and Identification of UMAC (Pack) Boards

The 16-Axis MACRO CPU identifies all the UMAC Servo IC type boards in the Ubus backplane automatically. From this list, it then attaches them to the two MACRO ICs. Each MACRO IC will support and configure eight servo channels and six encoder feedback channels.

BITn	Fault/Status Description (MI4)
15	Detected Ubus Encoder IC #7 Attached to MACRO IC 0 and 1 (2 channels each)
16	Detected Ubus Encoder #6 Attached to MACRO IC 1 (4 encoder channels)
17	Detected Ubus Encoder #5 Attached to MACRO IC 0 (4 encoder channels)
18	Detected Ubus Servo IC #4 Attached to MACRO IC 1 (4 full servo channels)
19	Detected Ubus Servo IC #3 Attached to MACRO IC 1 (4 full servo channels)
20	Detected Ubus Servo IC #2 Attached to MACRO IC 0 (4 full servo channels)
21	Detected Ubus Servo IC #1 Attached to MACRO IC 0 (4 full servo channels)
22	Detected CPU MACRO IC 1 ($C0C0)
23	Detected CPU MACRO IC 0 ($C080)

Wiring into the MACRO Station

The connections detailed in the Hardware Reference manuals establish the first mapping required between the physical devices and the machine interface channels on the MACRO Station. For the UMAC pack, the interface circuitry and breakout connectors are on the same rack-mounted boards, so the field wiring is made directly into these boards.

UMAC (Pack) Interface/Breakout Boards

The UMAC boards available presently that interface to the MACRO CPU board through the UBUS backplane include:

- Acc-24E2 2/4-channel PWM servo interface/breakout board
- Acc-24E2A 2/4-channel analog servo interface/breakout board
- Acc-24E2S 4-channel stepper/encoder interface/breakout board
- Acc-51E high-resolution encoder-interpolator board
- Acc-9E isolated 48-input board
- Acc-10E isolated 48-output board
- Acc-11E isolated 24 in/24 out board
- Acc-12E isolated 24 in/24 high-power out board
- Acc-14E 48-TTL output board
- Acc-28E 2/4-channel 16-bit ADC board
- Acc-36E 16-channel 12-bit ADC board (V1.115 or newer firmware required)
- Acc-53E SSI encoder interface board
- Acc-59E 8-channel 12-bit ADC/8-channel 12-bit DAC board (V1.115 or newer firmware required)
- Acc-65E self-protected sourcing 24 in/24 out board
- Acc-66E self-protected sourcing 48 input board
- Acc-67E self-protected sourcing 48 output board
- Acc-68E self-protected sinking 24 in/24 out board

All of these boards provide their own breakout connectors, so no additional breakout boards are required for the field wiring. Consult the manual for each of these accessory boards for detailed pinout information.

SW1 Setting

SW1 establishes how many servo nodes and which servo nodes, will be used on the 16-Axis MACRO CPU station for MACRO IC 0. It also establishes the mapping of MACRO node numbers to MACRO Station channel numbers – the second mapping step explained in the overview. This mapping information will be important in establishing the software setup. The following table shows the possible configurations and the SW1 settings to achieve them.

MACRO IC 0 ($$$ or $$$*)**

# I/O Nodes	# of Servo Channels and Nodes Used	Node Servo IC Base Address	Nodes Enabled	SW1 Setting	Y:MI18 MI188
0	4	$8000,$8008, $8010,$8018	0,1, 4,5	0	I181, I182, I183, I184
0	4	$8000,$8008, $8010,$8018	8,9, 12,13	1	I185, I186, I187, I188
0	2	$8000,$8008	0,1	2	I181, I182
0	2	$8000,$8008	4,5	3	I183, I184
0	2	$8000,$8008	8,9	4	I185, I186
0	2	$8000,$8008	12,13	5	I187, I188
0	6	$8000,$8008, $8010,$8018, $8040,$8048	0,1, 4,5, 8,9	6	I181, I182, I183, I184, I185, I186
0	8	$8000,$8008, $8010,$8018, $8040,$8058, $8050,$8058	0,1, 4,5, 8,9, 12,13	7	I181, I182, I183, I184, I185, I186, I187, I188
2	0		2,3	8	
2	0		6,7	9	
2	0		10,11	10	
4	0		2,3,6,7	11	
6	0		2,3,6,7,10,11	12	
1	0		11	13	
0	0		None	14	Ring Order
1	0		11	15	Performs a $$$***

SW2 Setting

The setting of rotary switch SW2 on the 16-Axis MACRO CPU board determines the number (0 to 15) of the master for MACRO IC 0. The master number for MACRO IC 1 it is SW1+1. That same master number on a Turbo PMAC2 will be the one exchanging data with one on the 16-Axis MACRO CPU. This establishes the third mapping step explained in the overview.

The Turbo PMAC2 can support up to four active MACRO ICs. The master numbers for these ICs are set by I6841, I6891, I6941, and I6991. This setup is covered in the next section.

Note:

The master number of a MACRO IC on a Turbo PMAC2 is not necessarily the same as the MACRO IC number (0, 1, 2, or 3) itself. However, if there is only a single Turbo PMAC2 on the ring, it is probable that each MACRO IC on the Turbo PMAC2 will be assigned a master number equal to the IC number.

TURBO PMAC2 SOFTWARE SETUP FOR MACRO STATION

Setting up the Turbo PMAC2 board to work with a MACRO Station requires the proper setup of several I-Variables for MACRO-specific features. The variables that have special considerations for use with MACRO stations are listed below.

Note:

These are I-Variables on the Turbo PMAC2 controller itself. The MACRO Station has its own set of setup I- Variables, called MI-Variables, which are detailed in a different section.

Typically, the Turbo Setup program for PCs is used to set up these I-Variables.

MACRO IC Address Specification

Turbo PMAC2 firmware provides automatic support for up to four MACRO ICs at one time, known as MACRO ICs 0, 1, 2, and 3. Prior to firmware revision V1.936, each of these four ICs had a fixed base address: $078400 for MACRO IC 0, $079400 for MACRO IC 1, $07A400 for MACRO IC 2, and $07B400 for MACRO IC 3.

Turbo PMAC2 boards without a built-in MACRO interface support only MACRO IC 0 at the fixed address of $078400. Turbo PMAC2 Ultralite boards can also support MACRO ICs 1, 2, and 3 if the appropriate options are ordered:

- Option 1U1: MACRO IC 1 at $079400
- Option 1U2: MACRO IC 2 at $07A400
- Option 1U3: MACRO IC 3 at $07B400

The introduction of the UMAC Turbo (3U Turbo PMAC2) allowed more possibilities for addressing MACRO ICs which requires a more flexible firmware structure. Therefore, starting in Turbo PMAC2 firmware revision V1.936, variables I20, I21, I22, and I23 are used to specify the base addresses of MACRO ICs 0, 1, 2, and 3, respectively. Usually, these will be at the default values of $078400, $079400, $07A400, and 07B400, but other values are possible in a UMAC Turbo system with multiple Acc-5E MACRO interface boards.

MACRO Ring Update Frequency Setup

The discussions of MACRO node addresses below all assume that I20, I21, I22, and I23 are set to their factory default values.

All stations on the MACRO ring must be set to the same ring update frequency. The ring update frequency is controlled fundamentally by the ring controller or synchronizing master. The ring update frequency is the same as the hardware phase clock frequency on the card. The synchronizing master initiates the start of the MACRO ring cycle. When it has finished sending its data, the control of the ring is passed to the next non-synchronizing master. This is done until there are no non-synchronizing masters. Then the ring goes quiet with no data being sent. Each MACRO slave exchanges data with its master when it satisfies the Master Address and Node enable check. This is determined by bits 0-15 and 20-23 in the MACRO stations MI996 and the Turbo PMACs I6841, I6891, I6941 and I6991.

MACRO Ring Rules

1. Only one synchronizing master can be on the ring. This is bit 4=1 and 5=1 of the Turbo PMAC's I6840. Set I6890, I6940, and I6990 bit 4=1 and bit 5=0 for them to be non-synchronizing masters or bit 4=0 and bit 5=0, if will not be sending data on the ring. As an exception to this rule, upon a ring break, a MACRO slave station becomes a synchronizing master to send ring break information to its following stations on the ring. To allow MACRO ASCII Communication Mode, bit 14 of I6840 at the Ultralite or Master must be set to one and bit 14 of MI996 at the MACRO CPU must be set. The Ultralite will read this bit at power-up and therefore this parameter must be saved to the Ultralite and then restarted at least once to enable the MACRO ASCII Communication Mode.

2. All MACRO stations on the ring should be running at the same ring (phase clock) rate. This is set on the 16-Axis MACRO stations MI992 and MI997 and the Turbo Ultralite's I6800, I6850, I6900, I6950 and I6801, I6851, I6901, I69551.
3. All MACRO stations on the ring should have their phase clock source synchronized to its own last node packet on the MACRO ring. Normally, this is node 15, so bit 16-19 = $F of MI996 and Turbo's I6841, I6891, I6941, I6991. This means that the MACRO IC that is the source of the phase clock on its station will have its phase clock resynchronized with its last received packet on the ring.
4. The synch node phase lock enable bit must not be set on the synchronizing master. Since it is the source of the MACRO ring data rate, it does not make sense to synchronize to itself and it may cause problems. This is bit 7=0 of Turbo's I6840.
5. The synch node phase lock enable bit must be set on the separate Turbo PMAC non-synchronizing masters and the MACRO slaves. It should be set only on the MACRO IC that is the source of the phase clock and not on the other MACRO IC on the board. This is bit 7=1 and typically is I6690 of the Turbo PMAC non-synchronizing masters and MI996 of MACRO IC 1 of the 16-Axis MACRO station.
6. The source of phase and servo clocks for the synchronizing master must be MACRO IC 0 and it must be master number 0. I6841 = $0FCxxx and I6807=0, I6857=3, I6907=3, I6997=3. Because MACRO ICs 1-3 cannot generate the servo clock, they report back a 1.
7. The source of phase and servo clocks for the non-synchronizing master must be the last MACRO IC that is enabled as a Master. If this is MACRO IC 3, then I6807=1, I6857=3, I6907=3, I6997=0. This allows all four MACRO ICs to receive their data before a phase interrupt occurs.
8. The source of phase and servo clocks for a slave must be the last MACRO IC that receives its data. For the 16-Axis MACRO CPU, this most likely will be MACRO IC 1. The idea is to receive the command data into both MACRO IC 0 and 1 and then resynch (reset) the phase clock and get the phase interrupt. MI142 determines which MACRO IC is the source of the phase clock and the synch node phase lock enable. When MI142=0, it is MACRO IC 1. If using only MACRO IC0, then set MI14=0.

I7: Phase Cycle Extension

On the Turbo PMAC2 board, it is possible to skip hardware phase clock cycles between executions of the phase update software. A Turbo PMAC2 board will execute the phase update software – commutation and/or current-loop closure – every (I7+1) hardware phase clock cycles. The default value for I7 is 0, so normally Turbo PMAC2 executes the phase update software every hardware phase clock cycle.

If the Turbo PMAC2 board is closing the current loop for direct PWM control over the MACRO ring, it should have two hardware ring update cycles (which occur at the hardware phase clock frequency) per software phase update. This eliminates one software cycle of delay in the current loop, which permits slightly higher gains and performance. To do this, set I7 to 1, so that the phase update software would execute every second hardware phase clock cycle and ring update cycle.

Normally the current loop should be closed at an update rate of about 9 kHz (the default rate). If two ring updates are wanted per current loop update, the ring update frequency must be set to 18 kHz. This is possible if there are no more than 40 total active nodes on the ring. To implement this, set I6800 or I7000 to one-half of the default value (see below).

Note:

When making this adjustment, change the Turbo PMAC's I6800/I7000 variable first, then the MACRO Station's MI992. Changing the MACRO Station's MI992 alone, followed by an **MSSAVE** command and an **MS$$$** command could cause the Station's watchdog timer to trip.

I19: Clock Source I-Variable Number

I19 determines which MACRO IC in a Turbo PMAC2 system is the source of the phase and servo clocks for the system. The Turbo PMAC2 defaults ($$$***) to the first MACRO IC found in the UMAC pack. This variable should be kept as the first MACRO IC.

It contains its I-Variable whose value is set to 0 by default to indicate that it is the source of the phase and servo clocks. (The equivalent I-Variable for other MACRO ICs should be set to 3 by default to indicate that these ICs should receive their clock signals as inputs. Note that MACROGATE MACRO ICs that are used typically for MACRO ICs 1, 2, and 3, have no servo clock – they cannot be used as the servo clock source, and even if their clock-direction I-Variable is set to 3, it will report back as 1 to indicate phase-clock input.)

Normally, in Turbo PMAC2 systems interfacing to a MACRO Station, MACRO IC 0 should be the source of the system servo and phase clock signals. The clock-direction I-Variable for MACRO IC 0 is I6807, so I19 on these systems should be set to 6807.

Turbo PMAC2 Ultralite: I6800 and I6801

On a Turbo PMAC2 Ultralite (or UMAC Turbo with Acc-5E) controller, the phase clock frequency is determined by I6800 and I6801. I6800 determines the frequency of the MaxPhase clock, and I6801 determines how the Phase clock frequency is divided down from the MaxPhase clock. I6800 sets the MaxPhase frequency according to the formula:

*MaxPhase Freq. (kHz) = 117,964.8 / [2*I6800+3]*

To set I6800 for a desired MaxPhase frequency, the following formula can be used:

*I6800 = (117,964.8 / [2*MaxPhase(kHz)]) - 1 (rounded down)*

I6801 sets the Phase clock frequency from the MaxPhase according to the formula:

Phase Freq. (kHz) = MaxPhase Freq. (kHz) / [I6801+1]

In MACRO applications, typically I6801 is set to 0, so the Phase clock frequency equals the MaxPhase clock frequency. In this case, I6800 sets the Phase clock frequency, and therefore the MACRO ring-update frequency, directly.

UMAC Turbo

In a UMAC Turbo (3U Turbo PMAC2) system, the Phase clock can come from many possible sources, set by a variety of different variables. However, if a UMAC Turbo system is controlling a MACRO Station through the ring with an Acc-5E MACRO interface board, the MACRO IC 0 on the Acc-5E should be the source of the phase clock for the system. To accomplish this, make sure that I19 is set to 6807 to specify MACRO IC 0 as the clock source for the system, with I6800 and I6801 set to specify the phase clock frequency as in Ultralite PMAC2 boards, explained above. Normally, the Turbo firmware will select MACRO IC 0 automatically on a UMAC, if present, as the clock source on a **$$$***** re-initialization command. MACRO Ring Rules 1-8 should be observed in this setting.

Notes on Servo Clock

On Turbo PMAC2 controllers, the Servo clock frequency is derived from the Phase clock frequency by an integer division, so the setting of the MACRO ring update frequency, which is the same as the Phase clock frequency, determines the possible Servo clock frequencies. The division of the Servo clock frequency from the Phase clock frequency is determined by:

- Turbo PMAC2 Ultralite — I6802 (Servo Frequency = Phase Frequency / [I6802+1])
- UMAC Turbo with Acc-5E — I6802 (Servo Frequency = Phase Frequency / [I6802+1])

Once the servo clock frequency has been established, the Turbo PMAC2 variable I10 must be set accordingly so trajectories execute at the proper speed. Several MACRO timing variables have units of servo clock cycles.

Even if the Turbo PMAC2 controller is not performing commutation or current-loop closure, and therefore not performing any software tasks at the Phase clock frequency, the Phase clock frequency should if possible be set to at least twice the Servo clock frequency. Because the MACRO ring data is transmitted at the Phase clock frequency, the over sampling of ring servo data that results, eliminates one servo cycle's delay in transmission of servo loop data, which permits higher servo gains and better performance.

Turbo PMAC2 MACRO Ring Setup I-Variables

I6840: MACRO IC 0 Master Configuration

Any MACRO IC on a Turbo PMAC2 talking to a MACRO Station must be configured as a master on the ring. For purposes of the MACRO protocol, each MACRO IC is a separate logical master with its own master number, even though there may be multiple MACRO ICs on a single physical Turbo PMAC2. Each ring must have one and only one ring controller (synchronizing master). This should be the MACRO IC 0 of the Turbo PMAC2 boards on the ring.

On a Turbo PMAC2, I6840 should be set to $4030 to make the card's MACRO IC 0 the ring controller and enable the MACRO ASCII Communication feature. This sets bits 4, 5, and 14 of the variable to 1. Setting bit 4 to 1 makes the IC a master on the ring; setting bit 5 to 1, makes the IC the ring controller starting each ring cycle by itself and setting bit 14 to 1 enables the MACRO ASCII Communication feature.

On a Turbo PMAC2 whose MACRO IC 0 will be a master but not ring controller, I6840 should be set to $90. This sets bits 4 and 7 of the variable to 1. Setting bit 4 to 1 makes the IC a master on the ring. Use MACRO Ring Rules for setting bit 7 to 1, which causes this IC to be synchronized to the ring controller IC every time it receives a ring packet specified by I6841. If not using the MACRO IC, set its corresponding I-variable to zero.

I6890/I6940/I6990: MACRO IC 1/2/3 Master Configuration

A Turbo PMAC2 Ultralite may have additional MACRO ICs if Options 1U1, 1U2, and/or 1U3 are ordered. A UMAC Turbo system may have additional MACRO ICs if Option 1 on an Acc-5E is ordered, or if multiple Acc-5E boards are ordered. These additional ICs should be set to be masters but not ring controllers by setting I6890, I6940, and I6990, respectively to $10. This sets bit 4 of the variable to 1, making the IC a master on the ring. Use MACRO Ring Rules for setting up these ICs.

I6841/I6891/I6941/I6991: MACRO IC 0/1/2/3 Node Activation Control

I6841, I6891, I6941, and I6991 on Turbo PMAC2 control which of the 16 MACRO nodes for MACRO ICs 0, 1, 2, and 3, respectively, on the card are activated. They also control the master station number for their respective ICs and the node number of the packet that creates a synchronization signal. The bits of these I-variables are arranged as follows:

Bits 0-15: Activation of MACRO Nodes 0 to 15, respectively (1 = active, 0 = inactive). These 16 bits (usually read as four hex digits) individually control the activation of the MACRO nodes in the MACRO IC on a Turbo PMAC2. Each node that is active on the matching MACRO Station, whether for servo, I/O, or auxiliary communications, should have its node activation bit set to 1. When working with a MACRO Station, Node 15 of each MACRO IC on a Turbo PMAC2 must be activated to permit auxiliary communications, so bit 15 of this variable should always be set to 1 if the IC is used to communicate with a MACRO Station. Node 14 of MACRO IC0 enables the MACRO ASCII Communication feature.

Bits 16-19: Packet Sync Node Slave Number. These four bits together (usually read as one hex digit) form the slave number (0 to 15) of the packet whose receipt by the PMAC2 will set the Sync Packet Received status bit in the MACRO IC. Usually, this digit is set to $F (15) because Node 15 is always activated and it is the last node transmitted. If it is the Sync Packet, then the phase clock will be resynched after all the ring data is in the MACRO registers. This is critical only on the MACRO IC that is the source of the phase clock. Its synch node phase lock bit must be set on this MACRO IC.

Bits 20-23: Master Number. These four bits together form the master number (0 to 15) of the MACRO IC on the MACRO ring.

Each MACRO IC acting as a master on the ring, whether on the same card or different cards, must have its own master number and acts as a separate master station for the purposes of the ring protocol. This master number forms half of the address byte with each packet sent by the PMAC2 over the MACRO ring.

The master number can be the same number as the MACRO IC number (e.g. MACRO IC 0 has master number 0, MACRO IC 1 has master number 1, and so on), and if there is only one Turbo PMAC2 in the ring, this will probably be the case. However, this is not required. The MACRO IC that is the ring controller must have master number 0.

I70/I72/I74/I76: MACRO IC 0/1/2/3 Node Auxiliary Function Enable

I70, I72, I74, and I76 are 16-bit I-Variables (bits 0 - 15) in which each bit controls the enabling or disabling of the auxiliary flag function for the MACRO node number matching the bit number for MACRO ICs 0, 1, 2, and 3, respectively. A bit value of 1 enables the auxiliary flag function; a bit value of 0 disables it. If the function is enabled, PMAC copies information automatically between the MACRO interface flag register and RAM register $00344*n*, $00345*n*, $00346*n*, and $00347*n* (where *n* is the IC's node number 0 – 15) for MACRO ICs 0, 1, 2, and 3, respectively.

Note:

Turbo PMAC MACRO node numbers (as opposed to individual MACRO IC node numbers) go from 0 to 63, with board nodes 0 – 15 on MACRO IC 0, board nodes 16 – 31 on MACRO IC 1, board nodes 32 – 47 on MACRO IC 2, and board nodes 48 – 63 on MACRO IC 3.

Each MACRO node *n* that is used for servo functions should have the corresponding bit *n* of I70, I72, I74, or I76 set to 1. I*xx*25 for the Motor *x* that uses Node *n* should then address $00344*n*, $00345*n*, $00346*n*, or $00347*n*, not the address of the MACRO register itself (see below). If Register 3 of a MACRO node *n* is used for other purposes, such as direct I/O, the corresponding bit *n* of I70, I72, I74, or I76 should be set to 0, so this copying function does not overwrite these registers.

Typically, non-servo I/O functions with a MACRO Station do not involve auxiliary flag functions, so this flag copy function should remain disabled for any node used to transmit I/O between the Turbo PMAC2 and the MACRO Station. If any auxiliary communications is done between the Turbo PMAC2 and the MACRO Station on Nodes 14 and/or 15, bits 14 and 15 of these variables must be set to 0. However, on the MACRO Acc-65M, the auxilary flags are used so its flag transfer bits would be enabled.

Examples:

```
I70=$3                  ; Enabled for MACRO IC 0 Nodes 0 and 1
I70=$7                  ; Enable Nodes 0 & 1 for servo control & 2 for Acc 65M
I72=$30                 ; Enabled for MACRO IC 1 Nodes 4 and 5
I74=$3300               ; Enabled for MACRO IC 2 Nodes 8,9,12, and 13
I76=$3333               ; Enabled for MACRO IC 3 Nodes 0,1,4,5,8,9,12, and 13
```

I71/I73/I75/I77: MACRO IC 0/1/2/3 Node Protocol Type Control

I71, I73, I75, and I77 are 16-bit I-Variables (bits 0 - 15) in which each bit controls whether PMAC uses MACRO Type 0 protocol or the MACRO Type 1 protocol for the node whose number matches the bit number for the purposes of the auxiliary servo flag transfer for MACRO ICs 0, 1, 2, and 3, respectively. A bit value of 0 sets a Type 0 protocol; a bit value of 1 sets a Type 1 protocol.

All 16-Axis MACRO CPU nodes use the Type 1 protocol, so each MACRO node *n* used for servo purposes with a MACRO Station must have bit *n* set to 1. Generally, I71 = I70, I73 = I72, I75 = I74, and I77 = I76 on a Turbo PMAC2 communicating with a MACRO Station.

Remember that if servo nodes for more than one MACRO Station are commanded from a single MACRO IC, the protocol must be selected for all of the active servo nodes on each station.

I78: MACRO Master/Slave Auxiliary Communications Timeout

If I78 is set greater than 0, the MACRO Type 1 Master/Slave Auxiliary Communications protocol using Node 15 is enabled. Turbo PMAC implements this communications protocol using the **`MACROSLAVE`** (**`MS`**), **`MACROSLVREAD`** (**`MSR`**), and **`MACROSLVWRITE`** (**`MSW`**) commands.

If this function is enabled, I78 sets the timeout value in PMAC servo cycles. In this case, if PMAC does not get a response to a Node 15 auxiliary communications command within I78 servo cycles, it will stop waiting and register a MACRO auxiliary communications error, setting Bit 5 of global status register X:$000006.

I78 must be set greater than 0 if any auxiliary communications is desired with a MACRO Station. This reserves Node 15 for the Type 1 Auxiliary Communications. A value of 32 is suggested. If I78 is set greater than 0, bit 15 of I70, I72, I74, and I76 must be set to 0, so Node 15 is not also used for flag transfers.

I79: MACRO Master/Master Auxiliary Communications Timeout

If I79 is set greater than 0, the MACRO Type 1 Master/Master Auxiliary Communications and MACRO ASCII Communication protocol using Node 14 is enabled. Turbo PMAC implements this communications protocol using the **`MACROMASTER`** (**`MM`**), **`MACROMSTREAD`** (**`MMR`**), **`MACROMSTWRITE`** (**`MMW`**), and MACSTAn commands. Only the Turbo PMAC that is the ring controller can execute these commands; other Turbo PMACs that are masters on the ring can respond to these commands from the ring controller.

If this function is enabled, I79 sets the timeout value in PMAC servo cycles. In this case, if the Turbo PMAC does not get a response to a Node 14 master/master auxiliary communications command within I79 servo cycles, it will stop waiting and register a MACRO auxiliary communications error, setting Bit 5 of global status register X:$000006.

I79 must be set greater than 0 if any auxiliary communications is desired with a MACRO Station. A value of 32 is suggested. If a value of I79 greater than 0 has been saved into PMAC's non-volatile memory, then at subsequent power-up/resets, bit 14 of I70 is set to 0, the node-14 broadcast bit (bit 14 of I6840) is set to 1, and activation bit for node 14 (bit 14 of I6841) is set to 1, regardless of the value saved for these variables. This reserves Node 14 of MACRO IC 0 for the Type 1 Master/Master Auxiliary Communications and MACRO ASCII Communication.

I80, I81, I82: MACRO Ring Check Period and Limits

If I80 is set to a value greater than zero, Turbo PMAC will monitor for MACRO ring breaks or repeated MACRO communications errors automatically. A non-zero value sets the error detection cycle time in Turbo PMAC servo cycles. Turbo PMAC checks to see that sync node packets (see I6840 and I6841) are received regularly and that there has not been regular communications errors.

The limits for these checks can be set with variables I81 and I82. If less than I82 sync node packets have been received and detected during this time interval, or if I81 or more ring communications errors have been detected in this interval, Turbo PMAC will assume a major ring problem and all motors will be shut down. Turbo PMAC will set the global status bit Ring Error (bit 4 of X:$000006) as an indication of this error.

Turbo PMAC looks for receipt of sync node packets and ring communications errors once per real-time interrupt – every (I8 + 1) servo cycles). The time interval set by I80 must be large enough that I82 real-time interrupts in PMAC can always execute within the time interval, or false ring errors will be detected. Remember that long motion program calculations can cause skips in the real-time interrupt. Typically, values of I80 setting a time interval of about 20 milliseconds are used. I80 can be set according to the formula:

*I80 = Desired cycle time (msec) * Servo update frequency (kHz)*

For example, with the default servo update frequency of 2.26 kHz, to get a ring check cycle interval of 20 msec, I80 would be set to 20 * 2.26 ≅ 45.

MACRO Node Addresses

The MACRO ring operates by copying registers at high speed across the ring. Therefore, each Turbo PMAC2 master controller on the ring communicates with its slave stations by reading from and writing to registers in its own address space. MACRO hardware handles the data transfers across the ring automatically.

Starting in Turbo firmware version 1.936, the base addresses of the up to four MACRO ICs must be specified in I20 – I23, for MACRO IC 0 – 3 respectively. Before this, the base addresses were fixed at $078400, $079400, $07A400, and $07B400, respectively. Only UMAC Turbo systems can support any other configuration, but only rarely will another configuration be used.

The following tables give the addresses of the MACRO ring registers for Turbo PMAC2 controllers. (Note that it is possible, although unlikely, to have other addresses in a UMAC Turbo system. In these systems, the fourth digit can also take the values 5, 6, and 7.)

Register Addresses for MACRO IC 0 with I20=$078400 (default)

	Turbo PMAC2	Addresses	MACRO IC 0	
Node #	**Reg. 0**	**Reg. 1**	**Reg. 2**	**Reg. 3**
0	Y:$078420	Y:$078421	Y:$078422	Y:$078423
1	Y:$078424	Y:$078425	Y:$078426	Y:$078427
2	X:$078420	X:$078421	X:$078422	X:$078423
3	X:$078424	X:$078425	X:$078426	X:$078427
4	Y:$078428	Y:$078429	Y:$07842A	Y:$07842B
5	Y:$07842C	Y:$07842D	Y:$07842E	Y:$07842F
6	X:$078428	X:$078429	X:$07842A	X:$07842B
7	X:$07842C	X:$07842D	X:$07842E	X:$07842F
8	Y:$078430	Y:$078431	Y:$078432	Y:$078433
9	Y:$078434	Y:$078435	Y:$078436	Y:$078437
10	X:$078430	X:$078431	X:$078432	X:$078433
11	X:$078434	X:$078435	X:$078436	X:$078437
12	Y:$078438	Y:$078439	Y:$07843A	Y:$07843B
13	Y:$07843C	Y:$07843D	Y:$07843E	Y:$07843F
14	X:$078438	X:$078439	X:$07843A	X:$07843B
15	X:$07843C	X:$07843D	X:$07843E	X:$07843F

Register Addresses for MACRO IC 1 with I21=$079400 (default)

	Turbo PMAC2	Addresses	MACRO IC 1	
Node #	**Reg. 0**	**Reg. 1**	**Reg. 2**	**Reg. 3**
0	Y:$079420	Y:$079421	Y:$079422	Y:$079423
1	Y:$079424	Y:$079425	Y:$079426	Y:$079427
2	X:$079420	X:$079421	X:$079422	X:$079423
3	X:$079424	X:$079425	X:$079426	X:$079427
4	Y:$079428	Y:$079429	Y:$07942A	Y:$07942B
5	Y:$07942C	Y:$07942D	Y:$07942E	Y:$07942F
6	X:$079428	X:$079429	X:$07942A	X:$07942B
7	X:$07942C	X:$07942D	X:$07942E	X:$07942F
8	Y:$079430	Y:$079431	Y:$079432	Y:$079433
9	Y:$079434	Y:$079435	Y:$079436	Y:$079437
10	X:$079430	X:$079431	X:$079432	X:$079433
11	X:$079434	X:$079435	X:$079436	X:$079437
12	Y:$079438	Y:$079439	Y:$07943A	Y:$07943B
13	Y:$07943C	Y:$07943D	Y:$07943E	Y:$07943F
14	X:$079438	X:$079439	X:$07943A	X:$07943B
15	X:$07943C	X:$07943D	X:$07943E	X:$07943F

Register Addresses for MACRO IC 2 with I22=$07A400 (default)

	Turbo PMAC2	Addresses	MACRO IC 2	
Node #	**Reg. 0**	**Reg. 1**	**Reg. 2**	**Reg. 3**
0	Y:$07A420	Y:$07A421	Y:$07A422	Y:$07A423
1	Y:$07A424	Y:$07A425	Y:$07A426	Y:$07A427
2	X:$07A420	X:$07A421	X:$07A422	X:$07A423
3	X:$07A424	X:$07A425	X:$07A426	X:$07A427
4	Y:$07A428	Y:$07A429	Y:$07A42A	Y:$07A42B
5	Y:$07A42C	Y:$07A42D	Y:$07A42E	Y:$07A42F
6	X:$07A428	X:$07A429	X:$07A42A	X:$07A42B
7	X:$07A42C	X:$07A42D	X:$07A42E	X:$07A42F
8	Y:$07A430	Y:$07A431	Y:$07A432	Y:$07A433
9	Y:$07A434	Y:$07A435	Y:$07A436	Y:$07A437
10	X:$07A430	X:$07A431	X:$07A432	X:$07A433
11	X:$07A434	X:$07A435	X:$07A436	X:$07A437
12	Y:$07A438	Y:$07A439	Y:$07A43A	Y:$07A43B
13	Y:$07A43C	Y:$07A43D	Y:$07A43E	Y:$07A43F
14	X:$07A438	X:$07A439	X:$07A43A	X:$07A43B
15	X:$07A43C	X:$07A43D	X:$07A43E	X:$07A43F

Register Addresses for MACRO IC 3 with I23=$07B400 (default)

	Turbo PMAC2	Addresses	MACRO IC 3	
Node #	**Reg. 0**	**Reg. 1**	**Reg. 2**	**Reg. 3**
0	Y:$07B420	Y:$07B421	Y:$07B422	Y:$07B423
1	Y:$07B424	Y:$07B425	Y:$07B426	Y:$07B427
2	X:$07B420	X:$07B421	X:$07B422	X:$07B423
3	X:$07B424	X:$07B425	X:$07B426	X:$07B427
4	Y:$07B428	Y:$07B429	Y:$07B42A	Y:$07B42B
5	Y:$07B42C	Y:$07B42D	Y:$07B42E	Y:$07B42F
6	X:$07B428	X:$07B429	X:$07B42A	X:$07B42B
7	X:$07B42C	X:$07B42D	X:$07B42E	X:$07B42F
8	Y:$07B430	Y:$07B431	Y:$07B432	Y:$07B433
9	Y:$07B434	Y:$07B435	Y:$07B436	Y:$07B437
10	X:$07B430	X:$07B431	X:$07B432	X:$07B433
11	X:$07B434	X:$07B435	X:$07B436	X:$07B437
12	Y:$07B438	Y:$07B439	Y:$07B43A	Y:$07B43B
13	Y:$07B43C	Y:$07B43D	Y:$07B43E	Y:$07B43F
14	X:$07B438	X:$07B439	X:$07B43A	X:$07B43B
15	X:$07B43C	X:$07B43D	X:$07B43E	X:$07B43F

Note: With the MACRO station, only nodes that map into Turbo PMAC2 Y registers (0, 1, 4, 5, 8, 9, 12, and 13) can be used for servo control. The nodes that map into X registers (2, 3, 6, 7, 10, 11, and 14) can be used for I/O control. Node 15 is reserved for Type 1 auxiliary communications. Node 14 is often reserved for broadcast communications.

Turbo PMAC2 Conversion Table Setup

Turbo PMAC2 processes the raw feedback data it receives through an encoder conversion table before the servo loop uses the data for feedback. This table permits various techniques, such as 1/T extension of encoder data, to refine the feedback values. However, when the 16-Axis MACRO CPU is used to provide the feedback, the Station has its own encoder conversion table to do the refinement before the data is sent across the ring. Therefore, all that the PMAC's encoder conversion table must do is a simple copying operation.

The encoder conversion table on Turbo PMAC uses I-variables I8000 through I8191. I8000 represents the first line of the first entry in the table. Each entry in the table produces one feedback value. The entry can occupy one, two, or three lines.

Position feedback data for a node from a 16-Axis MACRO CPU appears in the 24-bit Register 0 for the node. The least significant bit of the register represents 1/32 of a count (i.e. there are five bits of fraction).

To process this data for the servo loop (position and/or velocity loop feedback), the Turbo PMAC's conversion table will treat the data as a parallel Y word with no filtering (MACRO provides error detection). This makes bits 20 – 23 of the first line of the entry (the first hex digit) equal to $2. The conversion will be unshifted because the conversion result is expected also to have its LSB represent 1/32 of a count. This makes bit 19 of the first line equal to 1. Bits 0 – 18 contain the 19-bit address of the MACRO node's register 0 ($7x4yy, where 'x' varies with the MACRO IC, and 'yy' varies with the node number). This makes the line equal to $2Fx4yy.

The second line of the entry (the next I-variable) specifies the bit width of the source register in bits 12 – 23 (the first three hex digits) and the starting bit number in bits 0 – 11 (the last three hex digits). Because position feedback in Register 0 is a 24-bit value starting at bit 0, this line should be $018000, where $018 specifies the 24-bit width, and $000 specifies the bit-0 starting point.

The default conversion table in the Turbo PMAC2 Ultralite controller processes the position feedback registers of the eight servo nodes of MACRO IC 0. This yields the values in the following table:

Turbo PMAC2 Ultralite Defaults

I-Variable	Setting	Meaning	I-Variable	Setting	Meaning
I8000	$2F8420	MACRO Node 0 Reg 0 Read	I8008	$2F8430	MACRO Node 8 Reg 0 Read
I8001	$018000	24 bits, bit 0 LSB	I8009	$018000	24 bits, bit 0 LSB
I8002	$2F8424	MACRO Node 1 Reg 0 Read	I8010	$2F8434	MACRO Node 9 Reg 0 Read
I8003	$018000	24 bits, bit 0 LSB	I8011	$018000	24 bits, bit 0 LSB
I8004	$2F8428	MACRO Node 4 Reg 0 Read	I8012	$2F8438	MACRO Node 12 Reg 0 Read
I8005	$018000	24 bits, bit 0 LSB	I8013	$018000	24 bits, bit 0 LSB
I8006	$2F842C	MACRO Node 5 Reg 0 Read	I8014	$2F843C	MACRO Node 13 Reg 0 Read
I8007	$018000	24 bits, bit 0 LSB	I8015	$018000	24 bits, bit 0 LSB
I8016 - I8191 = 0					

The following table contains the entry first line for each servo node Register 0 for all MACRO ICs (with the addresses specified by I20 – I23 at their default values). Remember that the second line of the entry should always be $018000.

Register	First Line Value	Register	First Line Value
MACRO IC 0 Node 0 Reg. 0	$2F8420	MACRO IC 2 Node 0 Reg. 0	$2FA420
MACRO IC 0 Node 1 Reg. 0	$2F8424	MACRO IC 2 Node 1 Reg. 0	$2FA424
MACRO IC 0 Node 4 Reg. 0	$2F8428	MACRO IC 2 Node 4 Reg. 0	$2FA428
MACRO IC 0 Node 5 Reg. 0	$2F842C	MACRO IC 2 Node 5 Reg. 0	$2FA42C
MACRO IC 0 Node 8 Reg. 0	$2F8430	MACRO IC 2 Node 8 Reg. 0	$2FA430
MACRO IC 0 Node 9 Reg. 0	$2F8434	MACRO IC 2 Node 9 Reg. 0	$2FA434
MACRO IC 0 Node 12 Reg. 0	$2F8438	MACRO IC 2 Node 12 Reg. 0	$2FA438
MACRO IC 0 Node 13 Reg. 0	$2F843C	MACRO IC 2 Node 13 Reg. 0	$2FA43C
MACRO IC 1 Node 0 Reg. 0	$2F9420	MACRO IC 3 Node 0 Reg. 0	$2FB420
MACRO IC 1 Node 1 Reg. 0	$2F9424	MACRO IC 3 Node 1 Reg. 0	$2FB424
MACRO IC 1 Node 4 Reg. 0	$2F9428	MACRO IC 3 Node 4 Reg. 0	$2FB428
MACRO IC 1 Node 5 Reg. 0	$2F942C	MACRO IC 3 Node 5 Reg. 0	$2FB42C
MACRO IC 1 Node 8 Reg. 0	$2F9430	MACRO IC 3 Node 8 Reg. 0	$2FB430
MACRO IC 1 Node 9 Reg. 0	$2F9434	MACRO IC 3 Node 9 Reg. 0	$2FB434
MACRO IC 1 Node 12 Reg. 0	$2F9438	MACRO IC 3 Node 12 Reg. 0	$2FB438
MACRO IC 1 Node 13 Reg. 0	$2F943C	MACRO IC 3 Node 13 Reg. 0	$2FB43C

If Turbo PMAC2 is doing the commutation for the motor, it is best to use the previous phase position register in RAM as the servo feedback register. The commutation algorithm has already read the raw position data from the MACRO ring and copied it into this register, storing it for calculations in its next cycle. Using this register in the conversion table ensures that the servo algorithm uses the same position that the commutation used, even if new data has started coming in from the MACRO ring for the next cycle.

The following table lists the conversion table entries to use these registers. Remember that the second line of the entry should always be $018000:

Register	First Line Value	Register	First Line Value
Motor #1 Previous Phase Pos.	$2800B2	Motor #17 Previous Phase Pos.	$2808B2
Motor #2 Previous Phase Pos.	$280132	Motor #18 Previous Phase Pos.	$280932
Motor #3 Previous Phase Pos.	$2801B2	Motor #19 Previous Phase Pos.	$2809B2
Motor #4 Previous Phase Pos.	$280232	Motor #20 Previous Phase Pos.	$280A32
Motor #5 Previous Phase Pos.	$2802B2	Motor #21 Previous Phase Pos.	$280AB2
Motor #6 Previous Phase Pos.	$280332	Motor #22 Previous Phase Pos.	$280B32
Motor #7 Previous Phase Pos.	$2803B2	Motor #23 Previous Phase Pos.	$280BB2
Motor #8 Previous Phase Pos.	$280432	Motor #24 Previous Phase Pos.	$280C32
Motor #9 Previous Phase Pos.	$2804B2	Motor #25 Previous Phase Pos.	$280CB2
Motor #10 Previous Phase Pos.	$280532	Motor #26 Previous Phase Pos.	$280D32
Motor #11 Previous Phase Pos.	$2805B2	Motor #27 Previous Phase Pos.	$280DB2
Motor #12 Previous Phase Pos.	$280632	Motor #28 Previous Phase Pos.	$280E32
Motor #13 Previous Phase Pos.	$2806B2	Motor #29 Previous Phase Pos.	$280EB2
Motor #14 Previous Phase Pos.	$280732	Motor #30 Previous Phase Pos.	$280F32
Motor #15 Previous Phase Pos.	$2807B2	Motor #31 Previous Phase Pos.	$280FB2
Motor #16 Previous Phase Pos.	$280832	Motor #32 Previous Phase Pos.	$281032

Sometimes the conversion table will process data sent back to Turbo PMAC2 through I/O nodes, which are mapped into Turbo PMAC2 as X-registers. Often this is done in cases of dual feedback or loop-around-loop configurations. Because these I/O nodes use X-registers instead of Y-registers, they use the $6 conversion method (X/Y data) instead of the $2 conversion method (Y data only) and specify a 24-bit offset in the second line of the entry

The following table shows entries for processing the data in the 24-bit register 0 of the first six I/O nodes for each MACRO IC:

Register	First Line Value	Register	First Line Value
MACRO IC 0 Node 2 Reg. 0	$6F8420	MACRO IC 2 Node 2 Reg. 0	$6FA420
MACRO IC 0 Node 3 Reg. 0	$6F8424	MACRO IC 2 Node 3 Reg. 0	$6FA424
MACRO IC 0 Node 6 Reg. 0	$6F8428	MACRO IC 2 Node 6 Reg. 0	$6FA428
MACRO IC 0 Node 7 Reg. 0	$6F842C	MACRO IC 2 Node 7 Reg. 0	$6FA42C
MACRO IC 0 Node 10 Reg. 0	$6F8430	MACRO IC 2 Node 10 Reg. 0	$6FA430
MACRO IC 0 Node 11 Reg. 0	$6F8434	MACRO IC 2 Node 11 Reg. 0	$6FA434
MACRO IC 1 Node 2 Reg. 0	$6F9420	MACRO IC 3 Node 2 Reg. 0	$6FB420
MACRO IC 1 Node 3 Reg. 0	$6F9424	MACRO IC 3 Node 3 Reg. 0	$6FB424
MACRO IC 1 Node 6 Reg. 0	$6F9428	MACRO IC 3 Node 6 Reg. 0	$6FB428
MACRO IC 1 Node 7 Reg. 0	$6F942C	MACRO IC 3 Node 7 Reg. 0	$6FB42C
MACRO IC 1 Node 10 Reg. 0	$6F9430	MACRO IC 3 Node 10 Reg. 0	$6FB430
MACRO IC 1 Node 11 Reg. 0	$6F9434	MACRO IC 3 Node 11 Reg. 0	$6FB434
The second line of one of these entries is $018018. The first 018 specifies a 24-bit width. The second 018 specifies a 24-bit offset from the Y-register's bit 0, which puts the least significant bit used at the X-register's bit 0.			

Turbo PMAC2 Motor I-Variables

The following section lists Motor setup variables that have particular considerations when using MACRO Stations.

Ixx01: Commutation Enable

Ixx01 specifies whether Turbo PMAC2 performs commutation for Motor xx, and whether it uses X or Y registers. Only Y registers are used when communicating to a MACRO Station over the ring.

If Turbo PMAC2 is not performing commutation for Motor xx, Ixx01 should be set to 0 (bit 0 = 0 specifies no commutation algorithm, bit 1 = 0 specifies the single output to the Y register whose address is set by Ixx02).

If Turbo PMAC2 is performing commutation for Motor xx over the MACRO ring, with or without digital current-loop closure, Ixx01 should be set to 3 (bit 0 = 1 specifies commutation, bit 1 = 1 specifies commutation feedback from the Y register whose address is set by Ixx83).

Ixx02: Command Output Address

Ixx02 specifies the address of the (first) register where Turbo PMAC2 writes the command output values for Motor xx. In the MACRO Type 1 protocol, this is Register 0 of a MACRO servo node, regardless of the output mode (velocity, torque, phase current, or phase voltage).

The default values of Ixx02 for Turbo PMAC2 Ultralite boards specify these registers for each servo node of each MACRO IC. These are listed in the following table:

Ixx02	Value	Register	Ixx02	Value	Register
I102	$078420	MACRO IC 0 Node 0 Reg. 0	I1702	$07A420	MACRO IC 2 Node 0 Reg. 0
I202	$078424	MACRO IC 0 Node 1 Reg. 0	I1802	$07A424	MACRO IC 2 Node 1 Reg. 0
I302	$078428	MACRO IC 0 Node 4 Reg. 0	I1902	$07A428	MACRO IC 2 Node 4 Reg. 0
I402	$07842C	MACRO IC 0 Node 5 Reg. 0	I2002	$07A42C	MACRO IC 2 Node 5 Reg. 0
I502	$078430	MACRO IC 0 Node 8 Reg. 0	I2102	$07A430	MACRO IC 2 Node 8 Reg. 0
I602	$078434	MACRO IC 0 Node 9 Reg. 0	I2202	$07A434	MACRO IC 2 Node 9 Reg. 0
I702	$078438	MACRO IC 0 Node 12 Reg. 0	I2302	$07A438	MACRO IC 2 Node 12 Reg. 0
I802	$07843C	MACRO IC 0 Node 13 Reg. 0	I2402	$07A43C	MACRO IC 2 Node 13 Reg. 0
I902	$079420	MACRO IC 1 Node 0 Reg. 0	I2502	$07B420	MACRO IC 3 Node 0 Reg. 0
I1002	$079424	MACRO IC 1 Node 1 Reg. 0	I2602	$07B424	MACRO IC 3 Node 1 Reg. 0
I1102	$079428	MACRO IC 1 Node 4 Reg. 0	I2702	$07B428	MACRO IC 3 Node 4 Reg. 0
I1202	$07942C	MACRO IC 1 Node 5 Reg. 0	I2802	$07B42C	MACRO IC 3 Node 5 Reg. 0
I1302	$079430	MACRO IC 1 Node 8 Reg. 0	I2902	$07B430	MACRO IC 3 Node 8 Reg. 0
I1402	$079434	MACRO IC 1 Node 9 Reg. 0	I3002	$07B434	MACRO IC 3 Node 9 Reg. 0
I1502	$079438	MACRO IC 1 Node 12 Reg. 0	I3102	$07B438	MACRO IC 3 Node 12 Reg. 0
I1602	$07943C	MACRO IC 1 Node 13 Reg. 0	I3202	$07B43C	MACRO IC 3 Node 13 Reg. 0

If Motor xx is used to produce a pulse-and-direction output on the MACRO Station to control a traditional stepper drive or a stepper-replacement servo drive, the command output should be written to Register 2 of the servo node.

In this mode, the proper values of Ixx02 are:

Ixx02	Value	Register	Ixx02	Value	Register
I102	$078422	MACRO IC 0 Node 0 Reg. 2	I1702	$07A422	MACRO IC 2 Node 0 Reg. 2
I202	$078426	MACRO IC 0 Node 1 Reg. 2	I1802	$07A426	MACRO IC 2 Node 1 Reg. 2
I302	$07842A	MACRO IC 0 Node 4 Reg. 2	I1902	$07A42A	MACRO IC 2 Node 4 Reg. 2
I402	$07842E	MACRO IC 0 Node 5 Reg. 2	I2002	$07A42E	MACRO IC 2 Node 5 Reg. 2
I502	$078432	MACRO IC 0 Node 8 Reg. 2	I2102	$07A432	MACRO IC 2 Node 8 Reg. 2
I602	$078436	MACRO IC 0 Node 9 Reg. 2	I2202	$07A436	MACRO IC 2 Node 9 Reg. 2
I702	$07843A	MACRO IC 0 Node 12 Reg. 2	I2302	$07A43A	MACRO IC 2 Node 12 Reg. 2
I802	$07843E	MACRO IC 0 Node 13 Reg. 2	I2402	$07A43E	MACRO IC 2 Node 13 Reg. 2
I902	$079422	MACRO IC 1 Node 0 Reg. 2	I2502	$07B422	MACRO IC 3 Node 0 Reg. 2
I1002	$079422	MACRO IC 1 Node 1 Reg. 2	I2602	$07B426	MACRO IC 3 Node 1 Reg. 2
I1102	$07942A	MACRO IC 1 Node 4 Reg. 2	I2702	$07B42A	MACRO IC 3 Node 4 Reg. 2
I1202	$07942E	MACRO IC 1 Node 5 Reg. 2	I2802	$07B42E	MACRO IC 3 Node 5 Reg. 2
I1302	$079432	MACRO IC 1 Node 8 Reg. 2	I2902	$07B432	MACRO IC 3 Node 8 Reg. 2
I1402	$079436	MACRO IC 1 Node 9 Reg. 2	I3002	$07B436	MACRO IC 3 Node 9 Reg. 2
I1502	$07943A	MACRO IC 1 Node 12 Reg. 2	I3102	$07B43A	MACRO IC 3 Node 12 Reg. 2
I1602	$07943E	MACRO IC 1 Node 13 Reg. 2	I3202	$07B43E	MACRO IC 3 Node 13 Reg. 2

Ixx03, Ixx04: Feedback Address

Ixx03 and Ixx04 specify the addresses of the registers that Turbo PMAC2 reads to get its position-loop and velocity-loop feedback values, respectively. Usually, this is a result register in the encoder conversion table. Unless the motor uses dual feedback, the values of Ixx03 and Ixx04 are the same, which means the same sensor is used for both position-loop and velocity-loop feedback.

Note:

The automatic servo node functions of a MACRO Station do not support dual feedback in a single node. For dual feedback, either a second servo node must be used, or one of the feedback values (usually the velocity-loop feedback position) must be sent back to the Turbo PMAC2 as an I/O value.

These variables should contain the address of the last line of the entry in the conversion table. With either of the conversion tables suggested above, this would be the second line of each entry: lines 1, 3, 5, 7, etc. of the conversion table. The default values for the Turbo PMAC2 Ultralite boards listed below contain the addresses of these registers.

Ixx03	Value	Register	Ixx03	Value	Register
I103	$003502	Conversion Table Line 1	I1703	$003522	Conversion Table Line 33
I203	$003504	Conversion Table Line 3	I1803	$003524	Conversion Table Line 35
I303	$003506	Conversion Table Line 5	I1903	$003526	Conversion Table Line 37
I403	$003508	Conversion Table Line 7	I2003	$003528	Conversion Table Line 39
I503	$00350A	Conversion Table Line 9	I2103	$00352A	Conversion Table Line 41
I603	$00350C	Conversion Table Line 11	I2203	$00352C	Conversion Table Line 43
I703	$00350E	Conversion Table Line 13	I2303	$00352E	Conversion Table Line 45
I803	$003510	Conversion Table Line 15	I2403	$003530	Conversion Table Line 47
I903	$003512	Conversion Table Line 17	I2503	$003532	Conversion Table Line 49
I1003	$003514	Conversion Table Line 19	I2603	$003534	Conversion Table Line 51
I1103	$003516	Conversion Table Line 21	I2703	$003536	Conversion Table Line 53
I1203	$003518	Conversion Table Line 23	I2803	$003538	Conversion Table Line 55
I1303	$00351A	Conversion Table Line 25	I2903	$00353A	Conversion Table Line 57
I1403	$00351C	Conversion Table Line 27	I3003	$00353C	Conversion Table Line 59
I1503	$00351E	Conversion Table Line 29	I3103	$00353E	Conversion Table Line 61
I1603	$003520	Conversion Table Line 31	I3203	$003540	Conversion Table Line 63

Ixx10, Ixx95: Absolute Position Address and Format

Ixx10 and Ixx95 permit an automatic read of an absolute position sensor at power-on/reset and/or on a subsequent **$*** or **$$*** command. If Ixx10 is set to 0, the power-on/reset position for the motor will be considered to be 0, regardless of the type of sensor used. If Ixx10 is set to a value greater than zero, an absolute position can be read, with Ixx10 specifying an address for reading this value.

When reading absolute position over the MACRO ring, the address contained in Ixx10 is specified in the format $0000mn, where m is the MACRO IC number (0, 1, 2, or 3) and n is the node number (0, 1, 4, 5, 8, 9, 12, or 13). If both m and n are 0, the fourth hex digit of Ixx10 should be set to 1 (Ixx10 = $000100) to keep the total value of Ixx10 greater than zero and activate the absolute position read.

The following table shows the required values of Ixx10 for all of the MACRO nodes that can be used.

Note:

MACRO IC 0 Node 0 uses an Ixx10 value of $000100 because Ixx10=0 disables the absolute position read function.

Ixx10 for MACRO Absolute Position Reads
(Ixx95=$720000 - $740000, $F20000 - $F40000)

MACRO Node Number	Ixx10 for MACRO IC 0	Ixx10 for MACRO IC 1	Ixx10 for MACRO IC 2	Ixx10 for MACRO IC 3
0	$000100	$000010	$000020	$000030
1	$000001	$000011	$000021	$000031
4	$000004	$000014	$000024	$000034
5	$000005	$000015	$000025	$000035
8	$000008	$000018	$000028	$000038
9	$000009	$000019	$000029	$000039
12	$00000C	$00001C	$00002C	$00003C
13	$00000D	$00001D	$00002D	$00003D

There are specific settings of Turbo PMAC2's Ixx95 for each type of MACRO sensor interface. The 16-Axis MACRO CPU has a corresponding variable MI11x for each node that must be set.

16-Axis MACRO CPU Feedback Type	Turbo Ixx95 (Unsigned)	Station MI11x (Unsigned)	Turbo Ixx95 (Signed)	Station MI11x (Signed)
Acc-8D Opt 7 Resolver/Digital Converter	$730000	$00xxxx - $07xxxx	$F30000	$80xxxx - $87xxxx
Acc-8D Opt 9 Yaskawa Absolute Encoder Converter	$720000	$71xxxx - $72xxxx	$F20000	$F1xxxx - $F2xxxx
Acc-49 Sanyo 26-bit Absolute Encoder Converter	$740000	$32xxxx	$F40000	$B2xxxx
Acc-28B/E Analog/Digital Converter	$740000	$31xxxx	--	--
Acc-1E/6E/36E/59E A/D Converter (low 12 bits)	$740000	$0Cxxxx	$F40000	$8Cxxxx
Acc-1E/6E/36E/59E A/D Converter (high 12 bits)	$740000	$33xxxx	$F40000	$B3xxxx
MACRO Station MLDT Input	$740000	$l7xxxx	--	--
MACRO Station Parallel Input (from 24-bit word	$740000	$08xxxx - $2Axxxx	$F40000	$88xxxx - $AAxxxx
Acc-3E/14E Parallel Input from two consecutive lower bytes	$740000	$2Bxxxx	$F40000	$ABxxxx
Acc-3E /14E Parallel Input from two consecutive middle bytes	$740000	$2Cxxxx	$F40000	$ACxxxx
Acc-3E /14E Parallel Input from two consecutive upper bytes	$740000	$2Dxxxx	$F40000	$ADxxxx
Acc-3E/14E Parallel Input from two consecutive lower bytes	$740000	$2Exxxx	$F40000	$AExxxx
Acc-3E /14E Parallel Input from two consecutive middle bytes	$740000	$2Fxxxx	$F40000	$AFxxxx
Acc-3E/14E Parallel Input from two consecutive upper bytes	$740000	$30xxxx	$F40000	$B0xxxx

When Turbo PMAC2's Ixx10 and Ixx95 are set to get absolute position over MACRO, it executes a station auxiliary read command **MS{node},MI920** to request the absolute position from the 16-Axis MACRO CPU. The station then references its own MI11x value to determine the type, format, and address of the data to be read. The data is returned to Turbo PMAC2 with up to 42 bits of data, sign extended to 46 bits. Note that the Turbo PMAC's Ixx95 and the Station's MI11x must agree as to whether the data is signed or unsigned.

Bit 48 is a Ready/Busy handshake bit and Bit 47 is a pass/fail status bit. If Bit 47 is set, the upper 24 bits of the 48 bits returned are an error word and are stored in X:$00320E of the PMAC/PMAC2.

Ixx25, Ixx24: Flag Address and Mode

If the auxiliary functions for Node *n* of MACRO IC 0, 1, 2, or 3 have been enabled by setting Bit *n* of I70, I72, I74, or I76, respectively, to 1, the flag information in Register 3 for the node is copied automatically to and from PMAC RAM register $00347n, $00357n, $00367n, or $00377n, respectively. In this case, Ixx25 should specify the address of the RAM copy, not the actual MACRO interface register.

The following table lists the default values for Ixx25 on a Turbo PMAC2 Ultralite, which shows the address of the RAM copy register for each MACRO servo node:

Ixx25	Value	Register	Ixx25	Value	Register
I125	$003440	MACRO Flag Register Set 0	I1725	$003460	MACRO Flag Register Set 32
I225	$003441	MACRO Flag Register Set 1	I1825	$003461	MACRO Flag Register Set 33
I325	$003444	MACRO Flag Register Set 4	I1925	$003464	MACRO Flag Register Set 36
I425	$003445	MACRO Flag Register Set 5	I2025	$003465	MACRO Flag Register Set 37
I525	$003448	MACRO Flag Register Set 8	I2125	$003468	MACRO Flag Register Set 40
I625	$003449	MACRO Flag Register Set 9	I2225	$003469	MACRO Flag Register Set 41
I725	$00344C	MACRO Flag Register Set 12	I2325	$00346C	MACRO Flag Register Set 44
I825	$00344D	MACRO Flag Register Set 13	I2425	$00346D	MACRO Flag Register Set 45
I925	$003450	MACRO Flag Register Set 16	I2525	$003470	MACRO Flag Register Set 48
I1025	$003451	MACRO Flag Register Set 17	I2625	$003471	MACRO Flag Register Set 49
I1125	$003454	MACRO Flag Register Set 20	I2725	$003474	MACRO Flag Register Set 52
I1225	$003455	MACRO Flag Register Set 21	I2825	$003475	MACRO Flag Register Set 53
I1325	$003458	MACRO Flag Register Set 24	I2925	$003478	MACRO Flag Register Set 56
I1425	$003459	MACRO Flag Register Set 25	I3025	$003479	MACRO Flag Register Set 57
I1525	$00345C	MACRO Flag Register Set 28	I3125	$00347C	MACRO Flag Register Set 60
I1625	$00345D	MACRO Flag Register Set 29	I3225	$00347D	MACRO Flag Register Set 61

Ixx24 specifies how the address in Ixx25 is to be used. Bit 0 of Ixx24 must be set to 1 to specify PMAC2-style flag arrangements, which are used in the MACRO protocol. Bit 18 of Ixx24 must be set to 1 to specify that the flags are sent and received across MACRO. Bit 23 of Ixx24, which specifies the polarity of the amplifier/node fault bit into the Turbo PMAC2, must be set to match the polarity defined in the Station with the appropriate bit of MI18. If the bit n of MI18, and bit 23 of Ixx24 are set to 0, a low-true fault (logical 0 means fault, regardless of the input voltage) is specified. If the bit n of MI18, and bit 23 or Ixx24 are set to 1, a high-true fault (logical 1 means fault) is specified.

If no other bits of Ixx24 are set, the value of Ixx24 is $040001 or $840001. $040001 is the default value for Ixx24 on Turbo PMAC2 Ultralite boards. Refer to the detailed description of Ixx24 in the Turbo PMAC Software Reference for descriptions of the other bits.

When Bit 18 of Ixx24 is set to 1, and bit *n* of I1000 is set to 1, then the Motor xx flag information is copied automatically between the holding registers at $00344*n*, $00345*n*, $00346*n*, or $00347*n*, and the MACRO interface registers for node *n* of MACRO IC 0, 1, 2, or 3, respectively, on the Turbo PMAC2. The command flags, such as amplifier enable, are held in the Y-register of $0034x*n*. The feedback flags, such as overtravel limits and amplifier fault, are held in the X-registers of $0034x*n*. Monitoring of flag values should use these holding registers in RAM, not the actual MACRO node registers.

The following tables show the locations of the individual flags in these registers:

Motor Command Flags: Y:$0034*xn* for MACRO IC (*x*-4) Node *n*

Bit #	Function	Notes
0	Position Capture Prepare Flag	Must be set to 1 to prepare for hardware capture over ring; to 0 when done
1-7	(Not Used)	
8-10	(Reserved for Future Use)	
11	Position Capture Enable Flag	Must be set to 1 to prepare for hardware capture over ring; to 0 when done
12	Node Position Reset Flag	
13	(Reserved for Future Use)	
14	Amplifier Enabled	Command to Station
15-23	(Reserved for Future Use)	

Motor Status Flags: X:$0034*xn* for MACRO IC (*x*-4) Node *n*

Bit #	Function	Notes
0-7	(Not Used)	
8-10	(Reserved for Future Use)	
11	Position Captured Flag	Latched from selected flag
12	Power-On Reset or Node Reset Occurred	
13	Ring Break Detected Elsewhere	
14	Amplifier Enabled	Status from Station
15	Amplifier/Node Shutdown Fault	1 is treated as shutdown if Ix25 bit 23 = 1; 0 if Ix25 bit 23 = 0
16	Home Flag (HOME) Input Value	
17	Positive Limit Flag (PLIM) Value	
18	Negative Limit Flag (MLIM) Value	
19	User Flag (USER) Input Value	
20	Flag W Input Value	
21	Flag V Input Value	
22	Flag U Input Value	
23	Flag T Input Value	

Ixx70, Ixx71: Commutation Cycle Size

If the Turbo PMAC2 is performing commutation for Motor xx, providing either two phase current commands (sine-wave output) or three PWM phase voltage commands (direct PWM output), the size of the commutation cycle is equal to Ixx71/Ixx70, normally expressed in encoder counts. Because the MACRO station provides position feedback in units of 1/32 count for both servo and commutation, the value of Ixx70 and Ixx71 must be set to provide a ratio 32 times the number of true counts in the commutation cycle.

For example, if the commutation cycle has 1000 encoder counts, Ixx70 could be set to 1 and Ixx71 could be set to 32,000.

Ixx75: Absolute Phase Position Offset

If Ixx81 (see below) is set to a value greater than 0, then PMAC will read an absolute sensor for power-on phase position. In this case, it will use Ixx75 to determine the difference between the absolute sensor's zero position and the phase commutation cycle's zero position (unless Hall commutation sensors are used, in which case Ixx91 contains the initial offset information, which needs to be corrected later).

Normally, this position difference in Ixx75 is expressed in counts multiplied by Ixx70. However, when the absolute position is read from the position feedback register, as from a Yaskawa absolute encoder through an Acc-8D Opt 9 and the MACRO Station, then Ixx75 is expressed in units of 1/32 of a count multiplied by Ixx70.

Ixx81, Ixx91: Power-On Phase Position Address and Mode

Ixx81 permits an automatic read of an absolute position sensor for phase referencing of a synchronous motor commutated by PMAC. This read can be done automatically at power-on/reset (if Ixx80 = 1 or 3), or subsequently on the **$** or **$$** command. If Ixx81 is set to 0, the power-on/reset phase position for the motor will be considered to be 0, regardless of the type of sensor used. If Ixx81 is set to a value greater than 0, Turbo PMAC2 will use the address or node specified by Ixx81 to read an absolute phase position, in a manner determined by Ixx91.

There are specific settings of Turbo PMAC2's Ixx91 for each type of MACRO interface to the MACRO Station. The 16-Axis MACRO CPU has a corresponding variable MI11x for each node that must be set. Since the MACRO Station's MI11x also affects the absolute servo position read by Ixx10 and Ixx95 on the Turbo PMAC2, and that position can be signed or unsigned, MI11x bit 23 specifies whether the absolute servo position is read as signed or unsigned. This bit does not matter for the purposes of absolute phase position which is always treated as unsigned.

The following table shows the possible settings of Ixx81 and Ixx91, along with the required matching settings of MI11x, for the different types of absolute phase position formats supported:

16-Axis MACRO CPU Feedback Type	Ixx81 Value	Ixx91 Value	Station MI11x Bits 16-23
Acc-8D Opt 7 Resolver/Digital Converter	$0000mn	$730000	$00-$07 ($80-$87)
Acc-8D Opt 9 Yaskawa Absolute Encoder Converter	$07x4yy	$bb0000	See Note 1
Acc-49 Sanyo Absolute Encoder Converter	$0000mn	$740000	$0D
Acc-28B/E Analog/Digital Converter	$0000mn	$740000	$31
MACRO Station Hall Sensor Flag Input	$0034xn	$vv0000	See Note 2
Acc-1E/6E/36E/59E A/D Converter (low 12 bits)	$0000mn	$740000	$0C ($8C)
Acc-1E/6E/36E/59E A/D Converter (high 12 bits)	$0000mn	$740000	$33 ($B3)
MACRO Station Parallel Input	$0000mn	$740000	$08-$30 ($88-$B0)
MACRO Station MLDT Input	$0000mn	$740000	$08-$30 ($88-$B0)
Acc-3E/14E Parallel Input from two consecutive lower bytes	$0000mn	$740000	$2B ($AB)
Acc-3E/14E Parallel Input from two consecutive middle bytes	$0000mn	$740000	$2C ($AC)
Acc-3E/14E Parallel Input from two consecutive upper bytes	$0000mn	$740000	$2D ($AD)
Acc-3E/14E Parallel Input from two consecutive lower bytes	$0000mn	$740000	$2E ($AE)
Acc-3E /14EParallel Input from two consecutive middle bytes	$0000mn	$740000	$2F ($AF)
Acc-3E /14EParallel Input from two consecutive upper bytes	$0000mn	$740000	$30 ($B0)

m is the number of the MACRO IC used: 0, 1, 2, or 3

n is the MACRO node number used for Motor xx: 0, 1, 4, 5, 8, 9, C(12), or D(13).

bb is the number of bits in a revolution of the encoder (e.g. 13 bits for 8192 counts) plus 5 (to account for the 5 fractional bits), expressed in hexadecimal format (e.g. for 8192 counts/rev, 13 + 5 = 18 = $12)

$07x4yy represents the address of the MACRO node's Register 0, which contains the position information. x is 8, 9, A, or B, for MACRO IC 0, 1, 2, or 3, respectively. yy varies with the node number.

vv is a value from $80 to $FF representing the Hall sensor offset and direction.

For purposes of absolute phase position, Turbo PMAC2 simply reads the encoder counter value in its own MACRO node, communicated automatically by the servo node functions. The MI11x setting is not used to obtain the phase position; usually it is set to $71xxxx or $F1xxxx for absolute servo position.

For purposes of absolute phase position, Turbo PMAC2 simply reads the hall sensor values in its own MACRO node, communicated automatically by the auxiliary servo node functions. The MI11x setting is not used here, but may be set to a non-zero value if an absolute servo position sensor is also used.

When Turbo PMAC2 has Ixx91 set to get absolute position over MACRO, it executes a station auxiliary read command **`MS{node},MI920`** to request the absolute position from the 16-Axis MACRO CPU.

The station then references its own MI11x value to determine the type, format, and address of the data to be read. The data is returned to Turbo PMAC2 with up to 42 bits of data, sign extended to 46 bits. Bit 48 is a Ready/Busy handshake bit and Bit 47 is a pass/fail status bit. If Bit 47 is set, the upper 24 bits of the 48 bits returned form a word and are stored in X:$00320E of the Turbo PMAC2.

Note:

With the Yaskawa absolute encoder format and with the Hall commutation sensor format, the Turbo PMAC2 is not going directly to the MACRO Station for absolute phase position information. This information has been copied already into a Turbo PMAC2 register with another software function.

Ixx82: Current Loop Feedback Address

If the Turbo PMAC2 is being operated in direct PWM mode, Ixx82 must specify the address of the Phase B current feedback register. (If it is not being operated in direct PWM mode, Ixx82 must be set to 0.) When in direct PWM mode over MACRO, the Phase B current feedback value appears in the MACRO servo node's Register 2, so Ixx82 must contain the address of this register. The following table shows the typical values of Ixx82 in this mode, listing the address of Register 2 for each servo MACRO node.

Ixx82	**Value**	**Register**	**Ixx82**	**Value**	**Register**
I182	$078422	MACRO IC 0 Node 0 Reg. 2	I1782	$07A422	MACRO IC 2 Node 0 Reg. 2
I282	$078426	MACRO IC 0 Node 1 Reg. 2	I1882	$07A426	MACRO IC 2 Node 1 Reg. 2
I382	$07842A	MACRO IC 0 Node 4 Reg. 2	I1982	$07A42A	MACRO IC 2 Node 4 Reg. 2
I482	$07842E	MACRO IC 0 Node 5 Reg. 2	I2082	$07A42E	MACRO IC 2 Node 5 Reg. 2
I582	$078432	MACRO IC 0 Node 8 Reg. 2	I2182	$07A432	MACRO IC 2 Node 8 Reg. 2
I682	$078436	MACRO IC 0 Node 9 Reg. 2	I2282	$07A436	MACRO IC 2 Node 9 Reg. 2
I782	$07843A	MACRO IC 0 Node 12 Reg. 2	I2382	$07A43A	MACRO IC 2 Node 12 Reg. 2
I882	$07843E	MACRO IC 0 Node 13 Reg. 2	I2482	$07A43E	MACRO IC 2 Node 13 Reg. 2
I982	$079422	MACRO IC 1 Node 0 Reg. 2	I2582	$07B422	MACRO IC 3 Node 0 Reg. 2
I1082	$079426	MACRO IC 1 Node 1 Reg. 2	I2682	$07B426	MACRO IC 3 Node 1 Reg. 2
I1182	$07942A	MACRO IC 1 Node 4 Reg. 2	I2782	$07B42A	MACRO IC 3 Node 4 Reg. 2
I1282	$07942E	MACRO IC 1 Node 5 Reg. 2	I2882	$07B42E	MACRO IC 3 Node 5 Reg. 2
I1382	$079432	MACRO IC 1 Node 8 Reg. 2	I2982	$07B432	MACRO IC 3 Node 8 Reg. 2
I1482	$079436	MACRO IC 1 Node 9 Reg. 2	I3082	$07B436	MACRO IC 3 Node 9 Reg. 2
I1582	$07943A	MACRO IC 1 Node 12 Reg. 2	I3182	$07B43A	MACRO IC 3 Node 12 Reg. 2
I1682	$07943E	MACRO IC 1 Node 13 Reg. 2	I3282	$07B43E	MACRO IC 3 Node 13 Reg. 2

Ixx83: Commutation Feedback Address

If the Turbo PMAC2 is performing commutation for Motor xx (Ixx01 bit 0 = 1), providing either two phase current commands (sine-wave output) or three PWM phase voltage commands (direct PWM), Ixx83 must specify the address of the ongoing commutation position feedback.

When commutating over MACRO, the position feedback comes from Register 0 of the MACRO node. In this case, Ixx83 must contain the address of this MACRO node register.

The following table contains the default Ixx83 values for Turbo PMAC2 Ultralite boards, listing the addresses of the position feedback registers for each MACRO servo node.

Ixx83	Value	Register	Ixx83	Value	Register
I183	$078420	MACRO IC 0 Node 0 Reg. 0	I1783	$07A420	MACRO IC 2 Node 0 Reg. 0
I283	$078424	MACRO IC 0 Node 1 Reg. 0	I1883	$07A424	MACRO IC 2 Node 1 Reg. 0
I383	$078428	MACRO IC 0 Node 4 Reg. 0	I1983	$07A428	MACRO IC 2 Node 4 Reg. 0
I483	$07842C	MACRO IC 0 Node 5 Reg. 0	I2083	$07A42C	MACRO IC 2 Node 5 Reg. 0
I583	$078430	MACRO IC 0 Node 8 Reg. 0	I2183	$07A430	MACRO IC 2 Node 8 Reg. 0
I683	$078434	MACRO IC 0 Node 9 Reg. 0	I2283	$07A434	MACRO IC 2 Node 9 Reg. 0
I783	$078438	MACRO IC 0 Node 12 Reg. 0	I2383	$07A438	MACRO IC 2 Node 12 Reg. 0
I883	$07843C	MACRO IC 0 Node 13 Reg. 0	I2483	$07A43C	MACRO IC 2 Node 13 Reg. 0
I983	$079420	MACRO IC 1 Node 0 Reg. 0	I2583	$07B420	MACRO IC 3 Node 0 Reg. 0
I1083	$079424	MACRO IC 1 Node 1 Reg. 0	I2683	$07B424	MACRO IC 3 Node 1 Reg. 0
I1183	$079428	MACRO IC 1 Node 4 Reg. 0	I2783	$07B428	MACRO IC 3 Node 4 Reg. 0
I1283	$07942C	MACRO IC 1 Node 5 Reg. 0	I2883	$07B42C	MACRO IC 3 Node 5 Reg. 0
I1383	$079430	MACRO IC 1 Node 8 Reg. 0	I2983	$07B430	MACRO IC 3 Node 8 Reg. 0
I1483	$079434	MACRO IC 1 Node 9 Reg. 0	I3083	$07B434	MACRO IC 3 Node 9 Reg. 0
I1583	$079438	MACRO IC 1 Node 12 Reg. 0	I3183	$07B438	MACRO IC 3 Node 12 Reg. 0
I1683	$07943C	MACRO IC 1 Node 13 Reg. 0	I3283	$07B43C	MACRO IC 3 Node 13 Reg. 0
Because these are all Y addresses, bit 1 of Ixx01 must be set to 1. With bit 0 of Ixx01 set to 1 to enable commutation, the net value of Ixx01 is 3.					

SOFTWARE SETUP

The software configuration of the 16-Axis MACRO CPU is accomplished through the setup of the Station's own variables. These MI (MACRO Initialization) variables on the station permit configuration of the station for a particular application.

Note:

The 16-Axis MACRO CPU's initialization variables can be referred to as either MI-Variables or I-Variables. This manual uses the MI-Variable terminology to distinguish them from the PMAC's own I-Variables.

Typically, the MI-Variable setup for a 16-Axis MACRO CPU is accomplished through a special program, such as Turbo Setup, which hides the actual PMAC commands. However, these variables may be written to and read from in other applications using special PMAC commands.

Station Variable Read/Write Commands

The most common PMAC commands used for setup are the basic on-line MI-Variable read and write commands. The syntax for the read command is:

MS{node #},MI{variable #}

where **{node #}** can be the number of any active node on the Station (usually that of the lowest active node) for most of the MI-Variables, or the number of the individual node for one of the node-specific MI-Variables (MI910 - MI939). **{variable #}** is the number of the Station MI-Variable (0 - 1023). Sending this command to PMAC causes PMAC to send a request to the Station using the auxiliary communications channel on Node 15 (which must be active) for the value of the specified MI-Variable. The Station then responds with the value, and PMAC reports the value to the host computer.

Example:

```
MS0,MI992                ; Have PMAC request of Station with active node 0
                         ; the value of MI992
6527                     ; MAC responds with the value it received from
6528                     ; Station
MS1,MI910                ; Have PMAC request of Station Node 1 the value of MI910
7                        ; PMAC responds with the value it received from
8                        ; Station
```

The syntax for the write command is

MS{node #},MI{variable #}={constant}

where **{node #}** can be the number of any active node on the Station (usually that of the lowest active node) for most of the MI-Variables, or the number of the individual node for one of the node-specific MI-Variables (MI910 - MI939). **{variable #}** is the number of the Station MI-Variable (0 - 1023). **{constant}** is the numerical value to be assigned to the variable. Sending this command to PMAC causes PMAC to send a command to the Station using the auxiliary communications channel on Node 15 (which must be active) to assign the value to the specified MI-Variable.

Example:

```
MS0,MI992=3263     ; Have PMAC command Station with active node 0 to
                   ; assign a value of 3263 to MI992
MS1,MI910=3        ; Have PMAC command Station Node 1 to assign a
                   ; value of 3to MI910
```

Station Variable Copy Commands

It is also possible to copy values between 16-Axis MACRO CPU MI-Variables and PMAC variables. This is done with MACRO Variable Copy commands, which can be used either as on-line commands or as buffered program commands in background PLC programs PLC1-31 and PLCC1-31 (but not in PLC0, PLCC0, or motion programs, which execute in foreground).

The command that copies from a Station MI-Variable to a PMAC variable (reading from the Station) is the **MSR** command. The syntax for the command is:

MSR{node #},MI{variable #},{PMAC Variable}

where **{node #}** can be the number of any active node on the Station (usually that of the lowest active node) for most of the MI- Variables, or the number of the individual node for one of the node-specific MI- Variables (MI910 - MI939). **{variable #}** is the number of the Station MI-variable (0 - 1023) from which the value is copied. **{PMAC Variable}** is the name of the Variable on PMAC (e.g. P10) to which the value is copied.

Example:

```
MSR0,MI984,P50    ; Copy from Station with active node 0 MI984 to PMAC P50
MSR1,MI922,P99    ; Copy from Station Node 1 MI922 to PMAC P99
```

The command that copies from a PMAC variable to a Station MI- Variable (writing to the Station) is the **MSW** command. The syntax for the command is:

MSW{node #},MI{variable #},{PMAC Variable}

where **{node #}** can be the number of any active node on the Station (usually that of the lowest active node) for most of the MI- Variables, or the number of the individual node for one of the node-specific MI- Variables (MI910 - MI939). **{variable #}** is the number of the Station MI- Variable (0 - 1023) to which the value is copied. **{PMAC Variable}** is the name of the variable on PMAC (e.g. P10) from which the value is copied.

Example:

```
MSW0,MI992,I992   ; Copy from PMAC I992 to Station with active node 0 MI992
MSW1,MI925,P103   ; Copy from PMAC P103 to Station Node 1 MI925
```

Ring Control Setup Variables

Ring Update Frequency

MI992 and MI997 for the 16-Axis MACRO CPU control the phase frequency on the Station, which is the frequency at which the Station expects the ring to be updated. The actual ring update frequency is determined by the ring controller master. For best operation, the 16-Axis MACRO CPU should be set to the same frequency. The values for MI992 and MI997 should be set the same for MACRO IC 0 and 1.

MI992 determines the MaxPhase clock frequency from which the phase clock frequency is derived. The equation is:

$$MaxPhase\ Frequency\ (kHz) = 117{,}964.8 / [2*MI992 + 3]$$

MI997 determines how the phase clock frequency is divided down from MaxPhase. The equation is:

$$Phase\ Frequency\ (kHz) = MaxPhase\ Frequency\ (kHz) / (MI997 + 1)$$

Generally, both the MaxPhase and Phase frequencies will be the same on the MACRO Station as they are on the PMAC2 controlling it. However, only the Phase frequency must be the same.

If the ring controller is a Turbo PMAC2 Ultralite or a UMAC Turbo with Acc-5E, the following relationship should hold:

*{MACRO Station) MI992 * (MI997 + 1) = I6800 * (I6801 + 1) {Turbo PMAC2 Ultralite}*

Note:

Even if I7 is set greater than 0 on a Turbo PMAC2 so that the phasing tasks on the Turbo PMAC2 are not done every phase clock cycle, it is important that the actual phase clock frequencies themselves be the same on the Turbo PMAC2 and the MACRO Station.

Station Servo Clock Frequency

MI998 on a MACRO Station controls how the servo clock on the Station is derived from the Station's phase clock. The phase clock frequency is divided by (MI998 + 1) to obtain the servo clock. No software tasks on the station are performed on the servo clock (all are done on the phase clock), but key feedback registers, such as encoder position and timers, are latched by the servo clock. MI998 should always be set to 0 on a MACRO Station to make the servo clock frequency equal to the phase clock frequency, so that the software algorithms always have updated position information to use.

Additional Node Enabling and Disabling

MACRO IC 0

If MACRO Station setup variables MI975 and MI976 are both set to 0 (the default values) at power-up/reset, only those MACRO servo nodes selected by rotary switch SW1 on the Station are enabled for ring communications. The following table lists which nodes are enabled for each SW1 setting:

Setting	Enabled Nodes	Setting	Enabled Nodes
0	0, 1, 4, 5	8	2, 3 (I/O only)
1	8, 9, 12, 13	9	6, 7 (I/O only)
2	0, 1	A (10)	10, 11 (I/O only)
3	4, 5	B (11)	2,3,6,7 (I/O only)
4	8, 9	C (12)	2,3,6,7,10,11 (I/O only)
5	12, 13	D (13)	11 (I/O only)
6	0, 1, 4, 5, 8, 9	E (14)	None (S/W Ring Order)
7	0, 1, 4, 5, 8, 9, 12, 13	F (15)	11 (I/O only)

Note:

If all motor nodes with MI976 are disabled and I/O nodes with MI975 are not enabled, after saving these values to flash memory and resetting the Station, communication with the Station will be only through Node 15. If there are any other Stations on the ring using the same Master number, this board cannot be talked to individually. The only command that can be sent is a broadcast message such as ***MS$$$***15*** which will reset the card to default, allowing communication to it using one of the nodes enabled by SW1.

Motor Node Disable: MI976 permits the disabling of motor (servo) nodes that would otherwise be enabled by the SW1 setting. This permits their use on other devices on the ring. Setting Bit *n* of MI976 to 1 forces the disabling of Node *n*, even if the SW1 setting would normally enable it. Used only on MACRO IC 0.

I/O Node Enable: MI975 permits the enabling of I/O nodes in addition to the motor nodes that are enabled automatically by the SW1 setting. This permits the automatic real-time transmission of I/O data between the PMAC and the MACRO Station through dedicated I/O nodes. Used only on MACRO IC 0.

S/W Ring Order: MI996 must be written by the Ring Order software setup, saved (**MSSAVE**) and the reset (**MS$$$**) to enable the desired motor and I/O nodes.

MACRO IC 1

There is no SW1 or active MI975 and MI976 for MACRO IC 1.

S/W Ring Order: MI996 must be written by the Ring Order software setup, saved (**`MSSAVE`**) and the reset (**`MS$$$`**) to enable the desired motor and I/O nodes. MI variables for MACRO IC 1 can be accessed by adding 1000 to the variable. For example, MI1996 = accesses MACRO IC 1's MI996.

Node Ring Order: MACRO IC 1's MI996 can also be accessed for setup through MACRO IC 0. To address its setup MI-Variables, add 1000 to the variable. For example, MS0,MI1996 = accesses MACRO IC 1's MI996.

Auto-Detecting the MACRO and Servo ICs

At power-on, $$$ and $$$*** the MACRO and Servo ICs are detected automatically similar to the Turbo PMAC's I4900. The MACRO and Servo IC's detection is stored in X:MI200 and the previously saved value is in Y:MI200. MI210 to MI225 are the IDENT Inn variables that further refine the card type, options and revision number for the Servo ICs similar to Turbo's I4910 – I48225.

MACRO/Servo ICs

Gate Addr	MInns	X:MI200 Bit	Chip Select	IDENT MInn's
$C080-$C0BF	I990-999	$1	MACRO CS4	NA
$C0C0-$C0FF	I990-999	$2	MACRO CS5	NA
NA	NA	$400	NA	NA
NA	NA	$800	NA	NA
$8000-$801F	I900-1939	$4	CS2	I210 ($88C8)
$8040-$805F	I900-1939	$8	CS3	I211 ($88CC)
$8020-$803F	I900-1939	$1000	CS2 Aux	I212 ($88E8)
$8060-$807F	I900-1939	$2000	CS3 Aux	I213 ($88EC)
$9000-$901F	I900-1939	$10	CS2	I214 ($98C8)
$9040-$905F	I900-1939	$20	CS3	I215 ($98CC)
$9020-$903F	I900-1939	$4000	CS2 Aux	I216 ($98E8)
$9060-$907F	I900-1939	$8000	CS3 Aux	I217 ($98EC)
$A000-$A01F	I900-1939	$40	CS2	I218 ($A8C8)
$A040-$A05F	I900-1939	$80	CS3	I219 ($A8CC)
$A020-$A03F	I900-1939	$10000	CS2 Aux	I220 ($A8E8)
$A060-$A07F	I900-1939	$20000	CS3 Aux	I221 ($A8EC)
$B000-$B01F	I900-1939	$100	CS2	I222 ($B8C8)
$B040-$B05F	I900-1939	$200	CS3	I223 ($B8CC)
$B020-$B03F	I900-1939	$40000	CS2 Aux	I224 ($B8E8)
$B060-$B07F	I900-1939	$80000	CS3 Aux	I225 ($B8EC)

If the firmware auto-detection finds that the configuration has changed from the saved, bit 14 (Configuration Error) is set in the System Status word (MI4).

If new Servo ICs are detected in (X:MI200) that were not saved previously in (Y:MI200), they will be loaded with default values.

Binding the Servo ICs to the MACRO ICs

Using the bits in X:MI200, the firmware attaches to each MACRO IC up to two Servo ICs and up to two Encoder ICs. This allows up to eight servo channels and six encoder feedback channels per MACRO IC. The Servo IC's base addresses are registered in read-only MI179 (Servo IC 1) and MI180 (Servo IC 1). The Encoder IC's base addresses are registered in read-only MI189 (Encoder IC 1) and MI190 (Encoder IC 1). The logic that is used to assign the detected Servo IC's to each of the MACRO ICs is defined below.

BITn	System Status (MI4)
15	Detected Ubus Encoder IC 7 Attached to MACRO IC 0 and 1 (two channels each)
16	Detected Ubus Encoder 6 Attached to MACRO IC 1
17	Detected Ubus Encoder 5 Attached to MACRO IC 0
18	Detected Ubus Servo IC 4 Attached to MACRO IC 1
19	Detected Ubus Servo IC 3 Attached to MACRO IC 1
20	Detected Ubus Servo IC 2 Attached to MACRO IC 0
21	Detected Ubus Servo IC 1 Attached to MACRO IC 0
22	Detected CPU MACRO IC 1 ($C0C0)
23	Detected CPU MACRO IC 0 ($C080)

Mapping Machine Interface Channels to MACRO Servo Nodes

MACRO IC 0

From MI179 and MI180 and SW1, the 48-bit MI181 – MI188 variables are created. These variables bind each of the machine interface channels (four channels per IC) to a MACRO Servo Node. The Y:MI18n portion has the machine interface channel base address and the X:MI18n portion has the MACRO Servo Node flag word address. These MI181 – MI188 are 48-bit status only variables and are read only, so this binding cannot be changed. **`$$$`** or **`$$$***`** is done for each power-on. The following describes the assignment for the Servo ICs in the standard address settings:

MACRO Node Address for the Command/Status Flag	X:I181…188
$C0A3	MI181
$C0A7	MI182
$C0AB	MI183
$C0AF	MI184
$C0B3	MI185
$C0B7	MI186
$C0BB	MI187
$C0BF	MI188

MACRO IC 0 ($$$ or $$$*)**

# I/O Nodes	# Servo Nodes	Node Servo IC Base Address	Nodes Enabled	SW1 Select	Y:I181…188
0	4	$8000,$8008, $8010,$8018	0,1, 4,5	0	I181, I182, I183, I184
0	4	$8000,$8008, $8010,$8018	8,9, 12,13	1	I185, I186, I187, I188
0	2	$8000,$8008	0,1	2	I181, I182
0	2	$8000,$8008	4,5	3	I183, I184
0	2	$8000,$8008	8,9	4	I185, I186
0	2	$8000,$8008	12,13	5	I187, I188
0	6	$8000,$8008, $8010,$8018, $8040,$8048	0,1, 4,5, 8,9	6	I181, I182, I183, I184, I185, I186
0	8	$8000,$8008, $8010,$8018, $8040,$8058, $8050,$8058	0,1, 4,5, 8,9, 12,13	7	I181, I182, I183, I184, I185, I186, I187, I188
2	0		2,3	8	
2	0		6,7	9	
2	0		10,11	10	
4	0		2,3,6,7	11	
6	0		2,3,6,7,10,11	12	
1	0		11	13	
0	0	$8000,$8008, $8010,$8018, $8040,$8058, $8050,$8058 (if two Servo ICs)	None	14 (Ring Order)	I181, I182, I183, I184, I185, I186, I187, I188
1	0		11	15 (Equivalent to a $$$***)	

MACRO IC 1

The 48-bit MI181 – MI188 variables are created from MI179 and MI180. These variables bind each of the Servo channels (four channels per IC) to a MACRO Servo Node. The Y:MI18n portion has the Servo channel base address and the X:MI18n portion has the MACRO Servo Node flag word address. These MI181 – MI188 are 48-bit status only variables and are read-only so this binding cannot be changed. $$$ and $$$*** is done for each power-on.

The following describes the Servo ICs being addressed in normal address settings:

MACRO Node Address for the Command/Status Flag	X:I181…188
$C0E3	MI181
$C0E7	MI182
$C0EB	MI183
$C0EF	MI184
$C0F3	MI185
$C0F7	MI186
$C0FB	MI187
$C0FF	MI188

MACRO IC 1 ($$$*)**

Number of Servo ICs (IC #)	Node Servo IC Base Address	Associated Node	Y:I181…188	Encoder Conversion Table (ECT)
1 (#3)	$9000,$9008, $9010,$9018, $9000,$9008, $9010,$9018,	0,1, 4,5, 8,9, 12,13	I181, I182, I183, I184, I185, I186, I187, I188	(ECT limited to four entries, (MI124=0))
2 (#3 and #4)	$9000,$9008, $9010,$9018, $9040,$9048, $9050,$9058,	0,1, 4,5, 8,9, 12,13	I181, I182, I183, I184, I185, I186, I187, I188	(ECT set to eight entries, (MI128=0))

MACRO IC 1 ($$$)

Number of Servo ICs (IC #)	Node Servo IC Base Address	Associated Node	I181…188	MI996 (# Servo Y: nodes enabled)
1 (#3)	$9000,$9008, $9000,$9008, $9000,$9008, $9000,$9008,	0,1, 4,5, 8,9, 12,13	I181, I182, I183, I184, I185, I186, I187, I188	1 or 2
1 (#3)	$9000,$9008, $9010,$9018, $9000,$9008, $9010,$9018,	0,1, 4,5, 8,9, 12,13	I181, I182, I183, I184, I185, I186, I187, I188	> 2
2 (#3 and #4)	$9000,$9008, $9010,$9018, $9040,$9048, $9050,$9058,	0,1, 4,5, 8,9, 12,13	I181, I182, I183, I184, I185, I186, I187, I188	> 2

Multi-Channel Servo Interface Setup

Several MI-Variables on the 16-Axis MACRO CPU affect the hardware setup of multiple machine interface channels on the Station. Because these variables are not specific to one channel or node, they can be accessed with an **`MS{anynode}`** command, where **`{anynode}`** is the number of any active node on the Station that is not active on another Station as well.

Channels 1-4 (First 4-Axis Board)

There are several variables that affect all of the machine interface channels 1 to 4. These MI-Variables reference MI179 for the Servo IC's base address.

PWM Frequency: MI900 controls the PWM frequency of Channels 1-4. Its setting is important only if the PWM outputs are used through the PMAC2-style connectors. The equation for the frequency is:

*PWM Frequency (kHz) = 117,964.8 / [4*MI900 + 6]*

Generally, MI900 is set to the same value as MI992 which controls the PWM frequency for Channels 9 and 10 and the MaxPhase clock frequency. The PWM frequency set by MI900 must be equal to N/2 times the Phase clock frequency set by MI992 and MI997, where N is a positive integer.

Hardware Clock Frequencies: MI903 controls the frequencies of the four hardware clock signals for Channels 1-4: the encoder sample SCLK, the pulse-and-direction PFMCLK, the analog output DAccLK, and the analog input ADCCLK. MI903 is a 12-bit value consisting of four independent 3-bit parts, each controlling one of the clock frequencies. The equation is:

*MI903 = SCLK Divider + 8*PFMCLK Divider + 64*DAccLK Divider*
*+ 512*ADCCLK Divider*

The value of each clock divider can take a value of 0 to 7 and the frequency of each clock signal is:

Clock Frequency = 39.3216 MHz / [2^Clock Divider]

The default value for MI903 of 2258 is suitable for most applications. Refer to the detailed description in the MACRO Station Hardware Reference Manual to change any of these frequencies.

PWM Deadtime/PFM Pulse Width: MI904 controls both the deadtime for PWM outputs on Channels 1-4 and the pulse width for the PFM pulse-and-direction outputs on Channels 1-4. The equations are:

*PWM Deadtime(μsec) = 0.135 * MI904*
PFM Pulse Width (μsec) = MI904 / PFMCLK Freq. (MHz)

DAC Strobe Word: MI905 controls the DAC strobe signal used to create analog outputs for Channels 1-4. It is a 24-bit word that is shifted out one bit per DAccLK cycle, MSB first, starting on the rising edge of the phase clock.

MI905 should be set to the default value of $7FFF00 for use with the on-board 16-bit DACs that come with the Option A PMAC-style interface on the Acc-2E 4-Axis piggyback board, or on an Acc-8A breakout board that attaches to the PMAC2-style connectors on the Acc-2E. MI905 should be set to $7FFFC0 for use with the 18-bit DACs on the Acc-8E PMAC2-style breakout board that attaches to the PMAC2-style connectors on the Acc-2E, or the Acc-24E2A backplane analog Axis-interface/breakout board.

ADC Strobe Word: MI940 controls the ADC strobe signal used to interface to serial A/D converters brought in on Channels 1-4. The default value of $FFFFFE is suitable for most A/D converter types used with the 16-Axis MACRO CPU, either for digital current loop feedback or for general-purpose use from an Acc-28B or Acc-28E board.

Channels 5-8 (Second 4-Axis Board)

There are several variables that affect all of the machine interface channels 5 to 8.

PWM Frequency: MI906 controls the PWM frequency of Channels 5-8. Its setting is important only if the PWM outputs are used through the PMAC2-style connectors. The equation for the frequency is:

*PWM Frequency (kHz) = 117,964.8 / [4*MI906 + 6]*

Generally, MI906 is set to the same value as MI992 which controls the MaxPhase clock frequency. The PWM frequency set by MI906 must be equal to N/2 times the Phase clock frequency set by MI992 and MI997 where N is a positive integer.

Hardware Clock Frequencies: MI907 controls the frequencies of the four hardware clock signals for Channels 5-8: the encoder sample SCLK, the pulse-and-direction PFMCLK, the analog output DAccLK, and the analog input ADCCLK. MI907 is a 12-bit value consisting of four independent 3-bit parts, each controlling one of the clock frequencies. The equation is:

*MI907 = SCLK Divider + 8*PFMCLK Divider + 64*DAccLK Divider*
*+ 512*ADCCLK Divider*

The value of each clock divider can take a value of 0 to 7 and the frequency of each clock signal is:

Clock Frequency = 39.3216 MHz / [2^Clock Divider]

The default value for MI907 of 2258 is suitable for almost all applications. Refer to the detailed description in the MACRO Station Hardware Reference Manual to change any of these frequencies.

PWM Deadtime/PFM Pulse Width: MI908 controls both the deadtime for PWM outputs on Channels 5-8 and the pulse width for the PFM pulse-and-direction outputs on Channels 5-8. The equations are:

*PWM Deadtime (μsec) = 0.135 * MI908*
PFM Pulse Width (μsec) = MI908 / PFMCLK Freq. (MHz)

DAC Strobe Word: MI909 controls the DAC strobe signal used to create analog outputs for Channels 5-8. It is a 24-bit word that is shifted out one bit per DAccLK cycle, MSB first, starting on the rising edge of the phase clock.

MI909 should be set to $7FFFC0 for use with the 18-bit DACs on the Acc-24E2A backplane analog axis-interface/breakout board.

ADC Strobe Word: MI941 controls the ADC strobe signal used to interface to serial A/D converters brought in on Channels 1-4. The default value of $FFFFFE is suitable for most A/D converter types used with the 16-Axis MACRO CPU, either for digital current loop feedback, or for general-purpose use from an Acc-28B.

On Board Auxiliary Channels (Handwheel/Pulse and Direction)

There are several variables that affect all of the machine interface channels 9 and 10 which are present on the 2-Axis piggyback board.

PWM Frequency: MI992 controls the PWM frequency of Channels 9 and 10, as well as the MaxPhase clock frequency from which the phase clock frequency for the entire MACRO Station is derived (see above). The equation is:

$$PWM\ Frequency\ (kHz) = 117{,}964.8 / [4*MI992 + 6]$$

The MaxPhase frequency is exactly twice the PWM 9-10 frequency.

Hardware Clock Frequencies: MI993 controls the frequencies of the four hardware clock signals for Channels 9 and 10: the encoder sample SCLK, the pulse-and-direction PFMCLK, the analog output DAccLK, and the analog input ADCCLK. MI993 is a 12-bit value consisting of four independent 3-bit parts, each controlling one of the clock frequencies. The equation is:

$$MI993 = SCLK\ Divider + 8*PFMCLK\ Divider + 64*DAccLK\ Divider + 512*ADCCLK\ Divider$$

The value of each clock divider can take a value of 0 to 7 and the frequency of each clock signal is:

$$Clock\ Frequency = 39.3216\ MHz / [2\text{^}Clock\ Divider]$$

The default value for MI993 of 2258 is suitable for most applications. Refer to the detailed description in the MACRO Station Hardware Reference Manual to change any of these frequencies.

PWM Deadtime/PFM Pulse Width: MI994 controls both the deadtime for PWM outputs on Channels 9 and 10 and the pulse width for the PFM pulse-and-direction outputs on Channels 9 and 10. The equations are:

$$PWM\ Deadtime\ (\mu sec) = 0.135 * MI994$$

$$PFM\ Pulse\ Width\ (\mu sec) = MI994 / PFMCLK\ Freq.\ (MHz)$$

DAC Strobe Word: MI999 controls the DAC strobe signal used to create analog outputs for Channels 9 and 10. It is a 24-bit word that is shifted out one bit per DAccLK cycle, MSB first, starting on the rising edge of the phase clock.

MI999 should be set to the default value of $7FFF00 for use with the on-board 16-bit DACs that come with the Option A PMAC-style interface on the Acc-1E 2-Axis piggyback board, or on an Acc-8A breakout board that attaches to the PMAC2-style connector on the Acc-1E. MI999 should be set to $7FFFC0 for use with the 18-bit DACs on the Acc-8E PMAC2-style breakout board that attaches to the PMAC2-style connector on the Acc-1E.

ADC Strobe Word: MI942 controls the ADC strobe signal used to interface to serial A/D converters brought in on Channels 9 and 10. The default value of $FFFFFE is suitable for most A/D converter types used with the 16-Axis MACRO CPU, either for digital current loop feedback, or for general-purpose use from an Acc-28B.

Single-Channel Servo Interface Channel Setup

The hardware of the machine interface channels on the 16-Axis MACRO CPU can be configured in software to interface to many different types of devices. To perform this configuration, set Station MI-Variables that are specific to the MACRO node that is mapped to the machine interface channel by the setting of the SW1 rotary switch on the Station. Each of these nodes has its own set of MI-variables in the MI910-MI930 range for this hardware setup of the machine interface channel mapped to the node.

These MI-Variables are set by the **`MS{node},MIxxx`** (xxx=910 to 939) command from PMAC, where **`{node}`** must represent the number of the specific MACRO node, not the number of the any active node on the station as for all of the other Station MI-variables.

The following table shows the physical location of each machine interface channel that could be on a 16-Axis MACRO CPU:

Machine Interface Channel #	Backplane Axis Board Used*	Location On Board
1	Acc-24E2x w/ S1-1, 2 ON	First channel
2	Acc-24E2x w/ S1-1, 2 ON	Second channel
3	Acc-24E2x w/ S1-1, 2 ON	Third channel
4	Acc-24E2x w/ S1-1, 2 ON	Fourth channel
5	Acc-24E2x w/ S1-1, 2 OFF	First channel
6	Acc-24E2x w/ S1-1, 2 OFF	Second channel
7	Acc-24E2x w/ S1-1, 2 OFF	Third channel
8	Acc-24E2x w/ S1-1, 2 OFF	Fourth channel
9	--	First channel
10	--	Second channel
* Acc-51E encoder interpolator boards can be used here as well as Acc-24E2x boards.		

Encoder Decode: MI910 for the node determines how the encoder signal input is decoded. This is commonly set to 3 or 7 for x4 quadrature decode, 8 for internal pulse-and-direction decode in the case of stepper outputs, or 12 for MLDT decode.

Command Output Format: MI916 for the node determines the format of the output signals from the channel. This is set to 0 for PWM format on A, B, and C outputs, or to 3 for DAC format on A and B outputs (for velocity-mode, torque-mode, or sine-wave drives), and pulse-and-direction on the C output (for stepper drives or for MLDT excitation).

Position Capture Control: MI912 and MI913 for the node determine which edges of which signals cause a hardware capture of the encoder position for the channel. This capture function is used for very accurate homing, registration, and probing.

MI912 is commonly set to 1 for capture on a high index-channel input, 2 for a high flag input, or 3 for both a high index and high flag. Usually, MI913 is set to 0 to select the home signal as the flag capture input.

Other Variables: Refer to the 16-Axis MACRO CPU Software Reference manual for details on the other variables, as well as for more details on the variables explained above.

Station Encoder Conversion Table Setup

The 16-Axis MACRO CPU has its own Encoder Conversion Table (ECT) that permits pre-processing of feedback data before it is transmitted back to the PMAC controller. This can simplify the transmission and reduce the amount of data to be transmitted.

Note:

The Turbo PMAC2 has its own Encoder Conversion Table that has much the same capabilities as that of the MACRO Station. However, when PMAC is getting its data from the MACRO Station, the Turbo PMAC2's table simply uses the parallel data format to copy the feedback data from the MACRO node. Refer to the PMAC2 Software Setup for MACRO Station section, above.

The ECT on the MACRO Station has a series of entries in 32 lines. Each line occupies one double word of MACRO Station memory. The Y-word has set-up information; the X-word has result information which is pointed normally to the position feedback MI-Variables MI101 to MI108.

Each line's Y-word has a MACRO Station MI-Variable assigned to it, so setting MI-Variable values completes the set up. The setup word for the first line of the table is assigned MI120; the setup word for the second line for the second line is assigned MI121, and so on, to the 32^{nd} line, whose set-up word is assigned MI151.

An entry in the table can occupy one, two, or three lines, which means that one, two, or three MI-Variables are used to define the set-up words for the entry. If the entry occupies more than one line, the final result of the entry is in the last (highest-numbered address) X-word of the entry, matching the last (highest-numbered) set-up MI-variable for the entry. Other X-words in the entry contain intermediate results.

The following table shows the relationship between ECT line numbers, MI-Variable numbers, and result addresses for MACRO IC 0:

Table Line #	Set-up MI-Variable	Result Address	Table Line #	Set-up MI-Variable	Result Address
1	MI120	X:$0010	17	MI136	X:$0020
2	MI121	X:$0011	18	MI137	X:$0021
3	MI122	X:$0012	19	MI138	X:$0022
4	MI123	X:$0013	20	MI139	X:$0023
5	MI124	X:$0014	21	MI140	X:$0024
6	MI125	X:$0015	22	MI141	X:$0025
7	MI126	X:$0016	23	MI142	X:$0026
8	MI127	X:$0017	24	MI143	X:$0027
9	MI128	X:$0018	25	MI144	X:$0028
10	MI129	X:$0019	26	MI145	X:$0029
11	MI130	X:$001A	27	MI146	X:$002A
12	MI131	X:$001B	28	MI147	X:$002B
13	MI132	X:$001C	29	MI148	X:$002C
14	MI133	X:$001D	30	MI149	X:$002D
15	MI134	X:$001E	31	MI150	X:$002E
16	MI135	X:$001F	32	MI151	X:$002F

The following table shows the relationship between ECT line numbers, MI-Variable numbers, and result addresses for MACRO IC 1:

Table Line #	Set-up MI-Variable	Result Address	Table Line #	Set-up MI-Variable	Result Address
1	MI120	X:$0090	17	MI136	X:$00A0
2	MI121	X:$0091	18	MI137	X:$00A1
3	MI122	X:$0092	19	MI138	X:$00A2
4	MI123	X:$0093	20	MI139	X:$00A3
5	MI124	X:$0094	21	MI140	X:$00A4
6	MI125	X:$0095	22	MI141	X:$00A5
7	MI126	X:$0096	23	MI142	X:$00A6
8	MI127	X:$0097	24	MI143	X:$00A7
9	MI128	X:$0098	25	MI144	X:$00A8
10	MI129	X:$0099	26	MI145	X:$00A9
11	MI130	X:$009A	27	MI146	X:$00AA
12	MI131	X:$009B	28	MI147	X:$00AB
13	MI132	X:$009C	29	MI148	X:$00AC
14	MI133	X:$009D	30	MI149	X:$00AD
15	MI134	X:$009E	31	MI150	X:$00AE
16	MI135	X:$009F	32	MI151	X:$00AF

Incremental Digital Encoder Feedback

If an incremental digital quadrature or digital pulse-and-direction encoder is used for feedback, the $00 conversion method is used typically for the timer-based 1/T extension of incremental encoders.

The following table shows the conversion table MI-Variable values for this type of feedback with channels in the UMAC MACRO pack configuration for MACRO IC 0:

Encoder #	Which Backplane Axis Board Used	Location On Board	Conversion Table MI-Variable Value
Encoder 1	Acc-24E2x S1-1,3,4 = ON, ON, ON	First channel	$008000
Encoder 2	Acc-24E2x S1-1,3,4 = ON, ON, ON	Second channel	$008008
Encoder 3	Acc-24E2x S1-1,3,4 = ON, ON, ON	Third channel	$008010
Encoder 4	Acc-24E2x S1-1,3,4 = ON, ON, ON	Fourth channel	$008018
Encoder 5	Acc-24E2x S1-1,3,4 = OFF, ON, ON	First channel	$008040
Encoder 6	Acc-24E2x S1-1,3,4 = OFF, ON, ON	Second channel	$008048
Encoder 7	Acc-24E2x S1-1,3,4 = OFF, ON, ON	Third channel	$008050
Encoder 8	Acc-24E2x S1-1,3,4 = OFF, ON, ON	Fourth channel	$008058

The following table shows the conversion table MI-Variable values for this type of feedback with channels in the UMAC MACRO pack configuration for MACRO IC 1:

Encoder #	Which Backplane Axis Board Used	Location On Board	Conversion Table MI-Variable Value
Encoder 1	Acc-24E2x S1-1,3,4 = ON, OFF, ON	First channel	$009000
Encoder 2	Acc-24E2x S1-1,3,4 = ON, OFF, ON	Second channel	$009008
Encoder 3	Acc-24E2x S1-1,3,4 = ON, OFF, ON	Third channel	$009010
Encoder 4	Acc-24E2x S1-1,3,4 = ON, OFF, ON	Fourth channel	$009018
Encoder 5	Acc-24E2x S1-1,3,4 = OFF, OFF, ON	First channel	$009040
Encoder 6	Acc-24E2x S1-1,3,4 = OFF, OFF, ON	Second channel	$009048
Encoder 7	Acc-24E2x S1-1,3,4 = OFF, OFF, ON	Third channel	$009050
Encoder 8	Acc-24E2x S1-1,3,4 = OFF, OFF, ON	Fourth channel	$009058

If not using the 1/T extension of the encoder value, the first hex digit of the MI-Variable value should be changed from 0 to C. This setting is recommended when using the simulated feedback from a pulse-and-direction output.

Analog Encoder Feedback

If an analog sine-wave encoder is processed through an Acc-51E high-resolution backplane interpolator board, the $F0 conversion method is used yielding 4096 states per encoder line. This entry is a three-line entry.

The following table describes the three-line MI-Variables that need to be configured for the Ubus Interpolator.

The tables below show the addresses of the quadrature register in the Acc-51E:

Interp SW1 Settings:	6	5	4	3	2	1	UMAC Servo IC 1
1st Intrp	on	on	on	on	on	on	

I-Variables	First Line Setting	Second Line Setting	Third Line Setting	Meaning
MI120,MI121,MI122	$F08000	$8005	$00	Acc-51E Encoder Ch 1
MI123,MI124,MI125	$F08008	$800D	$00	Acc-51E Encoder Ch 2
MI126,MI127,MI128	$F08010	$8015	$00	Acc-51E Encoder Ch 3
MI129,MI130,MI131	$F08018	$801D	$00	Acc-51E Encoder Ch 4

Interp SW1 Settings:	6	5	4	3	2	1	UMAC Servo IC 2
1st Intrp	on	on	on	on	on	off	

I- Variables	First Line Setting	Second Line Setting	Third Line Setting	Meaning
MI132,MI133,MI134	$F08040	$8045	$00	Acc-51E Encoder Ch 1
MI135,MI136,MI137	$F08048	$804D	$00	Acc-51E Encoder Ch 2
MI138,MI139,MI140	$F08050	$8055	$00	Acc-51E Encoder Ch 3
MI141,MI142,MI143	$F08058	$805D	$00	Acc-51E Encoder Ch 4

The third line contains a bias term that is added to both A/D converter readings before the arctangent value is calculated. It is used as a 24-bit value, with the used portion of the A/D converter readings being in the upper 12 bits. For example, if the cycle's average reading from the A/D converters were -5 LSBs of the 12-bit value, this line would be set to +5 x 2^{12}, or 20,480.

Resolver Feedback

If a resolver is used for servo feedback processed through an Acc-8D Opt 7 R/D Converter board, the feedback comes into the 16-Axis MACRO CPU as digital quadrature and is processed the same as a true incremental digital encoder.

MLDT Feedback

If a magnetostrictive linear displacement transducer (MLDT) is used for feedback with the 16-Axis MACRO CPU providing the excitation pulse and measuring the time until it receives the echo pulse using its encoder timer circuitry, then the $30 parallel feedback conversion method is used reading the encoder's timer register as the position value.

This conversion method uses three lines (MI-Variables) of the conversion table. The first MI-Variable contains the method and address. The following table shows the first conversion table MI-Variable values for this type of feedback with channels in the UMAC MACRO pack configuration:

Encoder #	Which Backplane Axis Board Used*	Location On Board	Conversion Table MI-Variable Value
Encoder 1	Acc-24E2x w/ S1-1,3,4 = ON, ON, ON	First channel	$308000
Encoder 2	Acc-24E2x w/ S1-1,3,4 = ON, ON, ON	Second channel	$308008
Encoder 3	Acc-24E2x w/ S1-1,3,4 = ON, ON, ON	Third channel	$308010
Encoder 4	Acc-24E2x w/ S1-1,3,4 = ON, ON, ON	Fourth channel	$308018
Encoder 5	Acc-24E2x w/ S1-1,3,4 = OFF, ON, ON	First channel	$308040
Encoder 6	Acc-24E2x w/ S1-1,3,4 = OFF, ON, ON	Second channel	$308048
Encoder 7	Acc-24E2x w/ S1-1,3,4 = OFF, ON, ON	Third channel	$308050
Encoder 8	Acc-24E2x w/ S1-1,3,4 = OFF, ON, ON	Fourth channel	$308058

The second line of the entry contains a bits used mask, a 24-bit value that contains a 1 in each bit that is to be used from the register. In this type of feedback, all 24 bits of the source register can be used, so this line (MI-Variable) can be $FFFFFF.

The third line of the entry contains the maximum change in the input value that the table will let through in one ring cycle. This provides a filter that is a protection against missing or added echo pulses. This value should be set to a value slightly greater than the maximum true velocity expected. The units are bits of the timer per ring cycle, where one bit of the timer represents 0.0009 inches or 0.024 mm at the 120 MHz timer frequency.

12-Bit A/D Converter Feedback

If an analog input processed through the Acc-36E or Acc-59E backplane A/D boards is used for servo feedback, then the $20 parallel feedback format is used.

The actual ADC inputs are all read through 16-Axis MACRO CPU I/O register in a multiplexed format. The Station firmware de-multiplexes them automatically into separate internal memory registers at Y:$0200 to Y:$0207 for MACRO IC 0 and Y:$208 to Y:$20F for MACRO IC 1 if Station variable MI987 has been set to 1 for both ICs.

Station MI-Variable MI989 specifies the address of the I/O register where the multiplexed A/D converters actually reside. The conversion table will read the de-multiplexed data in the internal memory registers.

The first line of the entry contains the $20 method and the source address. The second line contains the bits used mask word, which is a 24-bit value containing a 1 for every bit of the source register to be used. The first eight analog inputs occupy the low 12 bits of the 24-bit word, so their mask word is $000FFF. The second eight analog inputs occupy the high 12 bits, so their mask word is $FFF000.

The following table shows the conversion table MI-Variable values for the first and second lines (MI-variables) of these entries for MACRO IC 0:

Analog Input Pin	Entry First MI-Variable Value	Entry Second MI-Variable Value	Analog Input Pin	Entry First MI-Variable Value	Entry Second MI-Variable Value
ANAI00	$200200	$000FFF	ANAI08	$200200	$FFF000
ANAI01	$200201	$000FFF	ANAI09	$200201	$FFF000
ANAI02	$200202	$000FFF	ANAI10	$200202	$FFF000
ANAI03	$200203	$000FFF	ANAI11	$200203	$FFF000
ANAI04	$200204	$000FFF	ANAI12	$200204	$FFF000
ANAI05	$200205	$000FFF	ANAI13	$200205	$FFF000
ANAI06	$200206	$000FFF	ANAI14	$200206	$FFF000
ANAI07	$200207	$000FFF	ANAI15	$200207	$FFF000

The following table shows the conversion table MI-variable values for the first and second lines (MI-variables) of these entries for MACRO IC 1:

Analog Input Pin	Entry First MI-Variable Value	Entry Second MI-Variable Value	Analog Input Pin	Entry First MI-Variable Value	Entry Second MI-Variable Value
ANAI00	$200208	$000FFF	ANAI08	$200208	$FFF000
ANAI01	$200209	$000FFF	ANAI09	$200209	$FFF000
ANAI02	$20020A	$000FFF	ANAI10	$20020A	$FFF000
ANAI03	$20020B	$000FFF	ANAI11	$20020B	$FFF000
ANAI04	$20020C	$000FFF	ANAI12	$20020C	$FFF000
ANAI05	$20020D	$000FFF	ANAI13	$20020D	$FFF000
ANAI06	$20020E	$000FFF	ANAI14	$20020E	$FFF000
ANAI07	$20020F	$000FFF	ANAI15	$20020F	$FFF000

If the $30 filtered parallel method is used instead of $20, it is a 3-line entry instead of a 2-line entry. The third line of the entry contains the maximum change in the input value that the table will let through in one ring cycle. This provides a filter that is a protection against noise. This value should be set to a value slightly greater than the maximum true velocity expected. The units are bits of the ADC per ring cycle.

Note:

Station Variable MI988 controls whether the A/D converters are expecting inputs in the -2.5V to +2.5V range or in the 0 to +5V range.

This method can be used also for the 16-bit ADCs on an Acc-28E backplane board. The following table shows the possible entry settings, depending on the settings of dip switch S1 on the board.

S1-1	S1-2	ADC1	ADC2	ADC3	ADC4
ON	ON	$18FFE0	$18FFE1	$18FFE2	$18FFE3
OFF	ON	$18FFE8	$18FFE9	$18FFEA	$18FFEB
ON	OFF	$18FFF0	$18FFF1	$18FFF2	$18FFF3
OFF	OFF	$18B8C0*	$18B8C1*	$18B8C2*	$18B8C3*
* Requires Station firmware revision V1.115 or newer to use this setting.					

To integrate the A/D value before computing the result, the first hex digit of the entry should be changed from a 1 to a 5. In this case, there is a second line to the entry which specifies a bias value that is subtracted from the A/D reading before the integration. This bias is expressed as a 24-bit value, with the upper 16 bits matching the actual data from the A/D converter. For example, if zero voltage into the A/D converter produced a reading of three LSBs of the converter, the bias term should be set to 3 x 2^8, or 768.

14E Parallel Feedback

If parallel data brought in through one of the connectors on an Acc-14E I/O backplane board is used for servo feedback, the $3x conversion method is used for parallel feedback.

Each connector can bring in up to 24 bits of input, mapped as a byte in each of three consecutive words of memory, with the least significant byte mapped into the low address. The least significant bit of the input should be connected to the lowest-numbered I/O point on the connector. The J4 and J5 connectors map into the low byte of these words; the J6 and J7 connectors map into the middle byte; and the J8 and J9 connectors map into the high byte.

If the x digit in the method is 4, the low byte of the three words is used; if x is 5, the middle byte is used; if x is 6, the high byte is used. The address specified is the low address of the three words used.

The following table shows the conversion table MI-Variable values for the first line of the entry for this type of feedback through an Acc-14E backplane board:

S1-1	S1-2	Connector Used	Entry First MI-Variable Value
ON	ON	Top	$34FFE0
ON	ON	Bottom	$34FFE3
OFF	ON	Top	$34FFE8
OFF	ON	Bottom	$34FFEB
ON	OFF	Top	$34FFF0
ON	OFF	Bottom	$34FFF3
OFF	OFF	Top	$34B8C0*
OFF	OFF	Bottom	$34B8C3*
* Requires Station firmware revision V1.115 or newer to use this setting.			

Amplifier Fault Enable and Polarity Control

MI17 and MI18 define whether and how the amplifier-fault inputs to the station are used. Each is an 8-bit variable with one bit for each servo node. The following table shows which bit matches which servo node:

Node Number ‘n’	0	1	4	5	8	9	12 (C)	13 (D)
MI17, MI18 Bit #	0	1	2	3	4	5	6	7

The matching of servo nodes to hardware channel numbers is determined by the setting of rotary switch SW1.

MI17 Amplifier Fault Disable Control

MI17 disables the display of the amplifier fault inputs. It does not affect the AMP fault being transferred to the PMAC. Ix25 must be used to disable the PMAC AMP fault bit. If a bit of M117 is set to the default of 0, the display of the amplifier fault input for the channel connected to that node is enabled. If the bit is set to 1, the display of the amplifier fault input is disabled.

MI18 Amplifier Fault Polarity Control

MI18 determines the polarity of the amplifier fault inputs. If a bit of MI18 is set to the default of 0, the amplifier fault input for the channel connected to that node is considered low-true which means that a logical 0 read on this channel’s fault bit is considered a fault condition, regardless of the input voltage to create this state. If the bit is set to 1, the amplifier fault input is considered high true which means that a logical 1 read on this channel’s fault bit is considered a fault condition.

Typically, on the MACRO Station accessories the fault bit is brought in through an AC Opto component, for which current flowing in either direction creates a logical 0. In the default setup of MI18, this state is considered an amplifier fault. Note that if nothing is connected to such an amplifier fault input, the matching bit of MI18 must be set to 0 in order for the Station to consider the channel not to be in a fault condition.

Bit 23 of Ix25 on a PMAC2 or Bit 23 of Ixx24 on a Turbo PMAC2, for the motor assigned to this node, which controls the amplifier fault polarity at the controller, must be the same value as the matching bit of MI18. The Station will pass back the amplifier fault bit to the PMAC2 in the same polarity it receives it and any MACRO fault passed back using this same bit will be of the same polarity.

Servo Address Variable Setup

There are a few MI-Variables for each motor node. Because the motor nodes are not consecutively numbered (0, 1, 4, 5, 8, 9, 12, 13), these variables specify the node not by its number but by its order (e.g. Node 0 is the first motor node). The following table provides an easy reference:

Node Number n	0	1	4	5	8	9	12 (C)	13 (D)
Node Order x	1	2	3	4	5	6	7	8

The last digit of the MI-Variable number is represented generally by x, where x represents the order of the motor node – the xth motor node. In most cases, x will also represent the Machine Interface Channel used on the 16-Axis MACRO CPU, and the Motor number on PMAC.

MI10x Position Feedback Address

After the initial processing of the feedback in the Station's encoder conversion table, the data is copied to the feedback register of a motor node. Station MI-Variable MI10x for the xth motor node used contains the address of the register, usually one in the conversion table, from which the feedback data is copied into the position feedback register of the node. Because the conversion table occupies registers $0020 to $003F in the Station, typically the values of the MI10x variables contain address values in this range.

MI11x Power-On Position Feedback Address

If absolute power-on position is desired for either commutation phase referencing or complete position referencing, MI11x for the xth motor node on the 16-Axis MACRO CPU must be set to a value greater than zero.

MI16x Power-On MLDT Excitation Value

If a magnetostrictive linear displacement transducer (MLDT) is to receive its excitation pulses from the 16-Axis MACRO CPU, MI16x is used for the xth motor node to set the frequency of the excitation immediately upon power-up or reset so the absolute power-on position of the sensor can be read. If MI16x is greater than 0, this value is copied into the C output register for the machine interface channel corresponding to the xth motor node (as determined by the SW1 setting) as part of the reset function of the Station. Thereafter, only A and B command values are copied from the MACRO node command registers to the machine interface channel registers.

The period between output pulses should be slightly longer than the longest delay in receiving the echo pulse. This delay can be computed by multiplying the length of the MLDT by the speed of sound in the MLDT, usually about 2.8 mm/µsec (0.11 in/µsec). With the output period decided, MI16x can be computed according to the formula:

*MI16x = 16,777,216 / [Output_Period (µsec) * PFMCLKfreq (MHz)]*

For example, to get an output period of 500 µsec (2kHz frequency) with PFMCLK at the default frequency of 9.83 MHz, MI16x can be computed as:

*MI16x = 16,777,216 / (500 * 9.83) = 3413*

General-Purpose I/O Setup

The general-purpose I/O (that is not associated directly with a motor channel) on the 16-Axis MACRO CPU can be set up with just a few Station MI-Variables. The basic concept for real-time general-purpose I/O is that of automatic copying of data between the I/O registers and I/O MACRO nodes. Combined with the automatic copying of data between MACRO nodes on the Station and MACRO nodes on the PMAC controlling the Station, an automatic transfer is obtained between the PMAC and the I/O points on the Station.

MI975 I/O Node Enable

If the I/O MACRO nodes have not already been enabled as part of the initial setup of the MACRO Station, they can be enabled now with MI975. If switch SW1 on the Station has been set to enable any motor nodes, there are no active MACRO I/O nodes enabled by default. However, setting Bit n of MI975 to 1 enables Node n for I/O transfer over MACRO.

If switch SW1 has been set to E (14) and MI975 has been set to its default value of 0 at the most recent power-up/reset, then no motor nodes are enabled, but I/O Node 11 is enabled by default. This setting simply permits communications to an I/O-only MACRO Station before its configuration is finalized.

However, if MI975 is set to a non-zero value on a Station with SW1 set at E (14), then MI975 alone controls which I/O nodes are active. Setting Bit n of MI975 to 1 activates Node n for I/O transfer over MACRO. In this case, Node 11 is not active unless Bit 11 of MI975 is set to 1.

Changes in MI975 take place only at a Station power-up/reset. Therefore, to change which I/O nodes on a Station are active, MI975 must be changed, the new value stored to non-volatile flash memory with the **`MSSAVE{anynode}`** command, then the board reset (usually with the **`MS$$${anynode}`** command). Note that in determining the final active-node word reported in MI996, the MACRO Station clears bit 15 of MI975 to make sure that node 15 is reserved for auxiliary communications.

MI19 I/O Transfer Period

The general-purpose I/O copying or transfer functions on the MACRO Station are enabled by setting MI19 greater than zero. If MI19 is greater than 0, its value specifies the period of the transfer in phase cycles. Typically, this is set to 1 so the transfer is performed every phase cycle. If MI19 is set to 0, none of the transfer variables explained below have any effect.

Bi-Directional I/O Transfer Control

Several MI-Variables on the MACRO Station enable the bi-directional copying of I/O values between MACRO nodes and configurable input/output registers on MACRO Station I/O boards.

Copying from the MACRO node to the I/O register is used for setting outputs; copying from the I/O register to the MACRO node is used for reading inputs. The copying is always done in both directions for all I/O points, even though each I/O point can be used only as an input or an output at any given time. Only a zero value (output off) should be written to an I/O point that is currently being used as an input.

The following MACRO I/O boards will use these bi-directional copying variables:

- Acc-9E isolated 48-input backplane (UMAC) board
- Acc-10E isolated 48-output backplane (UMAC) board
- Acc-11E isolated 24-input/24-output backplane (UMAC) board
- Acc-12E isolated 24-input/24-high-power-output backplane (UMAC) board
- Acc-14E 48 I/O backplane (UMAC) board

The following Station MI-Variables perform the bi-directional transfers with these boards:

- **MI69 and MI70:** These variables copy I/O values between 16-bit MACRO node registers (Registers 1, 2, and 3) and accessory-board I/O registers. These are valuable particularly for single I/O boards with 48 I/O points.

- **MI71:** This variable copies I/O values between 24-bit MACRO node registers (Register 0) and accessory-board I/O registers. This is valuable particularly for single I/O boards with 48 I/O points.
- **MI169:** This variable copies 72 I/O values between an entire 72-bit MACRO node and accessory-board I/O registers.
- **MI171, MI172, and MI173:** These variables copy 144 I/O values between a pair of 72-bit MACRO nodes and accessory-board I/O registers. These are valuable for fully configured Acc-3E boards or a set of three backplane I/O boards sharing a common base address.

MACRO Node Addressing: In each of these variables, both the address of a MACRO I/O node register and an I/O board base address register must be specified. The following table lists the possible MACRO I/O node register addresses for MACRO IC0:

I/O Node #	Register 0 X-Address	Register 1 X-Address	Register 2 X-Address	Register 3 X-Address
2	$C0A0	$C0A1	$C0A2	$C0A3
3	$C0A4	$C0A5	$C0A6	$C0A7
6	$C0A8	$C0A9	$C0AA	$C0AB
7	$C0AC	$C0AD	$C0AE	$C0AF
10	$C0B0	$C0B1	$C0B2	$C0B3
11	$C0B4	$C0B5	$C0B6	$C0B7
14*	$C0B8	$C0B9	$C0BA	$C0BB
*Node 14 may be used only for these I/O transfers if no Type 1 Master/Master auxiliary communications are being performed on the ring between Turbo PMAC2 boards.				

I/O Board Addressing: The 3U-format I/O boards are built around IOGATE I/O ASICs. Each IOGATE IC controls 48 I/O points, mapped into the MACRO Station's addressing scheme as six bytes in consecutive registers (Base_address to Base_address+5). The MACRO Station has a 24-bit data bus, so it is possible to have up to three IOGATE ICs in the same address space. Many of the I/O boards support this, as do the bi-directional copying variables.

The single IOGATE IC of the Acc-4E is mapped into the low byte of the addresses it occupies. The Acc-3E may have up to three ICs, according to the options installed as listed by the following table:

Option	Byte on Data Bus	I/O Points
A	Low (bits 0 – 7)	I/O00 – I/O47
B	Middle (bits 8 – 15)	I/O48 – I/O95
C	High (bits 16 – 23)	I/O96 – I/O143

The following table lists the possible base addresses of the Acc-9E, 10E, 11E, and 12E backplane (UMAC) I/O boards:

Acc Board Address Jumper ON	Board Base Y-Address
E1	$FFE0
E2	$FFE8
E3	$FFF0
E4	$B8C0*
*Requires Station firmware revision V1.115 or newer	

The single IOGATE IC on each of these boards may occupy the low, middle, or high byte of the address space, depending on which rows of the E6 matrix are connected by jumpers:

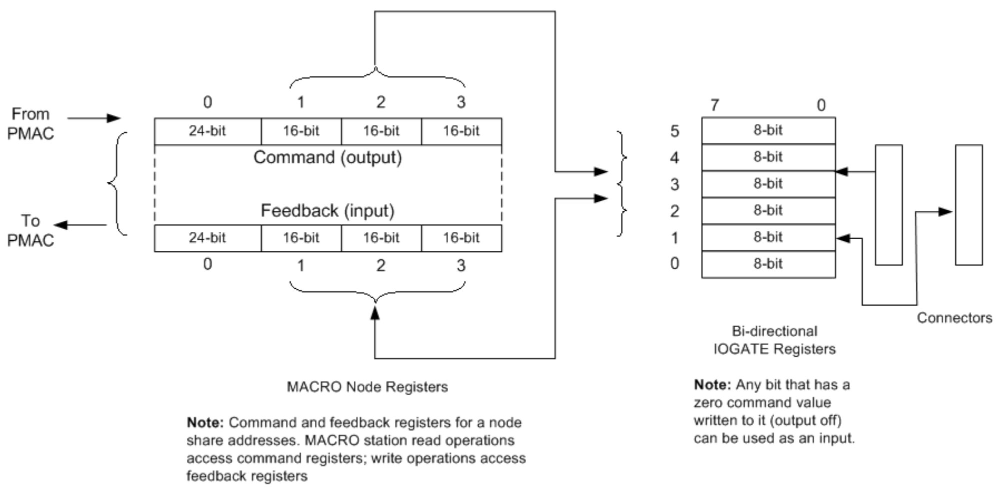
E6A – E6H Rows Connected	Byte on Data Bus
1 and 2	Low (bits 0 – 7)
2 and 3	Middle (bits 8 – 15)
3 and 4	Middle (bits 8 – 15)
4 and 5	High (bits 16 - 23

The single IOGATE on the Acc-14E board can occupy only the low byte of the address space.

Which of these variables is used in a MACRO Station is dependent on the exact configuration desired.

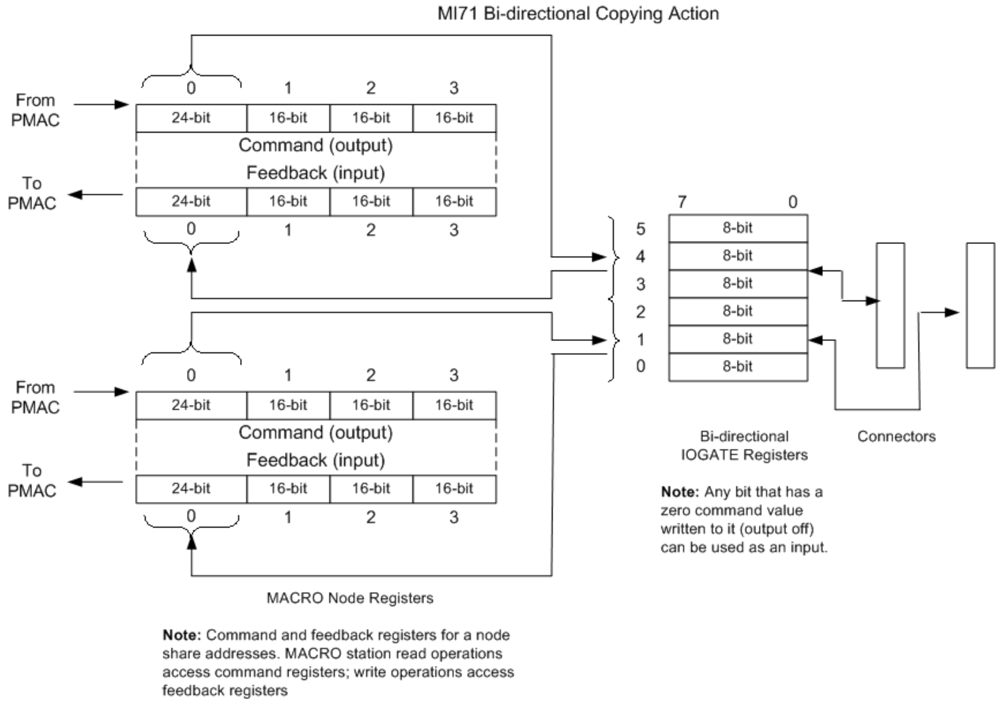
MI69 and MI70 can copy data between one, two, or three 48-bit IOGATE ICs at the same base address and one, two, or three sets of three 16-bit registers in MACRO I/O nodes. The first IOGATE must be in the low byte of the address, the second (if used) must be in the middle byte of this address, and the third (if used) must be in the high byte. The first IOGATE is matched to the three 16-bit registers in the MACRO I/O node whose address is specified, the second to these registers in the next MACRO I/O node, and third to the registers in the following MACRO I/O node.

MI69, MI70 Bi-directional Copying Action



MI71 can copy data between one, two, or three 48-bit IOGATE ICs at the same base address and pairs of 24-bit registers in adjacent MACRO I/O nodes. The first IOGATE must be in the low byte of the address, the second (if used) must be in the middle byte of this address, and the third (if used) must be in the high byte.

The first IOGATE is matched to the 24-bit register in the MACRO I/O node whose address is specified and the 24-bit register in the next MACRO I/O node. The second IOGATE (if used) is matched to the 24-bit registers in the next pair of MACRO I/O nodes. The third (if used) is matched to the 24-bit registers in the following pair of MACRO I/O nodes.



MI169 and MI170 can copy data between two 48-bit IOGATE ICs (although only using the first half of the second IC) and the full 72 bits of a MACRO I/O node (the three 16-bit registers and the single 24-bit register). The first IOGATE must be in the low byte of the address, the second (if used) must be in the middle byte of this address. The first IOGATE is matched to the 3 16-bit registers in the MACRO I/O node whose address is specified, and the half of the second IOGATE is matched to the 24-bit register.

MI171, MI172, and MI173 can copy data between three 48-bit IOGATE ICs at the same base address and the full 72 bits of two consecutive MACRO I/O nodes (the three 16-bit registers and the single 24-bit register of each). The first IOGATE must be in the low byte of the address, the second must be in the middle byte of this address, and the third must be in the high byte.

For more details and examples on the setting of these variables, consult the UMAC MACRO and the MACRO Stack Software Reference manuals and the individual manuals for the I/O accessories.

Uni-Directional I/O Transfer Control

MACRO Station variables MI21 through MI68 specify uni-directional copying functions between pairs of MACRO Station registers, usually some kind of I/O register and a MACRO node register. MI20 is a 48-bit mask variable that specifies which of the 48 possible transfers specified by MI21 through MI68 will actually occur. MI19 controls the frequency at which these transfers occur; it must be greater than 0 for these transfers to occur at all.

MI21 through MI68 are 48-bit variables expressed as 12 hexadecimal digits. Each controls one copying operation from a source register to a destination register. Each variable consists of four parts:

1. Digits 1 and 2: A code representing what part of the source register is used
2. Digits 3 – 6: The address of the source register in the MACRO Station
3. Digits 7 and 8: A code representing what part of the destination register is used
4. Digits 9 – 12: The address of the destination in the MACRO Station

The most commonly used code values are:

- $54: Y-register bits 0 – 11 (Lower 12-bit ADC registers)
- $60: Y-register bits 12 – 23 (Upper 12-bit ADC registers)
- $6C: Y-register bits 8 – 23 (16-bit MACRO servo node registers, Acc-28 ADCs)
- $78: Y-register bits 0 –23 (24-bit MACRO servo node registers)
- $DC: X-register bits 8 – 23 (16-bit MACRO I/O node registers)
- $E8: X-register bits 0 – 23 (24-bit MACRO I/O node registers, conversion table results)

A complete list is found in the Software Reference description of these variables.

Addresses of the registers used can be found in many places in the User Manual and Software Reference; a complete listing is found in the back of the Software Reference.

HOW TO USE THE 16-AXIS MACRO CPU

This section shows many useful examples about the setup of some of the hardware and also shows how to read any memory location at the MACRO CPU for troubleshooting purposes.

Example Setup for 16-Axis System

One concept that is not obvious to the first time user of the 16-Axis MACRO CPU is the node activation setup for the MACRO IC1 for axes 9-16. MACRO IC0 for Axes 1-8 node activation can be setup with SW1. Typically, SW1 is set to 7 and this will activate nodes 0, 1, 4, 5, 8, 9, 12, and 13. For the 16-Axis MACRO CPU, channels 9-16 can be setup only through the MI996 since it is associated with MACRO IC1. Initially, MI996 for MACRO IC1 can be set up by writing to MS0,MI1996 because node 0 will be active (based on SW1 set to 7).

For this example, the clock variables, communication variables and the node activation variables will be set up at both the MACRO CPU and the Master Controller (Turbo Ultralite).

Ultralite Clock Setup Variables

```
I6800=6527          ;default setup for 9.034 KHz Max Phase
I6801=0             ;Default setup for 9.034 Khz Phase and Ring Cycle
I6802=3             ;default setup for 2.258 KHz Servo Clock
```

Ultralite Node Activation

```
i6840=$4030
i6841=$0FF333       ;Master0: Activate nodes 0,1,4,5,8,9,12,13
i6890=$90
i6891=$1fB333       ;Master1: Activate nodes 16,17,20,21,24,25,28,29
i6940=$10
i6941=$2f8000
i6990=$10
i6991=$3f8000

i70=$3333           ;Activate Control/Status Flag Transfer for
i71=$3333           ;nodes 0,1,4,5,8,9,12,13
i72=$3333           ;Activate Control/Status Flag Transfer for
i73=$3333           ;nodes 16,17,20,21,24,25,28,29
i78=32              ;Set Timeout for MacroI/O to 32 servo cycles
i79=32
```

MACRO CPU Node Activation

```
Set SW1=7 and SW2=0     ;Activate nodes 0,1,4,5,8,9,12,13 at MACRO CPU
MS0,MI1996=$1FB333      ;Activate nodes 16,17,20,21,24,25,28,29 at MACRO CPU
                        ;Must MSSAVE & MS$$$ before activated

MS0,MI992=6527      ;Default for 9.034 KHz Max Phase (Always set to I6800)
MS0,MI997=0         ;Default- 9.034 Khz Phase and Ring Cycle (Always set to 0)
MS0,MI998=0         ;Default- 9.034 KHz Servo Clock (Always set to 0)

MS0,MI1992=6527     ;Default for 9.034 KHz Max Phase (Always set to I6800)
MS0,MI1997=0        ;Default- 9.034 Khz Phase and Ring Cycle (Always set to 0)
MS0,MI1998=0        ;Default- 9.034 KHz Servo Clock (Always set to 0)
```

Macro Station Position Capture Setup

The position-capture function latches the current encoder position at the time of an external event into a special register. It is executed totally in hardware, without the need for software intervention (although it is set up and later serviced, in software). This means that the only delays in the capture are the hardware gate delays (negligible in any mechanical system), so this provides an incredibly accurate capture function.

Setting the Trigger Condition

The position capture register can be used both automatically, as in homing routines where the firmware handles the register directly, and manually, where the user programs must handle the register information. Regardless of the mode, the event that causes the position capture is determined by Encoder I-Variables 2 and 3 on the Macro Station (ms0,mi912 and ms0,mi913 for encoder 1 at node 0). Encoder I-Variable 2 defines what combination of encoder third-channel transition and encoder flag transition triggers the capture (it also allows software trigger). If it says to use a flag, Encoder I-Variable 3 determines which flag (usually set to zero to specify the home flag).

Using for Homing

When using this feature for homing a motor, the motor flag address I-Variable (Ix25 for motor x) must point to the proper set of flags (this has to be done anyway to address the limit flags properly). For instance, the default value of I125 for a Turbo Ultralite is $3440, pointing to the first set of flags (with I70/I72/I74/I76 set up to automatically copy the flags to the Macro Station). Then Encoder/Flag I-Variable 2 (e.g. ms0, mi912) and Encoder/Flag I-Variable 3 (e.g. ms0, mi913) define the transition within this encoder and flags to cause the position capture. Once these have been set up properly, the homing function will use the position-capture feature automatically.

Using in User Program

If using the position-capture function in the program, these two I-Variables still control the capture event. Access the captured position through a full word Macro Station MI-Variable (for node0/channel 1 use ms0,mi921). To enable the manual function of position capture two bits in the PMAC MACRO flag command word must be set. Bit 0 and then

MACRO Station Position Compare Output Setup

The position compare outputs for the Servo IC card in a MACRO system are setup in a similar fashion to the traditional PMAC2 Style compare outputs. The functionality of the compare outputs is identical for a MACRO and non-MACRO system. The position compare output feature will allow the user to fire an output as soon as the position in the Servo IC up/down counter is Equal to the value placed into the position compare register.

To monitor the status of the actual position compare output, point a M-Variable definition to bit 9 of the flag copy register (Ixx25). Using this method, the IO copy register does not need to be set up to verify the operation of the compare outputs. These data bits are updated every ring cycle.

Non-Turbo	Turbo
M152->X:$0F70,9	M152->X:$003440,9
M252->X:$0F71,9	M252->X:$003441,9
M352->X:$0F74,9	M352->X:$003444,9
M452->X:$0F75,9	M452->X:$003445,9
M552->X:$0F78,9	M552->X:$003448,9
M652->X:$0F79,9	M652->X:$003449,9
M752->X:$0F7C,9	M752->X:$00344C,9
M852->X:$0F7D,9	M852->X:$00344D,9

The Position Compare Outputs (EQU Outputs) are set up using the following MACRO Node registers:

```
MS{node},MI912 Encoder n Capture Control
MS{node},MI923 Compare Auto-Increment Value
MS{node},MI925 Compare A Position Value
MS{node},MI926 Compare B Position Value
MS{node},MI928 Compare-State Write Enable
MS{node},MI929 Compare-Output Initial State
```

Setting up for a Single Pulse Output

If just a single compare pulse is wanted (not using the auto-increment feature), take the following steps:

1. Write the encoder value at the front edge into the Compare A register
2. Write the encoder value at the back edge into the Compare B register
3. Set the Auto-Increment register to zero
4. Set the initial state with the direct-write feature
   - Write a value to the initial state bit
   - Write a 1 to the direct-write enable bit (this is self-clearing to 0)
5. Start the move that will cause the compare function

Example: For axis 1 using node 0, with the axis sitting still at about encoder position 100, and a '1' value of position compare desired between encoder positions 1000 and 1010, the following code could be used:

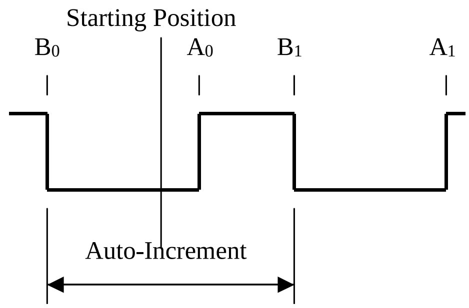
```
MS0,MI925=1000          ; Set front end compare in A
MS0,MI926=1010          ; Set back end compare in B
MS0,MI923=0             ; No auto-increment
MS0,MI929=0             ; Prepare initial value of 0
MS0,MI928=1             ; Enable direct write (resets immediately to zero)
                        {Command to start the move}
```

Setting up for Multiple Pulse Outputs

By using the auto-increment feature, it is possible to create multiple compare pulses with a single software setup operation. When the auto-increment register is a non-zero value, its value is added to or subtracted from one compare register's value automatically when the other compare value is matched. PMAC keeps track of the direction of incrementing, so only positive values should be used in the auto-increment register, even if the encoder will be counting in the negative direction.

The setup for multiple pulses is like the setup for a single pulse, except that a non-zero value must be entered into the auto-increment register. In addition, the value entered for the back edge must be that of the first back edge minus the auto-increment if the move will be positive or that of the first back edge plus the auto-increment value if the move will be negative.

In other words, the starting values to the two compare registers must bracket the starting position. When either compare value is matched by the encoder counter, the other compare value is incremented in the direction of movement.



Example: Starting from the above example, desiring the compare output on between 1000 (A_0) and 1010 (B_1) counts, but adding an auto-increment value of 2000 counts, with a starting position of about 100 counts, program code to start the sequence could be:

```
MS0,MI925=2000          ; Auto-increment of 2000 encoder counts
MS0,MI926=1000          ; First front edge (A0) at 1000 counts
MS0,MI923=(1010-1000)   ;[1010-MS0,MI926]First back edge (B1) at 1010 counts
MS0,MI929=0             ; Prepare initial value of 0
MS0,MI928=1             ; Enable direct write (resets immediately to zero)
```

Using the JHW Port Encoder Inputs

The 16-Axis MACRO CPU has the ability to read two supplemental channels of quadrature/pulse and direction encoders. The user has access to these inputs on the 20-pin JHW. These encoders can be read at rates up to 40 MHz. The only limitation on this port is that port does not allow the user to input an index pulse. Therefore, these inputs are ideal for handwheel encoders or for pure position/velocity feedback.

The data for the encoders are processed from the $C090 for supplemental channel 1 and from $C098 for supplemental channel 2. The two encoders can be processed using the MI990 and MI991 to determine the type of decode. The maximum encoder sample clock frequency is determined by MI993.

To use these supplemental inputs for encoders, typically the Encoder Conversion Table (ECT) must be modified at the MACRO CPU (MI120-MI151) and the encoder node transfer variables (MI101-MI108) must be modified. Typically since most users of the 16-Axis MACRO CPU will have more than eight servo channels being used, Delta Tau recommends modifying the ECT registers and Encoder Node Transfer variables associated with channels 9-16 (MACRO IC1).

Example: If there are ten axes and the two supplemental channels need to be used as channels 11 and 12 for dual feedback for channels 1 and 2. To read the data from the supplemental channels activate the nodes associated with channels 11 and 12 and modify the Encoder Conversion Table at the MACRO CPU.

```
MS0,MI990=7             ; Supplemental Channel 1 is setup for x4 CCW decode
MS0,MI991=7             ; Supplemental Channel 2 is setup for x4 CCW decode
;*** ECT Modification at MACRO IC1 (Channels 9-16)
MS16,MI120=$009000      ; 1/T Interpolation from Encoder 9 ($90) (Default)
MS16,MI121=$009008      ; 1/T Interpolation from Encoder 10  ($91) (Default)
..
MS16,MI126=$009050      ; 1/T Interpolation from Encoder 15 ($96) (Default)
MS16,MI127=$009058      ; 1/T Interpolation from Encoder 16 ($97) (Default)
MS16,MI128=$00C090      ; 1/T Interpolation from Supplemental Channel1 ($98)
MS16,MI129=$00C098      ; 1/T Interpolation from Supplemental Channel2 ($99)
;*** Encoder Node Transfer Modification at MACRO IC1 (Channels 9-16)
MS16,MI101=$90          ;Transfer ECT data from $90 to Node 16 (default)
MS16,MI102=$91          ;Transfer ECT data from $91 to Node 17 (default)
MS16,MI103=$98          ;Transfer ECT data from $90 to Node 20
MS16,MI104=$99          ;Transfer ECT data from $90 to Node 21
```

Once this is information is modified at the MACRO CPU, use the data from feedback devices 11 and 12 for dual feedback or handwheel-master-slave-following purposes. The data for supplemental channels 1 and 2 will be processed at the Ultralite at locations $3516 and $3518 respectively.

Using the JHW Pulse and Direction Outputs

The 16-Axis MACRO CPU has the ability to send pulse and direction signals on its two supplemental channels. The user has access to these inputs on the 20-pin JHW. These outputs are ideal step and direction style amplifiers or general purpose pulse and direction outputs.

The outputs for the Pulse and Direction signals are processed from the $C094 for supplemental channel 1 and from $C09C for supplemental channel 2. The maximum Pulse Frequency is determined by MI993 and the Pulse Frequency pulse-width is determined by MI994.

To use these supplemental pulse and direction outputs, set up the output mode for pulse and direction. The default output for the output mode for the supplemental channels is PWM. Unfortunately, there is not a MI-variable at the MACRO CPU to put its supplemental channels into pulse and direction mode, so write to this location over the ring.

Use these outputs as either general purpose outputs that can be controlled by writing to an IO node address or for servo-stepper control. General purpose outputs will require using an IO node to copy the frequency output command into the actual pulse output register ($C094 and $C09C). Using a servo-stepper style output will require modifying the servo output channel.

Example: Set the pulse and direction outputs for supplemental channels 1 and 2 as general purpose outputs. Two 16-bit registers from IO node 2 are used for this example:

```
MS0,MI19=1              ; transfer every ring cycle
MS0,MI20=3              ; activate MI21 and MI22 for data transfer
MS0,MI21=$DCC0A16DC094  ; Copy node2  X:$C0A1,8,16,s into Y:$C094,8,16,s
MS0,MI21=$DCC0A26DC09C  ; Copy node2  X:$C0A2,8,16,s into Y:$C09C,8,16,s
MS0,MI975=$4            ; Activate Node2 (after save and restart)
```

;*** Change the output mode to PFM for Supplemental Channels 1* and 2*

```
MS0,MI198=$97C095       ;set MI198= X:$C095,23
MS0,MI199=1             ;write 1 to bit 23 of X:$C095,23 for PFM mode
MS0,MI198=$97C09D       ;set MI198= X:$C09D,23
MS0,MI199=1             ;write 1 to bit 23 of X:$C09D,23 for PFM mode
```

or use the optional PLC to do this at power up to automatically put into PFM mode :

```
Open PLC1 Clear
  ms0,mi198=$97c095     ;set X:$C095,23 equal to 1 for PFM mode
  I5111=100*8388608
  While (I5111 > 0)endwhile
  ms0,mi199=1
  I5111=100*8388608
  While (I5111 > 0)endwhile
  ms0,mi198=$97c09D     ;set X:$C09D,23 equal to 1 for PFM mode
  I5111=100*8388608
  While (I5111 > 0)endwhile
  ms0,mi199=1
  I5111=100*8388608
  While (I5111 > 0)endwhile
  Disable PLC1
Close
```

;*** Set M-variable to the two 16-bit registers used at Ultralite

```
M3000->X:$78421,8,16,s  ;Point to 1st 16-bit node register of Node2
M3001->X:$78422,8,16,s  ;Point to 2nd 16-bit node register of Node2
```

After it is downloaded to the system, write directly to the PFM output using M3000 and M3001.

Using the JDISP Port

The JDISP port on the 16-Axis MACRO CPU can be used to display messages at the station with the use of a 40x2 character display. By default, the unit will not write to this port. To use the port, set MI13 to 1 for an LCD-style display (Acc-12A) or set MI13 to 2 for a vacuum fluorescent style display (Acc-12C). Setting MI13 to zero (default) will disable the port.

The display buffer is located in the 16-Axis MACRO CPU memory at:

```
Y: $00210 - $0025F      ; Display 80 Character Output buffer
```

The easiest way to work with this feature is to use MM-Variables to point to the locations at the MACRO CPU. Write to these locations using MS commands, use the MACRO PLCCs or download the values directly while in MACRO-ASCII mode. The display characters are written using standard 8-bit ASCII codes.

MACRO ASCII Communication Mode

MACRO ASCII Communication Mode allows direct access to the MACRO Device. This mode of communication allows our Master controller to setup all MACRO devices in the ring one at a time using the Ring Order Method. One other benefit to this method of communications is that it allows direct communication to the MACRO device without having to issue MS commands as in the traditional PEWIN Terminal window.

Usually, the MACRO ASCII Communication Mode is the mode that is used to set up the MM-variable definitions, set MP-variables, and down load the MACRO PLCC programs.

At a minimum, set the following I-variables to enable MACRO ASCII mode communications.

```
I6840=$4030       ;to enable MACRO IC0 as sync-master and node 14 for
                  ;auxiliary communications
I6841=$0FCxxx     ;to enable node 15 and 14.  If activating nodes 0,1,4,5
                  ;we would set I6841=$0FC033
I79=32            ;Timeout value for Node 14 Auxiliary communications
```

If using more than one MACRO IC, then set up I6890, I6891, I6940, I6941, I6990, and I6991 appropriately. Once the communication variables are modified, they must be saved to the memory of the controller with the save command and then reset the controller with either a **$$$** command or power cycle the controller.

Note:

The PMAC Controller will be able to communicate to the MACRO Device in MACRO ASCII communication mode after the unit has been restarted with the changes saved to its memory.

How to Enable and Disable MACRO ASCII Communication Mode

To start the MACRO ASCII Mode, issue the **MACSTAn** (n stands for the assigned station number for the device) command to the device in the ring. In many cases, there will be one device only and a number may not be assigned to the device. In that case, use the **MACSTA255** or **MACSTA0** commands. The actual number that is assigned to the device resides in MI11 of the MACRO Device and the default value is 0. If there are multiple MACRO devices in the ring and communication is in MACRO ASCII mode, set up the systems with the Ring Order Method and assign station numbers to each device. If the assigned station number is not known, check MI11.

Once in MACRO ASCII Mode, communicate to the MACRO device is done directly. To change an MI-variable or create an MM-variable definition, write directly to the Variable in the terminal window.

```
MI996=$0F803F           ;To activate Nodes 0,1,2,3,4,5 at the MACRO Device
MM80->X:$8001,0,24,s    ;Set M80 to the Counter register of channel 1
```

To exit or disable MACRO ASCII Communication mode, issue the **<CTRL>T** command.

Note:

MACSTA255 command will look for the first MACRO device that does not have have a station number assigned to it (MI11=0). As soon as MI11 is changed to a value greater than zero, then it will look immediately for the next device with MI11 set to zero.

The Ring Order Method

The Ring Order Method has been developed to allow MACRO Devices to be set up with software. Since the MACRO CPU has hardware switches (SW1 and SW2) to activate nodes and assign it to a master, the ring order method is not necessary. However, if the MACRO CPU is used with other MACRO devices like the Geo MACRO Drives or MACRO Peripheral Devices (Acc-65M, Acc-24E2M, etc.), then it is necessary to know how the Ring Order Method is works.

To initiate the Ring Order Method, start with the new hardware and then enable the MACRO ASCII Communication Mode by typing **MACSTA255** in the terminal window. At this point, the Software Interface will seek the first device that has not been setup (i.e. MI11=0). Once communicating with the device, activate the nodes with MI996 and set up any critical MI-variables that need to be set for the application. Upon completion of these MI-variable settings, assign a Station Number to the device with the **STN=n** command where n can be set from 1 to 254. As soon a station number is assigned to the device, the system will look for the next device that has not been set up (MI11=0). If assigning a MACRO device as Station Number 20, type **STA=20** in the terminal window and MI11 will be set to 20.

Using MM-Variables to Verify MACRO Station Memory Locations

The MACRO M-Variables MM0 through MM511 can be used to look at any MACRO Station memory location. This can be useful especially when trying to test the hardware at the MACRO CPU that does not have MI-Variables associated with it. The MM-variables can be used in a similar fashion to the traditional PMAC M-variables. The major difference between the MACRO M-Variables and the PMAC M-variables is that the MM-Variable definitions are downloaded while in the MACRO ASCII communications mode.

The best method of setting up MM-variables is to open a text file and create the definitions. Next, save the file for future use or modifications. To download the definitions to the MACRO CPU, set Communications mode to MACRO ASCII and then download the file.

Example: Read Using MM-Variables – Actual Encoder Read from Gate Array

Read the actual encoder up/down counters from the gate array on the Acc-24E2A for channel 1, 2, 3, and 4. This data is found on bits 0-23 of X register of the following locations.

Chan #	1	2	3	4	5	6	7	8
Hex	[$8001]	[$8009]	[$8011]	[$8019]	[$8041]	[$8049]	[$8051]	[$8059]

File Definitions:

```
MM81->X:$8001,0,24,s          ;Counter Register for Channel 1
MM82->X:$8009,0,24,s          ;Counter Register for Channel 2
MM83->X:$8011,0,24,s          ;Counter Register for Channel 3
MM84->X:$8019,0,24,s          ;Counter Register for Channel 4
```

After downloading the definitions to the MACRO CPU, access the data using standard MS commands. For example:

```
MS0,MM81                ;will return the value to the Terminal window
MSR0,MM81,P81           ;will copy the value into P81 (this can be used also
                        ;in Programs in the Utralite)
```

Data Transfer Examples (MI20-MI68)

The general purpose data transfer (MI20 through MI68) variables for MACRO allow the transfer of data to or from any location at the MACRO Station from or to the MACRO IO node registers. Details of the setup can be found in the 16-Axis MACRO CPU Software Reference Manual.

This section shows examples of how to set up the transfers for the following:

- Read Actual DAC output from Servo IC register
- Read Servo IC Up/Down Counter Position
- Write to the DACnB output on Acc-24E2A

Example: Read DAC Output from Servo IC Card

Transfer DAC1A, DAC2A, DAC3A, and DAC4A to Ultralite using the MACRO Data Transfer. Assume that the MACRO Station 0 is enabled and that the data is from Acc-24E2A from channels 1,2,3,4. Since the ADC data is 16-bit data, the most efficient method of transfer is through the MACRO 16-bit data registers from nodes 2 and 3.

MACRO Station Setup

MACRO Commands	Notes
`MS0,MI19=$4`	Transfer data once every 4 phase clocks (servo default)
`MS0,MI975=$C`	Activate first I/O nodes 2 and 3 at Station
`MS0,MI20=$F`	Transfer MI21, MI22,MI23, and MI24
`MS0,MI21=$6D8002DDC0A1`	Signed upper 16-bits data from Station address Y:$8002 to X:$C0A1 (node2)
`MS0,MI22=$6D800ADCC0A2`	Signed upper 16-bits data from Station address Y:$800A to X:$C0A2 (node2)
`MS0,MI23=$6D8012DCC0A3`	Signed upper 16-bits data from Station address Y:$8012 to X:$C0A3 (node2)
`MS0,MI24=$6D801ADCC0A5`	Signed upper 16-bits data from Station address Y:$801A to X:$C0A5 (node3)
`MSSAVE0`	Save these changes to the MACRO Sation
`MS$$$0`	Reset the MACRO Station for changes to take affect

Ultralite Setup

Ultralite (8 Axis)	Turbo Ultralite (8 Axis)	Description
`I996=$0FB33F`	`I6841=$0FB33F`	Enable nodes 0,1,2,3,4,5,8,9,12, & 13 at Ultralite
`M980->X:$C0A1,8,16,s`	`M980->X:$78421,8,16,s`	DAC1A, 1st 16 bit word node2
`M981->X:$C0A2,8,16,s`	`M981->X:$78422,8,16,s`	DAC2A, 2nd 16 bit word node 2
`M982->X:$C0A3,8,16,s`	`M982->X:$78423,8,16,s`	DAC3A, 3rd 16 bit word node 2
`M983->X:$C0A5,8,16,s`	`M983->X:$78425,8,16,s`	DAC4A, 1st 16 bit word node 3

Now the M-Variables in the PLC or motion programs can be used for data acquisition purposes.

Example: Monitor Up/Down Counter from Servo IC Card

This example shows how to monitor the up/down counter of the Servo IC on the Acc-24E2x family of cards. Monitoring this data is a method also used to verify the operation of the encoder because it does not rely on the PMAC to process the encoder data and display in the Position Window of the Pewin Pro Program.

Transfer Counter1, Counter2, Counter3, and Counter4 to the Ultralite using the MACRO Data Transfer (Assume MACRO Station 0). Assume Acc-24E2A from channels 1, 2, 3, and 4.

Since the counter data is 24-bit data, the most efficient method of transfer is through the MACRO 24-bit data registers from nodes 2, 3, 6, 7.

MACRO Station Setup

MACRO Commands	Notes
`MS0,MI19=$4`	Transfer data once every four phase clocks (servo default)
`MS0,MI975=$C`	Activate first I/O nodes 2 and 3 at Station
`MS0,MI20=$F`	Transfer MI21, MI22,MI23, and MI24
`MS0,MI21=$E98001E9C0A0`	Signed upper 24-bits data from Station address X:$8001 to X:$C0A0 (node2)
`MS0,MI21=$E98009E9C0A4`	Signed upper 24-bits data from Station address X:$8009 to X:$C0A4 (node3)
`MS0,MI22=$E98011E9C0A8`	Signed upper 24-bits data from Station address X:$8011 to X:$C0A8 (node6)
`MS0,MI23=$E98019E9C0AC`	Signed upper 24-bits data from Station address X:$8019 to X:$C0AC (node7)
`MSSAVE0`	Save these changes to the MACRO Sation
`MS$$$0`	Reset the MACRO Station for changes to take affect

Ultralite Setup

Ultralite (8 Axis)	Turbo Ultralite (8 Axis)	Description
I996=$0FB3FF	I6841=$0FB3FF	Enable nodes 0,1,2,3,4,5,6,7,8,9,12, & 13 at Ultralite
M980->X:$C0A0,0,24,s	M980->X:$78420,0,24,s	Up/Down Counter 1
M981->X:$C0A4,0,24,s	M981->X:$78424,0,24,s	Up/Down Counter 2
M982->X:$C0A8,0,24,s	M982->X:$78428,0,24,s	Up/Down Counter 3
M983->X:$C0AC,0,24,s	M983->X:$7842C,0,24,s	Up/Down Counter 4

Now the M-Variables in the PLC or motion programs can be used for data acquisition purposes.

Example: Write to DACnB on Servo IC Card

Transfer DAC1B, DAC2B, DAC3B, and DAC4B to the Ultralite using the MACRO Data Transfer (Assume MACRO Station 0). Assume Acc-24E2A from channels 1,2,3,4.

Since the ADC data is 16-bit data, the most efficient method of transfer is through the MACRO 16-bit data registers from nodes 2 and 3.

MACRO Station Setup

MACRO Commands	Notes
MS0,MI19=$4	Transfer data once every 4 phase clocks (servo default)
MS0,MI975=$C	Activate first I/O nodes 2 and 3 at Station
MS0,MI20=$F	Transfer MI21, MI22,MI23, and MI24
MS0,MI21=$DDC0A16D8003	Signed upper 16-bits data from Station address X:$C0A1 (node 2) to Y:$8003
MS0,MI22=$DDC0A26D800B	Signed upper 16-bits data from Station address X:$C0A2 (node 2) to Y:$800B
MS0,MI23=$DDC0A36D8013	Signed upper 16-bits data from Station address X:$C0A3 (node 2) to Y:$8013
MS0,MI24=$DDC0A56D801B	Signed upper 16-bits data from Station address X:$C0A5 (node 3) to Y:$801B
MSSAVE0	Save these changes to the MACRO Station
MS$$$0	Reset the MACRO Station for changes to take affect

Ultralite Setup

Ultralite (8 Axis)	Turbo Ultralite (8 Axis)	Description
I996=$0FB33F	I6841=$0FB33F	Enable nodes 0,1,2,3,4,5,8,9,12, & 13 at Ultralite
M980->X:$C0A1,8,16,s	M980->X:$78421,8,16,s	DAC1B, 1st 16 bit word node 2
M981->X:$C0A2,8,16,s	M981->X:$78422,8,16,s	DAC2B, 2nd 16 bit word node 2
M982->X:$C0A3,8,16,s	M982->X:$78423,8,16,s	DAC3B, 3rd 16 bit word node 2
M983->X:$C0A5,8,16,s	M983->X:$78425,8,16,s	DAC4B, 1st 16 bit word node 3

Now the M-Variables in the PLC or motion programs can be used to write to the DACnB registers.

Using MI198 and MI199 to Verify MACRO Station Memory Locations

The MACRO I-Variables MI198 and MI199 can be used to look at any MACRO Station memory location. This can be useful especially when trying to test the hardware at the MACRO Station. MI198 contains the register to be read, and the information in MI198 can be read by querying MI199. Users of the older MACRO8 (602804) CPU might be familiar with this method of simple non-synchronous data transfer. Since the 16-Axis MACRO CPU supports MM-Variable reads and writes, using the MM-variables might be easier and more flexible than using of MI198 and MI199.

MSn,MI198 is a 24-bit register where the lower 16 bits have the address and the upper eight bits contain the number of bits and tell us whether it is an X or Y memory address.

MSn,MI199 will respond back with the value in MS0,MI198.

Example: Read Using MI198 and MI199 – Direct Hall Effect Read

Read the T, U, V, W inputs directly at the gate array on the Acc-24E2 for channel 1. This data is found on bits 20-23 of X:$8000 at the MACRO Station.

```
MS0,MI198=$E88000 ;read X:$8000,0,24
MS0,MI199               ;Request Data
$000000AFDB87           ;TUVW = $A (1010 binary) from bits 20-23
```

To write this data to a PMAC variable (P,M,or Q), use an MSRn,MI199,{pmac variable} on-line or buffer command.

```
MSR0,MI199,P1000        ;writes the value of X:$C000,0,24 to P1000
```

Then use P1000 to query the TUVW signals from the MACRO Station.

Example: Read Using MI198 and MI199 – Actual DAC Read

Read the actual DAC Outputs from the gate array on the Acc-24E2A for channel 1. This data is found on bits 8-23 of Y:$8002 at the MACRO Station.

Chan #	1	2	3	4	5	6	7	8
Hex	[$8002]	[$800A]	[$8012]	[$801A]	[$8042]	[$804A]	[$8052]	[$805A]

```
MS0,MI198=$6D8002       ;read Y:$8002,8,16,s
MS0,MI199               ;Request Data
$0000000C35             ;DAC out equals 3125 DAC bits (1V)
```

Hardware Re-initialization of MACRO CPU

MACRO hardware re-initialization to factory defaults is enabled when the SW1 setting is set to 15 or F (hexadecimal) and the power is cycled at the MACRO Station. The only time this should be used with the MACRO Station would be if the MACRO Station always powers up with a watchdog (typically if the ring clock at the Ultralite is different than the ring clock at the MACRO Station). Node 11 will be the only MACRO Station node enabled. Therefore, enable node 11 of the MACRO IC at the Ultralite to communicate to the MACRO Station.

Turbo Ultralite Example: Servo nodes 0,1,4,5 enabled at Ultralite (I6841=$0F8033)

1. Enable node 11, I996=$0F8833.
2. Re-establish communications with MS11,(MIvar) commands.
3. Issue **`MS$$$***11`** to ensure re-initialization.
4. Issue **`MSSAVE11`** command to save the factory defaults to the station.

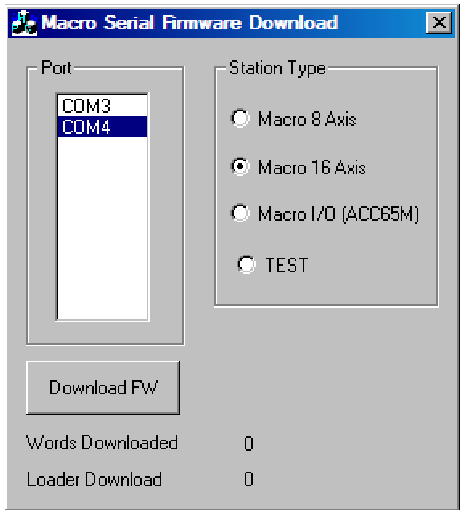
Firmware Updates

Downloading new firmware to the MACRO Station is a simple process once the MACRO board is set up properly. To download new firmware to the MACRO station, obtain the following items:

- Two jumpers
- DB9 Male Serial Port
- MACRO Firmware Download Software (MacroFWDown.exe)
- New firmware file (MiniMac2.bin)

To download the software to the MACRO station:

1. Copy the firmware into a directory (C:\Macro\Firmware).
2. Pull the MACRO CPU out of the 3U rack and then jumper the E2 (1-2) and E1 (1-2).
3. Plug MACRO CPU into a 3U rack.
4. Place the serial cable to the J4 serial connection on the MACRO-station and place the other end to the COM port on the PC.
5. Power up the 3U rack and then launch MacroFWDown.exe. Choose the com port and select **16-Axis Macro** and then press the **Download FW** button.



6. After the download is complete, power down the system and remove jumper E1 and place jumper E2 from 2-3.

MACRO Flag Transfer Location

For proper servo operations, the Master Controller must process information in real time. For MACRO systems, this information is brought to the Master via Ixx25. For Turbo systems, the locations are at $3440, $3441, $3444, $3445, $3448, $3449, $344C, $344D, etc. The following tables list the data that is transferred through these locations.

NodeCntrlStatus: (Sent by Slave - Located in gate array except for bits 0-7)
(Turbo at: X:$3440 .. X:$347F)

Bit	Function
0	Not Used
1	Not Used
2	Not Used
3	Not Used
4	Not Used
5	Not Used
6	Not Used
7	Not Used
8	Encoder Count Error
9	Position compare (EQUn)output
10	Position captured-on-gated -index flag.
11	*** Position Captured (Triggered Event Occurred) Flag.**
12	**A Power On Reset POR has occurred.**
13	**This Node detected a MACRO Ring Break (MRB).**
14	**Amplifier Enabled.**
15	*** Amplifier or Station Node shutdown Fault.**
16	Home Flag(HMFLn) Input Value
17	*** Positive End Limit Flag (PILMn) Input Value.**
18	*** Negative End Limit Flag (NILMn) Input Value**
19	Fast User Status Flag(UserSatus1) or USERn Input Value if have PMAC Gate Array
20	Fast User Status Flag(UserSatus2) or FlgWn Input Value if have PMAC Gate Array
21	Fast User Status Flag(UserSatus3) or FlgVn Input Value if have PMAC Gate Array
22	Fast User Status Flag(UserSatus4) or FlagUn Input Value if have PMAC Gate Array
23	Fast User Status Flag(UserSatus5) or FlagTn Input Value if have PMAC Gate Array
Note: The items in bold are reserved and defined flag locations.	

NodeCntrlCmd: (Sent by Master - Located in Turbo at: Y:$3440 .. Y:$347F / Y-portion of Flag Address)

Bit	Function
0	*** Position Capture (Triggered Event) Enable Flag**
1	Not Used
2	Not Used
3	Not Used
4	Not Used
5	Not Used
6	Not Used
7	Not Used
8	Reserved for future ring protocol control
9	Reserved for future ring protocol control
10	Reserved for future ring protocol control
11	*** Position Capture (Triggered Event) Enable Flag**
12	Node Reset Command
13	**This Slave detected a MACRO Ring Break (MRB) & became a Synchronizing Master**
14	*** Real-time Data or Amp. Enable**
15	**When B13 = 1 then B15 = 1 & is a Station Fault.**
16	Reserved for future ring protocol control
17	Reserved for future ring protocol control
18	Reserved for future ring protocol control
19	Fast User Defined Command Flag (UserCmd1)
20	Fast User Defined Command Flag (UserCmd2)
21	Fast User Defined Command Flag (UserCmd3)
22	Fast User Defined Command Flag (UserCmd4)
23	Fast User Defined Command Flag (UserCmd5)
Note: The items in bold are reserved and defined flag locations.	

MACRO STATION TYPE 1 PROTOCOLS

The 16-Axis MACRO CPU, as a multi-node station, implements the Type 1 MACRO protocol. In this protocol, all four registers in each node are used for real-time communications; Node 15 is used for auxiliary communications for the entire station through the MS auxiliary communications commands.

The Type 1 protocol uses the MACRO node registers for each mode of operation as explained below:

Velocity/Torque Mode

Node Register #	0 (24 bits)	1 (16 bits)	2 (16 bits)	3 (16 bits)
Command	Velocity/Torque Command	(Reserved)	(Reserved)	Motor Command Flags
Feedback	Position Feedback	(Reserved)	(Reserved)	Motor Status Flags

Phase Current (Sinewave) Mode

Node Register #	0 (24 bits)	1 (16 bits)	2 (16 bits)	3 (16 bits)
Command	Phase A Current Command	Phase B Current Command	(Reserved)	Motor Command Flags
Feedback	Position Feedback	(Reserved)	(Reserved)	Motor Status Flags

Phase Voltage (Direct PWM) Mode

Node Register #	0 (24 bits)	1 (16 bits)	2 (16 bits)	3 (16 bits)
Command	Phase A Voltage Command	Phase B Voltage Command	Phase C Voltage Command	Motor Command Flags
Feedback	Position Feedback	Phase A Current Feedback	Phase B Current Feedback	Motor Status Flags

All of the 16-bit registers appear in the top 16 bits of the 24-bit word on both PMAC/PMAC2 and the 16-Axis MACRO CPU.

MACRO EQUIVALENT SERVO IC MEMORY LOCATIONS

Most of the locations have direct real-time access or are accessed through MI-variables. All of the Status Registers (Mxx30-Mxx49), the Calculation Registers (Mxx60-Mxx89), and the Axis Definitions (M191-M194) are identical for all Turbo PMACs. The only difference for a MACRO system is the Servo IC Variables (Mxx00-Mxx28) and the table below will show the equivalent location for these suggested M-variables. For access to the locations that do not have direct reads, create MM-variable definitions for them.

Servo IC 0 Registers for Channel 1 (usually for Motor #1)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8001,0,24,S	-	-	M101	ENC1 24-bit counter position
Y:$8002,8,16,S	-	-	M102	OUT1A command value; DAC or PWM
X:$8003,0,24,S	MS0,MI921	-	M103	ENC1 captured position
Y:$8003,8,16,S	-	-	M104	OUT1B command value; DAC or PWM
Y:$8005,8,16,s	MS0,MI922	Y:$78421,8,16,s	M105	ADC1A input value
Y:$8006,8,16,s	MS0,MI924	Y:$78422,8,16,s	M106	ADC1B input value
Y:$8004,8,16,s	-	-	M107	OUT1C command value; PFM or PWM
Y:$8007,0,24,s	MS0,MI925	-	M108	ENC1 compare A position
X:$8007,0,24,s	MS0,MI926	-	M109	ENC1 compare B position
X:$8006,0,24,s	MS0,MI923	-	M110	ENC1 compare auto-increment value
X:$8005,11	MS0,MI928	-	M111	ENC1 compare initial state write enable
X:$8005,12	MS0,MI929	-	M112	ENC1 compare initial state
X:$8005,14	-	Y:$003440,14	M114	AENA1 output status
X:$8000,19	-	Y:$003440,19	M115	USER1 flag input status
X:$8000,9	-	Y:$003440,9	M116	ENC1 compare output value
X:$8000,11	-	Y:$003440,11	M117	ENC1 capture flag
X:$8000,8	-	Y:$003440,8	M118	ENC1 count error flag
X:$8000,14	-	-	M119	CHC1 input status
X:$8000,16	-	Y:$003440,16	M120	HMFL1 flag input status
X:$8000,17	-	Y:$003440,17	M121	PLIM1 flag input status
X:$8000,18	-	Y:$003440,18	M122	MLIM1 flag input status
X:$8000,15	-	Y:$003440,15	M123	FAULT1 flag input status
X:$8000,20	-	Y:$003440,20	M124	Channel 1 W flag input status
X:$8000,21	-	Y:$003440,21	M125	Channel 1 V flag input status
X:$8000,22	-	Y:$003440,22	M126	Channel 1 U flag input status
X:$8000,23	-	Y:$003440,23	M127	Channel 1 T flag input status
X:$8000,20,4	-	Y:$003440,20,4	M128	Channel 1 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM10->X:$8001,0,24,s          ; ENC1 24-bit counter position
MM11->Y:$8002,8,16,s          ; OUT1A command value
MM12->Y:$8003,8,16,s          ; OUT1B command value
MM13->Y:$8004,8,16,s          ; OUT1C command value
```

Servo IC 0 Registers for MACRO Channel 2 (usually for Motor #2)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8009,0,24,S	-	-	M201	ENC2 24-bit counter position
Y:$800A,8,16,S	-	-	M202	OUT2A command value; DAC or PWM
X:$800B,0,24,S	MS1,MI921	-	M203	ENC2 captured position
Y:$800B,8,16,S	-	-	M204	OUT2B command value; DAC or PWM
Y:$800D,8,16,s	MS1,MI922	Y:$78425,8,16,s	M205	ADC2A input value
Y:$800E,8,16,s	MS1,MI924	Y:$78426,8,16,s	M206	ADC2B input value
Y:$800C,8,16,s	-	-	M207	OUT2C command value; PFM or PWM
Y:$800F,0,24,s	MS1,MI925	-	M208	ENC2 compare A position
X:$800F,0,24,s	MS1,MI926	-	M209	ENC2 compare B position
X:$800E,0,24,s	MS1,MI923	-	M210	ENC2 compare auto-increment value
X:$800D,11	MS1,MI928	-	M211	ENC2 compare initial state write enable
X:$800D,12	MS1,MI929	-	M212	ENC2 compare initial state
X:$800D,14	-	Y:$3441,14	M214	AENA2 output status
X:$8008,19	-	Y:$3441,19	M215	USER2 flag input status
X:$8008,9	-	Y:$3441,9	M216	ENC2 compare output value
X:$8008,11	-	Y:$3441,11	M217	ENC2 capture flag
X:$8008,8	-	Y:$3441,8	M218	ENC2 count error flag
X:$8008,14	-	-	M219	CHC2 input status
X:$8008,16	-	Y:$003441,16	M220	HMFL2 flag input status
X:$8008,17	-	Y:$003441,17	M221	PLIM2 flag input status
X:$8008,18	-	Y:$003441,18	M222	MLIM2 flag input status
X:$8008,15	-	Y:$003441,15	M223	FAULT2 flag input status
X:$8008,20	-	Y:$003441,20	M224	Channel 2 W flag input status
X:$8008,21	-	Y:$003441,21	M225	Channel 2 V flag input status
X:$8008,22	-	Y:$003441,22	M226	Channel 2 U flag input status
X:$8008,23	-	Y:$003441,23	M227	Channel 2 T flag input status
X:$8008,20,4	-	Y:$003441,20,4	M228	Channel 2 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM20->X:$8009,0,24,s          ; ENC2 24-bit counter position
MM21->Y:$800A,8,16,s          ; OUT2A command value
MM22->Y:$800B,8,16,s          ; OUT2B command value
MM23->Y:$800C,8,16,s          ; OUT2C command value
```

Servo IC 0 Registers for MACRO Channel 3 (usually for Motor #3)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8011,0,24,S	-	-	M301	ENC3 24-bit counter position
Y:$8012,8,16,S	-	-	M302	OUT3A command value; DAC or PWM
X:$8013,0,24,S	MS4,MI921	-	M303	ENC3 captured position
Y:$8013,8,16,S	-	-	M304	OUT3B command value; DAC or PWM
Y:$8015,8,16,s	MS4,MI922	Y:$78429,8,16,s	M305	ADC3A input value
Y:$8016,8,16,s	MS4,MI924	Y:$7842A,8,16,s	M306	ADC3B input value
Y:$8014,8,16,s	-	-	M307	OUT3C command value; PFM or PWM
Y:$8017,0,24,s	MS4,MI925	-	M308	ENC3 compare A position
X:$8017,0,24,s	MS4,MI926	-	M309	ENC3 compare B position
X:$8016,0,24,s	MS4,MI923	-	M310	ENC3 compare auto-increment value
X:$8015,11	MS4,MI928	-	M311	ENC3 compare initial state write enable
X:$8015,12	MS4,MI929	-	M312	ENC3 compare initial state
X:$8015,14	-	Y:$003444,14	M314	AENA3 output status
X:$8010,19	-	Y:$003444,19	M315	USER3 flag input status
X:$8010,9	-	Y:$3444,9	M316	ENC3 compare output value
X:$8010,11	-	Y:$3444,11	M317	ENC3 capture flag
X:$8010,8	-	Y:$3444,8	M318	ENC3 count error flag
X:$8010,14	-	-	M319	CHC3 input status
X:$8010,16	-	Y:$003444,16	M320	HMFL3 flag input status
X:$8010,17	-	Y:$003444,17	M321	PLIM3 flag input status
X:$8010,18	-	Y:$003444,18	M322	MLIM3 flag input status
X:$8010,15	-	Y:$003444,15	M323	FAULT3 flag input status
X:$8010,20	-	Y:$003444,20	M324	Channel 3 W flag input status
X:$8010,21	-	Y:$003444,21	M325	Channel 3 V flag input status
X:$8010,22	-	Y:$003444,22	M326	Channel 3 U flag input status
X:$8010,23	-	Y:$003444,23	M327	Channel 3 T flag input status
X:$8010,20,4	-	Y:$003444,20,4	M328	Channel 3 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM30->X:$8011,0,24,s          ; ENC3 24-bit counter position
MM31->Y:$8012,8,16,s          ; OUT3A command value
MM32->Y:$8013,8,16,s          ; OUT3B command value
MM33->Y:$8014,8,16,s          ; OUT3C command value
```

Servo IC 0 Registers for MACRO Channel 4 (usually for Motor #4)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8019,0,24,S	-	-	M401	ENC4 24-bit counter position
Y:$801A,8,16,S	-	-	M402	OUT4A command value; DAC or PWM
X:$801B,0,24,S	MS5,MI921	-	M403	ENC4 captured position
Y:$801B,8,16,S	-	-	M404	OUT4B command value; DAC or PWM
Y:$801D,8,16,s	MS5,MI922	Y:$7842D,8,16,s	M405	ADC4A input value
Y:$801E,8,16,s	MS5,MI924	Y:$7842E,8,16,s	M406	ADC4B input value
Y:$801C,8,16,s	-	-	M407	OUT4C command value; PFM or PWM
Y:$801F,0,24,s	MS5,MI925	-	M408	ENC4 compare A position
X:$801F,0,24,s	MS5,MI926	-	M409	ENC4 compare B position
X:$801E,0,24,s	MS5,MI923	-	M410	ENC4 compare auto-increment value
X:$801D,11	MS5,MI928	-	M411	ENC4 compare initial state write enable
X:$801D,12	MS5,MI929	-	M412	ENC4 compare initial state
X:$801D,14	-	Y:$3445,14	M414	AENA4 output status
X:$8018,19	-	Y:$3445,19	M415	USER4 flag input status
X:$8018,9	-	Y:$3445,9	M416	ENC4 compare output value
X:$8018,11	-	Y:$3445,11	M417	ENC4 capture flag
X:$8018,8	-	Y:$3445,8	M418	ENC4 count error flag
X:$8018,14	-	-	M419	CHC4 input status
X:$8018,16	-	Y:$003445,16	M420	HMFL4 flag input status
X:$8018,17	-	Y:$003445,17	M421	PLIM4 flag input status
X:$8018,18	-	Y:$003445,18	M422	MLIM4 flag input status
X:$8018,15	-	Y:$003445,15	M423	FAULT4 flag input status
X:$8018,20	-	Y:$003445,20	M424	Channel 4 W flag input status
X:$8018,21	-	Y:$003445,21	M425	Channel 4 V flag input status
X:$8018,22	-	Y:$003445,22	M426	Channel 4 U flag input status
X:$8018,23	-	Y:$003445,23	M427	Channel 4 T flag input status
X:$8018,20,4	-	Y:$003445,20,4	M428	Channel 4 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM40->X:$8019,0,24,s          ; ENC4 24-bit counter position
MM41->Y:$801A,8,16,s          ; OUT4A command value
MM42->Y:$801B,8,16,s          ; OUT4B command value
MM43->Y:$801C,8,16,s          ; OUT4C command value
```

Servo IC 1 Registers for Channel 1 (usually for Motor #5)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8041,0,24,S	-	-	M501	ENC5 24-bit counter position
Y:$8042,8,16,S	-	-	M502	OUT5A command value; DAC or PWM
X:$8043,0,24,S	MS8,MI921	-	M503	ENC5 captured position
Y:$8043,8,16,S	-	-	M504	OUT5B command value; DAC or PWM
Y:$8045,8,16,s	MS8,MI922	Y:$78431,8,16,s	M505	ADC5A input value
Y:$8046,8,16,s	MS8,MI924	Y:$78432,8,16,s	M506	ADC5B input value
Y:$8044,8,16,s	-	-	M507	OUT5C command value; PFM or PWM
Y:$8047,0,24,s	MS8,MI925	-	M508	ENC5 compare A position
X:$8047,0,24,s	MS8,MI926	-	M509	ENC5 compare B position
X:$8046,0,24,s	MS8,MI923	-	M510	ENC5 compare auto-increment value
X:$8045,11	MS8,MI928	-	M511	ENC5 compare initial state write enable
X:$8045,12	MS8,MI929	-	M512	ENC5 compare initial state
X:$8045,14	-	Y:$003448,14	M514	AENA5 output status
X:$8040,19	-	Y:$003448,19	M515	USER5 flag input status
X:$8040,9	-	Y:$003448,9	M516	ENC5 compare output value
X:$8040,11	-	Y:$003448,11	M517	ENC5 capture flag
X:$8040,8	-	Y:$003448,8	M518	ENC5 count error flag
X:$8040,14	-	-	M519	CHC5 input status
X:$8040,16	-	Y:$003448,16	M520	HMFL5 flag input status
X:$8040,17	-	Y:$003448,17	M521	PLIM5 flag input status
X:$8040,18	-	Y:$003448,18	M522	MLIM5 flag input status
X:$8040,15	-	Y:$003448,15	M523	FAULT5 flag input status
X:$8040,20	-	Y:$003448,20	M524	Channel 5 W flag input status
X:$8040,21	-	Y:$003448,21	M525	Channel 5 V flag input status
X:$8040,22	-	Y:$003448,22	M526	Channel 5 U flag input status
X:$8040,23	-	Y:$003448,23	M527	Channel 5 T flag input status
X:$8040,20,4	-	Y:$003448,20,4	M528	Channel 5 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM50->X:$8041,0,24,s          ; ENC5 24-bit counter position
MM51->Y:$8042,8,16,s          ; OUT5A command value
MM52->Y:$8043,8,16,s          ; OUT5B command value
MM53->Y:$8044,8,16,s          ; OUT5C command value
```

Servo IC 1 Registers for Channel 2 (usually for Motor #6)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8049,0,24,S	-	-	M601	ENC6 24-bit counter position
Y:$804A,8,16,S	-	-	M602	OUT6A command value; DAC or PWM
X:$804B,0,24,S	MS9,MI921	-	M603	ENC6 captured position
Y:$804B,8,16,S	-	-	M604	OUT6B command value; DAC or PWM
Y:$804D,8,16,s	MS9,MI922	Y:$78435,8,16,s	M605	ADC6A input value
Y:$804E,8,16,s	MS9,MI924	Y:$78436,8,16,s	M606	ADC6B input value
Y:$804C,8,16,s	-	-	M607	OUT6C command value; PFM or PWM
Y:$804F,0,24,s	MS9,MI925	-	M608	ENC6 compare A position
X:$804F,0,24,s	MS9,MI926	-	M609	ENC6 compare B position
X:$804E,0,24,s	MS9,MI923	-	M610	ENC6 compare auto-increment value
X:$804D,11	MS9,MI928	-	M611	ENC6 compare initial state write enable
X:$804D,12	MS9,MI929	-	M612	ENC6 compare initial state
X:$804D,14	-	Y:$3449,14	M614	AENA6 output status
X:$8048,19	-	Y:$3449,19	M615	USER6 flag input status
X:$8048,9	-	Y:$3449,9	M616	ENC6 compare output value
X:$8048,11	-	Y:$3449,11	M617	ENC6 capture flag
X:$8048,8	-	Y:$3449,8	M618	ENC6 count error flag
X:$8048,14	-	-	M619	CHC6 input status
X:$8048,16	-	Y:$003449,16	M620	HMFL6 flag input status
X:$8048,17	-	Y:$003449,17	M621	PLIM6 flag input status
X:$8048,18	-	Y:$003449,18	M622	MLIM6 flag input status
X:$8048,15	-	Y:$003449,15	M623	FAULT6 flag input status
X:$8048,20	-	Y:$003449,20	M624	Channel 6 W flag input status
X:$8048,21	-	Y:$003449,21	M625	Channel 6 V flag input status
X:$8048,22	-	Y:$003449,22	M626	Channel 6 U flag input status
X:$8048,23	-	Y:$003449,23	M627	Channel 6 T flag input status
X:$8048,20,4	-	Y:$003449,20,4	M628	Channel 6 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM60->X:$8049,0,24,s          ; ENC6 24-bit counter position
MM61->Y:$804A,8,16,s          ; OUT6A command value
MM62->Y:$804B,8,16,s          ; OUT6B command value
MM63->Y:$804C,8,16,s          ; OUT6C command value
```

Servo IC 1 Registers for Channel 3 (usually for Motor #7)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8051,0,24,S	-	-	M701	ENC7 24-bit counter position
Y:$8052,8,16,S	-	-	M702	OUT7A command value; DAC or PWM
X:$8053,0,24,S	MS12,MI921	-	M703	ENC7 captured position
Y:$8053,8,16,S	-	-	M704	OUT7B command value; DAC or PWM
Y:$8055,8,16,s	MS12,MI922	Y:$78329,8,16,s	M705	ADC7A input value
Y:$8056,8,16,s	MS12,MI924	Y:$7832A,8,16,s	M706	ADC7B input value
Y:$8054,8,16,s	-	-	M707	OUT7C command value; PFM or PWM
Y:$8057,0,24,s	MS12,MI925	-	M708	ENC7 compare A position
X:$8057,0,24,s	MS12,MI926	-	M709	ENC7 compare B position
X:$8056,0,24,s	MS12,MI923	-	M710	ENC7 compare auto-increment value
X:$8055,11	MS12,MI928	-	M711	ENC7 compare initial state write enable
X:$8055,12	MS12,MI929	-	M712	ENC7 compare initial state
X:$8055,14	-	Y:$00344C,14	M714	AENA7 output status
X:$8050,19	-	Y:$00344C,19	M715	USER7 flag input status
X:$8050,9	-	Y:$00344C,9	M716	ENC7 compare output value
X:$8050,11	-	Y:$00344C,11	M717	ENC7 capture flag
X:$8050,8	-	Y:$00344C,8	M718	ENC7 count error flag
X:$8050,14	-	-	M719	CHC7 input status
X:$8050,16	-	Y:$00344C,16	M720	HMFL7 flag input status
X:$8050,17	-	Y:$00344C,17	M721	PLIM7 flag input status
X:$8050,18	-	Y:$00344C,18	M722	MLIM7 flag input status
X:$8050,15	-	Y:$00344C,15	M723	FAULT7 flag input status
X:$8050,20	-	Y:$00344C,20	M724	Channel 7 W flag input status
X:$8050,21	-	Y:$00344C,21	M725	Channel 7 V flag input status
X:$8050,22	-	Y:$00344C,22	M726	Channel 7 U flag input status
X:$8050,23	-	Y:$00344C,23	M727	Channel 7 T flag input status
X:$8050,20,4	-	Y:$00344C,20,4	M728	Channel 7 TUVW inputs as 4-bit value

Suggested MM-Variables

```
MM70->X:$8051,0,24,s          ; ENC7 24-bit counter position
MM71->Y:$8052,8,16,s          ; OUT7A command value
MM72->Y:$8053,8,16,s          ; OUT7B command value
MM73->Y:$8054,8,16,s          ; OUT7C command value
```

Servo IC 1 Registers for Channel 4 (usually for Motor #8)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$8059,0,24,S	-	-	M801	ENC8 24-bit counter position
Y:$805A,8,16,S	-	-	M802	OUT8A command value; DAC or PWM
X:$805B,0,24,S	MS13,MI921	-	M803	ENC8 captured position
Y:$805B,8,16,S	-	-	M804	OUT8B command value; DAC or PWM
Y:$805D,8,16,s	MS13,MI922	Y:$7843D,8,16,s	M805	ADC8A input value
Y:$805E,8,16,s	MS13,MI924	Y:$7843E,8,16,s	M806	ADC8B input value
Y:$805C,8,16,s	-	-	M807	OUT8C command value; PFM or PWM
Y:$805F,0,24,s	MS13,MI925	-	M808	ENC8 compare A position
X:$805F,0,24,s	MS13,MI926	-	M809	ENC8 compare B position
X:$805E,0,24,s	MS13,MI923	-	M810	ENC8 compare auto-increment value
X:$805D,11	MS13,MI928	-	M811	ENC8 compare initial state write enable
X:$805D,12	MS13,MI929	-	M812	ENC8 compare initial state
X:$805D,14	-	Y:$00344D,14	M814	AENA8 output status
X:$8058,19	-	Y:$00344D,19	M815	USER8 flag input status
X:$8058,9	-	Y:$00344D,9	M816	ENC8 compare output value
X:$8058,11	-	Y:$00344D,11	M817	ENC8 capture flag
X:$8058,8	-	Y:$00344D,8	M818	ENC8 count error flag
X:$8058,14	-	-	M819	CHC8 input status
X:$8058,16	-	Y:$00344D,16	M820	HMFL8 flag input status
X:$8058,17	-	Y:$00344D,17	M821	PLIM8 flag input status
X:$8058,18	-	Y:$00344D,18	M822	MLIM8 flag input status
X:$8058,15	-	Y:$00344D,15	M823	FAULT8 flag input status
X:$8058,20	-	Y:$00344D,20	M824	Channel 8 W flag input status
X:$8058,21	-	Y:$00344D,21	M825	Channel 8 V flag input status
X:$8058,22	-	Y:$00344D,22	M826	Channel 8 U flag input status
X:$8058,23	-	Y:$00344D,23	M827	Channel 8 T flag input status
X:$8058,20,4	-	Y:$00344D,20,4	M828	Channel 8 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM80->X:$8059,0,24,s          ; ENC8 24-bit counter position
MM81->Y:$805A,8,16,s          ; OUT8A command value
MM82->Y:$805B,8,16,s          ; OUT8B command value
MM83->Y:$805C,8,16,s          ; OUT8C command value
```

Servo IC 4 Registers for Channel 1 (usually for Motor #9)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9001,0,24,S	-	-	M901	ENC9 24-bit counter position
Y:$9002,8,16,S	-	-	M902	OUT9A command value; DAC or PWM
X:$9003,0,24,S	MS16,MI921	-	M903	ENC9 captured position
Y:$9003,8,16,S	-	-	M904	OUT9B command value; DAC or PWM
Y:$9005,8,16,s	MS16,MI922	Y:$79421,8,16,s	M905	ADC9A input value
Y:$9006,8,16,s	MS16,MI924	Y:$79422,8,16,s	M906	ADC9B input value
Y:$9004,8,16,s	-	-	M907	OUT9C command value; PFM or PWM
Y:$9007,0,24,s	MS16,MI925	-	M908	ENC9 compare A position
X:$9007,0,24,s	MS16,MI926	-	M909	ENC9 compare B position
X:$9006,0,24,s	MS16,MI923	-	M910	ENC9 compare auto-increment value
X:$9005,11	MS16,MI928	-	M911	ENC9 compare initial state write enable
X:$9005,12	MS16,MI929	-	M912	ENC9 compare initial state
X:$9005,14	-	Y:$003450,14	M914	AENA9 output status
X:$9000,19	-	Y:$003450,19	M915	USER9 flag input status
X:$9000,9	-	Y:$003450,9	M916	ENC9 compare output value
X:$9000,11	-	Y:$003450,11	M917	ENC9 capture flag
X:$9000,8	-	Y:$003450,8	M918	ENC9 count error flag
X:$9000,14	-	-	M919	CHC9 input status
X:$9000,16	-	Y:$003450,16	M920	HMFL9 flag input status
X:$9000,17	-	Y:$003450,17	M921	PLIM9 flag input status
X:$9000,18	-	Y:$003450,18	M922	MLIM9 flag input status
X:$9000,15	-	Y:$003450,15	M923	FAULT9 flag input status
X:$9000,20	-	Y:$003450,20	M924	Channel 9 W flag input status
X:$9000,21	-	Y:$003450,21	M925	Channel 9 V flag input status
X:$9000,22	-	Y:$003450,22	M926	Channel 9 U flag input status
X:$9000,23	-	Y:$003450,23	M927	Channel 9 T flag input status
X:$9000,20,4	-	Y:$003450,20,4	M928	Channel 9 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM90->X:$9001,0,24,s          ; ENC9 24-bit counter position
MM91->Y:$9002,8,16,s          ; OUT9A command value
MM92->Y:$9003,8,16,s          ; OUT9B command value
MM93->Y:$9004,8,16,s          ; OUT9C command value
```

Servo IC 4 Registers for Channel 2 (usually for Motor #10)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9009,0,24,S	-	-	M1001	ENC10 24-bit counter position
Y:$900A,8,16,S	-	-	M1002	OUT10A command value; DAC or PWM
X:$900B,0,24,S	MS17,MI921	-	M1003	ENC10 captured position
Y:$900B,8,16,S	-	-	M1004	OUT10B command value; DAC or PWM
Y:$900D,8,16,s	MS17,MI922	Y:$79425,8,16,s	M1005	ADC10A input value
Y:$900E,8,16,s	MS17,MI924	Y:$79426,8,16,s	M1006	ADC10B input value
Y:$900C,8,16,s	-	-	M1007	OUT10C command value; PFM or PWM
Y:$900F,0,24,s	MS17,MI925	-	M1008	ENC10 compare A position
X:$900F,0,24,s	MS17,MI926	-	M1009	ENC10 compare B position
X:$900E,0,24,s	MS17,MI923	-	M1010	ENC10 compare auto-increment value
X:$900D,11	MS17,MI928	-	M1011	ENC10 compare initial state write enable
X:$900D,12	MS17,MI929	-	M1012	ENC10 compare initial state
X:$900D,14	-	Y:$003541,14	M1014	AENA10 output status
X:$9008,19	-	Y:$003541,19	M1015	USER10 flag input status
X:$9008,9	-	Y:$003541,9	M1016	ENC10 compare output value
X:$9008,11	-	Y:$003541,11	M1017	ENC10 capture flag
X:$9008,8	-	Y:$003541,8	M1018	ENC10 count error flag
X:$9008,14	-	-	M1019	CHC10 input status
X:$9008,16	-	Y:$003541,16	M1020	HMFL10 flag input status
X:$9008,17	-	Y:$003541,17	M1021	PLIM10 flag input status
X:$9008,18	-	Y:$003541,18	M1022	MLIM10 flag input status
X:$9008,15	-	Y:$003541,15	M1023	FAULT10 flag input status
X:$9008,20	-	Y:$003541,20	M1024	Channel 10 W flag input status
X:$9008,21	-	Y:$003541,21	M1025	Channel 10 V flag input status
X:$9008,22	-	Y:$003541,22	M1026	Channel 10 U flag input status
X:$9008,23	-	Y:$003541,23	M1027	Channel 10 T flag input status
X:$9008,20,4	-	Y:$003541,20,4	M1028	Channel 10 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM100->X:$9009,0,24,s         ; ENC10 24-bit counter position
MM101->Y:$900A,8,16,s         ; OUT10A command value
MM102->Y:$900B,8,16,s         ; OUT10B command value
MM103->Y:$900C,8,16,s         ; OUT10C command value
```

Servo IC 4 Registers for Channel 3 (usually for Motor #11)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9011,0,24,S	-	-	M1101	ENC11 24-bit counter position
Y:$9012,8,16,S	-	-	M1102	OUT11A command value; DAC or PWM
X:$9013,0,24,S	MS20,MI921	-	M1103	ENC11 captured position
Y:$9013,8,16,S	-	-	M1104	OUT11B command value; DAC or PWM
Y:$9015,8,16,s	MS20,MI922	Y:$79429,8,16,s	M1105	ADC11A input value
Y:$9016,8,16,s	MS20,MI924	Y:$7942A,8,16,s	M1106	ADC11B input value
Y:$9014,8,16,s	-	-	M1107	OUT11C command value; PFM or PWM
Y:$9017,0,24,s	MS20,MI925	-	M1108	ENC11 compare A position
X:$9017,0,24,s	MS20,MI926	-	M1109	ENC11 compare B position
X:$9016,0,24,s	MS20,MI923	-	M1110	ENC11 compare auto-increment value
X:$9015,11	MS20,MI928	-	M1111	ENC11 compare initial state write enable
X:$9015,12	MS20,MI929	-	M1112	ENC11 compare initial state
X:$9015,14	-	Y:$003544,14	M1114	AENA11 output status
X:$9010,19	-	Y:$003544,19	M1115	USER11 flag input status
X:$9010,9	-	Y:$003544,9	M1116	ENC11 compare output value
X:$9010,11	-	Y:$003544,11	M1117	ENC11 capture flag
X:$9010,8	-	Y:$003544,8	M1118	ENC11 count error flag
X:$9010,14	-	-	M1119	CHC11 input status
X:$9010,16	-	Y:$003544,16	M1120	HMFL11 flag input status
X:$9010,17	-	Y:$003544,17	M1121	PLIM11 flag input status
X:$9010,18	-	Y:$003544,18	M1122	MLIM11 flag input status
X:$9010,15	-	Y:$003544,15	M1123	FAULT11 flag input status
X:$9010,20	-	Y:$003544,20	M1124	Channel 11 W flag input status
X:$9010,21	-	Y:$003544,21	M1125	Channel 11 V flag input status
X:$9010,22	-	Y:$003544,22	M1126	Channel 11 U flag input status
X:$9010,23	-	Y:$003544,23	M1127	Channel 11 T flag input status
X:$9010,20,4	-	Y:$003544,20,4	M1128	Channel 11 TUVW inputs as 4-bit value

Suggested MM-Variables

```
MM110->X:$9011,0,24,s         ; ENC11 24-bit counter position
MM111->Y:$9012,8,16,s         ; OUT11A command value
MM112->Y:$9013,8,16,s         ; OUT11B command value
MM113->Y:$9014,8,16,s         ; OUT11C command value
```

Servo IC 4 Registers for Channel 4 (usually for Motor #12)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9019,0,24,S	-	Y:$7942C,0,24,s	M1201	ENC12 24-bit counter position
Y:$901A,8,16,S	-	-	M1202	OUT12A command value; DAC or PWM
X:$901B,0,24,S	MS21,MI921	-	M1203	ENC12 captured position
Y:$901B,8,16,S	-	-	M1204	OUT12B command value; DAC or PWM
Y:$901D,8,16,s	MS21,MI922	Y:$7942D,8,16,s	M1205	ADC12A input value
Y:$901E,8,16,s	MS21,MI924	Y:$7942E,8,16,s	M1206	ADC12B input value
Y:$901C,8,16,s	-	-	M1207	OUT12C command value; PFM or PWM
Y:$901F,0,24,s	MS21,MI925	-	M1208	ENC12 compare A position
X:$901F,0,24,s	MS21,MI926	-	M1209	ENC12 compare B position
X:$901E,0,24,s	MS21,MI923	-	M1210	ENC12 compare auto-increment value
X:$901D,11	MS21,MI928	-	M1211	ENC12 compare initial state write enable
X:$901D,12	MS21,MI929	-	M1212	ENC12 compare initial state
X:$901D,14	-	Y:$003545,14	M1214	AENA12 output status
X:$9018,19	-	Y:$003545,19	M1215	USER12 flag input status
X:$9018,9	-	Y:$003545,9	M1216	ENC12 compare output value
X:$9018,11	-	Y:$003545,11	M1217	ENC12 capture flag
X:$9018,8	-	Y:$003545,8	M1218	ENC12 count error flag
X:$9018,14	-	-	M1219	CHC12 input status
X:$9018,16	-	Y:$003545,16	M1220	HMFL12 flag input status
X:$9018,17	-	Y:$003545,17	M1221	PLIM12 flag input status
X:$9018,18	-	Y:$003545,18	M1222	MLIM12 flag input status
X:$9018,15	-	Y:$003545,15	M1223	FAULT12 flag input status
X:$9018,20	-	Y:$003545,20	M1224	Channel 12 W flag input status
X:$9018,21	-	Y:$003545,21	M1225	Channel 12 V flag input status
X:$9018,22	-	Y:$003545,22	M1226	Channel 12 U flag input status
X:$9018,23	-	Y:$003545,23	M1227	Channel 12 T flag input status
X:$9018,20,4	-	Y:$003545,20,4	M1228	Channel 12 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM120->X:$9019,0,24,s         ; ENC12 24-bit counter position
MM121->Y:$901A,8,16,s         ; OUT12A command value
MM122->Y:$901B,8,16,s         ; OUT12B command value
MM123->Y:$901C,8,16,s         ; OUT12C command value
```

Servo IC 5 Registers for Channel 1 (usually for Motor #13)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9041,0,24,S	-	-	M1301	ENC13 24-bit counter position
Y:$9042,8,16,S	-	-	M1302	OUT13A command value; DAC or PWM
X:$9043,0,24,S	MS24,MI921	-	M1303	ENC13 captured position
Y:$9043,8,16,S	-	-	M1304	OUT13B command value; DAC or PWM
Y:$9045,8,16,s	MS24,MI922	Y:$79431,8,16,s	M1305	ADC13A input value
Y:$9046,8,16,s	MS24,MI924	Y:$79432,8,16,s	M1306	ADC13B input value
Y:$9044,8,16,s	-	-	M1307	OUT13C command value; PFM or PWM
Y:$9047,0,24,s	MS24,MI925	-	M1308	ENC13 compare A position
X:$9047,0,24,s	MS24,MI926	-	M1309	ENC13 compare B position
X:$9046,0,24,s	MS24,MI923	-	M1310	ENC13 compare auto-increment value
X:$9045,11	MS24,MI928	-	M1311	ENC13 compare initial state write enable
X:$9045,12	MS24,MI929	-	M1312	ENC13 compare initial state
X:$9045,14	-	Y:$003458,14	M1314	AENA13 output status
X:$9040,19	-	Y:$003458,19	M1315	USER13 flag input status
X:$9040,9	-	Y:$003458,9	M1316	ENC13 compare output value
X:$9040,11	-	Y:$003458,11	M1317	ENC13 capture flag
X:$9040,8	-	Y:$003458,8	M1318	ENC13 count error flag
X:$9040,14	-	-	M1319	CHC13 input status
X:$9040,16	-	Y:$003458,16	M1320	HMFL13 flag input status
X:$9040,17	-	Y:$003458,17	M1321	PLIM13 flag input status
X:$9040,18	-	Y:$003458,18	M1322	MLIM13 flag input status
X:$9040,15	-	Y:$003458,15	M1323	FAULT13 flag input status
X:$9040,20	-	Y:$003458,20	M1324	Channel 13 W flag input status
X:$9040,21	-	Y:$003458,21	M1325	Channel 13 V flag input status
X:$9040,22	-	Y:$003458,22	M1326	Channel 13 U flag input status
X:$9040,23	-	Y:$003458,23	M1327	Channel 13 T flag input status
X:$9040,20,4	-	Y:$003458,20,4	M1328	Channel 13 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM130->X:$9041,0,24,s         ; ENC13 24-bit counter position
MM131->Y:$9042,8,16,s         ; OUT13A command value
MM132->Y:$9043,8,16,s         ; OUT13B command value
MM133->Y:$9044,8,16,s         ; OUT13C command value
```

Servo IC 5 Registers for Channel 2 (usually for Motor #12)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9049,0,24,S	-	-	M1401	ENC14 24-bit counter position
Y:$904A,8,16,S	-	-	M1402	OUT14A command value; DAC or PWM
X:$904B,0,24,S	MS25,MI921	-	M1403	ENC14 captured position
Y:$904B,8,16,S	-	-	M1404	OUT14B command value; DAC or PWM
Y:$904D,8,16,s	MS25,MI922	Y:$79435,8,16,s	M1405	ADC14A input value
Y:$904E,8,16,s	MS25,MI924	Y:$79436,8,16,s	M1406	ADC14B input value
Y:$904C,8,16,s	-	-	M1407	OUT14C command value; PFM or PWM
Y:$904F,0,24,s	MS25,MI925	-	M1408	ENC14 compare A position
X:$904F,0,24,s	MS25,MI926	-	M1409	ENC14 compare B position
X:$904E,0,24,s	MS25,MI923	-	M1410	ENC14 compare auto-increment value
X:$904D,11	MS25,MI928	-	M1411	ENC14 compare initial state write enable
X:$904D,12	MS25,MI929	-	M1412	ENC14 compare initial state
X:$904D,14	-	Y:$003549,14	M1414	AENA14 output status
X:$9048,19	-	Y:$003549,19	M1415	USER14 flag input status
X:$9048,9	-	Y:$003549,9	M1416	ENC14 compare output value
X:$9048,11	-	Y:$003549,11	M1417	ENC14 capture flag
X:$9048,8	-	Y:$003549,8	M1418	ENC14 count error flag
X:$9048,14	-	-	M1419	CHC14 input status
X:$9048,16	-	Y:$003549,16	M1420	HMFL14 flag input status
X:$9048,17	-	Y:$003549,17	M1421	PLIM14 flag input status
X:$9048,18	-	Y:$003549,18	M1422	MLIM14 flag input status
X:$9048,15	-	Y:$003549,15	M1423	FAULT14 flag input status
X:$9048,20	-	Y:$003549,20	M1424	Channel 14 W flag input status
X:$9048,21	-	Y:$003549,21	M1425	Channel 14 V flag input status
X:$9048,22	-	Y:$003549,22	M1426	Channel 14 U flag input status
X:$9048,23	-	Y:$003549,23	M1427	Channel 14 T flag input status
X:$9048,20,4	-	Y:$003549,20,4	M1428	Channel 14 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM140->X:$9049,0,24,s         ; ENC14 24-bit counter position
MM141->Y:$904A,8,16,s         ; OUT14A command value
MM142->Y:$904B,8,16,s         ; OUT14B command value
MM143->Y:$904C,8,16,s         ; OUT14C command value
```

Servo IC 5 Registers for Channel 3 (usually for Motor #15)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9051,0,24,S	-	-	M1501	ENC15 24-bit counter position
Y:$9052,8,16,S	-	-	M1502	OUT15A command value; DAC or PWM
X:$9053,0,24,S	MS28,MI921	-	M1503	ENC15 captured position
Y:$9053,8,16,S	-	-	M1504	OUT15B command value; DAC or PWM
Y:$9055,8,16,s	MS28,MI922	Y:$79439,8,16,s	M1505	ADC15A input value
Y:$9056,8,16,s	MS28,MI924	Y:$7943A,8,16,s	M1506	ADC15B input value
Y:$9054,8,16,s	-	-	M1507	OUT15C command value; PFM or PWM
Y:$9057,0,24,s	MS28,MI925	-	M1508	ENC15 compare A position
X:$9057,0,24,s	MS28,MI926	-	M1509	ENC15 compare B position
X:$9056,0,24,s	MS28,MI923	-	M1510	ENC15 compare auto-increment value
X:$9055,11	MS28,MI928	-	M1511	ENC15 compare initial state write enable
X:$9055,12	MS28,MI929	-	M1512	ENC15 compare initial state
X:$9055,14	-	Y:$00354C,14	M1514	AENA15 output status
X:$9050,19	-	Y:$00354C,19	M1515	USER15 flag input status
X:$9050,9	-	Y:$00354C,9	M1516	ENC15 compare output value
X:$9050,11	-	Y:$00354C,11	M1517	ENC15 capture flag
X:$9050,8	-	Y:$00354C,8	M1518	ENC15 count error flag
X:$9050,14	-	-	M1519	CHC15 input status
X:$9050,16	-	Y:$00354C,16	M1520	HMFL15 flag input status
X:$9050,17	-	Y:$00354C,17	M1521	PLIM15 flag input status
X:$9050,18	-	Y:$00354C,18	M1522	MLIM15 flag input status
X:$9050,15	-	Y:$00354C,15	M1523	FAULT15 flag input status
X:$9050,20	-	Y:$00354C,20	M1524	Channel 15 W flag input status
X:$9050,21	-	Y:$00354C,21	M1525	Channel 15 V flag input status
X:$9050,22	-	Y:$00354C,22	M1526	Channel 15 U flag input status
X:$9050,23	-	Y:$00354C,23	M1527	Channel 15 T flag input status
X:$9050,20,4	-	Y:$00354C,20,4	M1528	Channel 15 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM150->X:$9051,0,24,s         ; ENC15 24-bit counter position
MM151->Y:$9052,8,16,s         ; OUT15A command value
MM152->Y:$9053,8,16,s         ; OUT15B command value
MM153->Y:$9054,8,16,s         ; OUT15C command value
```

Servo IC 5 Registers for Channel 4 (usually for Motor #16)

MACRO Location	MIVariable	Turbo Location	M-Var	Description
X:$9059,0,24,S	-	-	M1601	ENC16 24-bit counter position
Y:$905A,8,16,S	-	-	M1602	OUT16A command value; DAC or PWM
X:$905B,0,24,S	MS29,MI921	-	M1603	ENC16 captured position
Y:$905B,8,16,S	-	-	M1604	OUT16B command value; DAC or PWM
Y:$905D,8,16,s	MS29,MI922	Y:$7943D,8,16,s	M1605	ADC16A input value
Y:$905E,8,16,s	MS29,MI924	Y:$7943E,8,16,s	M1606	ADC16B input value
Y:$905C,8,16,s	-	-	M1607	OUT16C command value; PFM or PWM
Y:$905F,0,24,s	MS29,MI925	-	M1608	ENC16 compare A position
X:$905F,0,24,s	MS29,MI926	-	M1609	ENC16 compare B position
X:$905E,0,24,s	MS29,MI923	-	M1610	ENC16 compare auto-increment value
X:$905D,11	MS29,MI928	-	M1611	ENC16 compare initial state write enable
X:$905D,12	MS29,MI929	-	M1612	ENC16 compare initial state
X:$905D,14	-	Y:$00354D,14	M1614	AENA16 output status
X:$9058,19	-	Y:$00354D,19	M1615	USER16 flag input status
X:$9058,9	-	Y:$00354D,9	M1616	ENC16 compare output value
X:$9058,11	-	Y:$00354D,11	M1617	ENC16 capture flag
X:$9058,8	-	Y:$00354D,8	M1618	ENC16 count error flag
X:$9058,14	-	-	M1619	CHC16 input status
X:$9058,16	-	Y:$00354D,16	M1620	HMFL16 flag input status
X:$9058,17	-	Y:$00354D,17	M1621	PLIM16 flag input status
X:$9058,18	-	Y:$00354D,18	M1622	MLIM16 flag input status
X:$9058,15	-	Y:$00354D,15	M1623	FAULT16 flag input status
X:$9058,20	-	Y:$00354D,20	M1624	Channel 16 W flag input status
X:$9058,21	-	Y:$00354D,21	M1625	Channel 16 V flag input status
X:$9058,22	-	Y:$00354D,22	M1626	Channel 16 U flag input status
X:$9058,23	-	Y:$00354D,23	M1627	Channel 16 T flag input status
X:$9058,20,4	-	Y:$00354D,20,4	M1628	Channel 16 TUVW inputs as 4-bit value

Suggested MM-Variables:

```
MM160->X:$9059,0,24,s         ; ENC16 24-bit counter position
MM161->Y:$905A,8,16,s         ; OUT16A command value
MM162->Y:$905B,8,16,s         ; OUT16B command value
MM163->Y:$905C,8,16,s         ; OUT16C command value
```

NODE TRANSFER ADDRESSES CHART

16-Axis MACRO CPU Node Addresses

IC0 Node	Node 24-bit Transfer Addresses	Node 16-bit (upper 16 bits) Transfer Addresses	Node Type
0	Y:$C0A0	Y:$C0A1, Y:$C0A2, Y:$C0A3	Servo
1	Y:$C0A4	Y:$C0A5, Y:$C0A6, Y:$C0A7	Servo
2	X:$C0A0	X:$C0A1, X:$C0A2, X:$C0A3	I/O
3	X:$C0A4	X:$C0A5, X:$C0A6, X:$C0A7	I/O
4	Y:$C0A8	Y:$C0A9, Y:$C0AA, Y:$C0AB	Servo
5	Y:$C0AC	Y:$C0AD, Y:$C0AE, Y:$C0AF	Servo
6	X:$C0A8	X:$C0A9, X:$C0AA, X:$C0AB	I/O
7	X:$C0AC	X:$C0AD, X:$C0AE, X:$C0AF	I/O
8	Y:$C0B0	Y:$C0B1, Y:$C0B2, Y:$C0B3	Servo
9	Y:$C0B4	Y:$C0B5, Y:$C0B6, Y:$C0B7	Servo
10	X:$C0B0	X:$C0B1, X:$C0B2, X:$C0B3	I/O
11	X:$C0B4	X:$C0B5, X:$C0B6, X:$C0B7	I/O
12	Y:$C0B8	Y:$C0B9, Y:$C0BA, Y:$C0BB	Servo
13	Y:$C0BC	Y:$C0BD, Y:$C0BE, Y:$C0BF	Servo
14	X:$C0B8	X:$C0B9, X:$C0BA, X:$C0BB	Master-Master Communication
15	X:$C0BC	X:$C0BD, X:$C0BE, X:$C0BF	Master-Slave Communication

IC1 Node	Node 24-bit Transfer Addresses	Node 16-bit (upper 16 bits) Transfer Addresses	Node Type
16	Y:$C0C0	Y:$C0C1, Y:$C0C2, Y:$C0C3	Servo
17	Y:$C0C4	Y:$C0C5, Y:$C0C6, Y:$C0C7	Servo
18	X:$C0C0	X:$C0C1, X:$C0C2, X:$C0C3	I/O
19	X:$C0C4	X:$C0C5, X:$C0C6, X:$C0C7	I/O
20	Y:$C0C8	Y:$C0C9, Y:$C0CA, Y:$C0CB	Servo
21	Y:$C0CC	Y:$C0CD, Y:$C0CE, Y:$C0CF	Servo
22	X:$C0C8	X:$C0C9, X:$C0CA, X:$C0CB	I/O
23	X:$C0CC	X:$C0CD, X:$C0CE, X:$C0CF	I/O
24	Y:$C0D0	Y:$C0D1, Y:$C0D2, Y:$C0D3	Servo
25	Y:$C0D4	Y:$C0D5, Y:$C0D6, Y:$C0D7	Servo
26	X:$C0D0	X:$C0D1, X:$C0D2, X:$C0D3	I/O
27	X:$C0D4	X:$C0D5, X:$C0D6, X:$C0D7	I/O
28	Y:$C0D8	Y:$C0D9, Y:$C0DA, Y:$C0DB	Servo
29	Y:$C0DC	Y:$C0DD, Y:$C0DE, Y:$C0DF	Servo
30	X:$C0D8	X:$C0D9, X:$C0DA, X:$C0DB	Master-Master Communication
31	X:$C0DC	X:$C0DD, X:$C0DE, X:$C0DF	Master-Slave Communication

Turbo PMAC2 Node Addresses

MACRO IC Node	Axis/IO	User Node	Node 24-bit Transfer Addresses	Node 16-bit (upper 16 bits) Transfer Addresses
(IC0) 0	Axis 1	0	Y:$078420	Y:$078421, Y:$078422, Y:$078423
(IC0) 1	Axis 2	1	Y:$078424	Y:$078425, Y:$078426, Y:$078427
(IC0) 2	I/O	2	X:$078420	X:$078421, X:$078422, X:$078423
(IC0) 3	I/O	3	X:$078424	X:$078425, X:$078426, X:$078427
(IC0) 4	Axis 3	4	Y:$078428	Y:$078429, Y:$07842A, Y:$07842B
(IC0) 5	Axis 4	5	Y:$07842C	Y:$07842D, Y:$07842E, Y:$07842F
(IC0) 6	I/O	6	X:$078428	X:$078429, X:$07842A, X:$07842B
(IC0) 7	I/O	7	X:$07842C	X:$07842D, X:$07842E, X:$07842F
(IC0) 8	Axis 5	8	Y:$078430	Y:$078431, Y:$078432, Y:$078433
(IC0) 9	Axis 6	9	Y:$078434	Y:$078435, Y:$078436, Y:$078437
(IC0) 10	I/O	10	X:$078430	X:$078431, X:$078432, X:$078433
(IC0) 11	I/O	11	X:$078434	X:$078435, X:$078436, X:$078437
(IC0) 12	Axis7	12	Y:$078438	Y:$078439, Y:$07843A, Y:$07843B
(IC0) 13	Axis 8	13	Y:$07843C	Y:$07843D, Y:$07843E, Y:$07843F
(IC0) 14	Master/Master	14	X:$078438	X:$078439, X:$07843A, X:$07843B
(IC0) 15	Master/Slave	15	X:$07843C	X:$07843D, X:$07843E, X:$07843F
(IC1) 0	Axis 9	16	Y:$079420	Y:$079421, Y:$079422, Y:$079423
(IC1) 1	Axis 10	17	Y:$079424	Y:$079425, Y:$079426, Y:$079427
(IC1) 2	I/O	18	X:$079420	X:$079421, X:$079422, X:$079423
(IC1) 3	I/O	19	X:$079424	X:$079425, X:$079426, X:$079427
(IC1) 4	Axis 11	20	Y:$079428	Y:$079429, Y:$07942A, Y:$07942B
(IC1) 5	Axis 12	21	Y:$07942C	Y:$07942D, Y:$07942E, Y:$07942F
(IC1) 6	I/O	22	X:$079428	X:$079429, X:$07942A, X:$07942B
(IC1) 7	I/O	23	X:$07942C	X:$07942D, X:$07942E, X:$07942F
(IC1) 8	Axis 13	24	Y:$079430	Y:$079431, Y:$079432, Y:$079433
(IC1) 9	Axis 14	25	Y:$079434	Y:$079435, Y:$079436, Y:$079437
(IC1) 10	I/O	26	X:$079430	X:$079431, X:$079432, X:$079433
(IC1) 11	I/O	27	X:$079434	X:$079435, X:$079436, X:$079437
(IC1) 12	Axis 15	28	Y:$079438	Y:$079439, Y:$07943A, Y:$07943B
(IC1) 13	Axis 16	29	Y:$07943C	Y:$07943D, Y:$07943E, Y:$07943F
(IC1) 14	Master/Master	30	X:$079438	X:$079439, X:$07943A, X:$07943B
(IC1) 15	Master/Slave	31	X:$07943C	X:$07943D, X:$07943E, X:$07943F
(IC2) 0	Axis 17	32	Y:$07A420	Y:$07A421, Y:$07A422, Y:$07A423
(IC2) 1	Axis 18	33	Y:$07A424	Y:$07A425, Y:$07A426, Y:$07A427
(IC2) 2	I/O	34	X:$07A420	X:$07A421, X:$07A422, X:$07A423
(IC2) 3	I/O	35	X:$07A424	X:$07A425, X:$07A426, X:$07A427
(IC2) 4	Axis 19	36	Y:$07A428	Y:$07A429, Y:$07A42A, Y:$07A42B
(IC2) 5	Axis 20	37	Y:$07A42C	Y:$07A42D, Y:$07A42E, Y:$07A42F
(IC2) 6	I/O	38	X:$07A428	X:$07A429, X:$07A42A, X:$07A42B
(IC2) 7	I/O	39	X:$07A42C	X:$07A42D, X:$07A42E, X:$07A42F
(IC2) 8	Axis 21	40	Y:$07A430	Y:$07A431, Y:$07A432, Y:$07A433
(IC2) 9	Axis 22	41	Y:$07A434	Y:$07A435, Y:$07A436, Y:$07A437
(IC2) 10	I/O	42	X:$07A430	X:$07A431, X:$07A432, X:$07A433
(IC2) 11	I/O	43	X:$07A434	X:$07A435, X:$07A436, X:$07A437
(IC2) 12	Axis 23	44	Y:$07A438	Y:$07A439, Y:$07A43A, Y:$07A43B
(IC2) 13	Axis 24	45	Y:$07A43C	Y:$07A43D, Y:$07A43E, Y:$07A43F
(IC2) 14	Master/Master	46	X:$07A438	X:$07A439, X:$07A43A, X:$07A43B
(IC2) 15	Master/Slave	47	X:$07A43C	X:$07A43D, X:$07A43E, X:$07A43F
(IC3) 0	Axis 25	48	Y:$07B420	Y:$07B421, Y:$07B422, Y:$07B423

(IC3) 1	Axis 26	49	Y:$07B424	Y:$07B425, Y:$07B426, Y:$07B427
(IC3) 2	I/O	50	X:$07B420	X:$07B421, X:$07B422, X:$07B423
(IC3) 3	I/O	51	X:$07B424	X:$07B425, X:$07B426, X:$07B427
(IC3) 4	Axis 27	52	Y:$07B428	Y:$07B429, Y:$07B42A, Y:$07B42B
(IC3) 5	Axis 28	53	Y:$07B42C	Y:$07B42D, Y:$07B42E, Y:$07B42F
(IC3) 6	I/O	54	X:$07B428	X:$07B429, X:$07B42A, X:$07B42B
(IC3) 7	I/O	55	X:$07B42C	X:$07B42D, X:$07B42E, X:$07B42F
(IC3) 8	Axis 29	56	Y:$07B430	Y:$07B431, Y:$07B432, Y:$07B433
(IC3) 9	Axis 30	57	Y:$07B434	Y:$07B435, Y:$07B436, Y:$07B437
(IC3) 10	I/O	58	X:$07B430	X:$07B431, X:$07B432, X:$07B433
(IC3) 11	I/O	59	X:$07B434	X:$07B435, X:$07B436, X:$07B437
(IC3) 12	Axis 31	60	Y:$07B438	Y:$07B439, Y:$07B43A, Y:$07B43B
(IC3) 13	Axis 32	61	Y:$07B43C	Y:$07B43D, Y:$07B43E, Y:$07B43F
(IC3) 14	Master/Master	62	X:$07B438	X:$07B439, X:$07B43A, X:$07B43B
(IC3) 15	Master/Slave	63	X:$07B43C	X:$07B43D, X:$07B43E, X:$07B43F