<u>///AMASSER PAIM</u>
Pliant Analog Input Module (SDI-12 Device)

User Reference Manual

Hardware Revision A
Firmware Revision #100

Updated: October 26,1999

http://www.amassdata.com

TABLE OF CONTENTS

PAIM
SDI-12 Pliant Analog Input Module
Pliant Technology Specialists

Pliant: readily yielding to influence

1. PAIM: Pliant Analog Input Module

The AMASSER PAIM is an analog input module in the form of a standalone SDI-12 device providing A/D conversion of analog inputs as well as excitation ports that are configurable to your needs. This A/D converter is a standard feature on our PDAS-II data loggers.

The PAIM is configured for monitoring the following maintenance parameters:

- Primary Input Voltage of the system (0-20 V input)
- Internal Temperature of the system

The inputs also include two external 0-2.5V, one external 0-20V and three external 0-5V. The user can configure these external inputs by defining a custom equation to convert the raw data to a desirable engineering unit. The A/D provides these inputs with 18 bit resolution and buffered bipolar/unipolar ranges of 25mV, 55mV, 100mV, 1V, 2.5V and 5V.

The PAIM provides the following excitation :

- 2.5 V voltage reference
- 5 V voltage reference
- 250 microAmp constant current source

The constant current source is very accurate and temperature-compensated, which provides an ideal excitation for sensors such as RTDs by resolving the problems associated with temperature fluctuations as a result of excessive current draw.

It also includes 2 event counters. The event counters are 16 bit (0 to 65535) and are practical and adaptable to your needs. They comprise mode-selection that allows the user to configure the firmware for counting switch closures or pulses as well as to enable or disable the counter-reset function. These modes are named as follows: Switch Closure, Pulse, Read/Reset, and Overflow. In addition, they feature a setup parameter called *scale* which allows the user to define the value of each count.

Hardware is powered from a +10.5 to 15 V input for external battery, charger or power supply. A 256byte EEPROM provides non-volatile storage of set-up and device address data.   This set-up includes factory calibration which is stored in the EEPROM. Upon power-up the processor loads the EEPROM calibration values into the Gain and Offset Registers of the A/D converter.

The PAIM is a SDI-12 device and as such communicates with the DCP or portable computer using the commands and guidelines set out in that protocol. Included in this datasheet are the SDI-12 commands as they are implemented in the PAIM. **For detailed information concerning SDI-12 specifications visit the SDI-12 Support Group's web-site from our home page at http://www.amassdata.com.**

2. __SDI-12 Protocol__

The PAIM is an SDI-12 device and as such requires a controller which is compliant with the SDI-12 communications protocol. The Master SDI-12 controller, which may be a portable computer or a DCP equipped with a SDI-12 serial interface, is used for set-up and operation of the PAIM. Commands available to the PAIM are listed in *3. SDI-12 PAIM Sensor Commands*. A single computer or controller (logger) can supervise an array of up to 10 SDI-12 sensors along the 3-wire SDI-12 cable.

In order to communicate with the PAIM, or any other SDI-12 sensor, the controller sends a "break" signal which consists of sending a spacing signal (5 V) for more than 12 millisec. This will wake all sensors connected to the line. The controller then sends the appropriate command for the desired action (refer to *SDI-12 PAIM Sensor Commands*). The format of each byte of data is 1 start bit, 7 data bits with the least significant bit first, 1 parity bit with even parity, and 1 stop bit. The first significant byte of any message is the address of the sensor; all sensors with different addresses return to the "sleep" state and ignore the rest of the command. The last character of a command is always "!", and the controller turns off its transmitter within 7.5 millisec after sending the "!". The sensor responds by setting the data line to marking (0 V) for 8.33 millisec, and then begins its response. The first bit of the response must be sent within 15 millisec of receiving the last byte of the command.

All communication with the PAIM takes place as described above. For example, if the controller is requesting the current value of the Primary Voltage it would begin by sending a "break" signal as described above. It would then send the appropriate command which in this case would be "aM0!" (see 3.2.1 Convert both 0-20 V Inputs). The PAIM would perform the conversion as requested and respond to the controller in accordance with the SDI-12 specification.   As shown in section 3.2.1, this response would consist of the address of the sensor, the number of seconds of delay before data may be retrieved (one second for "aM0!) as well as the number of values stored in the output buffer as a result of the command (two for "aM0!). The controller then issues the "Retrieve Data" (aD!) command (section *3.2.5 Return Data Buffer*) so that the contents of the PAIM output buffer are sent to it. The format of this response would be as follows:

 a+<external voltage>+<primary voltage><cr><lf>
where a: address of the sensor (0 to 9)

Note that the "aM0!" command returns the value of both the Primary Voltage and the external 0-20 V input, that is, channels #6 and #7 (see section *5.3 Connectors* for channel number and pin assignments of the 8 analog inputs). Please refer to Appendix A for a list of the SDI-12 commands supported by the PAIM.

For detailed information on SDI-12 protocol visit the SDI-12 Support Group's web-site via our home page at http://www.amassdata.com.

3. SDI-12 PAIM Sensor Commands

The PAIM is SDI-12 compatible and as such follows the guidelines set out in that protocol.

3.1. SDI-12 Command Symbols

a : address byte

! : Terminates the command

<cr> : The carriage return character which is represented as '\0d' in the ASCII string.

<lf> : The line feed character which is represented as '\0a' in the ASCII string. All responses are terminated with <cr><lf>.

Command bytes : one or two byte ASCII character command. Ex.: M0

Identification String : "12AMASS Data PAIMxxx" where xxx = Revision Level

3.2. Retrieve Data Commands

3.2.1. Convert both 0-20 V Inputs

- command: aM0!
- response: a0012<cr><lf>   data buf: +<external voltage>+<primary voltage>

This command instructs the PAIM unit to convert analog inputs #6 and #7 (refer to section 5.3 Connectors). The SDI-12 format for the response to this command is : atttn<cr><lf>. As always 'a' is the address byte, 'ttt' is the specified time, in seconds, until the sensor will have the measurement(s) ready and 'n' is the number of measurement values the sensor will make and return in one or more subsequent D commands (see 3.2.5 Return Data Buffer below).   The implementation of this command to the PAIM yields the response shown above. 'ttt' indicates that the measurement is ready for retrieval in one second and 'n' shows that there are two values (0-20 V external voltage and primary voltage) waiting in the buffer. The measurement is then retrieved using the D! command.

Note that the "divide-by-4" circuitry that allows a range of 0 to 20 V for these inputs is jumper-selectable. Therefore, by removing jumpers 'E4' and 'E3' respectively, channels #6 and #7 become 0 to 5 V inputs.

Example:

The current host address is 0 and you wish to measure the Primary Voltage and the 0-20V external input.

Command: 0M! or 0M0!
Response: 00012\0d\0a

This response indicates that two values will be available for retrieval from the buffer in one second. The retrieval is accomplished with the D command. The issuance of the D command is done automatically by your logger provided that it is SDI-12 compliant. Compliant loggers will follow-up with the D command once a service request has been received from the PAIM. (Refer to **http://www.sdi-12.org** for the SDI-12 standard).

3.2.2.   Convert all Three 0-5 V Inputs

- command: aM1!
- response: a0013<cr><lf>   data buf: +<ext. voltage>+<ext. voltage>+<ext. voltage>

This command instructs the PAIM unit to convert analog inputs #1, #2 and #5. These three analog inputs are 0-5V and are connected as shown in 5.3 Connectors.

Example:
The current host address is 3 and you wish to measure the 0-5V analog inputs.

Command: 3M1!
Response: 30013\0d\0a

This indicates that the three values will be ready for retrieval from the buffer in one second. The retrieval is accomplished with the D command.

3.2.3.   Convert both 0-2.5V Inputs

- command: aM2!
- response: a0012<cr><lf>   data buf: +<ext. voltage>+<ext. voltage>

This command instructs the PAIM unit to convert analog inputs #3 and #4, that is, the two 0-2.5 V inputs (see *5.3 Connectors)*. Refer to 5.3 Connectors for the pin assignments of these inputs to the 28 pin AMP CPC connector.

Example:
The current host address is 0 and you want to measure the internal temperature as well as the two external temperature inputs.

Command: 0M2!
Response: 00012\0d\0a

This indicates that the two values will be ready for retrieval from the buffer in one second. The retrieval is accomplished with the D command (see 3.2.5 Return Data Buffer)

3.2.4. Convert the Internal Temperature

- command: aM3!
- response: a0011<cr><lf>   data buf: +<int. temp.>

This command instructs the PAIM unit to convert analog input #8, that is, to measure the current internal temperature. Note that channel #8 makes use of the on-board temperature sensor but that this is in fact jumper-deselectable (jumper 'E5'). Therefore, although the PAIM is shipped with jumper 'E5' installed so that the user can monitor the internal temperature, channel#8 can in fact be used as an external 0-2.5V input. Refer to 5.3 Connectors for the pin assignments of this input to the 28 pin AMP CPC connector.

Example:
> The current host address is 0 and you want to measure the internal temperature.
>
> Command: 0M3!
> Response: 00011\0d\0a

This indicates that the value will be ready for retrieval from the buffer in one second. The retrieval is accomplished with the D command (see 3.2.5 Return Data Buffer)

3.2.5. Return Data Buffer

- command: aD! or aD0!
- Response: a<data buffer><cr><lf>

This command allows the user to read the most recent sensor measurement that was stored in the buffer. It is issued after an M command and as such is used to read the analog inputs.

Example 1:
> The address is 0 and the last "Measure" command was 0M0! (convert both 0-20 V inputs, channel #6 and #7). The response would typically be as follows:
>
> Command: 0D0!
> Response: 0+9.345+12.324\0d\0a

This indicates that the external 0-20V input (channel #6) is currently 9.345V and that the primary voltage source (channel #7) is at 12.324V.

Example 2:
> The address is 2 and the last "Measure" command was 2M3! (measure the internal temperature):
>
> Command: 2D0!
> Response: 2+18.3\0d\0a

This indicates that the current internal temperature is 18.3°C.

3.2.6. Get the Event Counters Values

- command: aM4! or aC4! (event counter 1); aM5! or aC5! (event counter 2)
- response: a0001<cr><lf> data buf <event count 1> or <event count 2>

This command is used to obtain the event counter values.

<u>Example:</u>

The current host address is 4 and you wish to obtain the reading of event counter 1.

Command: 443!
Response: 40001\0d\0a

This indicates that the event counter 1 value is stored in the buffer and can be immediately retrieved using a D command.
Note that the event counter input must not exceed 5 kHz (or 10 Hz in Switch Closure mode) in order to obtain reliable results (see *3.3.2 Set Mode Of Event Counters*).

3.2.7. Return Identification String

- command: aI!
- Response: a<identificationstring><cr><lf>

This command instructs the PAIM to return its identification string.   The SDI-12 format for the response to this command is : allccccccccmmmmmmvvvxxx…xx<cr><lf>.   'll' is the SDI-12 version number indicating SDI-12 version compatibility ; for example, version 1.2 is encoded 12. 'cccccccc' is the vendor identification (ex: AMASSData), 'mmmmmm' is the sensor model (ex: PAIM), 'vvv' is the sensor version number (ex: 100) and 'xxx…xx' is an optional field not currently used.

<u>Example:</u>

The current host address is 0 and you wish to obtain the identification string.

Command: '0I!'
Response: '012AMASSDATA PAIM100\0d\0a'

Therefore the sensor is manufactured by AMASS Data Technologies, it is a PAIM version 100 and is compatible with SDI-12 version 1.2.

3.2.8. Acknowledge Active Command

- command: a!
- Response: a<cr><lf>

This command allows the user to confirm that the current host address is active.

Example:

The current host address is 3 and you would like confirmation that it is active.

Command: '3!'
Response: '3\0d\0a'

This response is confirmation that host address 3 is active.

3.2.9. Address Query Command

- command: ?!
- Response: a<cr><lf>

This command allows the user to obtain the current node address. This command is simply the *acknowledge active* command but substituting the '?' as a wild card for the host address. The result of this "wild card" feature is that a sensor will respond regardless of its node address. The user must realize, however, that if more than one sensor is connected to the bus, they will all respond, causing a bus contention. An SDI-12 sensor must support the wild card character '?' with the *acknowledge active* command and can, but only optionally, similarly support all other commands to be SDI-12 compliant (Visit **http://www.sdi-12.org** for the SDI-12 standard).

Example:

There is only one sensor on the bus and you wish to obtain its node (device) address.

Command: ?!
Response: 6\0d\0a

The address of the sensor is therefore 6. To resume communications the host address can be updated to 6 or simply left at ?.

3.2.10. <u>Change SDI-12 Address</u>

- Command: aA<new address>!
- Response: <new address><cr><lf>

This command changes the node address of the PAIM. In order to communicate with the PAIM all commands issued by the host (logger) must be preceded with the proper address, therefore, the host must use the new address following the use of this command.

<u>Example:</u>

The current host (and sensor) address is 4 and you wish to change the sensor address to 6.

Command: '4A6!'
Response: '6\0d\0a'

This response confirms that the new node address is 6. The host (logger) address must now be updated to 6 in order to communicate with the PAIM. The user can then confirm that the new host address is active by sending the *acknowledge active* command if desired.

3.2.11. <u>Read Value of Set-up Parameter</u>

- command: aS2?!, aS3?! or aS4?!
- Response: a<data buffer><cr><lf>

This command enables the user to read the current value of any given set-up parameter. Its use is simple. Send the command normally used for setting the parameter in question but substitute the data value with the character '?'.

<u>Examples:</u>

The current sensor address is 3 and you wish to determine the current "scale" of event counter #1.

Command: '3S3?!'
Response: '3+0.25\0d\0a'

So the scale of event counter #1 is set at 0.25 units/count. Refer to *3.3.1 Set Scale Of Event Counters* .

3.3. **Set Commands**

3.3.1. Set Scale Of Event Counters

- command: aS3+<scale>! (counter 1) or aS4+<scale>! (counter 2)
- response: a<cr><lf>

This command is used to set the scale of the event counters. The scale is set according to the application in question. For instance, with tipping buckets the scale can be set to the amount of rain that each count represents. The engineering unit that is used for the scale will also represent the event counter value.

The following range of values may be used for the scale: 0.001 < range < 32.766. Negative values cannot be used for the scale.

Example:

The current host address is 7 and you wish to set the scale of event counter #1 for a tipping bucket which has 0.01"(0.25mm)/tip.

Command: 7S3+0.01!
Response: 7\0d\0a

This response confirms that the scale has been successfully set. The event counter readings will henceforth represent inches of rain.

3.3.2. **Set Mode Of Event Counters**

- command: aS2+<mode for event counters>!   (refer to APPENDIX B for mode codes)
- response: a<cr><lf>

This command is used to set the mode for the event counters. The mode-selection feature of the event counters allows the user to configure the system to a particular application. The modes that are available are as follows: Pulse versus Switch Closure modes and Read/Reset versus Overflow modes.   The four options are applied individually to each event counter and are defined by a mode code as in APPENDIX B. With reference to this table and the brief descriptions that follow the user will easily prepare the event counters for use.

The first two options are in regards to the type of switch to be used. The Switch Closure mode features a "debouncing" capability that is required in order to obtain an accurate count with switches that produce a "noisy" signal. Mechanical switches must be used in Switch Closure mode unless they are "debounced" externally. This mode is limited to a frequency of 10 Hz.

The Pulse mode is to be used with switches that generate electronic pulses, i.e. noiseless signals. This mode allows reliable readings up to a frequency of 5 kHz.

The last two options are presented as a means to control whether or not the event counters are reset upon retrieval. For instance, if the PAIM is to be operated by multiple users with a portable computer you may want the event counters to maintain a cumulative count which is not cleared when read; this would require setting them to Overflow mode. Alternatively, if the unit is accessed by a data logger which provides date/time stamping the Read/Reset mode is suitable.   While in the Read/Reset mode the event counter is automatically reset to zero when retrieved with an M, C (subsequently D) or R command. Therefore it is the user's responsibility to record the event counter readings upon retrieval since these are then automatically erased. If the event counter should ever reach the 16-bit maximum (65535) it overflows and starts over again.

While in the Overflow mode the event counter is not (and cannot) be reset. In order to obtain the proper value therefore, the user must subtract the previous reading from the current reading and take overflow into consideration. So if the last reading was 64300 and the current reading is 1200, the actual count would be

(65536 - 64300) + 1200 = 2436 counts.

The modes are represented internally as a 4-bit number as defined in the following table. The mode code is the decimal representation of this number and is converted in APPENDIX B as a convenience for the user.

Bit 3	Bit 2	Bit 1	Bit 0
EC 2 Read/Reset (1) or Overflow (0)	EC 2 Pulse (1) or Switch Closure (0)	EC 1 Read/Reset (1) or Overflow (0)	EC 1 Pulse (1) or Switch Closure (0)

Examples:

1. The current address is 3 and you wish to set the event counters as follows: event counter 1 to Pulse and Read/Reset modes and event counter 2 to Switch Closure and Overflow modes. The table shows that this corresponds to "0011" which translates to a mode code of 3. Therefore the following command must be sent:

Command: 3S2+3!
Response: 3\0d\0a

This response confirms that the event counters have been successfully set.

2. Event counter 1 to Switch Closure and Read/Reset modes and event counter 2 to Pulse and Overflow modes with a current address of 4. From the table above the binary mode code is found to be 0110, or 6 decimal .

Command: 4S2+6!
Response: 4\0d\0a

3.3.3.   <u>Reset Event Counters</u>

- command: aS5!
- response: a\<cr>\<lf>

This command is used for resetting both event counters to a count of zero.

<u>Example:</u>
> The current sensor address is 7 and you wish to reset both event counters to zero.
>
> Command: 7S5!
> Response: 7\0d\0a
>
> Both event counters now have a value of zero counts.

3.3.4. <u>Configure the Analog Channels</u>

- command: aSA+<channel>+<state>+/-<scale>+/-<offset>!
- response: a<cr><lf>

This command is used to configure the analog channels to your application. As mentioned in the introduction to this manual, the PAIM is equipped with 8 channels, two of which, channel #7 and #8, are dedicated to maintenance parameters (the primary voltage and the internal temperature). The remaining six channels are configurable to your needs.   Note that all set-up of the PAIM is stored in non-volatile memory and is thus not lost during power-down.

The remaining channels, that is, #1 to #6, are for external inputs and are configurable to the operator's needs.   The analog sensors are connected (refer to *5.3 Connectors*) to the appropriate channels based on the range of output of those sensors and whether they require single-ended or differential measurement. Here is the range of the channels:

- Channels #1, #2 and #5 is 0-5V
- Channels #3 and #4 are 0-2.5V
- Channel #6 is 0-20V

For a differential measurement of -5 to 5 V use channels #1 and #2 with the '+V' connection to #1 and the '-V' connection to #2.   Once configured as differential as described below, channel #1 will henceforth refer to the difference between the potential at #1 and that at #2, that is,

$$V_{diff} \ = \ V_1 \ - \ V_2$$

Similarly for a differential measurement of –2.5 to 2.5 V, use channels #3 and #4 with the '+V' connection to #3 and the '-V' connection to #4.   Once configured as differential as described below, channel #3 will henceforth refer to the difference between the potential at #3 and that at #4.

Each channel that is connected to an analog sensor must be configured for that sensor with the 'aSA' command while specifying the proper *State* and a "scale" and "offset" if required. Select the proper *State* from the table below:

STATE	TYPE OF CONVERSION	APPLICABLE CHANNEL(S)
1	0-5V conversion	#1, #2 or #5
2	0-2.5V conversion	#3 or #4
3	Custom Equation	#1, #2, #3, #4 or #5
4	4-20mA (Remove 4mA bias)+Custom Equation	#1, #2, #3 or #4
5	AD-34* temperature conv. (0-2.5V)	#3 or #4
6	Differential (1-2) with +/-5V	#1 AND #2
7	Differential (1-2) with Custom Equation	#1 AND #2
8	Differential (3-4) with +/-2.5V	#3 AND #4
9	Differential (3-4) with Custom Equation	#3 AND #4
10	0-20V conversion	#6 or #7

* The AMASS Data Technologies AD-34 Temperature Sensor.

It is from the above *State Table* that the analog channels are configured for use.   The operator must select a the proper type of conversion based on the application.   This is determined based on the following criteria:

- Single-Ended or Differential input?
- Range of output of the sensor (refer to *5.3 Connectors* to connect to the proper channel)?
- Is the data going to be stored in Volts or converted to some other engineering unit? If converted, to how many decimal places?
- Is this a 4-20 mA application?
- Are you connecting the AD34 temperature sensor?

These various options are best explained with a few examples. Further explanations are added where warranted.

EXAMPLE 1:
The sensor address is 3 and you have connected a 0 to 2.5V analog sensor to channel #4.   You wish the analog data to be converted to Volts.

        Command: 3SA+4+2!
        Response: 3\0d\0a

The set-up is complete! The PAIM will convert the analog data of channel #4 to Volts based on a range of 0 to 2.5V.   To then read the value at channel #4 use 'aM2!' as described in *3.2.3 Convert both 0-2.5V Inputs*. Note that the "*Scale*" and "*Offset*" parameters are omitted unless specifying a *Custom Equation*.

EXAMPLE 2:
The sensor address is 2 and you have connected a –5V to +5V sensor to channels #1 and #2 with '+V' to #1 and '-V' to #2. You wish the analog data to be converted to Volts.

        Command: 2SA+1+6!
        Response: 2\0d\0a

The set-up is complete! The converted data for <u>channel #1</u> as returned by the command 'aM1!' will henceforth be the difference between channels #1 and #2, expressed in Volts. Note that the value returned for channel #2 will represent the potential at channel with respect to ground and NOT the difference between #1 and #2.   Note that the parameters "*Scale*" and "*Offset*" are omitted here because a *Custom Equation* was not specified.

EXAMPLE 3:
The sensor address is 6 and you have connected a 0 to 5V sensor to channel #2 and it is being used to monitor a tank of pressurized gas.   You wish to convert the readings to pressure in "PSI".   The measurement range of the sensor is 0 to 3000 psi within the operational output range of 0.5 to 4.5V. Assuming that you wish to display the results to one decimal place, proceed as follows:

        Command: 6SA+2+3+0.05722-375.0!
        Response: 6\0d\0a

The set-up is complete! The PAIM will convert the analog data of channel #2 to 'PSI' based on a full-scale value of 3000psi when the sensor outputs 4.5 V and 0psi for 0.5V.

Note that the number of decimal places of the converted data is specified by the value entered for the offset. <u>The number of decimal places of the converted data will be equal to that of the value specified for the "offset".</u> Therefore, given that the offset was expressed to one decimal place in the example above, the converted data (pressure) will be also be to one decimal place, for example, 1507.4 psi.

The values for "**Scale**" and "**Offset**" are simply determined by the properties of the linear sensor in question:

$$\text{Scale} = \text{Slope} = \frac{(P_{FS} - P_0)}{\{(4.5/5)\text{max. count} - (0.5/5)\text{ max. count}\}}$$

$$= \frac{(3000 - 0)}{\{(4.5/5)\ 65535 - (0.5/5)\ 65535\}}$$

$$\text{Scale} = 0.05722$$

Note that the maximum count is equal to $(2^{16}-1)=65535$ because the conversion is done with 16 bits.

EXAMPLE 4:
The sensor address is 0 and you have connected a 4-20 mA sensor to channel#3.

       Command: 0SA+3+4!
       Response: 0\0d\0a

The set-up is complete! The 4mA bias has been removed. The analog data of channel #3 will henceforth be converted such that the output range of the sensor is from 0.5 (zero counts) to 2.5V (full scale). By selecting *State*'4', the user has the option of defining a custom equation by specifying a *Scale* and an *Offset*.

NOTE: When connecting a 4-20 mA sensor to the PDAS-II be sure to use the proper precision resistor in series with the sensor in the current loop as follows:

Case 1: Channel #1 and #2 (0-5V) → use 250 Ω

Case 2: Channel #3 and #4 (0-2.5V) → use 125 Ω
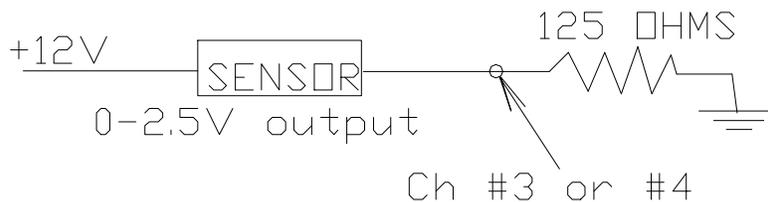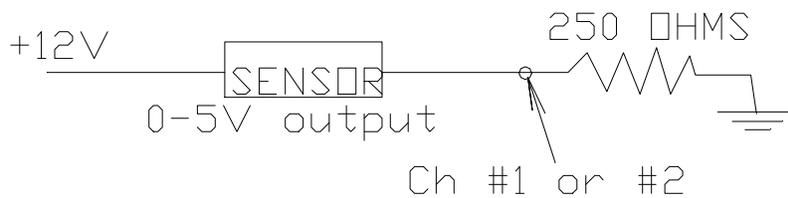
4. Host Software Operation

The demo software that is supplied with the PAIM is menu-driven host software (PAIM.EXE) that may be used as an interface for sending and receiving the proper commands that are described above. To initiate the program type "PAIM" from the command prompt (the program is currently available for MS-DOS and QNX4). Possible command line options include: "PAIM -*s com#*" where # represents the port number which is being used. When the default port is being used the latter is unnecessary.   Another command line option is to use "PAIM -a #" to initiate the program with host address # (host addresses are actually numbered from 0 to 9).

Upon start-up the program displays the main menu with the default host address of 0 displayed at the *Command line prompt*. The user may then confirm that this host address is active, that is, equal to the node address of the device, by selecting item 'a' to send the *acknowledge active* command. The display would then be as follows:

[Addr=0] Command?   (Command line prompt)

+ - Change Host address   (Main menu)

s - Setup menu   (Access to *Setup* sub-menu)

r - Read Analog Inputs and Event Counters   (Access to *Read* sub-menu)

I - Read sensor Identification (aI!)

a - Request Acknowledgment (a!)

n - Change Node (device) address (aAb!)

m - Send Manual command

c - Toggle Continuous mode

q - Quit

The current node address: 0   (Infoline message)

Sending: '0!'   (Communications section)
Received: '0\0d\0a'

The Infoline indicates that the host address is indeed active. If it had not been, the user could have changed the host address to ? by using the '+' item, and then selecting item 'a' to effectively send an *address query* command and in so doing determining the active host address.

The *Change node address* item sets the PAIM node address to something other than the default address of 0 (see 3.2.10 Change SDI-12 Address). To reestablish communication after such a change the host address must be set to the new node address by using '+'. A successful link occurs if the Infoline message does not display "No response from sensor" (see *Trouble Shooting*). One can always confirm that the current host address is active by using the *acknowledge* or *identify* items.

Item '**i**' is to send the *identification* command respectively (see 0).

Initializing the parameters of the PAIM unit is accomplished in the ***Setup* sub-menu**. For instance, to set the scale of the event counters press '3' from the sub-menu. The command line then prompts the user to enter the new scale. Upon entering a value the *communications section* displays the command being sent and received by the host PC as a result of your menu selection. The *InfoLine* then displays something like "**Scale Set**" or an error message such as "**No response from sensor**" (see *Trouble Shooting*). The screen would appear as below if the scale of event counter #1 had been set to 0.25. To return to the main menu press the *ESC* key.

[Addr=0] Command?   (Command line prompt)

2 - Set mode of event counters: Pulse, Switch, Read/Reset and Overflow (aS2+x!)   (Setup sub-menu)

3 - Set event counter 1 scale(aS3+x!)

4 - Set event counter 2 scale(aS4+x!)

5 - Reset event counters (aS5!)

A – Configure the Analog Subsystem (aSA+ch+state+scale+offset)

Scale set   (Infoline message)

Sending: '0S3+0.25!'   (Communications section)

Received: '0\0d\0a'

Item '**2**' of this menu is used to set the mode of operation of the event counters which provide the user with some control as to how the counters are reset as well as being able to configure them for use with either mechanical switches or electronic pulse-generating-type switches (see *3.3.2 Set Mode Of Event Counters*).

Items '**3**' and '**4**' are used to set the scale of the event counters as described in section *3.3.1*.

Item '**A**' is used to configure the A/D as described in section *3.3.4 Configure the Analog Channels*.

Obtaining analog input values and event counter readings is accomplished from the ***Read* sub-menu**. For instance, to get the Primary Voltage and 0-20V External Voltage select '**0**' (see 3.2.1 Convert both 0-20 V Inputs). The screen would be something as follows:

[Addr=0] Command?   (Command line prompt)

0 - Get 0-20V Inputs (aM0!,aD!)   (Read sub-menu)

1 - Get 0-5V Inputs (aM1!,aD!)

2 – Get 0-2.5V Inputs (aM2!,aD!)

3 - Get Temperature Inputs (aM3!,aD!)

4 - Get event counter 1 (aM4!,aD!)

5 - Get event counter 2 (aM5!,aD!)

m - Send Manual command

c - Toggle Continuous mode

External 0-20V= 9.234   **Primary Voltage= 12.123**

Sending: '0M0!'

Received: '00012\0d\0a'

Sending: '0D!'

Received: '0+9.234+12.123\0a\0a'

Note that two commands were required to obtain the readings (0M0! and 0D!) as described in *SDI-12 AMASSER SE Sensor Commands.*

Item '**m**' allows the user to type any command which appears in the *SDI-12 AMASSER SE Sensor Commands* section. This is done at the *command line* at the following prompt: "Manual cmd:". When entering manual commands however, the host address need not be put at the beginning of the string since the current host

address is added by the software before it is sent to the PAIM. For instance, the command aR0! should be entered as R0!.

Press 'c' to enter the *continuous mode*.   In this mode any subsequent readings will be continuously updated at the Infoline. For instance, if '**0**' had been selected in *continuous mode* in the example above, the external 0-20V and Primary Voltage would be automatically updated at the InfoLine while your host is continually polling the PAIM. Any menu item may be selected while in continuous mode. Manual commands may also be viewed in continuous mode by pressing 'm' so that the last manual command that was entered is continuously updated. When in the *continuous mode* the command line reads: [Addr=0] Continuous command ?.

The command line editor that appears when entering manual or set commands uses a buffered input which allows the user to edit what is typed before it is entered. It also allows one to recall previous entries by using the up and down arrow keys at the command prompt.

4.1.   Trouble Shooting

Problem: The InfoLine displays "No response from sensor"
Solutions:

- You must use the "CONV-SDI" RS232 to SDI-12 interface available from AMASS Data Technologies Inc. Without this product your PC cannot communicate with the PAIM. Contact ADT for the current price of the CONV-SDI.
- The current host address is not active, that is, it is not equal to the node (device) address. If these addresses are not equal there cannot be communication between host and PAIM. The default address for the PAIM is 0, however, if this is changed the host must be updated to resume communication. To determine the current node address the user can send the *Address Query* command by selecting host address '?' (located after '9') and by selecting the *acknowledgment* item from the main menu.
- The improper port was specified upon startup of the host software. Recall that the port designation is specified as a command line option. For instance, if the connection is made on "com2" the startup would be as follows: "PAIM -s com2".
- Of course, a faulty connection is possible. Check the integrity of all connections.
- No power to PAIM.
- Reset the PAIM and reattempt a command. To reset simply remove the connector from the PAIM and reconnect.

5.   Installation

5.1.   Mechanical

The PAIM may be attached to a flat surface using #10 bolts or screws through the four holes located in the mounting bracket on the PAIM housing.   For information on the physical characteristics of the PAIM refer to *Specifications.*

5.2.   Electrical

Connect the analog inputs and outputs to the 28 pin AMP CPC connector as shown in the next section as well as the SDI-12 cable to the DB9 connector.

The PAIM is internally protected against transient voltages and lightning surges on the signal input lines. However,

 IT IS ESSENTIAL TO PROVIDE PRIMARY EXTERNAL LIGHTNING PROTECTION.

The event counter input must not exceed 10 Hz when in Switch Closure mode or 5 kHz when in Pulse mode (see *3.3.2 Set Mode Of Event Counters*).

5.3. Connectors

1   DB9 for SDI-12 communications and +12VDC
1   28 pin AMP CPC connector for analog inputs and outputs.

Connections for the DB9 connector are as follows:

PIN	Signal
1	SDI-12 data
2	RXD
3	TXD
4	Secondary SDI-12 data
5	GND
6	+12 VDC power
7	RTS
8	CTS
9	none

Connections for the 28 pin AMP CPC are as follows:

PIN	SIGNAL
1	Analog channel 3 (**0-2.5V Ext.**)
2	Analog GND
3	Analog channel 6 (**0-20V Ext.**)
4	Analog channel 2 (**0-5V Ext.**)
5	Analog GND
6	Analog GND
7	Analog GND
8	Analog channel 7 (**Primary Voltage: 0-20V**)
9	Analog channel 1 (**0-5V Ext.**)
10	Analog GND
11	Analog channel 4 (**0-2.5V Ext.**)
12	Analog channel 5 (**0-5V Ext.**)
13	Analog GND
14	Analog channel 8 (**Internal Temp: 0-2.5V**)
15	Analog GND
16	5VDC ref.
17	5VDC ref.
18	5VDC ref.
19	5VDC ref.
20	Analog GND
21	2.5 VDC ref.
22	2.5 VDC ref.
23	Event Counter #2 GND
24	250 μA constant source
25	250 μA constant source
26	Event Counter #1 GND
27	Event Counter #2
28	Event Counter #1

6.   Specifications

Processor - Atmel 89S8252
Word Size : 8 bit data - 8 bit instruction
Memory : 8 Kbytes ROM, 256 bytes RAM
EEPROM 256 bytes

PAIM Input

Primary Voltage: 0 - 20V (ch. #7)
Internal Temperature: 0 - 2.5V (ch. #8)
One 0-20V Ext. Input (ch. #6)
Three 0-5V Ext. Input (ch. #1, #2 and #5)
Two 0-2.5V Ext. Input (ch. #3, #4)

Physical Characteristics

Height - 152.0 mm. (6.0 in.)
Width - 196.0 mm. (7.75 in.)
Depth - 41.3 mm. (1.625 in.)
Weight : 1 Kg (2.2 lb.)
Mounting : One Mounting bracket, Standard
                        Use four #10 bolts or screws.

A/D Converter

Converter: Crystal CS5524/28
Type: Delta-Sigma A/D Converter
Noise-free resolution: 18 bits
Buffered Bipolar/Unipolar Ranges of 25mV, 55mV,
100mV, 1V, 2.5V and 5V

PAIM Output

2.5V and 5V reference
250 microAmp constant current source

Connectors

- 9 pin DB9S for SDI-12, +12 VDC
- 28 pin AMP CPC for analog inputs and outputs

Environmental Characteristics

Operating : -40 to +55 C
Storage : -60 to +100C
Humidity : <= 100% non-condensing

APPENDIX A: PAIM SDI-12 Command Set: Quick reference

Retrieve Data Commands

1. Convert and return both 0-20 V inputs (channel #6 and #7)
   - command: aM0!
   - response: a0012<cr><lf> data buf: +<primary voltage>+<external voltage>

2. Convert and return all three 0-5 V inputs (channel #1, #2 and #5)
   - command: aM1!
   - response: a0013<cr><lf> data buf: +<ext. volt.#1>+<ext. volt.#2>+<ext. volt.#5>

3. Convert and return both 0-2.5V inputs (channel #3, #4)
   - command: aM2!
   - response: a0013<cr><lf> data buf: +<ext. volt #3>+< ext. volt #4>

4. Convert and return the Internal Temperature
   - command: aM3!
   - response: a0011<cr><lf>   data buf: +<Internal temperature>

5. Get event counter values
   - command: aM4! or aC4! (event counter 1); aM5! or aC5! (event counter 2)
   - response: a0001<cr><lf> data buf <event count 1> or <event count 2>

6. Return data buffer
   - command: aD! or aD0!   Response: a<data buffer><cr><lf>

7. Return identification string
   - command: aI!   Response: a<identificationstring><cr><lf>

8. Acknowledge active command
   - command: a!   Response: a<cr><lf>

9. Address query command
   - command: ?!   Response: a<cr><lf>

10. Change SDI device (node) address
    - command: aA<new address>!   Response: <new address><cr><lf>

11. Read Value of Set-up Parameter
    - command: aS2?!, aS3?! or aS4?!   Response: a<data buffer><cr><lf>

Set Commands

1. Set mode of event counters
   - command: aS2+<mode for event counters>!   (refer to APPENDIX B for mode codes)
   - response: a<cr><lf>

2. Set scale of event counters
   - command: aS3+<scale>! (counter 1)   or   aS4+<scale>! (counter 2)
   - response: a<cr><lf>

3. Reset event counters
   - command: aS5!   Response: a<cr><lf>

4. Configure the A/D channels
   - command: aSA+<ch>+<state>+<scale>+<offset>!   Response: a<cr><lf>

All other commands are reserved for use by the factory for R/W of calibration data into/from the EEPROM.

APPENDIX B: Mode Codes for the Event Counters

Mode code	EC2 R/R or Overflow	EC 2 Pulse or Switch	EC1 R/R or Overflow	EC 1:Pulse or Switch
0	Overflow (0)	Switch Closure(0)	Overflow (0)	Switch Closure(0)
1	Overflow (0)	Switch Closure(0)	Overflow (0)	Pulse (1)
2	Overflow (0)	Switch Closure(0)	Read/Reset (1)	Switch Closure(0)
3	Overflow (0)	Switch Closure(0)	Read/Reset (1)	Pulse (1)
4	Overflow (0)	Pulse (1)	Overflow (0)	Switch Closure(0)
5	Overflow (0)	Pulse (1)	Overflow (0)	Pulse (1)
6	Overflow (0)	Pulse (1)	Read/Reset (1)	Switch Closure(0)
7	Overflow (0)	Pulse (1)	Read/Reset (1)	Pulse (1)
8	Read/Reset (1)	Switch Closure(0)	Overflow (0)	Switch Closure(0)
9	Read/Reset (1)	Switch Closure(0)	Overflow (0)	Pulse (1)
10	Read/Reset (1)	Switch Closure(0)	Read/Reset (1)	Switch Closure(0)
11	Read/Reset (1)	Switch Closure(0)	Read/Reset (1)	Pulse (1)
12	Read/Reset (1)	Pulse (1)	Overflow (0)	Switch Closure(0)
13	Read/Reset (1)	Pulse (1)	Overflow (0)	Pulse (1)
14	Read/Reset (1)	Pulse (1)	Read/Reset (1)	Switch Closure(0)
15	Read/Reset (1)	Pulse (1)	Read/Reset (1)	Pulse (1)