Technical Information Manual

Revision n. 2

21 October 2002

MOD. A1303

PCI CAENET
CONTROLLER

NPO:

00109/01:A1303.MUTx/02

CAEN will repair or replace any product within the guarantee period if the Guarantor declares that the product is defective due to workmanship or materials and has not been caused by mishandling, negligence on behalf of the User, accident or any abnormal conditions or operations.

CAEN declines all responsibility for damages or injuries caused by an improper use of the Modules due to negligence on behalf of the User. It is strongly recommended to read thoroughly the CAEN User's Manual before any kind of operation.

CAEN reserves the right to change partially or entirely the contents of this Manual at any time and without giving any notice.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Overview

The Mod. A1303 PCI CAENET CONTROLLER is a 32-bit PCI interface card. It allows the control of a HS CAENET network through a standard PC.

The communication line uses a simple 50 Ω coaxial cable as its physical transmission medium. In order to avoid ground loops line connectors are optocoupled.

A functional block diagram of an HS CAENET node is shown in Fig. 1.1

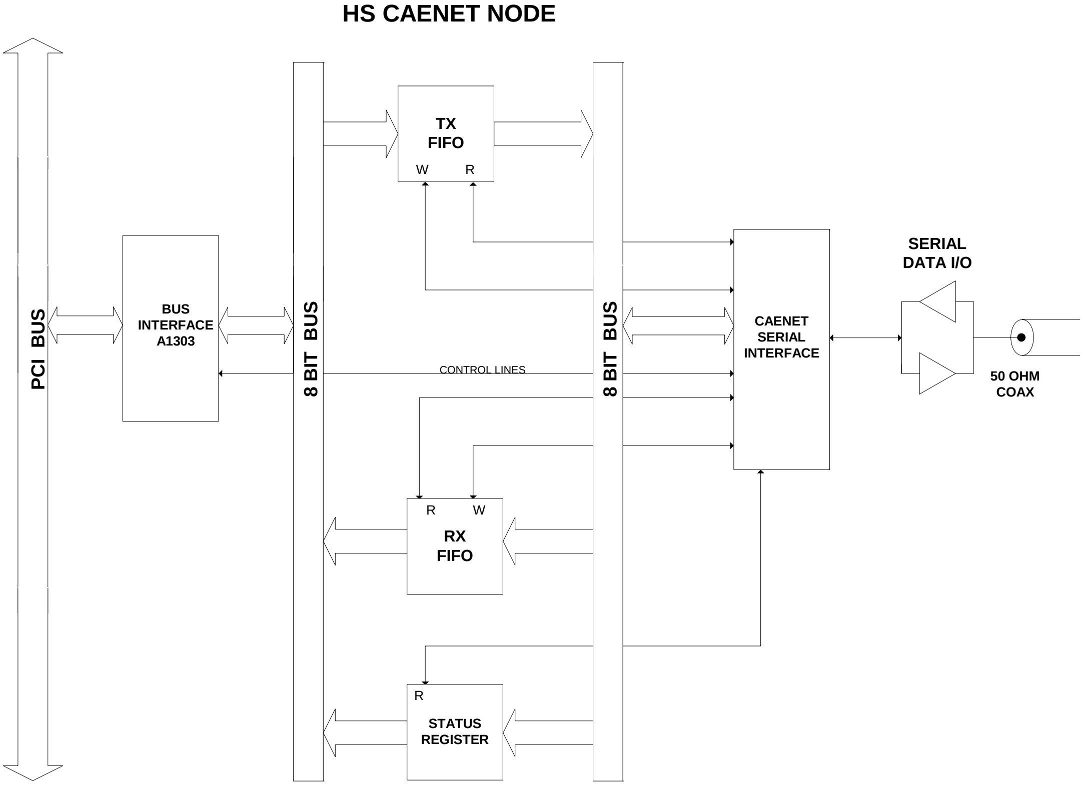

Fig. 1.1: HS CAENET node functional block diagram

2. Configuration and installation

2.1. Getting started

The Mod. A1303 is a “Plug and Play” card, any Operating System detects its installation or removal from the PC automatically; the card does not need to be manually configured by the User.

2.2. Hardware installation

The card must be plugged into one PCI slot (either 5 V or 3.3 V supplied) of the PC motherboard. Then it has to be connected to the HS CAENET line, via the two bridged LEMO type connectors. The line must be terminated on 50 Ω on both its sides.

2.3. Software installation

The Mod. A1303 drivers are available for download from the CAEN website, such drivers will allow to manage one or more A1303 cards installed on one PC. Moreover CAEN provides a C functions library[1] for the control of Caenet modules; useful demo programs are provided as well.

2.3.1. Driver installation

The Mod. A1303 drivers are available for the following platforms: Windows 98, Windows 2000, Windows XP, Windows NT, Linux kernel 2.2 and Linux kernel 2.4;

such drivers are available for download at:

http://www.caen.it/nuclear/product.php?mod=A1303

detailed instructions will guide the User through driver download and installation.

[1] The library and the demos are also suitable for the CAEN ISA BUS card Mod. A303 A

2.4. Software development

2.4.1. C functions library

CAEN provides, in addition to the Mod. A1303 drivers, a C function library which allows the User to both develop and run software tools for the modules control via HS CAENET. The following sections will describe carefully such functions.

2.4.1.1. *HSCAENETLibSwRel function

It returns the library revision serial number.

Synthax:

```
char   *HSCAENETLibSwRel(void);
```

2.4.1.2. HSCAENETCardInit function

It must be called before any other call to routines (see the following sub-sections) referred to the A1303 identified by the *param* argument. It initialises operating system dependent structures.

Arguments:

**CardName*	the card model, in this case A1303
**param*	a index (from 0 to 9) which identifies the plugged-in A1303's

Return value: -1: an error has occurred
≥1: successful initialisation

Synthax:

```
int     HSCAENETCardInit(const  char  *CardName,  const  void
*param);
```

2.4.1.3. HSCAENETSendCommand function

It sends a command to a module in the Caenet chain.

Arguments:

device	the A1303 which handles the command (the Return value of the HSCAENETCardInit function, see § 2.4.1.2)
Code	the code which identifies the command (see the relevant module's User's manual)
CrateNumber	it identifies, by the Caenet crate number, the module which receives the command
**SourceBuff*	a byte-array containing arguments eventually needed by the specified command
WriteByteCount	length (bytes) of the **SourceBuff* byte-array

Return value: =0: command sent successfully
≥1: error during communication with the A1303 (see § 2.4.1.9)
<0: problem occurred inside the module which receives the command (see the relevant module's User's manual)

Synthax:

```
int HSCAENETSendCommand(int device, int Code, int
CrateNumber, void *SourceBuff, int WriteByteCount);
```

2.4.1.4. HSCAENETReadResponse function

It waits for a response from a module in the Caenet chain.

Arguments:

device the A1303 which handles the command (the Return value of the HSCAENETCardInit function, see § 2.4.1.9)

**DestBuff* a byte-array containing the response from the module

**ReadByteCount* the length (bytes) of the **DestBuff* byte-array

Return value: =0: command sent successfully
≥1: error during communication with the A1303 (see § 2.4.1.9)
<0: problem occurred inside the module which receives the command (see the relevant module's User's manual)

Synthax:

```
Int HSCAENETReadResponse(int device, void *DestBuff, int
*ReadByteCount);
```

2.4.1.5. HSCAENETComm function

It sends a command to a module in the Caenet chain, then it waits for a response.

Arguments:

device the A1303 which handles the command (the Return value of the HSCAENETCardInit function, see § 2.4.1.2)

Code the code which identifies the command

CrateNumber it identifies, by the Caenet crate number, the module which receives the command

**SourceBuff* a byte-array containing arguments eventually needed by the specified command

WriteByteCount length (bytes) of the **SourceBuff* byte-array
**DestBuff* the response from the module

Return value: =0: command sent successfully
≥1: error during communication with the A1303 (see § 2.4.1.9)
<0: problem occurred inside the module which receives the command (see the relevant module's User's manual)

Synthax:

```
int HSCAENETSendCommand(int device, int Code, int CrateNumber, void *SourceBuff, int WriteByteCount, void *DestBuff)
```

2.4.1.6. HSCAENETCardReset function

It resets the A1303.

Arguments:

device – the A1303 which is reset (the Return value of the HSCAENETCardInit function, see § 2.4.1.9)

Synthax:

```
int   HSCAENETReset(int device);
```

2.4.1.7. HSCAENETTimeout function

It sets the time to wait for a response from a module in the Caenet chain.

Arguments:

device – the A1303 which is waiting for a response (the Return value of the HSCAENETCardInit function, see § 2.4.1.9)

Timeout – time to wait in tenth of seconds.

Synthax:

```
int   HSCAENETTimeout(int device, unsigned long Timeout);
```

2.4.1.8. HSCAENETCardEnd function

It must be called after any other call to routines declared here.
It resets operating system dependent structures.

Arguments:

device – the A1303 whose task is ending (the Return value of the HSCAENETCardInit function, see § 2.4.1.9)

Synthax:

```
int   HSCAENETCardEnd(int device);
```

2.4.1.9. Error codes

This is the list of the codes which identify the errors that can occurr while communicating with the A1303:

Code:	Name:	Description:
1	E_OS	Operating system error
2	E_LESS_DATA	The received Data are less than what expected
3	E_TIMEOUT	'timeout' (see § 2.4.1.7) exceeded
4	E_NO_DEVICE	'device' argument is not associated to any A1303 (see § 2.4.1.2)
5	E_A303_BUSY	A303A already busy in data transfer (not used with the A1303)

3. Technical specifications

3.1. Packaging

Standard 32-bit 33 MHz universal 3.3 V / 5 V PCI card

3.2. External components

CONNECTORS: 2 bridged LEMO 00 type communication line connectors, high impedance[2].

DISPLAYS: HS CAENET active LED

3.3. Physical line and node capabilities

Output Driver Signal characteristics:	0 to +4 V on 50 Ω impedance
Input Receiver characteristics:	+1.2 V threshold discriminator
RG174 Cable attenuation:	-6.2 dB/100 m @ 1 MHz
HS CAENET Node attenuation:	≈ -0.07 dB @ 500 kHz

It is worth noting that the maximum frequency of the HS CAENET signal is 500 kHz (a stream of bits toggling between 0 and 1 transmitted at the rate of 1 MBaud).
Through the use of appropriate adapters, one can connect the HS CAENET nodes via an RG58 cable, that has better attenuation features (-1.4 dB/100 m @ 1 MHz) at approximately the same cost.

3.4. Power requirements

PCI class 7.5 W maximum

[2] The HS CAENET line must be terminated on 50 Ω on both sides.

4. Mod. A1303 hardware description

4.1. HS CAENET Network operations

HS CAENET Network is a send and receive half duplex system; it permits asyncronous serial transmission (1Mbaud rate) of data packet along a simple 50 Ω coaxial cable. Several devices (HS CAENET nodes) are able to share the same line both to transmit and receive data.

Each node is able to receive the serial data packet and store it automatically in the RX FIFO and transmit the data contained in the TX FIFO. Both FIFOs are 4096 byte deep (see § 4.3.1 and § 4.3.5).

The HS CAENET node listen for clear coax before transmitting but it is not able to detect collisions on the cable; for this reason it is important to avoid line contention i.e. the nodes should not attempt to transmit at the same time.

Usually transfers between HS CAENET nodes take place according to the typical MASTER/SLAVES communication: there is a single HS CAENET MASTER that initiates the transmission, all the SLAVEs receive the data, and only the SLAVE addressed then accesses the serial line to transmit the data requested by the MASTER.
The maximum data packet length is 4096 bytes.

4.2. HS CAENET Node operation

Basically an HS CAENET node can work in 3 distinct modes: Transmit, Receive and Restart.

- in the Transmit mode the node accesses the data stored in the TX FIFO and transmits the data on the cable.
- in the Receive mode the serial packet is stored in the RX FIFO.
- in Restart mode the node does not accept any commands, all the TX and RX buffers are cleared and the interrupt is removed; it remains in this mode until the line is cleared.

The A1303 card directly interfaces the 32-bit PCI bus with the two FIFO buffers (TX and RX FIFO, see Fig. 1.1), and with 6 internal registers which are used for various functions such as FIFOs clearing, starting transmission and reading the node status.
The Host processor can control the node operation in Polling or Interrupt mode.
Interrupts to the CPU are generated by the A1303:

- when the transmission of a data packet has been completed
- at the reception of a data packet
- when the RX FIFO has been completely unloaded.

4.3. Registers and buffers addressing

The address map of the A1303 is shown in the following Figure (each location is 8 bit wide)

Register/Buffer	Operation	Address
TX FIFO	WR	BASE ADDRESS + 0
START TX	WR	BASE ADDRESS + 1
LED	WR	BASE ADDRESS + 2
RESET	WR	BASE ADDRESS + 3
RX FIFO	RD	BASE ADDRESS + 0
STATUS REGISTER	RD	BASE ADDRESS + 1
RESET INTERRUPT (rd STATUS)	RD	BASE ADDRESS + 2
CLEAR RX FIFO	RD	BASE ADDRESS + 3

Table 4.1: Address map of Model A1303

4.3.1. TX FIFO

(Base address + 0, write only)

This is the buffer which is loaded with the data to be transmitted; it is arranged in a FIFO logic. By writing at this location the HS CAENET active LED lights up.

4.3.2. START TX

(Base address + 1, write only)

By writing at this location the node enters into the transmit mode; and the HS CAENET LED turns on.

4.3.3. LED

(Base address +2, write only)

By writing at this location the HS CAENET active LED turns on.

4.3.4. RESET

(Base address + 3, write only)

A write access to this location causes the node to enter Restart mode; this causes the following operations:

- the buffers TX FIFO and RX FIFO are cleared
- every interrupt pending is cleared
- every data transfer is aborted
- the HSCAENET LED turns off
- the node does not accept any command

It remains in this status until the line is cleared.

4.3.5. RX FIFO

(Base address + 0, read only)

This is the buffer where the node automatically stores the received data; it is arranged in a FIFO logic.

4.3.6. STATUS REGISTER

(Base address + 1, read only)

This register contains the status bits of the HS CAENET node; in particular, the cause of the interrupt can be ascertained by reading the STATUS REGISTER (see § 4.5 for the status bits description).

4.3.7. RESET INTERRUPT (rd STATUS)

(Base address + 2, read only)

At this address the status register previously described is available, moreover a read access causes the following operations:

- the HS CAENET interrupt is removed (if asserted)
- the three interrupt status bits RFEFF, RXEFF and TXEFF are reset to 1
- the HS CAENET LED turns off.

4.3.8. CLEAR RX FIFO

(Base address + 3, read only)

A read access to this address clears the receive buffer RX FIFO.

4.4. HS CAENET LED

The CAENET LED turns on in these cases:

- after a write access to Base address + 2 (LED);
- after a write access to Base address + 0 (TX FIFO); i.e. when data are stored in the TX FIFO;
- after a write access to Base address + 1 (START TX);

It remains on until one of the following operations is performed:

- a write access to Base address + 3 (RESET);
- a read access to Base address + 2 (RESET INTERRUPT and read STATUS);

4.5. Status register definition

The following tables show the Status register structure:

Table 4.2: Status register structure

7	6	5	4	3	2	1	0
TXACT	RXACT	TXEFF	TFEM	RESTART	RXEFF	RFEFF	RXFEM

Table 4.3: Status register description

BIT	NAME	DESCRIPTION
0	RXFEM	RX FIFO EMPTY: IF =0 THE RX FIFO IS EMPTY. RXFEM IS SET TO 1 WHEN VALID DATA ARE PRESENT IN THE RX FIFO, AND IS CLEARED AFTER THE RX FIFO HAS BEEN UNLOADED. THIS BIT IS ALSO CLEARED AFTER A RESET OPERATION OR AFTER AN RX FIFO CLEAR.
1	RFEFF	RX FIFO EMPTY: IF =0 THE RX FIFO HAS BEEN COMPLETELY UNLOADED. RFEFF IS SET TO 0 AFTER THE LAST DATA OF THE RECEIVED PACKET HAS BEEN READ FROM THE RX FIFO. RFEFF IS RESET TO 1 AFTER A RESET OPERATION OR AFTER A RESET INTERRUPT.
2	RXEFF	RX END: IF =0 THE HS CAENET NODE HAS RECEIVED A DATA PACKET. RXEFF IS SET TO 0 AT THE END OF THE RECEPTION. RXEFF IS RESET TO 1 AFTER A RESET OPERATION OR AFTER A RESET INTERRUPT.
3	RESTART	RESTART MODE: IF =0 THE HS CAENET NODE IS IN RESTART MODE; RESTART IS SET TO 0 AFTER A RESET OPERATION, IN THIS STATE THE HS CAENET NODE DOES NOT ACCEPT ANY COMMAND. RESTART IS RESET TO 1 WHEN THE NODE DETECTS THAT THE LINE IS CLEAR.
4	TFEM	TX FIFO EMPTY: IF =0 THE TX FIFO IS EMPTY. TFEM IS SET TO 1 WHEN DATA ARE STORED IN THE TX FIFO, AND IS CLEARED WHEN THE TX FIFO HAS BEEN UNLOADED BY THE HS CAENET NODE OR AFTER A RESET OPERATION.
5	TXEFF	TX END FLIP FLOP: IF =0 THE HS CAENET NODE HAS TRANSMITTED A DATA PACKET. TXEFF IS SET TO 0 AT THE END OF THE TRANSMISSION. TXEFF IS RESET TO 1 AFTER A RESET OPERATION OR AFTER A RESET INTERRUPT.
6	RXACT	RECEIVER ACTIVE: IF =0 THE HS CAENET IS IN RECEIVE MODE. RXACT IS SET TO 0 WHEN THE NODE STARTS TO RECEIVE A DATA PACKET. RXACT IS RESET TO 1 AT THE END OF RECEPTION OR AFTER A RESET OPERATION.

4.6. Interrupt generation

Interrupts to the CPU are generated by the A1303:

– after the transmission of a data packet has been completed: TXEFF goes to level = 0
– at the end of reception of a data packet: RXEFF goes to level = 0
– when the RX FIFO has been completely unloaded: RFEFF goes to level = 0

4.7. Interrupt release

The A1303 removes its interrupt and the corresponding interrupt bit (s) are reset to 1 after the following operation:

– read access to the RESET INTERRUPT location (Base address + 2).
– write access to the RESET location (Base address + 3).