<u>///AMASSER ECAIM</u>

Embedded Control Analog Input Module
User's Reference Manual

Updated: February 15,1997
http://www.amassdata.com

Table of Contents

Pliant: readily yielding to influence

1.0 AMASSER ECAIM Analog Input Module

The 2500 Vrms isolated ECAIM 20 bit Analog Input Module is a compact DIN Rail mountable, intelligent full function analog input subsystem that is compatible with the AMASS RS485 Multi-drop Embedded Control Network (ECN). This module provides firmware based intelligent analog input functions for its host Embedded PC from standard serial COM Ports using an RS485 based multi-drop environment supporting up to 32 modules over a distance of 4000 feet on single or dual twisted pair cable..

1.1 AMASSER ECAIM Overview

The ///AMASSER ECAIM Analog Input Module provides a standalone, full function analog input subsystem which is fully compatible with the ECN network. The ECAIM module includes a delta-sigma charge-balance 16 or 20 bit converter, a precision bandgap 2.5V voltage reference, an independent calibration microcontroller with SRAM for the A/D converter, a digital filter and an RS-485 serial interface.

The ECAIM can measure either unipolar or bipolar signals and the self-calibration feature insures excellent offset and gain accuracy. The A/D is optimized to operate from a 32.768 kHz crystal and can provide up to 20 conversions per second. The digital filter when operated with a 32.768 kHz clock has zeros precisely at 50 and 60 Hz line frequencies and multiples thereof.

The ECAIM module has a 2500Vrms isolated onboard DC-DC converter which provides a dual polarity power supply (+5 and -5), for the ADC. The communications port is 2500Vrms optically isolated RS485 multi-drop interface and can be configured for half of Full duplex operation. Network termination resistor sockets are provided on the module.

1.2 Converter Performance

The ECAIM A/D converters have excellent linearity performance. Calibration minimizes the errors in offset and gain. The ECAIM/16 has no missing code performance to 16-bits. The ECAIM/20 has no missing codes guaranteed to 18-bits. Some codes may be missing in the ECAIM/20 at the 20-bit level because of truncation in the digital computation of the output digital word. Only a few codes will be missing at the 19-bit level. The converters achieve Common Mode Rejection (CMR) at dc of 105 dB typical, and CMR at 50 and 60 Hz of 120 dB typical.

The ECAIM uses chopper-stabilised techniques to minimize drift. Measurement errors due to offset or gain drift can be eliminated at any time by recalibrating the converter.

1.3 Digital Filter Characteristics

The hardware digital filter on the ECAIM module is the combination of a comb filter and a low pass filter. The comb filter has zeros in its transfer function which are optimally placed to reject line interference frequencies (50 and 60 Hz. At exactly 50, 60, 100, and 120 Hz the filter provides over 120 dB of rejection. The converter yields excellent attenuation of these interference frequencies even if the fundamental line frequency should vary

1% from its specified frequency. The -3 dB corner frequency of the filter when operating from a 32.768 kHz clock is 17 Hz.

1.4 <u>Anti-Alias Considerations for Spectral Measurement Applications</u>

Input frequencies greater than one half the output word rate (10 conversions per second)may be aliased by the converter. To prevent this, the input signals are limited in frequency to no greater than one half the output word rate of the converter by a passive single pole RC filter. Frequencies close to the modulator sample rate (XIN/2) and multiples thereof may also be aliased.

1.5 <u>Firmware Support</u>

The ECAIM module has been designed to provide the user with intelligent Analog Input functions that are easily executed over a robust & reliable network from a COMM port on your industrial PC. The firmware provides input, network and specialized functions which are detailed on the following page.

Input functions include read data in hexadecimal format or in decimal signed integer format. Set functions include initiate a calibration. The basic clock tick by which the firmware operates is 50 milliseconds. This provides for a maximum conversion rate of 20 times per second.

Optional firmware can be provided that provides a watchdog function for the communications port or a 2 sec time constant digital filter which is implemented in firmware.

2.0 <u>ECN Protocol</u>

2.1 <u>Physical</u>

The Embedded Control Network (ECN) is configured as an RS-485 based multi-drop environment supporting up to 32 modules over a distance of 4000 feet on single or dual twisted pair cable. The network operates at 9.6 or 19.2 Kbaud rates with the ability to increase these rates up to 115.2 Kbaud. An additional pair is also specified for the routing of 24Volt or 5Volt DC power. An important feature of the AMASSER ECxxx line of products is that each module provides 2500 Vrms isolation between the network and the I/O functions being provided.

2.2 <u>Protocol</u>

The Embedded Control Network protocol is configured as a master slave environment supporting a simple & reliable multi-node ASCII based command - response message system. Message strings begin with an address byte followed by the message body, followed by a 8 bit hexadecimal checksum and terminated with a carriage return byte. The protocol allows broadcast messages, module identification, and individual module firmware reset ability.

2.3 <u>Command Messages</u>

In order to communicate with the AMASSER ECAIM Unit, the host controller sends a message to the module. The first byte of the message string is an "address" byte, which consists of an ASCII character which represents the AMASSER ECAIM's node address.   In order to support up to 32 remote node addresses on a network, these addresses range from ASCII "0" (30H) to ASCII "O" (4FH). Following the Node address byte is a one or two byte command which allow various I/O operations to be initiated. The commands available to the host machine for the ECAIM module are listed below. Following the command byte is the body of the message which

contains the parameters required for the execution of the command. Following the body of the message are two checksum bytes which are the ASCII hex value of the complement of the 8 bit checksum of all the bytes in the message up to and not including the two checksum bytes. The last character of the message is always "CR" (0DH). This "CR" character must never be used any where else in the message string. The host controller must turn off its transmitter and enable its receiver within 1 millisec after sending the "CR" in half duplex operation mode.

2.4 <u>Response Messages</u>

   The AMASSER ECAIM responds to the host computer by sending a response message to the host computer. The first byte of the response message string is an "address" byte, which consists of an ASCII character which represents the AMASSER ECAIM's node address. The response address ranges from ASCII "0" (30H) to ASCII "O" (4FH) and is set by jumpers located on the ECAIM module. Following the address byte is the body of the response message which contains the parameters (if any) requested by the host computer. Following the body of the message are two checksum bytes which are the ASCII hex value of the complement of the 8 bit checksum of all the bytes in the message up to and not including the two checksum bytes. The last character of the response message is always "CR" (0DH). The AMASSER ECAIM turns off its transmitter within 1 millisec after sending the "CR" in half duplex operation mode.

2.5 <u>Message Length</u>

 A string of bytes or characters will be at least four bytes in length and not more than 36 bytes in length.

2.6 <u>Byte Frame Format</u>

 All transmission will be asynchronous having 1 start bit, 8 data bits, no parity and 1 stop bit.

2.7 <u>Baud Rate</u>

 19,200 - installed or 9600 - not Jumper Block Position 3-4 (see section *Jumper Block Configuration*)

3.0 <u>AMASSER ECAIM Command Set</u>

3.1 <u>ECN Command Symbol Definitions:</u>

a : address byte - Jumper Block Positions 3-4, to 11-12
        no jumpers installed equals address 0 where 30H < a < 4FH
        for the host commands and for AMASSER ECAIM Responses.

xx.xxx : 20 or 16 bit decimal ASCII representation of voltage in volts i.e. 99.453 or 0.001 volts dc.

b : Analog Input Channel Identifier i.e Input 2 b = 2

+ : Delimiter character - used to separate analog channel identifier and data values i.e. +2+ 256.45+.01+

<xxxx>h : 4 ASCII Hex bytes representing 16 bit data where '0'< x > '9' or 'a' < x > 'f' 5 bytes for 20 bit data.

Cksum <x><x> : the ASCII hex value of the complement the 8 bit checksum of all the bytes in the message not including the two checksum bytes and the carriage return byte. Ex: Message '0!ae' where 'ae' is the hex value of the complement of the checksum of the ASCII characters '0' and '!'. ASCII '0' is 30H and ASCII '!' is 21H which when added and "complemented" results in 'ae' which is then appended to the command.

Command bytes : one or two byte ASCII character command

Identification String : "10AMASSDataECAIM xxx" where xxx = Revision Level

3.2 ECN Set Commands: Set ECAIM Parameters

3.2.1 Calibrate A-D Converter S0
- command: aS0<cksum><cr> (calibrate ADC)
- response: a<cksum><cr>

<u>Example:</u>
The current host address is 5 and you wish to calibrate the ECAIM module.

Command: 5S047
Response: 5ca

3.3 ECN Measure Commands: Read Analog Input Data

3.3.1 Retrieve Hex Format ADC Data M0
- command: aM0<cksum><cr>
- response: a<Hex ADC Value><cksum><cr> where value is 4 or 5 ASCII hexadecimal characters
.

This command is used to obtain the analog input conversion in hexidecimal format.

<u>Example:</u>
The current host address is 0 and you wish to read the digital conversion of the analog input in hexidecimal format.

Command: 0M052
Response: 01e8c9e

Therefore the hexidecimal data is 1e8c which represents a voltage of 1.193 (unipolar mode). Recall that the conversion is done using 16 bit representation, i.e., a total of 65535 counts for a full scale of 10 V in unipolar mode. Given that 1e8c is equivalent to 7820 decimal, the voltage is converted as:

$$\frac{7820 \times 10}{65535} = 1.193 \text{ V}$$

3.3.2 Retrieve Decimal ADC Data M1
- command: aM1<cksum><cr>
- response: a<xx.xxx><cksum><cr> Leading zeros are suppressed.

This command is used to obtain the analog input conversion in decimal format.

<u>Example:</u>
The current host address is 3 and you wish to read the digital conversion of the analog input in decimal format.

Command: 3M14e
Response: 31.202d9

The input voltage was converted to 1.202 V.

3.4 <u>ECN Network Commands:</u>

3.4.1 Change Node Address Command A
- command: aAb<cksum><cr> where b = new node address
- response: b<cksum><cr>

This command is used to change the node address of the ECAIM. Successful communication between the host and the ECAIM can only take place if the host and ECAIM node addresses are the same. Therefore the host address must be updated following the use of this command (see *ECAIM Operation*).
Upon start-up the default node address is set by the jumper block configuration (see *Jumper Block Configuration*) so that the user must adjust the host address accordingly.

<u>Example:</u>
The active host address is 2 and you wish to change it to B.

Command: 2AB4a
Response: Bbd

The host address must now be changed to B in order to communicate with the ECAIM.

3.4.2 Send ECN Identification String   I
- command: aI<cksum><cr>
- response: a<identification string><cksum><cr>

This command is used to obtain identification of the ECAIM module on the bus. Recall that the Embedded Control Network (ECN) is configured to support up to 32 modules over a distance of 4000 feet on single or dual twisted pair cable. Any module on the bus may be identified by its manufacturer, model and serial number.

Example:
The current (and active) host address is 2 and you wish to identify the module at this address.

Command: 2I84
Response: 210AMASSDataECAIM11285

Therefore, the module at address 2 is an ECAIM module manufactured by AMASS Data Technologies and uses firmware version 112 and ECN version 1.0.

3.4.3 Send acknowledgment   !
- command: a!<cksum><cr>
- response: a<cksum><cr>

This command allows the user to confirm the communications link with the ECAIM without having to send a Read or Set command.

Example:
The active host address is 2 and you wish to confirm your link with the ECAIM.

Command: 2!ac
Response: 2cd

3.4.4 Send Address Query *!

- command: *!<cksum><cr>
- response: a<cksum><cr>

This command allows the user to obtain the current node address. This command is simply the *acknowledgment* command but substituting the '*' as a wild card for the host address. The result of this "wild card" feature is that a module will respond regardless of its node address. The user must realize, however, that if more than one module is connected to the bus, they will all respond, causing a bus contention.

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

Command: *!b4
Response: 6c9

Therefore the module has node address 6.

3.4.5 Reset The Module #

- command: a#<cksum><cr>
- response: none Wait 10 milliseconds, then send Acknowledgement

This command allows remote resetting of the ECAIM to reestablish communications with the module.

Example:
The current host address is 3 and you wish to reset the module.

Command: 3#a9
Response: none

To confirm that the ECAIM is reset send the acknowledgment command with the default node address as configured on the jumper block (see *Jumper Block Configuration*).

4.0 Host Software Operation

The ECAIM user-interface is menu-driven host software that enables the user to send and receive the proper commands that are described above. To initiate the program type "ECAIM". Possible command line options include: "ECAIM -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 "ECAIM -a #" to initiate the program with host address # (host addresses are actually numbered from 0 to 9 and A to V).

Upon start-up the program displays the main menu with the default host address 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 some device on the bus, by selecting item 'a' to send the *acknowledge* command. The display would then be as follows:

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

+ - Increment host address   (Main menu)
- - Decrement host address
n - Change node address
a - Request acknowledgment
i - Read sensor identification
r - READ converter data   (Access to READ sub-menu)
b - Calibrate ECAIM
m - Send manual command
c - Toggle continuous mode
q - quit

Response: 0   (Infoline message)

Sending: '0!ae'   (Communications section)
Received: '0bf'

The Infoline indicates that the host address is indeed active.

The *Change node address* item sets the ECAIM node address to something other than the default address set by the jumper block (see *Jumper Block Configuration*) . To reestablish communication after such a change the host address (the current value of which is always shown in the command line at the top of the screen. Ex: 0) must be set to the new ECAIM node address by using '+' and '-' . A successful link occurs when the Infoline message displays the host address (Ex: 0), otherwise it displays "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 'm' allows the user to type any command which appears in the *AMASSER ECAIM Command Set* section. This is done at the *command line* at the following prompt: "Manual cmd:". When entering manual commands however, the host address and checksum need not be entered since these are added by the software before being sent to the ECAIM. For instance, the command aM1<checksum> should be entered as M1. The command line editor uses a buffered input which allows the user to edit what is typed at the prompt before it is entered. It also allows you to recall previous entries by using the up and down arrow keys.

Select item 'b' to calibrate the ECAIM.

The sub-menu allow the user to retrieve the converter data by sending the Read commands described above. To return to the main menu press the *ESC* key.

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

h - Retrieve Hex format ADC data   (READ sub-menu)
d - Retrieve Decimal format ADC data
m - Send manual command
c - Toggle continuous mode

Response: 1.194   (Infoline message)

Sending: '0M151'   (Communications section)
Received: '01.194d2'

        This is a typical response when selecting item 'd' to read the converter data in decimal format. In this instance the analog input was converted to 1.194 V.

        Press 'c' to enter the *continuous mode*. In this mode any subsequent readings will be continuously updated at the Infoline.   For instance, if one selects item 'd' the instantaneous analog input would be displayed at the InfoLine. Any *Read* menu item may be selected while in continuous mode. Manual commands may also be viewed in continuous mode. To do this press 'm' so that the last manual command that was entered is continuously sent. Note that this is usually done for "read" commands only since one would not normally "write" continuously to a module. When in the *continuous mode* the command line reads: [Addr=0] Continuous command ?.

4.1 Trouble Shooting

        Problem: The InfoLine displays "No response from sensor"
        Solutions:
- 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 ECAIM. The default address for both host and ECAIM is 0, however, if one is changed the other must be updated to resume functions.   To determine the active host address the user can send the *address query* command by selecting host address '*' (located after 'V') followed by item 'a' for an *acknowledgment* command.
- 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: "ECAIM -s com2".
- Of course, a faulty connection is possible. Check the integrity of all connections.
- No power to the ECAIM.
- Reset the ECAIM and reattempt a command. To reset simply install and remove a jumper at pins 19-20.

5.0 <u>ECAIM Hardware Configuration:</u>

      The AMASSER ECAIM module is available in 16 or 20 bit ADC resolutions. The baud rate jumper 3-4 selects between 9600 or 19200 Kbaud. Jumper 1-2 selects the mode of operation for the ADC either unipolar 0 - 10Vdc or bipolar -10 to +10Vdc. The watchdog timer jumper 17-18 selects a processor non-activity hardware reset interval of 1.2 seconds or 150 msec.

5.1 <u>Jumper Block Configuration</u>

	Uninstalled	Installed
1-2 ADC Bipolar/Unipolar	Bipolar	Unipolar
3-4 Baud Rate	9600	19200
5-6 Not Defined		
7-8 Bit 0 Node Address	0	1
9-10 Bit 1 Node Address	0	1
11-12 Bit 2 Node Address	0	1
13-14 Bit 3 Node Address	0	1
15-16 Bit 4 Node Address	0	1
17-18 TD Watchdog Timer	1.2sec	150 ms.
19-20 Hardware Reset	Install & Remove	

6.0 <u>Installation</u>

All AMASSER ECN products, including the ECAIM, are DIN rail mountable. The mounting rails may be NS 15, NS 35/7,5, NS 35/15 or NS 32.

The ECN and power supply connections on the ECAIM are by means of a 5X2 position ribbon connector. The power supply connections are made to 5 V whereas the ECN connections are made to the standard serial COM port of your host PC.

The analog input is connected to a 3 position screw terminal on the ECAIM.

6.1 <u>Connectors</u>

1   5 x 2 position Ribbon Connector
1   3 position Screw Connector

<u>10 Pin Ribbon Connector</u>

RXD   -------------- 1   2 -------------- RXD*
TXD   -------------- 3   4 -------------- TXD*
5 Vcc   -------------- 5   6 -------------- 5 Vcc
none   -------------- 7   8 -------------- none
GND   -------------- 9   10 ------------ GND

<u>3 Position Screw Terminal</u>

Analog input (+) --------------- 1
Analog input (-) --------------- 2
none ---------------------------- 3

7.0 <u>Specifications</u>

Processor - Atmel 89C2051 @ 11.0592MHz.
Word Size - 8 bit data - 8 bit instruction
Memory - 89C2051, 2 Kbytes FLASH 128 bytes RAM

<u>Connectors</u>

ECNet & Power - 5 x 2 position Ribbon Connector
Analog Input - 3 position Screw Connector

<u>Analog Input</u>

Resolution	Linearity Error
16-Bits	0.0030%
20-Bits	0.0015%

<u>2.5 Volt Reference</u>

Temperature Drift - 5ppm/degree C

<u>Power Requirements</u>

+ 5V +/- 5% @ 150mA max.

Physical Characteristics

Height - 32.0 mm. (1.26 in.)
Width - 45.0 mm (1.75 in.)
Depth - 77.0 mm (3.00 in.)
Weight - 0.12 Kg (0.25 lb.)
Mounting - Universal Foot on module snaps on commercially available DIN EN mounting rails.

Environmental Characteristics

Temperature - At 61.5 Linear meters/min air velocity
Operating -40 to +85 C
Storage - -55 to +105 C
Humidity: <= 90% non-condensing

The above information is believed to be true at the time of printing. AMASS Data reserves the right to modify specifications without notice.
All trademarks are owned by their respective corporations.

AMASS Data Technologies Inc
34 Chemin Helene
Val des Monts, Quebec, CANADA
J8N 2L7
TEL: 1-819-457-4926
FAX: 1-819-457-9802
EMAIL: amassinf@amassdata.com

AMASS Data Technologies Inc
812 Proctor Ave., Box 707
Ogdensburg, New York
13669
TEL: 1-315-393-3793

Web site at **http://www.amassdata.com**

APPENDIX A: ECAIM Command Set: Quick Reference

ECN Set Commands: Set ECAIM Parameters

1.   Calibrate A-D Converter S0
- command: aS0<cksum><cr> (calibrate ADC)
- response: a<cksum><cr>

ECN Measure Commands: Read Analog Input Data

1.   Retrieve Hex Format ADC Data M0
- command: aM0<cksum><cr>
- response: a<Hex ADC Value><cksum><cr> where value is 4 or 5 ASCII hexadecimal characters
2.   Retrieve Decimal ADC Data M1
- command: aM1<cksum><cr>
- response: a<xx.xxx><cksum><cr> Leading zeros are suppressed.

ECN Network Commands:

1.   Change Node Address Command   A
- command: aAb<cksum><cr> where b = new node address
- response: b<cksum><cr>

2.   Send ECN Identification String   I
- command: aI<cksum><cr>
- response: a<identification string><cksum><cr>

3.   Send acknowledgment   !
- command: a!<cksum><cr>
- response: a<cksum><cr>

4.   Send address query   *!
- command: *!<cksum><cr>
- response: a<cksum><cr>

5.   Reset The Module #
- command: a#<cksum><cr>
- response: none Wait 10 milliseconds, then send Acknowledgment

APPENDIX B : Electrical Schematics