ADDvantage-32
ADVANCED CONTROL MODULE (ACM)

Cleveland, Ohio

November 12,1993
Rev. Aug. 2, 2010

PROPRIETARY NOTE

This document contains information PROPRIETARY to Avtron Manufacturing, Inc., is furnished solely to provide information sufficient for instruction, operation, maintenance, evaluation, and testing of the equipment herein disclosed, is not to be used for manufacturing or procurement and is not to be disclosed to anyone other than persons in the Division, or the Company, or the Government, as the case may be, responsible for action relating to this document without the express written permission of Avtron Manufacturing, Inc.

WARRANTY

The last page of this document contains an express limited warranty. The provisions of this warranty cover any and all rights extended to holders of Avtron equipment.

Feb. 1995

ADDvantage-32
ADVANCED CONTROL MODULE (ACM)

TABLE OF CONTENTS

SAFETY SUMMARY

****

W A R N I N G

HAZARDOUS VOLTAGES ARE USED IN THE OPERATION OF THIS EQUIPMENT AND MAY CAUSE SEVERE PERSONAL INJURY OR THE LOSS OF LIFE IF PROPER PRECAUTIONS ARE NOT TAKEN. THE FOLLOWING PRECAUTIONS SHOULD BE TAKEN TO REDUCE THE RISK OF INJURY OR DEATH.

****

1. Only qualified personnel familiar with this equipment should be permitted to install, operate, troubleshoot, or repair the apparatus after reading and understanding this manual.

2. Installation of the equipment must be performed in accordance with the National Electrical Code and any other state or local codes. Proper grounding, conductor sizing, and short circuit protection must be installed for safe operation.

3. During normal operation, keep all covers in place and cabinet doors shut.

4. When performing hands-on inspections and maintenance, be sure the incoming AC feed is turned off and locked out. The ADD-32 and motor may have hazardous voltages present even if the AC feed is turned off. ****NOTE** THE ARMATURE CONTACTOR DOES NOT REMOVE HAZARDOUS VOLTAGES WHEN OPENED.**

5. When necessary to take measurements with the power turned on, do not touch any electrical connection points. Remove all jewelry from wrists and fingers. Make sure test equipment is in safe operating condition.

6. While servicing with the power on, stand on approved insulating material and be sure not to be grounded.

7. Follow the instructions in this manual carefully and observe all danger notices.

****

W A R N I N G

ACCURACY OF CUSTOMER-INSTALLED CALIBRATION AND CONFIGURATION DATA IS IMPERATIVE IN THE OPERATION OF THIS EQUIPMENT. INCORRECT DATA MAY CAUSE DAMAGE TO THE ADD-32, MOTOR, AND PROCESS EQUIPMENT.

****

W A R N I N G

DO NOT OPERATE RADIO TRANSMITTERS or CELL PHONES IN THE VICINITY OF THE ADD-32. The ADD-32 is an electronic device. Although it is designed to operate reliably in typical industrial environments, the ADD-32 can be affected by radio and/or cell phone transmitters. It is possible to cause drive faults, inappropriate/unintended drive I/O activity, and unpredictable operation that could result in damage to the ADD-32, damage to other equipment, or serious injury to personnel.

Radio transmitter interference is a site specific phenomena. Generally, electrical wires connected to terminals on the ADD-32 are the conduits for radio interference. Interference can be minimized by good wiring design and installation practice. It is recommended that signs be posted in and around the drive system, warning of the possibility of interference if the drive is in operation. DO NOT USE radio transmitters or cell phones in the area.

Absence of a radio interference problem is no guarantee that a problem will never occur as conditions and environments can change.

****

****

W A R N I N G

System Safety Considerations
ADD-32 DOK Fault Contact

The ADD-32 is a sophisticated microprocessor device incorporating many self diagnostic tests. The function of its DOK (Drive OK) fault contact is to detect a variety of faults in the ADD-32, but it cannot assure fault-free operation.

BE AWARE THAT NOT ALL ADD-32 FAULTS CAN BE SELF DIAGNOSED AND/OR ALARMED. THEREFORE, THE ADD-32 CANNOT BE CONSIDERED TO BE FAIL SAFE NOR CAN ITS "DOK" CONTACT BE RELIED ON TO RESPOND TO ALL POSSIBLE ADD-32 FAULT CONDITIONS.

The "DOK" contact should be incorporated into system interlock logic chains to ensure safe system response to a drive fault "DOK" output contact opening. It is the responsibility of the system designer to understand the system interlock logic chains and to apply the "DOK" contact appropriately.

IN SAFETY SENSITIVE APPLICATIONS, IT IS STRONGLY SUGGESTED THAT THE SYSTEM DESIGNER UTILIZE A SEPARATE MONITORING DEVICE TO CHECK THE ADD-32 INPUTS AND OUTPUTS, AND OTHER OPERATING CHARACTERISTICS, TO ENHANCE THE SAFETY OF PERSONNEL AND PROPERTY.

****

ADDvantage-32
ADVANCED CONTROL MODULE (ACM)

SECTION I

INTRODUCTION AND GENERAL INFORMATION

1.1 DESCRIPTION

The ADDvantage-32 Advanced Control Module (ACM) provides a digital regulator for retrofitting existing converters and inverters. A complete digital system can be created using ACM's with the existing high power circuitry. This provides a cost effective solution for digital control on existing equipment. The ACM is fully compatible with the entire ADDvantage-32 product line and utilizes the same microprocessor and system boards.

STANDARD FEATURES INCLUDE:

- Advanced 32-bit, 20 MHz RISC microprocessor
- Fully digital
- Maintenance keypad with a 2-line alphanumeric digital display of plain English messages and engineering units
- 12 status LED indicators (eight user definable)
- Integral 4-Channel, high speed memory signal analyzer
- Fully digital, standard control loops which may include:
  - field CEMF regulation
  - speed control regulation
  - tension control regulation
  - center driven winder regulation
- Tach loss and overspeed protection, configurable with automatic switchover to a redundant digital tach or armature voltage feedback.
- DC motor temperature sensing with overtemperature protection
- Chassis type construction

- Field economy
- Low AC line voltage and frequency protection
- 16 event Fault FIFO
- All information accessible by remote operator interface over an RS485 multidrop link or 802.4 LAN
- Retentive setpoints (not available in 690XXX software)

1.2 HARDWARE AND EQUIPMENT FEATURES

Figure 1-1 illustrates the typical hardware configuration of the ADDvantage-32. Each hardware configuration contains a microprocessor board, system board, and power supply board.

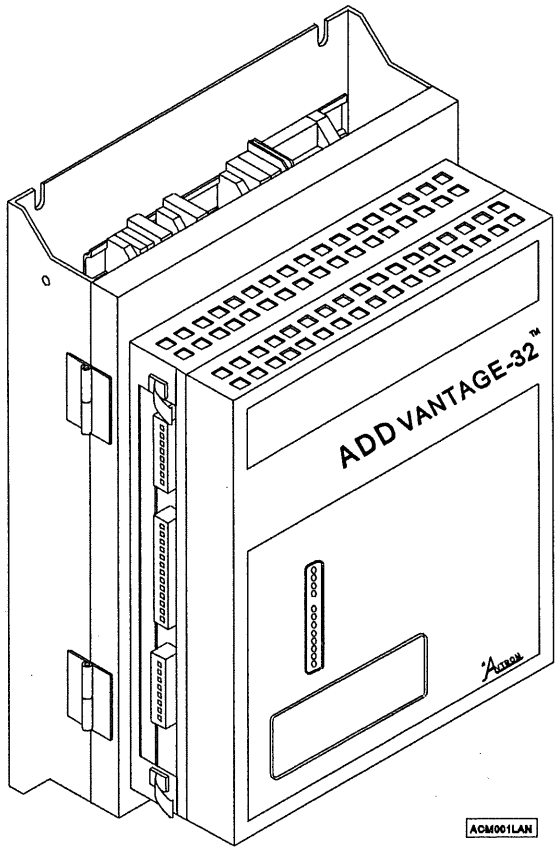


FIGURE 1-1. ADDvantage-32 ADVANCED CONTROL MODULE

Several components are used to make up a single ADDvantage-32 unit. The microprocessor and system boards are compatible with those of the entire ADDvantage-32 product line.

1.2.1 MICROPROCESSOR BOARD

The MICROPROCESSOR BOARD controls all functions including diagnostics, communications, and control block functions. The microprocessor board includes the following:

- The Intel i960, 32 bit superscaler microprocessor performs all ADDvantage-32 functions. The microprocessor computes multiple instructions per clock cycle, enabling the advanced features of the ACM to function.

- The Flash PROM stores the ADDvantage-32 application software program. Software upgrades can be accomplished over the serial link, eliminating the possibility of incorrectly installing or destroying integrated circuits.

- The EEPROM stores all of the calibration and configuration information, allowing all information to be retained during power loss and stored without battery-backup.

- The status LED's are mounted on the back of the microprocessor board. Except for the +5 POWER and PROCESSOR FAIL LED's, the LED's are software driven.

- The four-button keypad allows viewing or modification of all calibration, configuration, and running information.

- The 32-character alphanumeric display allows viewing of all ADDvantage-32 information. The 32 characters indicate English messages to the operator and represent the parameters in process units.

There are three microprocessor board options. Refer to Identification of Part Numbers for identification of a specific board.

ORIGINAL MICROPROCESSOR BOARD

This is the original microprocessor board installed on early versions of the ADDvantage-32. It has been replaced by updated boards; however, original microprocessor boards are still available for replacement parts. Consult the factory for availability and delivery.

LOW MEMORY MICROPROCESSOR BOARD

Low memory boards are used on ADDvantage-32's which do not require operation of the 802.4 LAN option. If the LAN is required, a high memory board must be used. The low memory microprocessor boards are used with non-LAN software numbers 69XX0X or 69XX1X.

HIGH MEMORY MICROPROCESSOR BOARD

High memory boards are used on ADDvantage-32's which use the 802.4 LAN option. This board can also be used on applications which do not use the LAN option. Software which is designed for LAN applications is number 69XX5X. The high memory microprocessor board may also be used with non-LAN software numbers 69XX0X or 69XX1X.

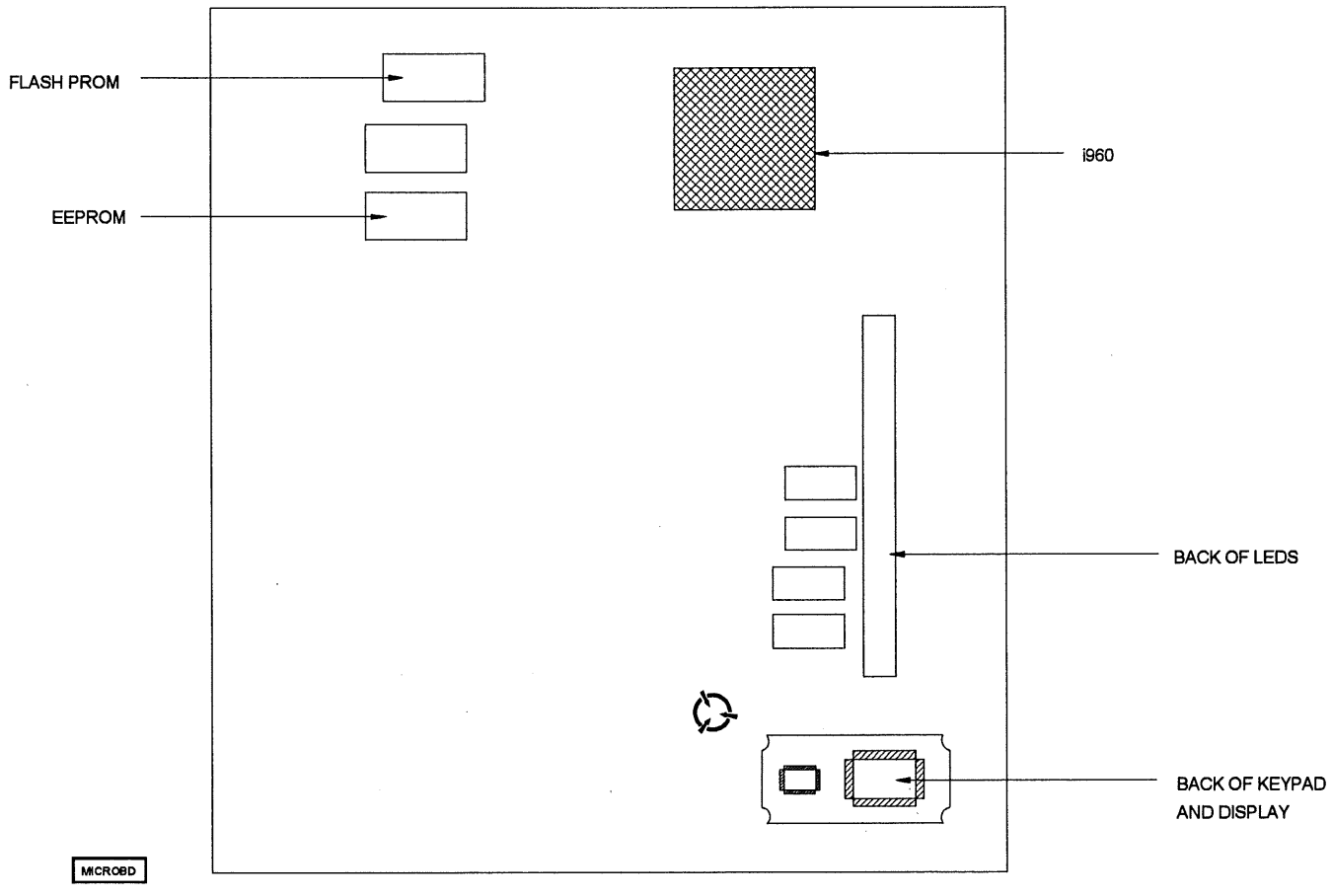


FIGURE 1-2. MICROPROCESSOR BOARD (A1)

1.2.2 SYSTEM PC BOARD

The SYSTEM BOARD isolates customer interconnections from the ADDvantage-32 circuitry and contains quick disconnect terminal blocks for user wiring. Several types of system boards are available as follows:

MINI SYSTEM BOARD - Contains the smallest amount of I/O available. Features include:

(3) Analog Inputs
(1) Analog Output
(4) Digital Inputs
(1) RS485 Serial Link

MAXI SYSTEM BOARD - Provides more I/O than the MINI version of this board. Features include:

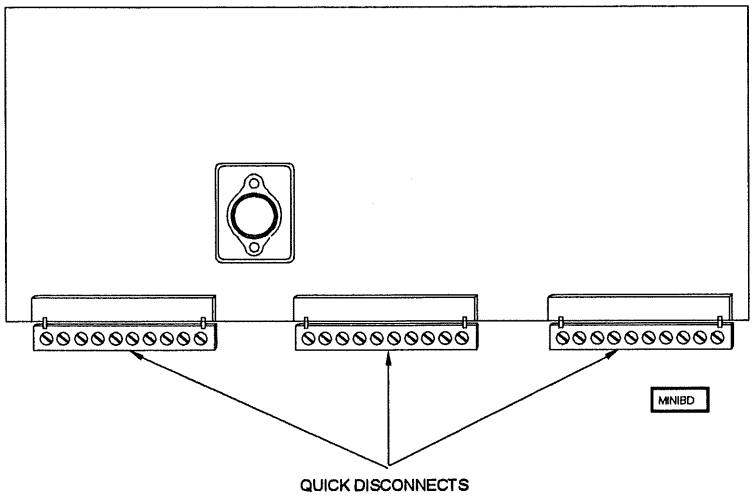
(6) Analog Inputs
(4) Analog Outputs
(6) Digital Inputs
(4) Form C Digital Outputs
(2) Two-Phase Tachometer Inputs
(1) Buffered Tachometer Output
(1) RS485 Serial Link

MAXI SYSTEM BOARD WITH 802.4 LAN - Contains the same I/O features as the standard MAXI board but also contains the LAN chip set to activate the 802.4 LAN.

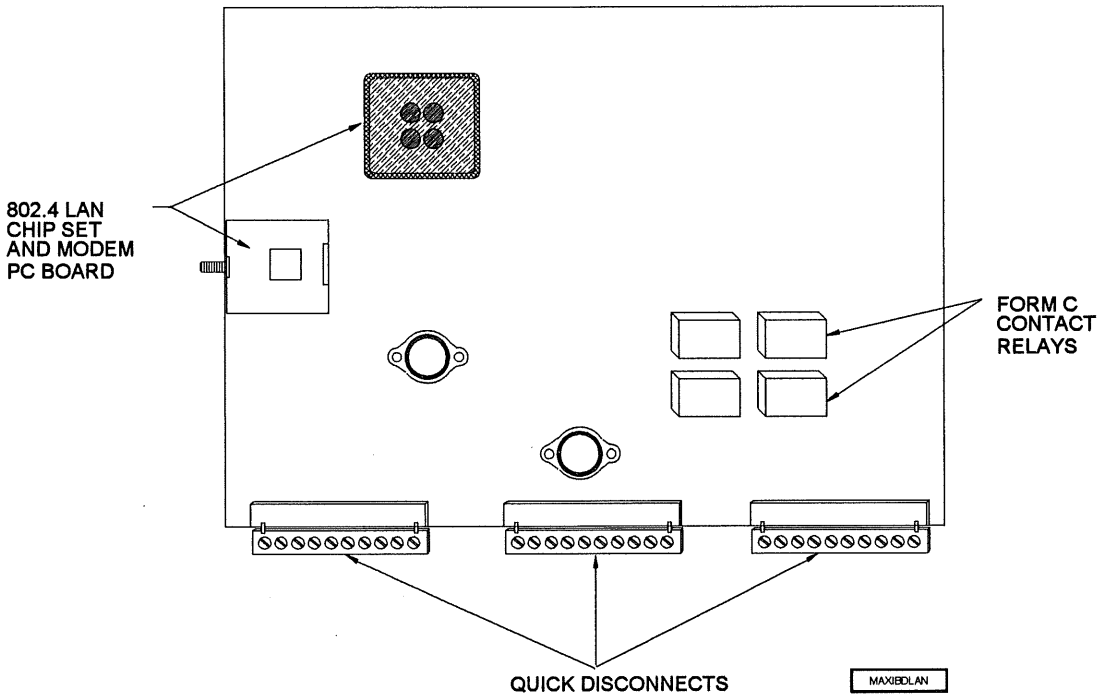
MAXI SYSTEM BOARD WITH FAX-32 BOARD - Same I/O features of the standard MAXI board with the addition of the FAX-32 piggyback board. The FAX-32 board provides the following additional I/O features:

(8) Digital Inputs
(2) Frequency Outputs capable of providing 200 to 20,200 Hz

MAXI SYSTEM BOARD w/FAX-32 AND 802.4 LAN - Contains the same I/O features as the standard MAXI board and FAX-32 board with the addition of the 802.4 LAN chip set to enable LAN operation.



MINI SYSTEM BOARD



MAXI SYSTEM BOARD

FIGURE 1-3. SYSTEM PC BOARD (A2)

1.2.3 FAX-32 BOARD

The FAX-32 BOARD is mounted piggyback to the Maxi System PC Board. Eight additional digital inputs are added to the maxi system board when the FAX-32 board is used. The FAX-32 board also provides a frequency output of 200 - 20,200 Hz. This frequency output can be used to cascade references.

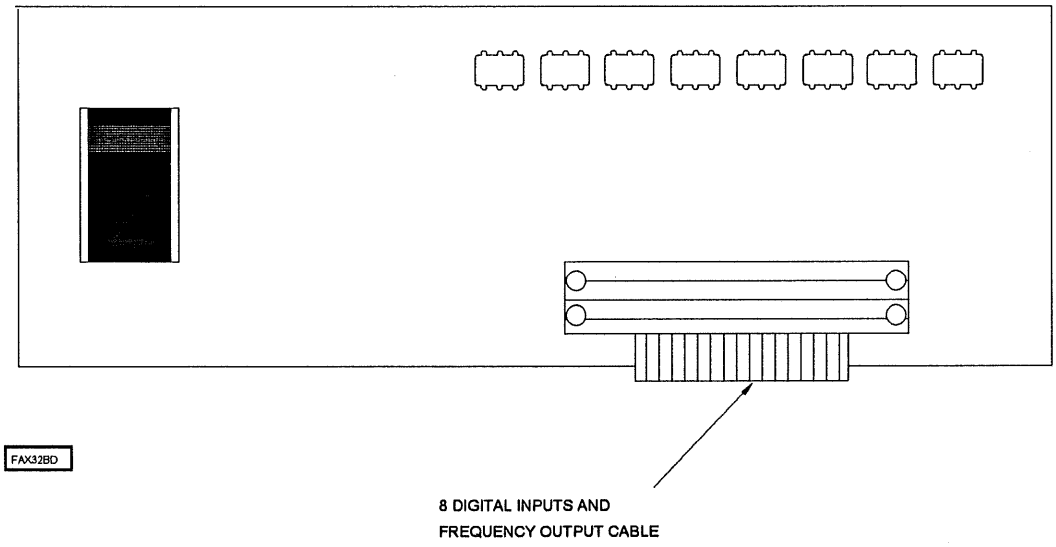


FIGURE 1-4. FAX-32 BOARD

1.2.4 POWER SUPPLY BOARD

The POWER SUPPLY BOARD provides DC power to the system and microprocessor boards. The main functional circuits on this board are as follows:

- Power Supplies for all PC Boards
- ESTOP and Drive OK Relays

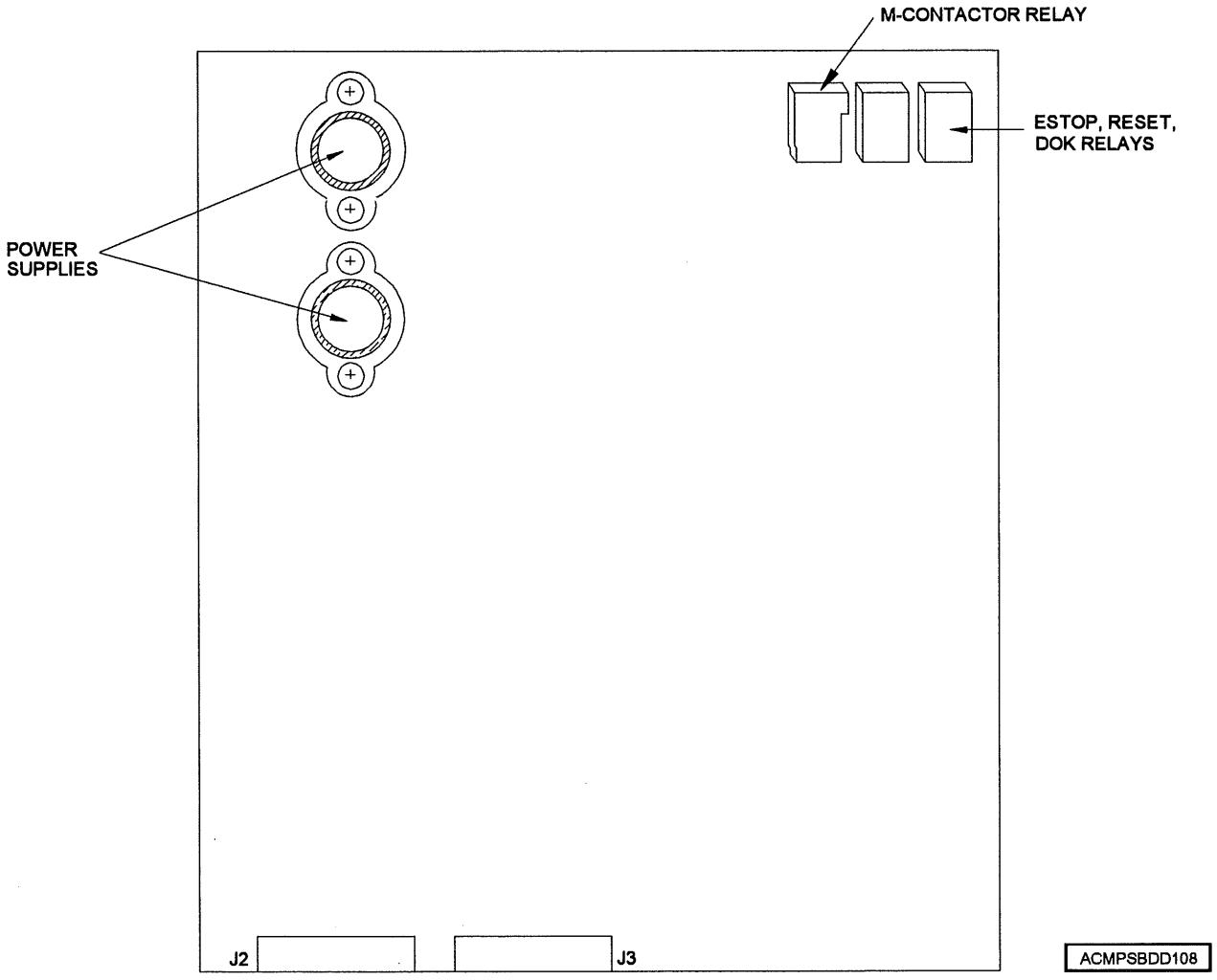


FIGURE 1-5. POWER SUPPLY BOARD (A3)

1.3 SOFTWARE FEATURES

Programming of the ADDvantage-32 is accomplished by configuring and calibrating information contained in the application software. Application software consists of control blocks, core software, and communications software. Each software type performs a specific task.

Application of the ADDvantage-32 programming is made easier by using a block diagram format. The block arrangement is specific to the type of application software being used. Blocks are factory calibrated but are configurable. For example, a speed/tension block diagram has a different block arrangement than a center driven winder. Both will use the same types of blocks, although the quantity and position of the blocks will vary. Appendix A contains the software specific block diagrams.

The core and communication software control such things as ACM diagnostics, I/O scanning, RS485 serial link, and 802.4 LAN communications.

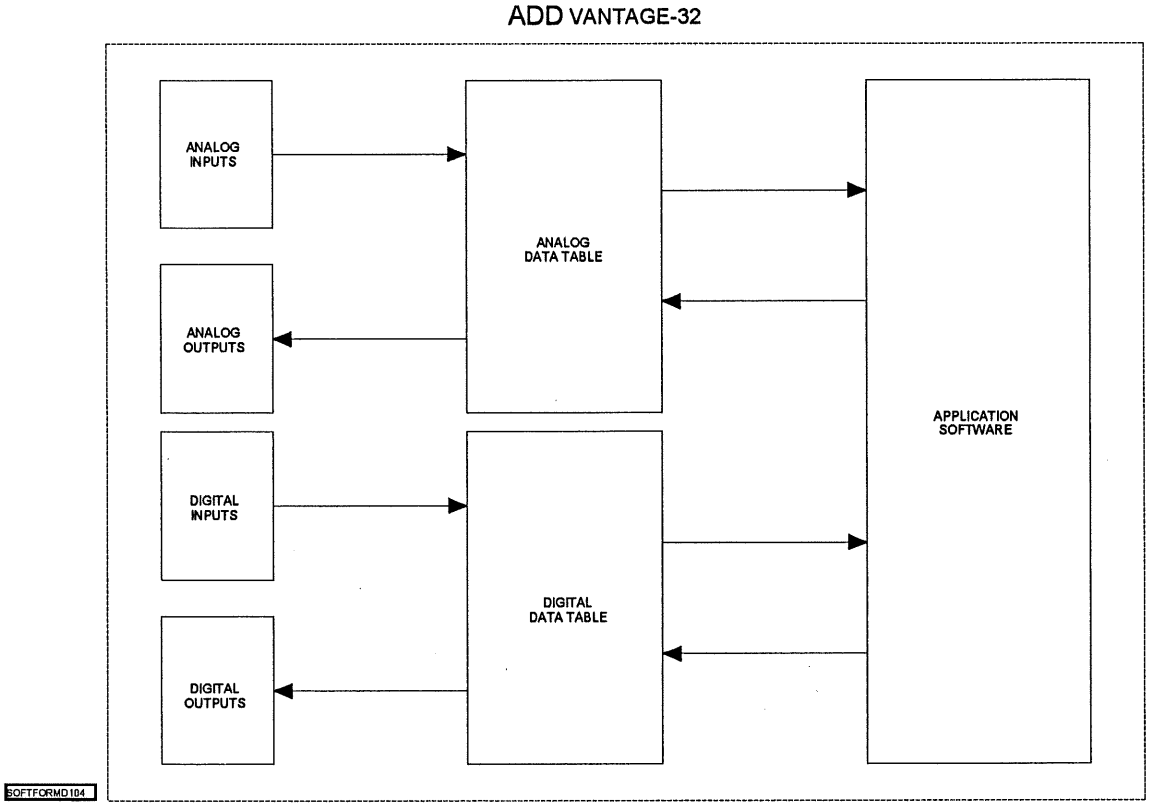


FIGURE 1-6. SOFTWARE FORMAT

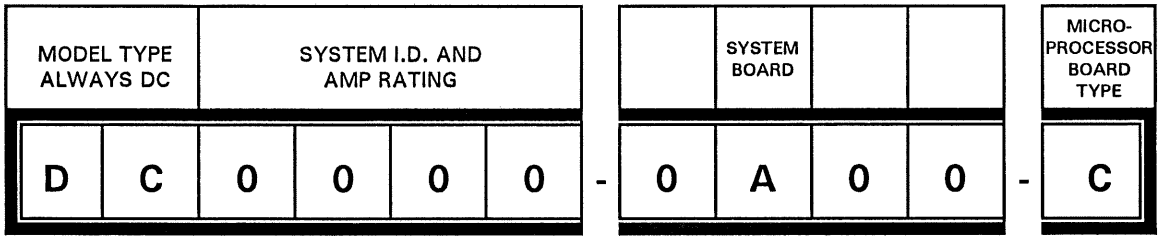
Information used by the ADDvantage-32 is in either analog or digital format. This information is separated into two data tables. These tables provide the information source for the entire ADDvantage-32. The user simply configures a particular point in the block software to a position in one of the tables, "patching" the information to the block.

Descriptions of the parameter types used by the ADDvantage-32 and how to calibrate them can be found in Section II of this manual.

1.4 IDENTIFICATION OF PART NUMBERS

The part numbering of the ADDvantage-32 consists of two separate numbers. The hardware part number is located on nameplates found on both the inside and outside of the unit and reflects the hardware configuration of the ADDvantage-32. The software part number, as well as the version number, identify the application software used with the ADDvantage-32 hardware.

HARDWARE PART NUMBER (EXAMPLE)



SOFTWARE PART NUMBER (EXAMPLE)

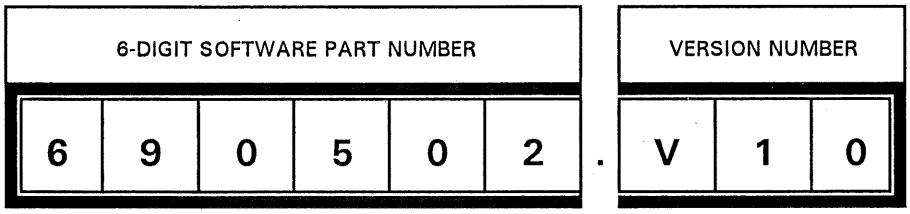


FIGURE 1-7. PART NUMBER BREAKDOWN

1.4.1 HARDWARE PART NUMBER BREAKDOWN

MODEL TYPE

SELECTION	DESCRIPTION
DC	Always DC

The DC designation in the part number of the ACM identifies that it is part of the ADDvantage-32 family of products.

SYSTEM BOARDS

SELECTION	DESCRIPTION
A	MINI System Board
B	MAXI System Board
C	MAXI With 802.4 LAN
D	MAXI With FAX-32
E	MAXI With 802.4 LAN and FAX-32

Refer to Section 1.3 of this manual for additional system board descriptions.

MICROPROCESSOR BOARDS

SELECTION	DESCRIPTION
A	Original ADDvantage-32 Microprocessor Board
B	Low Memory Microprocessor Board
C	High Memory Microprocessor Board

Refer to Section 1.3 of this manual for additional microprocessor board descriptions.

1.4.2 SOFTWARE PART NUMBER BREAKDOWN

The software part number is a six-digit number which represents the application software installed in the ADDvantage-32. Several different types are available including a standard speed regulator with tension (P/N 690502) and a center driven winder (P/N 690503). Most applications can be covered by the two application programs described. Other application software is available. To obtain a list or to discuss custom software applications, consult the factory.

NOTE

A two-digit version number is also required. If a specific version is not referenced, the latest version will be supplied.

1.5 SPECIFICATIONS

TABLE 1-1. ADDvantage-32 SPECIFICATIONS

DESCRIPTION	SPECIFICATION*
Control Power Input Rating	Single Phase, 115 VAC
Line Voltage Variation	+/-10% of nominal
Line Frequency Range	47 to 63 Hz
Mini System Board	(3) analog inputs, (1) analog output, (4) digital inputs, (1) RS485 Serial Link
Maxi System Board	(6) analog inputs, (4) analog outputs, (6) digital inputs, (4) digital outputs, (2) two-phase tach inputs, (1) buffered tach output, (1) RS485 Serial Link or (1) 802.4 LAN Interface
FAX-32 Board	(8) digital inputs, (2) identical frequency outputs, 200 - 20,200 Hz frequency range, Duty cycle of 50% ± 1%, 50 mA Imax, V_{out} of 5V differential, ±.017 commanded accuracy, ±.017 commanded resolution

Digital Input Ratings

Input Voltage (DC)	MIN	NOM	MAX
Common Mode (Logic High)	10 VDC	24 VDC	30 VDC
(Logic Low)	-1 VDC	0 VDC	9 VDC
Common Mode Rejection			100 VDC
Input Impedance		2.4Kohm	
Operating Current Required	4 mA		60 mA

* Additional Specifications are located in the supplemental drawing package.

TABLE 1-1. ADDvantage-32 SPECIFICATIONS--Cont.

DESCRIPTION	SPECIFICATION*
Digital Output Ratings	Output Voltage - 240 VAC Max. Output Current - 2.5 Amps Max. Form C relay contacts
Analog Input Ratings	Differential Input Voltage - ±10 VDC Max. Common Mode Input Voltage - 200 VDC Max. Common Mode Noise Rejection - 60 VDC Max. Accuracy - 0.25% F.S. Max. Drift - 0.12% F.S. Max.
Analog Output Ratings	Output Voltage - ±10 VDC Nom., ±24 VDC Max. Output Current - 10 mADC Nom., 40 mADC Max. Common Mode Noise Rejection - 2 VDC Max. Drift - ±1.2% F.S. Max. Resolution - 0.025% F.S. Max. Accuracy - ±1.8% F.S. Max.
Diagnostics	40 fault messages w/corrective action 12 front panel LED annunciators 16 event fault memory Signal Analyzer Sampling Rate - 90 Samples/Second Max., 1 Sample/110.8 Seconds Min. Recording Duration - 30.8 hours Max., 11 seconds Min.
RS485 Serial Link	Opto-isolated (Requires external 24V, 120 mA power supply), 120V common mode noise rejection

* Additional Specifications are located in the supplemental drawing package.

TABLE 1-1. ADDvantage-32 SPECIFICATIONS--Cont.

DESCRIPTION	SPECIFICATION*
Pulse Generator Inputs: 2-Phase	1-30V peak-to-peak, 4700 ohm impedance, Transformer isolated, 120V common mode rejection, 0-20,200 Hz frequency range, 50% ± 20% duty cycle■
Buffered Pulse Generator Outputs: 2-Phase	5V @ 50mA, 0-25 KHz frequency range
Auxiliary Power Supplies	Digital I/O Supply Voltage Output = 24 VDC Nom., 26 VDC Max. Current Output = 80 mADC Max. Auxiliary AC Supply Voltage Output = 115 VAC Nom. Power Output = 50 VA Max. (units 56A and under) 100 VA Max. (units 110A and above) Serial Link Supply Voltage Output = 24 VDC Nom., 26 VDC Max. Current Output = 100 mADC Max. Safety Interlock Supply Voltage Output = 24 VDC Nom., 26 VDC Max. Current Output = 50 mADC Max. Pulse Generator Supply Voltage Output = 12 VDC Nom., 13.2 VDC Max. Current Output = 300 mA Max. (Short circuit proof) Pot Supplies Voltage Output = ±10 VDC Max. Current Output = 5 mA Max.

* Additional Specifications are located in the supplemental drawing package.

■ Consult Avtron regarding UNIPULSER™ information for additional duty cycle applications.

TABLE 1-1. ADDvantage-32 SPECIFICATIONS--Cont.

DESCRIPTION	SPECIFICATION*
Chassis Ambient Temperature	
- Enclosed (See outline drawing for minimum enclosure size.)	0 to 40° C (32 to 104° F)
- Open Chassis	0 to 50° C (32 to 122° F)(50° C maximum surrounding air temperature rating)
Storage Temperature***	-20 to 55° C (-4 to 131° F)
Relative Humidity	95% non-condensing
Operational Altitude	0 to 3,300 feet above sea level - no derating required Above 3,300 feet - derated linearly by 1% per 300 ft.

* Additional Specifications are located in the supplemental drawing package.

*** A space heater may be necessary if condensation or excessive moisture is expected.

SECTION II

KEYBOARD AND PARAMETER FUNCTIONS

****

W A R N I N G

DO NOT OPERATE RADIO TRANSMITTERS or CELL PHONES IN THE VICINITY OF THE ADD-32. The ADD-32 is an electronic device. Although it is designed to operate reliably in typical industrial environments, the ADD-32 can be affected by radio and/or cell phone transmitters. It is possible to cause drive faults, inappropriate/unintended drive I/O activity, and unpredictable operation that could result in damage to the ADD-32, damage to other equipment, or serious injury to personnel.

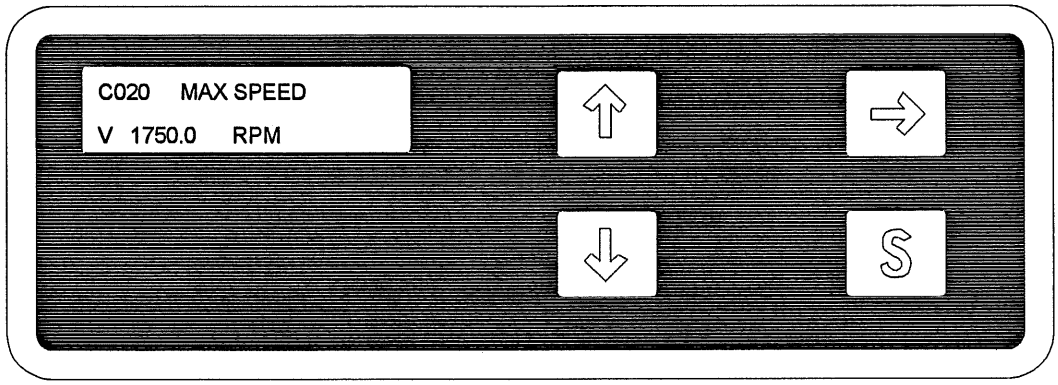
Radio transmitter interference is a site specific phenomena. Generally, electrical wires connected to terminals on the ADD-32 are the conduits for radio interference. Interference can be minimized by good wiring design and installation practice. It is recommended that signs be posted in and around the drive system, warning of the possibility of interference if the drive is in operation. DO NOT USE radio transmitters or cell phones in the area.

Absence of a radio interference problem is no guarantee that a problem will never occur as conditions and environments can change.

****

2.1 KEY FUNCTIONS

Information generated by the ADDvantage-32 can be accessed using the alphanumeric keypad and LCD display located on the front of the chassis. The LCD display provides two rows of sixteen alphanumeric characters. Editing of displayed information is accomplished using the four keys located next to the LCD display. Keystroke functions are as follows:



KYPDDISPD105

FIGURE 2-1. KEYPAD AND DISPLAY

The up arrow key moves up through the loop of options in the menus, submenus, and parameters.

The down arrow key moves down through the loop of options in the menus, submenus, and parameters.

The right arrow key moves from menus to submenus to parameters.

The "S" key (S = select) moves from parameter to submenu to menu.

ADDKFUNCLAN

2.2 MENUS AND SUBMENUS

Information used in the ADDvantage-32 is organized into a menu format. Each menu contains either a submenu or a list of parameters. Similar values are organized together to simplify the location of information.

Figure 2-2 illustrates the menu format used. All information is set up in a loop format for easy access. Path flow through the menus is indicated by the arrows on the chart. Press the right arrow key to access the submenus and parameter values. Use the up and down arrow keys to scroll through the information. To return to the main menu, press the "S" key.

For example, the following sequence is used to view information in the FAULT FIFO menu.

1. Press the up or down arrow keys to scroll through the main menu options.

2. When FAULT FIFO appears on the display, press the right arrow key.

3. Scroll through the submenu options using the up and down arrow keys until the VIEW FAULT FIFO submenu appears.

4. Press the right arrow key to enter the VIEW FAULT FIFO submenu.

5. Press the up or down arrow keys to scroll through the parameter information.

6. Press the "S" key twice to return to the FAULT FIFO main menu.

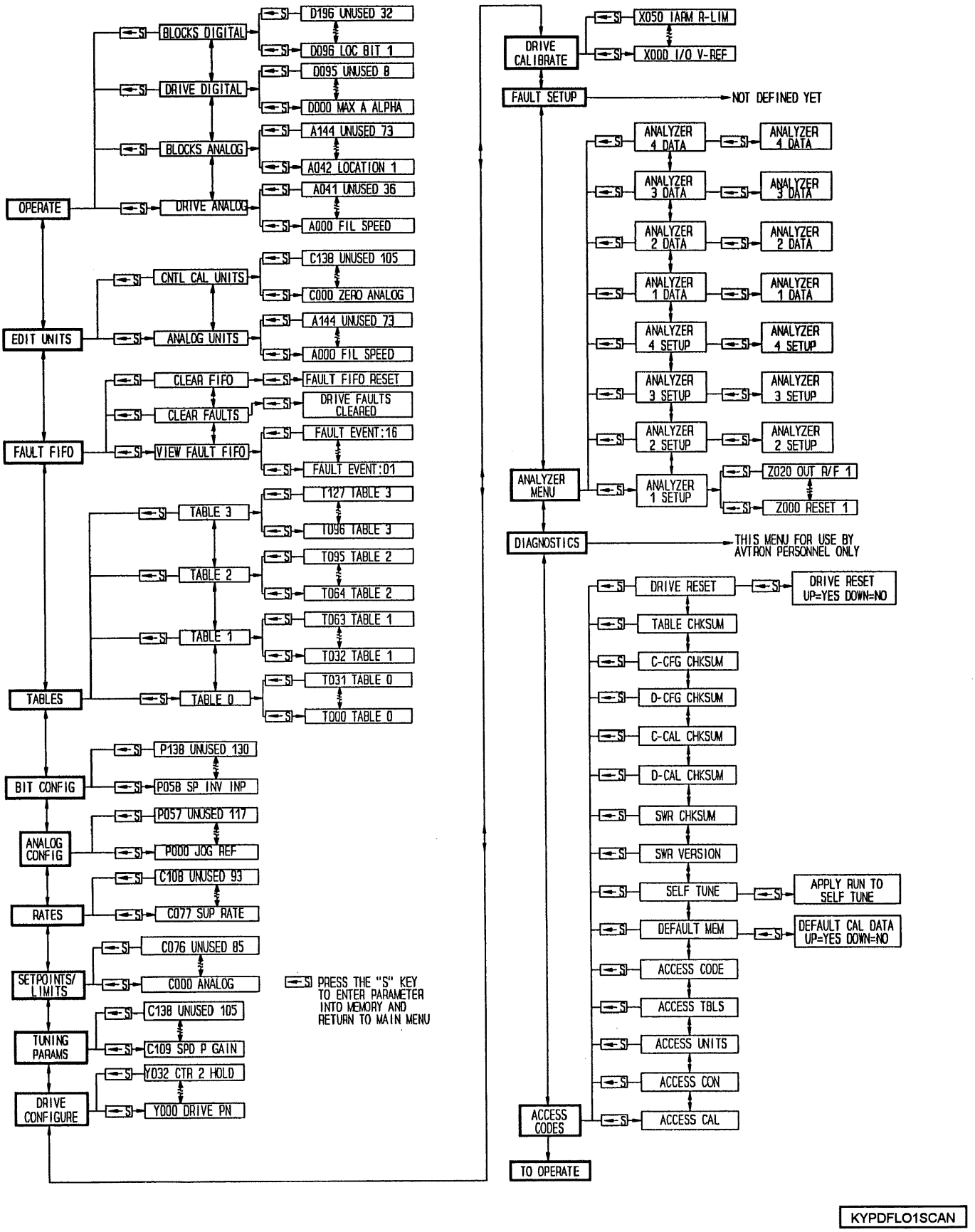


FIGURE 2-2. KEYPAD FLOWCHART

2.3 PARAMETER DEFINITION AND GROUPINGS

Parameters within the menus and submenus of the ADDvantage-32 are used to implement specific features in the application software.

There are several different types of parameters which are described as follows:

1. Calibration Parameters are numeric values entered for data such as limits, setpoints, or ramp rates.

2. Configuration Parameters are values used to select where information is going to come from or how specific features will function.

3. Analog Data is real time information contained in the ADDvantage-32 analog data table. This data is updated by the core software and hardwired I/O.

4. Digital Data is information contained in the ADDvantage-32 digital data table. This data is updated by the core software and hardwired I/O.

2.3.1 PARAMETER GROUPS

Parameters are defined by a four-digit code followed by a name or abbreviated description. The letter used as the first digit represents the particular group to which a parameter belongs. A typical parameter is shown as follows:

P000:JOG REF

Refer to Appendix C for specific parameter numbers and descriptions. Parameters are organized as shown in Table 2-1 and Figure 2-3.

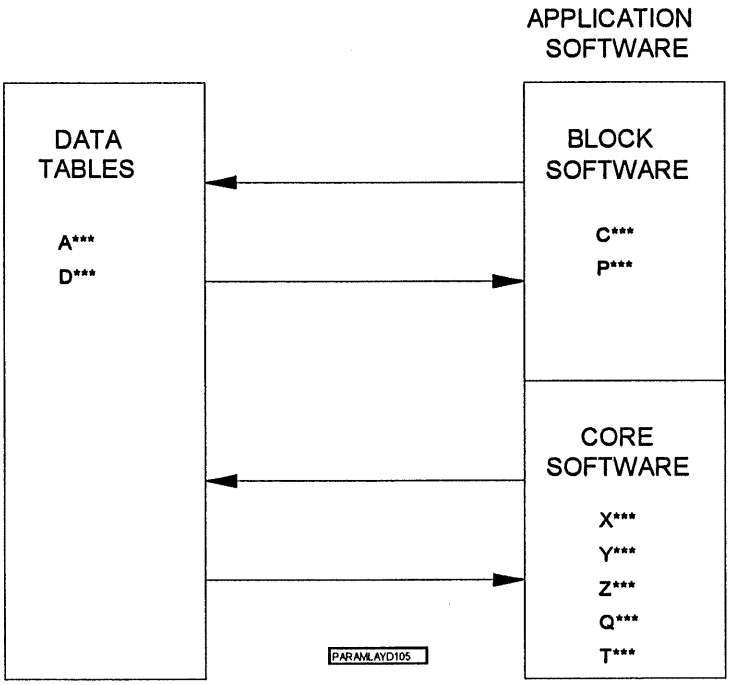


FIGURE 2-3. PARAMETER LAYOUT

TABLE 2-1. PARAMETER ORGANIZATION

PARAMETER GROUP	PARAMETER TYPE	DESCRIPTION
X***	Calibration	X*** parameters are part of the core software. Core parameters perform tasks specific to the ADDvantage-32 hardware being used.
Y***	Configuration	Y*** parameters are also part of the core software. These parameters set up the functionality of the ADDvantage-32 hardware.
C***	Calibration	C*** parameters are specific to the application block software being used. Many of these parameters are permanently fixed to specific blocks. Others can be configured using P*** parameters.

TABLE 2-1. PARAMETER ORGANIZATION (Cont.)

PARAMETER GROUP	PARAMETER TYPE	DESCRIPTION
P***	Configuration	P*** parameters are also specific to the type of application block software used with the ADDvantage-32. P*** parameters link specific blocks together, or configure a C*** parameter to another block.
A***	Analog Data	A*** parameters are read-only parameters. They represent all the information in the analog data table. P*** and Y*** parameters can map this information into the block software.
D***	Digital Data	D*** parameters are also read-only. They represent all the values in the digital data table. A D*** parameter has a value of either one or zero. A one means the condition is true. These values can also be mapped using the P*** and Y*** parameters.
T***	Calibration	T*** parameters are table entry values for the ADDvantage-32 tables. (Defined as X,Y points)
Z***	Calibration and Configuration	Z*** parameters are a mix of configurable and calibratable values for the four built-in signal analyzers.
Q***	Data Value	Q*** values are read-only parameters used by the Y*** and Z*** parameters. They represent selections available for certain parameters.
R***	LAN Auto Scan	R*** parameters configure the information to be accessed over the 802.4 LAN.

2.4 NUMERIC TYPE PARAMETER FORMAT

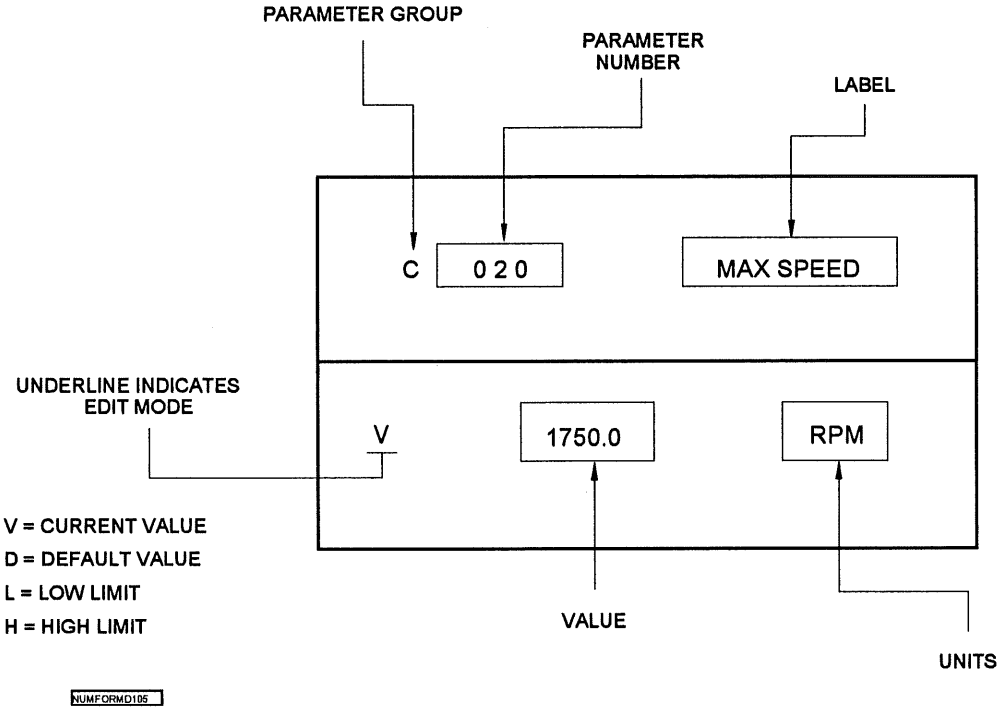


FIGURE 2-4. NUMERIC FORMAT

1. Parameter Group - Represents the type of data to be edited. (See Table 2-1.)

2. Parameter Number - Each parameter within a data type has a unique parameter number. (See Appendix A.)

3. Label - Each parameter has a unique eleven character label. (See Appendix B.)

4. Bottom Left Character - Determines what type of value is displayed. The only value which can be edited is "V" which represents current value.

5. Value - Represents the value of a particular parameter to nine significant digits.

6. Units - Represents the value's units of measure. This text can be edited using the Edit Units menu.

2.5 EDITING A NUMERIC PARAMETER

To edit a parameter, press the right arrow key. The bottom left most character will be underlined, indicating the location of the cursor. The cursor shows which character will be edited. Characters are edited one at a time.

NOTE

If access to the parameter is locked out, the message "ACCESS DENIED" will be displayed. To allow entry to the edit mode, the proper parameter must be enabled in the ACCESS CODES menu.

1. Press the right arrow key until the cursor underlines the digit to be changed.

2. Press the up or down arrow key to display the possible choices.

3. When a choice is made, press the right arrow key to move the cursor or press the "S" key to enter the choice and exit the edit mode.

2.6 CONFIGURATION TYPE PARAMETER FORMAT

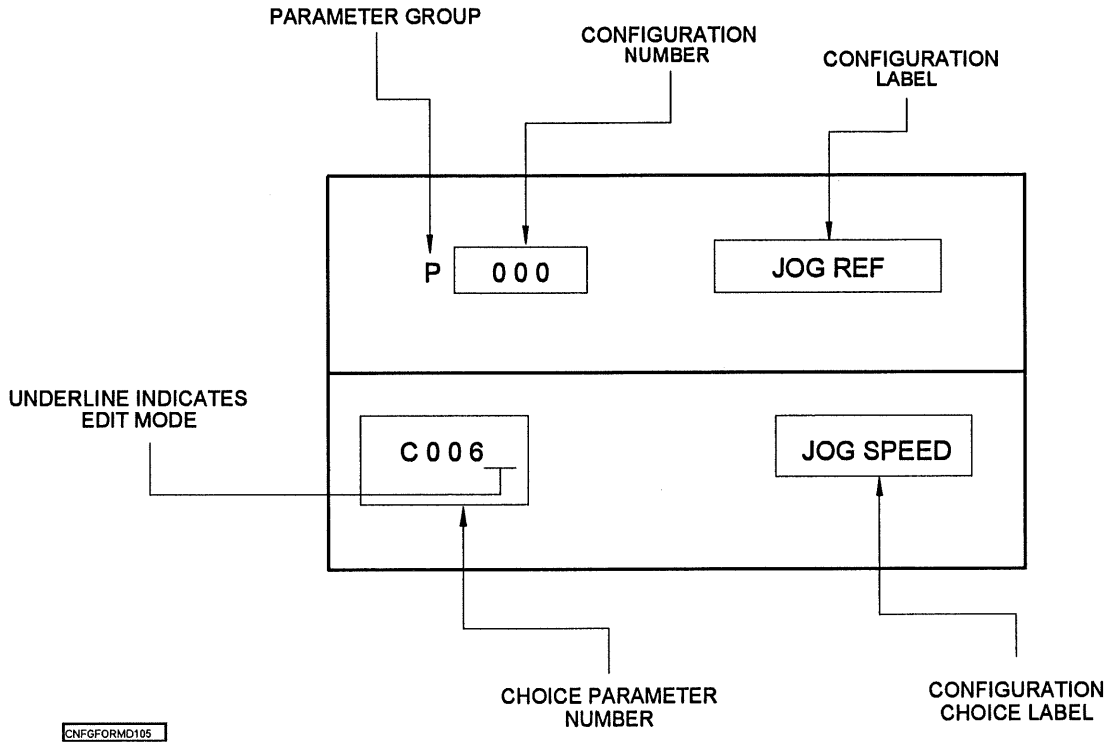


FIGURE 2-5. CONFIGURATION FORMAT

1. Parameter Group - Represents the type of data to be edited. (See Table 2-1.)

2. Configuration Number - Each configuration parameter has a unique number. (See Appendix A.)

3. Configuration Label - Each configuration parameter has a unique label. (See Appendix B.)

4. Choice Parameter Number - Represents data type and number of the configuration choice.

5. Configuration Choice Label - Each configuration parameter has a unique eleven character label.

2.7 EDITING A CONFIGURATION PARAMETER

1. Press the right arrow key to enter the edit mode at the parameter to be changed. Verify that an underline, indicating location of the cursor, appears beneath the last number of the parameter ID number.

2. Press the up or down arrow keys to view the configuration choices.

3. Press the "S" key to store the new configuration and to exit the edit mode.

****

C A U T I O N

THE NEW CONFIGURATION WILL BE USED ON THE NEXT POWERUP OF THE UNIT.

****

The keypad flowchart in Figure 2-2 indicates with directional arrows the path to follow for viewing/editing software parameters. Table 2-2 provides descriptions of the various menus, submenus, and parameters.

TABLE 2-2. PARAMETER DESCRIPTION CHART

NOTE: The following table describes the menus and submenus available on the ADDvantage-32. Ranges shown for the parameters vary for each application software part number. They are provided for sample purposes only. Actual parameter names may vary. Refer to Appendix D provided with your specific nomenclature.

MAIN MENU / SUBMENU	DESCRIPTION	RANGE OF PARAMETERS
EDIT UNITS	Allows the user to change the units of measure of the parameter shown on the display. (i.e. Changing ACT SPEED units from FPM to RPM.)	
CNTL CAL UNITS	Allows the user to change the units of measure for the calibration parameters.	C000 ZERO ANALOG | C124 TEN I GAIN
ANALOG UNITS	Allows the user to change the units for the analog data parameters.	A000 FIL SPEED | A132 UNUSED 61
FAULT FIFO	From this menu, the user can reset the FAULT FIFO, view the FIFO, or reset the drive faults.	
VIEW FAULT FIFO	The last sixteen events can be viewed from this submenu, last fault shown first. If ----- is displayed, FIFO has been cleared and is not filled with events.	FAULT EVENT:16 | FAULT EVENT:1
CLEAR FIFO	By pressing the right arrow key when in the submenu, the message FAULT FIFO RESET appears on display, erasing the events stored in the FIFO.	
FAULT FIFO RESET	Message FAULT FIFO RESET appears for approximately two seconds, indicating that the fault FIFO is being cleared.	
CLEAR FAULTS	By pressing the right arrow key, when in this submenu, the message DRIVE FAULTS CLEARED appears on the display. This will clear the present fault.	
DRIVE FAULTS CLEARED	Message DRIVE FAULTS CLEARED appears for approximately two seconds, indicating that the current faults are being cleared.	
TABLES	Four tables are available to perform nonlinear functions such as square root. Sixteen data points are available to describe the function in each table. A table is implemented by executing the TABLE BLOCK. (See Section 4.)	
TABLE 0	First of four user available tables. Enter the sixteen X and Y data points for TABLE 0.	T000 TABLE 0 X0 | T031 TABLE 0 Y15
TABLE 1	Second of four user available tables. Enter the sixteen X and Y data points for TABLE 1.	T032 TABLE 1 X0 | T063 TABLE 1 Y15

MAIN MENU / SUBMENU	DESCRIPTION	RANGE OF PARAMETERS
TABLE 2	Third of four user available tables. Enter the sixteen X and Y data points for TABLE 2.	T064 TABLE 2 XO | | T095 TABLE 2 Y15
TABLE 3	Fourth of four user available tables. Enter the sixteen X and Y data points for TABLE 3.	T096 TABLE 3 XO | | T127 TABLE 3 Y15
BIT CONFIG *	All digital application configuration parameters can be viewed and edited. Drive must be reset to update edited values. (i.e. Configuring SUP ENABLE to USER 3 allows the third digital input to implement a speed slack step.)	P058 SP INV INP | | P138 UNUSED 130
ANALOG CONFIG *	All analog application configuration parameters can be viewed and edited in this menu. Drive must be reset to update edited values. (i.e. Configuring SLACK UP to ANALOG IN 1 allows the first analog input to be the slack step value.)	P000 JOG REF | | P057 UNUSED 117
CALIBRATION 1 *	First application calibration parameter menu. The parameters are broken down into different menus to reduce the size of the queues.	C077 SUP RATE | | C108 UNUSED 93
CALIBRATION 2 *	Second application calibration parameter menu.	C000 ZERO ANALOG | | C076 UNUSED 85
DRIVE CONFIGURE	Configuration parameters for drive setup can be viewed and edited in this menu. The drive does not need to be powered up again for the new value to be used. (i.e. Configure RUN INPUT to USER 1.)	Y000 DRIVE PN | | Y032 CTR 2 HOLD
DRIVE CALIBRATE	Calibration parameters for drive setup and armature current loop tuning can be viewed and edited in this menu. (i.e. Scale the analog outputs.)	X000 I/0 V REF | | X050 IARM R-LIM
FAULT SETUP	Not defined yet.	
ANALYZER MENU	Allows user to set up, activate, and view the signal analyzer. There are setup and data submenus for each of the four analyzers.	

*Name can be different based on application. See Appendix D for correct menu name for a particular parameter.

MAIN MENU / SUBMENU	DESCRIPTION	RANGE OF PARAMETERS
ANALYZER 1 SETUP	Set up and activate signal analyzer 1. See Section V for operation of the Signal Analyzer.	Z000:RESET 1 | | Z020:OUT R/F 1
ANALYZER 2 SETUP	Set up and activate signal analyzer 2. Similar to the configuration for channel #1. See Section V for operation of the Signal Analyzer.	Z100:RESET 2 | | Z120:OUT R/F 2
ANALYZER 3 SETUP	Set up and activate signal analyzer 3. Similar to the configuration for channel #1. See Section V for operation of the Signal Analyzer.	Z200:RESET 3 | | Z220:OUT R/F 3
ANALYZER 4 SETUP	Set up and activate signal analyzer 4. Similar to the configuration for channel #1. See Section V for operation of the Signal Analyzer.	Z300:RESET 4 | | Z320:OUT R/F 4
ANALYZER 1 DATA	User can view data stored in analyzer. The bottom left character indicates the status of the analyzer as follows: U = Unused E = Enabled A = Armed and Enabled T = Triggered (gathering data) D= Done Collecting Data I = In Progress (outputting data) If the channel is done recording, the trigger event number and its value will be shown. Use the up/down arrow keys to view all 1000 events.	ANALYZER 1 DATA D500 100.01
ANALYZER 2 DATA	User can view data stored in analyzer 2.	ANALYZER 2 DATA D500 100.01
ANALYZER 3 DATA	User can view data stored in analyzer 3.	ANALYZER 3 DATA D500 100.01
ANALYZER 4 DATA	User can view data stored in analyzer 4.	ANALYZER 4 DATA D500 100.01
DIAGNOSTICS	This menu for use for LAN diagnostics. See Section VIII for details.	
ACCESS CODES	An access code is needed to allow entry into this menu. If the proper code is not entered, the message ACCESS DENIED will appear. User code 0.0 permits entry without a code. Code 216 allows entry into the menu at all times.	
ACCESS CAL	Allows modification of the calibration values.	Choices are NO, YES, and YES NONVOLATILE. If YES is selected it will default back to NO on the next drive powerup.
ACCESS CON	Allows modification of configuration parameters.	Choices are NO, YES, and YES NONVOLATILE. If YES is selected it will default back to NO on the next drive powerup.
ACCESS UNITS	Allows modification of parameter's units of measure such as Ft/Min of analog and calibration parameters.	Choices are NO, YES, and YES NONVOLATILE. If YES is selected it will default back to NO on the next drive powerup.

MAIN MENU / SUBMENU	DESCRIPTION	RANGE OF PARAMETERS
ACCESS TBLS	Allows modification of parameter values in the Tables menu.	Choices are NO, YES, and YES NONVOLATILE. If YES is selected it will default back to NO on the next drive powerup.
ACCESS CODE	Allows creation of a new access code for the main menu, which must be a 3-digit number. Access code 216 will always work with the user-defined code. Code may include decimal values along with 3-digit number.	Any number from 0-999 is valid.
DEFAULT MEM	Pressing the down arrow key (DOWN=NO) does not default the drive and returns to the submenu. Press the up arrow key (UP=YES) to default all drive calibration and configuration values. Record existing parameters prior to defaulting the drive or they will be lost. Drive will not default if run is energized; a warning will be issued if this occurs.	DEFAULT MEM UP=YES DWN=NO
SELF TUNE	To enable Self Tune, the following permissives must be met: bridge self test enabled, field disabled, and drive ESTOP picked up. Press the right arrow key to enable Self Tune. At that point, the drive run input will start Self Tune. To abort Self Tune at any time, remove the run or ESTOP input or press the S key. CAUTION Lock the motor from rotation. If the motor starts to rotate, abort the Self Tune routines by pressing "S", Remove Run, or Remove ESTOP. When the Self Tune is completed, the drive will trip out on either a TUNE FAIL or TUNE PASS condition. Reset the fault to continue. The Self Tune finds the following drive calibration parameters: :I ARM CONTIN :ARM RESIST :Z-C CORRECT :I ARM PGAIN :IARM IGAIN	SELF-TUNE PRESS "S" TO STOP
SWR VERSION	The first six numbers represent the Avtron software part number. The next two digits represent the version number of the software.	69XXXX (Software P/N) XX (Version No.)
SWR CKSUM	This parameter represents the software checksum of the EEPROM integrated circuit. The drive checks the actual checksum to this value upon powerup to determine the integrity of the program.	
D-CAL CKSUM	This is the total sum of all drive calibration parameters. The purpose of this parameter is to determine if any of the parameters have been altered. Record this number after setup of the drive.	
C-CAL CKSUM	This is the sum of all application control calibration parameters. Record this number after setup of the drive.	

MAIN MENU / SUB MENU	DESCRIPTION	RANGE OF PARAMETERS
D-CFG CKSUM	This is the sum of all drive configuration parameters. Record this number after setup of the drive.	
C-CFG CKSUM	This is the sum of all application configuration parameters. Record this number after setup of the drive.	
TABLE CKSUM	This is the sum of all table data points. Record this number after setup of the drive.	
DRIVE RESET	Pressing the down arrow key (DOWN=NO) does not reset the drive but returns to the submenu. Press the up arrow key (UP=YES) to reset the drive. Drive will not reset if the run is energized; a warning will be issued if this occurs.	RESET DRIVE UP=YES DWN=NO
OPERATE	Display real time drive data	Default menu upon initial powerup:
DRIVE ANALOG SUBMENU	Displays drive's real time analog data values such as line voltage, armature voltage and current, field voltage and current, I loop (CEMF, firing angles...)	A000 FIL SPEED | | A041 UNUSED 36
BLOCKS ANALOG SUBMENU	Displays all analog application data values generated and used in the software control blocks.	A042 LOCATION 1 | | A144 UNUSED 73
DRIVE DIGITAL SUBMENU	Displays digital status bits not associated with software control blocks.	D000 MAX A ALPHA | | D095 UNUSED 8
BLOCKS DIGITAL SUBMENU	Displays all digital application data values generated and used in the software control blocks.	D096 LOC BIT 1 | | D196 UNUSED 32

SECTION III

CORE SOFTWARE OPERATION

The core software performs the following fundamental functions of the ADDvantage-32 hardware and software:

1. Hardware Setup and Diagnostics
2. I/O Configuration and Calibration
3. Input and Output Scanning
4. Individual Control Block Operation
5. Four Channel Signal Analyzer Operation

This section describes the capabilities and functions of the core software and X*** and Y*** parameters, as well as the corresponding A*** and D*** data table parameters. (The data table parameters are shown for reference purposes only and are not explained in detail unless required.)

NOTE

Because parameter numbers vary according to software part number, specific parameter **numbers** are not shown. To locate a specific parameter number, refer to Appendix D and find the abbreviated parameter label and parameter number. Then refer to Appendix C to locate the parameter number, description, default value, and range of values.

3.1 DEDICATED INPUTS

Dedicated inputs are inputs located on the power supply board which perform a specific task. Each dedicated input is hard wired and operates independently of the application software, allowing failsafe operation even if a software problem occurs.

3.1.1 Emergency Stop (ESTOP)

When the ESTOP input is removed, the RUN command contact opens immediately. Notice that the green EMERG STOP LED on the front of the ADDvantage-32 goes out.

After an ESTOP condition has occurred, the ESTOP and ESTOP RESET inputs must be applied to reset the ESTOP circuit. This illuminates the green EMERG STOP LED.

3.1.2 Emergency Stop Reset (ESTOP RESET)

A normally open pushbutton should be used for the ESTOP RESET. If the ESTOP input and DOK are true, then pressing the pushbutton resets the circuit. A reset must also be performed if a drive fault has occurred.

3.2 DEDICATED OUTPUTS

The following dedicated outputs are Form C contact outputs used for ADDvantage-32 status indication. Each output is hard wired and is not user programmable.

3.2.1 Drive O.K.

This Form C contact output signifies that there are no fault conditions in the ADDvantage-32. If a critical fault occurs, this output contact opens and remains open until the fault is cleared.

3.2.2 RUN Command

Form C contact output is used to provide a RUN signal to existing hardware.

3.3 DIGITAL INPUTS

D*** Parameters

D***:USER 1
thru
D***:USER 14

The 24 VDC digital inputs are located on the ADDvantage-32 system board and are represented by a value in the digital data table. Each input is scanned by the core software to determine if it is in an ON or OFF state. Once this state is determined, the corresponding D***:USER X parameter is set in the digital data table. This value can then be used by other areas of the application software to enable specific functions.

Six digital inputs are located on the system board and are represented by parameters D:***USER 1 thru D***USER 6. Eight more inputs are available by adding the FAX-32 board. These inputs are represented by D***:USER 7 thru D***:USER 14 in the

digital data table. If the FAX-32 board is not used, the values for the additional eight parameters are always OFF.

3.4 DIGITAL OUTPUTS

Y*** Parameters

Y***:USER LED 1
Y***:USER LED 2
Y***:USER LED 3
Y***:USER LED 4
Y***:USER LED 5
Y***:USER LED 6
Y***:USER LED 7
Y***:USER LED 8
Y***:DIG OUT 1
Y***:DIG OUT 2
Y***:DIG OUT 3
Y***:DIG OUT 4

Digital outputs are Form C contact outputs located on the System Board and LED's located on the front of the ADDvantage-32. Each output is represented by a value in the digital data table. If the value in the data tables is ON, the output configured is also ON. When the data table value is OFF, the output is OFF. Outputs are always functional and can be disabled by configuration to either D***:ZERO BIT or D***:ONE BIT, which holds the output in a constant state.

For example, if a particular application requires that a contact output be closed when a fault occurs, perform the following:

1. Locate the value in the digital data table which signifies an ADDvantage-32 fault. (D***:FAULT)

2. Configure Y***:DIG OUT 1 to D***:FAULT. Any time a fault occurs, the first digital output closes.

3.5 LOGIC SEQUENCE

Y*** Parameters	D*** Parameters
Y***:RUN INPUT	
Y***:JOG INPUT	D***:ESTOP IN
Y***:THRD INPUT	D***:M TIMER PT
Y***:CLR FLT INP	D***:ARM ENABLE
Y***:USR LED 1	D***:DOK OUT
Y***:USR LED 2	D***:M CONTACTOR
Y***:USR LED 3	D***:USER 1 LED
Y***:USR LED 4	D***:USER 2 LED
Y***:USR LED 5	D***:USER 3 LED
Y***:USR LED 6	D***:USER 4 LED
Y***:USR LED 7	D***:USER 5 LED
Y***:USR LED 8	D***:USER 6 LED
	D***:USER 7 LED
	D***:USER 8 LED
	D***:RUN LED
	D***:RUN ENABLE
	D***:JOG ENABLE
	D***:THRD ENABLE
	D***:FAULT
	D***:RUNX
	D***:RUN REQUEST
	D***:USER 1
	D***:USER 2
	D***:USER 3
	D***:USER 4
	D***:USER 5
	D***:USER 6
	D***:CNTL INHIB
	D***:NOT USER 1
	D***:NOT USER 2
	D***:NOT USER 3
	D***:NOT USER 4
	D***:NOT USER 5
	D***:NOT USER 6

3.6 DRIVE SEQUENCE RUNG DESCRIPTIONS

Figure 3-1 is a ladder logic diagram of the internal interlocks for the core software. These rungs determine when specific features are enabled and disabled. D*** parameters listed are values found in the digital data table. See Section II and Appendix C for parameter explanations. Following is a brief description of each rung.

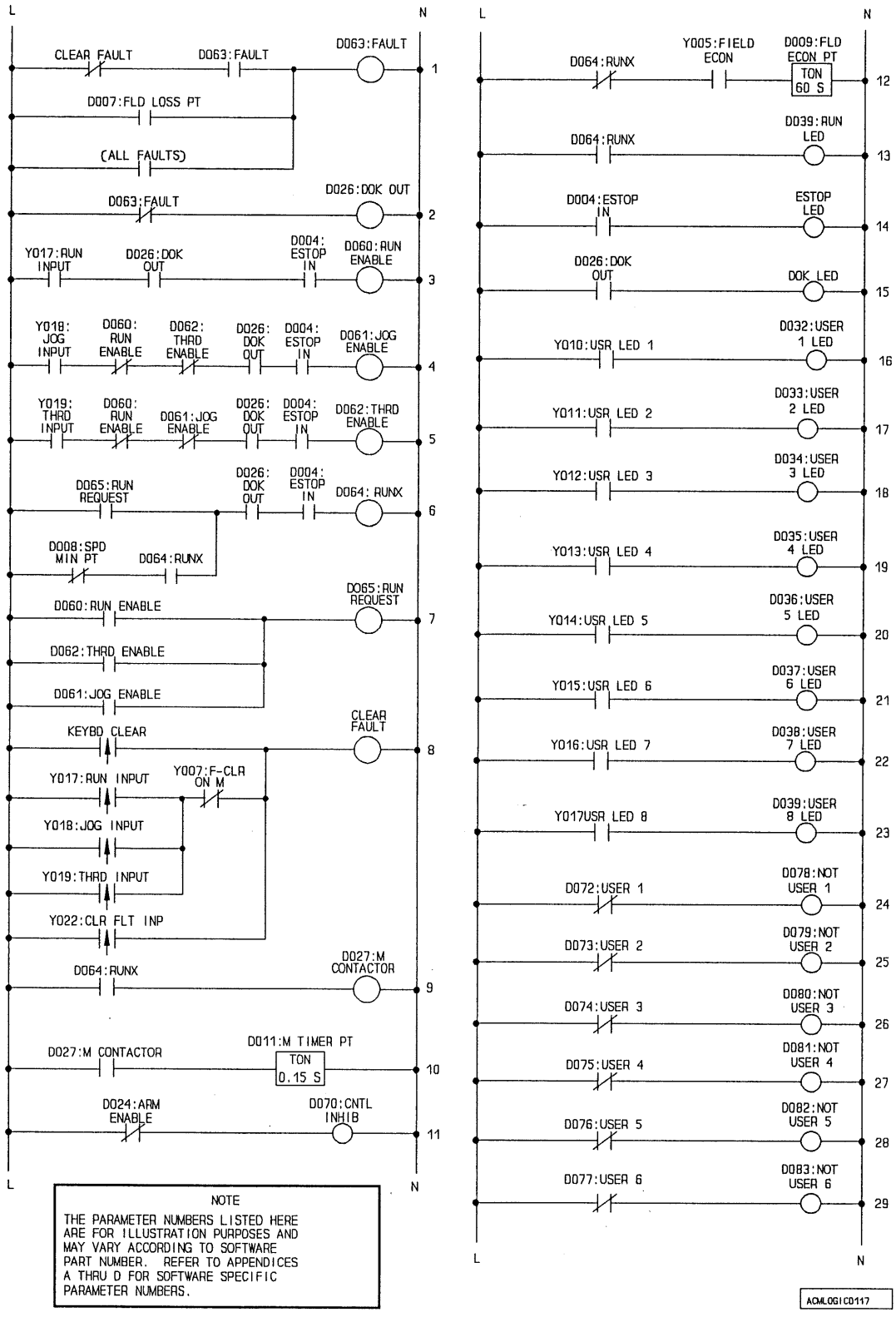


FIGURE 3-1. ADDvantage-32 ADVANCED CONTROL MODULE LOGIC

3.6.1 Rung 1 - (FAULT) Any fault that shuts down the ACM picks up this FAULT bit. The bit seals itself in until CLEAR FAULT goes high.

3.6.2 Rung 2 - (DOK_OUT) The drive ready output drops out when the FAULT bit goes high.

3.6.3 Rung 3 - (RUN ENABLE) This bit picks up the RUN command contact and enables the run reference to the speed loop. If the ACM is faulted or ESTOPped, this bit cannot be enabled.

3.6.4 Rung 4 - (JOG ENABLE) This bit picks up the RUN command contact and enables the jog reference to the speed loop. If the ACM is faulted, ESTOPped, running, or in thread mode, this bit cannot be enabled.

3.6.5 Rung 5 - (THRD ENABLE) This bit picks up the RUN command contact and enables the thread reference to the speed loop. If the ACM is faulted, ESTOPped, running, or in jog mode, this bit cannot be enabled.

3.6.6 Rung 6 - (RUNX) The RUNX bit is set high when the RUN REQUEST is high and there is no fault. When the RUN REQUEST goes low, the RUNX stays in until the speed feedback goes to zero, keeping the RUN command contact enabled. If there is a fault or an ESTOP, RUNX immediately drops out, causing a dynamic braking stop.

3.6.7 Rung 7 - (RUN REQUEST) Bit is set when run, jog, or thread is requested.

3.6.8 Rung 8 - (CLEAR FAULT) Bit clears the fault interlock on the ACM and is enabled by any of the following conditions:

1. A keyboard bit can clear the fault.
2. If selected by a keyboard parameter, turning the RUN command back on will clear the fault.
3. Setting the CUR FLT INP by the serial link or LAN will clear the fault.

3.6.9 Rung 9 - (M CONTACTOR) The M CONTACTOR picks up when RUNX is picked up, closing the RUN command contact.

3.6.10 Rung 10 - (M TIMER PT) This timer delays the ARM ENABLE for 150 ms, giving the motor contactor time to close.

3.6.11 Rung 11 - (CNTL INHIB) This coil picks up when ARM ENABLE drops out and is used to reset integrators in the control loops.

3.6.12 Rung 12 - (FLD ECON PT) This timer delays the field economy from coming on until 60 seconds after the RUNX drops out.

3.6.13 Rung 13 - (RUN LED) The green RUNNING LED on front panel illuminates when the RUNX coil is picked up.

3.6.14 Rung 14 - (ESTOP LED) If the ESTOP IN input is high and ESTOP RESET has been pressed, the green EMERG STOP LED on the front panel will illuminate.

3.6.15 Rung 15 - (DOK LED) The green DRIVE READY LED illuminates when the DOK_OUT output goes high.

3.6.16 Rungs 16-23 - (USER LED) When the associated USER LED PT bit goes high, the USER LED illuminates. Any bit in the data table can be connected to the USER LED points making these configurable LED'S.

3.6.17 Rungs 24-29 - (NOT USER 1-6) These coils are picked up when their associated DIGITAL IN contacts are not picked up.

3.7 RUN, THREAD, AND JOG INPUTS

Y*** Parameters

Y***:RUN INPUT
Y***:JOG INPUT
Y***:THRD INPUT

There are three different parameters which enable operation of the ADDvantage-32. Each parameter can be configured to a value in the digital data table which enables its particular function. These three parameters are interlocked. Therefore, only one can be enabled at a time. Consult the Logic Sequence description provided in this section for specific interlock information. Each parameter functions as follows:

Y***:RUN INPUT - This parameter is configured to the digital data table value which runs the ADDvantage-32. When the data table value goes to an ON state, the speed loop is enabled. The block portion of the application software then determines how the unit will operate.

Y***:JOG INPUT - This parameter functions in the same manner as the run input. This parameter provides a separate input for jogging the section.

Y***:THRD INPUT - This parameter functions in the same manner as the jog input, providing a third input for threading the section.

For example, assume the following inputs are being used to perform a listed function:

Digital Input 1 = Run Section
Digital Input 2 = Jog Section
Digital Input 3 = Thread Section

1. Find the values for the three digital inputs in the Digital Data Table: D***:USER 1 for digital input 1, D***:USER 2 for digital input 2, and D***:USER 3 for digital input 3.

2. Configure the parameters as follows:

Y***:RUN INPUT = D***:USER 1
Y***:JOG INPUT = D***:USER 2
Y***:THRD INPUT = D***:USER 3

3.8 ANALOG INPUTS

X*** Parameters	A*** Parameters
X***:INPT #1 CAL	A***:ANALOG IN 1
X***:INPT 1 ZERO	A***:ANALOG IN 2
X***:INPT #1 TC	A***:ANALOG IN 3
X***:INPT #2 CAL	A***:ANALOG IN 4
X***:INPT #2 ZERO	A***:ANALOG IN 5
X***:INPT #2 TC	A***:ANALOG IN 6
X***:INPT #3 CAL	
X***:INPT #3 ZERO	
X***:INPUT #3 TC	
X***:INPUT #4 CAL	
X***:INPUT #4 ZERO	
X***:INPUT #4 TC	
X***:INPUT #5 CAL	
X***:INPUT #5 ZERO	
X***:INPUT #5 TC	
X***:INPT #6 CAL	
X***:INPT #6 ZERO	
X***:INPT #6 TC	

There are six -10 to +10 VDC analog inputs located on the ADDvantage-32 system board. Each input is scanned, calibrated, and written into the analog data table to be used

by the application software. The value of the input is determined by the following equation:

```
A***:ANALOG IN 1 =
      Input Volts (X***:INPT #1 CAL) + X***:INPT 1 ZERO
```

For example, to scale analog input #1 so that a -10 VDC to 10 VDC signal reads 0 to 2500 FPM in the analog data table, perform the following:

1. Find A***:ANALOG IN 1 in the analog data table.

2. Use this information in the following equations:

```
   0 FPM = -10VDC (X***:INP #1 CAL) +X***:INPT 1 ZERO
2500 FPM =  10VDC (X***:INP #1 CAL) +X***:INPT 1 ZERO
```

   Solving the above equations yields the following results:

```
X***:INPT #1 CAL = 125 FPM/VDC
X***:INPT 1 ZERO = 1250 FPM
```

3.9 ANALOG OUTPUT FUNCTIONS

X*** Parameters	Y*** Parameters
X***:OUTPT 1 CAL	Y***:ANLG OUT 1
X***:OUTPT 1 ZER	Y***:ANLG OUT 2
X***:OUTPT 2 CAL	Y***:ANLG OUT 3
X***:OUTPT 2 ZER	Y***:ANLG OUT 4
X***:OUTPT 3 CAL	
X***:OUTPT 3 ZER	
X***:OUTPT 4 CAL	
X***:OUTPT 4 ZER	

Analog outputs read information from the analog data table, scale it, and produce a -10 VDC to +10 VDC output signal. Each analog output has a parameter to configure it to the data table value that will be represented. Two additional parameters scale the value into the voltage signal. The output is scaled by the following function:

```
Output VDC =
  (Data Table Value + X***:OUTPUT 1 ZERO) X***:OUTPT 1 CAL
```

For example, to generate a -10 VDC to +10 VDC signal from the the second analog output of the ADDvantage-32 system board which represents actual speed (0 - 2500 FPM), perform the following:

1. Find A***:ACT SPEED in the analog data table.

2. Configure Y***:ANLG OUT 2 to A***:ACT SPEED. This will send the output to the second analog output.

3. Find the calibration values as follows:

```
-10VDC=(   0 FPM+X***:OUTPUT 1 ZERO)(X***:OUTPT 1 CAL)
+10VDC=(2500 FPM+X***:OUTPUT 1 ZERO)(X***:OUTPT 1 CAL)
```

Solving the above equations yields the following:

```
X***:OUTPT 1 CAL  = .008 VDC/FPM
X***:OUTPT 1 ZERO = -1250 FPM
```

3.10 FREQUENCY INPUTS

X*** Parameters	Y*** Parameters	A*** Parameters
X***:TACH 1 CAL	Y***:TACH 1 TYPE	A***:SPEED IN 1
X***:TACH 1 ZERO	Y***:TACH 2 TYPE	A***:SPEED IN 2
X***:TACH 2 CAL		
X***:TACH 2 ZERO		

Frequency inputs convert an incoming pulse train into an analog value by counting the number of pulse edges and calibrating it into a user value. Counting the number of edges allows for a more accurate representation of the incoming pulse train. The analog value is then stored in the analog data table.

There are two methods used for counting the pulse train edges: the averaging method and the direct count method. Configuring Y***:TACH 1 TYPE and Y***:TACH 2 TYPE determines which method is used. Both methods measure one or two phase inputs and one or two pulse edges.

3.10.1 AVERAGING METHOD (TWO PHASE, 1 PHASE 2X, 1 PHASE 1X)

When these options are selected for the TACH TYPE, the pulse train edges are counted and averaged over an internal sample period. This averaging method allows for smoother control but can lead to inaccuracies if the duty cycle of the pulse is not 50/50. Selection of the 1 PHASE 1X (one edge) option

eliminates these inaccuracies but decreases the resolution. This method should be used when the frequency input is used for speed feedback and the frequency being applied is fairly low. It should also be used for feedback applications that have significant cyclical resonance on the feedback.

3.10.2 DIRECT COUNT METHOD (TWO PHASE-S, 1 PHASE-2XS, 1 PHASE-1XS)

If these options for TACH TYPE are selected, the exact number of edges that occur in the same internal sample period are counted. This method leads to a more accurate count of the pulse train input but is not quite as smooth as the above method. This method should be used for all applications other than those stated for the averaging method.

Calibrate the pulse train into a user value by using the following equation:

```
A***:SPEED IN 1=
  [(Input Freq. * (1,2 or 4)♦-X***:TACH 1 ZERO]*X***:TACH 1 CAL

     ♦1 is used for one phase, one edge applications
      2 is used for one phase, two edge applications
      4 is used for two phase, two edge applications
```

For example, if a 2-phase M737A pulse generator provides feedback to an ADDvantage-32 and the pulse generator is mounted to a motor running at 1200 RPM moving the process at 2000 FPM, use the following steps to scale the frequency input so that the analog value generated is in process units.

1. Locate A***:SPEED IN 1, the analog data table value for the first frequency input. This value must read 0 - 2500 FPM.

2. Calculate the frequency for the speed desired. The M737A is a 240 PPR tach; therefore, the frequency would be calculated as follows:

$$\text{Frequency} = \frac{\text{Motor RPM} * \text{Tach PPR}}{60 \text{ SEC/MIN}} = \frac{1200 \text{ RPM} * 240 \text{ PPR}}{60 \text{ SEC/MIN}}$$

   Frequency = 4800 HZ

3. Select the TACH TYPE required. The M737A is a 2-phase tach, so two phases will be used. Frequency is fairly high, so the direct count method will be used. Configure Y***:TACH 1 TYPE to Q***:2 PHASE-S.

4. Calibrate the frequency input by inserting the numbers into the equation:

```
2500 FPM = [(4800 HZ*4) - X***:TACH 1 ZERO] * X***:TACH 1 CAL

   0 FPM = [(   0 HZ*4) - X***:TACH 1 ZERO] * X***:TACH 1 CAL
```

Solving the above equations yields:

```
X***:TACH 1 CAL  = 0.1302
X***:TACH 1 ZERO = 0.0000
```

3.11 COUNTERS

X*** Parameters	Y*** Parameters	Analog Data Table
X***:COUNT 1 CAL	Y***:CTR 1 RESET	A***:FOOTAGE 1
X***:COUNT 2 CAL	Y***:CTR 1 HOLD	A***:FOOTAGE 2
	Y***:CTR 2 RESET	
	Y***:CTR 2 HOLD	

A counter is a device that counts units based on the number of pulses detected at the frequency inputs. The frequency inputs correspond to an associated counter. The counter increases one unit for every pulse edge detected. The increment value is determined as follows:

A***:FOOTAGE 1 = (1,2 or 4)X***:COUNT 1 CAL

Where: 1 - Is used when the frequency input is set up for 1 PHASE, one edge operation.

2 - Is used when the frequency input is set up for 1 PHASE, two edge operation.

4 - Is used when the frequency input is set up for 2 PHASE, two edge operation.

Each counter can be reset or held by configuring Y***:CTR 1 RESET and Y***:CTR 1 HOLD to the appropriate bit in the digital data table.

For example, a 2-phase tach is connected to frequency input number 2 and produces 750 pulses for every foot of product. A reset button is wired into the first digital input on the system board. The counter is to be held any time the second digital input is on. To calibrate and configure the counter to count inches of product, perform the following:

1. Locate D***:USER 1 and D***:USER 2, the parameters in the digital data table which represent the two digital inputs being used.

2. Configure Y***:CTR 2 RESET to D***:USER 1. Configure Y***:CTR 2 HOLD to D***:USER 2.

   This will reset the counter when the first digital input turns on and will hold the counter when the second input is on.

3. To determine the calibration number for the counter, the length per pulse must be found. It is given that 750 pulses = 1 foot. Converting this into inches yields:

   $$\frac{750 \text{ pulses/ft}}{12 \text{ inch./ft}} = 62.5 \text{ pulses/inch or}$$

   1 pulse = 1/(62.5 pulses/inch) = .016 inches

   The increment value for the counter is .016 inches. The counter increases by this number for every pulse edge detected. To determine the calibration parameter, set the equation up for one increment value as follows:

   A***:FOOTAGE 2 = .016 inches = (4)*X***:COUNT 2 CAL

   X***:COUNT 2 CAL = .004

   This value causes the counter to increment 0.016 inches for every pulse detected at the frequency input.

3.12 HARDWARE CONFIGURATION AND DIAGNOSTIC PARAMETERS

Y*** Parameters	A*** Parameters
Y***:LINE FREQ	
Y***:ARM ENABLE	A***:LINE FREQ
Y***:LINK BAUD	
Y***:SPD OPEN MC	
Y***:F-CLR ON M	

Hardware configuration parameters are used to set up fundamental operations of the ADDvantage-32. Refer to Appendix C for an explanation of these parameter functions.

3.13 USER FAULT INPUTS

Y*** Parameters

Y***:USR FAULT 1
Y***:USR FAULT 2
Y***:USR FAULT 3
Y***:USR FAULT 4
Y***:USR FAULT 5
Y***:USR FAULT 6
Y***:USR FAULT 7
Y***:USR FAULT 8

User fault inputs allow the user to enable fault conditions beyond those generated internally by the ADDvantage-32. When the fault condition occurs, the ADDvantage-32 faults out and a message is stored in the FAULT FIFO. The ADDvantage-32 must be reset before operating again.

For example, two of the most common user faults are overspeed and tach loss conditions. To set up the ADDvantage-32 to fault on these two conditions, perform the following:

1. Find D***:OVER SPEED and D***:TACH LOSS, the two parameters in the digital data table that represent overspeed and tach loss.

2. Configure D***:OVER SPEED to Y***:USR FAULT 1. Configure D***:TACH LOSS to Y***:USR FAULT 2.

When either one of these conditions occurs, the ADDvantage-32 faults out.

3.14 MOTOR THERMAL SETTINGS

X*** Parameters	A*** Parameters
X***:THERMAL TC	A***:IIR INTEGR

Thermal capacity of the ADDvantage-32 is represented by an internal counter called the I^2R integrator. Any time the ADDvantage-32 is operating at over 100% of the motor's rated armature current, the integrator begins counting up. If the integrator counts up to 100, the ADDvantage-32 will generate a fault and shut down. If the current goes below 100% motor rating before the counter reaches 100, it begins to count down in value until it reaches zero.

X***:THERMAL TC - Used to set up the I^2R integrator, it is the amount of time the ADDvantage-32 can operate before a value of 100 is obtained. If the heat sink temperature rises above the safe limit before the I^2R integrator reaches 100, the ADDvantage-32 trips out on a fault.

3.15 MOTOR STALL PROTECTION

X*** Parameters	Y*** Parameters	D*** Parameters
X***:STALL SPEED	Y***:STALL PROT	D***:MOTOR STALL
X***:STALL % AMP		
X***:STALL TIME		

Motor stall is used to protect a motor that is frozen or overloaded. If left in this condition, the motor can be permanently damaged.

Motor stall is enabled or disabled by parameter Y***:STALL PROT. A stall condition is realized when:

% Armature Current Command	> X***:STALL % AMP
Motor Speed (In Process Units)	< X***:STALL SPEED
Elapsed Time	> X***:STALL TIME

When a motor stall is determined, digital data bit D***:MOTOR STALL turns on. This bit can then be configured to a USER FAULT input or digital output for fault indication.

SECTION IV

CONTROL BLOCK DESCRIPTION

Table of Contents

Table of Contents (Cont.)

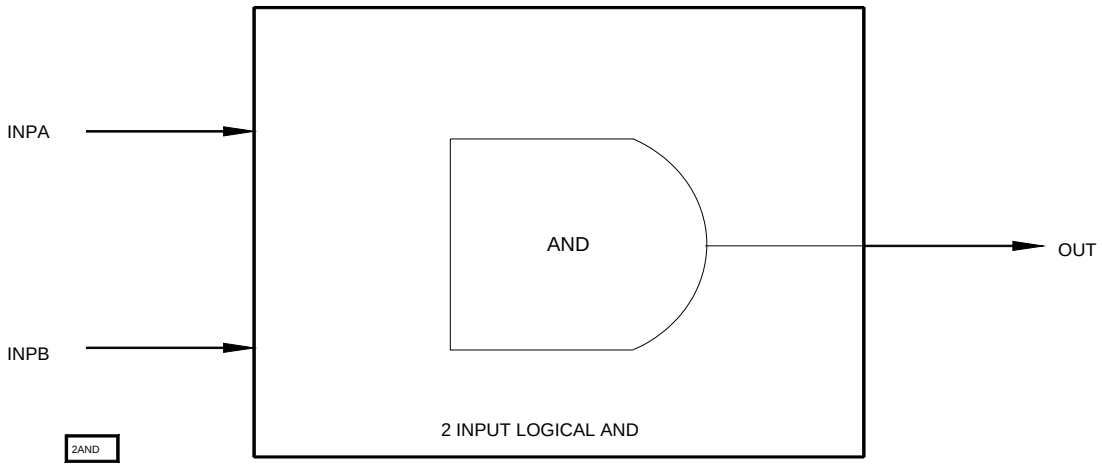
SECTION IV

CONTROL BLOCK DESCRIPTION

The ADDvantage-32 application software control scheme is based on control blocks. A control block is a software procedure which takes the inputs to the block, performs its function, and outputs the results. The following control blocks are not used in all software applications. They are a combination of all available control blocks. Refer to Appendix A for software specific control block interconnections.

4.1 2 AND

This block implements a 2 input digital AND gate.



FI

GURE 4-1. 2 AND BLOCK

1. Inputs

INPA: Bit
INPB: Bit

2. Outputs

   OUT: Bit

3. Implementation

   OUT is set to one if both INPA and INPB are equal to one.

   OUT is set to a zero bit if either INPA or INPB is equal to zero.

4.2 2 OR

This block implements a 2 input digital OR gate.

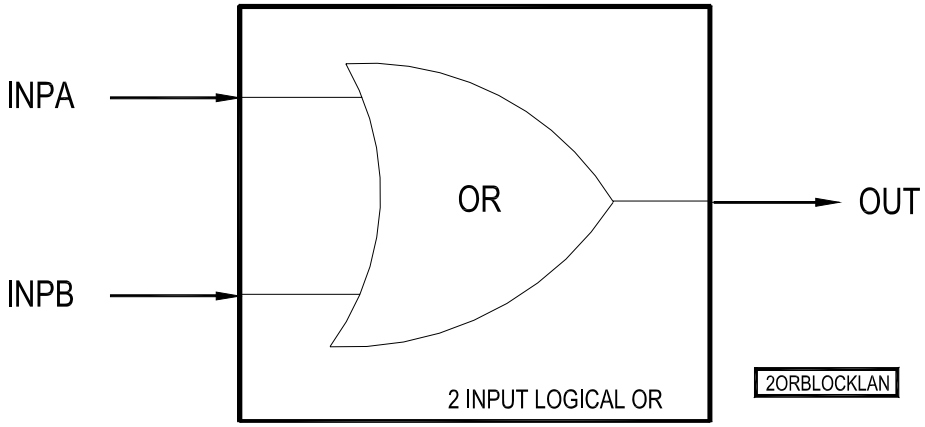


FIGURE 4-2. 2 OR BLOCK

1. Inputs

   INPA: Bit
   INPB: Bit

2. Outputs

   OUT: Bit

3. Implementation

   OUT is set to one if either INPA or INPB is equal to one, else OUT = 0.

4.3 4 ANALOG SELECT

This block is used to select one of four possible analog signal paths. It can be used to control the application of multiple reference signals to a single input point.

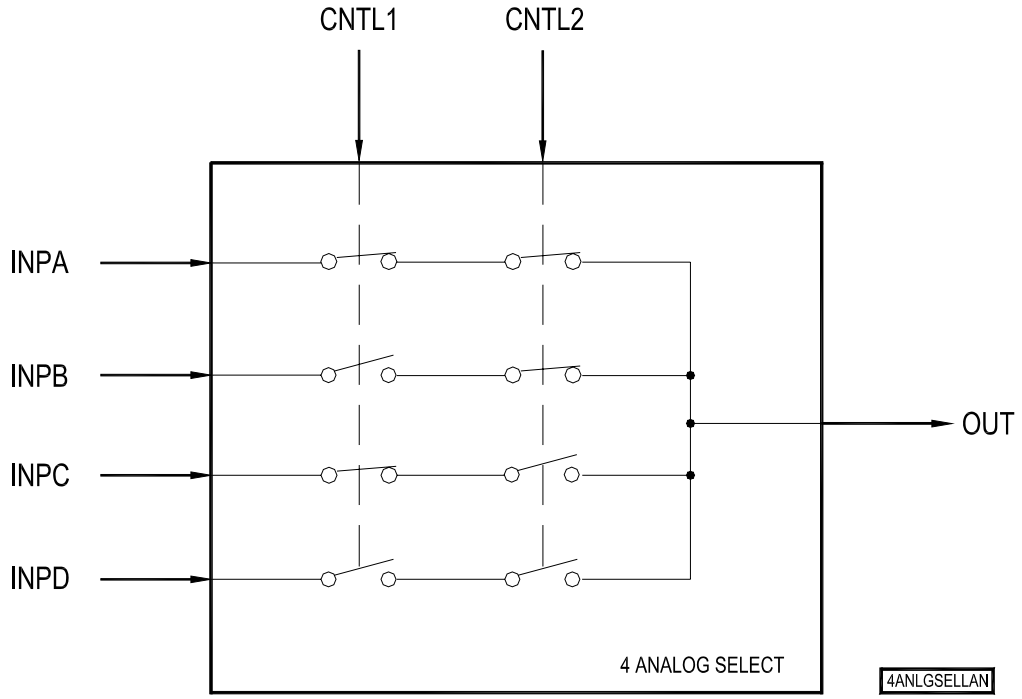


FIGURE 4-3. 4 ANALOG SELECT BLOCK

1. Inputs

INPA:	Analog
INPB:	Analog
INPC:	Analog
INPD:	Analog
CNTL1:	Bit
CNTL2:	Bit

2. Output

OUT:	Analog

3. Implementation

One of the input signals will be directed to the block output by the following combinations of control bits.

CNTL1	CNTL2	OUT
0	0	INPA
1	0	INPC
0	1	INPB
1	1	INPD

4.4 5 AND

This block implements a 5 input digital AND gate.

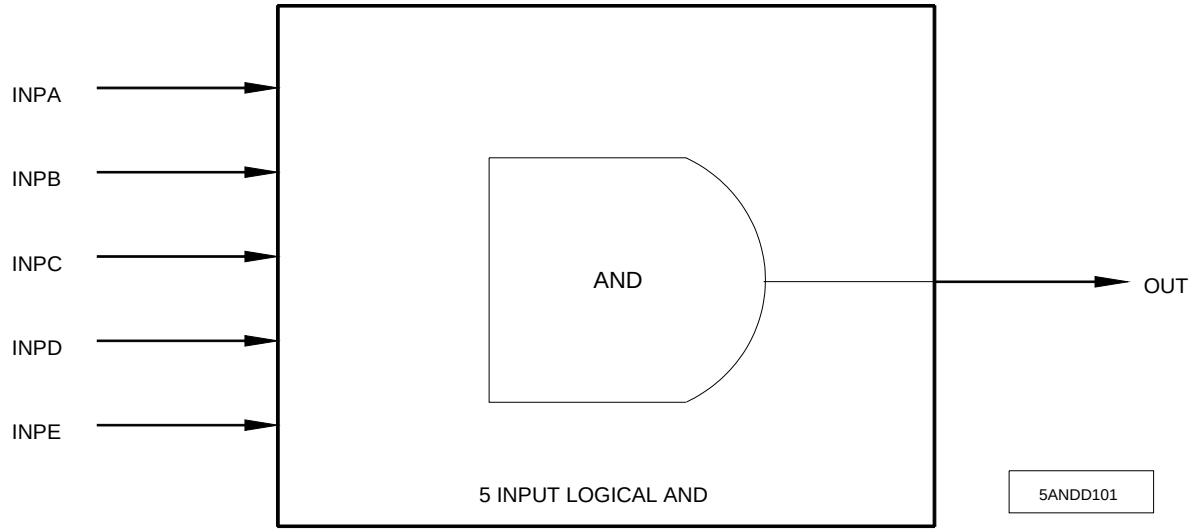


FIGURE 4-4. 5 AND BLOCK

1. Inputs

   INPA: Bit
   INPB: Bit
   INPC: Bit
   INPD: Bit
   INPE: Bit

2. Outputs

   OUT: Bit

3. Implementation

   OUT = 1 if INPA, INPB, INPC, INPD, and INPE are equal to one.

   OUT = 0 if any input is equal to zero.

4.5 5 SUMMER

The 5 Summer block selectively sums up to five analog input signals. The selection of which inputs are to be summed is set using a series of digital bits.

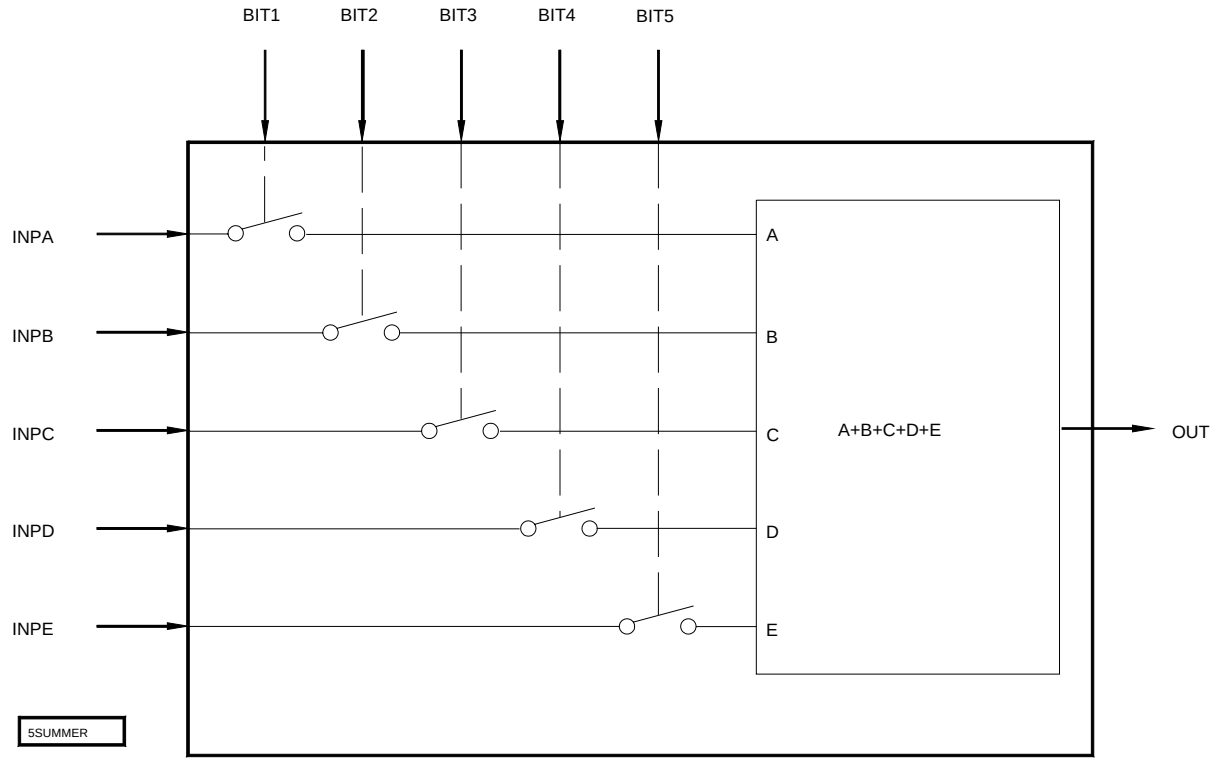


FIGURE 4-5. FIVE SUMMER BLOCK

1. Inputs

INPA:	Analog
INPB:	Analog
INPC:	Analog
INPD:	Analog
INPE:	Analog
BIT1:	Bit
BIT2:	Bit
BIT3:	Bit
BIT4:	Bit
BIT5:	Bit

2. Output

OUT: Analog

3. Implementation

OUT = (INPA × BIT1) + (INPB × BIT2) + (INPC × BIT3) + (INPD × BIT4) + (INPE × BIT5)

If all 5 bits are low, then OUT = 0.

4.6 8 BIT INVERT

The 8 BIT INVERT block takes the INP bit and the next seven and inverts them.

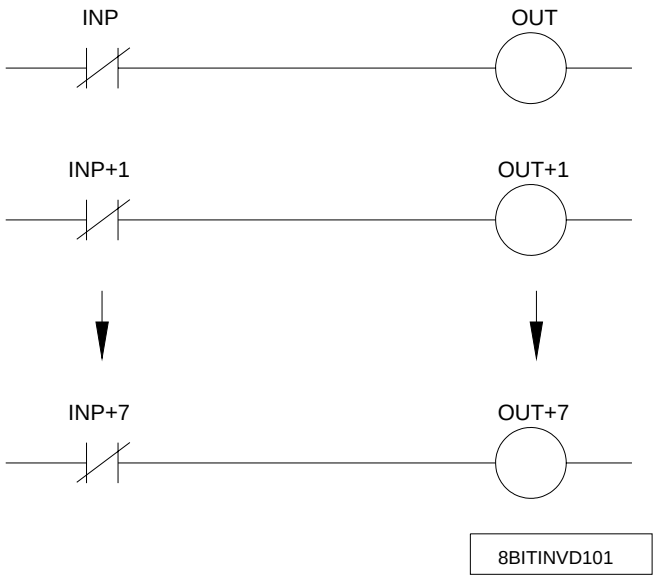


FIGURE 4-6. 8 BIT INVERT BLOCK

1. Inputs

   INP: Digital

2. Outputs

   OUT: Digital

3. Implementation

   If INP is high (1), then OUT will be low (0).
   If INP is low (0), then OUT will be high (1).

   The same occurs for the next seven input bits. They are outputted to the next seven output addresses.

4.7 ABSOLUTE VALUE (ABS)

The Absolute Value block selectively takes the absolute value of an analog variable. The state of EN BIT determines if the OUT value equals INP or the absolute value of the INP value.

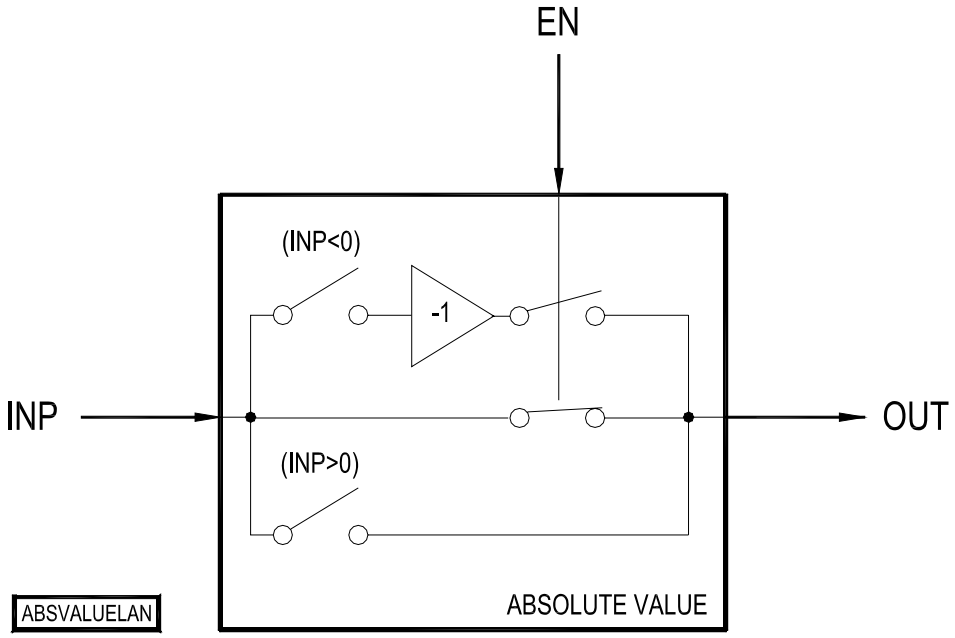


FIGURE 4-7. ABSOLUTE VALUE BLOCK

1. Inputs

   INP: Analog
   EN: Bit

2. Output

   OUT: Analog

3. Implementation

   If EN is low, then OUT = INP.

   If EN is high, then OUT = the absolute value of INP.

4.8 ANALOG INVERT

The Invert block is used to invert the value of an analog signal.

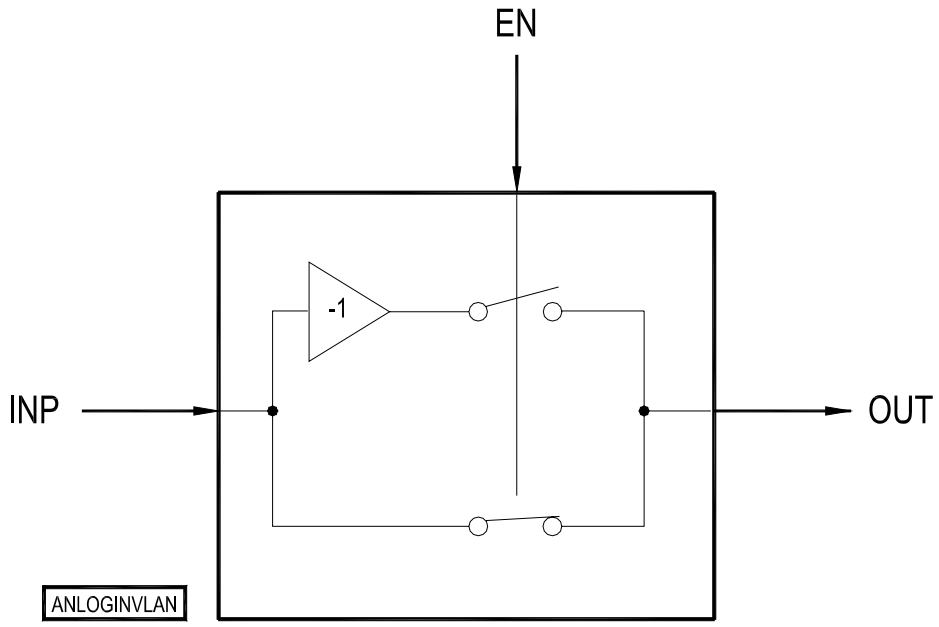


FIGURE 4-8. ANALOG INVERT BLOCK

1. Inputs

   INP: Analog
   EN: Bit

2. Outputs

   OUT: Analog

3. Implementation

   If EN is low, then OUT = INP.

   If EN is high, then OUT = -INP.

4.9 ANALOG SELECT

This block is used to select one of two different analog signal paths; for example, switching from field current reference to field economy reference.

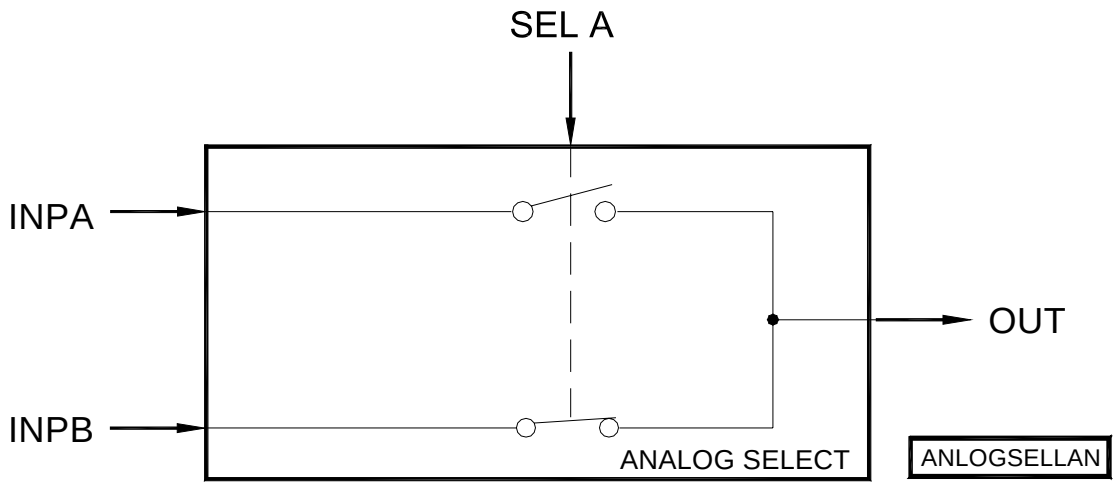


FIGURE 4-9. ANALOG SELECT BLOCK

1. Inputs

   INPA: Analog
   INPB: Analog
   SELA: Bit

2. Outputs

   OUT: Analog

3. Implementation

   If SELA bit is high (set at 1), the OUTPUT is equal to INPA. If SELA bit is low (set at 0), the OUTPUT is equal to INPB.

4.10 ANALOG SWITCH

This block is used to switch in references to control the passage of an analog value or signal between control blocks.

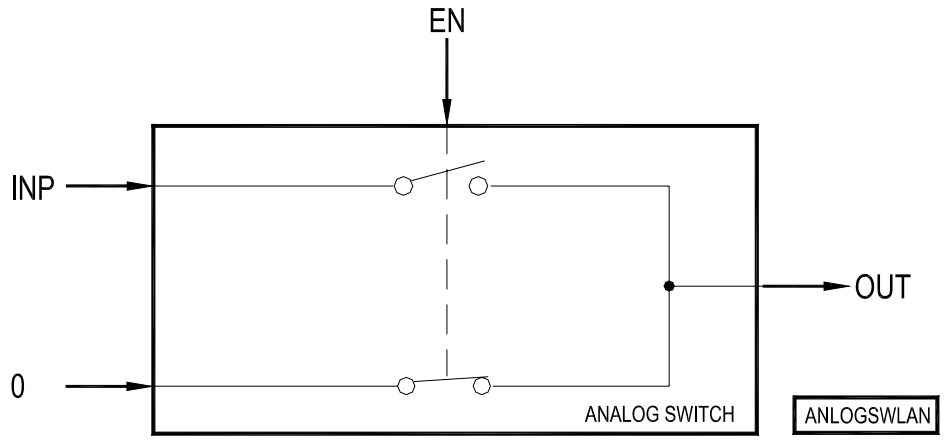


FIGURE 4-10. ANALOG SWITCH BLOCK

1. Inputs

   INP: Analog
   EN: Bit

2. Outputs

   OUT: Analog

3. Implementation

   If the EN bit is high, then OUT is equal to INP.
   If the EN bit is low, OUT is equal to zero.

4.11 ASM AUTO SHEET MARKER

This block sets an output bit high when the number of sheets cut equals an operator-entered setpoint (MARK STPT). The output bit also goes high when the number of sheets cut equals integer multiples of the setpoint. Therefore, every nth sheet sets the output bit high. The output bit remains high for the number of sheets entered in calibration as MARK HOLD; then the bit goes low. The internal counter resets when the MARK RESET bit is high.

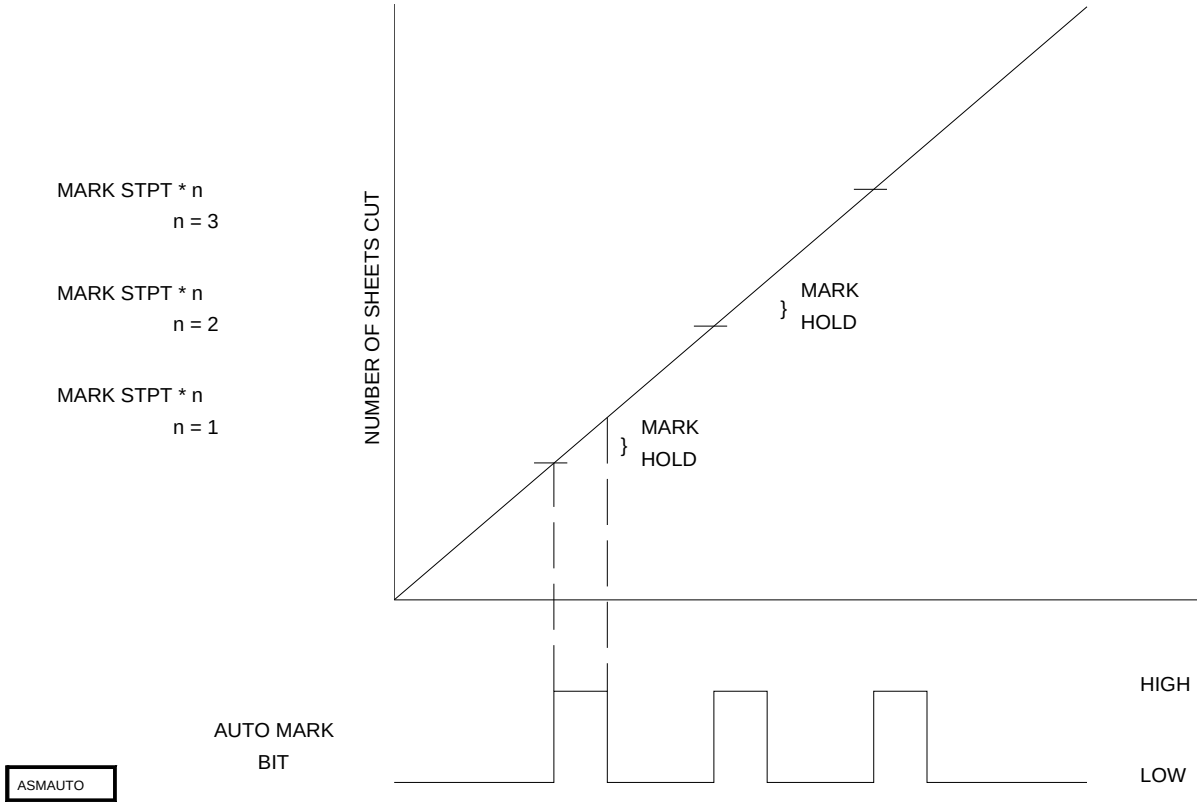


FIGURE 4-11. ASM AUTO SHEET MARKER BLOCK

1. Inputs

NUMBER SHTS: Analog
MARK STPT: Analog
MARK HOLD: Analog
MARK RESET: Bit

2. Outputs

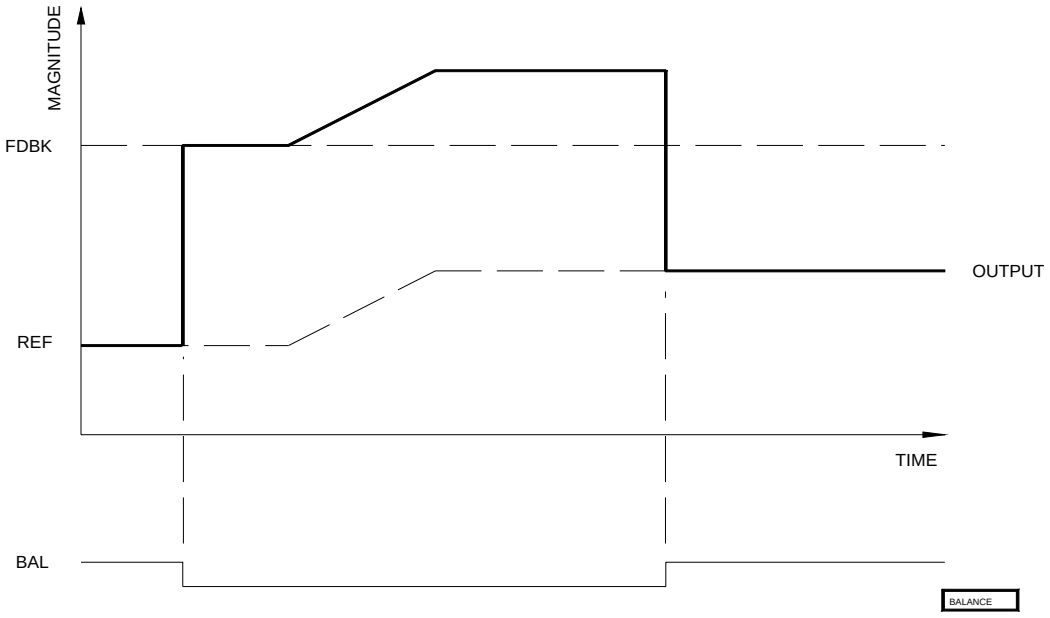
   AUTO MARK: Bit

3. Implementation

   If NUMBER SHTS = MARK STPT, then AUTO MARK goes high. If NUMBER SHTS = MARK STPT + MARK HOLD, then AUTO MARK goes low and n = n + 1. If MARK RESET bit is high, then n = 1.

4.12 BALANCE

The Balance block switches from one control scheme to another without changing the current operating reference.



FIG

URE 4-12. BALANCE BLOCK

1. Inputs

REF:	Analog
FDBK:	Analog
BAL:	Bit
RET:	Bit

2. Outputs

OUT:	Analog
DIF:	Analog

3. Implementation

On a falling edge of the BAL input, DIF will be sampled as REF-FDBK. "DIF" will not change until the next falling edge of the BAL input.

When the BAL input is low, OUT = REF - DIF.

When the BAL input is high, OUT = REF.

Non-retentive Block

On powerup of the ADDvantage-32, "err"=0.

Retentive Block

On powerup of the ADDvantage-32, DIF will be initialized under the following conditions:

If RET = 0, DIF = 0.

If RET = 1, DIF is set to its last value. DIF must also be configured to a retentive point (Y***:RET SETPT*) to be updated automatically on powerup.

4.13 BIT CONVERT

The Bit Convert block converts 4 input bits into decimal numbers. The output changes only when the ENABLE bit is high. This block can also be used as the X_IN value of a Table block enabling 16 separate setpoints.

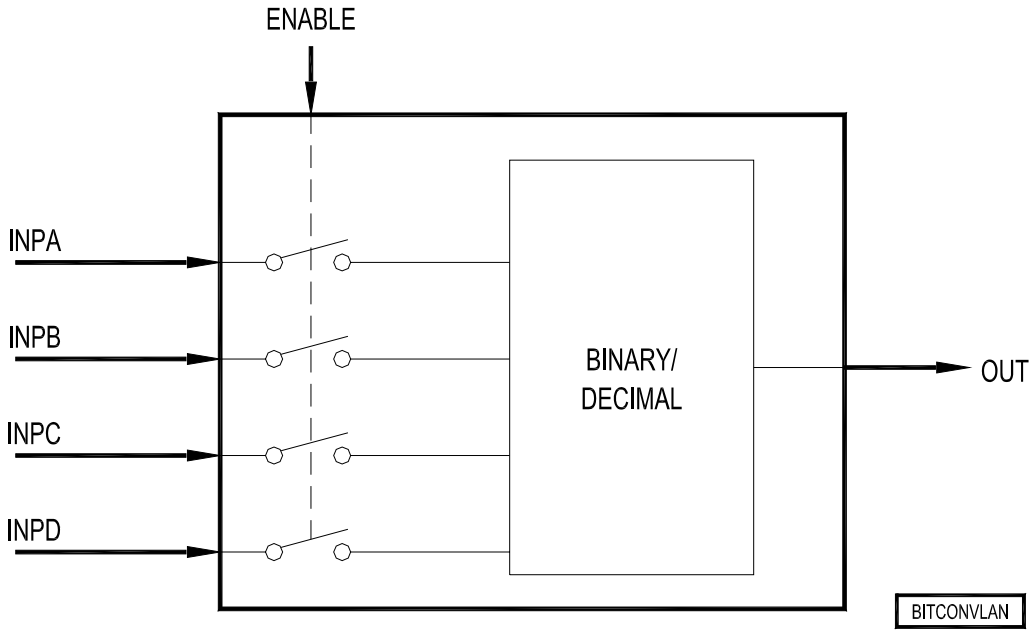


FIGURE 4-13. BIT CONVERT BLOCK

1. Inputs

   INPA: Bit
   INPB: Bit
   INPC: Bit
   INPD: Bit
   ENABLE: Bit

2. Outputs

   OUT: Analog

3. Implementation

   OUT latches only when the ENABLE input is low. OUT defaults to 0 on powerup.

   When the ENABLE bit is high, refer to the following table to determine the value of OUT.

TABLE 4-13. Bit Convert Block OUT Values

INPA	INPB	INPC	INPD	OUT
0	0	0	0	0
1	0	0	0	1
0	1	0	0	2
1	1	0	0	3
0	0	1	0	4
1	0	1	0	5
0	1	1	0	6
1	1	1	0	7
0	0	0	1	8
1	0	0	1	9
0	1	0	1	10
1	1	0	1	11
0	0	1	1	12
1	0	1	1	13
0	1	1	1	14
1	1	1	1	15

4.14 BIT INVERT

A Bit Invert block is used to provide an output bit which is always the opposite state of the blocks input bit.

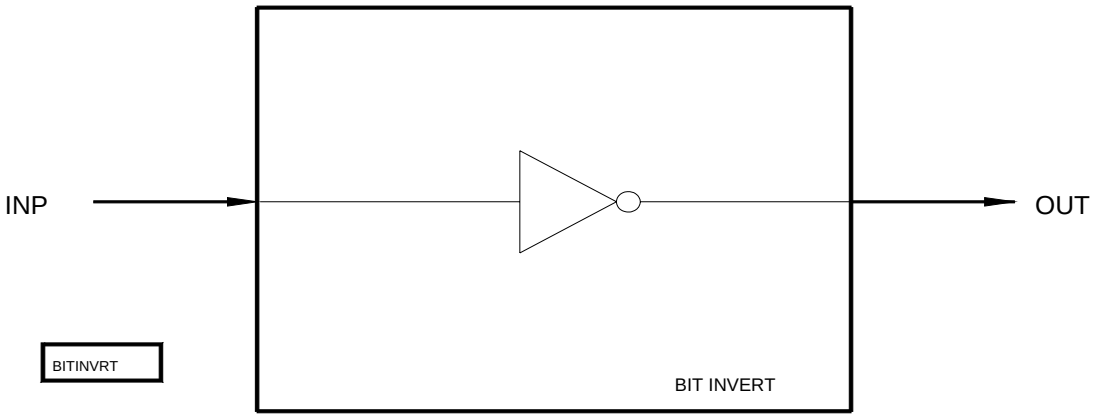


FIGURE 4-14. BIT INVERT BLOCK

1. Inputs

   INP: Bit

2. Outputs

   OUT: Bit

3. Implementation

   If INP bit is high, OUT bit is set low.

   If INP bit is low, OUT bit is set high.

4.15 BIT SELECT

This block is used to select one of two different bit signal paths.

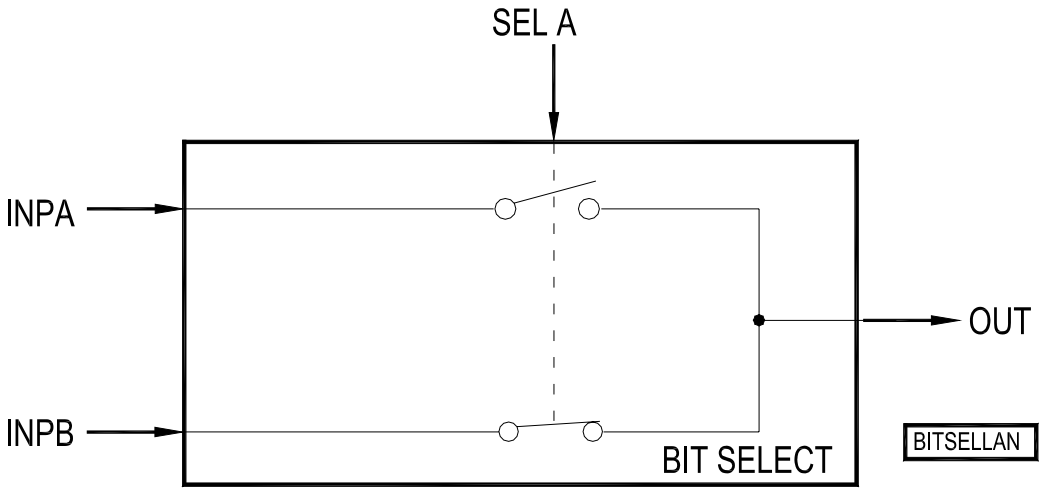


FIGURE 4-15. BIT SELECT BLOCK

1. Inputs

   INPA: Bit
   INPB: Bit
   SEL A: Bit

2. Outputs

   OUT: Analog

3. Implementation

   If SEL A bit is high (set at 1), OUT is equal to INPA.
   If SEL A bit is low (set at 0), OUT is equal to INPB.

4.16 BUMPLESS SWITCH

This block is used to provide a smooth "BUMPLESS" transition when switching control, reference, or feedback between two analog signals.

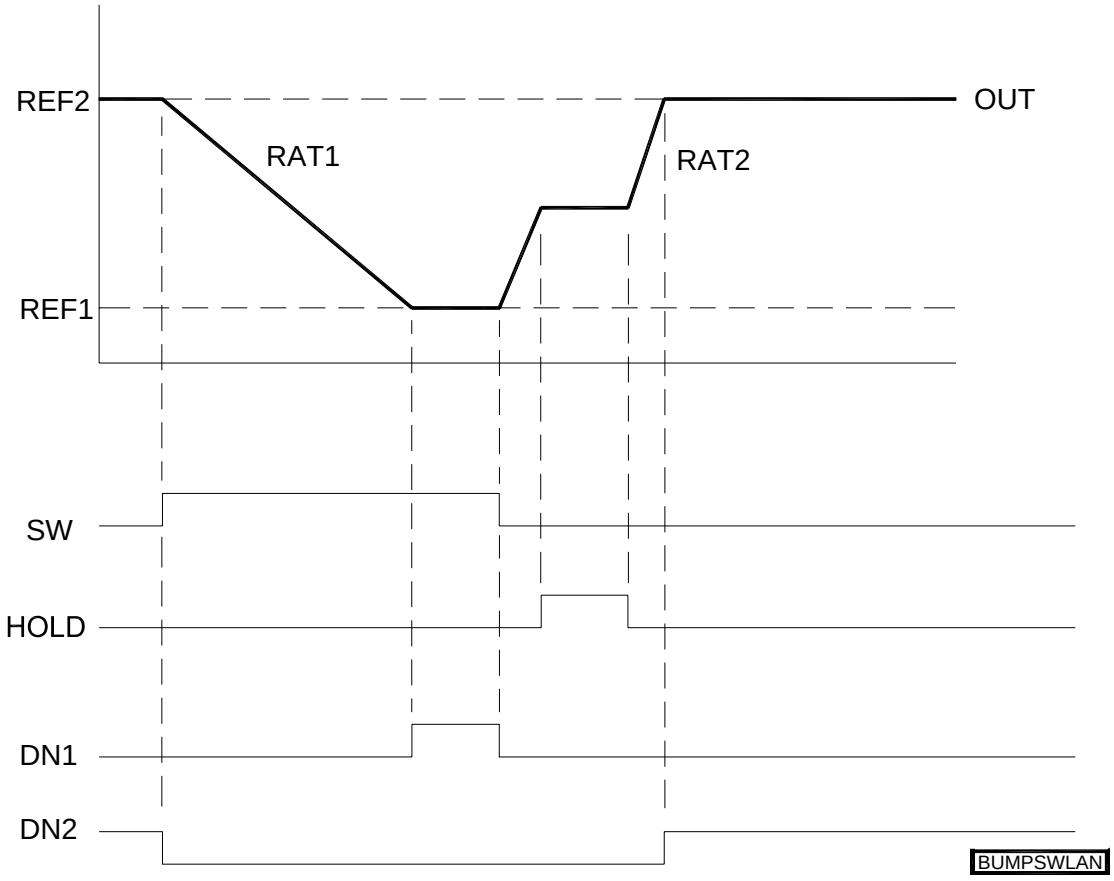


FIGURE 4-16. BUMPLESS SWITCH BLOCK

1. Inputs

REF1:	Analog
REF2:	Analog
RAT1:	Analog
RAT2:	Analog
SW:	Bit
HOLD:	Bit

2. Outputs

OUT:	Analog
DN1:	Bit
DN2:	Bit

3. Implementation

The RAT1 input is the rate in units/second that the transfer takes place when switching from REF2 to REF1. RAT2 is the rate used when switching from REF1 to REF2. If the rate input = 0, the transfer is performed without ramping.

When SW goes high, the OUT ramps from REF2 to REF1 until OUT equals REF1, or when REF1 - REF2 polarity switches from the starting polarity. When this condition occurs, the DN1 bit goes high and the OUT follows REF1 without ramping.

When SW goes low, the OUT ramps from REF1 to REF2 in the same manner as previously described. When OUT equals REF2, the DN2 bit goes high and the OUT follows REF2 without ramping.

When the HOLD bit goes high, the OUT freezes at the current value. When deactivated, the OUT ramps using the associated rate value to the input value selected by SW. If the appropriate DONE BIT was set, it will be cleared at the removal of the hold bit until the ramping is complete.

4.17 CDS COUNTS DURING STOP

This block calculates the number of pulses that will be counted during a controlled stop. It is used in turret applications to determine when the drive run should be removed so that the turret will stop at the appropriate index position.

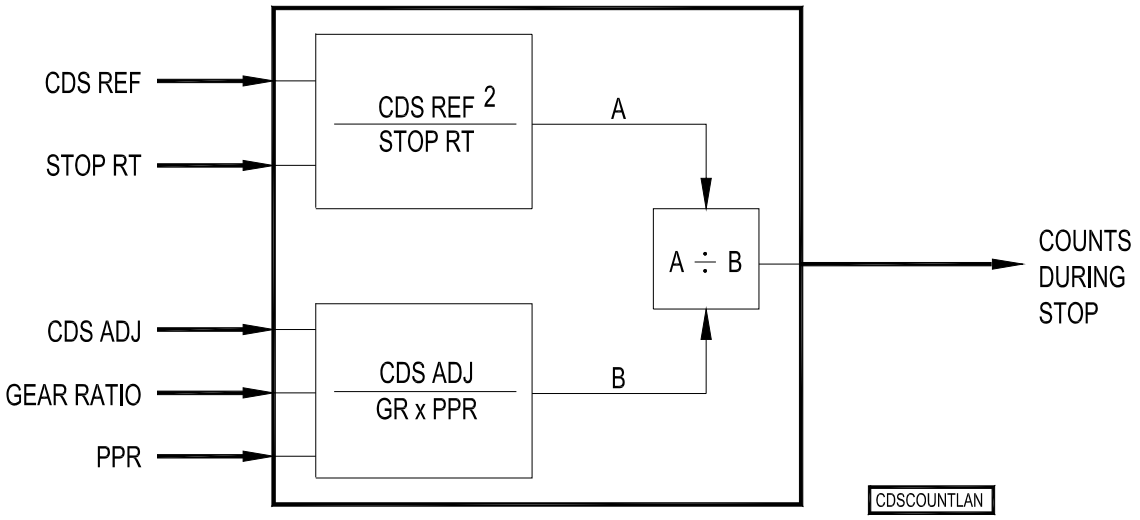


FIGURE 4-17. CDS COUNTS DURING STOP BLOCK

1. Inputs

CDS REF:	Analog
STOP RT:	Analog
CDS ADJ:	Analog
GEAR RATIO:	Analog
PPR:	Analog

2. Outputs

COUNT STOP:	Analog

3. Implementation

$$CDS = \frac{\frac{(CDS\ REF)^2}{STOPPING\ RT}}{\frac{CDS\ ADJ}{GR \cdot PPR}} = \frac{(CDS\ REF)^2 \cdot GR \cdot PPR}{STOPPING\ RT \cdot CDS\ ADJ}$$

Where:

CDS REF = REFERENCE IN RPM
STOPPING RT = RATE IN RPM/SEC
CDS ADJ = 2 × 60 SEC/MIN
GEAR RATIO = NO UNITS
PPR = PULSES/REV

4.18 CLAMPING

A Clamping block is used to restrict an analog signal to a value between user selectable high (MAXL) and low (MINL) limits.

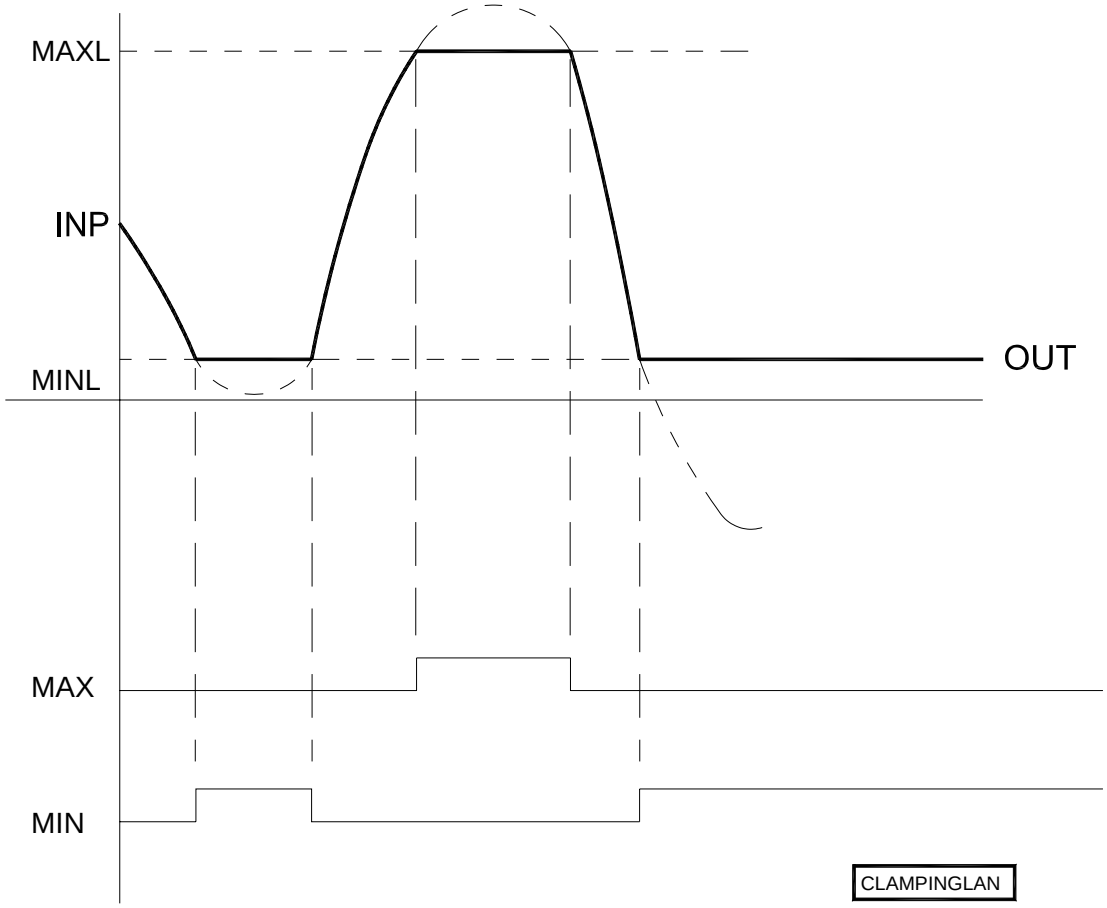


FIGURE 4-18. CLAMPING BLOCK

1. Inputs

INP:	Analog
MAXL:	Analog
MINL:	Analog

2. Outputs

OUT:	Analog
MAX:	Bit
MIN:	Bit

3. Implementation

If INP > MAXL, then
OUT = MAXL
MAX = high, MIN = low

If INP < MINL, then
OUT = MINL
MIN = high, MAX = low

If MINL < INP < MAXL, then
OUT = INP
MAX = low
MIN = low

4.19 COM LOSS

The Com Loss block is used as a communication watchdog.

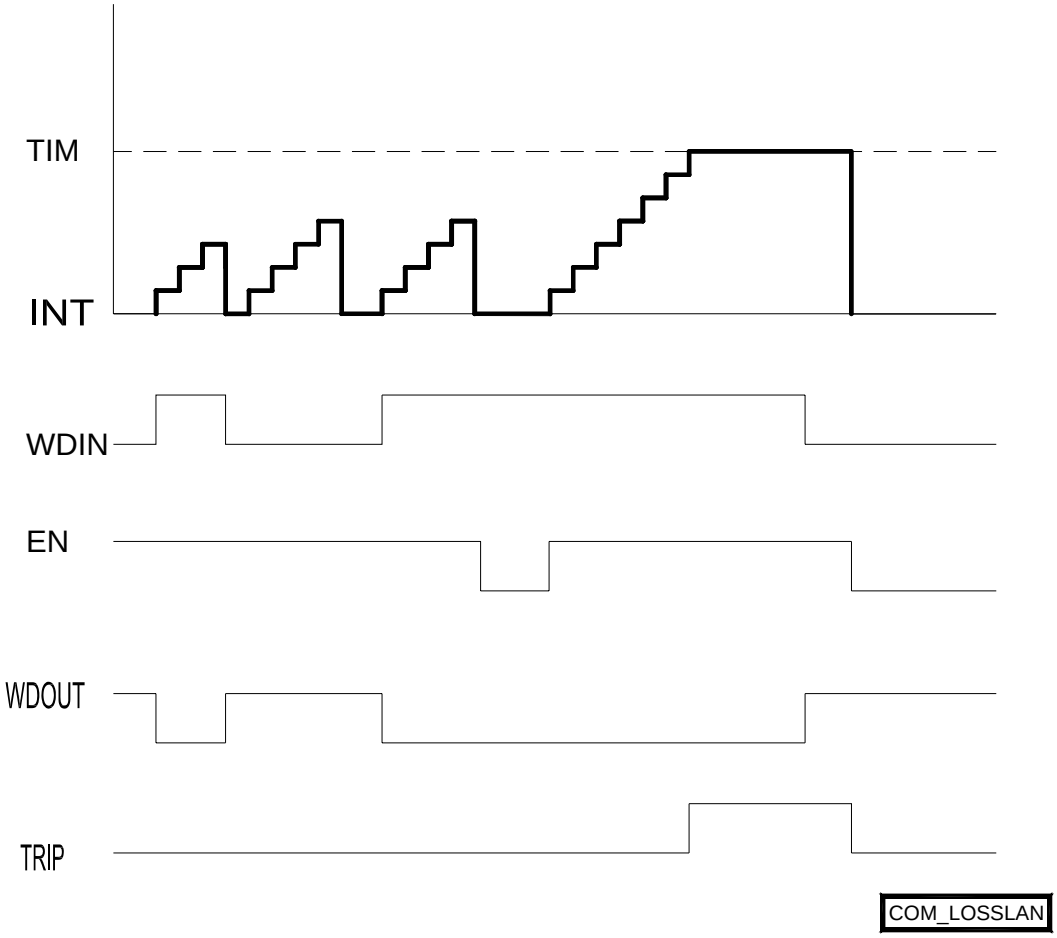


FIGURE 4-19. COM LOSS

1. Inputs

   TIM: Bit
   EN: Bit
   WDIN: Bit

2. Outputs

   WDOUT: Bit
   TRIP: Bit

4.20 COM WD

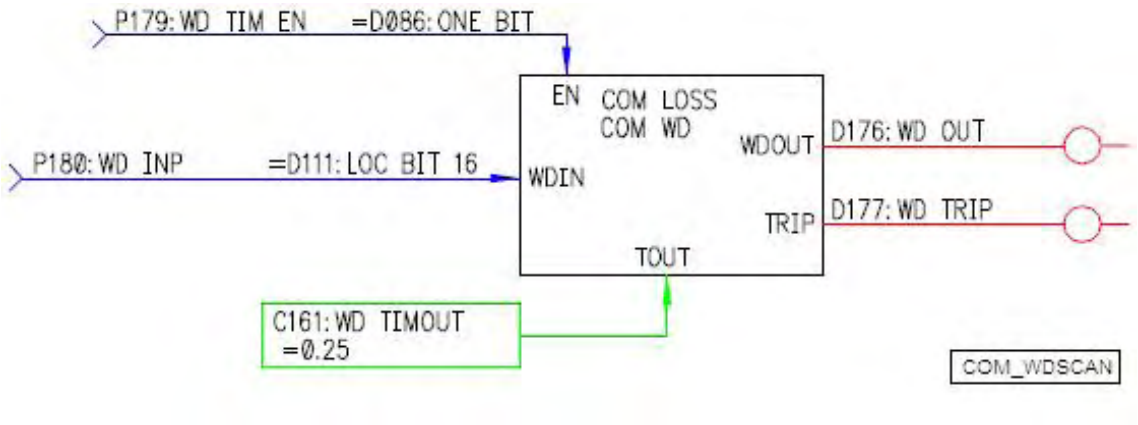


FIGURE 4-20. COM WD BLOCK

1. Inputs

   EN: Bit
   WD IP: Bit
   TOUT: Analog

2. Outputs

   WDOUT Bit
   TRIP Bit

3. Implementation

   When the EN input of the WD COM block is a logic level high (One Bit), the block monitors the WDIN input and will set the TRIP output if the input fails to toggle (transition from low to high or transition from high to low) within the time period (seconds) defined by the TOUT input. The WD OUT bit equates to the bit inverted state of the WDIN input. The TRIP output will be reset to a logic level low (Zero Bit) when the EN input is set to a logic level low signal or if the WDIN begins to toggle at a rate greater than the TOUT time period.

4.21 COMPARATOR

Use this block to set an output bit when the input is greater than a setpoint. The HYS input sets up hysteresis to debounce the output bit.

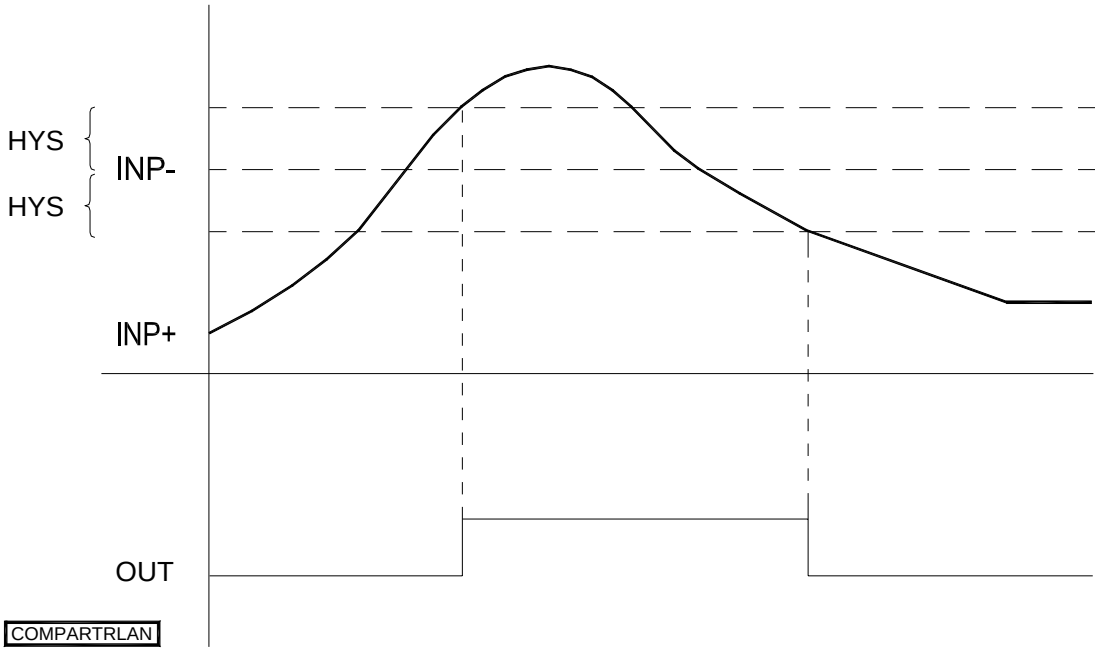


FIGURE 4-21. COMPARATOR BLOCK

1. Inputs

   INP+: Analog
   INP-: Analog
   HYS : Analog

2. Outputs

   OUT: Bit

3. Implementation

   If INP+ increases so: INP+ - HYS > INP- then the OUT bit will go high.

   If INP+ decreases so: INP+ + HYS ≤ INP- then the OUT bit will go low.

4.22 COPY

The Copy block takes the analog value at the input and copies the value to the output.

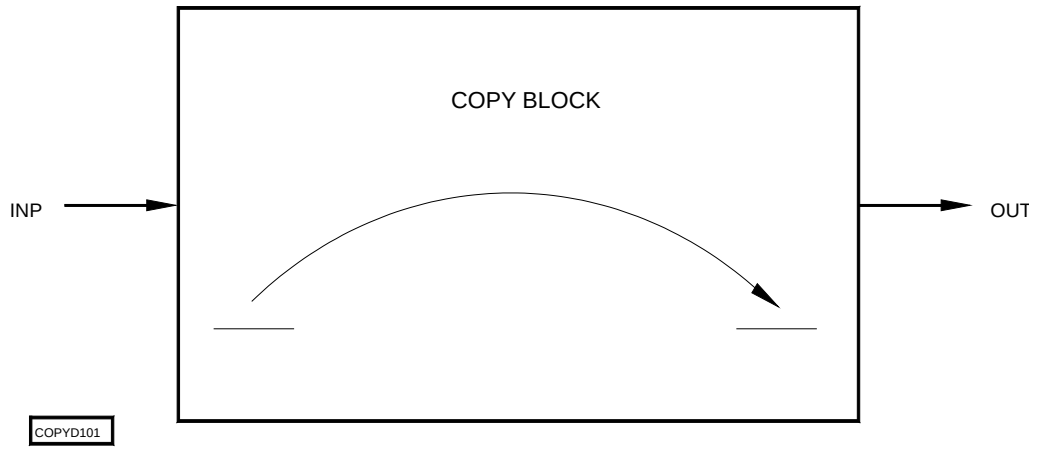


FIGURE 4-22. COPY BLOCK

1. Inputs

   INP: Analog

2. Outputs

   OUT: Analog

3. Implementation

   OUT = INP

4.23 CURRENT LIMIT

This block provides the user with a feature for selecting an automatic taper back of the drive current limits by two events or using fixed current limits.

The first event, if enabled by P***:BYPASS I2R = ZERO BIT (DEFAULT), prevents the drive from faulting on a MOTOR (II) T fault. The current limit block accomplishes this by lowering the current limit when A***:IIR INTEGR is $\geq$ to 25 counts. The MOTOR (II) T fault indicates a motor thermal overload condition where the motor armature current was above 100% for more than the time specified by X***:Thermal TC.

The motor will be allowed to operate at 150% armature current for 60 seconds if the Thermal TC is set to 60.0 (DEFAULT). The motor could operate up to 200% armature current, but the MOTOR (II) T fault would occur much sooner than 60 seconds. The user can increase the Thermal TC value up to a maximum of 3,000 seconds.

The drive will fault on a BRDGE (II) T fault when drive armature current is 150% of the drive D.C. MAX CONT nameplate current for 60 seconds. This is an Underwriters Laboratories (UL) mandated fault where the magnitude and time cannot be altered by users.

When actual motor armature current rises above 110% motor nameplate, IIR INTEGR will count up from zero when current is rising and count down when current falls below 110%. The IIR INTEGR is used to calculate motor temperature. The rate at which IIR INTEGR counts is determined by the magnitude of armature current above 100% and the thermal TC value.

The current limit taper feature can be disabled by setting P***:BYPASS I2R = ONE BIT. In this nontaper mode, IIR INTEGR will continue counting when motor armature current is above 110% until it reaches the maximum count of 100 and faults the drive on a MOTOR (II) T fault.

The second event provides automatic taper back of the drive current limit to prevent commutation faults due to overcurrenting the armature above base speed. The current limit should be set below 150% at maximum motor speed to prevent flashovers.

The user can configure the motor current limits using parameters P***:MAX I LIMIT and P***:MIN I LIMIT. These parameters are defaulted to C***:POS CUR LMT = 100 and C***:NEG CUR LMT = 100 respectively. The user can enter different current limits by entering a new value for the "C" parameter corresponding to the POS CUR LMT or NEG CUR LMT. The value is entered as a percentage of the motor nameplate armature current. It is assumed the user has correctly scaled the motor armature current to the drive D.C. MAX CONT nameplate current by setting X***:MOTOR IARM.

Regardless of the POS I LIMIT or NEG I LIMIT values for the motor current limits, actual armature current cannot exceed 200% of the drive D.C. MAX CONT nameplate current.

1. Inputs

POS I:	Analog	0 - 300
NEG I:	Analog	0 - 300
START PER:	Analog	100 - 200
END PER:	Analog	0 - 200
I2R TRIP:	Analog	0 - 100
BY I2R:	Bit	
SFDBK:	Analog	
START PER S:	Analog	0 - 200
END PER S:	Analog	0 - 200
BRK SPD:	Analog	
MAX SPD:	Analog	
BLOCK POS:	Bit	
BLOCK NEG:	Bit	

2. Outputs

POS LIM:	Analog	0 - 200
NEG LIM:	Analog	0 - 200

3. Implementation

The Current Limit block emulates the block diagram in Figure 4-21.

The POS I LIMIT and NEG I LIMIT outputs are usually configured to the MAXL and MINL inputs on the speed loop PI Control block.

Non-Tapered Current Limits

If non-tapered current limits are desired, the BY I2R input must be high by configuring P***:BYPASS I2R = ONE BIT. The current limit block constantly monitors the I2R Trip input (A***:IIR INTEGR) to detect when the motor armature current rises above 110%. The IIR INTEGR value will count up as long as the current is above 110%. Should the armature current fall below 110%, the IIR INTEGR will count down. IF the armature current stays above 110% long enough for IIR INTEGR to accumulate 100 counts, the drive will fault on a MOTOR (II) T fault.

The POS I LIMIT and NEG I LIMIT outputs will equal the POS I and NEG I input values respectively.

Tapered Current Limits

If tapered current limits are desired, the BY I2R input must be low by configuring P***:BYPASS I2R = ZERO BIT. The current limit block constantly monitors the I2R Trip input (A***:IIR INTEGR) to detect when the motor armature current rises above 110%. A hidden I2R Trip table is created within the current limit block to limit the POS LIM and NEG LIM outputs when the motor armature current rises above 110%. The POS LIM and NEG LIM outputs will equal the smaller value of either the table output or the POS I and NEG I inputs. The table output will equal the START PER (C***:START PER S) when IIR INTEGR < 25 counts. When IIR INTEGR = 100 counts, the table output will equal END PER (C***:END PER S) where the user should enter a value of 100% or less. When IIR INTEGR is between 25 to 100 counts, the table output is equal to the interpolation between START PER and END PER.

The POS LIM and NEG LIM outputs are then checked against another hidden SFDBK table within the block to check the current limits by the SFDBK input (A***:ABS ACT SPD).

When SFDBK < BRK SPD, the speed table output will equal START PER S. IF SFDBK > MAX SPD, the speed table output will equal END PER S where the user should enter a value of 100% or less. When SFDBK is between BRK SPD and MAX SPD, the speed table output will equal the interpolation between START PER S and END PER S. The POS LIM and NEG LIM outputs will equal the smaller value of either the table output or the POS I and NEG I inputs.

Block Positive Bridge

The Positive Bridge can be blocked by making the BLOCK POS input high by configuring P***:BLOCK POS B = ONE BIT. The POS LIM output will equal zero when BLOCK POS is set high.

If BLOCK POS is set low (zero bit), POS LIM output will equal the smallest value between POS I input, the output of the I2R Trip table or the SFDBK table.

Block Negative Bridge

The Negative Bridge can be blocked by making the BLOCK NEG input high by configuring P***:BLOCK NEG B = ONE BIT. The NEG LIM output will equal zero when BLOCK NEG is set high.

If BLOCK NEG is set low (zero bit), NEG LIM output will equal the negative of the smallest value between NEG I input, the output of the I2R Trip table or the SFDBK table.

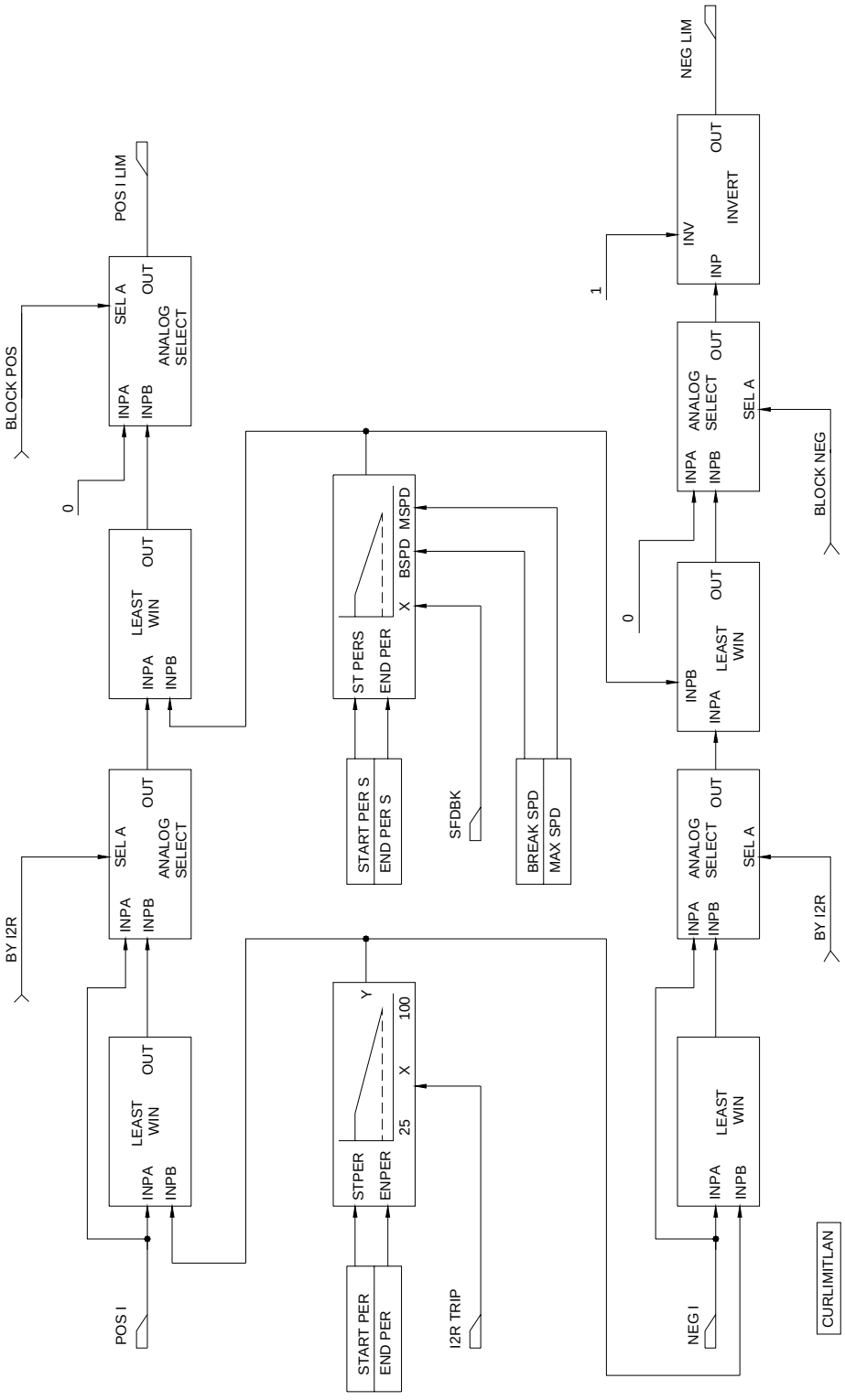


FIGURE 4-23. CURRENT LIMIT BLOCK DIAGRAM

4.24 DEADBAND

A Deadband block is used to reset an analog signal to zero when it is less than a user selected value.

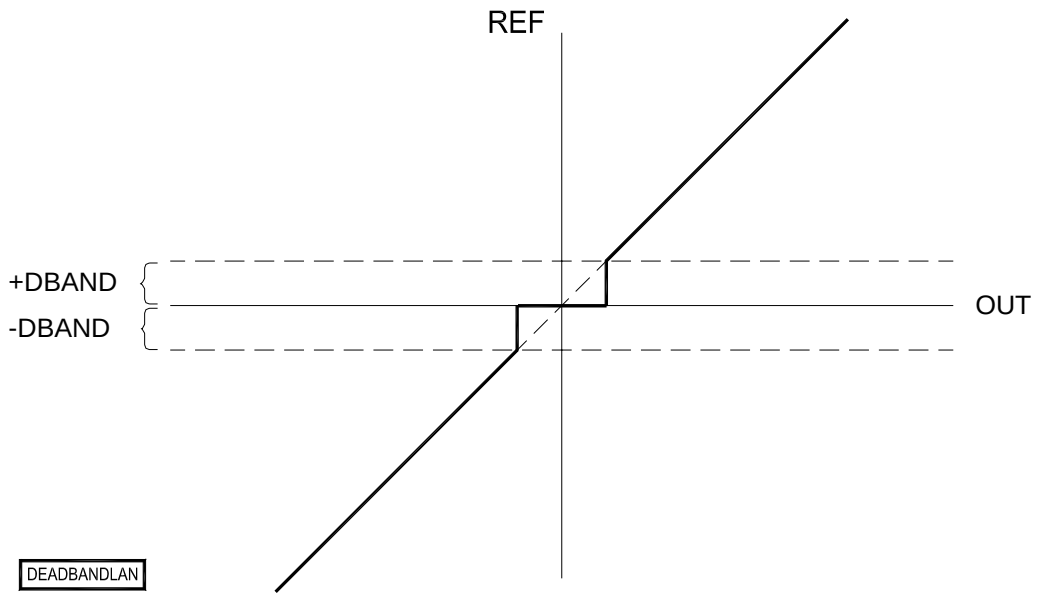


FIGURE 4-24. DEADBAND BLOCK

1. Inputs

   REF: Analog
   DBAND: Analog

2. Outputs

   OUT: Analog

3. Implementation

   If -DBAND < REF < DBAND then
   OUT = 0
   else
   OUT = REF

4.25 DEMUX

1. Inputs

BIT1: Bit
BIT2: Bit
INP: Bit

2. Outputs

OUT0 Bit
OUT1 Bit
OUT2 Bit
OUT3 Bit

3. Implementation

The output bits are set as follows:

If BIT1 = 0 and BIT2 = 0 and INP = 1 then OUT0 = 1,
else OUT0 = 0
If BIT1 = 1 and BIT2 = 0 and INP = 1 then OUT1 = 1,
else OUT1 = 0
If BIT1 = 0 and BIT2 = 1 and INP = 1 then OUT2 = 1,
else OUT2 = 0
If BIT1 = 1 and BIT2 = 1 and INP = 1 then OUT3 = 1,
else OUT3 = 0

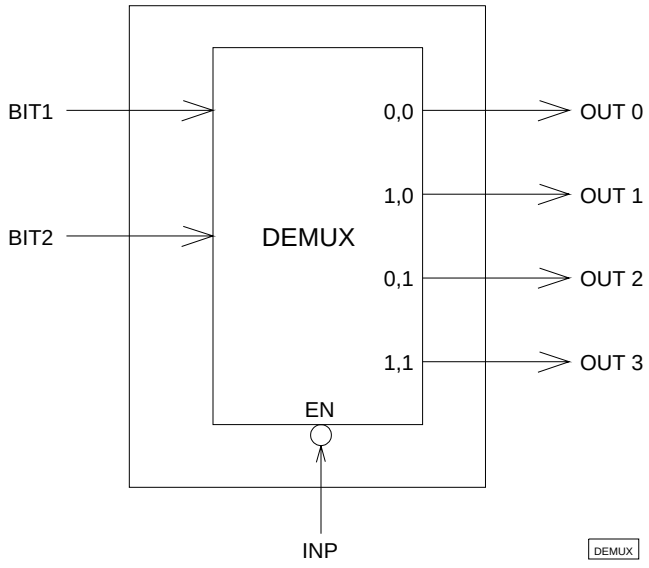


FIGURE 4-25. DEMUX BLOCK

4.26 DENSITY

This block is used to calculate the approximate density of a winding or unwinding reel of product. It calculates the density of a roll over specific diameter ranges. The range needs to be large enough to make the calculation accurately, taking into account error in the diameter measurements.

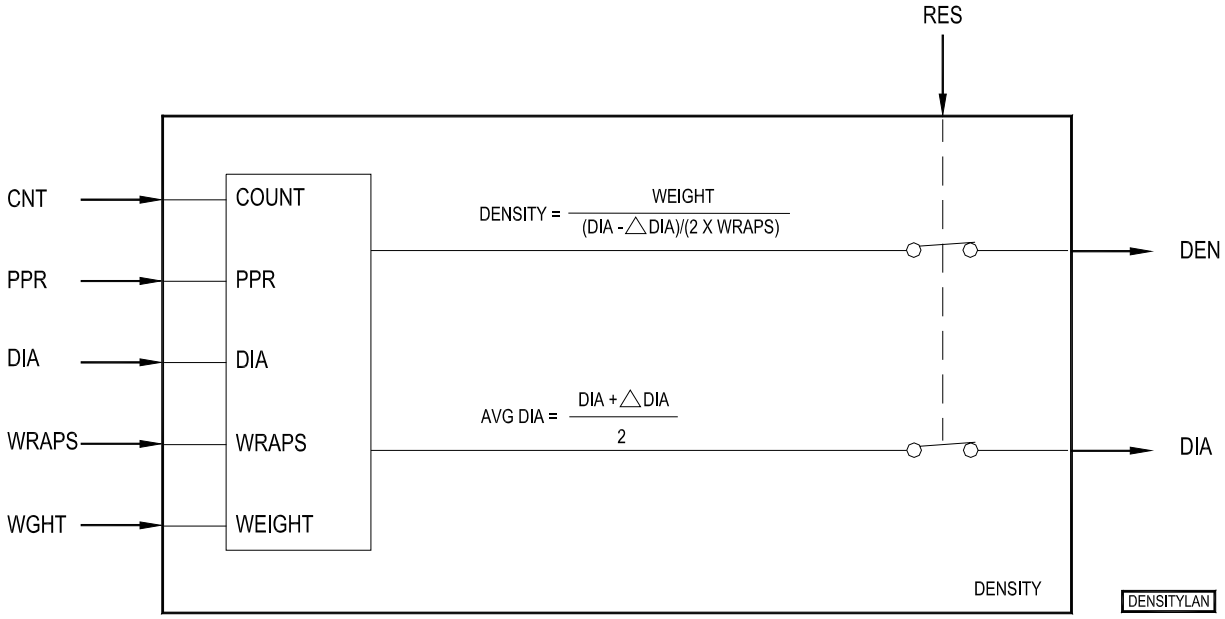


FIGURE 4-26. DENSITY BLOCK

1. Inputs

CNT:	Analog
PPR:	Analog
DIA:	Analog
WRAPS:	Analog
WGHT:	Analog
RES:	Bit

2. Outputs

DENSITY:	Analog
AVG DIA:	Analog

3. Implementation

When RES is high or on power up, the outputs = 0.
When not RES, the following occurs:

The outputs do not change until the number of winder revolutions equals WRAPS. (1 winder revolution = PPR amount of change in the CNT value) When this occurs, the following calculations are done.

DIA = DIA at start of wraps count
thick = (DIA - DIA)/(2 × WRAPS)
DEN = (WGHT/thick)
ADIA = (DIA + DIA)/2

The outputs do not change until the next time the number of winder evolutions equals WRAPS or a RES occurs.

4.27 DERIVATIVE GAIN (D/DT)

The D/DT block performs a derivative gain function. It can be used for inertia compensation.

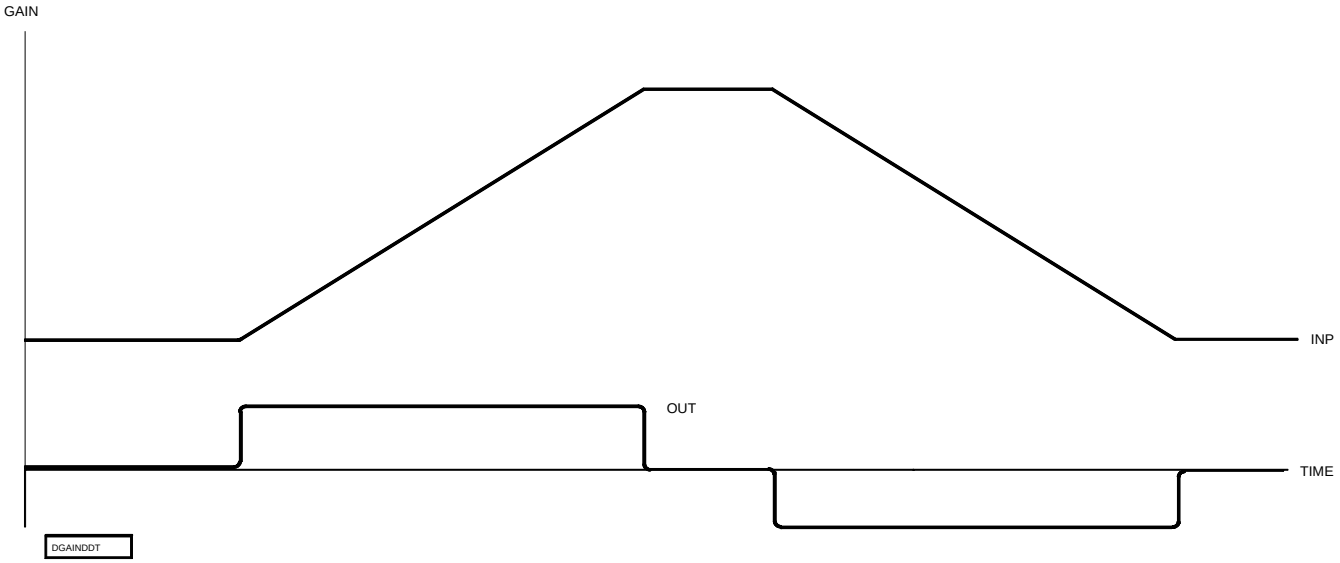


FIGURE 4-27. DERIVATIVE GAIN BLOCK

1. Inputs

   INP: Analog
   GAIN: Analog
   LP: Analog

2. Outputs

   OUT: Analog

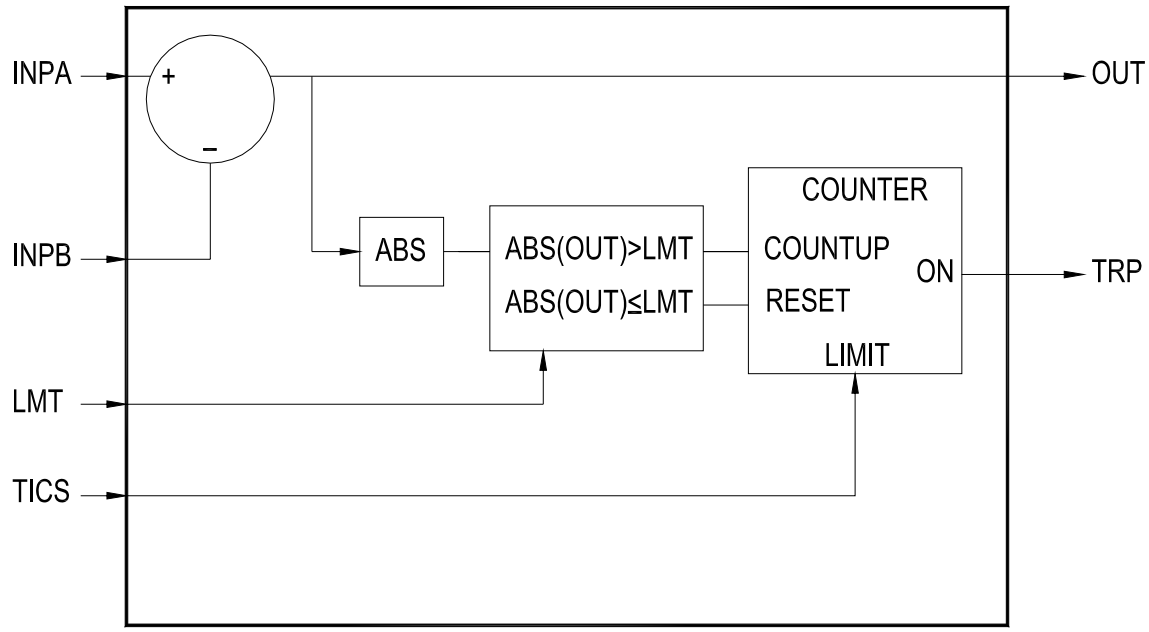
3. Implementation

   INP goes through a third order low pass filter with time constant LP in seconds. This filters amplification of high frequency bounce.

   OUT = (Rate of change of filtered INP) × GAIN

4.28 DIFF TRIP

This block is a combination differential comparator and timer. It is used to detect alarm or fault conditions.



DIFFTRIPLAN

FIGURE 4-28. DIFF TRIP BLOCK

1. Inputs

INPA:	Analog
INPB:	Analog
TICS:	Analog
LMT:	Analog

2. Outputs

OUT:	Analog
TRP:	Bit

3. Implementation

OUT is equal to INPA - INPB.

If the absolute value of OUT is greater than LMT, then the internal timer will start to count up each time the block is executed.

When it counts up to the TICS input amount, the TRP output will go high. It will stay high as long as the absolute value of OUT is greater than LMT.

As soon as the limit condition goes false, the timer is reset to zero and the TRP output goes low.

NOTE: The amount of time associated with a block execution can vary depending on the application. Look up the TICS input label in Appendix C of the manual for the timing.

4.29 DIGITAL IN

The Digital Input block enables the eight digital inputs (USER 7 through USER 14) from the optional FAX-32 board to be mapped to eight consecutive digital data table board addresses.

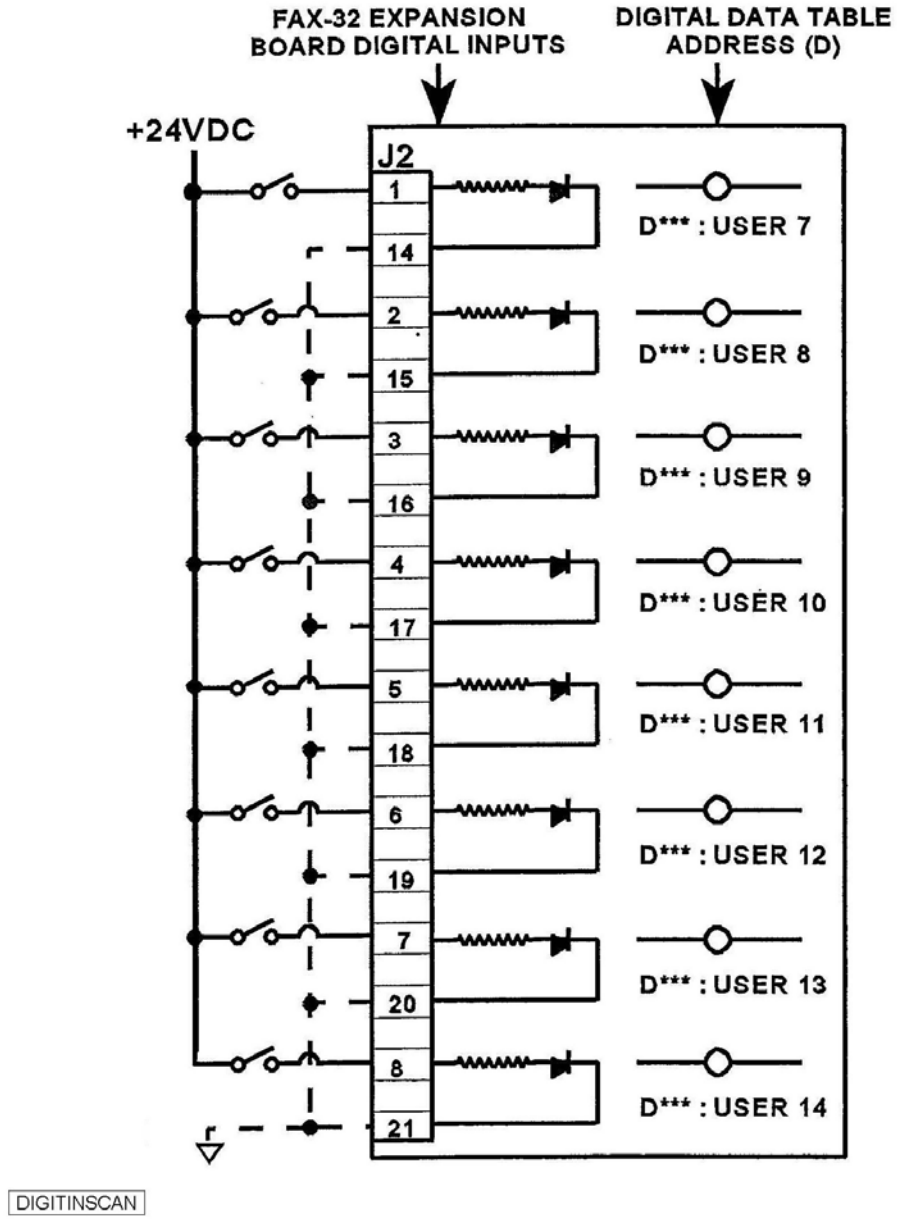


FIGURE 4-29. DIGITAL IN BLOCK

1. Inputs

   Reference FAX-32 digital inputs on J2 of board.

2. Outputs

   START: BIT

3. Implementation

   The digital input for USER 7 through USER 14 will equal ONE/ON when the input's corresponding FAX-32 J2 terminal is switched to +24 VDC.

   The digital input for USER 7 through USER 14 will equal ZERO/OFF when the input's corresponding FAX-32 J2 terminal is switched to 0 VDC.

4.30 DIGITAL OR

This block implements a 4 input digital OR.

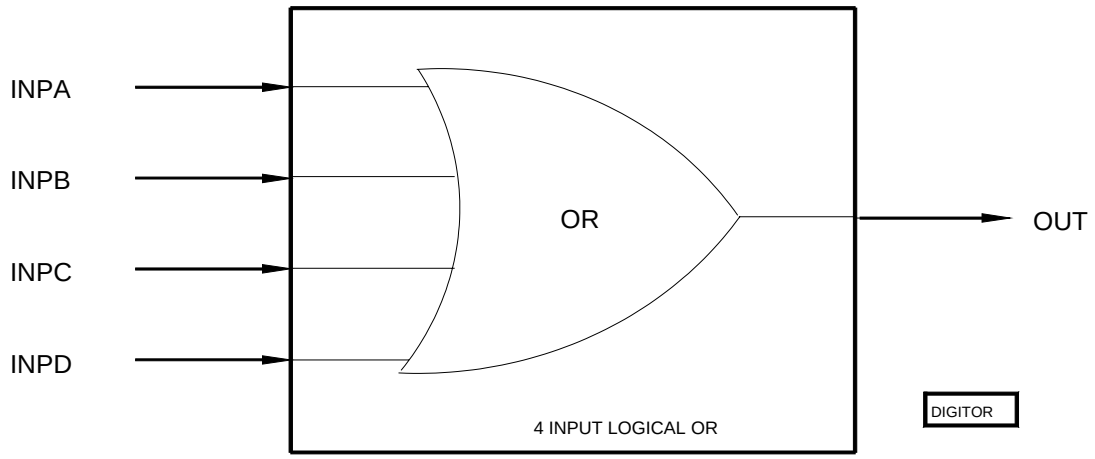


FIGURE 4-30. DIGITAL OR BLOCK

1. Inputs

   INPA: Bit
   INPB: Bit
   INPC: Bit
   INPD: Bit

2. Outputs

   OUT: Bit

3. Implementation

   OUT is set to one if either INPA, INPB, INPC, or INPD is equal to one; otherwise OUT equals zero.

4.31 DIVIDE

The Divide block is used to divide two numbers. The block checks for divide by zero to avoid a fault.

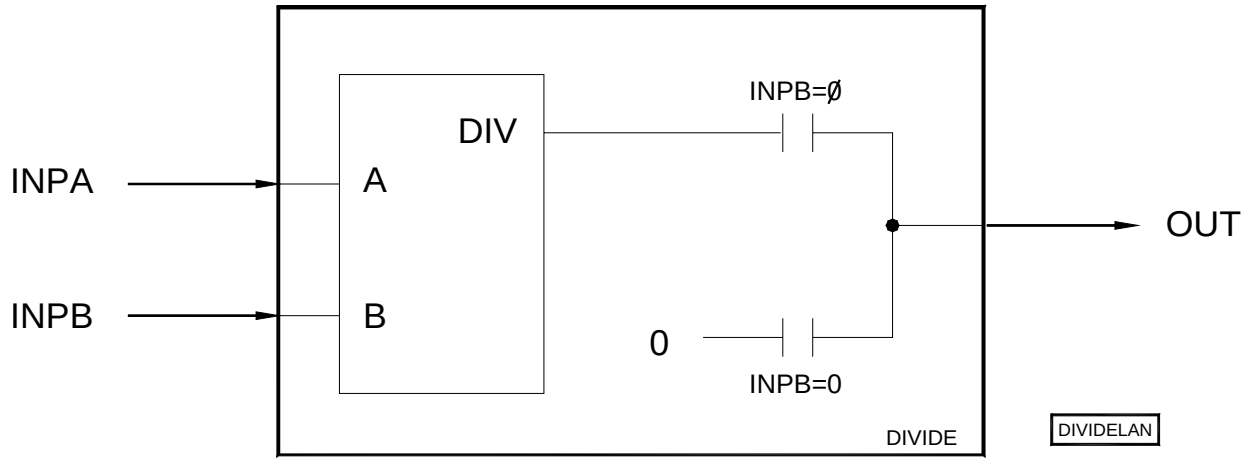


FIGURE 4-31. DIVIDE BLOCK

1. Inputs

   INPA: Analog
   INPB: Analog

2. Output

   OUT: Analog

3. Implementation

   If INPB = 0 then OUT = 0
   else

$$OUT = \frac{INPA}{INPB}$$

4.32 DROOP

The Droop block modifies the speed reference to keep the drive armature current within a particular range. This is useful for controlling a nipped roll where it is in contact with another speed controlled section, or anywhere a "SOFT" speed regulator is required.

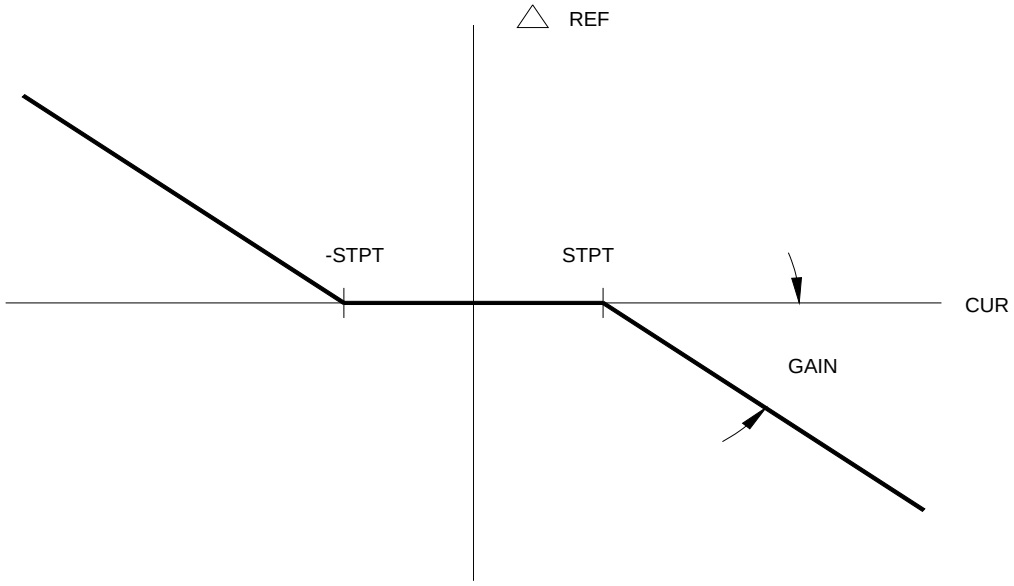


FIGURE 4-32. DROOP BLOCK

1. Inputs

CUR:	Analog
STPT:	Analog
GAIN:	Analog
REF:	Analog
ENABLE:	Bit

2. Outputs

OUT:	Analog

3. Implementation

If ENABLE bit low, OUT = REF.
If ENABLE bit high;
If (- STPT) < CUR < STPT, then OUT = REF.
If STPT < CUR, then OUT = (STPT - CUR) × GAIN + REF
If STPT < - CUR, then OUT = - (STPT + CUR) × GAIN + REF

4.33 EIP TOUT

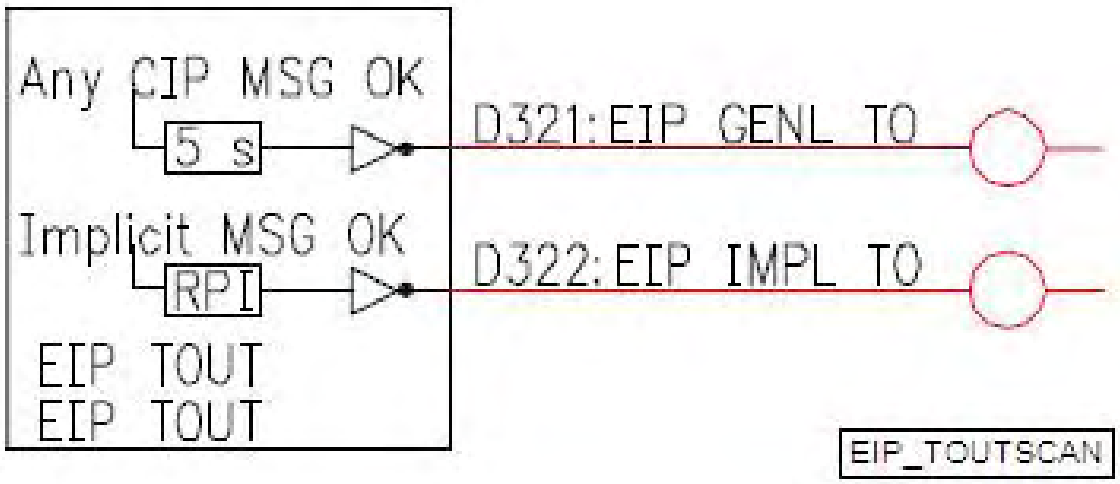


FIGURE 4-33. EIP TOUT BLOCK

1. Inputs

   NONE:

2. Outputs

EIP GENL TO	Bit
EIP IMPL TO	Bit

3. Implementation

   The EIP TOUT block resides in the ESBX module firmware, versions 682767v16 and later. The outputs of this block are written from the ESBX module directly to the drive application software, digital parameters D321:EIP GENL TO and D322:EIP IMPL TO. The functionality of the block outputs is as follows:

   D321:EIP GENL TO – This bit will remain low (Zero Bit) while there is at least one CSP message transfer occurring to the ESBX board within 5 seconds. This bit will transition from low to high, 5 seconds following the point where all CIP message transfers are terminated. The bit will reset to a logic level low when CIP message transfers resume. Refer to Appendix I for detailed information regarding the ESBX module interface to Allen-Bradley CSP communication protocol.

   D322:EIP IMPL TO - This bit will remain at a logic low while the Ethernet/IP *Implicit* message connection to the ESBX board is active and messaging is taking place at a period less than or equal to the R.P.I. (requested packet interval) time the implicit

message is scheduled to occur. This bit will transition from low to high, if the implicit communication becomes inactive for a time period greater than the R.P.I. time the implicit message was scheduled to occur. The bit will reset to a logic level low when *Implicit* messaging resumes. Refer to Appendix I for detailed information regarding the ESBX module interface to Allen-Bradley Ethernet/IP communications protocol.

4.34 ERROR

This block is used to generate an error signal such as speed error or current error.

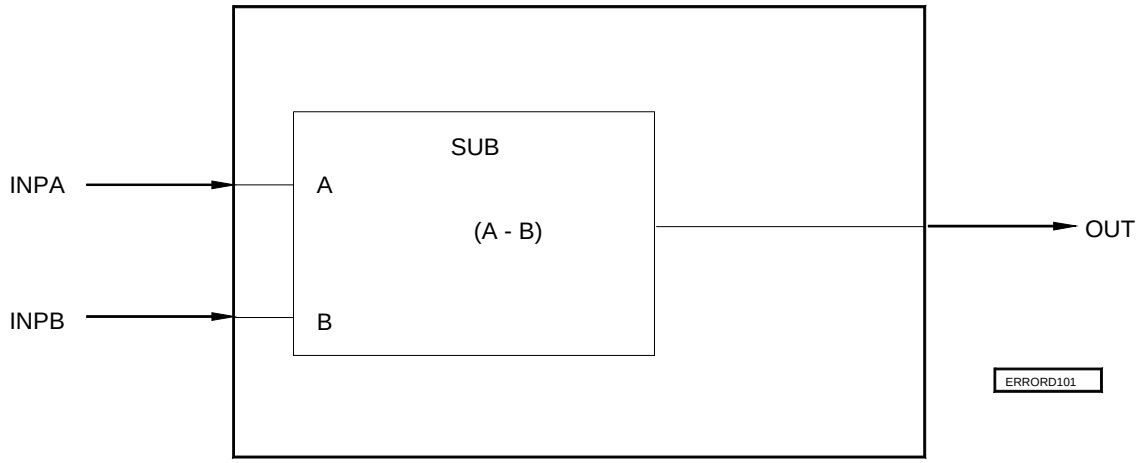


FIGURE 4-34. ERROR BLOCK

1. Inputs

   INPA: Analog
   INPB: Analog

2. Outputs

   OUT: Analog

3. Implementation

   OUT = INPA - INPB

4.35 FREQUENCY OUT

The Frequency Out block is used to output a value to the frequency output located on the optional FAX-32 board. For accuracy of the frequency output, the frequency cannot go below 200 Hz. The full range of the frequency output is from 200 to 20,200 Hz. To use the frequency as bidirectional reference, set the offset so 10,100 Hz equals zero; then scale frequency from 20,200 to 200. The INP to the frequency block is user configurable by P***:FREQUENCY.

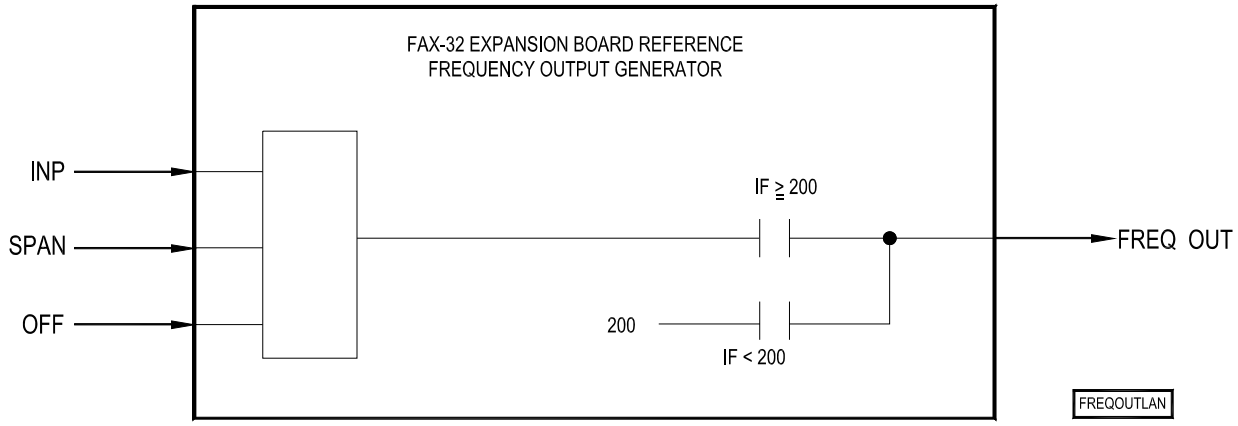


FIGURE 4-35. FREQUENCY OUT BLOCK

1. Inputs

   INP: Analog
   SPAN: Analog
   OFF: Analog

2. Outputs

   Not Applicable

3. Implementation

   The output frequency will be:

   Frequency = (INP × SPAN) + OFF

   Frequency is clamped so as not to go below 200 Hz.

4.36 GAIN

The Gain block is used to scale and offset an analog signal.

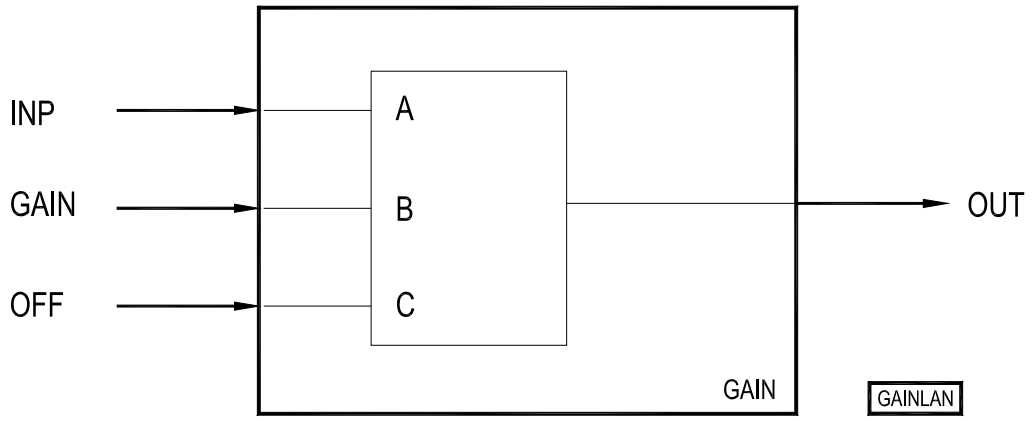


FIGURE 4-36. GAIN BLOCK

1. Inputs

   INP: Analog
   GAIN: Analog
   OFF: Analog

2. Outputs

   OUT: Analog

3. Implementation

   OUT = INP × GAIN + OFF

4.37 HI/LOW COMPARATOR

This block sets the appropriate output bits when the input goes out of limits. Corrective action can be taken at this point. Overspeed and AT ZERO SPEED detection are implemented using a HI/LOW Comparator. The user can enable overspeed protection by configuring Y***:USR FAULT 2 = OVER SPEED.

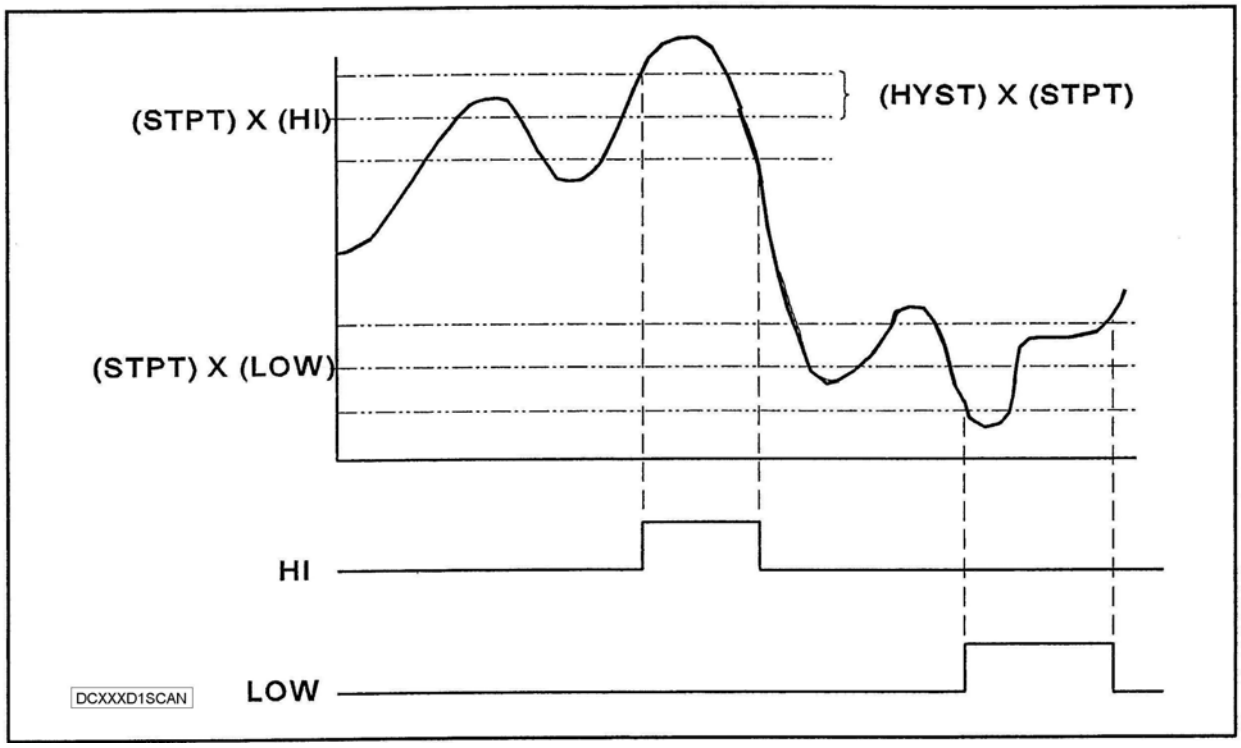


FIGURE 4-37. HI/LO COMPARATOR BLOCK

1. Inputs

HI:	Analog
LOW:	Analog
INP:	Analog
HYS:	Analog
STPT:	Analog

2. Outputs

HI:	Bit
LOW:	Bit

3. Implementation

If INP increases so:
INP - (STPT × HYS/100) ≥ (STPT × HI/100), HI bit goes high.

If INP decreases so:
INP + (STPT × HYS/100) < (STPT × HI/100), HI bit goes low.

If INP increases so:
INP - (STPT × HYS/100) ≥ (STPT × LOW/100), LOW bit goes low.

If INP decreases so:
INP + (STPT × HYS/100) < (STPT × LOW/100), LOW bit goes high.

4.38 IIT

1. Inputs

IA	Analog
STPT	Analog
IGN	Analog
FST	Analog
WST	Analog
SQU	Bit
ZER	Bit
IRES	Bit

2. Outputs

VAL	Analog
TRP	Bit
WRN	Bit

3. Implementation

The IIT block is used to determine the current overload for a motor. It can be used when more than one motor is used with the drive.

When IRES is high, the integrator is reset to zero (VAL = 0).

When IRES is low, the integrator starts integrating the error.

If ZER is low, then the internal error signal is equal to absolute value of INP minus the STPT input quantity times 0.1.

If ZER is HIGH, the error is equal to -(STPT) × 0.1.

This error is fed into the integrator if the SQU input bit is low, giving an IIT function.

If the SQU input is high, then this error is squared (the sign of signal is kept after the squaring) before it goes into the integrator.

The integrator works the same as the PI block with the IGN input as its gain value and VAL as its output.

FST is the upper limit of the integrator output and 0 is the lower limit.

When VAL goes above the WST input, the WRN output bit will go high.

When VAL goes above the FST input, the TRP output bit will go high.

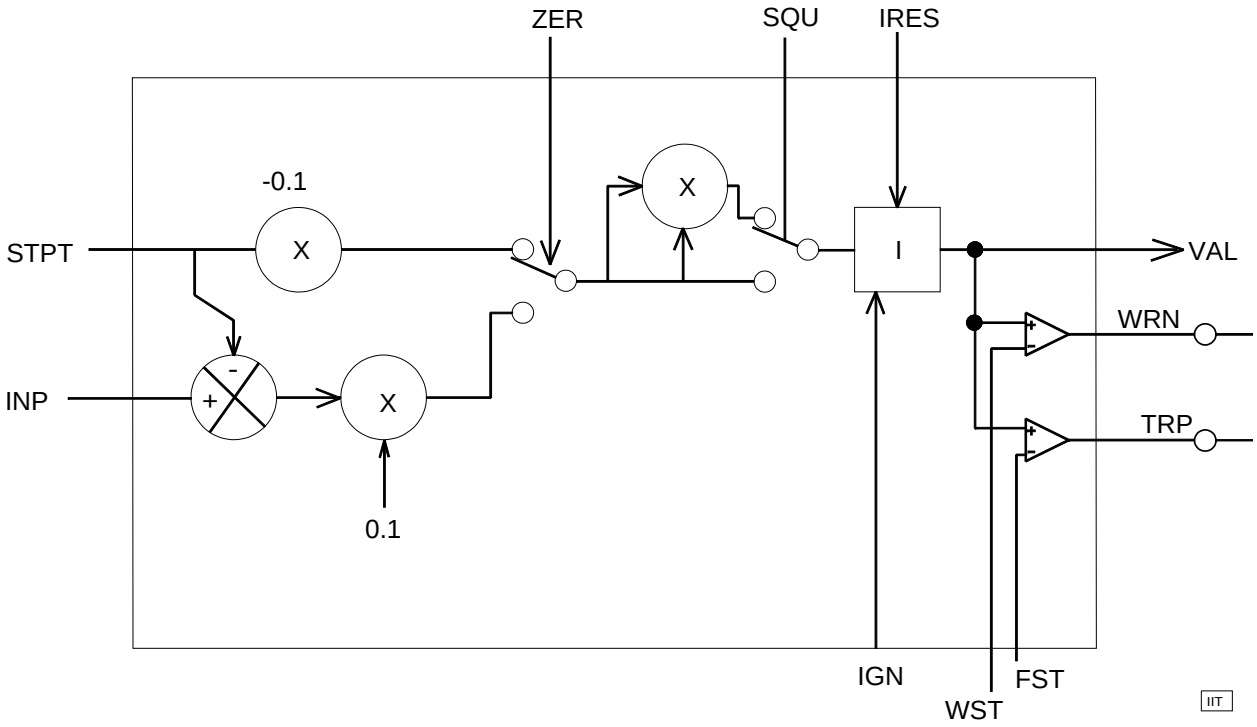


FIGURE 4-38. IIT BLOCK

4.39 LATCH

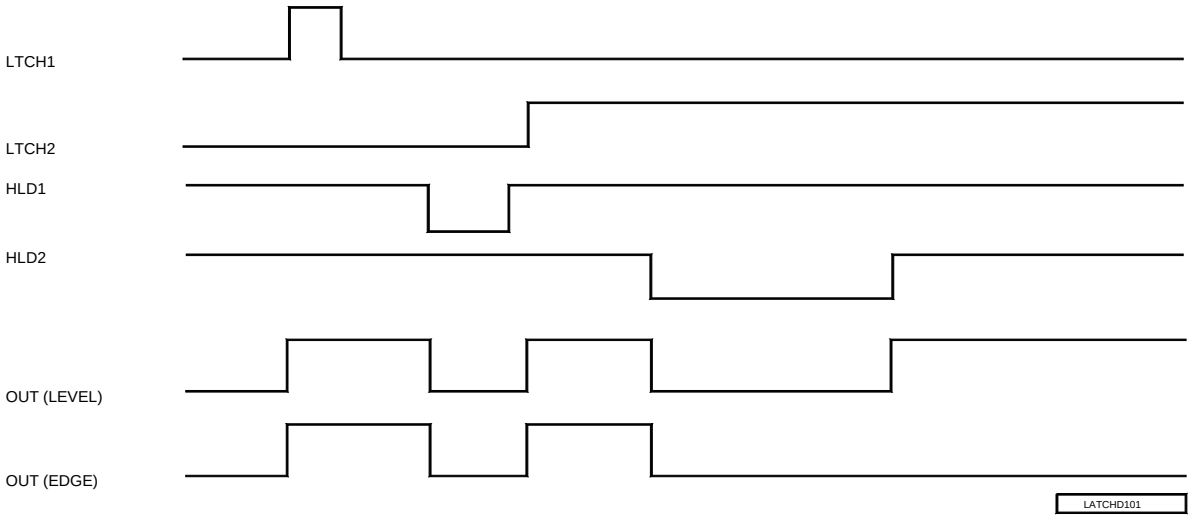


FIGURE 4-39. LATCH BLOCK

1. Inputs

   LTCH1: Bit
   LTCH2: Bit
   HLD1: Bit
   HLD2: Bit
   EDGE: Bit

2. Outputs

   OUT: Bit

3. Implementation

   If EDGE is low, then:

   If either latch (LTCH1 or LTCH2) is high and either hold (HLD1 or HLD2) is high, then OUT goes high and stays high, even if both latches later go low, as long as hold stays high.

   When either hold (HLD1 or HLD2) is low, then OUT goes low no matter what the state of the latch bits. The hold inputs have a higher priority than the latch bits. If the hold bits later go high, the condition of OUT will be determined by the condition of the latch bits at that time.

If EDGE is high, then:

If either latch (LTCH1 or LTCH2) transitions to high, the OUT goes high and stays high, even if both latches later go low.

Upon transition of the latch bit high, it does not matter what the state of the hold (HLD1 or HLD2) is in determining the state of the OUT bit, as it does in "If Edge is low."

If OUT is high, when either hold (HLD1 or HLD2) transitions to low, the OUT will go low. It is important to note that the hold must be a transition, not just the current state.

4.40 LEAD/LAG

LEAD COMPENSATION is added to a control loop to improve rise time and the damping. The disadvantage of LEAD COMPENSATION is that it adds instability to the system by increasing high frequency closed loop gain.

LAG/LEAD is added to a control loop to improve overshoot and relative stability. The disadvantage of LAG COMPENSATION is that it results in a lower rise time.

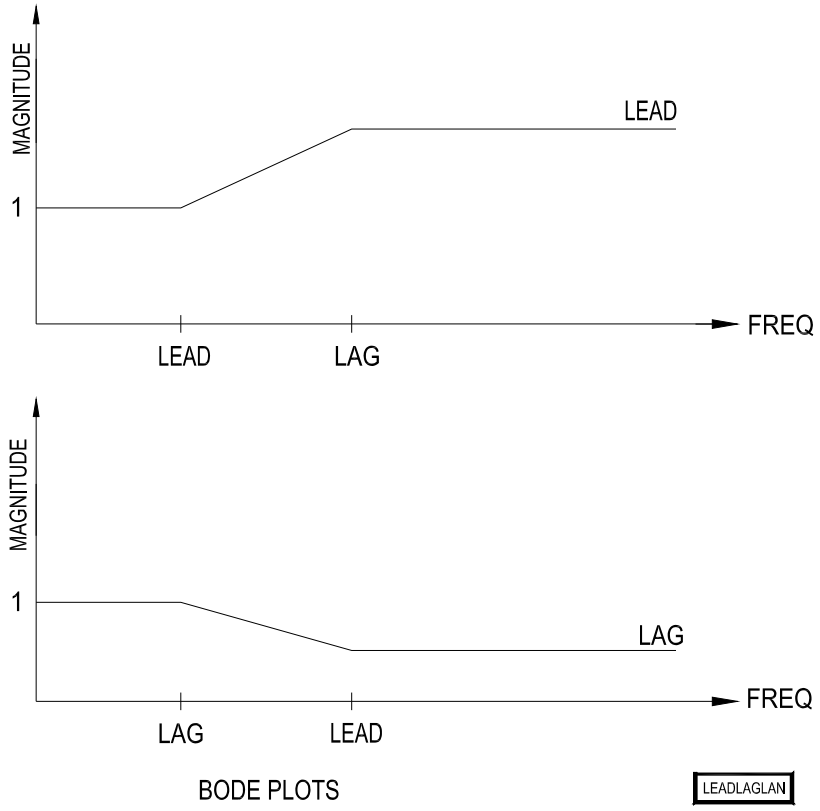


FIGURE 4-40. LEAD LAG BLOCK

1. Inputs

   INP: Analog
   LEAD: Analog
   LAG: Analog

2. Outputs

   OUT: Analog

3. Implementation

The LEAD/LAG block is implemented to simulate the following equation.

$$H(s) = LAG/LEAD \times (s + LEAD) / (s + LAG)$$

Setting the LEAD value equal to LAG gives the block a unity gain.

4.41 LEAST WIN

The block is used to select the lowest value of the inputs for a setpoint reference.

1. Inputs

INPA: Analog
INPB: Analog
INPC: Analog

2. Outputs

OUT: Analog

3. Implementation

OUT will equal the lowest input value.

e.g.

If INPA = 40
INPB = -20
INPC = -30

then OUT = -30

4.42 LOWPASS FILTER

Use to filter out high frequency noise from analog signals, such as tension feedback.

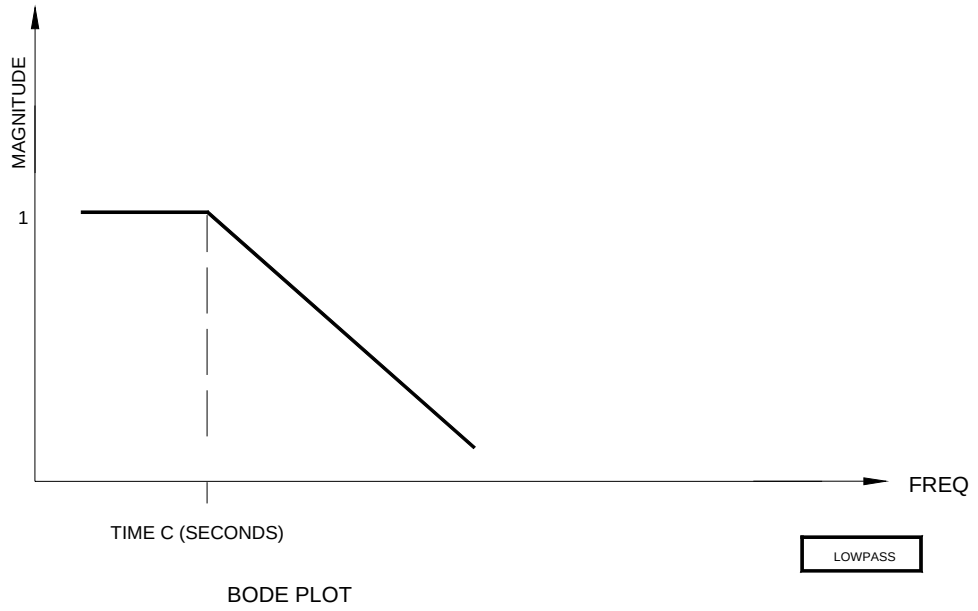


FIGURE 4-42. LOWPASS FILTER BLOCK

1. Inputs

   INP: Analog
   TC: Analog

2. Outputs

   OUT: Analog

3. Implementation

   The lowpass filter takes the INP, filters it for high frequency, then outputs it to OUT. The time constant in seconds for the filter is determined by TC. Limits of the time constant are from 0.00277 to 2.5 seconds.

4.43 MOST WIN

The block is used to select the highest value of the inputs for a setpoint reference.

1. Inputs

INPA: Analog
INPB: Analog
INPC: Analog

2. Outputs

OUT: Analog

3. Implementation

OUT will equal the highest input value.

e.g.

If INPA = 40
INPB = -20
INPC = -30

Then OUT = 40

4.44 MOV8

This block is used to move eight variables to another location, when enabled.

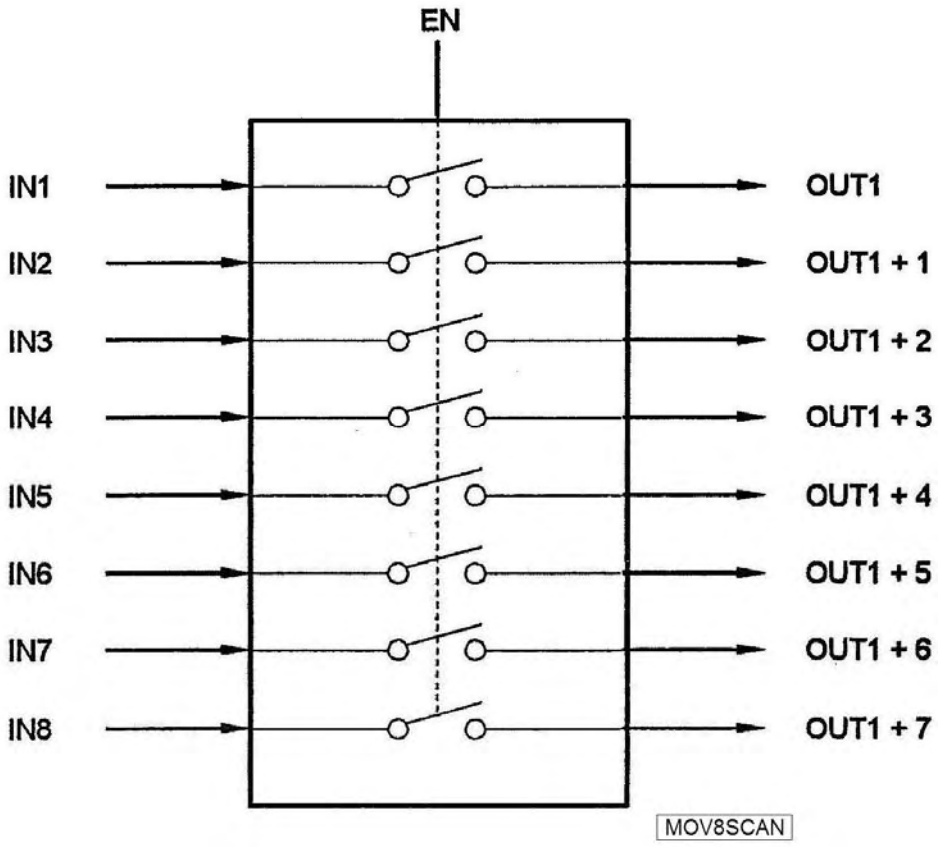


FIGURE 4-44. MOV8 BLOCK

1. Inputs

IN1:	Analog
IN2:	Analog
IN3:	Analog
IN4:	Analog
IN5:	Analog
IN6:	Analog
IN7:	Analog
IN8:	Analog
EN:	Bit

2. Outputs

OUT1: Analog

3. Implementation

If EN is low, the output does not change.

When EN is high,

OUT1 = IN1

Next location = IN2
Next location = IN3
Next location = IN4
Next location = IN5
Next location = IN6
Next location = IN7
Next location = IN8

4.45 MULTIPLY

This block is used to multiply two analog signals. For example, draw input × speed reference signal.

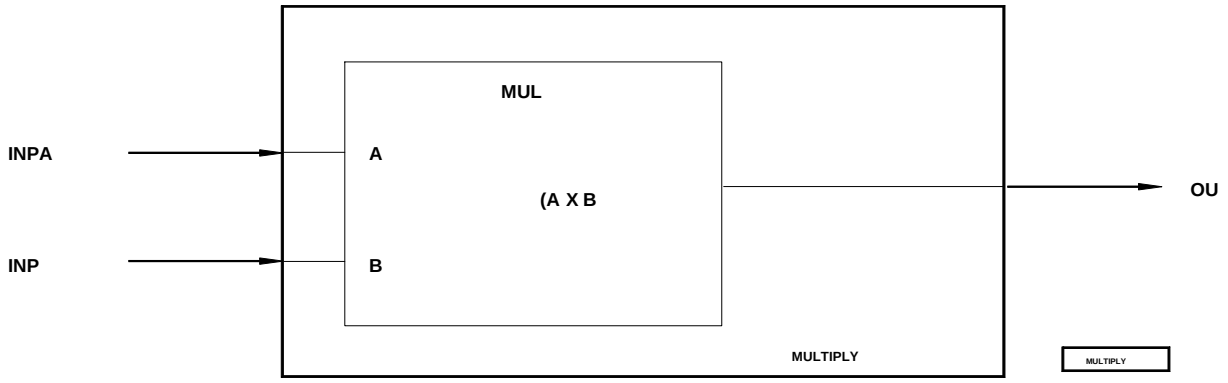


FIGURE 4-45. MULTIPLY BLOCK

1. Inputs

   INPA: Analog
   INPB: Analog

2. Outputs

   OUT: Analog

3. Implementation

   OUT = INPA × INPB

4.46 NOTCH FILTER

The Notch Filter block is a band reject filter designed to damp out machine resonance. In many cases this resonance is speed independent and occurs at approximately 5-7 Hz.

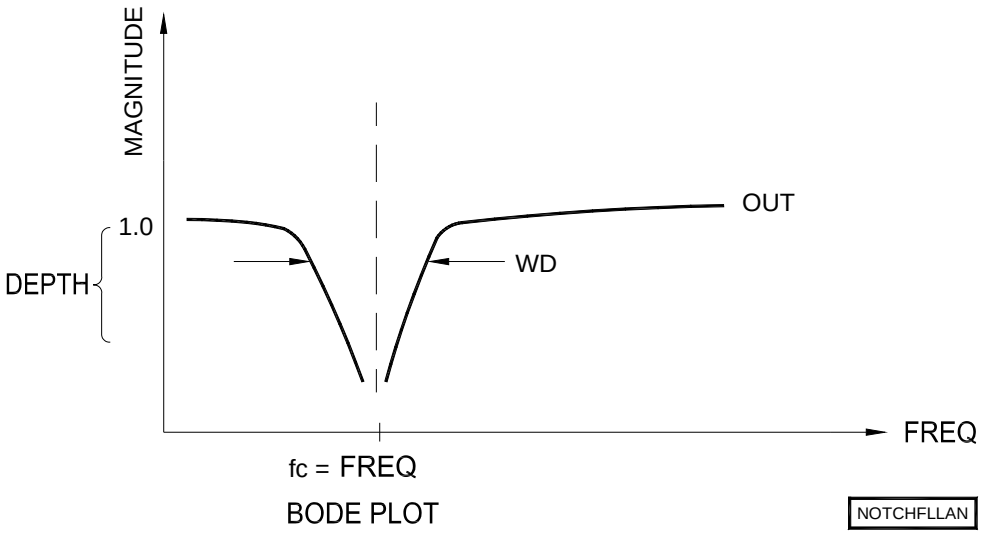


FIGURE 4-46. NOTCH FILTER BLOCK

1. Inputs

INP:	Analog
WD:	Analog
DEPTH:	Analog
FREQ:	Analog

2. Outputs

OUT:	Analog

3. Implementation

The DEPTH parameter sets the depth of the notch. It can range from 0 to 100. Entering a number less than 2 will yield a notch depth of 0. A depth of 3 will reduce the gain by 1/3 (33%) at frequency point.

The FREQ parameter is used to set the center frequency of the notch. The usable range is from 2 to 15 Hz.

The WD parameter sets the width of the notch and is a unitless quantity ranging from 0.1 to 5.0, where 0.1 is the narrowest and 5.0 is the widest.

For example, WD = 2 will reach approximately 90% of the input at 2 Hz from the frequency point.

4.47 OFF TIMER

The block is used to delay actions after the input bit goes low.

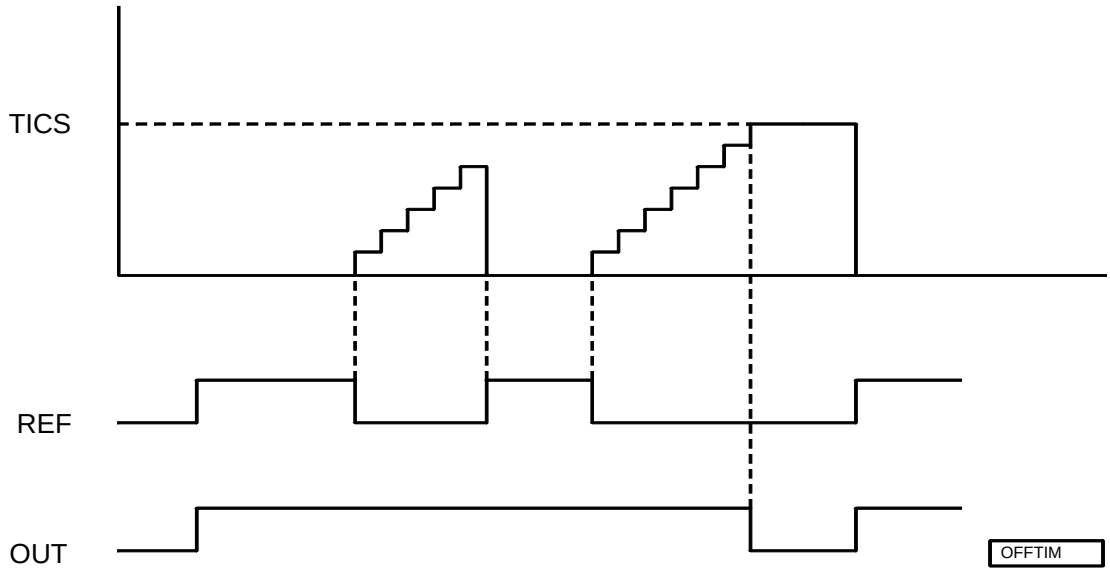


FIGURE 4-47. OFF TIMER BLOCK

1. Inputs

   TICS: Analog
   REF: Bit

2. Outputs

   OUT: Bit

3. Implementation

   If REF is high, then OUT is high.

   When REF goes low, the block waits for TICS amount of block executions before OUT goes low. If during the waiting REF goes back high, the counter is reset.

   NOTE: The amount of time associated with a block execution can vary depending on the application. Look up the TICS input label in Appendix C of the manual for the timing.

4.48 ON TIMER

1. Inputs

   TICS: Analog
   REF: Bit

2. Outputs

   OUT: Bit

3. Implementation

   If REF is low, then OUT is low.

   When REF goes high, the block waits for TICS amount of block executions before OUT goes high. If during the waiting REF goes back low, the counter is reset.

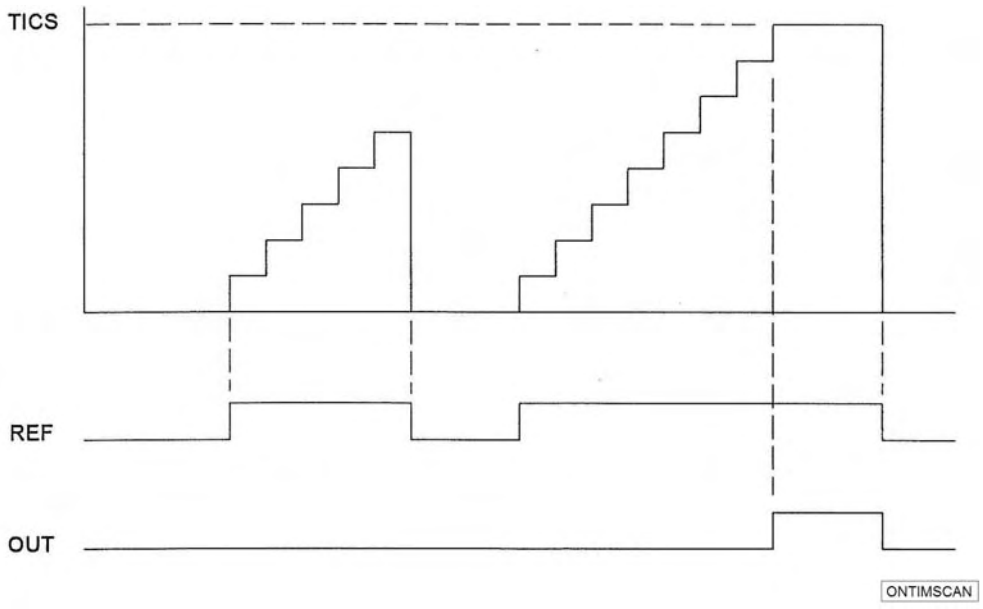


FIGURE 4-48. ON TIMER BLOCK

4.49 ONE SHOT

The block is used to initiate an action only once.

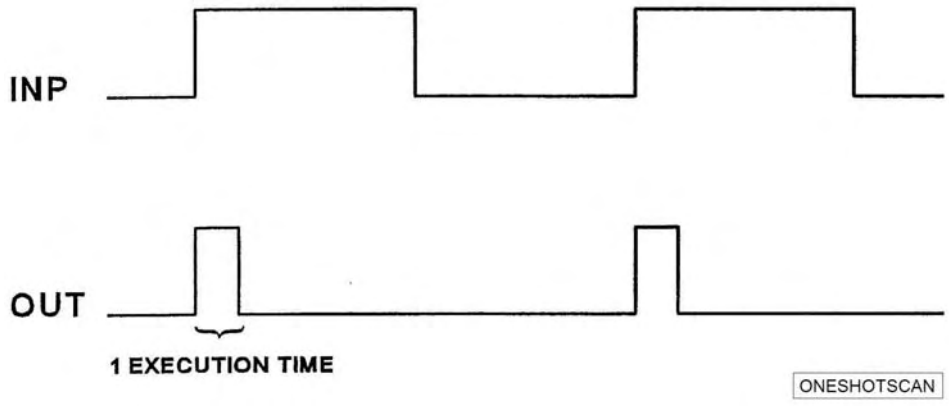


FIGURE 4-49. ONE SHOT BLOCK

1. Inputs

   INP: Bit

2. Outputs

   OUT: Bit

3. Implementation

   If INP is low, then OUT is low.

   When INP goes high, OUT will go high for only one execution time.

   NOTE: The amount of time associated with a block execution can vary depending on the application. Look up the input or output label in Appendix C of the manual for the timing.

4.50 PEAK DETECT

1. Inputs

INP	Analog
HLD	Bit
RES	Bit
POS	Bit

2. Outputs

OUT	Analog

3. Implementation

As long as the RES and HLD bits are low
OUT = the greater of Old OUT or INP (Greatest wins) if POS = 1
OUT = the least of Old OUT or INP (Greatest wins) if POS = 0

If the HLD bit is high, then OUT is held and INP is ignored.

If RES bit is high, OUT = 0.

RES and HLD are both level triggered bits.

RES has higher priority than HLD.

On powerup OUT = 0.

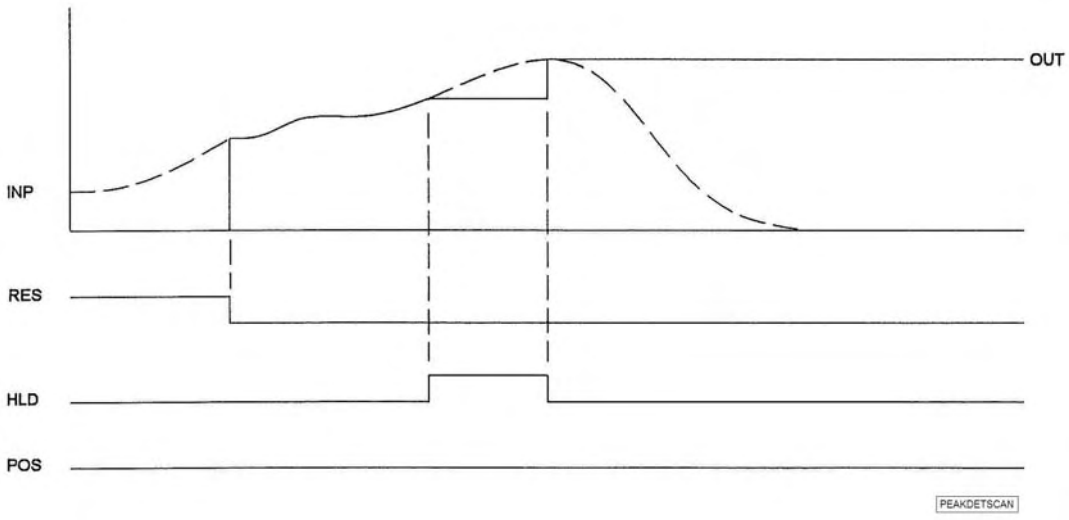


FIGURE 4-50. PEAK DETECT BLOCK

4.51 PERCENT DIFFERENCE

The Percent Difference block indicates that the percent difference between the reference and feedback is greater than the setpoint. It can be used to detect a web loss condition on a center driven winder application.

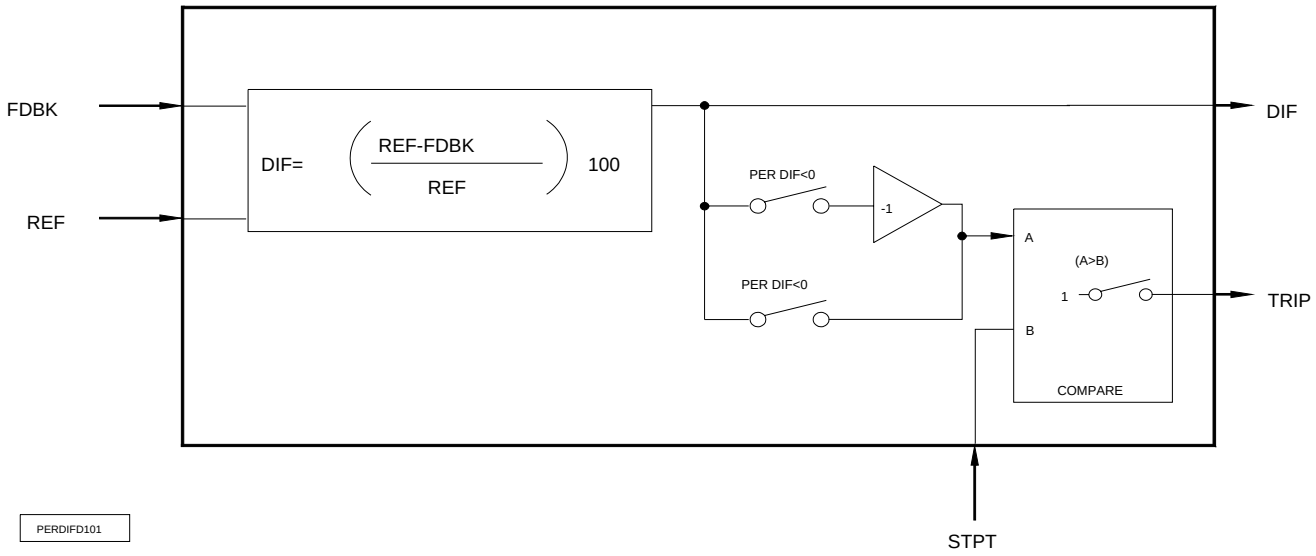


FIGURE 4-51. PERCENT DIFFERENCE BLOCK

1. Inputs

   REF: Analog
   FDBK: Analog
   STPT: Analog

2. Outputs

   DIF: Analog
   TRIP: Bit

3. Implementation

   DIF = (1 - FDBK/REF) × 100

   TRIP = 1, when the ABSOLUTE | DIF | > STPT, else TRIP = 0

   If REF = 0, then DIF = 0

4.52 PERCENT MULTIPLY (Per Mult)

The Percent Multiply block converts percentages entered using the keypad into actual values to be used in other blocks. For example, convert percent current limit into actual current limit.

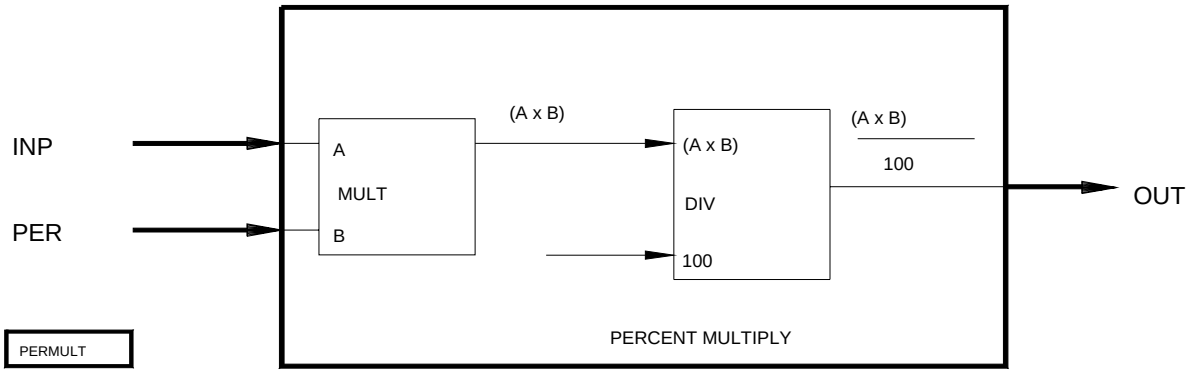


FIGURE 4-52. PERCENT MULTIPLY BLOCK

1. Inputs

   PER: Analog
   INP: Analog

2. Outputs

   OUT: Analog

3. Implementation

$$OUT = \frac{(PER \times INP)}{100}$$

4.53 PROPORTIONAL AND INTEGRAL CONTROL (PI)

The PI block performs a proportional and integral gain function on an error signal. It has inputs for maximum and minimum limits which prevent the loop from overcompensating.

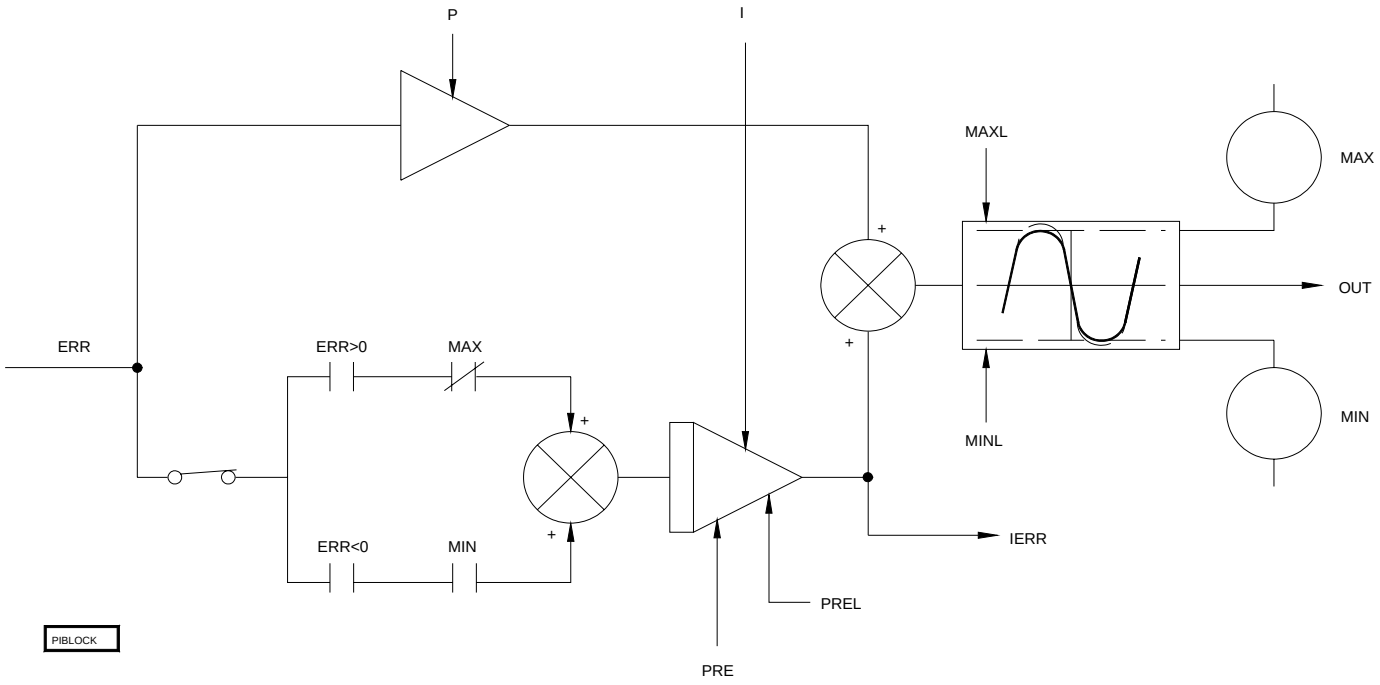


FIGURE 4-53. PI BLOCK

1.	Inputs	Data Type	Description
	ERR:	Analog	Input Signal
	P:	Analog	Proportional Gain Value
	I:	Analog	Integral Gain Value
	MAXL:	Analog	Output High Limit Value
	MINL:	Analog	Output Low Limit Value
	PREL:	Analog	Integrator Preload Value
	HOLD:	Bit	Integrator Hold Enable
	PRE:	Bit	Integrator Preload Enable

2.

Outputs	Data Type	Description
OUT:	Analog	Output Signal
IERR:	Analog	Integral Error Value
MAX:	Bit	Indication Output is Clamped at High Value
MIN:	Bit	Indication Output is Clamped at Low Value

3. Implementation

If HOLD = LOW and PRE = LOW, then:

OUT = Proportional + Integral error values of block

where:

Proportional ERROR (PERR) = ERR × P

Integral Error (IERR) =

$$\text{Old Integral Error Value} + \frac{ERR}{I\left(\frac{Sample\ Time}{1\ Second}\right)}$$

I is entered in seconds. If I is entered at less than 0.001 seconds, then I is set to zero. The sample time is used to convert to the block execution rate where typically:

Armature Current Loop =	2.77 msec
Speed Loop	= 8 msec
Tension Loop	= 16 msec

If OUT > MAXL: The following sequence of events will occur:

The MAX bit = 1 (HIGH). The Proportional Error is set so OUT will not exceed MAXL. If the Proportional Error = 0, the Integral Error will start to increase to keep OUT below the MAXL limit.

If OUT < MINL: The following sequence of events will occur:

The MIN bit = 1 (HIGH). The Proportional error component is set so OUT will not exceed MINL. If the Proportional Error = 0, the Integral Error component will start to increase to keep OUT above the MINL limit.

If HOLD = HIGH and PRE = LOW, then:

The Integral Error component is held at its present value until the HOLD input goes low. The integrator hold input is used to hold the output value during major process disturbances.

If PRE = HIGH, then:

The Integral Error component is set to the PREL value and the Proportional Error component is set to zero so that:

OUT = IERR = PREL

The IERR output is equal to the integrator component and can be used for diagnostic purposes.

4.54 PROPORTIONAL AND INTEGRAL CONTROL WITH CASCADED HOLD BITS (PI2)

The PI2 block performs a proportional and integral gain function on an error signal. It has inputs for maximum and minimum limits which prevent the loop from overcompensating as well as separate bits to hold the integrator in either the positive or negative direction only.

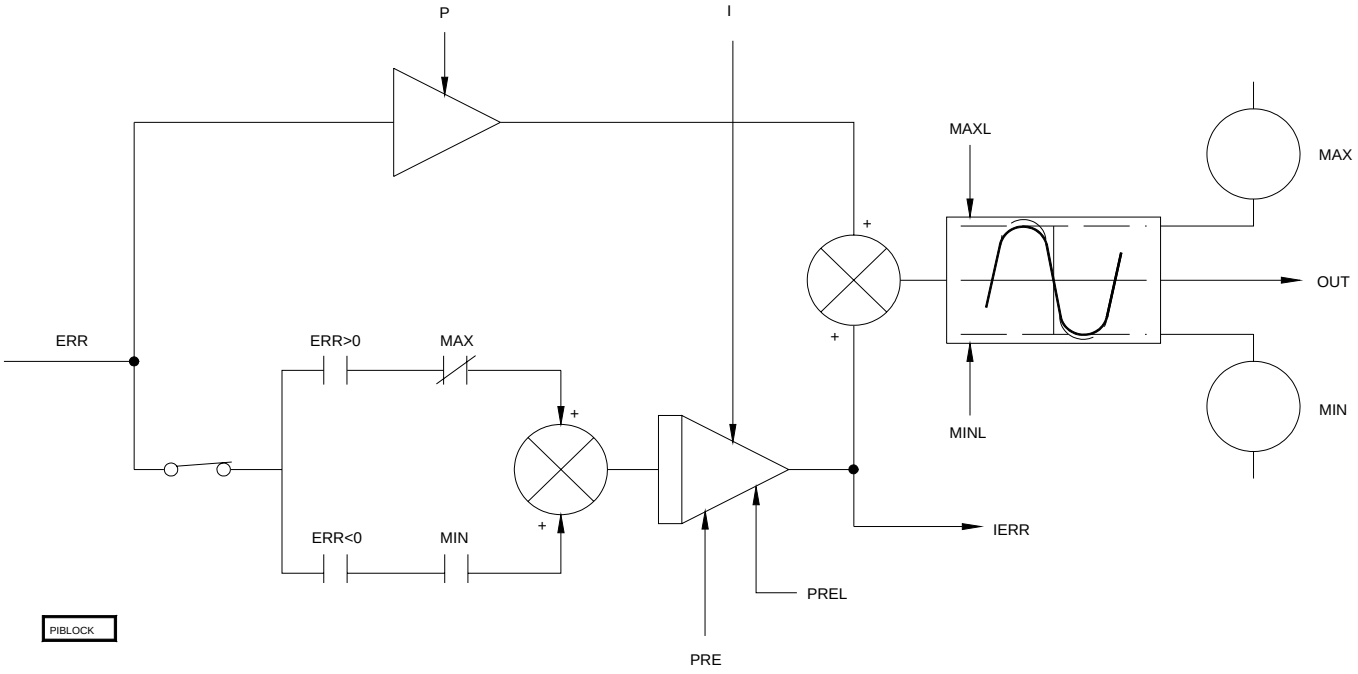


FIGURE 4-54. PI2 BLOCK

1.	Inputs	Data Type	Description
	ERR:	Analog	Input Signal
	P:	Analog	Proportional Gain Value
	I:	Analog	Integral Gain Value
	MAXL:	Analog	Output High Limit Value
	MINL:	Analog	Output Low Limit Value
	PREL:	Analog	Integrator Preload Value
	H-UP:	Bit	Integrator Positive Hold Enable
	H-DN:	Bit	Integrator Negative Hold Enable
	PRE:	Bit	Integrator Preload Enable

2. Outputs

Outputs	Data Type	Description
OUT:	Analog	Output Signal
IERR:	Analog	Integral Error Value
MAX:	Bit	Indication Output is Clamped at High Value
MIN:	Bit	Indication Output is Clamped at Low Value
CMX:	Bit	Indication Output is Clamped at High Value or is Being Held in the Positive Direction.
CMN:	Bit	Indication Output is Clamped at Low Value or is Being Held in the Negative Direction.

3. Implementation

If H-UP = LOW, H-DN = LOW, and PRE = LOW, then:

OUT = Proportional + Integral error values of block

where:

Proportional ERROR (PERR) = ERR × P

Integral Error (IERR) =

$$\text{Old Integral Error Value} + \frac{ERR}{I\left(\frac{Sample\ Time}{1\ Second}\right)}$$

I is entered in seconds. If I is entered at less than 0.001 seconds, then I is set to zero. The sample time is used to convert to the block execution rate where typically:

Armature Current Loop = 2.77 msec
Speed Loop = 8 msec
Tension Loop = 16 msec

If OUT > MAXL: The following sequence of events will occur:

The MAX bit = 1 (HIGH). The Proportional Error is set so OUT will not exceed MAXL. If the Proportional Error = 0, the Integral Error will start to increase to keep OUT below the MAXL limit.

If OUT < MINL: The following sequence of events will occur:

The MIN bit = 1 (HIGH). The Proportional error component is set so OUT will not exceed MINL. If the Proportional Error = 0, the Integral Error component will start to increase to keep OUT above the MINL limit.

If OUT > MAXL or H-UP = 1: The CMX bit = 1 (HIGH).

If OUT < MINL or H-DN = 1: The CMN bit = 1 (HIGH).

If H-UP = HIGH and PRE = LOW, then:

The Integral Error component cannot increase in the positive direction until the H-UP input goes low. Positive ERR values are ignored by the integral error component. The integrator hold up input is used to keep the integrator from increasing in the positive direction during major process disturbances. Note that the Proportional Error is not affected by the H-UP input.

If H-DN = HIGH and PRE = LOW, then:

The Integral Error component cannot increase in the negative direction until the H-DN input goes low. Negative ERR values are ignored by the integral error component. The integrator hold down input is used to keep the integrator from increasing in the negative direction during major process disturbances. Note that the Proportional Error is not affected by the H-DN input.

If PRE = HIGH, then:

The Integral Error component is set to the PREL value and the Proportional Error component is set to zero so that:

OUT = IERR = PREL

The IERR output is equal to the integrator component and can be used for diagnostic purposes.

4.55 QUAD LTCH

1. Inputs

BIT1:	Bit
BIT2:	Bit
BIT3:	Bit
BIT4:	Bit
EN:	Bit

2. Outputs

OUT1	Bit
OUT2	Bit
OUT3	Bit
OUT4	Bit

3. Implementation

On a low to high transition of the EN input, the following outputs are sampled.

If BIT1 = 1 then OUT1 = 1 else OUT1 = 0
If BIT2 = 1 then OUT2 = 1 else OUT2 = 0
If BIT3 = 1 then OUT3 = 1 else OUT3 = 0
If BIT4 = 1 then OUT4 = 1 else OUT4 = 0

If EN is low on power-up, then all outputs will equal 0.

If EN is high on power-up, then sample the bits.

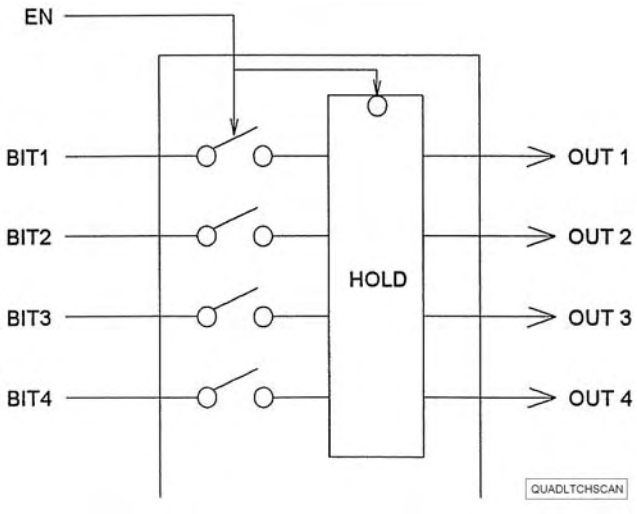


FIGURE 4-55. QUAD LTCH BLOCK

4.56 RAMP

The Ramp block provides a variable rate linear ramp with user programmable smoothing. The purpose of this block is to provide a smooth reference from changing setpoint values.

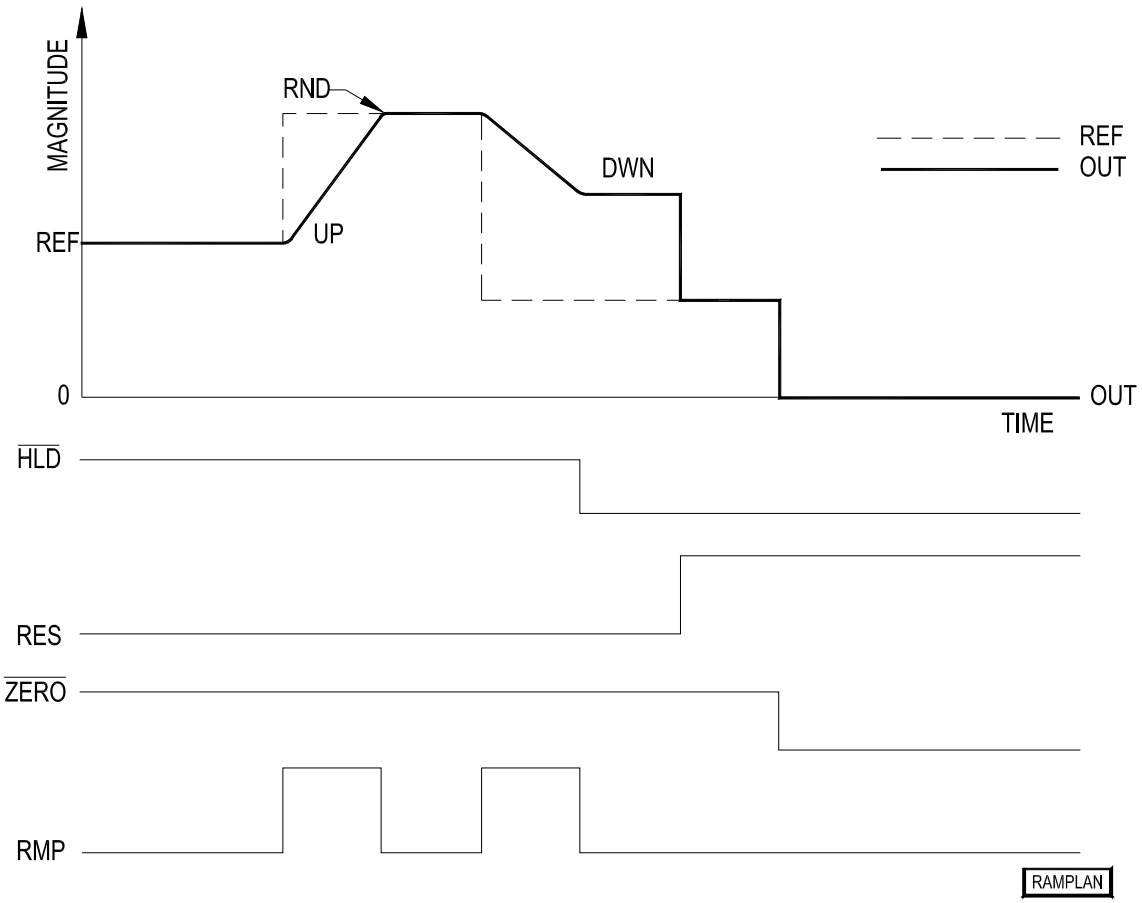


FIGURE 4-56. RAMP BLOCK

1. Inputs

REF:	Analog
UP:	Analog
DWN:	Analog
RND:	Analog
RES:	Bit
$\overline{\text{HLD}}$:	Bit
$\overline{\text{ZERO}}$:	Bit

2. Outputs

OUT: Analog
RMP: Bit

3. Implementation

If $\overline{\text{ZERO}}$ bit is low, OUT = 0. When $\overline{\text{ZERO}}$ bit goes high, OUT ramps to REF by the UP or DWN ramp rates. (Highest bit priority.)

If the RES bit is high, OUT = REF. (Second in priority.)

If the $\overline{\text{HLD}}$ bit is low, OUT is held at its present value. The rounding continues to prevent a step response. When $\overline{\text{HLD}}$ goes high, OUT ramps to REF by the appropriate rate.

The UP/DWN inputs are entered in units/second.

The following holds true while the $\overline{\text{HLD}}$, $\overline{\text{ZERO}}$ bits are set high and the RES bit is low:

If REF is increasing faster than UP, the RMP bit is set high and OUT ramps at the UP value. If UP is equal to zero, then the OUT ramps with the REF. If REF decreases faster than DWN, the RAMP bit is set high and OUT ramps at the DWN value. If DWN is equal to zero, then the OUT ramps with the REF. If neither of the preceding conditions are true, the RMP bit is set low and OUT equals REF.

The RND input determines the amount of S-ramp to be applied to the OUT. This is implemented as a low pass filter to the linear ramp. The RND value is entered as a time constant in seconds. If RND < .005, no rounding will take place.

NOTE

On powerup, OUT = REF.

4.57 RATE CHANGE

This block is used to restrict the rate of change of an analog signal to a user programmable maximum rate limit.

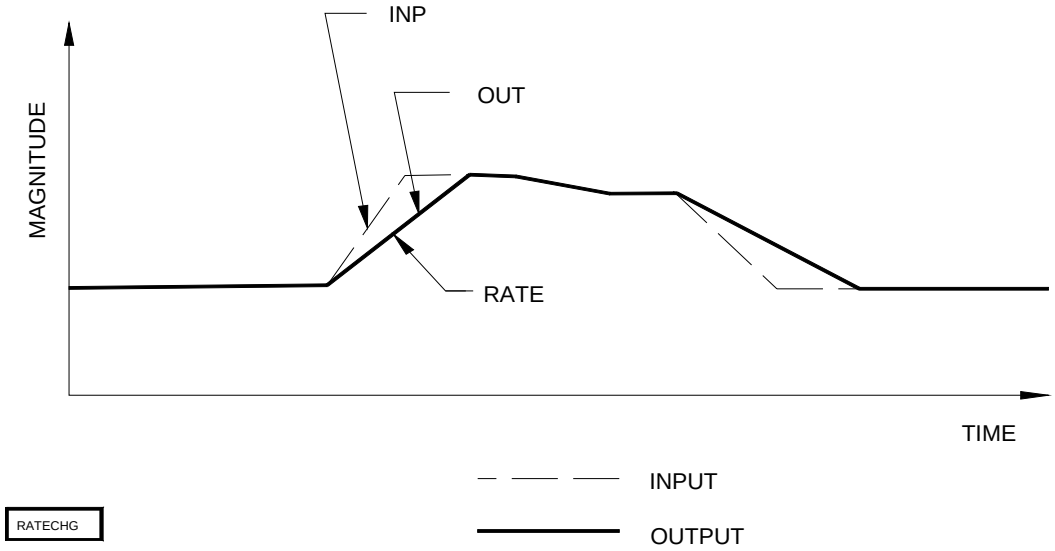


FIGURE 4-57. RATE CHANGE BLOCK

1. Inputs

   INP: Analog
   RATE : Analog

2. Outputs

   OUT: Analog

3. Implementation

   RATE is scaled in units/second.

   If INP is changing less than RATE, OUT is equal to INP. If INP is changing faster than RATE, OUT changes at the RATE value.

4.58 RATIO

The Ratio Block calculates the diameter of a center driven winder. It can be used for a winder or an unwinder. The initial diameter is a preset input value to obtain the correct diameter starting speed.

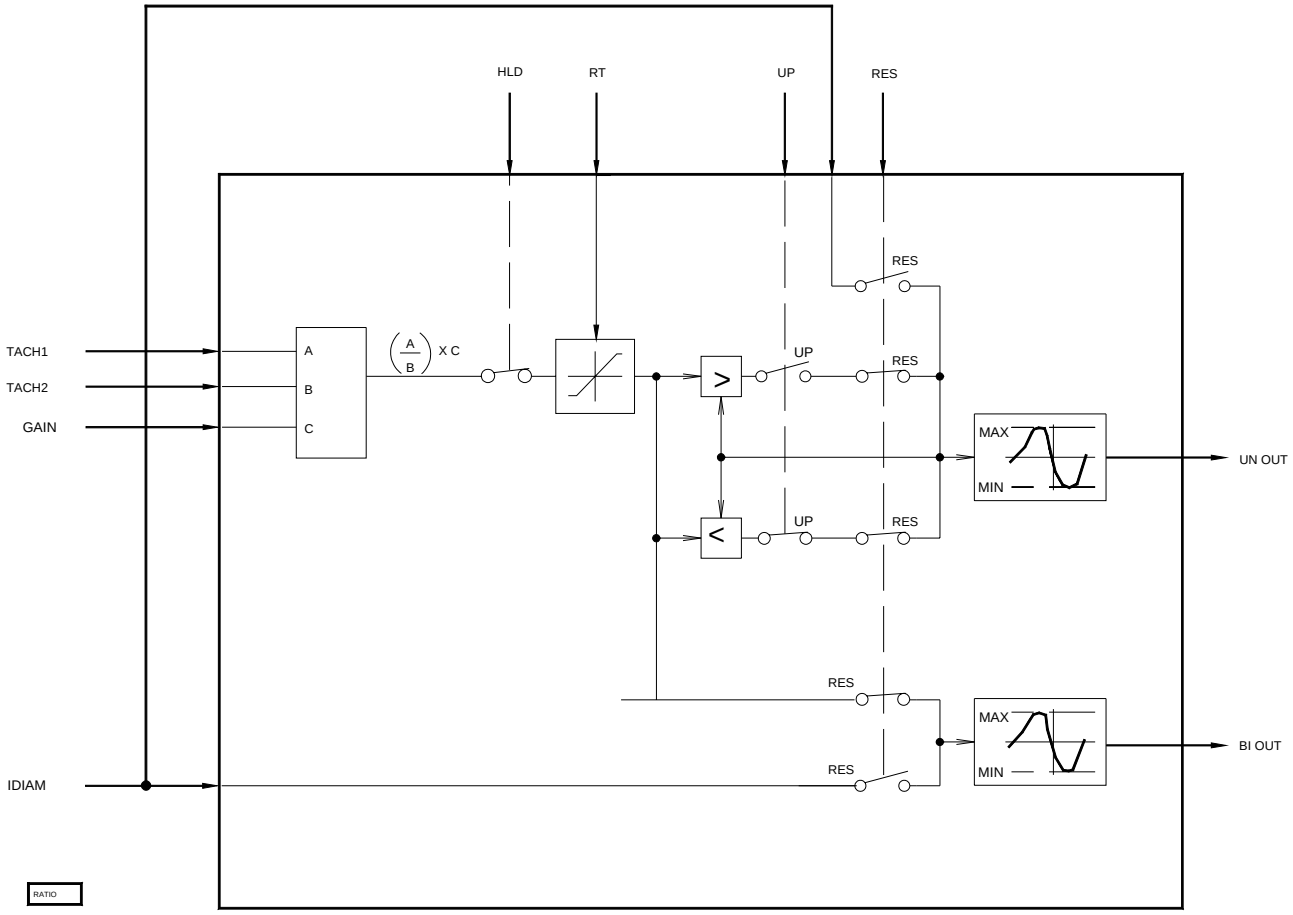


FIGURE 4-58. RATIO BLOCK

1. Inputs

IDIAM:	Analog
TACH1:	Analog
TACH2:	Analog
GAIN:	Analog
MAX:	Analog
MIN:	Analog
RT:	Analog
HLD:	Bit

Inputs (Cont.)

UP:	Bit
RES:	Bit
RET:	Bit

2. Outputs

BIOUT:	Analog
UNOUT:	Analog

3. Implementation

[(TACH1/TACH2) × GAIN] = BIOUT if it is within MIN and MAX, else BIOUT will equal the limit value. BIOUT is rate limited by the RT value. RT is entered in units/second and should be set at maximum rate of change at core at max speed.

If the HOLD bit goes high, then BIOUT and UNOUT will be held at their current values. Releasing this bit lets the block resume calculations.

If UP bit is high, then UNOUT equals the highest BIOUT since the last reset. If the UP bit is low, then the UNOUT equals the lowest BIOUT since the last reset.

The RES bit is used to reset both BIOUT and UNOUT to the IDIAM value. It stays at this value as long as the RES bit is high. The RES bit has a higher priority than the HOLD bit.

Non-retentive Block

On powerup of the ADDvantage-32, UNOUT = IDIAM
BIOUT = IDIAM

Retentive Block

On powerup of the ADDvantage-32, BIOUT and UNOUT will be initialized under the following conditions:

If RET = 0, then UNOUT = IDIAM and
BIOUT = IDIAM

If RET = 1, UNOUT and BIOUT are set to their last value. UNOUT and BIOUT must also be configured to retentive points to be updated automatically on powerup.

4.59 RECIPE

1. Inputs

RECA:	Analog
RECB:	Analog
RECC:	Analog
ENA:	Bit
ENB:	Bit
ENC:	Bit
EN:	Bit

2. Outputs

OUT:	Analog
FLT:	Bit

3. Implementation

RECA, RECB, RECC points to the first address of 10 consecutive calibration locations.

OUT points to the first address of 10 consecutive analog locations.

On power up, the OUT locations (10 values) are set to the RECA values.

When the EN bit is high, the block will select which block of 10 values to output based on the ENA, ENB, and ENC bits.

ENA transfers RECA data.
ENB transfers RECB data.
ENC transfers RECC data.

When EN is low, no transitions occur.

ENA has the highest priority and is also used when no enable is present.

ENB is the second highest priority.

On power up, FLT output bit is low.

Whenever EN is enabled, the FLT output is checked. FLT will go high if no bits are selected or if more than one is selected.

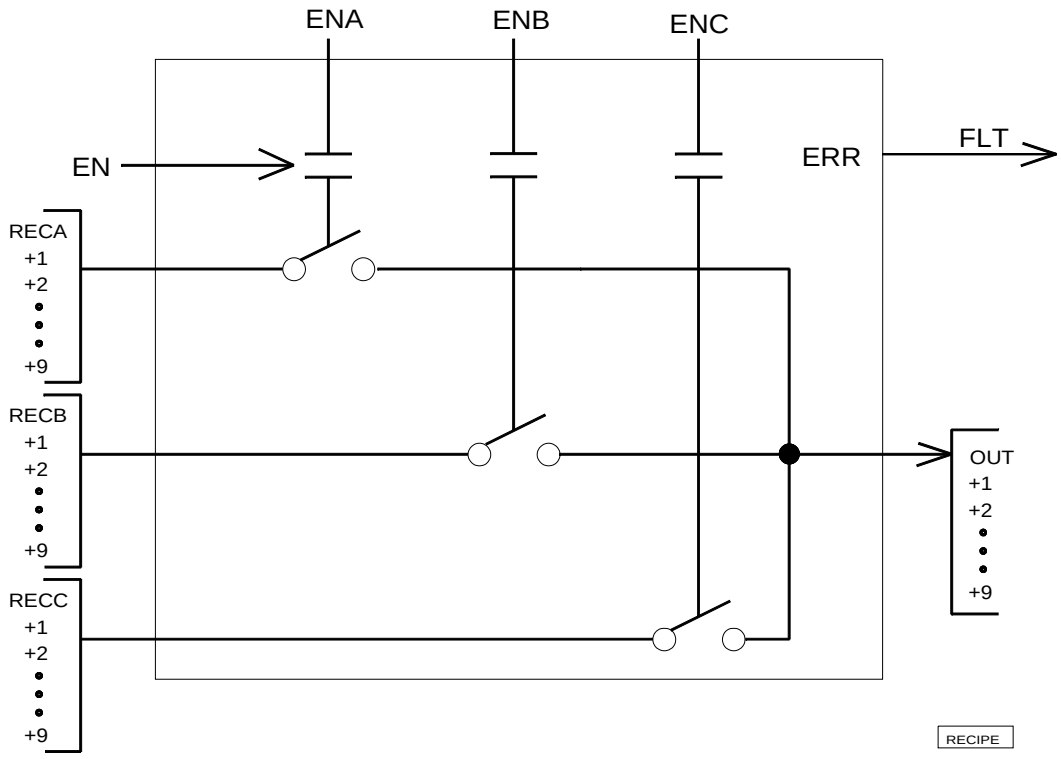


FIGURE 4-59. RECIPE BLOCK

4.60 RESOLVER 1

The RESOLVER 1 block is only used in conjunction with the Avtron optional Resolver board. The Resolver board mounts on the first SBX site on the Maxi system board. It provides 16 digital inputs and two digital outputs. This block provides the software interface to the board.

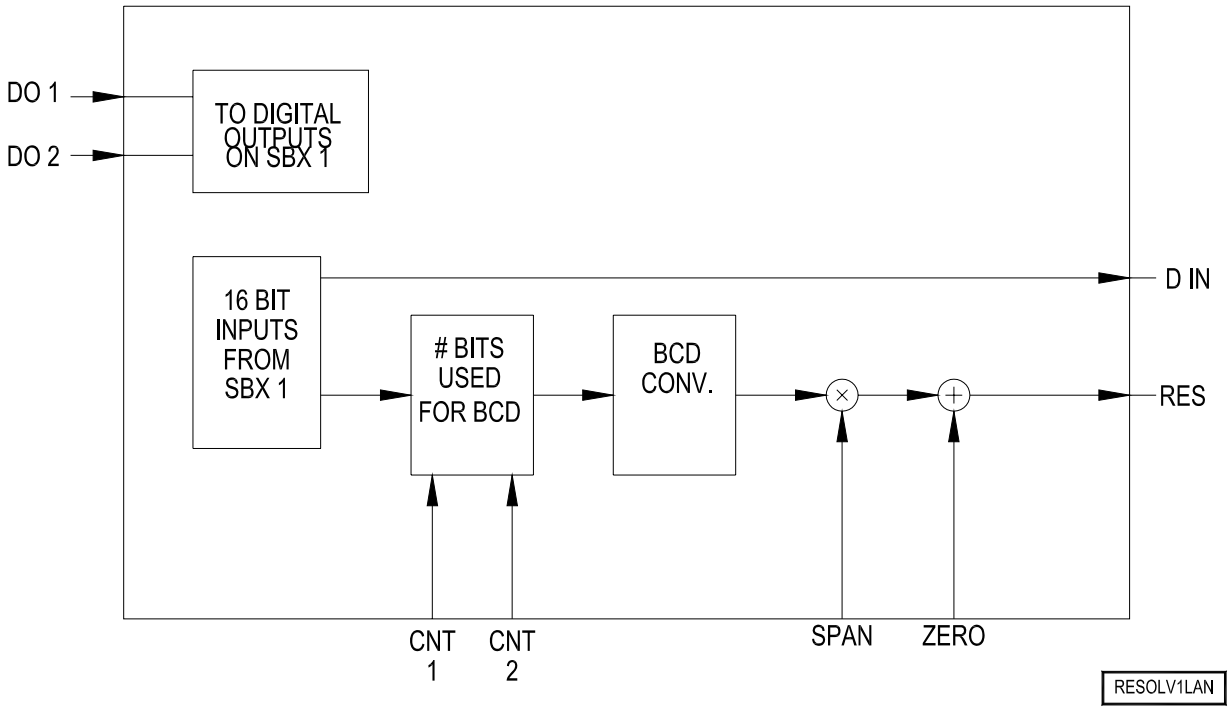


FIGURE 4-60. RESOLVER 1 BLOCK

1. Inputs

SPAN:	Analog
ZERO:	Analog
DO 1:	Digital
DO 2:	Digital
CNT 1:	Digital
CNT 2:	Digital

2. Outputs

RES:	Analog
D IN:	Digital

3. Implementation

DO 1 and DO 2 bits control the two digital outputs on the resolver board. When they are equal to one, the corresponding output is high.

CNT 1 and CNT 2 decide how many of the input bits are used in a BCD conversion for a resolver as follows:

CNT 1	CNT 2	Number of bits used
0	0	4 bits
1	0	8 bits
0	1	12 bits
1	1	16 bits

The number of bits determined by the above table is then converted to a BCD number and then scaled to the block's output value (RES) as follows:

(BCD number × SPAN) + ZERO = RES

EXAMPLE:

-CNT 1: LOW
-CNT 2: HIGH

–INPUT BITS	0100	0101	0000
–BCD VALUE	4	5	0

–SPAN = 2

–OFF = 100

–RES = (450 × 2) + 100 = 1,000

All 16 input bits are outputted as digital bits in consecutive order starting at D IN location. Name the next 16 locations, but do not connect them to an output of another block or unpredictable results will occur.

4.61 RMP2

The RMP2 block provides a variable rate linear ramp with user programmable smoothing. The purpose of this block is to provide a smooth reference from changing setpoint values.

The difference between the RAMP block and RMP2 is the RND input. RAMP uses a low-pass filter for rounding, causing a "J" curve. The RMP2 implements a rate of change limit on the ramp, causing a uniform "S" curve.

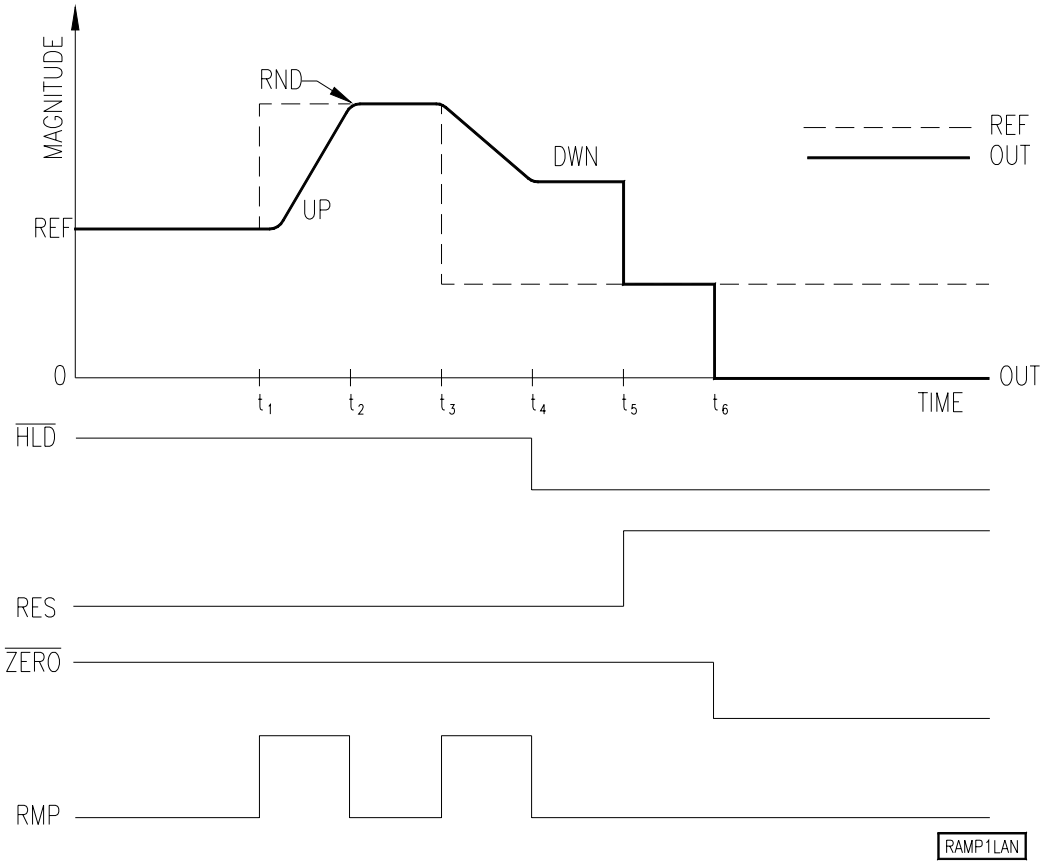


FIGURE 4-61. RMP2 BLOCK

The RMP2 block can be divided internally into three components to aid in the understanding of its operation. These components are a Ramp Change Block, a Rate of Change Limit Block, and Additional Control I/O as can be seen in Figure 4-61A.

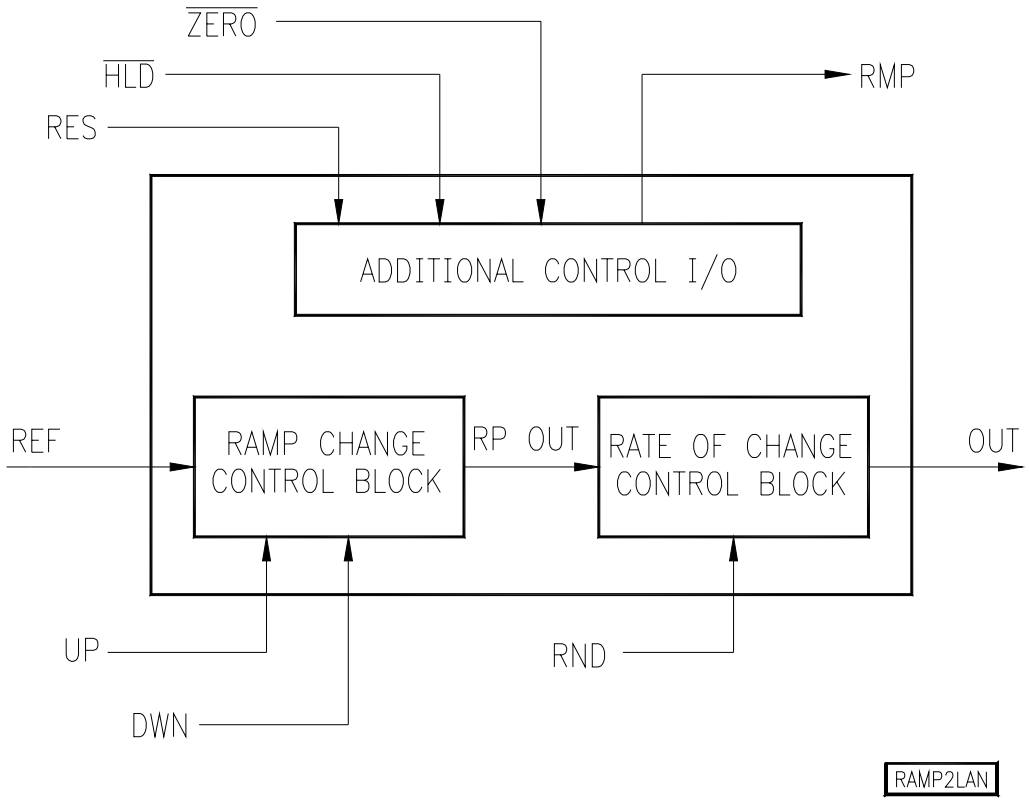


FIGURE 4-61A

RAMP CHANGE BLOCK

1. Inputs

   UP: Analog
   DWN: Analog
   REF: Analog

2. Outputs

   RP OUT: Analog
   RMP: Bit

3. Implementation

   The following holds true while the HLD, ZERO bits are set high and the RES bit is low (explained in detail in ADDITIONAL CONTROL I/O).

If REF is increasing faster than UP, the RMP bit is set high and RP OUT ramps at the UP value (time period t_1 to t_2 of Figure 4-61). If REF decreases faster than DWN, the RMP bit is set high and RP OUT ramps at the DWN value (time period t_3 to t_4 of Figure 4-61). If neither of the preceding conditions is true, the RMP bit is set low and RP OUT equals REF.

RATE OF CHANGE LIMIT BLOCK

1. Inputs

   INP: Analog (Internally connected to the Ramp Output)
   RND: Analog

2. Outputs

   OUT: Analog

3. Implementation

   The rate of change limit block (ROC filter) is internal to the RMP2 Control Block. The purpose of this stage of the RMP2 block is to round the corners created by the ramp change block (Figure 4-61B). The ROC filter produces rounding by limiting the ramp rate of the input to the output over time. The limiting occurs as a result of a calculated value internal to the RMP2 called the Maximum Rate of Change. When the input is greater than the Maximum Rate of Change value, the output equals the Maximum Rate of Change value. When the Maximum Rate of Change equals the rate of the input, the ROC stops limiting the output.

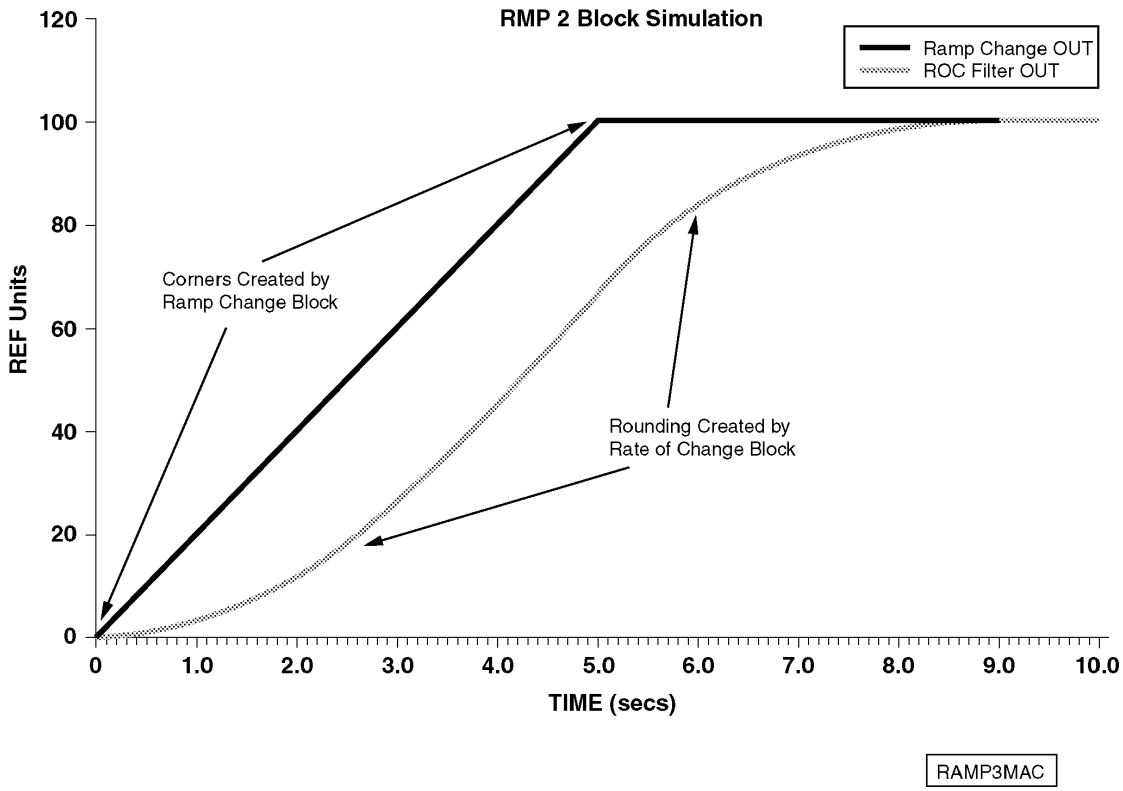


FIGURE 4-61B

The maximum rate of change allowed by the RMP2 ROC filter block depends upon time. This time is measured from the beginning of a ramp input to the present time instant. The maximum rate of change of the output of this filter is time dependent of the form:

$$Rate = S * T$$

Equation 4-61.1

where S is the analog value RND (units/Sec.^{2}), and T (Sec.) is the point in time since the start of the ramp input. The maximum rate of change of the filter output will not exceed that of the input over a long enough time interval.

As an example, note on Figure 4-61B that the plot starts at time t = 0 Sec. The rate of change at the input of the filter block is a constant 20 FPM/S. The filter limits the rate of change block to 5FPM/Sec^2 thus:

Exact Time	ROC Filter (Slope) at OUT
1 Sec.	5 FPM/SEC
2 Sec.	10 FPM/SEC
3 Sec.	15 FPM/SEC
4 Sec.	20 FPM/SEC
5 Sec.	20 FPM/SEC

Notice that since the input to the filter is 20.0 FPM/SEC, the rate from 4 to 5 Sec. of the output does not increase past 20 FPM/SEC.

At 5 seconds, the Rate Change block reaches its final value (100 FPM) and the rate of change becomes zero. The filter will now cause the output rate to decrease by 5 FPM/SEC^2 from its maximum value of 20FPM/S while the input rate is zero, thus:

Exact Time	Maximum rate of change (Slope)
1 Sec.	5 FPM/SEC
2 Sec.	10 FPM/SEC
3 Sec.	15 FPM/SEC
4 Sec.	20 FPM/SEC
5 Sec.	20 FPM/SEC
6 Sec.	15 FPM/SEC
7 Sec.	10 FPM/SEC
8 Sec.	5 FPM/SEC
9 Sec.	0 FPM/SEC
10 Sec.	0 FPM/SEC

To accurately select an appropriate value for the RND input, apply Equation 4-61.1 in the form:

$$S = Rate/T$$

Where:
S = Value of Analog input RND
Rate = Value of UP or DWN.
T = Desired time length for ROC to provide rounding

Equation 4-61.2

ADDITIONAL CONTROL I/O

1. Inputs

$\overline{\text{ZERO}}$	Bit
$\overline{\text{HLD}}$	Bit

2. Outputs

RES	Bit

If $\overline{\text{ZERO}}$ bit is low, OUT = 0. When $\overline{\text{ZERO}}$ bit goes high, OUT ramps to REF by the UP or DWN ramp rates in addition to the RND input.

If the RES bit is high, OUT = REF. (Second in priority)

If the $\overline{\text{HLD}}$ bit is low, OUT is held at its present value. The rounding continues to prevent a step response. When HLD goes high, OUT ramps to REF by the appropriate rate.

The following figures 4-61C, 4-61D, and 4-61E graphically represent various RMP2 Control block input results.

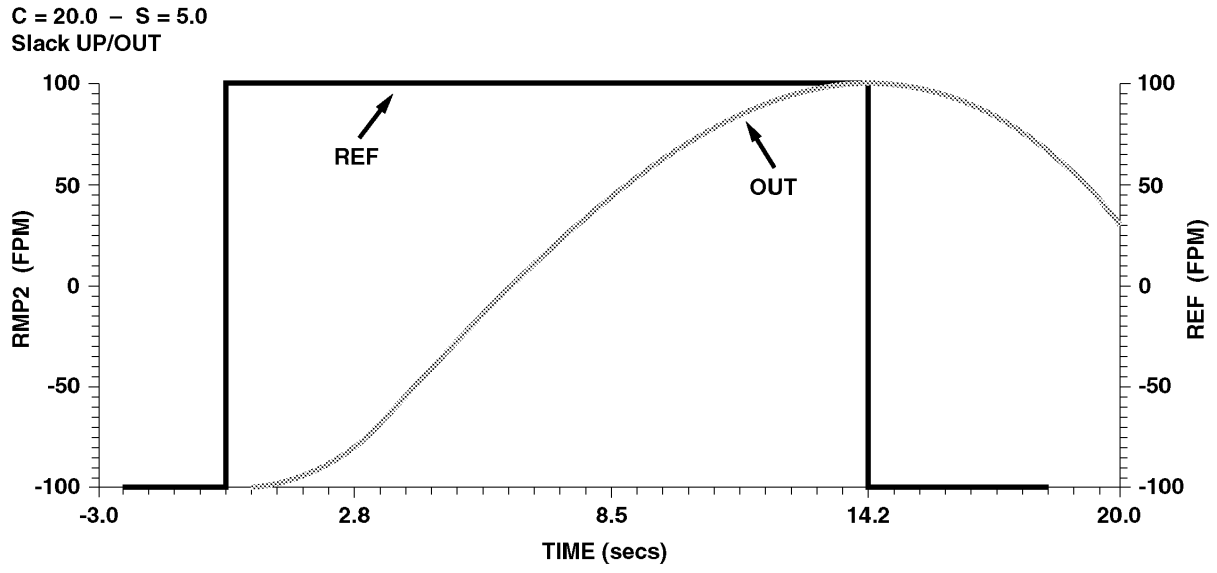
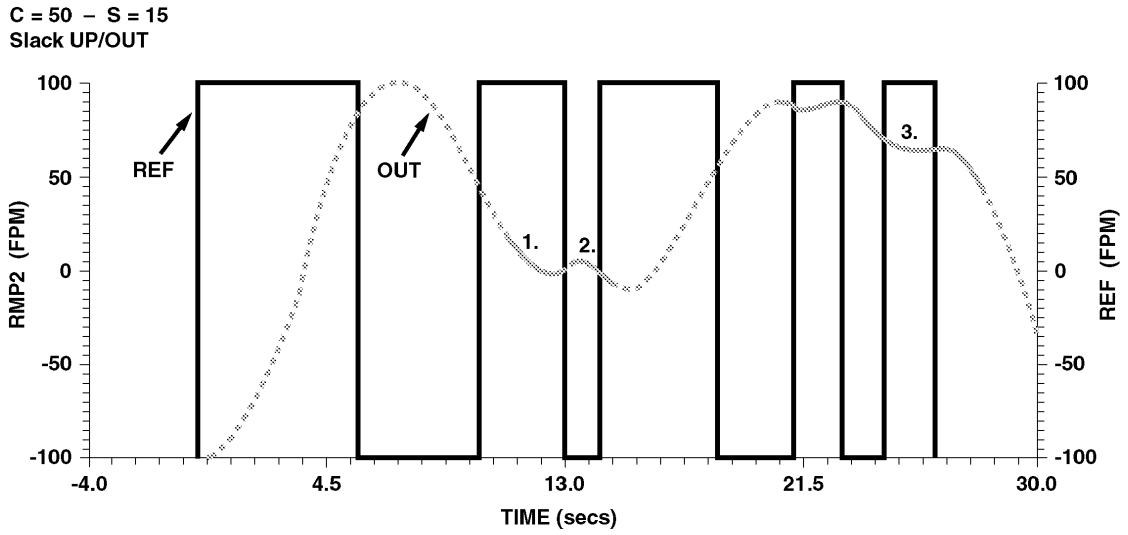


FIGURE 4-61C



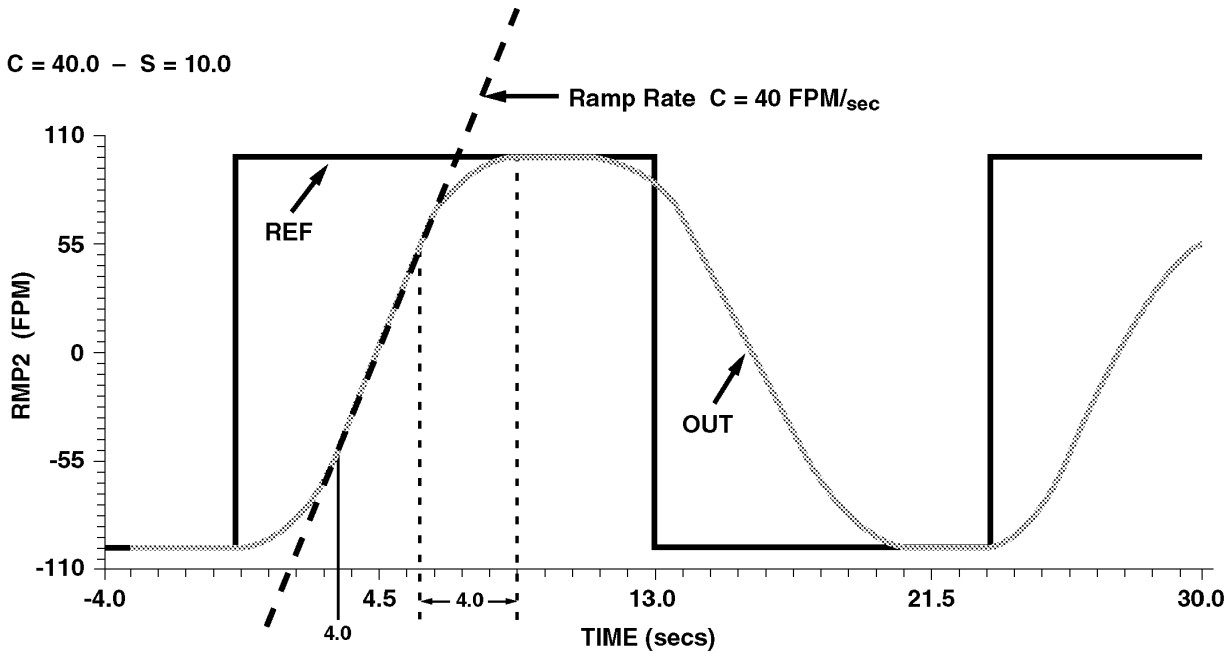
1.) RMP2 tries to turn around to follow the input while continuing to perform rounding.

2.) Easy transition from a small positive slope to change negative and follow input.

3.) RMP2 not very responsive to switching rates > RND

UP and DWN RATE = 50 FPM/$_{sec}$; RND = 15 FPM/$_{sec^2}$ RAMP5MAC

FIGURE 4-61D



Calculation for S (where S = RND in units/sec^2)
to obtain desired output, which is described below.

$$S = \frac{\text{RATE (Value of UP)}}{\text{T (Chosen as 4 sec.)}} \qquad \frac{40 \text{ FPM/sec}}{4 \text{ sec}} = 10 \text{ FPM/sec}^2$$

With a process that ramps up at a 40 FPM/sec rate, entering a SLCK SCURVE (RND; rounding rate) of 10 FPM/sec^2, will result in a starting and ending curve that takes 4 seconds to complete.

RAMP6MAC

FIGURE 4-61E

4.62 RRAMP3

The RRAMP3 block provides a variable rate linear ramp with user programmable smoothing. The purpose of this block is to provide a smooth reference from changing setpoint values.

The difference between the RRAMP3 block and RMP2 is the additional RAMP inputs. The RRAMP3 block has four ramp rates instead of two. See below.

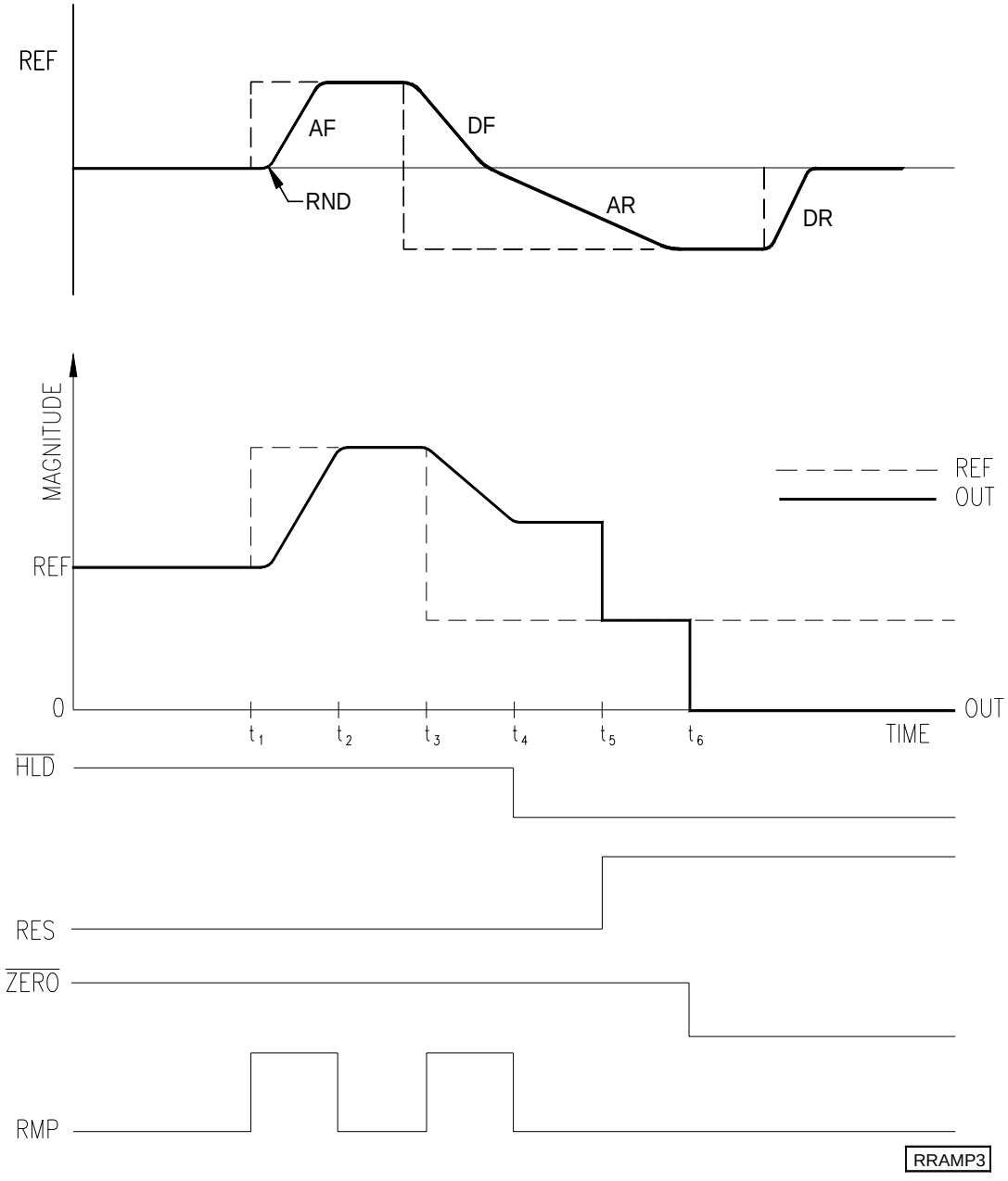


FIGURE 4-62. RRAMP3 BLOCK

The RRAMP3 block can be divided internally into three components to aid in the understanding of its operation. These components are a Ramp Change Block, a Rate of Change Limit Block and Additional Control I/O as can be seen in Figure 4-62A.

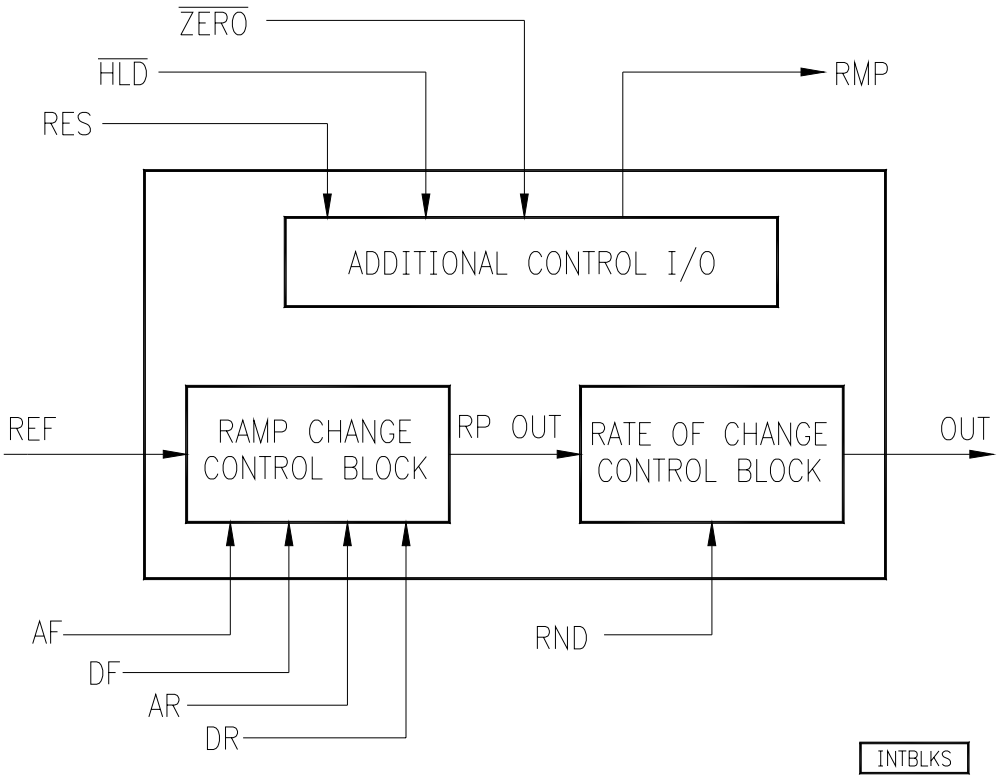


FIGURE 4-62A. INTERNAL CONTROL BLOCKS

RAMP CHANGE BLOCK

1. Inputs

REF:	Analog
AF:	Analog
DF:	Analog
AR:	Analog
DR:	Analog
RND:	Analog
RES:	Digital
$\overline{\text{HLD}}$:	Digital
$\overline{\text{ZERO}}$:	Digital

2. Outputs

RP OUT: Analog
RMP: Bit

3. Implementation

The following holds true while the HLD, ZERO bits are set high and the RES bit is low (explained in detail in ADDITIONAL CONTROL I/O).

If REF is increasing (positive) faster than AF, the RMP bit is set high and RP OUT ramps at the AF value (time period t_1 to t_2 of Figure 4-61). If AF is equal to zero, then the RP OUT ramps with REF. If REF decreases (but still positive) faster than DF, the RMP bit is set high and RP OUT ramps at the DF value (time period t_3 to t_4 of Figure 4-62). If DF is equal to zero, then the RP OUT ramps with REF. If neither of the preceding conditions is true, the RMP bit is set low and RP OUT equals REF. AR and DR work the same as AF and DF but are used when RP OUT becomes negative.

RATE OF CHANGE LIMIT BLOCK

The rest of section 4.62 RRAMP3 is similar to the corresponding part of section 4.61 RMP2.

4.63 SBXDIA

The SBXDIA block calculates the diameter by relating a reference pulse generator with a once-per-revolution pickup. This block requires the GSBX board connected to the Maxi system board. The board provides the following:

Reference pulse generator input
(2) Once-per-revolution pickups
(8) Definable digital inputs
(4) Definable digital outputs

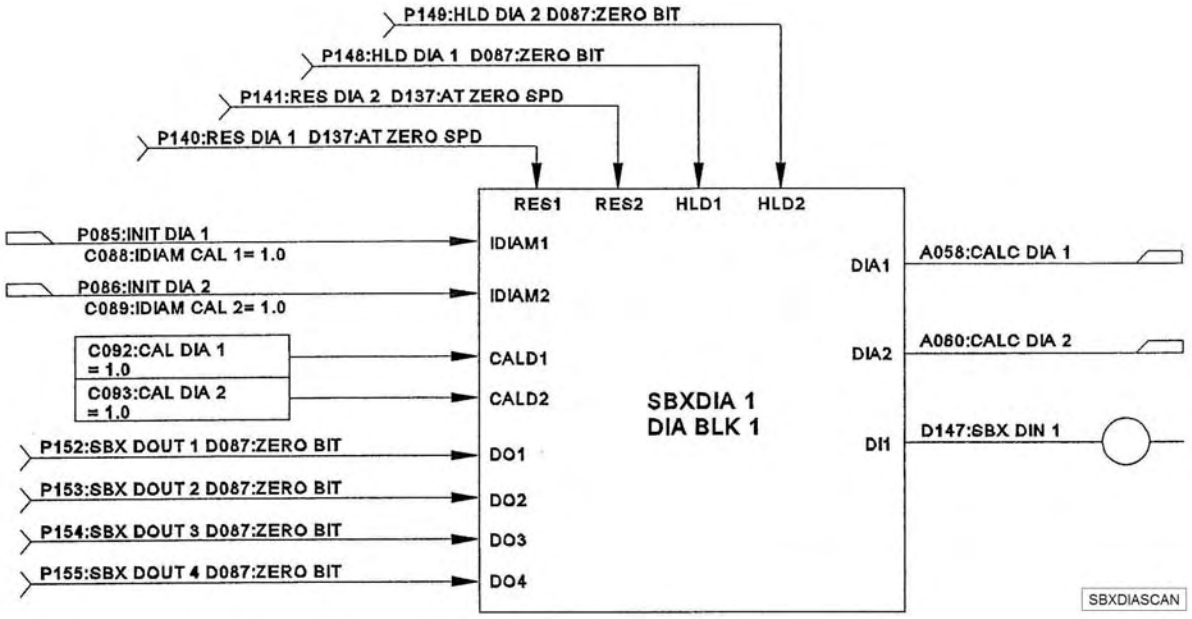


FIGURE 4-63. SBXDIA BLOCK

1. Inputs

IDIAM1	Analog
IDIAM2	Analog
CALD1	Analog
CALD2	Analog
RES1	Digital
RES2	Digital
HLD1	Digital
HLD2	Digital

Inputs (Continued)

DO1	Digital
DO2	Digital
D03	Digital
D04	Digital

2. Outputs

DIA1	Analog
DIA2	Analog
DI1	Digital

3. Implementation

If the RES1 is high, then DIA1 = IDIAM1.

If HLD1 is high, the output is frozen at its last value. RES1 has a higher priority.

If RES1 and HLD1 are low, then DIA1 is calculated as follows:

DIA1 = (Edges read since last pickup reading) * CALD1/PI

The first reading after a hold or reset is ignored by the block. This is to avoid a false diameter recording.

RES2, HLD2, DIA2, CALD2 work on the same principles.

DO1 through DO4 are the programmable digital outputs available on the board. These outputs must be isolated.

DI1 - First location where the eight digital inputs are stored. It will use the next seven locations to store the information.

4. Application Note

The GSBX board has the following hardware assignments for this block.

Pin 15 - Reference Frequency Input
Pin 17 - Pick-up pulse 1
Pin 4 - Pick-up pulse 2
Pins 9, 13, 22 - Common

Additionally, this block can be used to calculate values other than diameter, by factoring out PI in the calibration number. One such use is for a length calculator, where product is metered and a proximity input resets when complete. The result is length.

4.64 SBXDIA4

The SBXDIA4 block calculates the diameter by relating a reference pulse generator with a once-per-revolution pickup. This block requires the GSBX board connected to the Maxi system board. The board provides the following:

Reference pulse generator input
(4) Once-per-revolution pickups
(8) Definable digital inputs
(4) Definable digital outputs

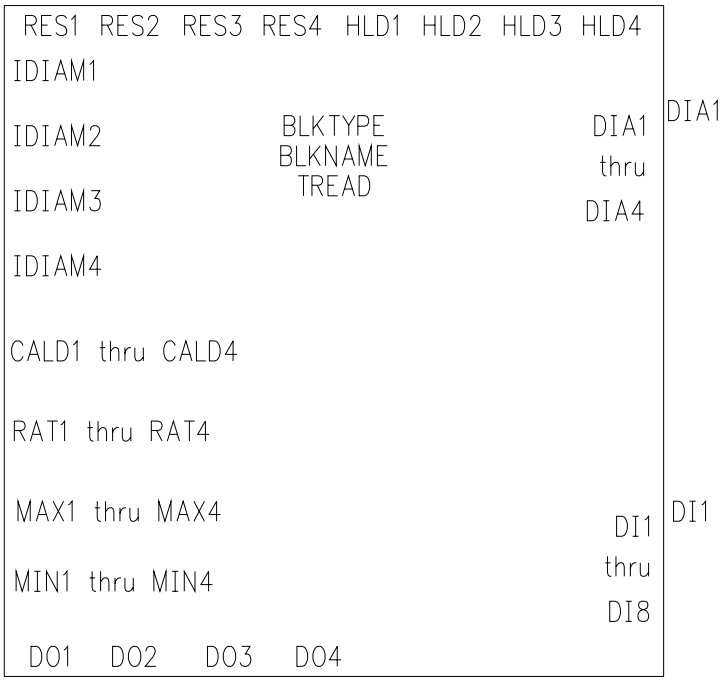


FIGURE 4-64. SBXDIA4 BLOCK

1. Inputs

IDIAM1	Analog
IDIAM2	Analog
IDIAM3	Analog
IDIAM4	Analog
CALD1	Analog
RATE1	Analog
MAX DIA1	Analog
MIN DIA1	Analog

Inputs (Cont.)

RES1	Digital
RES2	Digital
RES3	Digital
RES4	Digital
HLD1	Digital
HLD2	Digital
HLD3	Digital
HLD4	Digital
DO1	Digital
DO2	Digital
DO3	Digital
DO4	Digital

2. Outputs

DIA1	Analog
DI1	Digital

3. Implementation

This block uses the GSBX board to generate four diameter readings. It does this by calculating the distance the material has traveled based on one revolution of the roll. The GSBX board has a pulse generator input to determine the distance and four inputs (one for each roll), which record once per revolution.

CALD1, RATE1, MAX DIA1, MIN DIA1, and DIA1 all point to the first diameter's parameters. The other three diameter parameters are in consecutive order.

If the RES1 is high, then DIA1 = IDIAM1.

If HLD1 is high, the output is frozen at its last value. RES1 has a higher priority.

If RES1 and HLD1 are low, then DIA1 is calculated as follows:

DIA1 = (Edges read since last pickup reading) * CALD1/PI

The first reading after a hold or reset is ignored by the block. This is to avoid a false diameter recording.

RATE1 limits the rate of change of DIA1 when RES1 is not high. This is set in diameter process units per second.

DIA1 is limited to be within MAX DIA1 and MIN DIA1. This is also true while RES1 is active.

DO1 through DO4 are the programmable digital outputs available on the board. These outputs must be isolated.

DI1 is the first location where the eight digital inputs are stored. The next seven locations store the next seven inputs.

4. Application Note

The GSBX board has the following hardware assignments for this block.

Pin 15 - Reference Frequency Input
Pin 17 - Pick-up pulse 1
Pin 4 - Pick-up pulse 2
Pin 16 - Pick-up pulse 3
Pin 3 - Pick-up pulse 4
Pins 9, 13, 22 - Common

DIN1 - Pin 21
DIN2 - Pin 8
DIN3 - Pin 20
DIN4 - Pin 7
DIN5 - Pin 19
DIN6 - Pin 6
DIN7 - Pin 18
DIN8 - Pin 5

DOUT1 - Pin 12
DOUT2 - Pin 24
DOUT3 - Pin 11
DOUT4 - Pin 23

Additionally, this block can be used to calculate values other than diameter. To do this, factor out PI in the calibration number. One such use is for a length calculator, where product is metered and a proximity input resets when complete.

4.65 SDS SHEETS DURING STOP

This block calculates the number of sheets to be cut during a controlled stop (SDS). When using the AUTO STOP* option, the number of sheets entered by the operator is subtracted from the number of sheets cut during the stop. When the selected number of sheets has been cut, the line will automatically begin to ramp to a stop. The line will stop after cutting the selected number of sheets.

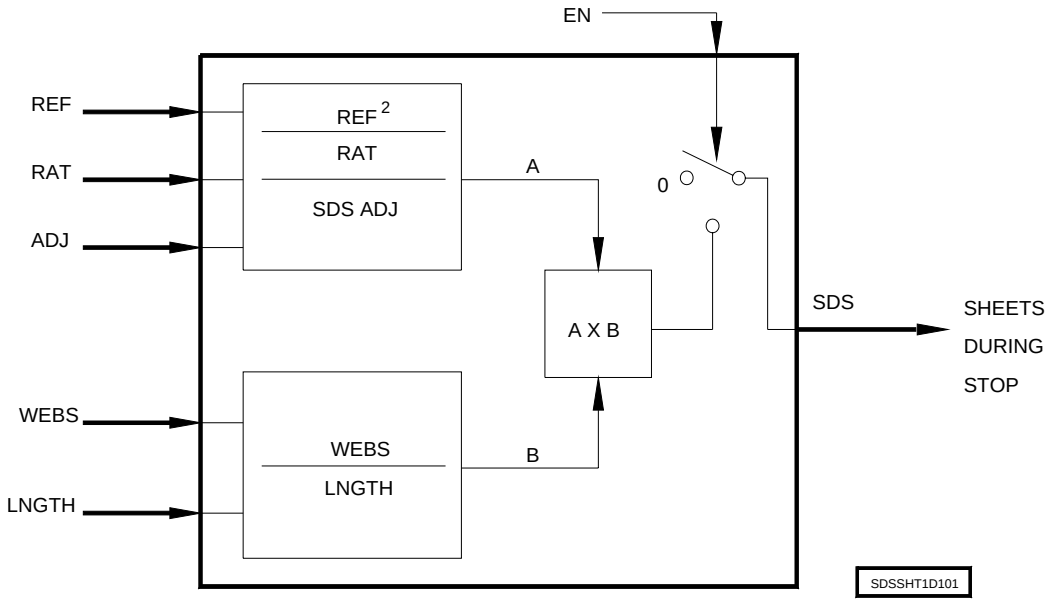


FIGURE 4-65. SDS SHEETS DURING STOP BLOCK

1. Inputs

REF:	Analog
RAT:	Analog
ADJ:	Analog
WEBS:	Analog
LNGTH:	Analog
EN:	Digital

* This function is not part of the ADDvantage-32 but is part of relay logic.

2. Outputs

SDS: Analog

3. Implementation

If EN is high:

$$SDS = \frac{(REF^2) \times WEBS}{RAT \times ADJ \times LNGTH}$$

Where:

RAT = line speed in units per second

$$ADJ = \frac{(2)(60)^{**}}{(\text{CUT LENGTH UNITS} \div \text{LINE SPEED UNITS})} = 10$$

Else:

SDS = 0

** 60 = 60 seconds/1 minute to adjust stopping rate in seconds and line speed in minutes.

2 = the factor used to obtain the integral number of sheets during the ramped stop.

4.66 SETPOINT

The Setpoint block enables the use of the INC and DEC inputs to modify a setpoint. The rates for the INC/DEC are adjustable along with the limits. The limits can be defined as min/max difference or ratio draw. This block is used in the beginning of the speed loop and tension loop.

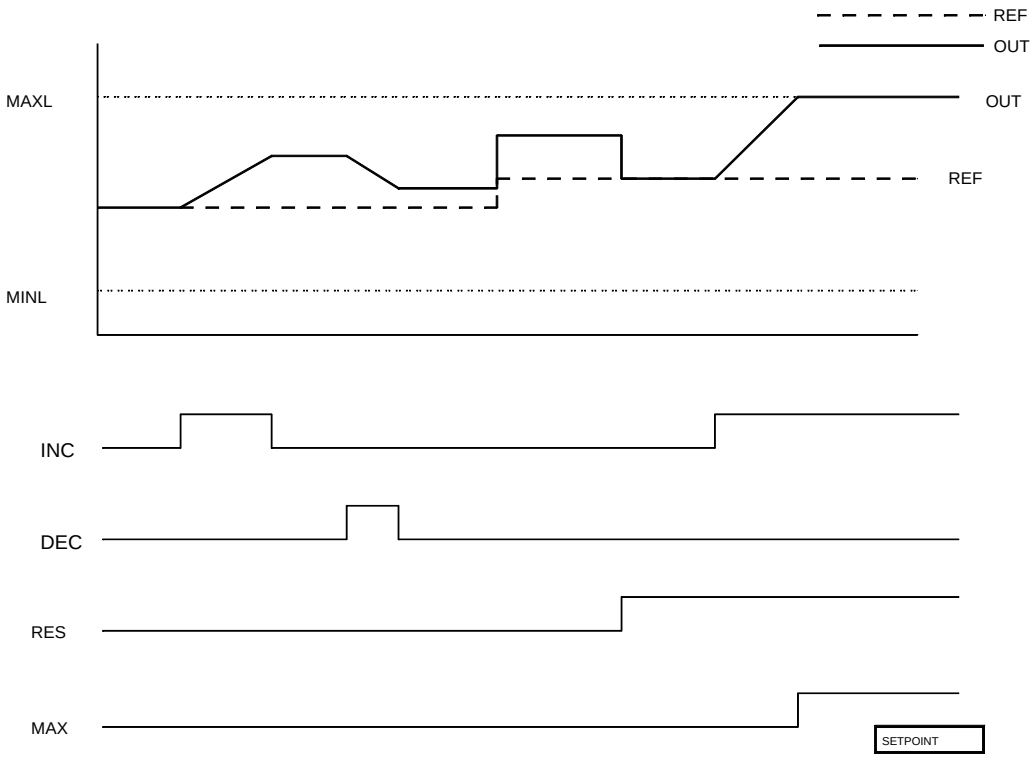


FIGURE 4-66. SETPOINT BLOCK

1. Inputs

REF:	Analog
RES:	Bit
INC:	Bit
DEC:	Bit
INCR:	Analog
DECR:	Analog
MAXL:	Analog
MINL:	Analog
ABS:	Bit
UP:	Bit
RET:	Bit

2. Outputs

OUT:	Analog
DIF:	Analog
MAX:	Bit
MIN:	Bit

3. Implementation

When the INC bit is high, the OUT value starts to ramp up at INCR rate. INCR is a rate entered in process units per second. The ramp continues until the INCR bit goes low or the OUT value reaches a limit. Upon reaching the limit, the MAX bit goes high and stays high until the OUT value goes below the limit. The limit is defined as follows.

If the ABS bit is low, the OUT value clamps at MAXL × REF.

If the ABS bit is high, the OUT value clamps at MAXL + REF.

The DEC, DECR and MIN operate in the same manner as the INC functions.

If the UP input is high and REF changes, OUT will reset to equal REF and the INC/DEC changes will be negated.

If the UP input is low and REF changes, OUT changes depending on the ABS input. If the ABS bit is low, the OUT value changes by the REF value multiplied by the ratio of the INC/DEC change.

Example: ABS BIT = LOW

REF equals 100 and the INC bit is set high until OUT is increased to 150. Then if REF is changed to 200, OUT will equal 300.

$$\frac{(150)}{100} \times 200 = 300$$

If the ABS bit is high, the OUT value changes by the REF plus the INC/DEC change.

Example: ABS BIT = HIGH

REF equals 100 and the INC bit is set high until OUT is increased to 150. Then if REF is changed to 200, OUT will equal 250.

$$(150\text{-}100) + 200 = 250$$

On a low to high transition of the RES bit, OUT resets to the REF value.

DIF is equal to the change of the setpoint.

If ABS = 1, then DIF =
OUT - REF (within the limits)

If ABS = 0, then DIF = $\frac{\text{OUT}}{\text{REF}}$ (within the limits)

RETENTIVE BLOCK

On powerup of the drive, OUT will be initialized under the following conditions:

If RET = 0 and ABS = 0,
then DIF = 1 (within limits)

If RET = 0 and ABS = 1,
then DIF = 0 (within limits)

If RET = 1, then DIF is not initialized.
If ABS = 0, then OUT = INP × DIF.
If ABS = 1, then OUT = INP + DIF.

If a retentive setpoint is desired, configure a retentive point Y***:RET SETPT* to DIF so it updates automatically upon powerup. Configure the RES input to a zero bit to prevent resetting the output to the REF value instead of the retentive value.

4.67 SNAPAVG

1. Inputs

INP:	Analog
TICS:	Analog
RES:	Bit
HLD:	Bit
RET:	Bit

2. Outputs

OUT:	Analog
TTL:	Analog
CNT:	Analog

3. Implementation

The SNAPAVG block is used to get an average of a value with a specific time frame. The block has two internal count values to enable the block to function on short windows or longer windows.

TICS is internally limited to positive whole numbers.

On power up if RET is High, then the internal Count2 and Sum2 is set equal to Outputs CNT and TTL. Otherwise they are set equal to zero. Also, internal Count1 and Sum1 is set to zero on powerup. All three outputs go to zero on powerup also.

When the RES bit is high, then both counters and sums are set equal to zero.

When the hold bit is high, both counters and sums are held at their current value.

If both RES and HLD are low, then:

> Every time the block is executed, INP is added to the internal Sum1. Also, the internal value Count1 is incremented.

When Count1 is greater than TICS, then:

Sum2 = Sum1 / Count1
Count2 is incremented
Sum1 and Count1 is reset to zero.
OUT = Sum2 / Count2
TTL = Sum2
CNT = Count2

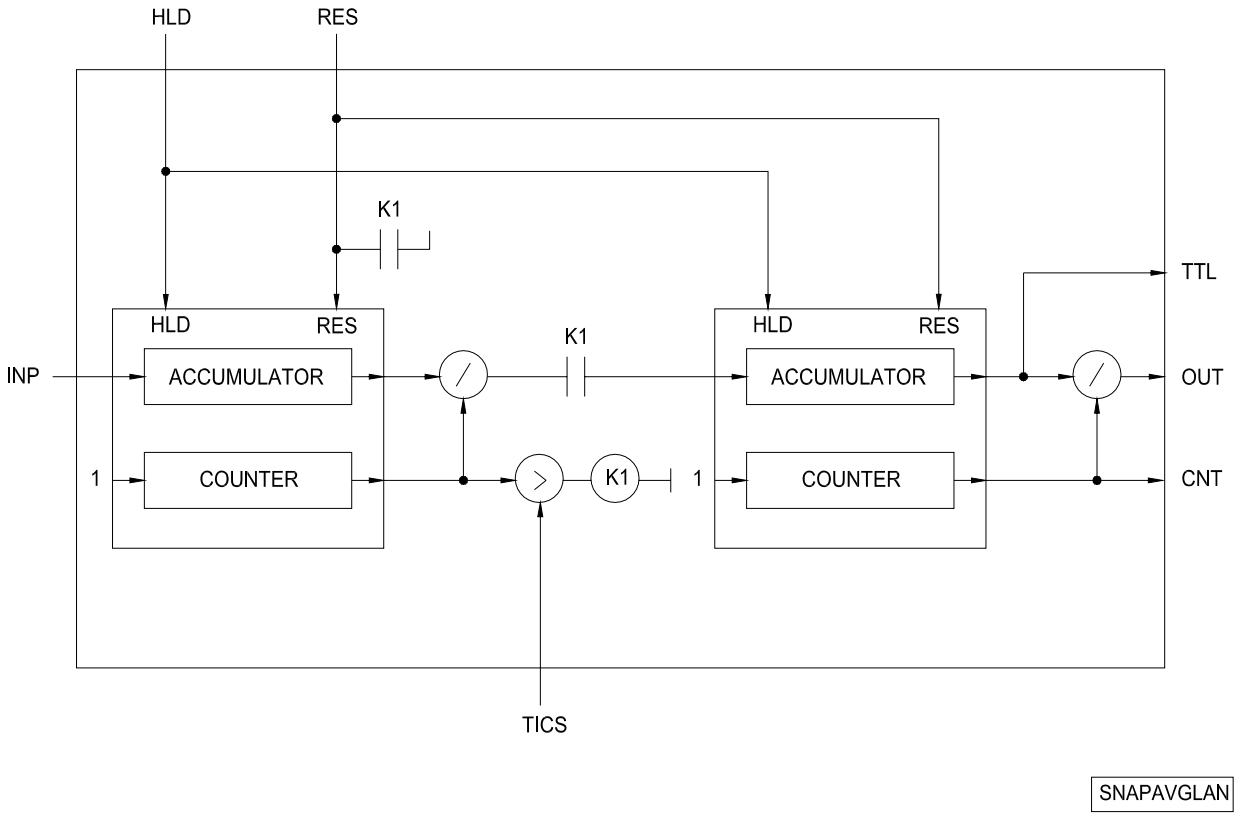


FIGURE 4-67. SNAPAVG BLOCK

4.68 SPLICER

The Splicer block calculates two analog pulse count values, one for the pressure roll and one for the knife. Each value equals the amount of pulses to be received from the oncoming roll pulse generator before enabling the pressure roll and knife firing outputs.

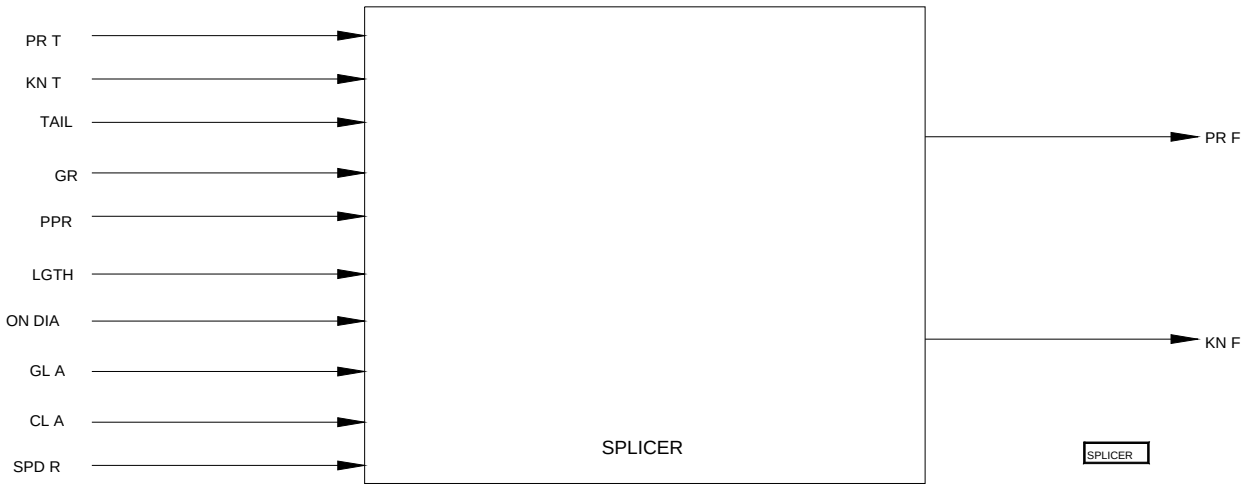


FIGURE 4-68. SPLICER BLOCK

1. Inputs

PR T:	Analog	Pressure Roll Reaction Time (Sec)
KN T:	Analog	Knife Reaction Time (Sec)
TAIL:	Analog	Tail Length (Inches)
GR:	Analog	Gear Ratio for Oncoming Roll
PPR:	Analog	Tach PPR for Oncoming Roll
LGTH:	Analog	Static Length (Inches)
ON DIA:	Analog	Oncoming Roll Diameter (Inches)
GL A:	Analog	Glue Line Angle (Degrees)
CL A:	Analog	Clearance Angle (Degrees)
SPD R:	Analog	Speed Reference (FPM)

2. Outputs

PR F:	Analog	Pressure Roll Pulse Count Preset
KN F:	Analog	Knife Pulse Count Preset

3. Implementation

$K = (PPR)(GR)$

$$RPS = \frac{(SPD\ R)}{5\pi\ (ON\ DIA)}$$

$PPS = K\ (RPS)$

$ANGLE = (RPS)(PR\ T)(360^{o})$

$TMP = (GL\ A) - ANGLE$

If $TMP < 0.1$, $TMP = TMP + 360^{o}$

Else, $TMP = TMP + (CL\ A)$

$$PR\ F = K\left(\frac{TMP}{360^{\circ}}\right)$$ Pressure Roll Firing Count

$TMP = PR\ F$

$TMP = TMP + [(PPS)(PR\ T)]$

$$TMP = TMP + K\ \left(1 - \frac{CL\ A}{360^{\circ}}\right)$$

$TMP = TMP - (PPS)(KN\ T)$

$$TMP2 = K\left[\frac{TAIL - LGTH}{\pi\ (ON\ DIA)}\right]$$

$TMP = TMP + TMP2$

If $TMP < 0$, $TMP = TMP + K$

$PR\ F = (PR\ F) + K$

Else, $KN\ F = TMP$ Knife Firing Count

4.69 SUMMING JUNCTION, SELECTABLE 3 INPUT

This block adds input signals such as speed reference and tension trim. It selects which inputs to add using the bit select lines.

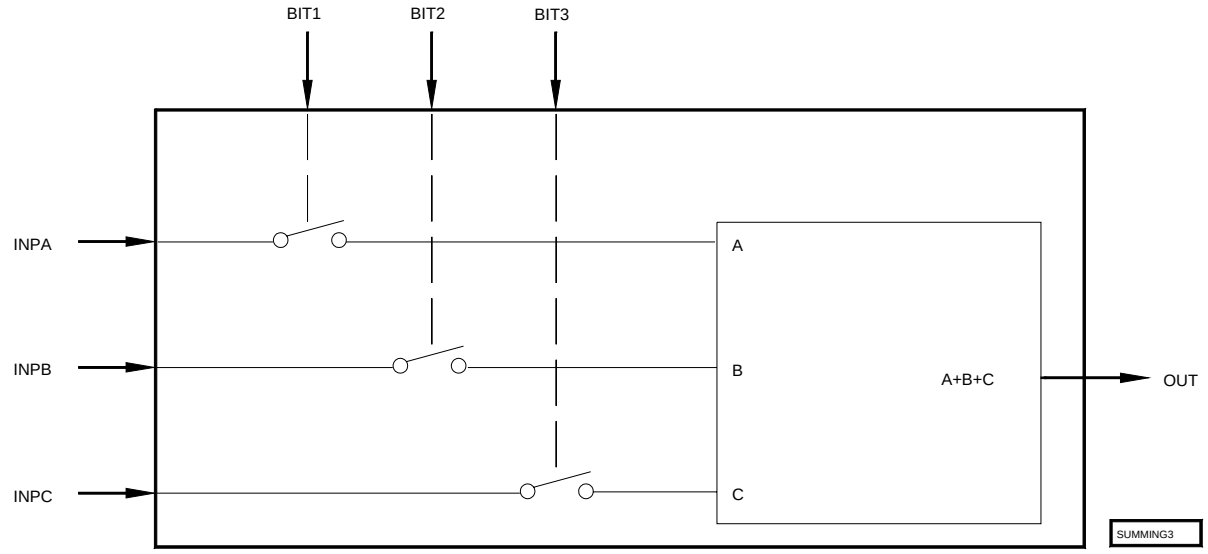


FIGURE 4-69. SUMMING BLOCK

1. Inputs

INPA:	Analog
INPB:	Analog
INPC:	Analog
BIT1:	Bit
BIT2:	Bit
BIT3:	Bit

2. Outputs

OUT:	Analog

3. Implementation

OUT = [(INPA × BIT1) + (INPB × BIT2) + (INPC × BIT3)]

If all 3 bits are low, then OUT = 0.

4.70 TABLE

The Table block is used to modify an analog variable "INP" by a factor which is proportional to a second analog variable "X_IN". An example of such an application would be to taper a tension setpoint based on roll diameter.

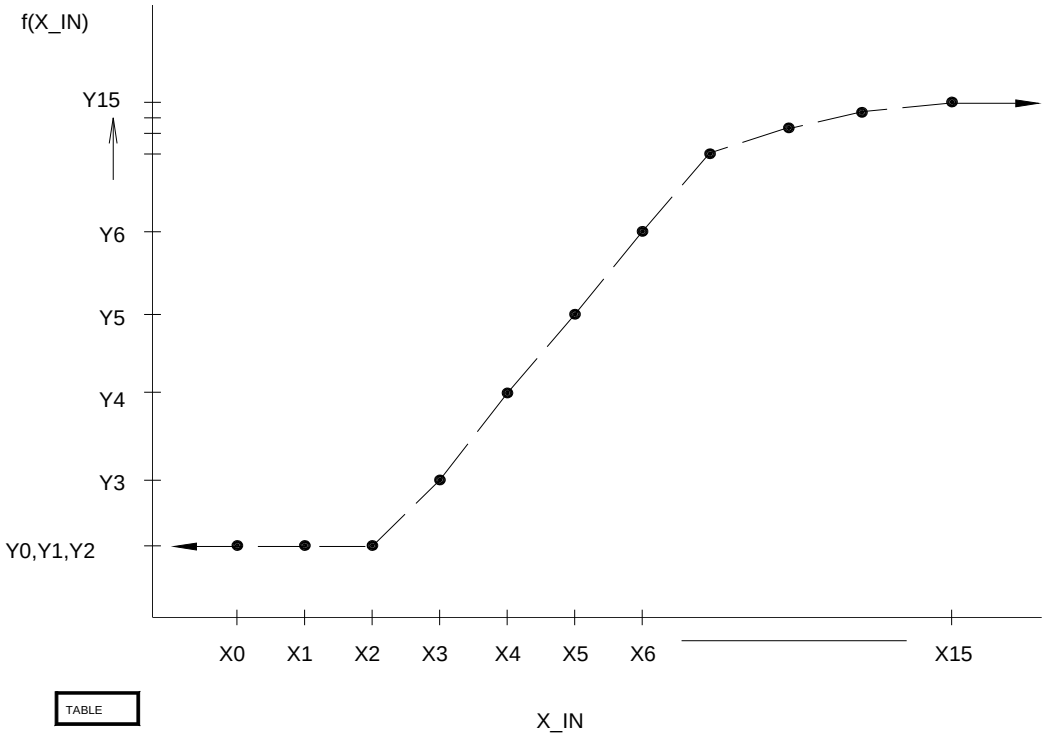


FIGURE 4-70. TABLE BLOCK

1. Inputs

INP:	Analog
X_IN:	Analog
GAIN:	Analog
TABLE:	0 - 3

2. Outputs

OUT:	Analog

3. Implementation

The Table block is used to perform a non-linear look up operation y = f(X_IN), and output a value using the following formula: OUT = f(X_IN) × GAIN × INP.

Where:

Y is calculated as a function of X_IN by using the selected TABLE values.

A TABLE is defined as an array (2 × 16) in size, of x,y points where x values are entered in ascending order. (Refer to section 6.3.4, Tables Menu, for additional explanation.)

If X_IN is less than the first X value in the TABLE, then f(X_IN) = the first y_0 value.

If X_IN is greater than the last X value in the TABLE, then f(X_IN) = the y_{15} point.

If X_IN equals an X value in the TABLE, then f(X_IN) = the y value associated with that X point.

If X_IN falls between two X values in the TABLE, then f(X_IN) is the interpolated between the two associated y values.

4.71 TACH SELECT AND TACH SELECT-W

The blocks TACH SELECT and TACH SELECT-W are almost identical and behave as described below. The difference between the two blocks is described in the SPECIAL NOTE.

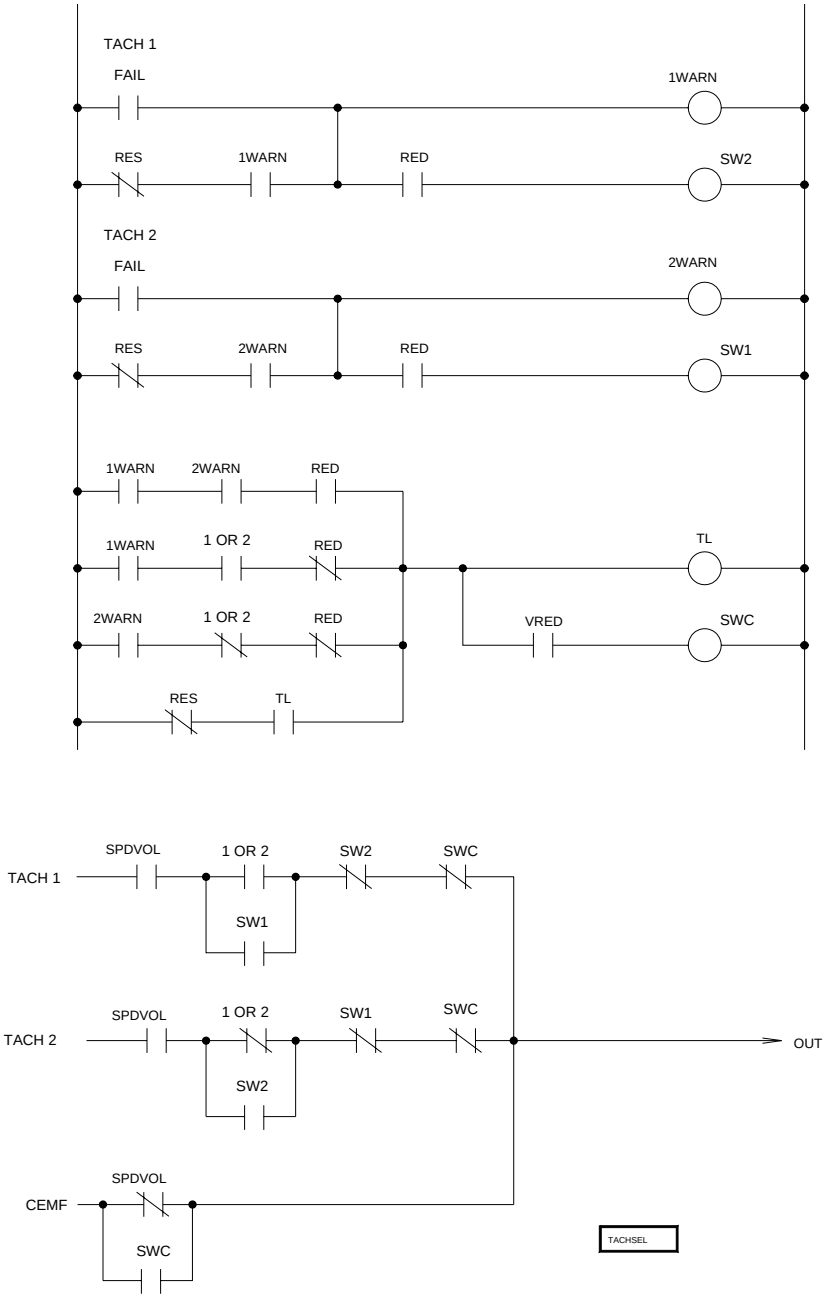


FIGURE 4-71. TACH SELECT AND TACH SELECT-W BLOCK

This block selects the type of speed feedback to be used by the speed loop. Either CEMF or speed feedback may be employed. When tach feedback is selected, tach loss detection can be used by configuring Y***:USR FAULT 1 = TACH LOSS.

This block also has the option of using redundant tach inputs. On a tach loss of the primary tach, it can be specified to switch to a second tach feedback while still running the drive. If the tachs fail, CEMF feedback can be selected for the speed feedback. The tach inputs can be configured to either analog or digital feedbacks.

1. Inputs

TACH1:	Analog
TACH2:	Analog
CEMF:	Analog
MXSPD:	Analog
GAIN:	Analog
SPDVOL:	Bit
1 OR 2:	Bit
V RED:	Bit
RED:	Bit
RES:	Bit

2. Outputs

OUT:	Analog
1WARN:	Bit
2WARN:	Bit
TL:	Bit

3. Implementation

The speed output of the TACH SELECT block is selected from CEMF, TACH1, or TACH2 inputs, and scaled by the GAIN input. In order to work properly, the CEMF, TACH1, and TACH2 inputs need to be properly scaled into the appropriate units. If TACH feedback is scaled in FPM or other process units, the motor's CEMF input will need to be scaled into the identical units. Note: If frequency inputs are used as TACH input, Section III gives details on how the scaling parameters need to be set in order to get the proper units. The GAIN input can be used to scale all inputs into process units that will match the Speed Reference. Typical operation of the TACH SELECT block is described below.

If SPDVOL = low, OUT = CEMF × GAIN.

If SPDVOL = high and 1 OR 2 = high, and no tach loss is detected, OUT = TACH1 × GAIN.

If SPDVOL = high and 1 OR 2 = low, and no tach loss is detected, OUT = TACH2 × GAIN.

Tach failure conditions are detected by comparing CEMF to TACH inputs as follows:

If CEMF × GAIN > 0.20 × MXSPD and
TACH × GAIN < 0.05 × MXSPD,

or If CEMF × GAIN >0.20 and the polarity of CEMF <> the polarity of TACH,

then tach failure will be detected regardless of whether the associated tach input is selected or not. Therefore, if CEMF feedback is used as the speed feedback, TACH LOSS should not be configured to a USR FAULT. Any time a tach failure is detected, the associated WARN bit will turn on. (That is, if a TACH1 failure is detected, the 1WARN bit turns on; if a TACH2 failure is detected, the 2WARN bit is turned on.)

If tach feedback is used (SPDVOL = high), and if a tach failure is detected, then the TACH SELECT block can be configured to automatically switch to the redundant tach or to control in CEMF. The logic is described in Figure 4-68 and as follows:

If SPDVOL = high and RED = high, then if a tach failure is detected, the output will switch from the failed tach to the redundant tach. For example, if 1 OR 2 = high, and the TACH1 input fails, OUT will switch from TACH1 × GAIN to TACH2 × GAIN.

If SPDVOL = high and V RED = high, then if a tach failure is detected, the output will switch from the failed tach feedback to CEMF feedback. For example, if 1 OR 2 = high, and the TACH1 input fails, OUT will switch from TACH1 × GAIN to CEMF × GAIN.

If SPDVOL = high and both RED and V RED = high, then if a tach failure occurs, the TACH SELECT block will switch first to the redundant tach; if the redundant tach fails, the output will switch to CEMF feedback.

****

W A R N I N G

If SPDVOL = high and no redundancy is selected (both RED and V RED = low), then if a tach failure is detected in the primary tach, the TL bit will turn on, and the output will not switch from the failed tach. Similarly, if RED = high, but V RED = low, and if both tachs fail, the TL bit will turn on and the output will not switch but will remain dependent on the failed tach. In such a situation, a tach failure could cause equipment to "run away" and damage machinery or injure personnel. For this reason, any time V RED = low and SPDVOL = high, Avtron HIGHLY RECOMMENDS configuring one of the USR FAULTS (described in Section III) to the TL output of the TACH SELECT block.

****

Any time a tach failure is detected, the TACH SELECT block will set appropriate output bits, and switch speed feedback as the logic requires. The outputs will remain in their states until the RES turns on. If the RES input is maintained in the high state, the TACH SELECT block will not switch to any redundant modes, and tach failures will not be detected.

SPECIAL NOTE:

The TACH SELECT block differs from the TACH SELECT-W block in how the GAIN input is used to scale input feedback. The TACH SELECT block scales all inputs (TACH1, TACH2, and CEMF) and compares the scaled inputs to the MXSPD input to determine tach failures (i.e., tach fail is sensed if CEMF × GAIN > 0.2 × MXSPD and TACH × GAIN< 0.05 × MXSPD). The TACH SELECT-W block compares the unscaled inputs to MXSPD in determining tach failures (i.e., tach fail is sensed if CEMF > 0.2 × MXSPD and TACH < 0.05 × MXSPD). All other block operation and logic are identical.

4.72 TIMER

The Timer block is used to delay setting a bit until an appropriate count has occurred. It can be used to buffer faults for FIFO logging or to shut the drive down if in maximum tension for a period of time.

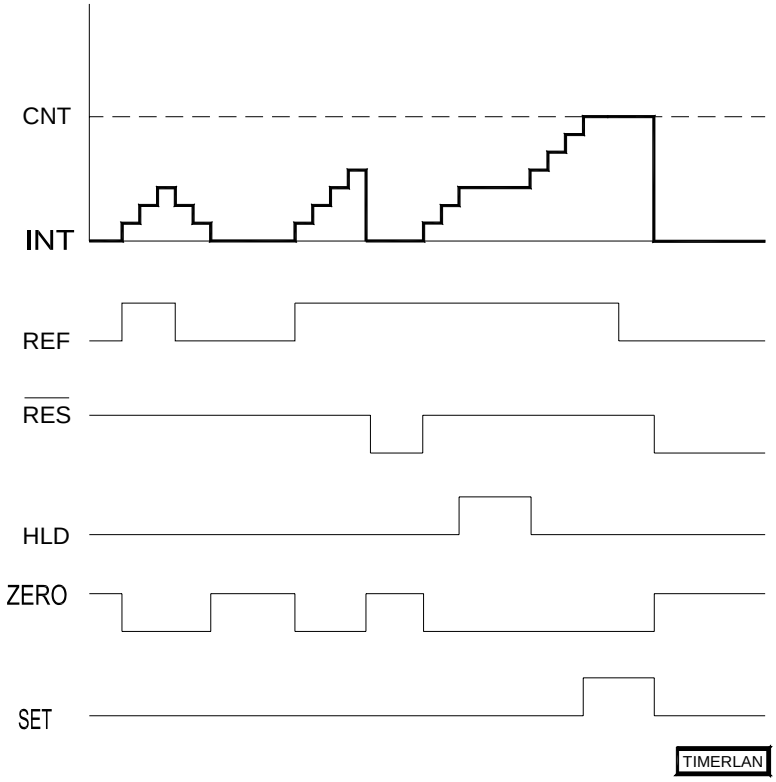


FIGURE 4-72. TIMER BLOCK

1. Inputs

HLD:	Bit
RES:	Bit
REF:	Bit
RET:	Bit
CNT:	Analog (1 - 1,000,000)

2. Outputs

SET:	Bit
ZERO:	Bit
VAL:	Analog

3. Implementation

When the RES bit equals zero, VAL is set to 0.

When VAL = 0, then ZERO bit = 1.
If VAL ≠ 0, then ZERO bit = 0.

When the HLD bit is high, VAL is frozen. The RES bit has higher priority than the HLD bit.

If RES = 1 and HLD = 0, then the following occurs:

If REF = 1 and VAL < CNT, then VAL = VAL + 1.
If REF = 0 and VAL > 0, then VAL = VAL - 1.
If VAL = CNT, then SET = 1, else SET = 0.
When SET = 1, it latches until a reset occurs.

Non-retentive Block

On powerup of the ADDvantage-32, VAL = 0

Retentive Block

On powerup of the ADDvantage-32, VAL will be initialized under the following conditions:

If RET = 0, then VAL = 0

If RET = 1, VAL is set to its last value prior to power loss. VAL must be configured to a retentive point Y***:RET SETPT* to be updated automatically on powerup.

4.73 TYPE 2 DIA

This block calculates the roll diameter of a center driven winder or unwinder section by using operator entered material thickness value and counting the number of spindle revolutions. This diameter calculation block can be used instead of the RATIO block. It is as accurate as the value of THCK that is entered. THCK is the average thickness of the material.

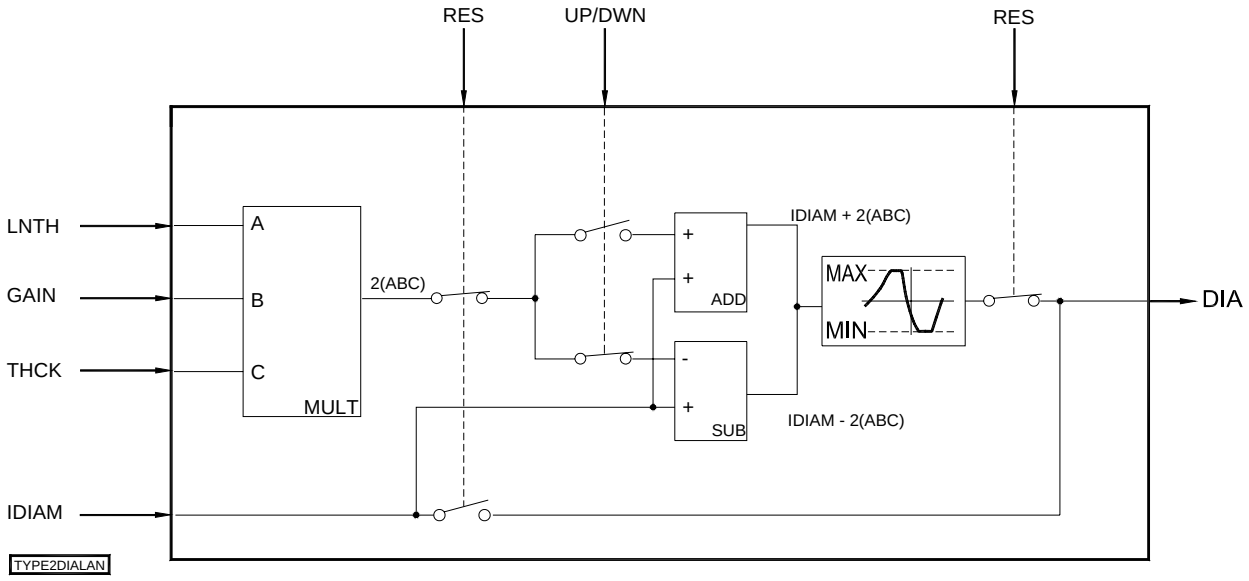


FIGURE 4-73. TYPE 2 DIA BLOCK

1. Inputs

THCK:	Analog
IDIAM:	Analog
LNTH:	Analog
MAX:	Analog
MIN:	Analog
GAIN:	Analog
UP/DWN:	Bit
RES:	Bit

2. Outputs

DIA:	Analog

3. Implementation

If RES is high, then the output DIA = IDIAM.

If DIA is calculated to go beyond the MAX diameter or MIN diameter limits, then DIA will be clamped at the limit.

When the RES bit is set low, the DIA is calculated by the following:

If UP/DWN is high, then:
DIA = (LNTH/GAIN × 2 × THCK) + IDIAM

If UP/DWN is low, then:
DIA = IDIAM - (LNTH/GAIN × 2 × THCK)

Length needs to be configured to the frequency counter from the winder tach. The counter needs to be reset at new roll.

Length divided by Gain should be equal to roll revolutions for the block to work properly.

4.74 TYPE 3 DIA

This block calculates the roll diameter of a winder by counting the footage of the surface roll section. The accuracy of this block is comparable to the value of THICK + DIA. (THICK is the average thickness of the material.)

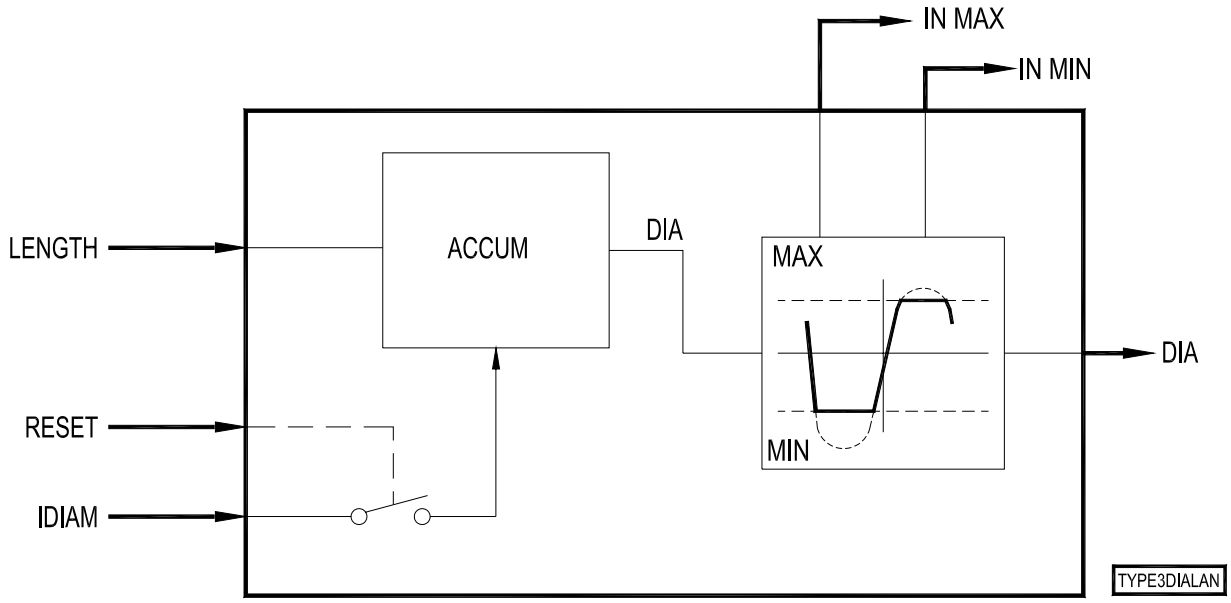


FIGURE 4-74. TYPE 3 DIA BLOCK

1. Inputs

THICK:	Analog
IDIAM:	Analog
LENGTH:	Analog
MAX DIA:	Analog
MIN DIA:	Analog
UP/DWN:	Bit
RESET:	Bit
RET:	Bit

2. Outputs

DIA:	Analog
IN MAX:	Bit
IN MIN:	Bit

3. Implementation

If RESET is set high, then DIA = IDIAM.

If DIA is calculated to exceed either the MAX DIA or MIN DIA limits, then DIA will be clamped at the limit and its corresponding output bit will go high.

For a winder system, set the UP/DWN bit high. The DIA increases by 2X THICK each time the following equation is true.

$$\text{LENGTH (New)} > \pi \times \text{DIA} + \text{Length (Old)}$$

For an unwinder system, set the UP/DWN bit low. DIA increases by 2X THICK each time the following equation is true.

$$\text{LENGTH (New)} > \pi \times \text{DIA} + \text{Length (Old)}$$

NOTE

LENGTH = Footage value of material from tach counters.

THICK = Average thickness of material (Entered in process units).

Non-retentive Block

On powerup of the ADDvantage-32, DIA = IDIAM

Retentive Block

On powerup of the ADDvantage-32, DIA will be initialized under the following conditions:

If RET = 0, then DIA = IDIAM

If RET = 1, DIA is set to its last value prior to power loss. DIA must be configured to a retentive point Y***:RET SETPT* to be updated automatically on powerup.

4.75 UNITY SCALE

The Unity Scale block rescales the minimum and maximum limits of a given value to 0-1.

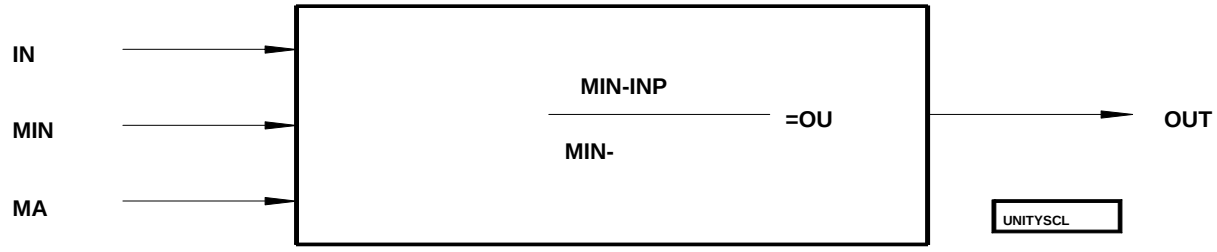


FIGURE 4-75. UNITY SCALE BLOCK

1. Inputs

   INP: Analog
   MIN: Analog
   MAX: Analog

2. Outputs

   OUT: Analog

3. Implementation

   If MIN - MAX ≠ 0, then OUT = MIN - INP/MIN - MAX.
   If MIN - MAX = 0, then OUT = 0.

4.76 UV PROTECT

The block can be used to try to prevent a shoot through condition on a low AC voltage. It does this by disabling the regenerating bridge current limit in a low voltage condition.

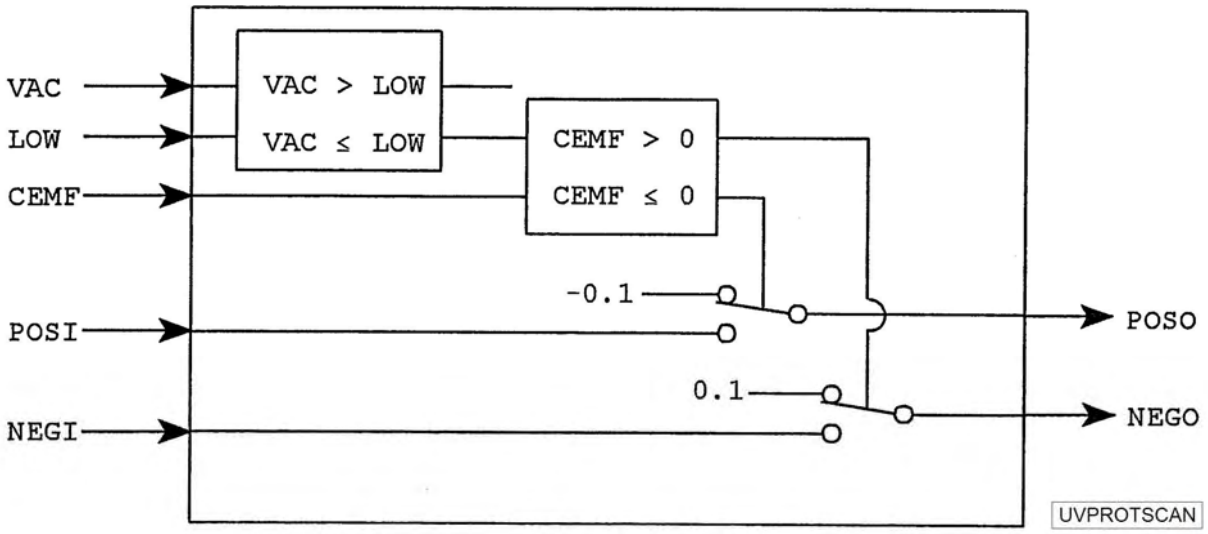


FIGURE 4-76. UV PROTECT BLOCK

1. Inputs

POSI:	Analog
NEGI:	Analog
VAC:	Analog
LOW:	Analog
CEMF:	Analog

2. Outputs

POSO:	Analog
NEGO:	Analog

3. Implementation

If VAC is greater than the LOW input, then POSO = POSI and NEGO = NEGI.

Otherwise a low line condition is present and the following occurs.

If CEMF is Positive
POSO = POSI
NEGO = 0.1%

If CEMF is Negative
POSO = -0.1%
NEGO = NEGI

In the application, execute the block between the current limit setpoints and the current loop limit to protect the drive.

4.77 WINDER WK/D

This block calculates the inertia of a winder/unwinder. The inertia of the material which is essentially a hollow cylinder is equal to WK2 = const × width × ((dia^4) - (core dia^4)). Therefore, total inertia of a roll of material is the result of the previous equation plus the fixed inertia.

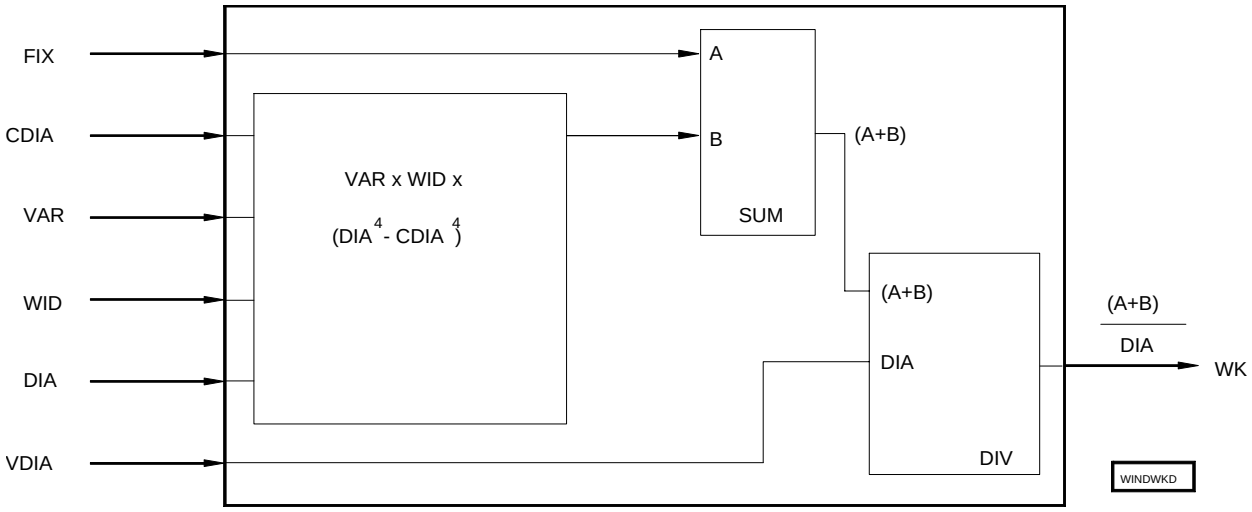


FIGURE 4-77. WINDER WK/D BLOCK

1. Inputs

VAR:	Analog
WID:	Analog
DIA:	Analog
FIX:	Analog
CDIA:	Analog
VDIA:	Analog

2. Outputs

WK: Analog

3. Implementation

$$WK = \frac{(VAR)(WID)(DIA^4 - CDIA^4) + FIX}{VDIA}$$

The torque required to accelerate a roll is equal to the WK^2 of the body times the change in RPM.

$$\text{Torque} = WK^2 \times \frac{\text{Change in RPM}}{\text{Time}} \qquad \text{(For a Fixed Diameter)}$$

$$\text{Torque} = WK^2 \times \frac{\text{Change in FPM}}{\text{Time}} \qquad \text{(For a Variable Diameter)}$$

The D/DT block is used for $\frac{\text{Change in FPM}}{\text{Time}}$.

Therefore, the amount of current needed to accelerate a roll should equal the output of the WK/D block times the output of the D/DT block. This should be calculated for a core first (DIA = CDIA) and then a full roll of material (CDIA = MAXIMUM DIAMETER).

4.78 WINDOW COMPARE

The Window Compare takes an input and checks to see if it is between a high and low setpoint. The block offers options for absolute value of the input and a complementary output bit.

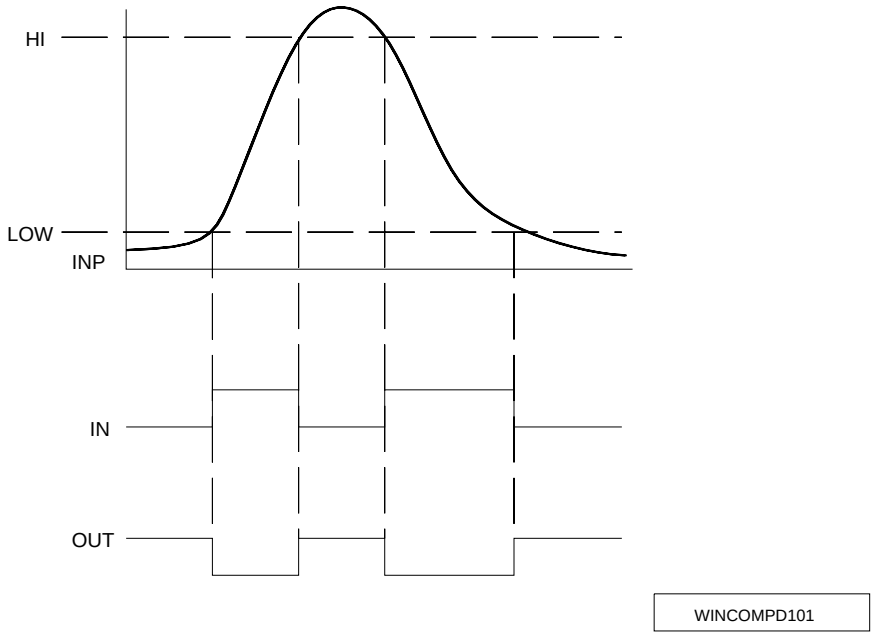


FIGURE 4-78. WINDOW COMPARE BLOCK

1. Inputs

INP:	Analog
HI:	Analog
LOW:	Analog
ABS:	Digital

2. Outputs

IN:	Digital
OUT:	Digital

3. Implementation

If ABS is Low (0), then

IN = 1 if (LOW <= INP <= HI)

Else IN = 0.

If ABS is High (1), then

IN = 1 if (|LOW| <= |INP| <= |HI|)

Else IN = 0.

If IN = 1 OUT = 0

Else IN = 0 OUT = 1

SECTION V

SIGNAL ANALYZER

The signal analyzer is a four-channel memory recorder capable of recording both digital and analog information. Each channel has separate setup and trigger parameters, allowing any information in the analog and digital tables to be recorded. After the information is captured, it can be directed to an analog output, displayed on the LCD display or uploaded to the human interface module.

Setup of the signal analyzer is accomplished using the Z*** parameters. These parameters consist of both calibration and configuration parameters which perform the following functions:

1. Data Collection
2. Channel Triggering
3. Sample Rate
4. Preview
5. Arming
6. Channel Enable
7. Channel Reset

The signal analyzer has 16K bytes of RAM. Each data point requires 4 bytes of RAM, allowing 1,000 points of information to be stored for each channel. If power is lost, all information captured by the analyzer is lost.

Specific parameter numbers have been omitted and replaced with asterisks due to differing parameter numbers between software. Consult Appendices A-D for exact parameter numbers and descriptions.

5.1 DATA COLLECTION

The data stored in the recorder channels can be any analog information or digital bit information located in the data tables. The recorder must be disabled to change the parameter. Defaults:

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:INP1 A/D	Q***:ANALOG
	Z***:INP1 ANALG	A***:G FLD I CMD
	Z***:INP1 DIG	D***:I-LIM LED
#2	Z***:INP2 A/D	Q***:ANALOG
	Z***:INP2 ANALG	A***:MFLD I
	Z***:INP2 DIG	D***:RUNX
#3	Z***:INP3 A/D	Q***:ANALOG
	Z***:INP3 ANALG	A***:ACT SPEED
	Z***:INP3 DIG	D***:FWD BR LED
#4	Z***:INP4 A/D	Q***:ANALOG
	Z***:INP4 ANALG	A***:GFLD I
	Z***:INP4 DIG	D***:REV BR LED

5.2 TRIGGER

Set for a rising edge or falling edge trigger, the recorder can be triggered from any analog or digital bit information located in the data tables. The recorder must be disabled to change these parameters. Default for all four recorders:

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:TRIG1 A/D	Q***:ANALOG
	Z***:TRIG1 ANALOG	A***:ACT SPEED
	Z***:TRIG1 DIG	D***:FAULT
	Z***:TRIG R/F 1	Q***:RISING
	Z***:TRIG LEVEL1	0.5
#2	Z***:TRIG2 A/D	Q***:ANALOG
	Z***:TRIG2 ANALOG	A***:ACT SPEED
	Z***:TRIG2 DIG	D***:FAULT
	Z***:TRIG R/F 2	Q***:RISING
	Z***:TRIG LEVEL2	0.5
#3	Z***:TRIG3 A/D	Q***:ANALOG
	Z***:TRIG3 ANALOG	A***:ACT SPEED
	Z***:TRIG3 DIG	D***:FAULT
	Z***:TRIG R/F 3	Q***:RISING
	Z***:TRIG LEVEL3	0.5

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#4	Z***:TRIG4 A/D	Q***:ANALOG
	Z***:TRIG4 ANALOG	A***:ACT SPEED
	Z***:TRIG4 DIG	D***:FAULT
	Z***:TRIG R/F 4	Q***:RISING
	Z***:TRIG LEVEL4	0.5

5.3 SAMPLE RATE

The sample rate determines the time span between the recording of each data point. The rate is determined by specifying the number of 11.11 msec time increments required between samples. For example, the shortest and longest sample times are:

Sample Time = Z***:RATE X * 11.11 msec
Range of Z***:RATE X = (1 to 99,999)

Shortest Sample Time = (1) * 11.11 msec = 11.11 msec

Longest Sample Time = (99,999) * 11.11 msec = 1111 sec

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE	RANGE
#1	Z***:RATE 1	18 (200.00 ms).	1-99,999
#2	Z***:RATE 2	18	1-99,999
#3	Z***:RATE 3	18	1-99,999
#4	Z***:RATE 4	18	1-99,999

5.4 PREVIEW

Information recorded by the signal analyzer channels is broken into 1,000 individual points. When a particular channel is triggered, a preview can be set to show a number of samples before the trigger occurred. This allows information leading up to the trigger point to be evaluated. For example, if the preview is set at 500, then 500 samples will be recorded before the trigger point and 500 after.

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE	RANGE
#1	Z***:PREVIEW 1	500	1-999
#2	Z***:PREVIEW 2	500	1-999
#3	Z***:PREVIEW 3	500	1-999
#4	Z***:PREVIEW 4	500	1-999

5.5 ARMING

The channel must be armed before it can start recording data. Defined as a digital bit in the data table, it can be defined to arm on a high or low bit. The arming bit can be defined as the one bit allowing the logic analyzer to be turned on or off using only the keyboard. Default for all four recorders:

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:ARM BIT 1	D***:ONE BIT
#1	Z***:ARM LEVEL 1	Q***:HIGH
#2	Z***:ARM BIT 2	D***:ONE BIT
#2	Z***:ARM LEVEL 2	Q***:HIGH
#3	Z***:ARM BIT 3	D***:ONE BIT
#3	Z***:ARM LEVEL 3	Q***:HIGH
#4	Z***:ARM BIT 4	D***:ONE BIT
#4	Z***:ARM LEVEL 4	Q***:HIGH

5.6 ENABLE

This bit enables or disables the recorder. The operator must disable the recorder for adjustments such as trigger level or sample rate. Defaults are as follows:

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:ENABLE REC1	Q***:DISABLED
#2	Z***:ENABLE REC2	Q***:DISABLED
#3	Z***:ENABLE REC3	Q***:DISABLED
#4	Z***:ENABLE REC4	Q***:DISABLED

5.7 RESET

Defined as a bit in the digital table which clears the done bit, it can be defined to reset on a high or low level. The recorder can also be reset by a keyboard bit. Default for all four recorders:

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:RESET BIT 1	D***:OUT DONE1
#1	Z***:RESET R/F 1	Q***:RISING
#2	Z***:RESET BIT 2	D***:OUT DONE2
#2	Z***:RESET R/F 2	Q***:RISING
#3	Z***:RESET BIT 3	D***:OUT DONE3
#3	Z***:RESET R/F 3	Q***:RISING
#4	Z***:RESET BIT 4	D***:OUT DONE4
#4	Z***:RESET R/F 4	Q***:RISING

5.8 DONE BIT

The done bit is high when the recorder has been armed, trigger activated, and the events stored. It stays high until the recorder has been disabled or reset. This bit can be used to relate to the human interface computer that recording is completed and data is ready to be uploaded.

RECORDER CHANNEL	CONFIGURATION PARAMETER
#1	D***:DONE REC1
#2	D***:DONE REC2
#3	D***:DONE REC3
#4	D***:DONE REC4

5.9 ANALOG OUT

The stored data can be output to an analog out. To do this the analog output must be configured to A***:RECORDER. The output trigger bit can be configured to any digital data point. It can be enabled to output on a rising or falling edge.

The following sequence occurs when the data has been triggered to output:

The arming for the recorder is turned off. The oldest data point is scaled and outputted. It will then output one event every 27.7 ms. After the recording is outputted, the out done bit is set and rearms the recorder.

The out done bit is reset if the recorder is retriggered or if the analog out is selected again.

RECORDER CHANNEL	CONFIGURATION PARAMETER	DEFAULT VALUE
#1	Z***:OUT TRIG 1	D***:ZERO BIT
	Z***:OUT R/F 1	Q***:RISING
	Z***:SPAN OUT 1	1.0
	Z***:OFF OUT 1	0.0
#2	Z***:OUT TRIG 2	D***:ZERO BIT
	Z***:OUT R/F 2	Q***:RISING
	Z***:SPAN OUT 2	1.0
	Z***:OFF OUT 2	0.0
#3	Z***:OUT TRIG 3	D***:ZERO BIT
	Z***:OUT R/F 3	Q***:RISING
	Z***:SPAN OUT 3	1.0
	Z***:OFF OUT 3	0.0
#4	Z***:OUT TRIG 4	D***:ZERO BIT
	Z***:OUT R/F 4	Q***:RISING
	Z***:SPAN OUT 4	1.0
	Z***:OFF OUT 4	0.0

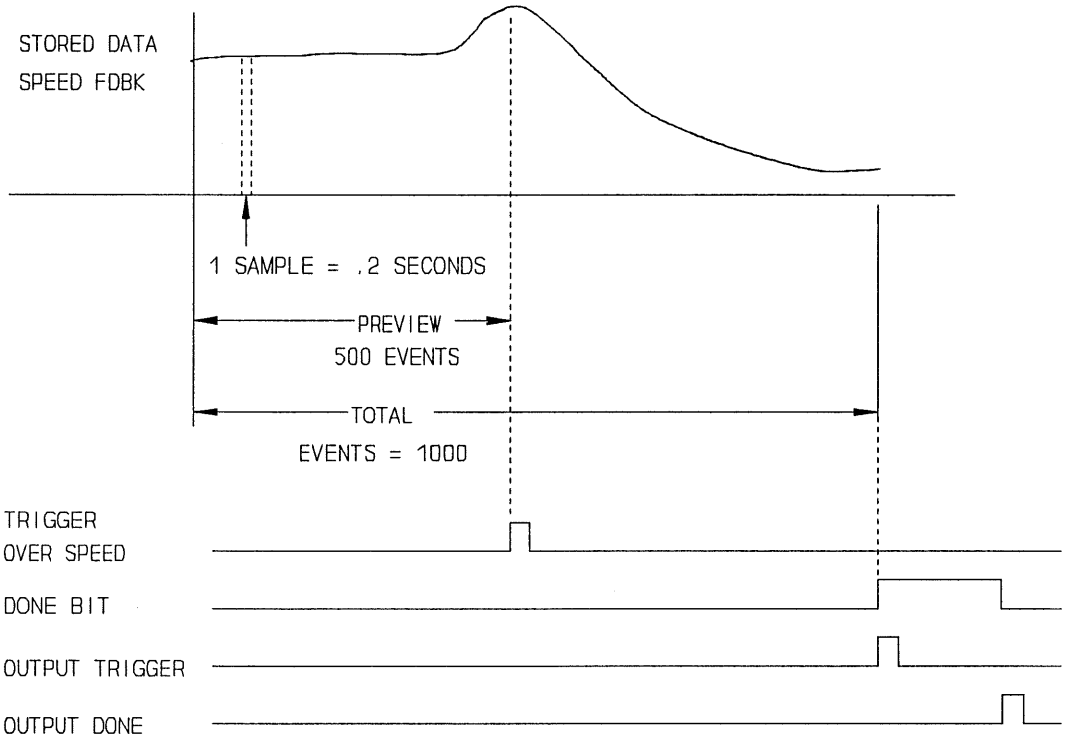


FIGURE 5-1. AUTOMATIC RECORDING OF SIGNAL ANALYZER

Setup

Input Data:	ABS ACT SPD
Trigger:	Rising edge of Fault bit
Sample Rate:	18 (18 X 11.11 ms = .2 seconds/event)
Total Time:	1,000 events X .2 seconds = 200 seconds
Preview:	500 events
Output:	Rising edge of Done bit
Reset:	Rising edge of Output Done bit
Arm:	Run Enable
Enabled:	One bit

Summary

When triggered by an overspeed, speed feedback recording is completed by the recorder. The done bit is set upon collection of data, and the data is sent to either a computer or an analog output. The output done bit is set upon completion of the output, causing resetting of the channel. The channel is then ready to capture data on the next fault.

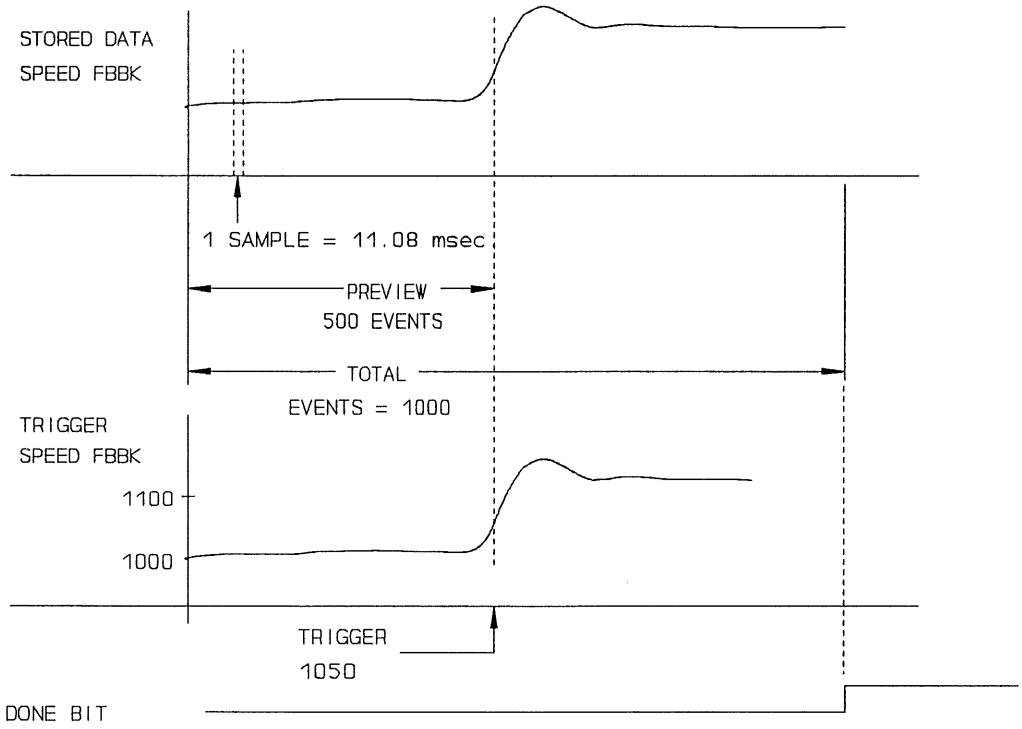


FIGURE 5-2. MANUAL RECORDING OF SIGNAL ANALYZER

Setup

Input Data:	ABS ACT SPD
Trigger:	Rising at 1,050 of ABS ACT SPD
Sample Rate:	1 = 11.11 ms
Total Time:	1,000 events X 11.11 ms = 11 seconds
Preview:	500 events
Output:	Zero bit
Reset:	Zero bit
Arm:	One bit
Enabled:	One bit

Summary

The trigger is set to activate when ABS ACT SP exceeds 1,050. When triggered, data recording is completed. Upon completion, the channel is set to wait but is not reset. The data can be output manually (using the keyboard) or viewed. The channel may also be reset.

SECTION VI

PREPARATION FOR USE

Preparation for use of the ADDvantage-32 Advanced Control Module includes unpacking, electrical interconnections, wiring instructions, and installation.

6.1 UNPACKING

Inspect all shipping containers for exterior damage. Notify the carrier of any damage detected so that the carrier representative may inspect the equipment and shipping containers.

When removing equipment from shipping containers, check carefully that loose items such as connectors, nomenclature plates, cables, manuals, etc., are removed before discarding packing materials. (Internal modules and connectors should be checked for proper seating.)

If the ADD-32 will not be installed right away, refer to Storage and Installation Specifications in Section 6.3 for storage instructions.

6.2 MECHANICAL INSTALLATION AND SPECIFICATIONS

Outline dimension drawings for the drive are found in the supplemental drawing manual and provide mounting dimensions for physical installation as well as electrical specifications, enclosure size specifications, grounding requirements, and wiring information. Recommendations for pulse generator coupling and timing devices are provided in the Avtron Rotary Pulse Generator Installation Instructions. Proper mounting of the pulse generator (speed feedback device) is critical for achieving optimum performance and reliability.

****

W A R N I N G

WHEN CUTTING OR DRILLING INTO ANY ENCLOSURE HOUSING, ALWAYS MAKE SURE THAT METAL SHAVINGS DO NOT DROP ONTO THE COMPONENTS. AVTRON TAPES LARGE SHEETS OF PLASTIC TO THE PANEL AROUND THE COMPONENTS WHEN METAL SHAVINGS ARE LIKELY TO BE PRODUCED. WE RECOMMEND THE SAME TO INSTALLERS OF OUR EQUIPMENT.

****

TABLE 6-1. ADD-32 MECHANICAL SPECIFICATIONS

DRIVE PART NUMBER	WEIGHT	HEIGHT	WIDTH	DEPTH
DC0000	35 lbs.	17.25"	11.25"	6.00"

6.3 STORAGE AND INSTALLATION SPECIFICATIONS

If the ADD-32 is to be stored for an extended period of time prior to its installation, the storage temperature should not fall below -20°C and noncondensing humidity should be no more than 95%. Installation conditions require an ambient temperature of 0-40°C when enclosed. Refer to the outline drawing provided in the drawing package for enclosure size and cooling specifications. No derating is needed at altitudes between 0-3300 feet above sea level.

6.4 MOUNTING SPECIFICATIONS

All ADDvantage-32 models should be mounted vertically against a flat panel to allow necessary cooling.

6.5 LOCATION

The equipment should be located where the operating ambient temperature does not exceed 50°C within an enclosure and where excessive shock and vibration are not transmitted to the units. The equipment should be isolated from or protected against such adverse environmental conditons as oil or water spray, and corrosive or conductive air contaminants. Ideally,

the equipment should be installed in an air-conditioned control room. Refer to appropriate outline drawing for specific enclosure and ambient temperature restrictions.

Location of Other Equipment

The ADD-32 ACM should be installed with sufficient distance from other equipment so that the door may be opened during installation, maintenance, and servicing without interfering with other equipment.

6.6 ELECTRICAL CONNECTIONS

****

W A R N I N G

DO NOT OPERATE RADIO TRANSMITTERS or CELL PHONES IN THE VICINITY OF THE ADD-32. The ADD-32 is an electronic device. Although it is designed to operate reliably in typical industrial environments, the ADD-32 can be affected by radio and/or cell phone transmitters. It is possible to cause drive faults, inappropriate/unintended drive I/O activity, and unpredictable operation that could result in damage to the ADD-32, damage to other equipment, or serious injury to personnel.

Radio transmitter interference is a site specific phenomena. Generally, electrical wires connected to terminals on the ADD-32 are the conduits for radio interference. Interference can be minimized by good wiring design and installation practice. It is recommended that signs be posted in and around the drive system, warning of the possibility of interference if the drive is in operation. DO NOT USE radio transmitters or cell phones in the area.

Absence of a radio interference problem is no guarantee that a problem will never occur as conditions and environments can change.

****

Use the following information for assistance in wiring the ADDvantage-32 in accordance with specifications set forth by the National Electrical Code. Always review local building and electrical codes before making electrical connections to this device. Refer to the appropriate outline drawing and rating nameplate for power requirements, fault current interrupt rating, and control power output rating. Wire units using copper wire with a minimum of 60°C insulation.

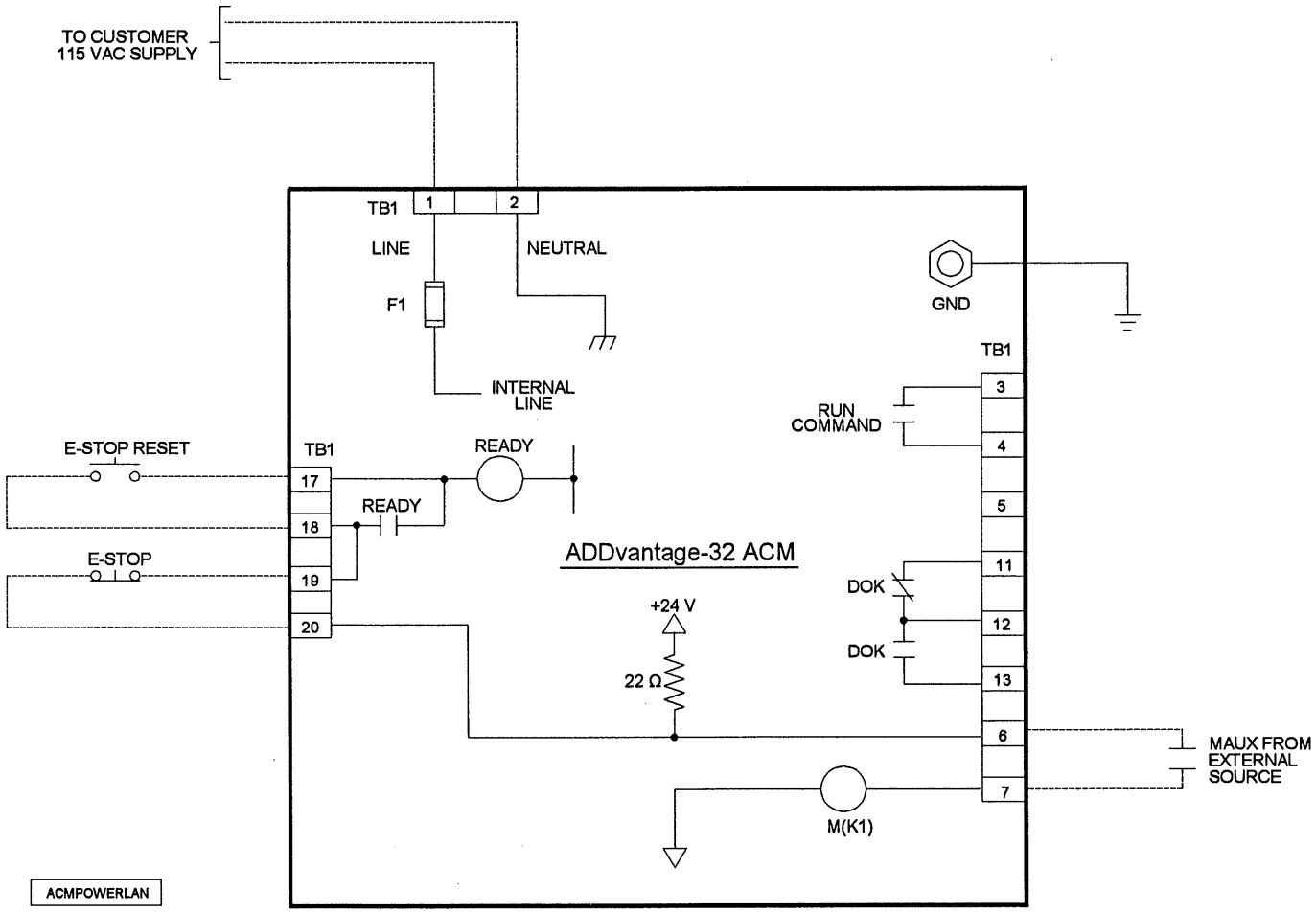


FIGURE 6-1. BLOCK DIAGRAM, POWER CONNECTIONS

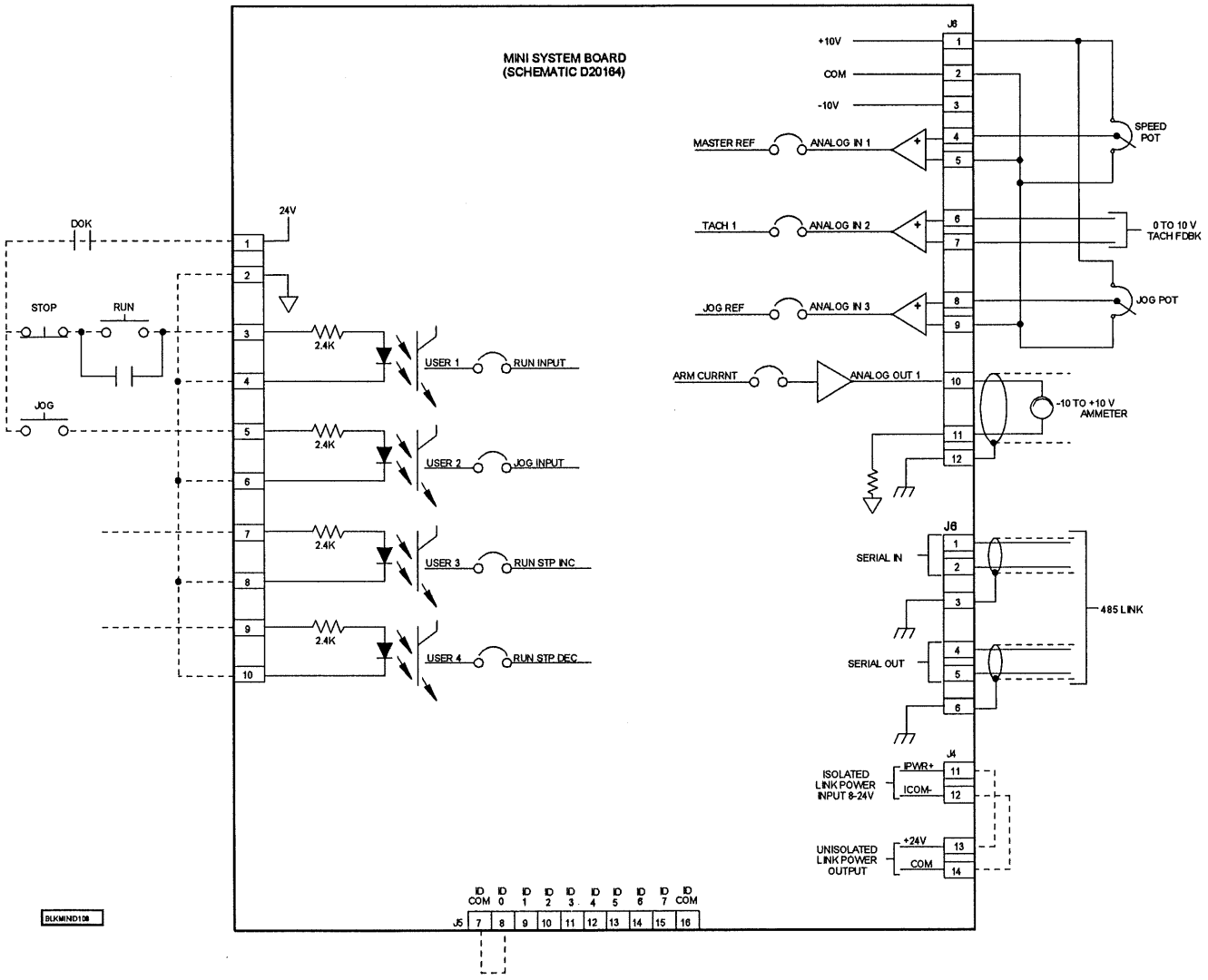


FIGURE 6-2. BLOCK DIAGRAM, MINI SYSTEM BOARD

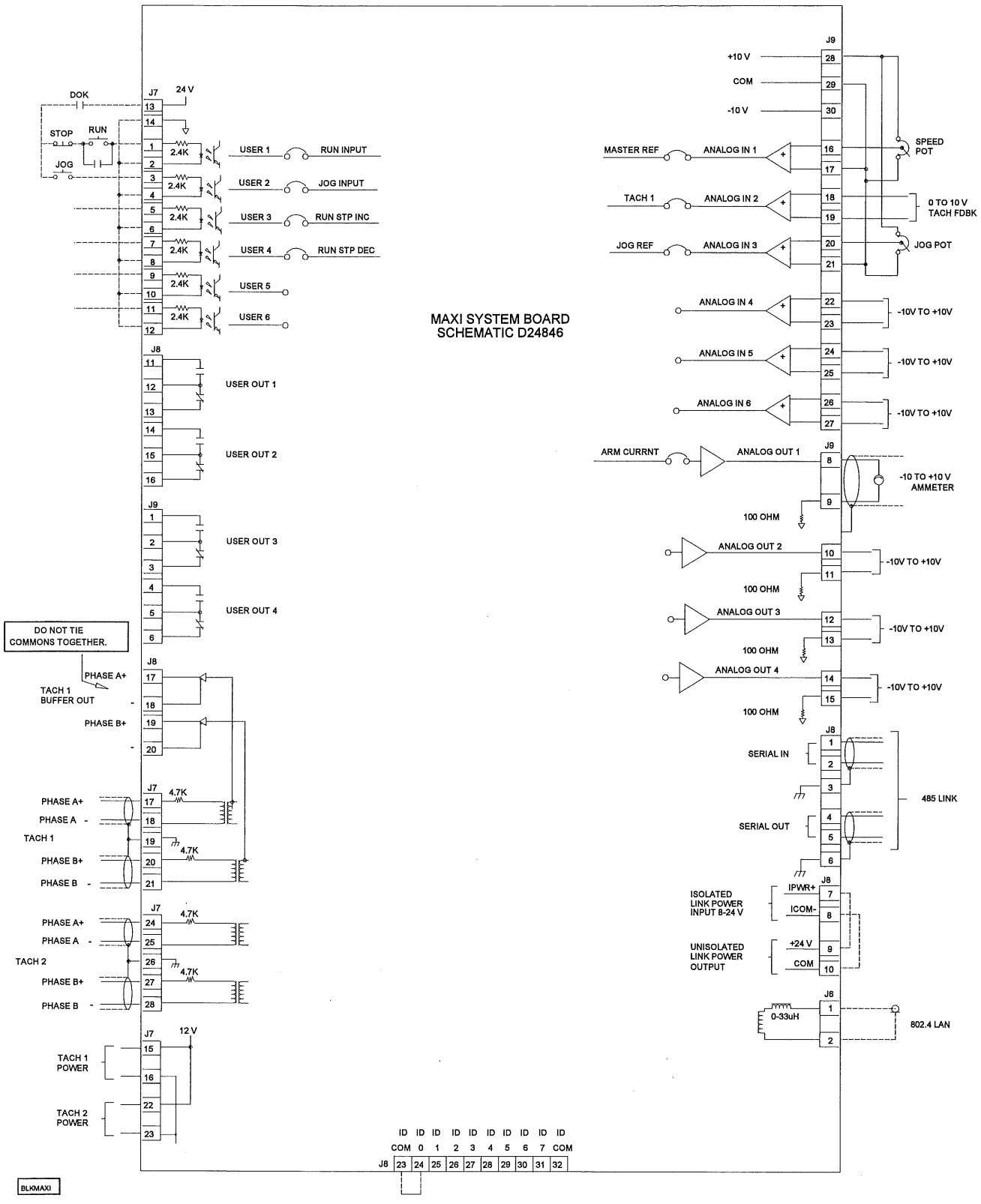


FIGURE 6-3. BLOCK DIAGRAM, MAXI SYSTEM BOARD

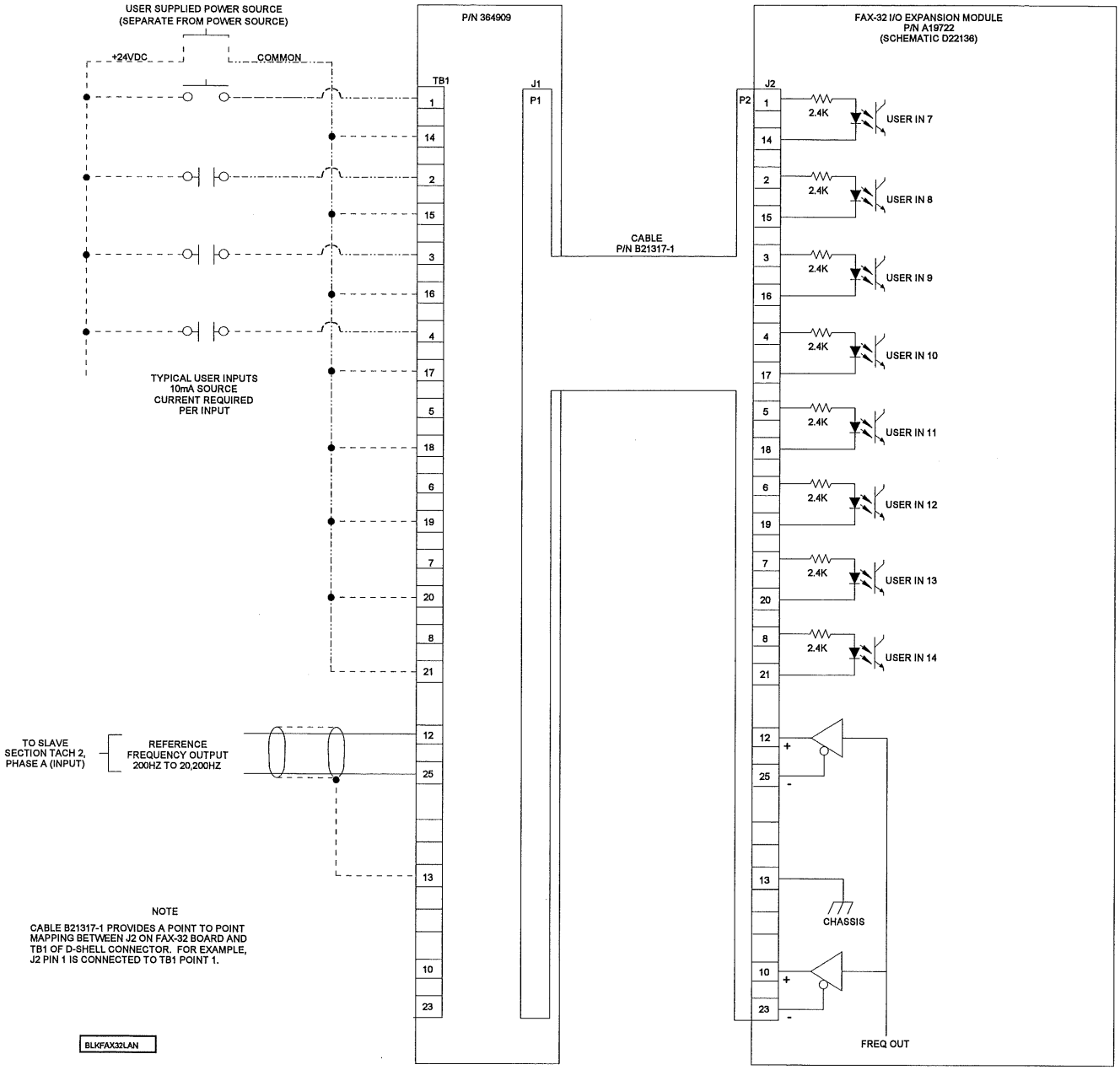


FIGURE 6-4. BLOCK DIAGRAM, FAX-32 BOARD

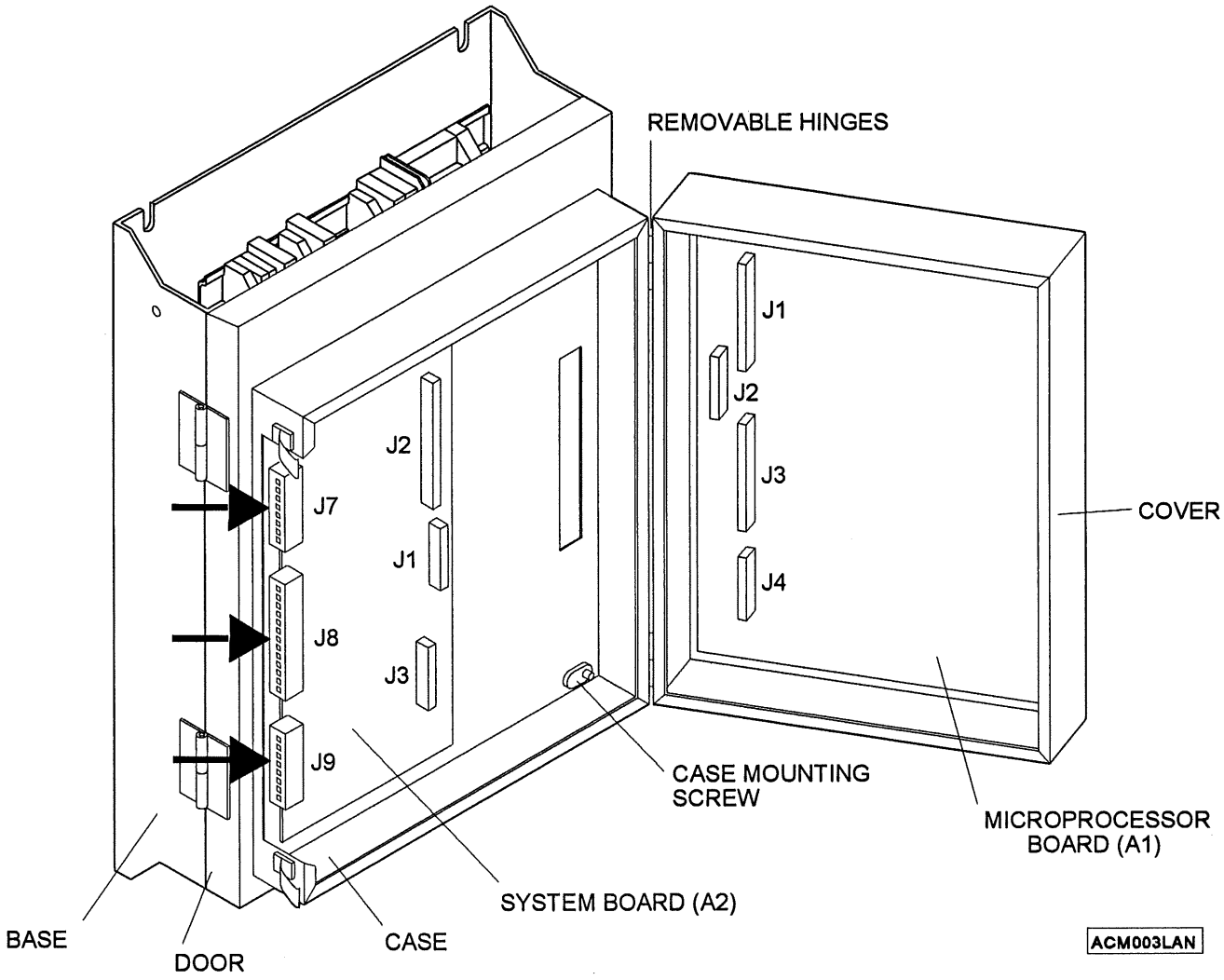


FIGURE 6-5. MINI SYSTEM BOARD INTERCONNECTIONS

NOTE

Board interconnections are the same for all ADD-32 units.

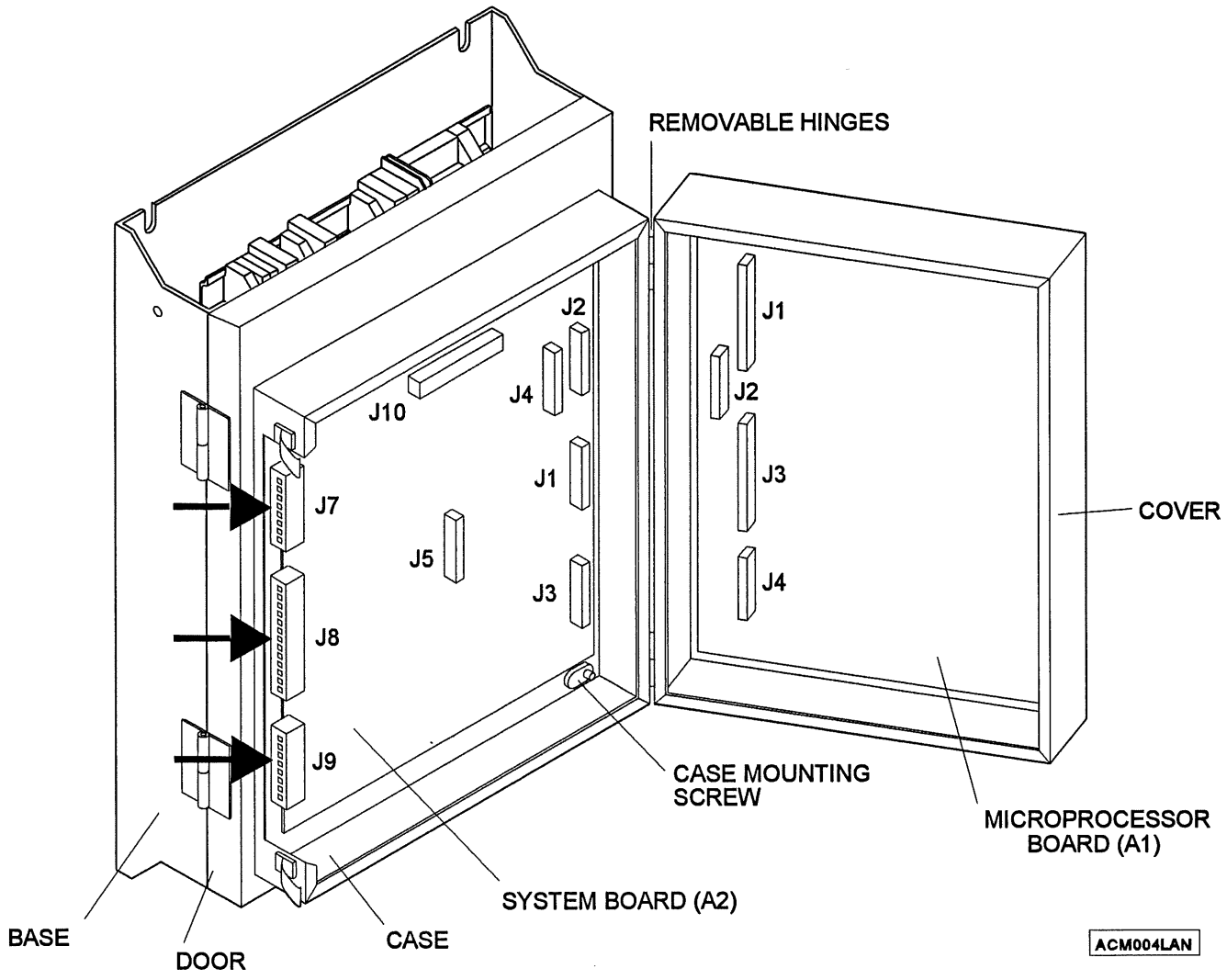


FIGURE 6-6. MAXI SYSTEM BOARD INTERCONNECTIONS

NOTE

Board interconnections are the same for all ADD-32 units.

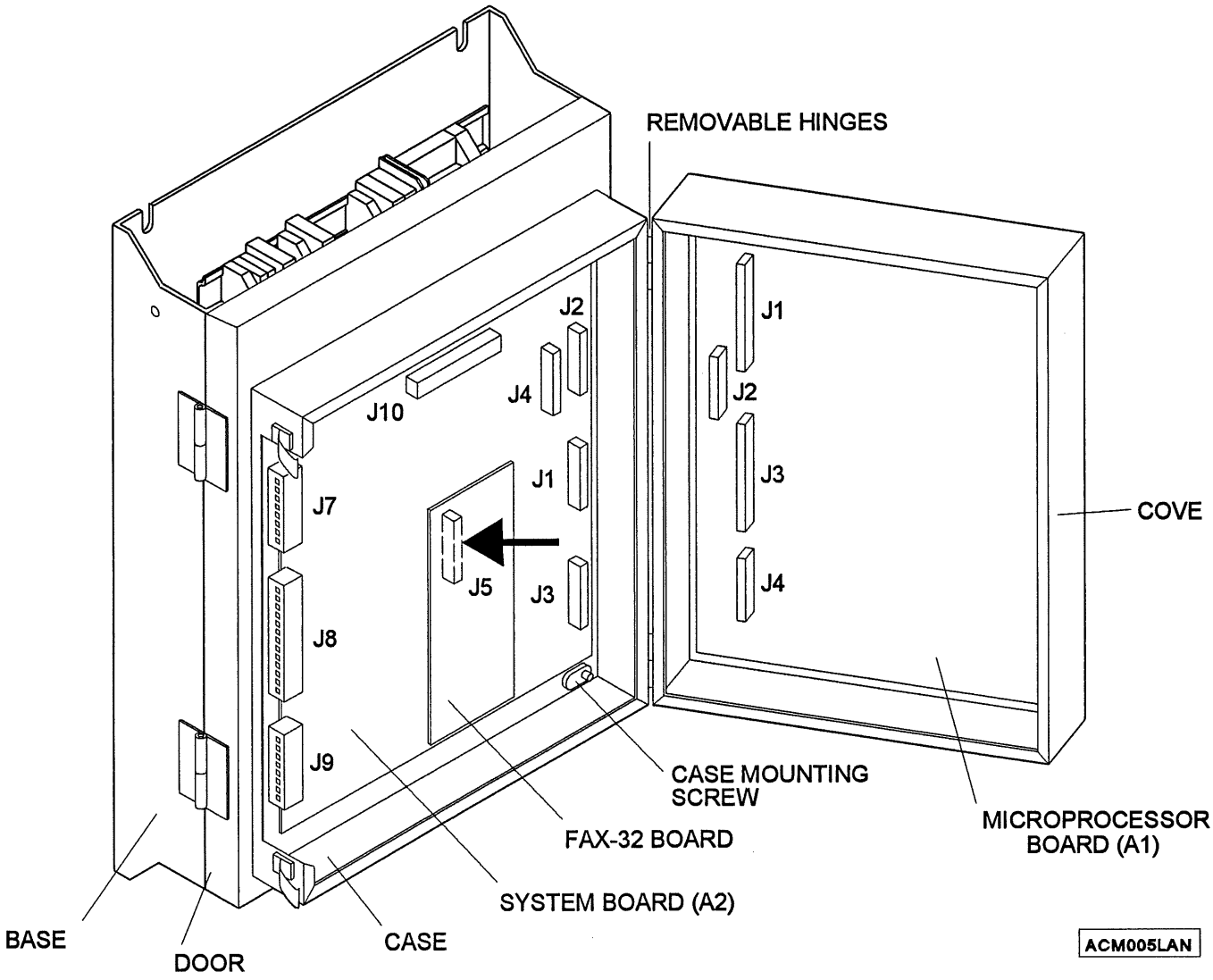


FIGURE 6-7. FAX-32 BOARD INTERCONNECTIONS

NOTE

Board interconnections are the same for all ADD-32 units.

6.7 ELECTRICAL INSTALLATION

Interconnection wiring and connection diagrams are provided in the supplemental drawing manual and are identified in that Table of Contents. These drawings include part numbers of the major assemblies, and corresponding Schematic/Interconnection Wiring Diagrams.

WIRING PRACTICES AND NOTES

****

W A R N I N G

THIS EQUIPMENT CONTAINS HAZARDOUS VOLTAGES. THE MAIN DISCONNECT DEVICE MAY NOT REMOVE ALL HAZARDOUS VOLTAGE SOURCES. TO AVOID LOSS OF LIFE, SEVERE PERSONAL INJURY, OR PROPERTY DAMAGE, FOLLOW ALL INSTRUCTIONS CONTAINED IN THIS MANUAL AS WELL AS PROPER SAFETY PRACTICES.

HAZARDOUS VOLTAGES MAY ALSO BE PRESENT ON EXTERNAL SURFACES OF THE CONTROLLER CASE, IF NOT PROPERLY GROUNDED. AVOID LOSS OF LIFE, SEVERE PERSONAL INJURY, OR SUBSTANTIAL PROPERTY DAMAGE BY FOLLOWING SAFETY PRACTICES.

****

NOTE

All conductor sizes referenced in this manual are for copper conductors with a minimum insulation rating of 60°C for less than 100 Amps and 75°C for 100 Amps or more.

Before wiring units according to the Interconnection Diagrams, review the following notes applying to routing and termination of interconnection cables.

It is recommended that power cables and signal cables be routed in separate conduit. When using cable trays, route power lines in different trays and keep parallel power trays at a maximum distance from signal wires to avoid possible "cross-talk".

All pulse generator cables may be run in common conduit or cable trays; these cables include an electromagnetic shield.

Use of conduit is generally recommended instead of cable trays because protection against physical or chemical damage is provided and, if grounded, electromagnetic shielding is provided.

Use the following guidelines to simplify wiring of the individual cables to respective terminal boards in the drive and to minimize the possibility of electrical noise ("cross-talk") between conductors in different cables.

1. Avoid excessive lengths of cable or individual conductors.

2. While preparing each cable for connection to the terminal strip, cut back and remove the cable covering so that exposed insulated wires extend the length of the terminal strip.

3. Locate the terminal connection point for each wire. Trim the wire to length, allowing just enough slack to avoid the wiring being pulled taut over terminal strips or their wiring after harnessing. All terminals should be accessible. Strip back the insulator approximately 3/8" on each conductor.

4. Printed circuit board mounted terminal blocks have clamping type terminals for interconnection of cables. Therefore, it is not necessary to tin and add lugs to the wires before connection to the terminal board. The stripped wire ends are inserted under a clamping plate, and a screw tightens the plate down over the wires, securing the connection.

NOTE

Use caution in stripping the insulation from wires and tightening terminals of terminal blocks. Incorrect stripping or tightening may result in damage to equipment and or injury to personnel.

NOTE

Do not bundle signal and power cables/wires in a common harness. Separate power wires from signal wires as much as possible.

6.8 INSTALLATION CHECKOUT

After all equipment is mounted and wired, check over the entire installation to be sure that all mounting hardware is secure, all pulse generators are properly aligned, and all wiring interconnections are in accordance with the diagrams in this manual.

NOTE

Insure that ground cables are properly attached to the drive chassis. Refer to the outline drawing (in drawing package) for location of grounding points.

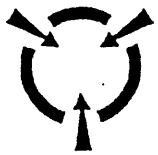
6.9 POWER UP AND START UP

Applying power and starting up the ADD-32 should be done only by trained and qualified personnel. Familiarity with the hardware, the software, and the environment in which the equipment is to be applied is critical to safe and efficient system operation.

6.10 PULSE GENERATOR INSTALLATION

Proper mechanical and electrical installation of the digital speed feedback device (Pulse Generator) is critical for optimum performance and reliability. Follow the instructions provided by the Pulse Generator manufacturer and system interconnection diagrams, if applicable.

Particular care should be exercised in selecting the shaft coupling. Use a zero backlash, Thomas Miniature Flexible or equivalent. When axial endplay exceeds +/-0.020", use Thomas CCX or equivalent. Do not use a helical spring type or rubber element type coupling.



ESD PRECAUTIONARY GUIDELINES

C A U T I O N

Certain circuit card assemblies and their components, typically integrated circuits, may be damaged by seemingly undetectable electrostatic discharge (ESD). Care must be exercised during handling/repair of these items. Use electrostatic discharge precautionary procedures.

The following guidelines are not necessarily all inclusive but rather serve as reminders for good shop practices for the handling/ repair of ESD sensitive circuit card assemblies and devices.

- Store ESD sensitive items in their original containers. These items are often marked with the symbol shown at the top of this page.
- Put on a grounded wrist strap before handling any ESD sensitive item.
- Clear work area of Styrofoam[R]*, plastic, and vinyl items such as coffee cups.
- Handle ESD items by the body, never the open edge connectors.
- Never slide ESD sensitive items over any surface.
- Transport ESD sensitive items in a static shielding container to a static-free work station.
- If a static-free work station is not available, ground the transport container before removing or inserting an ESD item.
- Electric tools used during repair should be grounded. For example, use only anti-static type solder suckers and grounded tip soldering irons. Discharge non-electric tools before use.
- Pack ESD items in static shielding containers before shipping them to Avtron for repair.

*Styrofoam[R] is a registered trademark of Dow Chemical.

SECTION VII

MAINTENANCE AND TROUBLESHOOTING

****

W A R N I N G

THIS EQUIPMENT CONTAINS HAZARDOUS VOLTAGES. AVOID LOSS OF LIFE, SEVERE PERSONAL INJURY, OR PROPERTY DAMAGE BY FOLLOWING THE INSTRUCTIONS CONTAINED IN THIS MANUAL.

THE POWER COMPONENTS AND POWER SUPPLY BOARD HAVE CIRCUITS WHICH CONTAIN HAZARDOUS VOLTAGES.

TURN OFF AND LOCKOUT ALL POWER SOURCES AT THE FEEDER TO THE ADDvantage-32 BEFORE PERFORMING MAINTENANCE OR REPAIR. BEWARE OF "FOREIGN" EXTERNAL POWER SOURCES WHICH MAY STILL BE LIVE ALTHOUGH THE MAIN AC FEED IS OPENED.

ONLY QUALIFIED PERSONNEL SHOULD PERFORM THE FOLLOWING PROCEDURES ON THIS EQUIPMENT DUE TO THE COMPLEXITY OF THE ADD-32. PERSONNEL SHOULD BE FAMILIAR WITH ALL SAFETY NOTICES, INSTALLATION, OPERATION AND MAINTENANCE PROCEDURES IN THE MANUAL PRIOR TO WORKING ON THE ADD-32.

****

7.1 MAINTENANCE REQUIREMENTS

Proper maintenance of the ADDvantage-32 and its associated motor and process equipment is required. Refer to the instructions supplied with the motor and associated process equipment for proper maintenance procedures.

PERIODIC MAINTENANCE

Under normal operating conditions, the ADDvantage-32 requires minimal periodic maintenance. The unit should be checked periodically for dirt accumulation within the enclosure. Regularly clean or replace filters on low pressure clean air

source ventilation openings. Wire terminations should be checked periodically for tightness.

To minimize downtime in the event of a component failure, spare parts should be stocked at the installation site. Refer to the spare parts list in the supplemental drawing package for recommendations on stocking of spare parts.

Use the following checklist for periodic maintenance.

1. Keep unit free from dirt. Vacuum or use low pressure air to keep clean.

2. If cabinet is equipped with filters, keep them free of dirt.

3. Check wire terminations for tightness.

****

W A R N I N G

THE USE OF UNAUTHORIZED PARTS IN THE REPAIR OF THIS EQUIPMENT, OR TAMPERING BY UNQUALIFIED PERSONNEL MAY RESULT IN DANGEROUS CONSEQUENCES. AVOID LOSS OF LIFE, SEVERE PERSONAL INJURY OR EQUIPMENT DAMAGE BY FOLLOWING RECOMMENDED SAFETY AND REPAIR PROCEDURES.

****

7.2 GENERAL DIAGNOSTIC FEATURES

The ADD-32 has an extensive fault and diagnostic check capability. Refer to Table 7-3 for a list of faults and warnings that can be detected. There is also a watchdog timer that is reset by the microprocessor under normal operation. If the processor fails to reset the timer, then the DOK contactor opens and a DOK fault indication is displayed. Although fault and diagnostics checks are frequently identified by the DOK (Drive OK) circuit, not all conditions will be caught. Examples of fault conditions that cannot be detected by the ADD-32 itself include, but are not limited to, the following:

I/O including digital and analog. A failure in the hardware associated with the inputs or outputs can occur on a chip level and not be detected.

Broken or loose interconnection/wire. There are several connections inside the ADD-32 including critical feedback signals and/or connections to I/O external to the ADD-32 that could fail.

Memory failure after initial power up check. The ADD-32 checks memory as part of initial power up diagnostics, but a subsequent failure may go undetected and result in unpredictable operation. The watchdog timer is included to guard against this, but cannot be guaranteed to catch all possible memory faults once initial power up checks are completed successfully.

SCR failure after initial power up check. The ADD-32 can be configured to check proper firing on initial start up. A subsequent failure of an SCR to gate for any reason cannot be detected until the next RUN command sequence.

The ADDvantage-32 uses built-in diagnostics which constantly monitor for faults that may occur in the controller, motor, or other external inputs. The diagnostic levels used to assist in troubleshooting are as follows:

1. INITIAL POWERUP CHECK

Upon powerup of the ADDvantage-32, the diagnostic test checks all boards, cables, and incoming power. During this test the 12 status LED's on the front panel illuminate simultaneously, then shut off one at a time. If a failure occurs during the test, the corresponding LED remains illuminated and a fault message appears on the display. Refer to Table 7-1 for definitions of the LED's and Table 7-2 for definitions and solutions to the powerup fault messages.

NOTE

Faults must be corrected to continue operation of the ADDvantage-32. There is a slight delay in operation while the powerup check is performed.

NOTE

LED's on the front of the Advanced Control Module are defined by a decal that is provided with the unit. This decal has locations for writing in information for the eight user LED's. Use a fine-tip permanent marker to write the appropriate L.E.D. designators on the label.

Table 7-1. LED Definitions

TITLE	COLOR	DEFINITION
+5 V POWER ON	Green	Lights when 5 VDC power is present.
USER 1 LED	Green	Can be set by user to illuminate when a digital bit in the data table goes high.
EMERG STOP OK	Green	Lights when emergency stop input is high and emergency stop reset has been pressed.
DRIVE READY	Green	Lights when all internal faults have been cleared and fault reset has been pressed. Drive ready and emergency stop inputs must be on to run the unit.
USER 2 LED	Amber	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 3 LED	Amber	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 4 LED	Amber	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 5 LED	Amber	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 6 LED	Red	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 7 LED	Red	Can be set by user to illuminate when a digital bit in the data table goes high.
USER 8 LED	Red	Can be set by user to illuminate when a digital bit in the data table goes high.
PROCESSOR FAIL	Red	Lights when watchdog timer on microprocessor board times out.

Table 7-2. Powerup Messages

FAULT MESSAGE	PROBLEM	SOLUTION
SYS BD BUS CABLE MISSING	Microprocessor board unable to communicate with system board.	a. Check cable B20623 at J2 on system board. b. Replace system board. c. Replace microprocessor board.
SYS BD ANALOG CABLE MISSING	Microprocessor board missing analog information from system board.	a. Check cable B20622 at J3 of system board. b. Replace system board. c. Replace microprocessor board.
SYS BD POWER CABLE MISSING	Microprocessor board detects missing power on system board.	a. Check cable B20727 at J1 of system board. b. Replace system board. c. Replace microprocessor board.
PWR I/F BOARD CABLE MISSING	Microprocessor board getting incorrect information from power supply board.	a. Check cable B20726-1 at J4 of microprocessor board. b. Replace power supply board. c. Replace microprocessor board.
EEPROM CHKSUM FAIL CODE	Stored checksum does not match actual checksum.	a. Displayed after software change. To default memory, Press S. "CORRECT CHECKSUM ?" is displayed. Press down (no) arrow. "DEFAULT CAL DATA?" is displayed. Press up (yes) arrow. After the display is defaulted, remove power to reset the ADDvantage-32. b. If software was not changed, replace microprocessor board. c. Bypass the warning by pressing S. "CORRECT CHECKSUM ?" is displayed. Press up (yes) arrow.
EEPROM VERSION INCOMPATIBILITY	Stored cal data version does not match program version.	Must default calibration data to prevent erratic functioning. To default memory, Press S. "CORRECT VERSION?" is displayed. Press down (no) arrow. "DEFAULT" CAL DATA?" is displayed. Press up (yes) arrow. After the display is defaulted, remove power to reset the ADDvantage-32.

2. FAULTS AND WARNINGS

The faults and warnings alert the user to the condition of the ADDvantage-32. Faults are conditions that can damage the system if the condition continues. To protect against damage, the ACM opens the DOK (Drive OK) contact. Warnings occur when limits in the ACM have been reached or when a fault is about to occur, allowing the user to correct the condition before a fault occurs.

Fault Queue - The fault queue stores the last sixteen faults using a FIFO method of data storage. The FIFO faults reside in RAM and if a power loss occurs, the faults are saved in EEPROM. The FIFO stores all faults and issues a FLT CLEARED message indicating that the operator cleared the fault but not the FIFO memory. To reset the Fault Queue, set the CLR FLT QU bit to logic 1.

TYPES OF WARNINGS AND FAULTS

A. Hardcoded Faults and Warnings - Hardcoded faults are not user-definable. See Table 7-3 for a list of faults and warnings. See Table 7-4 for fault messages and corrective action.

B. User Defined Faults/Warnings - There are six possible user defined fault messages which may be configured to any digital data point.

Table 7-3. Hardcoded Faults and Warnings

FAULT MESSAGE	FAULT (F) OR WARNING (W)	MOTOR CONTACTOR MUST BE PICKED UP?
A/D OFFSET	F	N
A/D SPAN	F	N
CNTL POWER	F	N
EEPRM WRITE	W	N
FLT CLEARED	W	N
INVALID A/D	F	N
L-FREQ HIGH	F	N
L-FREQ LOW	F	N
MOTOR STALL	F	Y
MOTOR TEMP	F	N
OPEN SYS CBL	F	N
USER 1 FLT	F	N
USER 2 FLT	F	N
USER 3 FLT	F	N
USER 4 FLT	F	N
USER 5 FLT	F	N
USER 6 FLT	F	N
USER 7 FLT	F	N
USER 8 FLT	F	N

Table 7-4. Fault Descriptions

FAULT MESSAGE	PROBLEM	SOLUTION
A/D OFFSET	Bad auto offset. Adjust on analog inputs.	Replace microprocessor board.
A/D SPAN	Bad auto span. Adjust on analog inputs.	Replace microprocessor board.
CNTL POWER	Loss of power to the unit.	a. If only fault, check plant power. b. Check other faults.
EEPRM WRITE	Noncritical fault. Check calibration parameters for bad data.	Replace microprocessor board.
FLT CLEARED	Faults have been manually cleared.	N/A
INVALID A/D	Calibration parameter I/O V-REF out of limits.	a. Correct calibration. b. Replace microprocessor board.
L-FREQ HIGH	Line frequency above 63 Hz.	a. Check plant power frequency. b. Replace power supply board. c. Replace microprocessor board.
L-FREQ LOW	Line frequency below 47 Hz.	a. Check plant power frequency. b. Replace power supply board. c. Replace microprocessor board.
MOTOR STALL	Motor failed to move with armature current.	a. Check setup of stall parameters. b. Check for mechanical binding. c. Check for correct armature and field current scaling. d. Check for correct speed feedback. e. Replace motor.
MOTOR TEMP	Motor Thermal Overload-Armature Current above 100% for more than rated time.	a. Enable P***:BYPASS I2R to auto phase back current limits. b. Check for proper field current. c. Check for a mechanical problem causing excess currents. d. Check for proper current limit settings.

Table 7-4. Fault Descriptions (Cont.)

FAULT MESSAGE	PROBLEM	SOLUTION
OPEN SYS CBL	Senses missing cable between microprocessor board and system board.	a. Check cable B20623. b. Replace system board. c. Replace microprocessor board.
USER 1 FLT	Digital bit configured to USER FLT 1 has gone high (1).	Check proper configuration for USER FLT 1.
USER 2 FLT	Digital bit configured to USER FLT 2 has gone high (1).	Check proper configuration for USER FLT 2.
USER 3 FLT	Digital bit configured to USER FLT 3 has gone high (1).	Check proper configuration for USER FLT 3.
USER 4 FLT	Digital bit configured to USER FLT 4 has gone high (1).	Check proper configuration for USER FLT 4.
USER 5 FLT	Digital bit configured to USER FLT 5 has gone high (1).	Check proper configuration for USER FLT 5.
USER 6 FLT	Digital bit configured to USER FLT 6 has gone high (1).	Check proper configuration for USER FLT 6.
USER 7 FLT	Digital bit configured to USER FLT 7 has gone high (1).	Check proper configuration for USER FLT 7.
USER 8 FLT	Digital bit configured to USER FLT 8 has gone high (1).	Check proper configuration for USER FLT 8.

3. SIGNAL ANALYZER

The Signal Analyzer (refer to Section V) also assists the user in diagnosing failures. The analyzer can record four parameters over a specified time period, which is useful in detecting machine or system problems. For example, speed feedback can be recorded to detect a bad gear tooth or out of balance roll. Refer to Section V for setup and operation of the signal analyzer.

4. FTYPE FAILURE TYPE PROCEDURES

If a FTYPE failure occurs, the following procedure should be followed to gather the information needed for Avtron to solve the fault.

TYPICAL DISPLAY:

```
Ftype:02 sub:01
AT:  FFFF81D0
```

a. Copy ALL information accurately on initial FTYPE DISPLAY -- BOTH LINES!!

b. PRESS the UP ARROW KEY very briefly. You should see the following display with the "AT" replaced by a level number in parenthesis. If you do not see a (1), then press the DOWN ARROW KEY to get back to the "AT" and try again.

```
Ftype:02 sub:01
(1)   800271C4
```

c. When the (1) level is displayed, copy down the lower line of the display (the level and the number to the right of the level).

d. AGAIN, briefly press the UP ARROW KEY and step to level 2 as shown:

```
Ftype:02 sub:01
(2)   80027484
```

e. When the (2) level is displayed, copy down the lower line of the display.

f. Press the UP ARROW key to view levels 3, 4, 5, 6. Copy down the level and the number to the right of the level for each level.

g. FAX/EMAIL the DRIVE HARDWARE and DRIVE SOFTWARE part numbers and all information collected above to Avtron for evaluation.

7.3 BOARD REPLACEMENT

****

C A U T I O N

THE ADDvantage-32 PC BOARDS CONTAIN COMPONENTS WHICH MAY BE DAMAGED BY ELECTROSTATIC DISCHARGE (ESD). EXERCISE CARE DURING HANDLING/REPAIR OF THESE BOARDS.

****

7.3.1 REPLACING MICROPROCESSOR BOARD (A1) (Refer to Figure 7-1.)

1. Record all calibration and configuration data.
2. Remove power from the unit.
3. Open front cover.
4. Remove all cables (J1-J4) from microprocessor board.
5. Remove ground wire.
6. Slide front cover (with microprocessor board) up and off the hinges.
7. Attach new cover and microprocessor board to the unit.
8. Replace ground wire.
9. Replace all four cables and close the door.
10. Reapply power and re-enter data.
11. Reset unit.
12. Measure voltage across TP1 and TP2. Enter value into X***:I/O V-REF.

7.3.2 REPLACING SYSTEM BOARD (A2) (Refer to Figure 7-1.)

1. Remove power from the unit.
2. Open front cover.
3. Remove cables (J1-J6) from system board.
4. Remove all four cables from microprocessor board.
5. Remove ground wire.
6. Detach cover and microprocessor board from unit.
7. Open base assembly door.
8. Detach system board by removing the four screws located on the inside of the door.

9. Attach new system board and case.
10. Replace ground wire.
11. Close door and attach the microprocessor board.
12. Reconnect all four cables on microprocessor board and all six cables on the system board.
13. Close cover and reapply power.

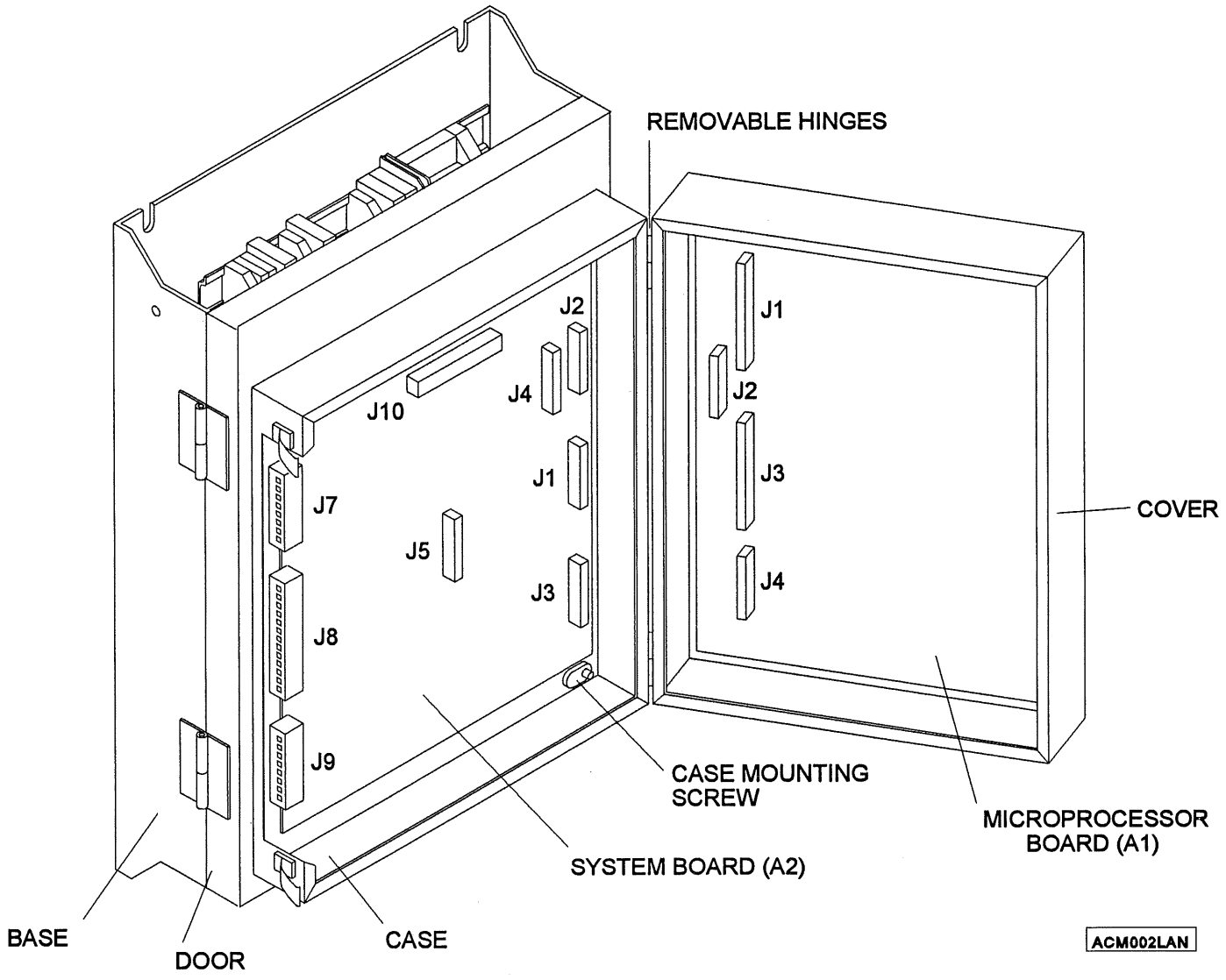


FIGURE 7-1. REPLACING SYSTEM AND MICROPROCESSOR BOARDS

NOTE

Replacement of the system and microprocessor boards is the same for all ADDvantage-32 units, regardless of size.

7.3.3 REPLACING POWER SUPPLY BOARD (A3)
(Refer to Figure 7-2.)

1. Remove power from the unit.
2. Open base assembly door.
3. Remove cables on power supply board.
4. Remove self-locking nuts that secure the power supply board to the standoffs on the base.
5. Lift the power supply board off of the standoffs.
6. Place the new board on the standoffs.
7. Install the self-locking nuts to secure the power supply board to the base.
8. Connect the cables on the power supply board and shut the base assembly door.
9. Reapply power to the unit.

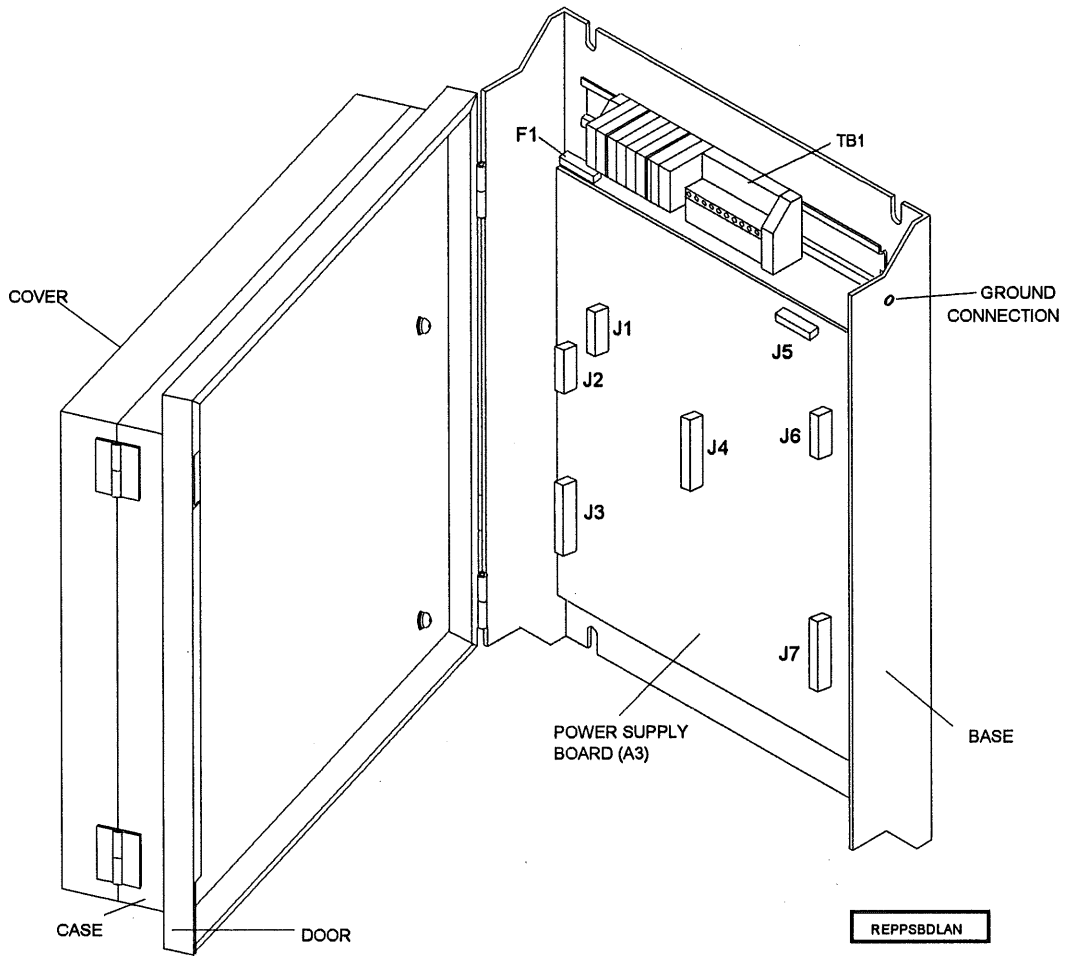


FIGURE 7-2. REPLACING POWER SUPPLY BOARD

7.4 FUSE REPLACEMENT

****

W A R N I N G

HAZARDOUS VOLTAGES WHICH CAN CAUSE SEVERE INJURY OR DEATH ARE PRESENT DURING THE OPERATION OF THIS EQUIPMENT. TURN OFF AND LOCK OUT ALL SOURCES OF POWER BEFORE MAKING ANY REPAIRS.

****

1. Remove all sources of power from the unit.
2. Open the base assembly door.
3. The only fuse, F1, is readily available near the upper-left corner of the base assembly. See Figure 7-2. Fuse F1 is 3 amp, 250 volt. The Avtron part number is 324070 and the Bussman part number is ABC-3.

****

W A R N I N G

REPLACE FUSES WITH FUSE OF SAME TYPE AND RATING. IMPROPER FUSE REPLACEMENT MAY RESULT IN DAMAGE TO THE ADVANCED CONTROL MODULE, A FIRE HAZARD, AND SEVERE INJURY OR DEATH TO PERSONNEL.

****

4. Shut the base assembly door and reapply power to the unit.

SECTION VIII

IEEE 802.3 ETHERNET COMMUNICATION

8.1 OVERVIEW

The Avtron ADDvantage-32 DC and AC Drives support an 802.3 Ethernet star structured network. The ADDvantage-32 supports both 100 base FX Fiber Optic and 10/100 base UTP Copper Ethernet media. The Avtron Ethernet network is implemented using an Avtron ESBX Ethernet board, P/N A26494 that can provide 100 Mbaud fiber full duplex IEEE 802.3 Ethernet communications. One ESBX board is required for each ADDvantage-32 drive. The ESBX board includes firmware to implement Ethernet protocols for GE, Modicon, and Rockwell PLC interfaces. In addition, the firmware includes Avtron's proprietary communications protocol, Peer Primitive. The Peer Primitive protocol enables Autoscan/Exchange communications among ADD-32 drives and provides the backbone for high speed communications to Avtron's Performance View Graphic Trending / Historical Event Recorder and Avtron's ADDAPT 2000 drive maintenance tool. The ESBX board can communicate using up to four different Ethernet protocols simultaneously over the same network cable which provides great flexibility in system design.

A managed high speed Ethernet Switch is usually provided with Avtron's Ethernet based drive systems. The Avtron provided Switch may include multiple 100 Mbaud fiber ports and multiple 10/100 Mbaud copper ports. The managed Ethernet Switch is the central node of the network and will guarantee deterministic communication to all nodes by providing one full duplex 802.3 communication port for each device on the network. Full duplex operation of a single device per port assures no data collisions can occur. Many of the provided Switches have built in Switch diagnostic functions. These diagnostics greatly enhance network maintenance capability by providing independent diagnostics for each port which can quickly identify and localize a network problem.

Patch/Wic boxes may have been provided to manage excess fiber cable and provide a means to connect each drive to the multi conductor field installed cable by way of an easily replaceable fiber patch cord cable. In the event a fiber patch cable connected to an ADD-32 drive is damaged, it is easy and inexpensive to replace the patch cable leaving the permanently installed fiber cable undisturbed.

Avtron PDC-6 drive control systems using 802.3 Ethernet communications provide a high level of reliability, performance, and diagnostics. Utilization of the 802.3 Ethernet requires ADDvantage-32 hardware and firmware capable of Ethernet support.

8.1.1 ESBX Ethernet Board

The ESBX Ethernet board P/N A26494 Rev. D and higher is hardware configurable to use either Fiber Optic or Twisted Pair UTP (Copper) media. This board is shipped from Avtron configured for use with fiber optic media. To reconfigure ESBX board for twisted pair (copper) media, move jumpers TP2 and TP3 on board from positions 1 and 2 to positions 2 and 3.

The ESBX Ethernet board P/N A26494 Rev. C and lower is pre-assembled to use fiber optic media and cannot be modified to use copper. The RJ45 connector exists on this fiber version simply for manufacturing simplicity. ESBX Ethernet board P/N A26495 is pre-assembled to use twisted pair (Copper) media and cannot be modified to use fiber. This copper ESBX board P/N A26495 has now been replaced with P/N A26494 Rev. D and higher.

P/N A26494 Rev D and higher = Fiber Optic or Twisted Pair UTP Copper Media

P/N A26494 Rev C and lower = Fiber Optic Media Only

P/N A26495 = Twisted Pair UTP Copper Media Only (Superseded by P/N A26494 Rev D and higher)

ESBX Physical Ethernet interface: 100 MHz Fiber, full duplex, MTRJ connector

10/100 Meg Copper UTP, half/full duplex, RJ-45 connector

Each ESBX board has been assigned a unique MAC Address. The MAC Address is printed on a label located on the component side of the ESBX board. The MAC Address has also been programmed into ESBX firmware by Avtron.

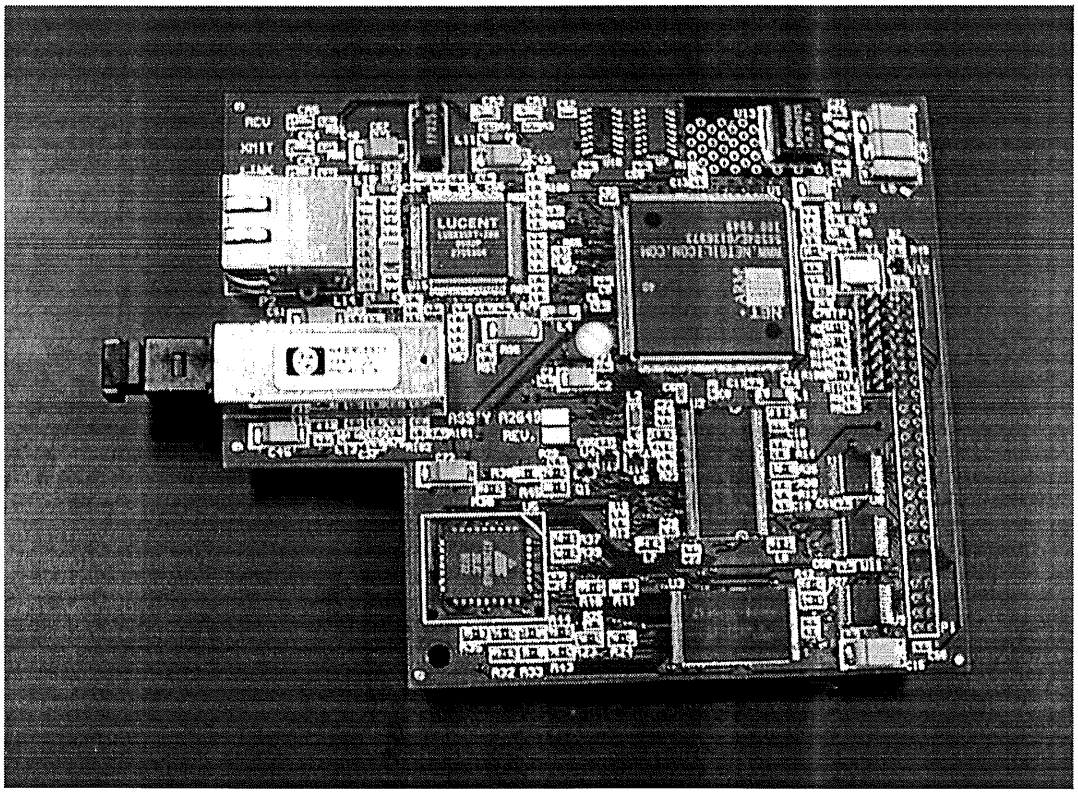


Figure 8-1. Avtron ESBX Module

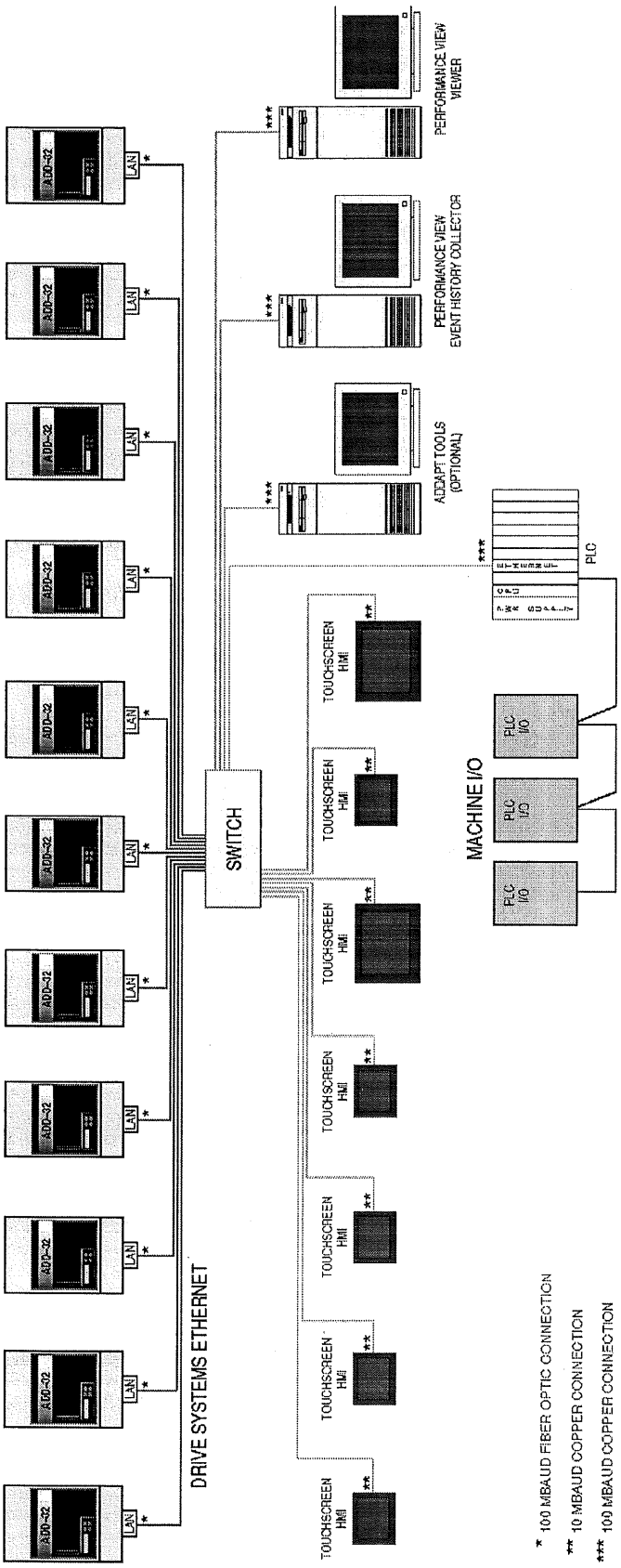
The ESBX boards are sensitive to Electro Static Discharge (ESD). These boards must be properly handled using an Anti-Static Wrist Wrap connected to a good ground! The ESBX boards must be carried and/or shipped in approved Anti Static bags!

One ESBX Ethernet board is required for each ADDvantage-32 Drive. The ESBX board plugs into SBX connector J4 on the System Maxi board for ADDvantage-32 DC Drives and SBX connector J11 on the Microprocessor board for ADDvantage-32 AC Drives. Install the ESBX board by inserting board connector into the SBX connector. Make sure that both the top and bottom of the ESBX board are fully inserted. Screw down the ESBX board using nylon screw.

The ESBX board has five LED's in the upper left corner for visual indication of board operation. The LED's are defined as:

CR1 LED:	Normally flashing amber about once per second. This LED indicates communication between the drive and the ESBX board.
CR2 LED:	Normally off. This LED is controlled by the processor and is for Avtron debug use only.
RCV LED:	Green flash during receive of incoming frame data.
TX LED:	Green flash during transmitting of outgoing frame data.
Link Active LED:	Illuminates solid Red after 30 second boot up following drive power cycle or drive reset, provided there is a valid cable connection to an Ethernet Switch or device.

The following picture shows an overview of a typical Avtron Drive System utilizing Ethernet communications.


ADD-32
LAN
DRIVE SYSTEMS ETHERNET
SWITCH
TOUCHSCREEN HMI
ADAPT TOOLS (OPTIONAL)
PERFORMANCE VIEW EVENT HISTORY COLLECTOR
PERFORMANCE VIEW VIEWER
PLC
MACHINE I/O
PLC I/O
* 100 MBAUD FIBER OPTIC CONNECTION
** 10 MBAUD COPPER CONNECTION
*** 100 MBAUD COPPER CONNECTION

8.2 ADDvantage-32 HARDWARE/SOFTWARE REQUIREMENTS

Most ADDvantage-32 drives manufactured after mid 1994 can be easily field upgraded for Ethernet support without requiring any drive board replacements. Avtron service engineers have routinely upgraded many systems during normally scheduled maintenance outages of 8 to 12 hours. If drives have previously used 802.4 LAN communications, installation of the ESBX Ethernet board requires removal of the 802.4 Modem (P/N A22122) on the Maxi System board.

The ADDvantage-32 boards can be identified by a part number and revision level which is ink stamped on the component side of the board. An existing drive can be field upgraded for Ethernet support if the drive currently uses 692xxx application software and has a MAXI System Board with a part number of A18876-1 or A19359-1. All Bridge Interface Boards (BIB) that are in drives running 691xxx application software (or later) will support Ethernet. Very old Bridge Interface Boards, typically the non "-1"'s, will not support Retentive Setpoints, which is the real compatibility issue, not Ethernet.

The following represents the hardware and software requirements for Ethernet Support:

Hardware Requirements

- High Memory Microprocessor Board P/N 630264 with Boot PROM (U9) software P/N 680635 V15 or higher.
- MAXI System Board must be P/N A18876-1 or A19359-1.
- Maxi System Board requires serial EPROM (U5) software P/N 681444.V11 or greater (sometimes labeled as 444).
- Removal of Modem Board P/N A22122 if 802.4 LAN communications has been previously used.
- ESBX Ethernet Board P/N A26494 (Rev. D and higher) - Hardware configurable for Fiber Optic or UTP Copper twisted pair media.
- Compatible Managed Ethernet Switch for network.
- Fiber Optic Cables and/or CAT 5 Cables.

Software Requirements

- ADDvantage-32 Application Software P/N 694xxx for DC Drives or P/N 695xxx for AC Drives.
- ADDAPT 2000 must be used if the ADDvantage-32 Application Programming Tools are required. ADDAPT Dos and ADDAPT 98 do not support Ethernet communication.
- Performance View System must be used if Historical trending and logging of I/O events is required as the DOS version of Event History Recorder does not support Ethernet.

8.3 ADDvantage-32 DRIVE ID/IP ADDRESS AND CONFIGURATION

8.3.1 DRIVE ID/IP Address

The ADDvantage-32 Drive ID is set by binary weighted inputs from hard-wired jumpers on the Maxi System Board J8 connector for DC Drives and the J6 connector on Microprocessor board for AC Drives. This is the only place you can confirm that the terminal blocks are wired as required to give you the desired Drive ID. The Drive's ID applies to the RS-485 Serial Link, 802.4 Local Area Network (LAN) and possibly to 802.3 Ethernet.

The ADDvantage-32 drive also has an Ethernet IP address which is relevant when the drive has an ESBX Ethernet board installed and running application software P/N 694xxx or 695xxx. The Ethernet IP address contains four numeric fields called "Octets" that are separated by a decimal point between fields. An Octet is a numerical value made up of eight binary places (bits). Octets can represent decimal numbers from zero (0000 0000) to 255 (1111 1111). Each of the four fields can contain a number ranging from 0 to 255. An example IP address is 10.1.206.6 (referred to as, " ten point one point two zero six point six).

The Ethernet IP address can be assigned to the ADDvantage-32 drive by either the "HARD SET" or "DRIVE ID BASED" method.

The "HARD SET" IP address method has all four fields of the IP address set by drive calibration parameters in the "DRIVE CALIBRATE" menu. The four calibration parameters that define the IP Address are "IP ADDR HI", "IP ADDR HM", "IP ADDR LM" and "IP ADDR LO". Note that IP ADDR LO must not be set to 256 for the "HARD SET" method. Be aware that if the "HARD SET" IP method is used, the DRIVE ID is still required for ADDAPT 2000 operation.

The "DRIVE ID" BASED address method is the same as the HARD SET IP, with the following exception: IP ADDR LO must be set to 256. The drive will then use the DRIVE ID as set by the hard wired jumpers for the number in the lowest octet field of the IP ADDRESS. The reason for the DRIVE ID BASED address is to allow the drive to have a default ETHERNET IP address that will allow ETHERNET communications from an initial power up

without any edit of drive cal data. The default IP ADDRESS for any drive is: 10.1.206.LAN ID. The ADDvantage-32 Drive reads the DRIVE ID jumper configuration on power up and displays the DRIVE ID as "DRIVE COM ID" in the "DIAGNOSTICS" menu. Changes to the DRIVE ID jumpers require a drive reset or power cycle before the new DRIVE ID takes effect.

An example of a DRIVE ID BASED address:

DRIVE ID = 12

IP ADDR HI = 10
IP ADDR HM = 1
IP ADDR LM = 206
IP ADDR LO = 256

This would result in an actual drive IP ADDRESS = 10.1.206.12

8.3.2 ADDvantage-32 DRIVE ETHERNET CONFIGURATION

Only ADDvantage-32 Application software P/N 694xxx (DC Drive) and P/N 695xxx (AC Drive) support Ethernet. A valid IP Address, Subnet Mask (MSK), and Gateway (GWY) Address must be entered for their respective "X" parameters in the "Drive Calibrate" menu. These numbers will be the same for all drives in a system except for the last octet (LO address) of the IP Address (if the 256 DRIVE ID address method is used, they can all match). It will be necessary to manually set the IP Address, Subnet Mask, Default Gateway and Machine NO parameters of a spare drive or Microprocessor board that is installed to replace a failed drive. Until these parameters are correctly set for the replacement drive, the drive will not communicate correctly over ETHERNET. You will not be able to download Calibration and Configuration data to the drive via ADDAPT 2000 over ETHERNET until these parameters are correctly set.

Obtain unused IP addresses from your MIS or IT department when setting up drives on a network so the ID jumper may be set accordingly (or set ADDR LO to the available last octet). If you are unsure if an IP Address is already used, use the "PING" command at a DOS prompt on a PC connected on the network. The "PING" command can be used to test the communication between two programs over the network. To invoke the PING command, go to a Dos prompt and type "ping xxx.xxx.xxx.yyy", where xxx.xxx.xxx is the 1st three octets of the other device's IP address and yyy is the number you picked. If you receive *Request Timeouts*, there is nothing currently on the network using that address. However, if the drives are connected to your company's network, and you don't obtain an address from your MIS Department, there could be a device that is currently off that could be turned on at a later time resulting in problems. The following illustrates a return following a PING command:

```
C:\>ping 10.1.207.213

Pinging 10 1.207.213 with 32 bytes of data:

Reply from 10.1.207.213: bytes=32 time<10ms TTL=128
Reply from 10.1.207.213: bytes=32 time<10ms TTL=128
Reply from 10.1.207.213: bytes=32 time<10ms TTL=128
Reply from 10.1.207.213: bytes=32 time<10ms TTL=128

Ping statistics for 10.1.207.213:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

"Request timed out" indicates the device is not present or not communicating.
"Destination host unreachable" indicates you are not on the correct network or are not connected.

There are ESBX Ethernet configuration parameters contained in "DRIVE CALIBRATE"(X) and "DRIVE CONFIGURE" (Y) menus of the ADDvantage-32 Drive. The parameter numbers will vary depending on the ADDvantage-32 Application software being used and will be noted below as "***". Please reference Appendix C of your ADDvantage-32 Drive manual that shows the parameter listing for your application software.

Drive Calibrate Menu – "X" Parameters

DRIVE IP ADDRESS:

X***: IP ADDR HI: Highest octet of the DRIVE IP ADDRESS. Range 1-255, default = 10
X***: IP ADDR HM: High Middle octet of the DRIVE IP ADDRESS. Range 1-255, default = 1
X***: IP ADDR LM: Low Middle octet of the DRIVE IP ADDRESS. Range 1-255, default = 206
X***: IP ADDR LO: Low octet of the DRIVE IP ADDRESS. Range 1-256, default = 256

Each ADDvantage-32 drive on the network must have a unique IP address assigned via X090-X093 parameters. Parameter X093 may be set to a special value of "256" that indicates the LOW OCTET of the IP address should be set to the DRIVE ID ADDRESS. The ADDvantage-32 drive IP address may need to be changed from default values to meet your specific company's requirements. If not specified by your MIS or IT Department, the IP address can be left at the default values. Be aware that duplicate IP Addresses on the network will cause communication problems and failures!

DRIVE SUBNET MASK:

X***: MSK ADDR HI:	Highest octet of the DRIVE SUBNET MASK – range 0-255, default = 255
X***: MSK ADDR HM:	High Middle octet of DRIVE SUBNET MASK – range 0-255, default = 255
X***: MSK ADDR LM:	Low Middle octet of DRIVE SUBNET MASK – range 0-255, default = 0
X***: MSK ADDR LO:	Lowest octet of the DRIVE SUBNET MASK – range 0-255, default = 0

The Drive Subnet Mask is defaulted to a value that will allow operation on the Avtron Cleveland Network. The values of the Subnet Mask may need altered to match the specifics of your company's network. Your MIS Department may specify the IP ADDRESS CLASS and subnet range. The Subnet Mask will be 255.255.255.0 or 255.255.0.0 depending on your network topology. A class A network would be 255.0.0.0, a class B network would be 255.255.0.0, and a Class C would be 255.255.255.0. Be aware that incorrect or inconsistent subnet mask values will cause communication problems and failures!

A subnet mask divides a network into sub networks. For example, we use 255.255.0.0, which is a Class B network. 255.0.0.0 would be Class A, and 255.255.255.0 is Class C. The 255.255.0.0 means that anyone with 10.1.xxx.xxx can communicate with each other. That is over 64,000 devices. A Class C subdivides this into 254 different networks containing up to 254 each. In any given subnet, a person can only talk to 253 other devices. Computer 10.1.207.213 cannot see a drive at 10.1.206.1 on the Class C network, but could on a Class B. This will normally be 255.255.0.0 unless the customer specifies otherwise. A subnet could be further divided; for example, 255.255.255.128, 10.1.206.1-10.1.206.127 would be one subnet and 10.1.206.129-10.1.206.254 another, not accessible by the other.

GATEWAY ADDRESS

X***: GWY ADDR HI:	Highest octet of the GATEWAY IP ADDRESS – range 0-255, default = 10
X***: GWY ADDR HM:	High Middle octet of GATEWAY IP ADDRESS – range 0-255, default = 1
X***: GWY ADDR LM:	Low Middle octet of GATEWAY IP ADDRESS – range 0-255, default = 205
X***: GWY ADDR LO:	Lowest octet of the GATEWAY IP ADDRESS – range 0-255, default = 3

The GATEWAY IP ADDRESS is defaulted to a value that will allow operation on the Avtron Cleveland Network. The Gateway IP Address must be set to a valid address on your network. Avtron normally sets the Gateway IP Address for all drives on a system to the Ethernet Switch's IP Address as users would be less likely to try and use the Switch IP Address by mistake. Failure to set the GATEWAY IP ADDRESS that is consistent with the DRIVE IP ADDRESS and DRIVE SUBNET MASK will result in a total loss of communications. Also,

failure to set the GATEWAY IP ADDRESS to a valid address may result in failures in communications to devices not on the drive's Subnet. Your MIS Department may want to specify the GATEWAY IP ADDRESS.

"Gateway" is a generic term for an internetworking system (a system that joins two networks together). Gateways can be implemented completely in software, completely in hardware, or as a combination of the two. We are not using gateways. The gateway address should be set to some legal IP address in the network or even the same as the drives IP. The reason for this is that an illegal address could cause software lockup.

MACHINE NUMBER

X***: MACHINE NO: Range 0-7, default 1.

The MACHINE NO parameter is used for customer installations that have multiple ADDvantage-32 based machines (process control systems) connected to and accessible to the same ETHERNET NETWORK. The MACHINE NO parameter is only relevant to ADDAPT 2000 (ADDvantage-32 Application Programming Tools) where a Channel # from 201 to 208 must be entered in the ADDAPT configuration (ADDAPT.cfg) for each drive section. The MACHINE NO is also used in the setup of ADDAPT Real Time Screens. The ADDAPT Channel # is equal to the X***: MACHINE NO + 200. Typically, all ADDvantage-32 drives on a specific machine/drive system would be set to the same Machine Number. ADDAPT 2000 uses both the drive's Machine Number and DRIVE ID for identifying and establishing Ethernet communication to drives for each machine/drive system within your facility.

WEB PASSWORD

X***: WEB passWrd: <<<<< FUTURE USE TO ENABLE ACCESS TO DRIVE'S WEB PAGE>>>>>>

Drive Configure Menu – "Y" Parameters

Y***: ETHRNET COM: DISABLE, ENABLE, AUTO-ENABLE, default = AUTO-ENABLE

This parameter determines if the drive will communicate via ETHERNET. If this parameter is set to DISABLE or AUTO-ENABLE, then an ESBX Ethernet board is not required on the drive for correct operation. If the parameter is set to ENABLE, then an ESBX Ethernet board must be installed for the drive to operate. If there is no ESBX Ethernet board installed but the ETHRNET COM parameter is set to ENABLE, the drive may operate erratically and/or reset unexpectedly.

If ETHRNET COM is set to AUTO-ENABLE, then the drive will attempt to determine if the ESBX Ethernet board is installed as part of the power-up sequence. If the ESBX Ethernet board

is detected and correctly initialized, its presence will be indicated under the DIAGNOSTICS menu of the drive keyboard/display. The drive LCD display will show:

DRIVE COM ID:

xx (ETH)

where xx = Drive ID # and ETH indicates presence of ESBX Ethernet board.

Y083: ETH AUT-RST: DISABLE, ENABLE, default = DISABLE

MUST BE SET TO DISABLE

8.4 ESBX ZAP – DOWNLOADING FIRMWARE TO ESBX ETHERNET BOARDS

The ESBX Ethernet boards contain firmware in their Flash Memory. Although not normally a field requirement, this firmware can be Field upgraded via Ethernet using Avtron's ESBX_Zap program (PN 682937) from the source computer, usually the ADDAPT 2000 PC. Normal operation of the ESBX Ethernet board will cease upon start of ESBX ZAP which will take up to 3 minutes to complete. The ESBX board will provide indication of completion of the ZAP program by an alternating flash of CR2 and CR3 LED's. Use caution as a power cycle or drive reset during the ZAP operation or before the ZAP is complete will render the ESBX board inoperable and will require factory reprogramming at Avtron.

Zapping an ESBX board on a running drive is not totally risk free. However, there should not be a problem unless the drive is configured to generate a fault on a COM LOSS, or you have other ETHERNET devices attempting to communicate to the drive during and after the ZAP. If you are attempting to download to a failed Ethernet board whose failure symptom was a total loss in communication, then the procedure below for downloading ESBX software to the board will not work as communication with board is required. Avtron's factory can program the ESBX board without communication but it is not available for field use at this time.

The IP ADDRESS that should be entered into the ESBX ZAP program is the IP ADDRESS that is set in the "DRIVE CALIBRATION" menu. The IP ADDRESS is not programmed into the ESBX board; the ESBX board obtains its IP ADDRESS from the ADDvantage-32 drive on which it is mounted. The reason the ESBX ZAP program needs the board's IP ADDRESS is because the programming data is sent to the board via ETHERNET and the IP ADDRESS is the board address.

MAC or Media Access Control address is a unique value associated with a network adapter. MAC addresses are also known as hardware addresses or physical addresses. The MAC address remains fixed and follows the network device, but the IP address changes as the network device moves from one network to another. TCP/IP protocol is a higher level and requires the IP address.

ESBX Ethernet Board Firmware Download Instructions:

1. Record Target ADDvantage32 drive information (Drive IP and Subnet mask addresses). From the drive keypad, enter the "Drive Calibrate" menu and record the following parameters: The "X" parameter numbers shown below are for drive application software P/N 694012 V13. Please reference Appendix C of your ADDvantage-32 Drive manual that shows the parameter listing for your specific application software.

X090: IP ADDR HI _____
X091: IP ADDR HM _____
X092: IP ADDR LM _____
X093: IP ADDR LO _____
X094: MSK ADDR HI _____
X095: MSK ADDR HM _____
X096: MSK ADDR LM _____
X097: MSK ADDR LO _____

NOTE

Do not enter the number 256 for X093. Use the actual DRIVE ID jumper settings on J8 as viewed for DRIVE COM ID in the drive's Diagnostic menu. If X093 does not equal 256 then use the number in X093.

2. Record the ESBX board MAC Address, located on label of U1 of the ESBX board.

MAC Address: ____ - ____ - ____ - ____ - ____ - ____

3. Record the Source computer's network settings information (Computer's IP and Subnet mask addresses). This information can usually be found under the Internet Protocol (TCP/IP) properties found in the Local Area Connection properties of computers Network and Dial Up Connections setup.

IP Address: _____ : _____ : _____ : _____
Subnet Mask: _____ : _____ : _____ : _____

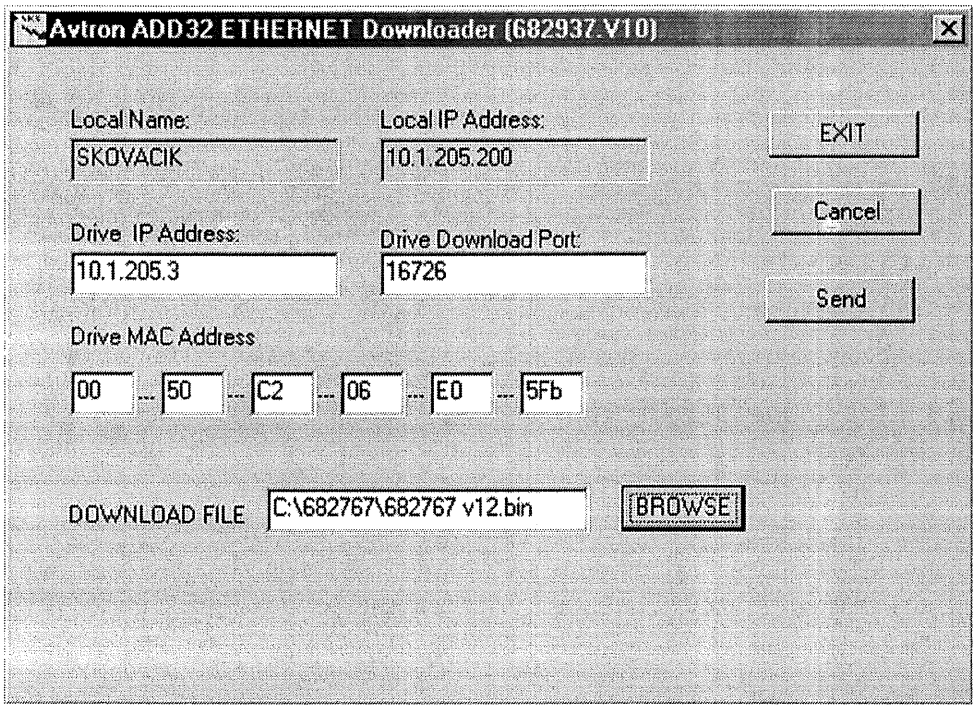
NOTE

The drive and the computer's network address settings must be such that they are both capable of communicating on the same network and have a physical communication path established.

4. Execute the ESBX_Zap program (PN 682937) from the source computer. Fill in the following information:

   Drive IP Address: ____.____.____.____
   Drive Download Port: 16726
   MAC Address: ____ - ____ - ____ - ____ - ____ - ____

5. Click the "BROWSE" button and select the ESBX firmware file 682767 V12.bin or the latest version.



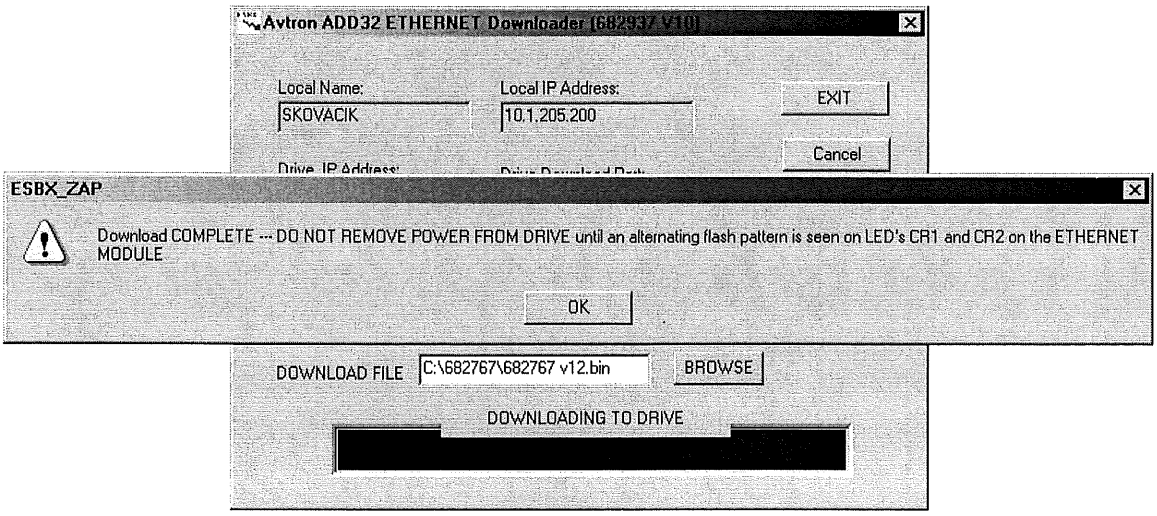
6. Click the "Send" button. A progress bar will indicate progression of file transfer. When the transfer is complete, the following dialog will appear:

****

WARNING

DO NOT POWER DOWN THE DRIVE DURING AND AFTER THE DOWNLOAD UNTIL THE GREEN AND AMBER LIGHTS ON THE ESBX BOARD START TO BLINK BACK AND FORTH AT A RATE OF ABOUT ONCE PER SECOND!!!!!!

****



7. As indicated, once the LED's on the ESBX module are alternately flashing, indicating the firmware has been successfully transferred into non-volatile memory, perform a cold restart of the drive by removing and then re-applying control power to the drive.

8.5 ETHERNET SWITCHES

A Switch is a network device that connects two or more separate network segments and allows traffic to be passed between them when necessary. A Switch determines if a packet should be blocked or transmitted, based on the destination address contained in that packet. A Switch is like a hub which is an Ethernet (10BaseT or 100BaseT UTP/STP) repeater. However, a Switch does not bombard all of the ports with all data. A Switch sends incoming data only to its intended device so as to prevent collisions. Avtron provides Ethernet Switches from several different manufacturers. Please reference your Switch manual for technical information specific to your Switch model.

While most Switches do not require an IP Address assignment for operation, Avtron recommends that Switches be assigned an IP Address so that Switch diagnostic information can be obtained via Internet Explorer web pages. Avtron also normally uses the Switch's IP Address for the ADDvantage-32 drive's Gateway Address as most users would know this IP Address as being used by the Switch and would be less likely to use the address by mistake for other devices.

Some Switches are factory set with an administrative account and can be accessed over the serial link using a serial link cable provided with the Switch. The default Log In User Name and Password for many Switches is:

User Name: admin
Password: admin

Be aware that some Switches do not have an administrative account or may use a different User Name and Password than above. See your Switch manual for specific technical information. If the User Name and Password are changed from their factory default, Avtron recommends using:

User Name: avtron
Password: 64183

Once logged in via the provided serial cable, an IP address can be assigned to the Switch such that the Web-based configuration can be utilized. Many Switch manufacturers allow creation of separate user accounts with different privilege levels. In general, the default Switch configuration settings are used. The 'admin' will usually always get you in. After the initial setup of Switch IP Address via the serial cable, the Switch can be accessed by typing the IP Address as the URL in Internet Explorer.

Switch ports labeled with an X are internally crossed over. When connecting two devices, a straight through cable is used if only one device has an X. If both or neither device have an X, a Crossover cable is required. Your computer network port is not internally crossed over. Straight through CAT5 cables are used to connect your ADDAPT computer, X-link, PLC, HMI Operator Displays, Performance View, etc., to a Switch, but not to each other. Crossover cables are used to connect your computer directly to a PLC, Xlink, or even another computer or to connect two or more Switches together. Two Switches should be connected by Fiber Optic cable and simply swap connections at one end. This generally requires popping the halves of one end out of a clip, swapping, and reattaching the clip.

Ethernet Switches can be provided with a combination of both 100 MHz base Fiber Optic and 10/100 MHz UTP Copper ports with the following specifications:

100baseTX, CAT5 cable, RJ45 connector (RJ21 for mini) max length 100m

100baseFX, fiber optic, 2km full duplex (multimode cable 412m half duplex, singlemode 10km+)

For Switches that provide diagnostics using a web browser, start Internet Explorer from Windows using the ADDAPT or PLC computer on the network. Type the Switch's IP Address on the Address Line. Enter User Name and Password if required for access to the Switch Diagnostics. You may want to create a shortcut on your desktop for faster future access to the Switch's Diagnostics.

8.6 PLC's AND ADDAPT 2000 VIA ETHERNET COMMUNICATION

8.6.1 PLC's

The Avtron ADDvantage-32 is a high performance and highly configurable digital drive with resident capability to perform drive control functions locally utilizing a PLC for operator interface, process I/O, and permissive logic. The PLC is not burdened with time critical and

complex drive control functions yet retains the ability to communicate to the drives via a high performance Ethernet network.

Based on high speed, open Ethernet communication protocols you get the most flexibility going into the future. Avtron's ESBX Ethernet board supports the following protocols that are compatible with popular Programmable Logic Controllers (PLC's):

- Modicon: Modbus TCP

- General Electric: Ethernet Global Data, (EGD) (drive-to-drive communication with or without a PLC)
  Includes GE 90/70, 90/30 PLC's

- Designed and interfaces with Rockwell: Ethernet IP
  Type 1 (ControlLogix) Implicit Messaging
  Type 3 ControlLogix, SLC, and PLC-5 (except 5/250)

  Includes Allen Bradley Control Logix, PLC-5 and SLC PLC's

- Avtron Peer Primitive: Autoscan, ADDAPT, and other Avtron tool interfaces

All protocols are included with ADDvantage-32 software and all are active simultaneously. These protocols allow high-speed communication from PLC's, HMI Operator Displays, ADDAPT 2000, Avtron Performance View Process Diagnostic System, and other external devices to the ADD-32 drives as well as drive to drive. The benefits are minimized system wiring and enhanced system reliability, while retaining maximum flexibility through the life of your drive system. All Drive related system I/O (except E-Stop) can be communicated over the Ethernet network. Detailed descriptions of these protocols can be rather complex and lengthy for this user manual. Please reference the Ethernet Engineering Reports and technical Application Notes for the most recent up-to-date list of supported Ethernet protocols. This information can be obtained on Avtron's web site at http://www.avtron.com/.

ADDAPT 2000 and PLC programming packages can operate simultaneously on the same computer if required. It does not require two network cards inside the desktop computer to communicate to two devices, unless the two devices exist on two totally isolated networks. Note: In a Windows 2000 system, if you disable multiple network adapter (NIC) cards, then re-enable them, they are re-assigned as ports in the order they are enabled. This means you can accidentally swap the NIC's used by eth0 & eth2.

8.6.2 ADDAPT 2000

ADDAPT 2000 Programming Tools must be used if communicating to drives over Ethernet. ADDAPT 98 and the older ADDAPT DOS software do not support Ethernet communication. ADDAPT DOS will not work using serial link or any form of communication when drive has Ethernet Application software. Before communicating with ADDvantage-32 drives over

Ethernet using the ADDAPT 2000 Programming Tools/PC, you will need to obtain an available IP address for your ADDAPT computer. Depending on who designed the Ethernet network, you may need to get this information from the Avtron Project Engineer, your plant Project Engineer or your MIS Department. If you arbitrarily pick an IP Address, make sure that it is not used by the drives, PLC, Operator Displays, Performance View or any other device that may be connected on the network. If you are unsure if an IP Address is used, use the PING command at a DOS command prompt on a network PC.

To set your ADDAPT computer IP settings using a Win 2000 operating system, go to *Start-Settings-Control Panel-Network and Dial-up Connections*. Right click the Local Area Connection for the network board being used and select "Properties". Highlight the "Internet Protocol (TCP/IP)" and click on Properties. Check circle for "Use the following IP address". Enter the desired IP Address, Subnet Mask and Default Gateway. Click "Ok", "Ok", and Yes to reboot if prompted. Note that Windows 2000 does require rebooting the computer when changing IP settings. These IP setting instructions will be different if using a WIN 98, WIN NT or WIN XP operating system. Refer to your computer/Windows documentation if required. To see your computer's own IP Address or MAC Address, type "IPCONFIG/ALL" from a DOS command prompt.

To set up ADDAPT 2000 software for Ethernet communication, click on *"Tools" – "Options"*. Inside the *Ethernet* tab are selections for Channels 201-208. The last digit of Channel number corresponds to the drives X102: MACHINE NO setting. Set the ADDAPT Channel # equal to the drive's Machine NO + 200. All drives in a system for a specific machine should be given the same MACHINE NO and Channel #. ADDAPT 2000 looks for all drives with the MACHINE NO and DRIVE ID corresponding to the channel(s) that are enabled. IP Address settings do not need entered into ADDAPT 2000. You must also set the Channel # for each drive section when setting the ADDAPT configuration file for the first time.

Reference your ADDAPT 2000 manual on the ADDAPT CD or in the C:\Program Files\Avtron Manufacturing\ADDAPT\Docs folder for further technical information.

8.7 ETHERNET COMMUNICATION TROUBLESHOOTING

In the event that you encounter any Ethernet communication problems, it will be necessary to isolate the root cause of the problem. Follow the Ethernet Troubleshooting Checklist below in the order presented. If the problem still cannot be resolved, call Avtron's Industrial Automation Field Service Department at (216) 642-1230 for further assistance. This checklist is a guide for determining where problems exist. Jot down notes for each step and present this information to the Avtron service engineer to help isolate the trouble area.

1. In what way did Ethernet board fail? Was there total loss of communication or intermittent loss of communication?

2. Are there any ADD-32 drive faults or drive LED abnormalities that may indicate a more serious drive problem? Refer to Section VII of the ADD-32 Drive Manual for LED, Power Up, and Fault definitions.

3. Has the ADD-32 ESBX board completed its 45 second boot up cycle following a power cycle?

4. Did Ethernet communication fail on one or more than one drive? If there is communication loss to multiple drives, can the problem be traced to a common Switch blade or a common Multi Fiber Cable? For communication loss to either single or multiple drives:

   a. Measure and verify that Switch power is applied.

   b. Is the Fiber or CAT5 Cable plugged in at both the Switch port and at the ESBX board or device end? Check connections.

   c. Are Switch port LED's turned "on" for the section(s) with communication problems?

   d. Temporarily move Fiber or CAT5 Cables from suspect ports to good ports. For communication loss on multiple drives, use a port on a different Switch blade. If this restores communication, the port may be bad or intermittent. Replace Switch blade or Switch.

   e. If possible, run a temporary Fiber or CAT 5 Cable between drive(s) in question and the Switch port. If this restores communication, the cable is bad. Replace the Fiber, Multi Fiber or CAT 5 Cable.

   f. Does cycling power to the Switch restore port communications to the drive(s)? If cycling Switch power restores communication, replace Switch blade or Switch.

5. Has the DRIVE COM ID connector been removed by mistake for connector J8 on System Maxi Board (DC Drive) or connector J6 on Microprocessor board (AC Drive)? If it has, re-install connector and **Reset** drive.

6. Has the ADD-32 drive's IP Address, Subnet Mask, or Gateway Address been changed by mistake in the "DRIVE CALIBRATE" menu? Has the drive's calibration and configuration data been defaulted by mistake? Verify all Ethernet Calibration and Configuration parameters defined in Section 8.3.2.

7. Has the Ethernet communication problem occurred following replacement of the System Maxi board? If it has, go in to the "DIAGNOSTICS" menu on the drive LCD display. Verify the ESBX PN/VER numbers. If garbage information is displayed, the Serial PROM U5 on the Maxi System board is older than the version V11 or higher required. It is also possible to observe erroneous "Bad Power Cable" drive faults. Replace Maxi board with spare containing Serial PROM U5 of V11 or higher.

8. Observe and document the LED's located on the drive's ESBX board at the time of failure:

   CR1 STATUS:
   CR2 STATUS:
   RCV LED:
   TX LED:
   LINK ACTIVE LED:

9. If ESBX board LINK ACTIVE LED is "off" and Switch port LED is "off":

   a. Cycle power to ADD-32 drive and wait for the 45 second boot up to complete. Did the LINK ACTIVE LED and Switch port LED turn "on"? If so, replace the ESBX Board.

10. If ESBX LINK ACTIVE LED is "on" but communication problems still exist:

    a. Can ADDAPT 2000 and/or Performance View communicate to the drive(s)?

    b. Has another device come online in the network with a duplicate IP Address? Isolate the drive network from any other company networks.

    c. Go to a DOS command prompt and attempt to "PING" the drive(s). Use ADDAPT or PLC computer that is connected on the network. Can you ping the drive?

       TYPE: ping <ip address>

       EXAMPLE: ping 10.1.206.6

    d. Start Internet Explorer from Windows using the ADDAPT or PLC computer on the network. Attempt to WEB BROWSE the drive by typing the drive's IP Address on the Address Line. Did ADD-32 Web page come up?

    e. Replace ribbon cable from connector J2 on Maxi System Board to J1 connector on Microprocessor Board.

    f. Replace ESBX board on drive if it was not already replaced in step 9 above.

    g. Replace Maxi System board if DC drive.

    h. Replace Microprocessor board on DC or AC drive. Setup IP Address, Subnet Mask, and Gateway Address. If communication is now working, use ADDAPT 2000 to download drive Calibration and Configuration data.

11. For Switches that can be accessed using a web browser, start Internet Explorer from Windows using the ADDAPT or PLC computer on the network. Type the Switch's IP Address on the Address Line. Enter User Name and Password if required for access to the Switch Diagnostics. Reference your Switch manual to confirm that the ports in question are

configured to be enabled, active, and configured for the proper bandwidth. Look for diagnostics that may show port status such as errors for frame packets transmitted/received.

12. In the ADD-32 Diagnostic Menu, write down all the ETHERNET STATISTICS. Wait 5 minutes and log the ones that have changed, then wait 5 more minutes and log the ones that have changed again. You only need to write the non-zero's as blanks will be assumed to be zero. Most diagnostics should be zero. Note that the Ethernet Statistics are found under DIAGNOSTICS/VIEW ETH HISTORY on the drive LCD interface. This data may require interpretation and analysis by Avtron engineers.

DIAGNOSTIC	INIT VALUE	5 MIN	10 MIN
UDP Mpool Empty			
UDP Mpool Err			
UDP Good Receive			
UDP Invalid Type			
UDP Null			
UDP Invalid Serv			
PPR Cmd Buf Busy			
PPR Cmd Buf Rclm:			
PPR Respnse Time			
PPR Reply			
PPR Overrun			
PPR Send Fail			
PPR Xmit Fail			
PPR No Reply			
QRY Frames			
QRY Responses			
QRY Overrun			
QRY Send Fail			
QRY Xmit Fail			
EXCHG Frames			
EXCHG Busy			
ASCAN Frames			
ASCAN Overrun			
ASCAN Send Fail			
ASCAN Xmit Fail			
BCAST Frames			
BCAST Busy			
BCAST Invalid			
BCAST Overrun			
BCAST Send Fail			

DIAGNOSTIC	INIT VALUE	5 MIN	10 MIN
Mb PP good rcv			
Mb PP bad rcv			
Mb PP err rcv,			
Mb PP timeout			
Mb PP good xmit			
Mb pp bad xmit			
MBus Connect Er			
MBus Max Connct			
MBus R-Connt Er			
MBus Disconncts			
MBus Invld Hdr			
MBus Rsp Snd Er			
MBus Connct 1's			
MBus Connct 2's			
MBus Connct 3's			
MBus Connct 4's			
MBus Connct 5's			
MBus Connct 6's			
MBus Connct 7's			
MBus Connct 8's			
EGD SND errors			
EGD RCV errors			
EGD CNSM Overrun			
EGD CFG Con ERR			
EGD CFG Con Good			
EGD RCV			
EGD Consume			
EGD No Match			
EXTRA 13			
EXTRA 14			
EXTRA 15			
EXTRA 16			
EXTRA 17			
EXTRA 18			
IP packets rcvd			
IP bad rcv chksm			

DIAGNOSTIC	INIT VALUE	5 MIN	10 MIN
IP too short rcv			
IP too small rcv			
IP hdr too short			
IP hdr bad len			
IP frag's rcv'd			
IP frags dropped			
IP frags timeout			
IP pks fwrd'd			
IP can't forward			
IP fwd same net			
IP unknwn proto			
IP sent up stack			
IP local out			
IP pks dropped			
IP reassm OK			
IP frag'd OK			
IP out frg made			
IP can't frag			
IP bad options			
IP no route			
IP bad version			
IP raw ip's			
TCP Connect Init			
TCP Connect Esbl			
TCP Connect Drpd			
TCP 25			
TCP 26			
TCP 27			
TCP 28			
TCP 29			
TCP 30			
TCP 31			
TCP 32			

ADDvantage-32 Drive Ethernet Diagnostics Definitions

UDP Mpool Empty: software error counter. Increments if a UDP frame is received and there is no memory buffer available to hold the frame. This is an internal error. Contact Engineering if this counter is ever observed to be incrementing.

UDP Mpool Err: software error counter. Increments if the operating system returns an error in response to a request for a memory buffer. Contact engineering if this counter is ever observed to be incrementing.

UDP Good Receive: counter containing the number of valid UDP frames that have been received by the drive. In order for the UDP frame to be valid, it must have a valid footprint.

UDP Invalid Type: counter containing the number of UDP frames that were not valid. A frame will be considered invalid if it does not have a valid footprint. The probable cause of this type error will be a programming error in the device transmitting the UDP frame.

UDP Null frame: counter containing the number of UDP frames received that had a zero data length. The probable cause of this type error will be a programming error in the device transmitting the UDP frame.

UDP Invalid Serv: counter containing the number of UDP frames received that had a valid footprint but did not indicate a valid service. The probable cause of this type error will be a programming error in the device transmitting the UDP frame.

PPR Cmd Buf Busy: counter containing the number of PEER PRIMITIVE commands that were aborted due to the buffer between the ESBX and the drive being busy. This counter incrementing indicates a communication error between the ESBX module and the drive and should be reported to Engineering.

PPR Cmd Buf Rclm: counter containing the number of times that the ESBX had to reclaim the PEER PRIMITIVE buffer from the ADD-32. If this counter is incrementing, it indicates a communication error between the ESBX module and the drive and should be reported to Engineering.

PPR Respnse Time: number of 5 millisecond time ticks that the ESBX module waited for a PEER PRIM response from the ADD-32. Values of 1-3 are to be expected. Values in excess of 10 should be reported to Engineering.

PPR Reply: number of valid PEER PRIM replies received from the ADD-32. This diagnostic incrementing indicates PEER PRIM communications activity to the ESBX.

PPR Overrun: counter containing the number of PEER PRIM commands that were discarded due to an overflow of the PEER PRIM command queue. Increments of this counter indicate that excessive volumes of PEER PRIM commands are being received by the ESBX module. This indicates an application problem.

PPR Send Fail: counter containing the number of failures returned by the PEER PRIM socket send operation. Increments of this counter indicate a probable network configuration problem, most likely related to the drive IP address, subnet mask, or gateway address.

PPR Xmit Fail: counter containing the number of PEER PRIM RESPONSE frame transmit failures. Increments of this counter could indicate a possible network configuration problem, (possibly related to the drive IP address, subnet mask, or gateway address) or a software related problem.

PPR No Reply: counter containing the number of times the ADD-32 did not reply to a PEER PRIMITIVE transaction initiated by the ESBX module. Increments of this counter indicate that the ADD-32 may not be responding to the ESBX and may be the result of a hardware problem or a possible software problem. Contact Engineering if this counter is incrementing.

QRY Frames: the number of QUERY frames received by the module.

QRY Responses: the number of QUERY RESPONSE frames generated by this drive. Note that a response will be generated only if a QUERY frame is received and the DRIVE ID and MACHINE number of the drive fall within the range specified by the QUERY frame.

QRY Overrun: counter containing the number of QUERY frames that were discarded due to an overflow of the QUERY command queue. Increments of this counter indicate that excessive volumes of QUERY frames are being received by the ESBX module. This indicates an application problem.

QRY Send Fail: counter containing the number of failures returned by the QUERY response socket send operation. Increments of this counter indicate a probable network configuration problem, most likely related to the drive IP address, subnet mask, or gateway address.

QRY Xmit Fail: counter containing the number of QUERY RESPONSE frame transmit failures. Increments of this counter could indicate a possible network configuration problem, (possibly related to the drive IP address, subnet mask, or gateway address) or a software related problem.

EXCHG Frames: number of EXCHANGE FRAMES received by the drive.

EXCHG Busy: counter containing the number of EXCHANGE commands that were aborted due to the buffer between the ESBX and the drive being busy. This counter incrementing indicates a communication error between the ESBX module and the drive and should be reported to Engineering.

ASCAN Frames: number of EXCHANGE FRAMES received by the drive.

ASCAN Overrun: counter containing the number of ASCAN or EXCHANGE frames that were discarded due to an overflow of the ASCAN/EXCHANGE command queue. Increments of this counter indicate that excessive volumes of ASCAN and/or EXCHANGE frames are being received by the ESBX module. This indicates an application problem.

ASCAN Send Fail: counter containing the number of failures returned by the ASCAN response socket send operation. Increments of this counter indicate a probable network configuration problem, most likely related to the drive IP address, subnet mask, or gateway address.

8.8 ETHERNET COMMUNICATIONS PROTOCOLS FOR ADD-32™

Ethernet communications with the ADD-32 drives are implemented using one of four common protocols or proprietary Avtron Peer-Primitives such as Avtron's ADDvantage-32 Application Programming Tools (ADDAPT™). The supported protocols are:

- Schneider Electric's Modbus TCP/IP (MB-TCP)
- GE-Fanuc's Ethernet Global Data (GE-EGD)
- Allen-Bradley's Client Server Protocol (AB-CSP), sometimes referred to as Programmable Controller Communication Commands (AB-PCCC)
- Allen-Bradley's Ethernet IP, Control/Internet Protocol (AB-CIP)

This section will describe the common concepts of Ethernet communications, network addressing and the supported protocols with the exception of the Avtron Peer-Primitives which are reserved for Avtron use such as ADDapt2000.

8.9 TRANSMISSION CONCEPTS

Most Ethernet communications use either the UDP or TCP formats as follows in 8.9.1 and 8.9.2.

The Requests for Comments (RFC) document series referenced below is a set of technical and organizational notes about the Internet (originally the ARPANET), beginning in 1969. Memos in the RFC series discuss many aspects of computer networking, including protocols, procedures, programs, and concepts.

8.9.1 User Datagram Protocol (UDP)

UDP is a standard protocol described by RFC 768, "User Datagram Protocol". UDP is an application interface to IP. It adds no reliability, flow-control, or error recovery to IP. It simply serves as a *multiplexer/demultiplexer* for sending and receiving datagrams, using ports to direct the datagrams, as shown in the following diagram.

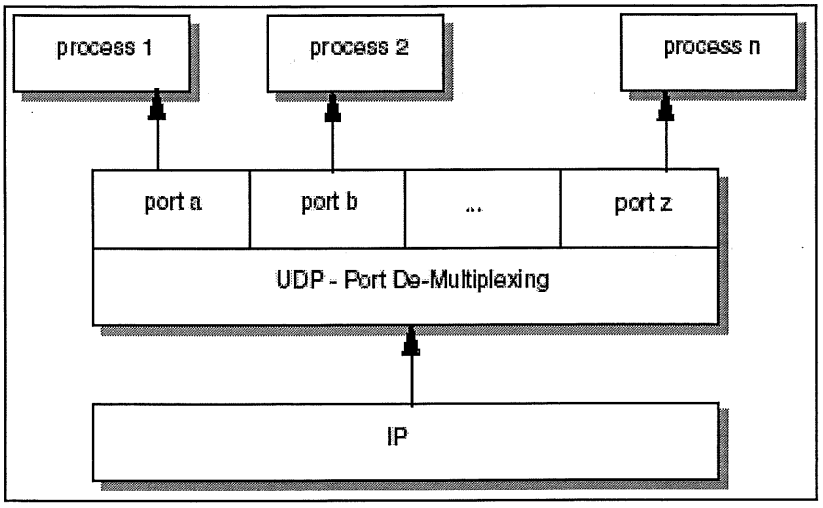


Figure 8.9.1 UDP - Demultiplexing based on ports

UDP provides a mechanism for one application to send a datagram to another. The UDP layer can be regarded as being extremely thin and consequently has low overheads, but it requires the application to take responsibility for error recovery and so on.

This format is used by the GE-Fanuc Ethernet Global Data (EGD) and Allen-Bradley Implicit Messaging.

8.9.2 Transmission Control Protocol (TCP)

TCP is a standard protocol described by RFC 793, "Transmission Control Protocol". TCP provides considerably more facilities for applications than UDP, notably error recovery, flow control, and reliability. TCP is a *connection-oriented* protocol, unlike UDP, which is *connectionless*. Most of the user application protocols, such as Telnet and FTP, use TCP. The two processes communicate with each other over a TCP connection (InterProcess Communication - IPC), as shown in Figure 8.9.2.

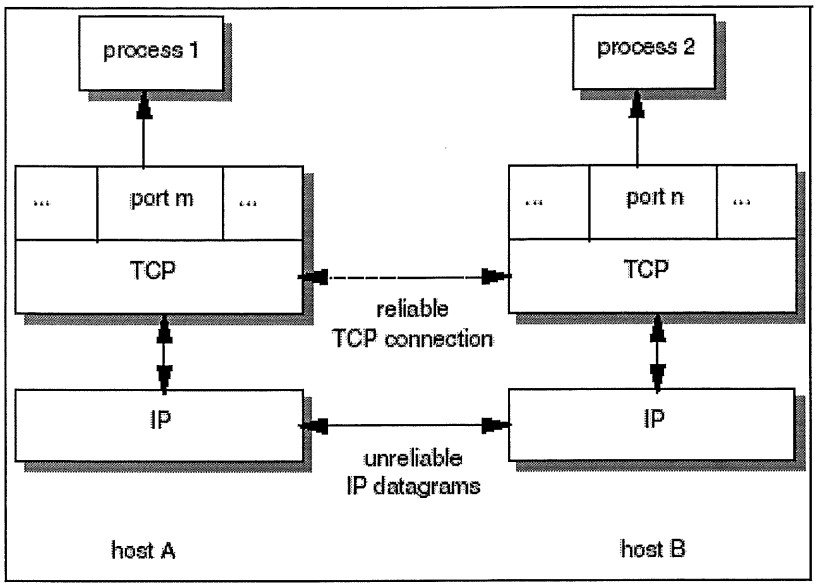


Figure 8.9.2 TCP - Connection between processes –
(Processes 1 and 2 communicate over a TCP connection carried by IP datagrams.)

A typical set of transactions is shown in Figure 8.9.3.

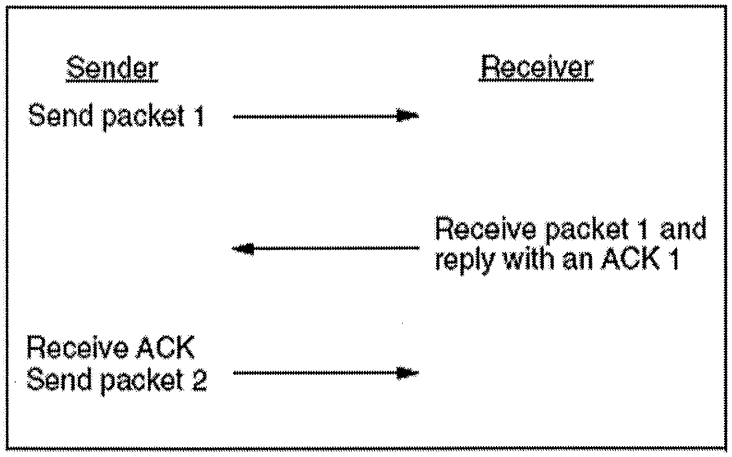


Figure 8.9.3 TCP – Transactions

This format is used by Modbus TCP and Allen-Bradley's Explicit and PCCC formats.

8.9.3 Network Addressing Concepts

The industry standard IP addressing definitions are as follows:

8.9.3.1 Class-based IP Addresses

The first bits of the IP address specify how the rest of the address should be separated into its network and host part. The terms *network address* and *netID* are sometimes used instead of network number, but the formal term, used in RFC 1166, "Internet Numbers" is network number. Similarly, the terms *host address* and *hostID* are sometimes used instead of host number. There are five classes of IP addresses. They are shown in Figure 8.9.3.1.

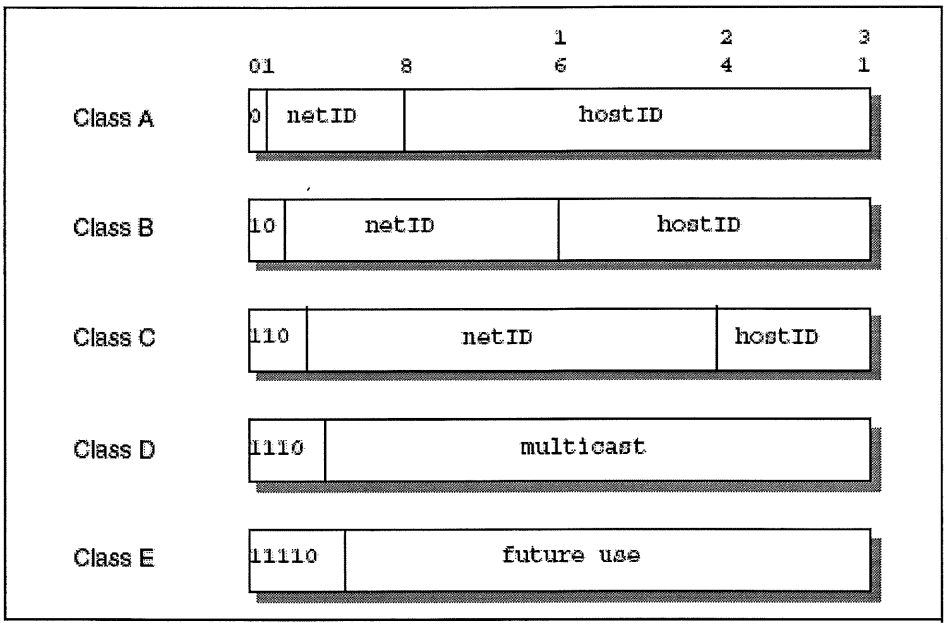


Figure 8.9.3.1 Sub-Net Classes

Where:

• Class A addresses: (1.x.x.x to 127.x.x.x), subnet mask 255.0.0.0
These addresses use 7 bits for the <network> and 24 bits for the <host> portion of the IP address. This allows for 27-2 (126) networks each with 224-2 (16777214) hosts; a total of over 2 billion addresses.

• Loopback: The class A network 127.0.0.0 is defined as the loopback network. Addresses from that network are assigned to interfaces that process data within the local system. These loopback interfaces do not access a physical network.

• Class B addresses: (128.x.x.x to 191.x.x.x), subnet mask 255.255.0.0
These addresses use 14 bits for the <network> and 16 bits for the <host> portion of the IP address. This allows for 214-2 (16382) networks each with 216-2 (65534) hosts; a total of over 1 billion addresses.

• Class C addresses: (192.x.x.x to 223.x.x.x), subnet mask 255.255.255.0
These addresses use 21 bits for the <network> and 8 bits for the <host> portion of the IP address. That allows for 221-2 (2097150) networks each with 28-2 (254) hosts; a total of over half a billion addresses.

• Class D addresses: (224.x.x.x to 239.x.x.x)
These addresses are reserved for multicasting (a type of broadcasting, but in a limited area, and only to hosts using the same class D address).

• Class E addresses: (240.x.x.x to 245.x.x.x)
These addresses are reserved for future use.

8.9.3.2 Broadcast Addressing

An address with all bits one is interpreted as *all* networks or *all* hosts. For example, 128.2.255.255 means all hosts on the class B address network 128.2. This is called a directed broadcast address because it contains both a valid <network address> and a broadcast <host address>.

8.9.3.3 Multicast Addressing

Multicast devices use Class D IP addresses to communicate. These addresses are contained in the range encompassing 224.0.0.0 through 239.255.255.255. For each multicast address, there exists a set of zero or more hosts that listen for packets transmitted to the address. This set of devices is called a *host group*. A host that sends packets to a specific group does not need to be a member of the group. The host may not even know the current members in the group. There are two types of host groups:

• Permanent: Applications that are part of this type of group have an IP address permanently assigned by the IANA. Membership in this type of host group is not permanent; a host can join or leave the group as required. A permanent group continues to exist even if it has no members. The list of IP addresses assigned to permanent host groups is included in RFC 1700. These reserved addresses include:
- 224.0.0.0: Reserved base address
- 224.0.0.1: All systems on this subnet
- 224.0.0.2: All routers on this subnet

8.9.4 Interconnect Issues and Notes

1) All devices should be connected via a Switch; the use of hubs is not recommended. For more information on Switches vs. hubs, refer to RFC 0970, "On packet Switches with infinite storage." A non-managed Switch may be used if cost is the only issue, but the management and diagnostic capabilities of a managed Switch make the additional cost a good investment to ensure a dependable network.
2) The Switch port connected to an ADD-32 drive should be set to 100mb, full-duplex for best performance.
3) All Avtron supported protocols may be used separately or together in one system, keeping in mind not to exceed 700 frames per second. Note that TCP/IP protocols use twice the frames of UDP protocols.
4) The maximum data refresh rate at the drive is 8.33 mSec. In most cases, read/write repetition rates should be expressed in multiples of 10 mSec, ie: 10, 20, 50, etc.

8.9.5 Avtron Supported Protocols

The following is a short overview of each of the implemented protocols. A more complete explanation of their use and programming is included in the application notes.

8.9.5.1 Ethernet Communications Using EGD Protocol (GE-Fanuc's Ethernet Global Data) (GE-EGD)

The Ethernet Global Data (EGD) protocol is a protocol invented by GE-Fanuc. It may be used to connect ADD-32 drives to GE-Fanuc PLCs and other EGD compliant hardware. This protocol is an unsolicited UDP message format. Instead of a "Server" or "Client", a node on the network can be a Producer or a Consumer. The data passed between nodes on the network are passed as lists known as Exchanges. The configuration of how these lists are produced and consumed is discussed in Appendix F. The Avtron ADD-32 drives are configured using Avtron's EGDConfig program, Avtron P/N 683247. The internal data structure of the drive as data is passed from the drive's data tables is illustrated in Figure 8.9.5.1.

NOTE

Additional information on using EGD protocol is provided in Appendix F.

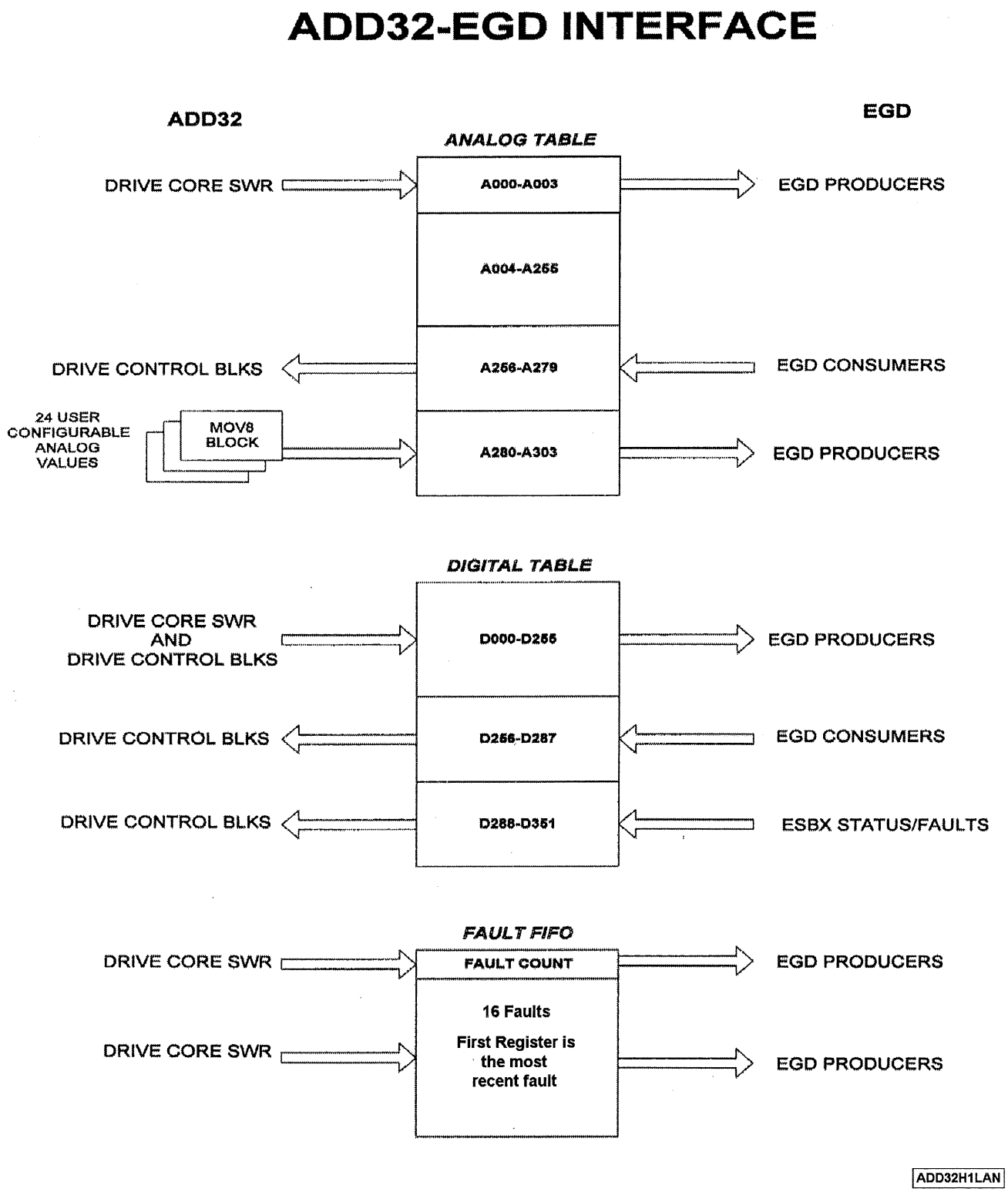


Figure 8.9.5.1 ADD-32 data transfers via EGD

EGD Exchanges are lists of data. Each node (each unique IP address) has its own list of exchanges that it can produce -- up to 32 exchanges of data. How these data lists are built and how they are interpreted at the consumer end are discussed in the Application Notes section. These Exchanges maybe configured as either Uni-Cast (Producer to a single Consumer) or Multi-Cast (Producer to a group of Consumers).

8.9.5.2 Schneider Electric's Modbus TCP/IP (MB-TCP)

The Modbus TCP/IP protocol was developed by Schneider Electric (Modicon) as an open standard for industrial ethernet communications. It is a Client/Server protocol of the "Master/Slave" variety, implemented via TCP/IP. This protocol can be used to connect ADD-32 drives to Modicon PLCs and other Modbus TCP/IP compliant hardware.

The Avtron implementation of Modbus TCP/IP (MB-TCP) supports the following Modbus Function Codes (MFC's):

1) Command Function Code= 03, Read Multiple Registers from ADD-32. This command is used to read a group of 1 to 63 consecutive registers (16 bit words) from a given address.
2) Command Function Code= 16, Write Multiple Registers to ADD-32. This command is used to write a group of 1 to 63 consecutive registers (16 bit words) to a given address.
3) Command Function Code= 01, Read Multiple Coils from ADD-32. This command is used to read a group of consecutive coils (bits) from a given address.
4) Command Function Code= 05, Write Coil to ADD-32. This command is used to write a single coil (1bit) to a given address.

NOTE

Additional information on Modbus TCP/IP is provided in Appendix G.

8.9.5.3 Allen-Bradley's Client Server Protocol (AB-CSP)

Allen-Bradley CSP Ethernet communications encapsulates the PCCC message structure (DF-1 style) into a packet, and delivers this packet to the PLC (or ADD-32). The CSP protocol is used predominately to talk to ADD-32 drives from PLC5 and SLC500 PLCs, programming typically done in RsLogix5 or RSLogix500. It may also be used from a ControlLogix PLC using RSLogix5000, but this is not recommended; the use of EtherNet/IP is suggested.

- For the PLC-5, the PCCC protocol supports PLC-5 Typed Writes, PLC-5 Typed Reads, and PLC-2 Unprotected Writes.
- For the SLC, the PCCC protocol supports SLC Protected Typed Logical Writes with three address fields available in the read and receive functions.

- For the ControlLogix, the PCCC protocol supports PLC-5 Typed Writes and PLC-5 Typed Reads.

NOTE

Additional information on using AB-CSP protocol is provided in Appendix H.

8.9.5.4 Ethernet Communications Using Allen-Bradley Ethernet IP, Control/Internet Protocol (AB-CIP)

Ethernet Industrial Protocol (Ethernet/IP) is an open industrial networking standard that supports implicit messaging (real-time I/O messaging) and explicit messaging (message exchange). Ethernet/IP is a further development of Allen Bradley's Common Industrial Protocol (CIP). The EtherNet/IP protocol is used to talk to ADD-32 drives from ControlNet or DeviceNet based PLCs, programming done with RSLogix5000.

Ethernet/IP consists of:

- IEEE 802.3 Physical and Data Link standard.
- Ethernet TCP/IP protocol suite (Transmission Control Protocol/Internet Protocol), the Ethernet industry standard.
- CIP, the protocol that provides real-time I/O messaging and information / peer-to-peer messaging. ControlNet and DeviceNet networks also use CIP.

TCP/IP is the transport and network layer protocol used by Explicit Messaging.

The UDP/IP (User Datagram Protocol) is used by Implicit Messaging.

NOTE

Additional information on using AB-CIP protocol is provided in Appendix I.

SECTION IX

SYSTEM DATA STRUCTURES

This section details the data structures of all necessary tables and data inputs used to set up and define operation of an ADDvantage-32. Some parameters are write protected, meaning the information is read only and cannot be written to. If a write is attempted on a protected area, an error message appears. All parameters, regardless of write-protect status, can be read over the serial link or 802.4 LAN.

For additional information on communicating to an ADDvantage-32, refer to Avtron Engineering Report 14363-001 for the RS-485 serial link or the Avtron ADDvantage-32 LAN Specification. These reports can be obtained upon request.

9.1 DATA TABLES

Information in the ADDvantage-32 is accessed using tables (Files). Each table contains a specific group of parameters or information associated with those parameters. For example, one table contains the numerical data for all the C*** parameters while a second table contains the units for these parameters.

9.2 DATA TABLE STRUCTURES

Data tables for the ADDvantage-32 start at number 9. Each element referenced in the table can be found by referencing its parameter number in Appendix C.

****

C A U T I O N

Data contained in the ADDvantage-32 data tables depends on the software part number and version being used by the unit. Each software part number uses different parameters and numbers. To locate a particular parameter within a table, find the table number that contains the parameter information required. Refer to Appendix C to locate the parameter number and its position in the table.

****
****

W A R N I N G

Particular locations have been set up for writing information to an ADDvantage-32. These locations are A***:LOCATION X for analog values and D***:LOC BIT X for digital values. It is highly recommended that all information written to an ADDvantage-32 be written to these locations. These parameters reside in regular system memory and can be written to often. If information is written to parameters such as C***, it is being written into the EEPROM memory which has a limited number of write cycles. Although writing to the EEPROM can occur thousands of times, it is not recommended.

****

Table 9-1 contains the data files (tables) available in the ADDvantage-32. The items listed in Table 9-1 are defined as follows:

1. File Number - Number of the table containing the data.

2. Table Name - Name of the table containing the data.

3. Structure - Number of bytes required for one element of data in the table.

4. Data Type - Data format such as:

   FP = Floating Point Numbers (Four bytes per data element)
   I = Integer (Two bytes per data element)
   LI = Long Integer (Four bytes per data element)
   A = ASCII (One byte per data element)

5. Data Mode - Data in the table is either RO or R/W.

   RO = Read Only (Write Protected)
   R/W = Read or Write (No Protection)

6. PAR - Represents the parameter group covered by the table.

TABLE 9-1. DATA FILES

File Number	Table Name	Structure Bytes/ Element	Data Type	Data Mode	(PAR)
9	ANALOG IO	4	FP	R/W	(A000-A***)
10	ADT LABELS	11	A	RO	(A000-A***)
11	ADT UNITS	6	A	R/W	(A000-A***)
12	DEFAULT ADT UNITS	6	A	RO	(A000-A***)
13	CAL TABLE UNITS	6	A	RO	(T000-T127)
14	CAL TABLE DEFAULT	4	FP	RO	(T000-T127)
15	CAL TABLE LOW LIMITS	4	FP	RO	N/A
16	CAL TABLE HIGH LIMITS	4	FP	RO	N/A
17	CAL TABLE	4	FP	R/W	(T000-T127)
18	CONTROL CAL LABELS	11	A	RO	(C000-C***)
19	CONTROL CAL HIGH LIMITS	4	FP	RO	(C000-C***)
20	CONTROL CAL LOW LIMITS	4	FP	RO	(C000-C***)
21	CONTROL CAL DEFAULTS	4	FP	RO	(C000-C***)
22	CONTROL CONFIG LEGALS	2	I	RO	(P000-P***)
23	CONTROL CONFIG LABELS	11	A	RO	(P000-P***)
24	CONTROL CAL UNITS	6	A	R/W	(C000-C***)
25	CONTROL CONFIG	2	I	R/W	N/A
26	DEFAULT CONTROL CAL UNIT	6	A	RO	(C000-C***)
27	CONTROL CAL	4	FP	R/W	(C000-C***)
28	DIGITAL I/O	2	I	R/W	(D000-D***)
29	DDT LABELS	11	A	RO	(D000-D***)
30	DRIVE CAL LABELS	11	A	RO	(X000-X***)
31	DRIVE CAL UNITS	6	A	RO	(X000-X***)
32	DRIVE CAL HIGH LIMITS	4	FP	RO	(X000-X***)
33	DRIVE CAL LOW LIMITS	4	FP	RO	(X000-X***)
34	DRIVE CAL DEFAULTS	4	FP	RO	(X000-X***)
35	DRIVE CONFIG LABEL	11	A	RO	(Y000-Y***)
36	DRIVE CONFIG LEGALS	2	I	RO	(Y000-Y***)
37	DRIVE CONFIG	2	I	RO	(Y000-Y***)
38	DRIVE CAL	4	FP	R/W	(X000-X***)
39	SELECT LIST	11	A	RO	N/A
40	CHECK SUM	4	LI	RO	N/A
41	CAL VERSION	2	I	RO	N/A
42	DRIVE SOFTWARE AND VERSION NUMBER	2	I	RO	N/A
43	TIME ARRAY	4	LI	R/W	N/A
44	ANALYZER CAL LABELS	11	A	RO	(Z002-Z006) (Z102-Z106) (Z202-Z206) (Z302-Z306)

File Number	Table Name	Structure Bytes/ Element	Data Type	Data Mode	(PAR)
45	ANALYZER CAL HIGH LIM	4	FP	RO	N/A
46	ANALYZER CAL LOW LIMITS	4	FP	RO	N/A
47	ANALYZER CAL DEFAULTS	4	FP	RO	N/A
48	ANALYZER CAL	4	FP	R/W	N/A
49	ANALYZER CONFIG PROMPT	11	A	RO	(Z002-Z006) (Z102-Z106) (Z202-Z206) (Z302-Z306)
50	ANALYZER CONFIG LEGALS	2	I	RO	N/A
51	ANALYZER CONFIG	2	I	R/W	(Z007-Z0**) (Z107-Z1**) (Z207-Z2**) (Z307-Z3**)
52	DIAGNOSTIC STATUS CHAN-1	2	I	RO	N/A
53	DIAGNOSTIC STATUS CHAN-2	2	I	RO	N/A
54	DIAGNOSTIC STATUS CHAN-3	2	I	RO	N/A
55	DIAGNOSTIC STATUS CHAN-4	2	I	RO	N/A
56	DIAG. TRACE DATA CHAN-1	4	FP	RO	N/A
57	DIAG. TRACE DATA CHAN-2	4	FP	RO	N/A
58	DIAG. TRACE DATA CHAN-3	4	FP	RO	N/A
59	DIAG. TRACE DATA CHAN-4	4	FP	RO	N/A
60	RETURN FAULTS	2	I	RO	N/A
61	KEYBOARD/DISPLAY MODE	2	I	R/W	N/A
62	VIRTUAL KEYBOARD DISPLAY	1	A	R/W	N/A
63	DOWNLOADING	2	I	R/W	N/A
64	RETURN FAULTS	2	I	R/W	N/A
65	ARM TRACE	2	I	R/W	N/A
66	TRACE	2	I	RO	N/A
67	FTRACE	2	I	RO	N/A
68	LINK CONTROL BITS	2	I	R/W	N/A

9.3 FILE NUMBER DESCRIPTION

The files listed in Table 9-1 are described as follows.

9.3.1 **FILE 9 - ANALOG I/O**

This file defines the analog data table where computed analog data values are stored in floating point format. This table contains application specific data where size and definition are functions of the software part number and the drive application.

9.3.2 **FILE 10 - ADT LABELS**

Contains all of the predefined label definitions for each analog input listed in File 9.

Typical values could be:

First element - FIL_SPEED__ (spaces are shown as __)
Next element - FIL_ARM_CUR

Reading the first element of File 10 would return the following data
..... 46H 49H 4CH 20H 53H 50H 45H 45H 44H 20H 20H ....
F I L S P E E D

9.3.3 **FILE 11 - ADT UNITS**

Contains the units associated with each of the labels defined in File 10.

Typical values could be:

First element - __FPM_

Reading this element would return the following data
..... 20H 20H 46H 50H 4DH 20H .....
F P M

This file may be written to and used to modify the units display. Sending a write message for this table and the first element of
..... 20H 20H 59H 50H 4DH 20H .....
Y P M
would modify the display units to be YPM instead of FPM.

9.3.4 **FILE 12 - DEFAULT ADT UNITS**

This file contains the default units for the corresponding analog input elements.

9.3.5 **FILE 13 - CAL TABLE UNITS**

This file contains all of the calibration unit definitions for the corresponding calibration element of the X-Y taper tables. This table organization differs from the normal organization. The corresponding calibration tables, File 14 and File 17, contain 128 elements total, but they are organized as four sub-tables of 32 elements each. (Refer to Files 14 and 17.)

9.3.6 **FILE 14 - CAL TABLE DEFAULTS**

This file contains all of the predefined default values for calibration.

The organization of this file differs in that the complete table is broken up into four sub-files. Each sub-file is

organized as one floating point value grouped in sub-files of 32 elements each. For example:

Sub-File 0	Value
element 0	FP
1	FP
.	.
element 31	FP

Sub-File 1	Value
element 32	FP
.	.
.	.
element 63	FP

9.3.7 **FILE 15 - CAL TABLE LOLIM**

This file contains the one predefined low limit value for all calibration entries. The element contains the value of -9999.0 as the low limit value. Any calibration entry must be greater than this value.

9.3.8 **FILE 16 - CAL TABLE HILIM**

This file contains the one predefined high limit value for all calibration entries. The element contains the value of +9999.0 as the high limit value. Any calibration entry cannot exceed this maximum value.

9.3.9 **FILE 17 - CAL TABLE**

This file contains all of the calibration values for each entry in the calibration file. Before writing any value to this file, check the value against the low and high limit values to insure that it is within limits. The write procedures in the receiving station do not check to see if a value is within limits. The organization of this file also differs in that the complete file is broken up into four sub-files. Each sub-file is organized as one floating point value grouped in 32 elements. For the file organization, refer to File 14.

On default initialization, the default values are copied from File 14 into this File. The user may modify the values which will then be used at all future power up conditions, unless the default power up is repeated.

9.3.10 **FILE 18 - CONTROL CAL LABELS**

This file contains all predefined label definitions for the control calibration. One element is used to produce one label. For example:

First element - ZERO_ANALOG (spaces are shown as __)
Next element - ONE_ANALOG_

Reading the first element returns the following data:
..... 5AH 45H 52H 4FH 20H 41H 4EH 42H 4CH 4FH 47H ....
Z E R O A N A L O G

9.3.11 **FILE 19 - CONTROL CAL HI LIMS**
This file contains all of the predefined high limit values for each control calibration value.

9.3.12 **FILE 20 - CONTROL CAL LO LIMS**
This file contains all of the predefined low limit values for each control calibration value.

9.3.13 **FILE 21 - CONTROL CAL DEFAULTS**
This file contains all of the predefined default values for control calibration.

9.3.14 **FILE 22 - CONTROL CONFIG LEGALS**
This file contains the drive configuration legal and is organized as a two-dimensional array of eight integer values per element with 139 elements maximum. Each element of eight values represents the legal choices that may be selected for that particular element of configuration in File 25.

9.3.15 **FILE 23 - CONTROL CFG LABELS**
This file contains the drive configuration labels. One element is used to produce one label.

For example:
First element - JOG REF
Next element - MASTER REF

9.3.16 **FILE 24 - CONTROL CAL UNITS**
This file contains the units associated with each of the labels defined for the control calibration values. One element is used to define one unit.

9.3.17 **FILE 25 - CONTROL CONFIG**
This file contains the control configuration data for the particular drive software version and part number.

9.3.18 **FILE 26 - DEFAULT CONTROL CAL UNITS**
This file contains all of the predefined default units for control calibration. One element is used to define one unit definition.

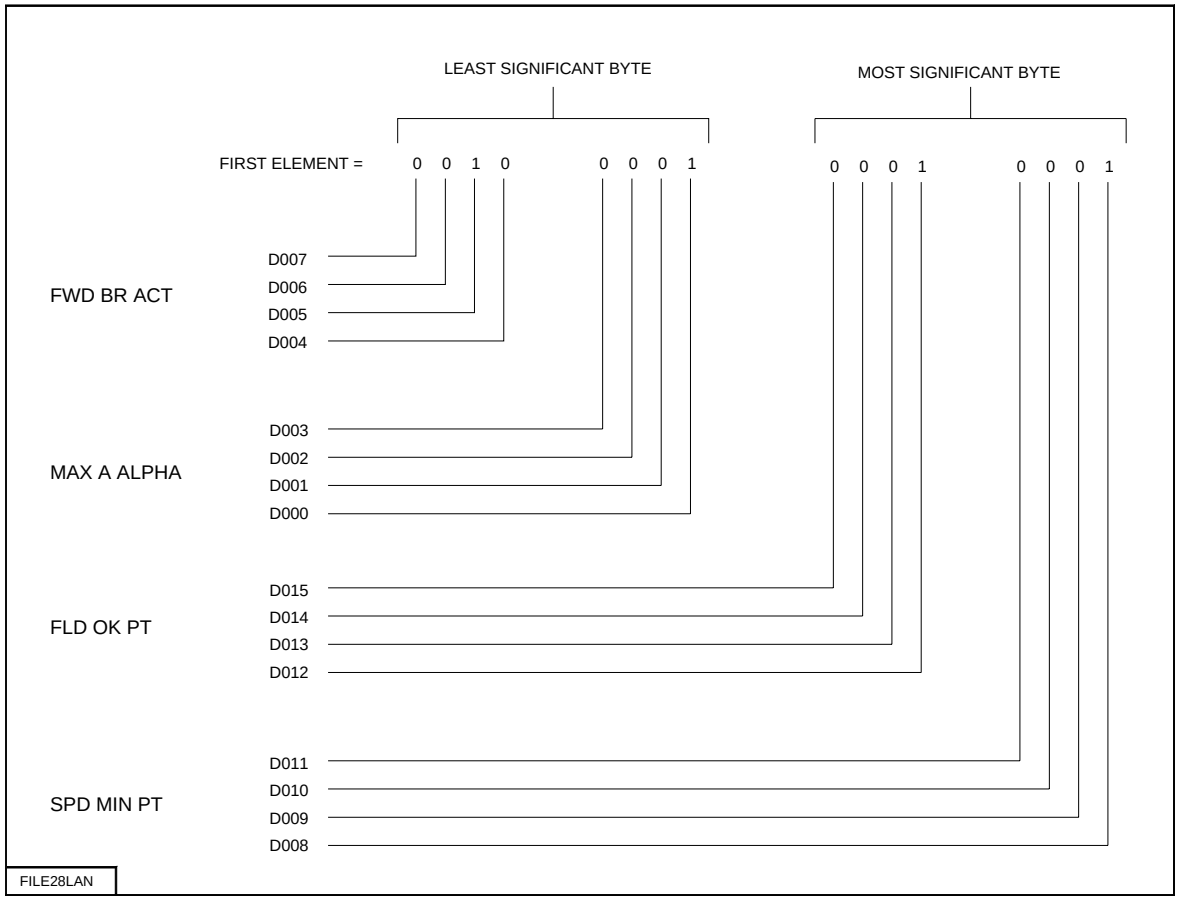
For example:
First element - UNITS
Next element - UNITS

9.3.19 **FILE 27 - CONTROL CAL**
This file contains all of the calibration values for each control calibration entry. Each element contains one calibration value in floating point format. This file will

be loaded with the contents of File 21 at initial power up, and if the drive is defaulted.

9.3.20 **FILE 28 - DIG IO**

This file defines the digital data table. Each bit in the structure represents real time digital data. A bit set equal to one (1) indicates an "ON" condition, while a zero (0) indicates an "OFF" condition.



The illustrated example for File 28 shows that the Forward Bridge is active and maximum firing angle has been reached. Additionally, the motor field feedback is within the 10% setpoint and the drive is configured for the contactor to remain sealed (upon RUN removal) until ZERO SPEED.

9.3.21 **FILE 29 - DDT LABELS**

This file contains all predefined label definitions for the digital data table.

One element is used to produce one label. For example:

First element - MAX_A_ALPHA (spaces are shown as __)
Next element - MIN_A_ALPHA

9.3.22 **FILE 30 - DRIVE CAL LABELS**

This file contains all predefined label definitions for the drive calibration. One element is used to produce one label.

For example:
First element - I/O_V-REF
Next element - MOTOR_IARM

9.3.23 **FILE 31 -DRIVE CAL UNITS**

This file contains the units associated with each of the labels defined in Table 30 for drive calibration. One element is used to define the units for the associated label.

For example:
First element - VOLTS
Next element - %_DRIVE

9.3.24 **FILE 32 - DRIVE CAL HI LIMS**

This file contains all of the predefined high limit values for each drive calibration value. Each element contains one high limit value.

9.3.25 **FILE 33 - DRIVE CAL LO LIMS**

This file contains all of the predefined low limit values for each drive calibration value. Each element contains one low limit value.

9.3.26 **FILE 34 - DRIVE CAL DEFAULTS**

This file contains all of the predefined default values for drive calibration. Each element contains one default drive calibration value.

For example:
First element - -7.00
Next element - 10.0

9.3.27 **FILE 35 - DRIVE CONFIG LABEL**

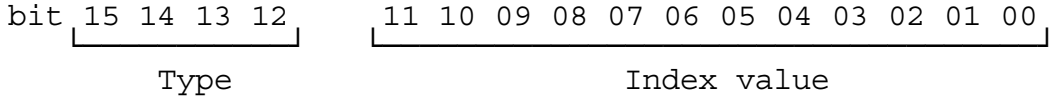
This file contains the drive configuration labels displayed during drive configuration. Each element contains one label.

For example:
First element - DRIVE_PN
Next element - BRIDGE_SELFTEST

9.3.28 **FILE 36 - DRIVE CONFIG LEGALS**

This file contains the drive configuration legal options.

The two bytes per element are grouped as shown:



The upper four bits define the type assigned, while the lower 12 bits define the index into the file elements defined by the type. For example, if the type bits were 03, then the index value would be the offset into the drive calibration table. Examples of type bits are as follows:

TYPE	File
0	Select List
1	Analog Data Table
2	Digital Data Table
3	Drive Calibration
4	Control Calibration
5	Drive Configuration
6	Control Configuration

9.3.29 **FILE 37 - DRIVE CONFIG**

This file contains the drive configuration settings. The file is organized the same as File 36.

For example:
First element - DRIVE_PN
Next element _ BRIDGE_SELFTEST

9.3.30 **FILE 38 - DRIVE CAL**

This file contains all of the calibration values for each drive calibration entry. At initial power up, or a power up with default conditions request, the default values in File 34 will be written into this file.

For example:
First element - AD_REF
Next element - IARM_SPAN

9.3.31 **FILE 39 - SELECT LIST LABELS**

The select list table contains ASCII data that may be read, but not written to.

Each element in this table contains generic alpha labels that may be displayed for a number of different variables. Some examples are as follows:
First element __DISABLED_
Next element __ENABLED__

9.3.32 **FILE 40 - CHECKSUM AREA**

This file contains the checksum values for each of the six calibration and configuration areas.

Each element is an unsigned long integer value. (Refer to section 6.2 of ER 14363-001 for recovery of long integers.) The checksum and element assignment are as follows:

CHECKSUM AREA	ELEMENT NUMBER
Drive Calibration	0
Control Calibration	1
Drive Configuration	2
Control Configuration	3
Calibration	4
Analyzer Calibration	5
Analyzer Configuration	6

9.3.33 **FILE 41 - CAL VERSION**

This file consists of one integer element containing the version number of the calibration data.

9.3.34 **FILE 42 - SOFTWARE PART NUMBER AND VERSION NUMBER**

This file contains the part number and version number for the particular version installed in the unit.

The table is organized as two bytes per element with two integer elements. They are organized as:

Part Number	Element 0
Version	Element 1

Reading this file would return the part number and version number of the installed software.

9.3.35 **FILE 43 - TIME ARRAY**

This file is organized as two long integer values of four bytes each. The lower element value is a time reference that may be written with a time reference value by the master station. If it is not written to, it will default to a value of zero (0). The upper element value will be incremented six times for each line cycle after the time reference is set. Each element is an unsigned long integer value. (Refer to section 6.2 of ER 14363-001 for recovery and writing to elements containing long integers.)

Integer 0 - Time reference value
Integer 1 - Time intervals since last time reference.

9.3.36 **FILE 44 - ANALYZER CAL LABELS**

This file contains the analyzer calibration labels. Each of the four groups (Channels 1 through 4) is made up of eight elements.

For example:
First element - TRIG LEVEL 1
Next element - RATE 1

NOTE

While scrolling through the ADDvantage-32™ keypad in the ANALYZER SETUP menu, parameters Z000:RESET 1 and Z001:OUTPUT 1 appear. Z000, Z001, Z100, Z101, Z200, etc., are not included elements in this file. The RESET and OUTPUT functions are only available when executed on the keypad.

9.3.37 **FILE 45 - ANALYZER CAL HI LIMS**

This file contains the analyzer calibration high limit values. This file contains eight floating point values for each of the four channels in File 44. Data limits for each channel are identical. Elements 0-4 correspond to (Z002-Z006). Elements 8-12 correspond to (Z102-Z106), etc.

EXAMPLE: **ANALYZER CAL HI LIMS - FILE 45**

Analyzer 1		Analyzer 2	
Element		Element	
0	9999.0	8	9999.0
1	9999.0	9	9999.0
2	2999.0	10	2999.0
3	9999.0	11	9999.0
4	9999.0	12	9999.0
5	0.0	13	0.0
6	0.0	14	0.0
7	0.0	15	0.0

Analyzer 3		Analyzer 4	
Element		Element	
16	9999.0	24	9999.0
17	9999.0	25	9999.0
18	2999.0	26	2999.0
19	9999.0	27	9999.0
20	9999.0	28	9999.0
21	0.0	29	0.0
22	0.0	30	0.0
23	0.0	31	0.0

9.3.38 **FILE 46 - ANALYZER CAL LO LIMS**

This file contains the analyzer calibration low limit values for each of the four channels in File 44. Data limits for each channel are identical. See Section V.

9.3.39 **FILE 47 - ANALYZER CAL DEFAULTS**

This file contains the analyzer calibration default values for each of the four channels in File 44 that will be loaded into calibration on initial power up. Data values for each channel are identical. See Section V.

9.3.40 **FILE 48 - ANALYZER CAL**

This file contains the actual calibration values used for each of the four channels in File 44 to process the collected analyzer data. The default values initially used for each channel are identical. On default initialization, the default values from File 47 are copied into this file. The user may modify the values which will then be used at all future power up conditions unless the default initialization is repeated. See Section V.

9.3.41 **FILE 49 - ANALYZER CONFIG LABEL**

This file contains the analyzer configuration prompts. (Refer to File 51.)

9.3.42 **FILE 50 - ANALYZER CONFIG LEGALS**

This file contains analyzer configuration legal data.

This file contains the legal references that may be used to configure control for each channel. This table must be used to generate configuration data changes. Any data reference not contained in this table will be invalid, and control action will be undefined. The table is organized as a two-dimension array with eight (8) by nineteen (19) unsigned short integer values for each of the four channels. Each of the tables consists of a maximum of eight choices for each of the nineteen variables.

NOTE

See Appendix A of Serial Link Specification, ER 14363-001 - Table 50 for valid values, and note that some are unused.

Channel 1:

choice 0 choice 1 .....choice 8

VARIABLE

```
Element 0      (SELECT_TYPE + RISING),.............,EOLST
Element 1                .    .                        .
     .                   .    .                        .
     .                   .    .                        .
     .                   .    .                        .
     .                   .    .                        .
Element 18     (SELECT_TYPE + FALLING),.............,EOLST
```

9.3.43 **FILE 51 - ANALYZER CONFIG**

This file contains analyzer configuration data. Writing to this table changes the configuration of the channel being monitored and the control variables for each of the four channels. The valid control configuration data is contained in File 50. This file is organized as 19 unsigned short

integer values for each of the four channels, for a total of 76 elements.

Channel 1:

VARIABLE
First element - ENABLE_REC1
Next element - INP1_A/D
.
.
.
Last element
Channels 2 through 4 are identical in form.

9.3.44 **FILES 52 THROUGH 55 - DIAGNOSTIC STATUS TABLES**

The diagnostic status tables allow recovery of the status of each of the four diagnostic channels, Files 56 through File 59. Refer to section 8.19 of ER 14363-001. The complete status data must be recovered to determine the status and state of the diagnostic data table before reading any data table. The data from the data tables may not be read until the status indicates that the data sampling and storage is complete and ready for recovery. The file number assignments are as follows:

File 52	-	The status of channel 1 - Data file 56
File 53	-	The status of channel 2 - Data file 57
File 54	-	The status of channel 3 - Data file 58
File 55	-	The status of channel 4 - Data file 59

Each of the four status files are organized as a collection of integers and floating point data elements.

52	DIAGNOSTIC STATUS- CHAN 1	2	I	RO
53	DIAGNOSTIC STATUS- CHAN 2	2	I	RO
54	DIAGNOSTIC STATUS- CHAN 3	2	I	RO
55	DIAGNOSTIC STATUS- CHAN 4	2	I	RO

9.3.45 **FILES 56 THROUGH 59 - DIAGNOSTIC TRACE DATA TABLES**

56	DIAGNOS. TRACE DATA- CHAN 1	4	FP	RO
57	DIAGNOS. TRACE DATA- CHAN 2	4	FP	RO
58	DIAGNOS. TRACE DATA- CHAN 3	4	FP	RO
59	DIAGNOS. TRACE DATA- CHAN 4	4	FP	RO

9.3.46 **FILE 60 - FAULT TABLE**

This file contains the drive fault FIFO and is organized as 16 short integer elements. Each element may contain one fault value detected by the drive.

60	RETURN FAULTS	2	I	RO

Each table element is defined as follows:

bit	15 14 13 12	11 10 09 08 07 06 05 04 03 02 01 00
	Type	Index value

The upper four bits define the type assigned, while the lower 12 bits define the index into the table defined by the type. For example, if the type bits were 00, then the index value would be the offset into the Select List table. The defined types and tables are as follows:

TYPE	TABLE
0	Select List
1	Analog Data Table
2	Digital Data Table
3	Drive Calibration
4	Control Calibration
5	Drive Configuration
6	Control Configuration
7	Reserved
8	Unused
9	Unused
10	Unused
11	Unused
12	Unused
13	Unused
14	Unused
15	Unused

NOTE

For File 60, the type bits must be set at 00 to view faults. Faults only reside in the Select List table.

9.3.47 FILE 61 THROUGH 62 KEYBOARD/DISPLAY MODE TABLES

61	KEYBOARD/ DISPLAY MODE	2	I	R/W

NOTE

When writing to file 61 using the Allen-Bradley Client Server Protocol (AB-CIP), the data packet received by the ADDvantage32 will be regarded as an AutoScan Write, with the expected data structure as given within Table 9.3.47-1 below. A sample RSLogix 5™ MSG (message) block configuration for performing an AutoScan Write is given in Figure 9.3.47-1. Reference Appendix H for further information regarding the AB-CSP protocol.

61	AUTOSCAN WRITE	Up to 24	I	W

Expected Data Packet Format:

Word.Bit	Description:
0	Analog Setpoint Data Starting Address (Word 1)
1	Analog Setpoint Data Starting Address (Word 2)
2	Digital Setpoint Data Starting Address (Word 1)
3	Digital Setpoint Data Starting Address (Word 2)
4	Digital Setpoint to ADDvantage32
4.1	Digital Setpoint to ADDvantage32
4.2	Digital Setpoint to ADDvantage32
4.3	Digital Setpoint to ADDvantage32
4.4	Digital Setpoint to ADDvantage32
4.5	Digital Setpoint to ADDvantage32
4.6	Digital Setpoint to ADDvantage32
4.7	Digital Setpoint to ADDvantage32
4.8	Digital Setpoint to ADDvantage32
4.9	Digital Setpoint to ADDvantage32
4.1	Digital Setpoint to ADDvantage32
4.11	Digital Setpoint to ADDvantage32
4.12	Digital Setpoint to ADDvantage32
4.13	Digital Setpoint to ADDvantage32
4.14	Digital Setpoint to ADDvantage32
4.15	Digital Setpoint to ADDvantage32
5	RESERVED 0x00
6	Number of Analog Setpoints (Word 1)
7	Number of Analog Setpoints (Word 2)
8 & 9	Analog Setpoint to ADDvantage32
10 & 11	Analog Setpoint to ADDvantage32
12 & 13	Analog Setpoint to ADDvantage32
14 & 15	Analog Setpoint to ADDvantage32
16 & 17	Analog Setpoint to ADDvantage32
18 & 19	Analog Setpoint to ADDvantage32
20 & 21	Analog Setpoint to ADDvantage32
22 & 23	Analog Setpoint to ADDvantage32

Table 9.3.47 - 1

- The Analog Setpoint Data Starting Address refers directly to the analog data table address where the Analog Setpoint values will be consecutively written. The Analog Setpoint Data Starting Address is interpreted as a 32-bit integer, with Word 1 being the least significant, Word 2 as most significant. For example, placing a value of 42 in Word 1 a value of 0 in Word 2, the analog data table address will be identified as A042:. Typical applications use A###:LOCATION 1 through A###:LOCATION 8 for this purpose. The analog setpoints must be of 32-bit IEEE 485 format, little endian, and occupy two consecutive 16-bit integer registers at the locations given in Table 9.3.47-1. The number of Analog Setpoint values written is scalable using the Number of Analog Setpoints parameter. The Number of Analog Setpoints is interpreted as a 32-bit integer, with Word 1 being the least significant, Word 2 as most significant. This value can be up to 8 floating point values and will determine the overall size of the message. As an example, if only two analog values are intended to be sent as part of the AutoScan Write message, the Number of Analog Setpoints parameter shall be set to two (2) and the total message length transferred to the ADDvantage32 should therefore be 12 integer words in length.

- The Digital Setpoint Data Starting Address refers to the digital data table address where the Digital Setpoint bits will be consecutively written. The Digital Setpoint Data Starting Address is interpreted as a 32-bit integer, with Word 1 being the least significant, Word 2 as most significant. For example, placing a value of 96 in Word 1 a value of 0 in Word 2, the digital data table address will be identified as D096:. Typical applications use D###:LOC BIT 1 through D###:LOC BIT 16 for this purpose. The number of digital setpoints is fixed at 16 and occupy integer word #4 within the AutoScan Write message.

RSLogix 5™ MSG configuration for AutoScan Write:

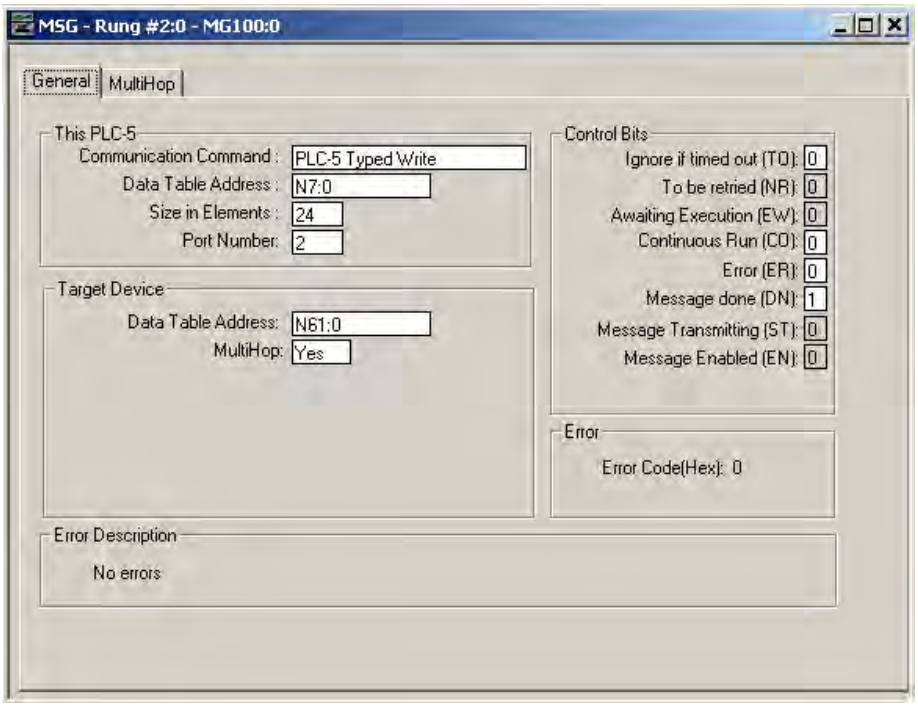


Figure 9.3.47 - 1

NOTE

Reading File 61 using the Allen-Bradley Client Server Protocol (AB-CIP), will be regarded as an AutoScan Read by the ADDvantage32, with the return data packet structured as given in Table 9.3.47-2 below. A sample RSLogix 5™ MSG (message) block configuration for performing an AutoScan Read is given in Figure 9.3.47-1. Reference Appendix H for further information regarding the AB-CSP protocol.

61 AUTOSCAN READ Up to 36 I R

Response Data Packet Format:

Word.Bit	Description:
0 & 1	Analog Feedback from ADDvantage32: A000:FIL SPEED
2 & 3	Analog Feedback from ADDvantage32: A001:FIL ARM CUR
4 & 5	Analog Feedback from ADDvantage32: P###:AUTOSCAN 1
6 & 7	Analog Feedback from ADDvantage32: P###:AUTOSCAN 2
8 & 9	Analog Feedback from ADDvantage32: P###:AUTOSCAN 3
10 & 11	Analog Feedback from ADDvantage32: P###:AUTOSCAN 4
12 & 13	Analog Feedback from ADDvantage32: P###:AUTOSCAN 5
14 & 15	Analog Feedback from ADDvantage32: P###:AUTOSCAN 6
16	Digital Feedback from ADDvantage32: D000: thru D015:
17	Digital Feedback from ADDvantage32: D016: thru D031:
18	Digital Feedback from ADDvantage32: D032: thru D047:
19	Digital Feedback from ADDvantage32: D048: thru D063:
20	Digital Feedback from ADDvantage32: D064: thru D079:
21	Digital Feedback from ADDvantage32: D080: thru D095:
22	Digital Feedback from ADDvantage32: D096: thru D111:
23	Digital Feedback from ADDvantage32: D112: thru D127:
24	Digital Feedback from ADDvantage32: D128: thru D143:
25	Digital Feedback from ADDvantage32: D144: thru D159:
26	Digital Feedback from ADDvantage32: D160: thru D175:
27	Digital Feedback from ADDvantage32: D176: thru D191:
28	Digital Feedback from ADDvantage32: D192: thru D207:
29	Digital Feedback from ADDvantage32: D208: thru D223:
30	Digital Feedback from ADDvantage32: D224: thru D239:
31	Digital Feedback from ADDvantage32: D240 thru D255:
32	Digital Feedback from ADDvantage32: D256: thru D271:
33	Digital Feedback from ADDvantage32: D272: thru D287:
34	Digital Feedback from ADDvantage32: D288: thru D303:
35	Digital Feedback from ADDvantage32: D2304: thru D319:

Table 9.3.47 - 2

- The Analog Feedback data will be returned as 32-bit IEEE 485 format, little endian and occupy two consecutive 16-bit integer registers at the locations given in Table 9.3.47-2. The number of Analog Feedback values is fixed at 8.

- The Digital Feedback data starts at the beginning of the digital data table (D000:), in 16-bit integral blocks who's length is determined by the requested length of the AutoScan Read message. For example, if an AutoScan Read with a length of 20 registers is requested, the ADDvantage32 will return a packet 20 integers in length, where the first 16 integers contain Analog Feedback data and the remaining four integers contain Digital Feedback data D000: through D063:

RSLogix 5™ MSG configuration for AutoScan Read:

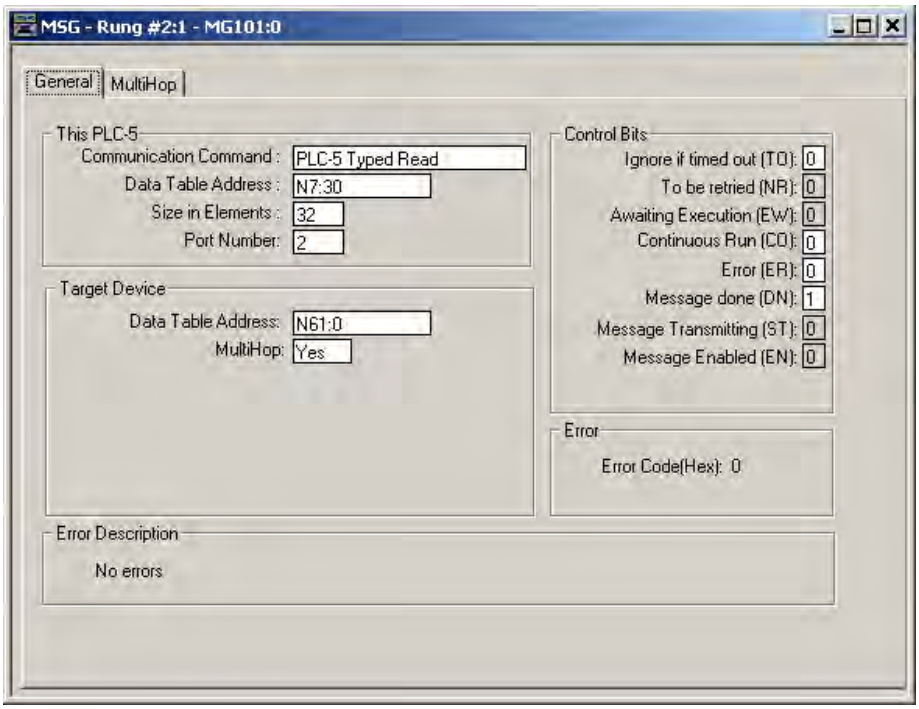


Figure 9.3.47 - 2

62	VIRTUAL KEYBOARD DISPLAY	1	A	R/W

9.3.48 **FILE 63 THROUGH 64 _MISC. STATUS FILES**

63	DOWNLOADING	2	I	R/W
64	RETURN FAULTS	2	I	R/W

9.3.49 **FILE 65 THROUGH 68 DIAGNOSTIC TRACE CONTROL FILES**

65	ARM TRACE ENABLE	2	I	R/W
66	TRACE	2	I	RO
67	FTRACE	2	I	RO
68	LINK CONTROL	2	I	R/W