Data Device Corporation

Multi-Protocol Data Bus Interface

NHi-ET Enhanced Terminals
Bus Controller, Remote Terminal, Bus Monitor

User's Manual

Version 2014.01.09
January 2014

105 Wilbur Place, Bohemia, NY 11716

1-800-DDC-5757 | 631-567-5600

service@ddc-web.com | www.ddc-web.com

TABLE OF CONTENTS

TABLE OF CONTENTS(continued)

TABLE OF CONTENTS(continued)

TABLE OF CONTENTS(continued)

TABLE OF CONTENTS(continued)

1.0.0 SCOPE

This document defines the functional and electrical specification for National Hybrid's series of MIL- STD- Data Bus Enhanced Terminals (NHi- ET).

2.0.0 NHi-ET PROTOCOL COMPLIANCE

MIL- STD- 1553A
MIL- STD- 1553B Notices I and II
MIL- STD- 1760B
MCAIR MDC A3818, A5690, A4905, A5332
EFA/ STANAG- 3838 requirements for Eurofighter Aircraft

3.0.0 INTRODUCTION

The NHi- ET is a low cost complete **Multi-Protocol** Mil- Std- Data Bus Interface between a dual redundant bus and a host processor. The device functions as a programmable Bus Controller, Remote Terminal, and Bus Monitor containing a protocol chip, two +5V monolithic transceivers and 16K or 64K word SRAM. The unit is available packaged in a 1.1" x 1.1" 69 pin ceramic PGA, or 1.1" x 1.1" 68 pin ceramic quad flatpack. The only external components required are two coupling transformers.

The NHi- ET appears to the host computer as 16K or 64K words of 16 bit wide memory controlled by standard RAM signals. The device can thus be easily interfaced with all popular processors and buses. The built in interrupt controller supports an internal FIFO which retains header information for queuing up to 6 pending interrupt requests plus an overflow interrupt.

All modes of operation access data tables via pointers residing in RAM which facilitates multiple buffering. This allows buffers to change without moving data and promotes efficient use of RAM space. The data tables have programmable sizes and locations.

The NHi-ET is plug in compatible with the popular NHi-RT family of remote terminal with no changes to hardware or software required. The NHi-ET defaults to the NHi-RT remote terminal operation on power up.

3.1.0 FEATURES

The NHi- ET is form, fit, and function compatible to the NHi- RT series of parts. This interchange ability gives the user a high degree of flexibility when configuring a system around the NHi family of parts.

3.1.1 GENERAL FEATURES

- Mulit-Protocol Interface
- Single +5 volt supply.
- Operates from 10 Mhz clock.
- Contains two monolithic +5V transceivers
- Appears to host as a Dual Port Double Buffered 16K or 64K x 16 SRAM
- Footprint less than 1.25 square inches
- Ensures integrity of all shared data and control structures
- Built- in interrupt controller
- Internal FIFO is configurable to retain header information for queuing up to 6 pending interrupt requests plus an overflow interrupt, or as a 7 interrupt revolving FIFO
- Provides interrupt priority input and output pins for daisy- chaining interrupt requests
- Contains a Timer Unit which provides 32 bit RTC (Real- Time- Clock) with 1, 2, 4, 8, 16, 32 and 64 uS internal, or user provided external clock resolution for data and event time tagging.
- Interfaces with an 8 bit discrete I/ O bus
- Selectable 768/ 672 us Failsafe Timer with complete Testability
- Low power CMOS technology

3.1.2 Bus Controller Highlights:

- Implements all Message Formats and Error Checking
- Simple setup and operation. Preset multiple pointer tables and message blocks. Only two Frame Pointer and Frame Length Registers are required to control unlimited number of message blocks
- BC initialized by writing to three Configuration Registers and the Interrupt Mask Register
- Executes lists of messages via Message Frame
- Configurable Local Retry and Interrupt Requests Enabled on Message by Message Basis
- Configurable Global Retry and Local Retry
- Programmable retries per message:
  - None
  - Retry Current Bus
  - Retry Alternate Bus
  - Retry Alternate Bus then Current Bus.
- Programmable response timeout of 14, 18, 26, or 42 microseconds.
- Programmable Intermessage Gap Time up to 4 mS with 1 uS resolution.
- Extended Intermessage Gap using NO- OP Feature.
- Programmable Frame Gap with 64 uS resolution.
- Programmable Interrupts for:
  - End of Message
  - End of Frame
  - Response Time Out, Message
  - Error
  - Message Retry
  - RT Status Bit Set
  - FIFO Overflow.
- Non- Maskable Bus Jam Interrupt.
- Host controlled commands:
  - Start BC
  - Continuous Mode
  - Stop at End of Message
  - Stop at End of Frame
  - Abort,
  - GOTO Alternate Frame.
- Dynamic Bus Switch Upon Successful Retry.

3.1.3 Remote Terminal Highlights:

- Dynamic Bus Control Acceptance
- DBCA_ L bit is set in configuration register.
- Message Illegality is internally programmable. DOES NOT require external PROMS or glue logic.
- Employs data tables with individual tag words which indicate whether or not the data is valid, updated since last read, in the process of being updated, was received via broadcast command, or has been lost (i. e. updated more than once by a receive message before being read).
- Optionally sets the subsystem flag bit whenever stale data is transmitted or received data is overwritten.
- Issues interrupts on any subset of T/ R bit, subaddresses, mode commands, broadcast messages and errors.
- Provides interrupt priority input and output pins for daisy- chaining interrupt requests. messages.
- Optionally resets the real- time clock in response to a "Synchronize" mode command.
- Optionally updates the lower 16 bits of the real- time clock in response to a "Synchronize WithData" command.

•Indicates the reception of specific commands by outputting pulses on any one of 8 pins.
•Internally loops- back messages under host control for test purposes.
•Employs a decoder algorithm which ensures high noise immunity and a low error rate.
•Software RT Address Lockout.
•MDC3818 Status Response, Error Handling, Status Bit Definition, Mode Code Operation.
•Separate Broadcast Interrupts.

3.1.4 Bus Monitor Highlights:

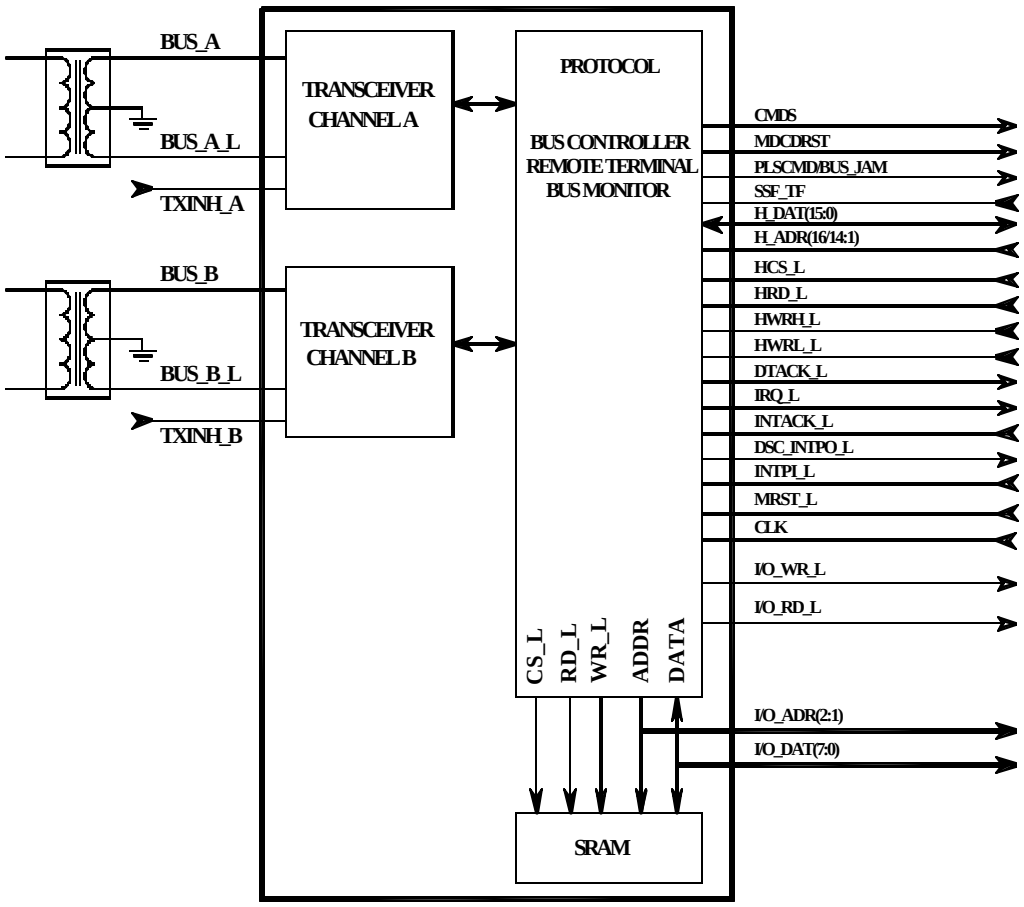
•Simple setup and operation
•Preset multiple data blocks.
•Only two MT Data Start Address and MT Data End Address Registers are required to control unlimited number of message blocks. The data block sizes and locations are totally Programmable.
•MT initialized by writing to three MT Configuration Registers and the MT Interrupt Mask Register.
•Error detection and reporting
•All encoding, timing and protocol errors defined by the Protocols are detected.
•Programmable Monitor Modes:
- Word Monitor, transfers all data with/ without ID and Time Tag words.
- Message Monitor, transfers all Command and Status words with/ without ID and Time Tag , while data words are transferred directly to conserve memory space.

•Concurrent Bus Monitor and Remote Terminal operation.
•Selective Message Monitor, based on RT Address.
•Programmable Interrupt for End of Block and End of Frame.

3.2.0 BLOCK DIAGRAM

The NHi- ETs contains two +5 volt transceivers, an ASIC, and an SRAM. The ASIC performs all multi protocol functions; BUS CONTROLLER, BUS MONITOR and REMOTE TERMINAL. It controls accesses to the RAM such that it appears to the host CPU 16bit wide dual port memory.

Since the NHi- ET appears to its host as RAM, no external logic is required when interfacing to the device. It is simply connected to the CPU's address bus, Mil Bus, and control lines. There are NO EPROMS required to illegalize commands in the RT mode. Illegalization is performed internal to the protocol chip in the NHi- ET.The user sets up command illegalization when the NHi- ET is initialized. See sections on Message Illegalization and Host Initialization.



NHi-ET ENHANCED TERMINAL FUNCTIONAL BLOCK DIAGRAM

The NHi- ET can be interfaced to an 8 bit CPU Bus by folding the upper and lower bytes on top of each other and performing byte wide data transfers.

By default, the host has priority in accessing the I/ O bus. When the host requests access to a device already in use by the protocol chip, the host *DTACK signal is delayed by the NHi- ET. If either side (protocol chip or host) waits for access during the current cycle, it is automatically granted priority for the next cycle. The host can retain priority for successive cycles accessing the same address (this is required to guarantee the proper operation of host read- modify- write instructions - see pin *HCS for details) by keeping *HCS low.

3.3.0 PROTOCOL CHIP DESCRIPTION

The protocol chip contains the following modules:

Host Bus Interface Unit	(HBIU)
I/ O Bus Interface Unit	(IBIU)
Interrupt Controller Unit	(ICU)
Dual Redundant multi protocol Front End	(DRFE)
Message Processor Unit	(MPU)

3.3.1 HOST BUS INTERFACE UNIT

The HBIU provides a standard RAM interface to the host bus. The module performs the following functions:

Provides NHi- ET device select and decodes host address to select registers.
Transfers data between the NHi-ET and the host (word and byte mode as well as read-modify- write are supported).
Provides priority input and output for daisy chaining host interrupts.
Outputs *DTACK signal indicating end of bus cycle.

3.3.2 I/O BUS INTERFACE UNIT

The IBIU controls the RAM and I/ O residing on the I/ O bus so that it appears to the host as a pseudo dual port RAM (i. e., shared memory). The unit implements the following functions:

Arbitrates between host and protocol chip initiated accesses to the RAM and host data bus.
Decodes address lines to select device (e. g. RAM, external byte- wide I/ O, external terminal address buffer, command output register).
Generates control signals to access the selected device.

3.3.3 INTERRUPT CONTROL UNIT

The ICU is an 8 input vectored interrupt controller. It contains eight registers as well as a FIFO for storing pending interrupt vectors.

3.3.3.1 ICU REGISTERS

The ICU contains the following registers

INTERRUPT REQUEST register	(IRR)
INTERRUPT MASK register	(IMR)
INTERRUPT VECTOR register	(IVR)
AUXILIARY VECTOR register	(AVR)

The INTERRUPT REQUEST register samples 8 inputs originating from internal modules. Since the host can write to this register, all interrupt sequences can be software driven for program debugging. The inputs and their priorities (level 7 has highest priority) are described in the following table.

3.3.3.1.1 INTERRUPT DEFINITION TABLE

PRIORITY	RTU INTERRUPT	BCU INTERRUPT	MTU INTERRUPT
0	VALID TX/RX EOM	END OF MESSAGE	N/A
1	INVALID TX/RX EOM	END OF FRAME	N/A
2	VALID MODE CODE	ERROR	N/A
3	INVALID MODE CODE	RETRY	N/A
4	FIFO OVERFLOW	FIFO OVERFLOW	FIFO OVERFLOW
5	VALID BROADCAST	STATUS SET	END OF FRAME
6	INVALID BROADCAST	NO RESPONSE	N/A
7	FAILSAFE TIMEOUT	FAILSAFE TIMEOUT	N/A

Note: RT Interrupts 5 & 6 are enabled only when separate Broadcast Tables are used. Masking interrupt 4 creates a revolving Fifo.

As soon as an interrupt is requested, its vector is pushed onto the FIFO - so the chronological order of the requests normally determines the order in which they will be serviced. Simultaneous requests, however, are pushed onto the FIFO according to the priority of the pending interrupts.

The INTERRUPT MASK register masks the corresponding inputs to the INTERRUPT REQUEST register. The INTERRUPT VECTOR register holds the 3 bit interrupt priority level and an additional 5 bit field (see paragraph on INTERRUPT VECTOR register for details).

The AUXILIARY VECTOR register contains an additional byte of information related to the interrupt request (see paragraph on AUXILIARY VECTOR register for details).

3.3.3.2 ICU FIFO

The ICU FIFO is 16 bits wide and 7 words deep. Whenever an unmasked interrupt request is issued by the message processor, a word is pushed onto the FIFO. When an interrupt is acknowledged by the host, a word is popped from the FIFO and used to update the IVR and the AVR.

The host can read the FIFO by simply popping its contents. This is done by reading the FIFO located at address 8 (refer to address map). The interrupt request output, *IRQ, will go inactive after the FIFO is emptied in this way.

The host can mask the *IRQ output by resetting the INTERRUPT REQUEST ENABLE bit in the CONTROL register; however this does not prevent the device from pushing interrupt requests onto the FIFO.

If an interrupt request occurs when the FIFO is full, a vector indicating FIFO overflow is first pushed onto the FIFO and then the vector which caused the overflow is pushed onto the FIFO. As a result, the 2 oldest vectors are lost. All further pushes are then inhibited until the host pops the vector indicating the overflow.

The above mechanism ensures that the host will always be notified of FIFO overflows and will always obtain the 2 interrupt vectors immediately preceding the overflow condition.

If interrupt 4 is masked, the FIFO operates in the revolving mode; vectors are continuously pushed onto the FIFO. After the 7th vector is pushed without any pops, each additional vector pushed causes the oldest vector to be lost.

The FIFO can be emptied by writing (any value) to address 8 (in words).

3.3.4 DUAL REDUNDANT FRONT END

The DRFE performs serial to parallel and parallel to serial conversion as well as basic format and timing validation. The unit contains the following:

Manchester encoders/ decoders
Gap counter
No response counter
Minimum response time counter
Timeout counter

3.3.4.1 MANCHESTER DECODER

The decoder translates serial Manchester bi- phase signals to 16- bit words and outputs the following signals:

Valid command word received
Valid data word received
Invalid word received (parity, incorrect bit count, invalid Manchester encoding, gap)
Broadcast command received
Begin new message (i. e., end of a valid legal command for this Remote Terminal)

3.3.4.2 MANCHESTER ENCODER

The encoder receives 16 bit words and transmits them with the appropriate sync and parity as a serial Manchester bi- phase signal. The outputs of the encoder can be loop- backed into either decoder for test purposes.

3.3.4.3 GAP COUNTER

The gap counter checks contiguity of successive words. If the time between "contiguous" words (measured from zero- cross of parity to zero- cross of sync) exceeds 3.5 - 3.7 microseconds, the message is invalidated.

3.3.4.4 RT - RT NO RESPONSE COUNTER

The no response counter checks the response time of the transmitting RT in a RT to RT transfer. If the response time is exceeded, the message is invalidated. The response time is software programmable (14, 18, 26, 42 microseconds) to accommodate systems with long cables and/ or slow terminals.

3.3.4.5 MINIMUM RESPONSE TIME COUNTER

The minimum response time counter ensures that the response will be no sooner than 4 microseconds
(measured from zero- cross of parity to zero- cross of sync).

3.3.4.6 FAIL -SAFE TIMEOUT COUNTER

This counter inhibits the encoder outputs and issues a TIMEOUT interrupt whenever continuous transmission exceeds 768/ 672 microseconds. Transmission will remain inhibited until a command is received on the same bus or the part is reset.

3.3.5 MESSAGE PROCESSOR UNIT

The MPU forms the heart of the protocol chip and controls the operation of the Decoders, Encoders, and Interrupt Controller. This unit is activated by the reception of a valid legal command addressed to the RT in the RT mode and the START bit in CONFIGURATION 1 in both the BC and MT modes.

The MPU performs the following functions:

Recognizes the various message types (for BC, MT, and RT) and responds with the appropriate sequence of control signals.
Validates format and timing of received data words.
Checks command legality.
Responds with status/ data.
Calculates all addresses for accessing the RAM and discrete I/ O.
Updates RAM data table contents, including tag words.
Optionally time tags data tables.
Issues interrupt requests to the ICU.
The maximum response time of the NHi- ET in the RT mode is less than 6.0 microseconds (measured from zero- cross to zero- cross).

3.4.0 RT HARDWIRE TERMINAL ADDRESS

The terminal address of the NHi- ET can be hardwired using I/ O DAT(5: 0). I/ O DAT(4: 0) are used for the terminal address, I/ O DAT0 being the LSB, and I/ O DAT5 is used to set odd parity in the address. These pins CANNOT be directly connected to +5 or ground since the I/ O data bus drives the NHi- ET's internal RAM.

The address must be wired using pull- up and pull- down resistors. There are 64K internal pull- up resistors in the protocol chip, so only external pull- down resistors of 4.7K are required. The Hardwire Address is read and loaded into the terminal at Power- On Reset, Hardware Reset, and Software Reset.

The terminal address can be changed at any time through software by writing a new address to the Basic Status Register, however, if any of the above resets occur, the Hardwire Address will be re- loaded into the terminal. The software address can be locked out by setting Bit2 in Configuration Register 1.

4.0.0 DATA STRUCTURE

4.1.0 ADDRESS MAP

The NHi- ET appears to the host as 16K or 64K words of memory divided into the following blocks:

ADDRESS RANGE	DESCRIPTION
0 -- 30	INTERNAL REGISTERS
31	I/O TAG WORD
32 -- 35	I/O SPACE
64 -- 16383/65535	SHARED RAM

INTERNAL REGISTER MAP

ADDRESS	REGISTER DEFINITION	ACCESS
0	CONTROL	R/W
1	POINTER TABLE ADDRESS	R/W
2	BASIC STATUS	R/W
3	INTERRUPT MASK(lower byte)	R/W
3	INTERRUPT VECTOR(upper byte)	R
3	INTERRUPT REQUEST(upper byte)	W
4	INTERRUPT VECTOR(lower byte)	R/W
4	AUXILLARY VECTOR(upper byte)	R
4	CONFIGURATION 2(upper byte, BCU/MTU only)	W
5	REAL TIME CLOCK HIGH WORD	R
6	REAL TIME CLOCK LOW WORD	R
7	REAL TIME CLOCK CONTROL	R/W
8	READ FIFO	R
8	RESET FIFO	W
9	CONFIGURATION 1	R/W
10	RESERVED	
11	LAST COMMAND	R
12	LAST STATUS	R
13	FRAME “A” LOCATION/BLOCK “A” START	R/W
14	FRAME “A” LENGTH/BLOCK :A” END	R/W
15	RESET TERMINAL(both bytes)	W
16	FRAME “B” LOCATION/BLOCK “B” START	R/W
17	FRAME “B” LENGTH/BLOCK “B” END	R/W
18	ENCODER STATUS	R
19	CONDITION	R
20	BCU FRAME GAP/WORD MTU END OF FRAME OPTIONS	R/W
21	CONFIGURATION 3	R/W
22	MESSAGE MONITOR ADDRESS FILTER(0 -- 15)	R/W
23	ENCODER DATA*	R/W
24	ENCODER DATA TX REQUEST*	W
25	ENCODER COMMAND TX REQUEST*	W
26	MESSAGE MONITOR ADDRESS FILTER(16 -- 31)	R/W
27	WORD MONITOR BLOCK “A” LAST ADDRESS	R
28	WORD MONITOR BLOCK “B” LAST ADDRESS	R
29	RESERVED	
30	EXTERNAL RTU ADDRESS BUFFER(lowre byte)	R
30	COMMAND OUTPUT PINS	W
31	I/O TAG WORD	R/W

*In order to write to addresses 23, 24, or 25, the ET must be in loop- back in the RT mode (see CONTROL register for details).

4.2.0 INTERNAL REGISTERS

4.2.1 CONTROL Address: 0 R/ W BC/ MT/ RT

This register controls the general operation of the NHi- ET.

15	14	13	12	11	10	9	8
HWD	RSP1	RSP0	TSTFST	NBCST	TXINH	LOOPB	LOOPA
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IRE	MIO	CMDO	SRQRST	SSF_TF	NTAG	BINH	AINH

HWD **Bits: 15** **BC/ RT**

1 = Enables high word detection.
This option allows extra words in a message to be detected,as required by some protocols.
0= Tterminal does not detect high word errors.

RSP1, RSP0 **Bits: 14,13** **BC/ RT**

These bits define the response timeout for RT- RT messages in the RT mode and terminal response timeout in the BC mode as follows:

RSP1	RSP0	TIMEOUT(us)
0	0	14
0	1	18
1	0	26
1	1	42

TSTFST **Bits: 12** **RT**

1= Enables testing of the FAIL SAFE time out.
When this feature is enabled, the RT will transmit continuously once it is enabled by a valid message. The encoder will be inhibited after 768/ 672us. It will be enabled by a reset or the reception of another valid message. If this bit is set to 0 during an RT transmission, before the required number of words have been transmitted, the encoder will return to normal operation and stop at the proper message length.. If it is set to 0 after the message length has been exceeded, the current word will be completed and normal operation resumed.
This feature can be used in the LOOPBACK mode to automatically transmit data words.
The RT encoder will remain in the tester mode until the CPU sets this bit to 0.

The TSTFST Bit Must Always Be Set to Zero During Normal Operation!!!

NBCST **Bits: 11** **RT**

1= Specifies that broadcast commands WILL be ignored by the RT.

TXINH **Bits: 10** **BC/ RT**

1= Inhibits transmission by forcing TXA= TXAN= 0 and TXB= TXBN= 0.

LOOPA(B) **Bits: 9, 8** **RT**

1= Defines that decoder A (B) inputs shall be connected internally to the encoder outputs rather than the transceiver for test purposes.

IRE **Bits: 7** **BC/ MT/ RT**

1= Globally enables the interrupt request output, *IRQ.
0= Disables all interrupt requests; however, interrupt vectors are still pushed onto the FIFO.

MIO **Bits: 6** **RT**

1= Defines that certain reserved mode commands with data shall be legal and access the I/ O bus without dependence on host initialization or the BUSY bit in the BASIC STATUS register. This feature can be used, for example, to set a watchdog timer or read a hardware status register via the Mil Bus even though the host's state may be undefined.

The I/ O operations are restricted to the data word's lower byte. The mode commands and their corresponding I/ O addresses in decimal are as follows:

T/R	MODE CODE	I/O ADR(2,1)	I/O WR_L	I/O RD_L
T	24	00	1	0
T	25	10	1	0
R	27	10	0	1
R	28	00	0	1

CMD0 **Bits: 5** **RT**

0= Specifies that after a legal valid command is received, a pulse shall be outputted on a pin specified by the PULSE field in the corresponding data table tag word. The pulse is activated together with 2 I/ O control signals (CMDS= 1 and *I/ O WR = 0).

1= Specifies that after a valid legal command is received, the word count/ mode code field (together with CMDS= 1 and *I/ O WR = 0) shall be outputted on the 5 least significant bits of the discrete I/ O bus. (Although the protocol chip outputs the entire command, only 5 bits are outputted by the NHi- RT due to pin- out restrictions).

SRQRST **Bits: 4** **RT**

1= Specifies that the service request bit in the STATUS word will be reset upon reception of a valid "Transmit Vector Word" mode command.

SSF_ TF **Bits: 3** **RT**

0= Specifies that the Sub- System Flag in the Status Word will be determined by the value of the SSF_ TF pin.

1= Specifies that the Terminal Flag in the Status Word will be determined by the value of the SSF_ TF pin.

NTAG **Bits: 2** **RT**

1= Specifies that all the data tables shall be without tag words. This mode of operation can be used to store received data from several subaddresses into a contiguous block without interspersed tag words. This feature can facilitate, for example, software upload.

BINH **Bits: 1** **BC/ RT**

1= Disables reception on bus B.

AINH **Bits: 0** **BC/ RT**

1= Disables reception on bus A.

4.2.2 POINTER TABLE ADDRESS **Address: 1** **R/ W** **RT**

This register holds the address of the table of pointers used in the RT mode when accessing data tables. The address is specified as a word address in the lower 4K of the memory space. After POR the register is initialized to 1000 (hex), with D1 as the LSB of the word address. D0 is a DON'T CARE and should be set to 0.

Note: The RT pointer table must always be located in the lower 4K words of memory.

4.2.3 BASIC STATUS Address: 2 R/ W RT

This register defines the terminal address as well as default values for all status bits. The Status Word is OR'ed with this register before transmission. The bits in the BASIC STATUS register correspond to the bits in the STATUS register and their function is defined in MIL- STD- 1553B. They can be redefined for other protocols.

15	14	13	12	11	10	9	8
TADR4	TADR3	TADR2	TADR1	TADR0	M_ERR	INSTR	SREQ
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RSVD2	RSVD1	RSVD0	BCR	BUSY	SSF	DBCA	TF

The mechanism employed by the protocol chip for initializing the terminal address is designed to avoid dedicated pins. Upon POR the terminal address and its parity are automatically read from address 30 on the I/ O bus. The value can be supplied in 2 ways: by enabling the output of an external terminal address buffer or by employing pull- up/ down resistors to define a default value for the 6 least significant bits of the I/ O data bus. Odd parity is used to define a valid terminal address; even parity will inhibit reception on both buses. After POR, the host can change the terminal address through software by writing to the TADR field with any desired value. In addition, this operation will enable reception. Providing Bit 2 of Configuration Register is set to "0".

The host can check the validity of the parity bit obtained from the I/ O bus by reading address 30; if the most significant bit in the lower byte equals 1, the parity is invalid.

If the TADR is not defined externally (by pull- down resistors or a buffer), there is no danger of a false response before host initialization because internal pull- up resistors on the I/ O bus guarantee an incorrect terminal address parity.

When BUSY= 1, 1553 message accesses to the RAM are inhibited, however the RT will respond with status as required by MIL- STD- 1553B. The mode commands "Transmit Status Word", "Transmit Last Command Word", "Reset Remote Terminal", "Transmitter Shutdown", "Override Transmitter Shutdown" and the reserved mode commands legalized by MIO (see the CONTROL register for details) are not affected by BUSY. In addition, all output pulses issued after valid command reception are inhibited when BUSY= 1 (except for the signal MDCDRST which is pulsed after receiving the mode command "Reset").

After POR(MRST), BUSY is set to "1"; this prevents the RT from using undefined pointers before the host has had a chance to initialize the POINTER TABLE. The default value for all other status bits is "0" and the TADR field is loaded with the hardwired address.

The BUSY Bit in the LAST STATUS REGISTER is cleared on receipt of the first command after a RESET, except if that command is TRANSMIT LAST STATUS or TRANSMIT LAST COMMAND mode command.

The BUSY Bit in the LAST STATUS REGISTER can be cleared by bit using BIT 5 in the RTC CONTROL REGISTER. See RTC CONTROL REGISTER for details.

4.2.4 INTERRUPT REQUEST Address: 3(Ubyte) W BC/ MT/ RT

The INTERRUPT REQUEST register holds 8 types of interrupt requests (see section on INTERRUPT CONTROL UNIT for details). Interrupt requests are active high and upon POR the register is cleared (see initialization section).

15	14	13	12	11	10	9	8
IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0

4.2.5 INTERRUPT MASK Address: 3 Lbyte R/ W BC/ MT/ RT

The INTERRUPT MASK register masks the corresponding interrupts. Upon POR, all interrupts are masked (see initialization section).

7	6	5	4	3	2	1	0
IMSK7	IMSK6	IMSK5	IMSK4	IMSK3	IMSK2	IMSK1	IMSK0

4.2.6 INTERRUPT VECTOR Address: 3(Ubyte) R BC/ MT/ RT
INTERRUPT VECTOR Address: 4(Lbyte) R/ W BC/ MT/ RT

The IVR is read only in the upper byte at address 3 and is read/ write in the lower byte at address 4. It contains interrupt header information which is popped off the FIFO.

ADDR3(4)	15(7)	14(6)	13(5)	12(4)	11(3)	9(2)	10(1)	8(0)
RT (BC/MT)	D4 (CNT4)	D3 (CNT3)	D2 (CNT2)	D1 (CNT1)	D0 (CNT0)	L2 L2	L1 L1	L0 L0

The Interrupt Vector register is loaded with LLL and CNT data from the fifo when it is popped. The fifo is popped by a hardware interrupt acknowledge or a read to address 8. This register in undefined at POR.

L(2: 0)
This is the interrupt priority determined by the message processor which is a function of the BC, MT and RT modes of operation.

D(4: 0)
In the RT mode the DDDDD field is inputted by the CPU. This is used as an offset for the interrupt vector. During a hardware interrupt acknowledge, this register is outputted on the upper and lower bytes of the CPU data bus. This output vector only occurs when the terminal is functioning as an RT.

CNT(4: 0)
In the BC and MESSAGE MONITOR modes the CNT field is the lower five bits of the number of the message in the frame which caused the interrupt.

4.2.7 CONFIGURATION REGISTER 2 Address: 4(Ubyte) W BC/ MT/ RT

This register is used for operational control of the part.

15	14	13	12	11	10	9	8
GO DEF FRAME	ABORT	STOP AT EOF	STOP AT EOM	CLR DISC	GO EOF CNTNU	GO NEXT	DO EOM CNTNU

NOTE: Bits 8 -10 are only used in the BC mode during a BUS JAM condition.

GO_ DEF_ FRAME Bits: 15 BC/ MT

When a "1" is written to this bit, the DEFAULT FRAME, defined by bit 12 in CONFIGURATION REGISTER 1, is made the active frame.

ABORT Bits: 14 BC/ MT

When a "1" is written to this bit, all BC and MT processing is terminated and the NHi- ET goes off- line. The BC or the MT must be re- started to again become active.

STOP END OF FRAME Bits: 13 BC/ MT

When a "1" is written to this bit, the BC or the MT will go off- line after the last message in the frame or block has been processed. The BC or the MT must be re- started to again become active.

STOP END OF MESSAGE Bits: 12 BC/ MT

When a "1" is written to this bit, the BC or the MT will go off- line after the current message in the frame or block has been processed. The BC or the MT must be re- started to again become active.

CLR DISC FLAG Bits: 11 RT

When a "1" is written to this bit, the 1760 DISCONNECT FLAG is cleared. This flag indicates that a store has been released and all the address bits and the parity bit on the hardwire address are "1's". The flag is read on the IPO_ DSC pin and bit 6 of the EXTERNAL TERMINAL ADDRESS REGISTER.

GO EOF & CONTINUE Bits: 10 BC

When a "1" is written to this bit after a BUS JAM condition has halted the BC operation, the BC will ignore all further messages in the current frame, proceed to the end of the frame and perform the programmed EOF operations.

GO NEXT MESSAGE Bits: 9 BC

When a "1" is written to this bit after a BUS JAM condition has halted the BC operation, the BC will abort the current message and NOT perform the programmed EOM operations; instead, the next message in the frame will be activated. If there are no more messages in the frame, the programmed EOF operations will be performed.

DO_ EOM & CONTINUE Bits: 8 BC

When a "1" is written to this bit after a BUS JAM condition has halted the BC operation, the BC will perform the programmed EOM operations, then the next message in the frame will be activated. If there are no more messages in the frame, the programmed EOF operations will be performed.

The contents of CONFIGURATION REGISTER 2 are not affected by these operations and all the bits are always read by the host as "0" after the indicated action has been completed.

4.2.8 AUXILIARY VECTOR REGISTER Address: 4(Ubyte) R BC/ MT/ RT

This register contains additional information related to the interrupt request. The data is popped from the FIFO and latched into the AVR during the interrupt acknowledge cycle or whenever the FIFO is popped by a host read instruction to address 8. Upon POR, this register is undefined.

MODE	15	14	13	12	11	9	10	8
RTU	EMP	BUS	T/R	SADR4 MODE4	SADR3 MODE3	SADR2 MODE2	SADR1 MODE1	SADR0 MODE0
BCU	EMP	BUS	FRAME	CNT9	CNT8	CNT7	CNT6	CNT5
MTU	EMP	BUS	FRAME	CNT9	CNT8	CNT7	CNT6	CNT5

EMP Bits: 15 BC/MT/RT

1= Fifo empty. Ignore data.
0= Fifo data valid. Use data.

BUS Bits: 14 BC/MT/RT

0= Indicates that the message was on bus A
1= Indicates that the message was on bus B.

T/ R Bits: 13 RT

0= Indicates a receive message.
1= Indicates a transmit message.

SADR / MODE Bits: (12-8) RT

This field defines the sub- address or mode code.
Note: the interrupt level distinguishes between regular transmit/ receive commands and mode commands.

CNT Bits: (12-8) BC/MT

The CNT field is the upper five bits of the number of the message in the frame which caused the interrupt.

FRAME Bits: 13 BC/MT

0= Indicates message in frame A .
1= Indicates message in frame B.

4.2.9 REAL- TIME CLOCK

RTC HIGH WORD Address: 5 R BC/MT/RT
RTC LOW WORD Address: 6 R BC/MT/RT

The RTC is a 32 bit up- counter which can be used for time- tagging in the BC, MT and RT modes. If the time- tagging option is in effect, the RTC is sampled and stored in 2 words in the data table The most significant word is stored first.

When messages are time- tagged in the RT mode, the host should not write data to the first 2 locations following the data table tag word since they will be overwritten with the value of the message time tag.

In the RT mode, the RTC can be reset by the mode command "Synchronize Without Data" and the least significant 16 bits can be updated by "Synchronize With Data". The full 32 bits can be updated using the first two data words in a receive command. See RTC CONTROL REGISTER for details.

The RTC can be read and reset by the host at any time. Since the RTC consists of 32 bits, at least 2 memory cycles are required to read all of its value. As a result, a carry- out from the lower word can occur between the read cycles. A mechanism is therefore provided to solve this potential difficulty.

If the host reads the RTC as two 16 bit words, *LOCK should be initialized to 1 in the RTC CONTROL register. In this case, when the host reads the upper word, all 32 bits are latched into the host output register. The value in the output register remains unchanged until the host finishes reading the lower word of the RTC.

If the host reads the RTC in bytes, *LOCK should be initialized to 0. In this case, when the host reads any of the bytes of the RTC, all 32 bits are latched into the host output register and its value remains unchanged until updating is re- enabled by reading the RTC CONTROL register. The RTC resolution can be programmed equal to 1, 2, 4, 8, 16, 32, or 64 microseconds.

4.2.10 RTC CONTROL REGISTER Address: 7 R/ W BC/ MT/ RT

The RTC CONTROL register controls the RTC as well as having other functions.

15	14	13	12	11	10	9	8
RTC RESET	RESET LAST	RES2	SYNUPD	*LOCK	SYNRST	RES1	RES0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
M1760	BUSY OPT	RESET BUSY	PRESET 4	PRESET 3	PRESET 2	PRESET 1	PRESET 0

RTC RESET **Bits: 15** **BC/ MT/ RT**

When a "1" is written to RTC RESET, a reset pulse is issued to the RTC. The contents of the register are not affected by this operation and RTC RESET is always read by the host as "0".

RESET LAST **Bits: 14** **BC/ MT/ RT**

When a "1" is written to RESET LAST, all the bits in the LAST STATUS REGISTER except the ADDRESS field and the BUSY bit are set to a "0". The contents of the register are not affected by this operation and RESET LAST is always read by the host as "0".

SYNUPD **Bits: 12** **RT**

1= Specifies that the lower 16 bits of the RTC will be updated whenever a valid mode command "Synchronize With Data" is received by the ET.

***LOCK** **Bits: 11** **BC/ MT/ RT**

0 = Enables updating of the host output register after the RTC CONTROL register is read (this feature is needed to support byte wide read cycles).

1 = Enables updating of the host output register after the lower RTC word is read.

SYNRST **Bits: 10** **RT**

1= Specifies that the RTC shall be reset whenever a valid mode command "Synchronize Without Data" is received by the ET.

RES **Bits: 13, 9, 8** **BC/ MT/ RT**

This field defines the resolution of the RTC in microseconds as follows:

RESOLUTION(us)	13	9	8
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
64	1	1	0
OFF/EXT	1	1	1

Note: Some NHi- ET device types have an external TIME TAG CLOCK input.

M1760 **Bits: 7** **RT**

1= Specifies that the RT shall comply with MIL- STD- 1760A. This mode of operation has two consequences: first, the mode command "Synchronize With Data" updates the lower 16 bits of the RTC only if the least significant data bit is "0" and second, the IPO_ DSC pin serves as a store disconnect signal rather than an interrupt priority output.

0= Specifies that the RT shall comply with MIL-STD-1553B.

BUSY _OPT **Bits: 6** **RT**

0= MRST, Software Reset and MODE CODE_ 08 RESET will set the BUSY bit in the LAST STATUS REGISTER and the BASIC STATUS REGISTER to a "1".

1= Only MRST will set the BUSY bit in the LAST STATUS REGISTER and the BASIC STATUS REGISTER to a "1".

RESET BUSY **Bits: 5** **RT**

When a "1" is written to RESET BUSY, the BUSY bit in the LAST STATUS REGISTER is set to a "0". The contents of the register are not affected by this operation and RESET BUSY is always read by the host as "0".

PRESET **Bits: (4: 0)** **RT**

These bits provide a method to perform a double word(32 bit) preset to the RTC. When this bit field is set to any number from 1 to 30(bit 0 = LSB), the first two words of a receive message whose subaddress is equal to this value will be used to preset the internal RTC. The most significant word is received first. If this field is equal to a "0" or "31", the RTC will not be preset. All bits in this register are cleared during initialization of the ET.

4.2.11 FIFO READ **Address: 8** **R** **BC/ MT/ RT**

This address is used to read the contents of the interrupt FIFO. Reading this address pops the FIFO, updates the IVR and the AVR; then outputs the AVR(upper byte) and IVR(lower byte).

4.2.12 FIFO RESET **Address: 8** **W** **BC/ MT/ RT**

Writing any value to this address empties the FIFO.

4.2.13 LAST COMMAND REGISTER **Address: 11** **R** **RT**

This register holds the last command word as defined by the MIL-BUS. The contents are not defined after initialization of the RT.

4.2.14 LAST STATUS REGISTER Address: 12 R RT

This register holds the Status Word assosciated with the last message. After initialization of the RT, the BUSY bit= 1, the TADR field contains the hardwire address, and all other bits are set to 0. See RTC CONTROL REGISTER for special options.

4.2.15 RESET REMOTE TERMINAL Address: 15 W BC/ MT/ RT

Writing a word to address 15 resets the RT and causes it to perform its initialization (see initialization section).

4.2.16 ENCODER STATUS Address: 18 R BC/ RT

This register contains flags indicating the status of the encoder. These flags are intended to facilitate transmission of messages in loop- back mode during self- test.

15	7	0
TXREQ_L	EOTX_L	FAILSAFE_L

TXREQ_ L Bits: 15 RT

0= Indicates that the encoder is ready to accept the next word for transmission. This bit should equal "0" before loading the Encoder Data register with the next word. In order to transmit contiguous words, the next word should be loaded within 18 microseconds after *TXREQ transitions to "0".

EOTX_ L Bits: 7 RT

0= Indicates that the encoder has completed transmission and that there are no pending requests.

FAILSAFE_ L Bits: 0 BC/ RT

0= FAILSAFE TIME OUT has occurred. This bit will be set to a "1" when a new message is received or during a reset.

4.2.17 CONDITION REGISTER Address: 19 R BC/ MT/ RT

This register contains information about the command being processed by the NHi- ET. and the operational condition.of the NHi-ET.

15	14	13	12	11	10	9	8
X	X	AXEN	BXEN	TFE	X	MDCD_L	X
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
X	CUFRM BUSY	EOF B	EOF A	CUR FRM	CUR BUS	BUSJAM B	BUSJAM A

AXEN Bits: 13 BC/ RT

1= Indicates that transmitter A is enabled. This bit is set to a "1": at POWER UP, if the NHi- ET is RESET, after receipt of a "Reset" mode code, or after receipt of an OVERRIDE TRANSMITTER SHUTDOWN mode code on the B bus.

0= Indicates that transmitter A is inhibited. This bit is set to a "0" after receipt of TRANSMITTER SHUTDOWN mode code on the B bus.

BXEN Bits: 12 BC/ RT

1= Indicates that transmitter B is enabled. This bit is set to a "1": at POWER UP, if the NHi- ET is RESET, after receipt of a "Reset" mode code, or after receipt of an OVERRIDE TRANSMITTER SHUTDOWN mode code on the A bus.

0= Indicates that transmitter B is inhibited. This bit is set to a "0" after receipt of a TRANSMITTER SHUTDOWN mode code on the A bus.

TFE **Bits: 11** **RT**

1= Indicates that the TERMINAL FLAG bit in the status word can be set to a "1" . This can be done in the BASIC STATUS REGISTER or by the TERMINAL FLAG pin on the NHi-ET. This bit is set to a "1" at POWER UP, if the NHi- ET is RESET, after receipt of a RESET MODE CODE, or after receipt of an OVERRIDE INHIBIT TERMINAL FLAG mode code.

0= Indicates that the TERMINAL FLAG bit in the status word CANNOT be set to a "1". This bit is set to a "0" after receipt of an INHIBIT TERMINAL FLAG mode code.

MDCD_ L **Bits: 9** **RT**

1= Indicates that the last command received was NOT a mode code.

0= This bit is set to a "0" when a mode code is received.

CUFRM BUSY **Bits: 6** **BC/ MT**

1= The current frame of data block is busy. It is active and could be receiving or transmitting data.

EOF B **Bits: 5** **BC/ MT**

1= Frame "B" or data block "B" has finished processing data and in now inactive.

EOF A **Bits: 4** **BC/ MT**

1= Frame "A" or data block "A" has finished processing data and in now inactive.

CUR FRM **Bits: 3** **BC/ MT**

0= Frame "A" or block "A" is the current active frame of block.

1= Frame "B" or block "B" is the current active frame of block.

CUR_ BUS **Bits: 2** **BC/ MT**

0= Bus "A" is the current bus.

1= Bus "B". is the current bus.

BUSJAM B **Bits: 1** **BC**

1 = Bus "B" has been jammed by continuous transmission from an RT. This condition is indicated when an RT transmits more extra words then the value set in CONFIGURATION REGISTER 3 (See CONFIG REG 3 for details).

BUSJAM A **Bits: 0** **BC**

1 = Bus "A" has been jammed by continuous transmission from an RT. This condition is indicated when an RT transmits more extra words then the value set in CONFIGURATION REGISTER 3 (See CONFIG REG 3 for details).

4.2.18 ENCODER DATA REGISTER Address: 23 R/W RT

This register contains data to be transmitted when performing a loop back test.

4.2.19 ENCODER DATA TRANSMIT RQST Address: 24 W RT

Writing (any value) to this address causes the contents of the ENCODER DATA REGISTER to be sent as a data word. This instruction together with the ENCODER COMMAND TRANSMIT REQUEST can be used to loop- back entire messages for self- test purposes. The received data can be read from the data table associated with the command.

4.2.20 ENCODER COMMAND TRANSMIT REQUEST Address: 25 W RT

Writing (any value) to this address causes the contents of the ENCODER DATA REGISTER to be sent as a command word. This instruction is useful for sending commands to the decoder while in loop- back mode. The command can then be read from the LAST COMMAND register.

4.2.21 EXTERNAL TERMINAL ADDRESS REGISTER Address: 30 R RT

This register contains information about the hardwire terminal address.

7	6	5	4	3	2	1	0
INVALP	DISCON	TADRP	TADR4	TADR3	TADR2	TADR1	TADR0

The terminal address may be hardwired using I/ O DAT(5: 0). External pull- down resistors of 4.7K are used to set a low, 64K internal pull- ups set a high. I/ O DAT5 is wired for odd parity in the address. The hardwire terminal address and its parity can be obtained by reading I/ O address 30. This address is unique since a read operation activates both the I/ O bus command strobe and the I/ O bus read signal (i. e., CMDS= 1 and *I/ O RD= 0). As a result, a buffer containing the terminal address can be selected without decoding address lines.
If an external buffer is not desired, pull- up/ down resistors on the I/ O data bus can be used instead (see BASIC STATUS register for details). The protocol chip also calculates the terminal address's parity and compares it to the value obtained from the I/ O bus.

INVALP Bits: 7

1= Specifies that the terminal address which was read automatically by the protocol chip following reset (from I/ O address 30) had invalid parity.

DISCON Bits: 6

0= Specifies that the store is disconnected because a terminal address of 31 was detected on the I/ O bus for at least 800 nanoseconds.

1= Specifies that the store is connected.

This bit indicates the "disconnected store" condition defined by MIL- STD- 1760A, provided that the store contains the pull- down resistors used for defining the terminal address (see BASIC STATUS register for details). After the store is disconnected, the standby state of all I/ O lines will be high and will therefore define an illegal terminal address of 31.

TADRP Bits: 5

TADRP equals the value of the terminal address parity read from I/ O address 30.

TADR Bits: (4:0)

TADR equals the value of the terminal address read from I/ O address 30.

4.2.22 COMMAND OUTPUT PINS Address: 30 W RT

Writing a word to the COMMAND OUTPUT PINS (address 30 in the I/ O space) can be used to simulate the option which outputs 5 bits onto the I/ O bus following valid command reception (see CMDO bit in the CONTROL register for details). This address is unique since a write operation activates both the I/ O bus, COMMAND STROBE and the I/ O bus write signal (i. e., CMDS= 1 and *I/ O WR= 0). As a result, the bits can be latched without decoding address lines.

4.2.23 I/ O TAG WORD REGISTER Address: 31 R/ W RT

When a data table is mapped to address 32 in the I/ O space, its tag word is contained in this register. This tag word can be used, for example, to specify an output pulse whenever the data table is accessed. All other I/ O space data tables are without internal tag words and have no pulses associated with them.

4.2.24 CONFIGURATION REGISTER 1 Address: 9 R/ W BC/ MT/ RT

This register is used to configure the functionality of the part.

15	14	13	12	11	10	9	8
MONITOR TYPE	GLOBAL BUS_SE1L	GLOBAL BUS_SEL0	FRAME BLOCK	START BCU_MTU	3818 MODE	FUNCTION SELECT1	FUNCTION SELECT0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	0	INHIBIT DBCA	GLOBAL DBS	LOCAL DBS	INHIBIT SOFTADR	CONVERT BUSY BIT	SEP BCST TABLES

Note: Reserved Bits 7- 6 must be set to "0".

MONITOR TYPE **Bits: 15** **MT**

0 = Word Monitor.
1 = Message Monitor .

GLOBAL BUS_ SEL **Bits: 14, 13** **BC**

These bits determine the Bus select options.

GLOBAL BUS	COMMENTS	14	13
DEFAULT	USE BC CONTROL WORD BUS	0	0
FORCE BUS “A”	FORCE ALL MESSAGES TO BUS “A”	0	1
FORCE BUS “B”	FORCE ALL MESSAGES TO BUS “B”	1	0
FORCE ALT BUS	USE OPPOSITE BUS OF BC CONTROL WORD	1	1

FRAME/ BLOCK **Bits: 12** **BC/ MT**

0 = Default Frame /Block is "A".
1 = Default Frame/Block is "B".

START_ BC_ MT **Bits: 11** **BC/ MT**

1 = Start Bus Controller or Monitor.

3818_ STATUS **Bits: 10** **RT**

0 = Status response and protocol operation as defined in Mil- Std- 1553B.
1 = Status response and protocol operation as defined in MDC A3818 and Mil- Std- 1553A.

FUNCTION SELECT **Bits: 9, 8** **BC/ MT/ RT**

OPERATIONAL MODE	9	8
REMOTE TERMINAL	0	0
BUS CONTROLLER	0	1
MONITOR	1	0
MONITOR & REMOTE TERMINAL	1	1

INHIBIT DBCA **Bits: 5** **RT**

0 = DBCA bit in Status Word is set upon receipt of a valid DBCA Mode Code.
1 = Prevents DBCA Bit in Status Word from being set upon receipt of a valid DBCA Mode Code.

GLOBAL_ DYNAMIC_ BUS_ SELECTION **Bits: 4** **BC**

0 = Message bus unchanged after successful Glocal Retry.
1 = Automatically switch message to alternate bus in BC Control Word after successful retry on alternate bus due to a Global Retry option.

LOCAL_ DYNAMIC_ BUS_ SELECTION **Bits: 3** **BC**

0 = Message bus unchanged after successful Local Retry.
1 = Automatically switch message to alternate bus in BC Control Word after successful retry on alternate bus due to Local Retry option.

INHIBIT SOFT_ ADR **Bits: 2** **RT**

0 = Bits (15: 11) of Basic Status Register set the RT Address when a Write Operation to that register is performed. The Hard Wired Address sets the RT Address at RESET.
1 = Prevents software change of RT Address when writing to the Basic Status Register. Bits (15: 11) of Basic Status Register are “Don’t Care”. Only the Hard Wired Address sets the RT Address at RESET.

CONVERT BUSY BIT **Bits: 1** **RT**

0 = BUSY Bit is compliant with Mil- Std- 1553B.
1 = Converts BUSY Bit to Non- 1553B operation. BUSY Bit becomes a standard bit with no special functionality. BUSY Bit is not set during software reset or MODE CODE_ 08 RESET.

SEP_ BCST_ TABLES **Bits: 0** **RT**

0 = Broadcast messages use the same pointers as receive message:. therefore, receive and broadcast messages are stored in the same data tables. The BCST bit in the tag word is used to differentiate between the two message types.
1 = An additional 30 pointers are activated which puts receive and broadcast messages in separate data tables.

4.2.25 FRAME "A" POINTER **Address: 13** **R/ W** **BC/ MT**
BLOCK "A" START

This register contains the 16 bit FRAME "A" POINTER. This is the address of the active message list to be used by the BC or the MESSAGE MONITOR for FRAME "A".
In the WORD MONITOR, this register contains the 16 bit start address of BLOCK "A".

4.2.26 FRAME "A" LENGTH **Address: 14** **R/ W** **BC/ MT**
BLOCK "A" END

In the BC and MESSAGE MONITOR modes, this register specifies the number of messages in FRAME "A" message list and several END- OF- FRAME options.
In the WORD MONITOR, this register contains the 16 bit end address of BLOCK "A".

15	14	13	12	11	10	9	8
RSVD	END OF A INT	STAT SET STOP A	FRAME A ERR STOP	FRAME A END OPT1	FRAME A END OPT0	FRAME A LEN9	FRAME A LEN8
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRAME A LEN7	FRAME A LEN6	FRAME A LEN5	FRAME A LEN4	FRAME A LEN3	FRAME A LEN2	FRAME A LEN1	FRAME A LEN0

Note: In the Message Monitor mode Bits 12 and 13 are reserved and always read 0.

END OF A INT **Bits: 14** **BC/ MT**

1 = The BC or the Message Monitor will interrupt when End of FRAME "A" is reached.
0 = End of Frame **WILL NOT** cause an interrupt.

STAT SET STOP A **Bits: 13** **BC**

1 = If any BC message in the frame causes a status bit set condition, then BC will stop at the end of the current frame and go off line.

FRAME A ERR STOP **Bits: 12** **BC**

1 = If any BC message in the frame causes an error condition, then BC will stop at the end of the current frame and go off line.

FRAME A END OPT **Bits: 11, 10** **BC/ MT**

These two bits determine a course of action at the end of FRAME "A" in the Message Monitor.

FRAME A END OPTIONS	11	10
STOP AT END OF FRAME A	0	0
REPEAT FRAME A	0	1
GOTO FRAME B	1	0
STOP AT END OF FRAME A	1	1

FRAME A LENGTH **Bits: 9: 0** **BC/ MT**

These ten bits determine the number of messages that the frame will contain, up to a maximum of 1023.

4.2.27 FRAME "B" POINTER **Address: 16** **R/ W** **BC/ MT**
BLOCK "B" START

This register contains the 16 bit FRAME "B" POINTER. This is the address of the active message list to be used by the BC or the MESSAGE MONITOR for FRAME "B".
In the WORD MONITOR, this register contains the 16 bit start address of BLOCK "B".

4.2.28 FRAME "B" LENGTH **Address: 17** **R/ W** **BC/ MT**
BLOCK "B" END

In the BC and MESSAGE MONITOR modes, this regiater specifies the number of messages in FRAME "B" message list and several END- OF- FRAME options.
In the WORD MONITOR, this register contains the 16 bit end address of BLOCK "B".

15	14	13	12	11	10	9	8
RSVD	END OF B INT	STAT SET STOP B	FRAME B ERR STOP	FRAME B END OPT1	FRAME B END OPT0	FRAME B LEN9	FRAME B LEN8
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FRAME B LEN7	FRAME B LEN6	FRAME B LEN5	FRAME B LEN4	FRAME B LEN3	FRAME B LEN2	FRAME B LEN1	FRAME B LEN0

Note: In the Message Monitor mode Bits 12 and 13 are reserved and always read 0.

END OF B INT **Bits: 14** **BC/ MT**

1 = The BC or the Message Monitor will interrupt when End of FAME "B" is reached.
0 = End of Frame **WILL NOT** cause an interrupt.

STAT SET STOP B **Bits: 13** **BC**

1 = If any BC message in the frame causes a status bit set condition, then BC will stop at the end of the current frame and go off line.

FRAME B ERR STOP **Bits: 12** **BC**

1 = If any BC message in the frame causes an error condition, then BC will stop at the end of the current frame and go off line.

FRAME B END OPT **Bits: 11, 10** **BC/ MT**

These two bits determine a course of action at the end of FRAME "B" in the Message monitor.

FRAME B END OPTIONS	11	10
STOP AT END OF FRAME B	0	0
REPEAT FRAME B	0	1
GOTO FRAME A	1	0
STOP AT END OF FRAME B	1	1

FRAME B LENGTH **Bits: 9: 0** **BC/ MT**

These ten bits determine the number of messages that the frame will contain, up to a maximum of 1023.

4.2.29 BC_ FRAME_ GAP **Address: 20** **R/ W** **BC/ MT**
WORD_ MONITOR_ EOF_ OPTIONS

In the BC mode, this 16 bit register specifies the END_ OF_ FRAME_ DELAY before starting the next frame. The delay resolution is 64 us.

In the WORD MONITOR, this register specifies the END- OF- FRAME options.

WORD MONITOR

15	14	13	12	11	10	9	8
0	0	0	0	0	0	0	0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
END OF B INT	RSVD	BLOCK B END OPT1	BLOCK B END OPT0	END OF A INT	RSVD	BLOCK A END OPT1	BLOCK A END OPT0

Note: Bits 15- 8 are reserved in the monitor mode and should be set to "0".

END OF B INT **Bits: 7** **MT**

1 = The Word Monitor will cause an interrupt when End of Block "B" is reached.
0 = End of Block **WILL NOT** cause an interrupt.

BLOCK B END_ OPT **Bits: 5, 4** **MT**

These two bits determine a course of action at the end of BLOCK "B" in the Word Monitor.

BLOCK B END OPTIONS	5	4
STOP AT END OF BLOCK B	0	0
REPEAT BLOCK B	0	1
GOTO BLOCK A	1	0
STOP AT END OF BLOCK B	1	1

ENDOF A INT **Bits: 3** **MT**

1 = The Word Monitor will cause an interrupt when End of Block "A" is reached.
0 = End of Block **WILL NOT** cause an interrupt.

BLOCK A END_ OPT **Bits: 1, 0** **MT**

These two bits determine a course of action at the end of BLOCK "A" in the Word Monitor.

BLOCK A END OPTIONS	1	0
STOP AT END OF BLOCK A	0	0
REPEAT BLOCK A	0	1
GOTO BLOCK B	1	0
STOP AT END OF BLOCK A	1	1

4.2.30 CONFIGURATION REGISTER 3 Address: 21 R/ W BC/ MT

This register is used to set global parameters for the BC and the MT.

15	14	13	12	11	10	9	8
RSVD	WORD MT NTTGDAT	WORD MT NTAG	WORD MT NTTAG	MSG MT NTAG	MSG MT NTTAG	GLOBAL RETRY1	GLOBAL RETRY0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
STAT SET RETRY	ADR LAT INHIBIT	BCST MSK BCST XOR	BUSJAM4	BUSJAM 3	BUSJAM 2	BUSJAM 1	BUSJAM 0

WORD MT NTTGDAT Bits: 14 MT

0 = A 32 bit time tag is stored with data words and command/ status words.
1 = No time tag on data words. Only command/Status words are time tagged.

WORD MT NTAG Bits: 13 MT

0 = A Tag word is stored with Data and Command/Status words.
1 = No tag word.

WORD MT NTTAG Bits: 12 MT

0 = Word Monitor time tag is enabled. Bit 14 determines the time tag format.
1 = No time tagging. Word Monitor time tag is disabled.

MSG MT NTAG Bits: 11 MT

0 = Tag word is stored with Command/Status words.
1 = No tag word.

MSG_ MT NTTAG Bits: 10 MT

0 = Message Monitor time tag is enabled. Command/Status words are time tagged.
1 = Message Monitor time tag is disabled.

GLOBAL RETRY Bits: 9, 8 BC

These bits define a global default retry scenario. If the BC control word defines no retry as the option for a message, then the global retry is enabled. If the global retry is defined as no retry, then their will not be a retry for the message.

GLOBAL RETRY OPTIONS	9	8
NO RETRY	0	0
RETRY ACTIVE BUS	0	1
RETRY ALTERNATE BUS	1	0
RETRY ALTERNATE BUS, THEN ACTIVE BUS	1	1

STAT SET RETRY Bits: 7 BC

This bit determines if a retry will be executed when a status word invokes a status set condition.
0 = No retry on status set.
1 = Retry if a status bit is set.

ADR LAT INHIBIT Bits: 6 RT

This bit determines whether or not the CPU address will be automatically latched by the HCS_ L.
0 = CPU address is automatically latched within 200ns after the falling edge of HCS_ L.
1 = CPU address is manually stored in a transparent when ADR_ LAT_ L input signal is a “1”.
Note:: This option is not available on all parts.

BCST_ MSK/ BCST_ XOR **Bits: 5** **BC**

This bit determines how the Broadcast bit in the returned status word will be treated.

0 = If the BCST bit in the status word **DOES NOT** equal bit 4 of ths BC CONTROL WORD, then the STAT_ SET bit in the BC CONTROL WORD will be set.

1 = If bit 4 of the BC CONTROL WORD is a "0", then the BCST bit in the status word is **DON'T CARE.** If bit 4 of the BC CONTROL WORD IS a "1" and the BCST bit in the status word is a "1", then the STAT_ SET bit in the BC CONTROL WORD will be set.

BUS JAM **Bits: 4: 0** **BC**

These bits determine the number of excess words that will be accepted from an RT without declaring that the bus has been jammed by an RT that is transmitting continuously. The range is from 0 to 31 words. The msb is bit 4. When a **BUS_ JAM** is detected, the BC issues a non-maskable interrupt by setting the PLSCMD_ BSJM_ H output pin to a “1” and Halts.

The CPU can cause the BC to continue by writing to one of the bits in CONFIGURATION REGISTER 2 after taking corrective action (ie: Globally switch all messages to good bus using Config Reg 1).

4.2.31 MT ADDRESS FILTER (15:0) Address: 22 R/ W MT

This register determines which RT addresses, from 0 to 15 will be monitored in the MESSAGE MONITOR mode.

0 = Accept RT address, store data.

1 = Ignore RT address, **NO** data stored.

15	14	13	12	11	10	9	8
MASK 15	MASK 14	MASK 13	MASK 12	MASK 11	MASK 10	MASK 09	MASK 08
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK 07	MASK 06	MASK 05	MASK 04	MASK 03	MASK 02	MASK 01	MASK 00

4.2.32 MT ADDRESS FILTER (31:16) Address: 26 R/ W MT

This register determines which RT addresses, from 16 to 31 will be monitored in the MESSAGE MONITOR mode.

0 = Accept RT address, store data.

1 = Ignore RT address, **NO** data stored.

15	14	13	12	11	10	9	8
MASK 31	MASK 30	MASK 29	MASK 28	MASK 27	MASK 26	MASK 25	MASK 24
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK 23	MASK 22	MASK 21	MASK 20	MASK 19	MASK 18	MASK 17	MASK 16

4.2.33 BLOCK "A" LAST ADDRESS Address: 27 R MT

This register contains the address of the last word in BLOCK "A" for the WORD MONITOR. The last address is calculated by the protocol chip. It is not necessarily equal to the BLOCK "A" end address in register 14. This is because any one of up to four words associated with the last in-coming word in the block could be stored in register 14. In order to keep all the words together, they are stored contiguously and the last address in BLOCK "A" is stored in this register; therefore, four addresses must always be reserved after the address in register 14 to accommodate this situation.

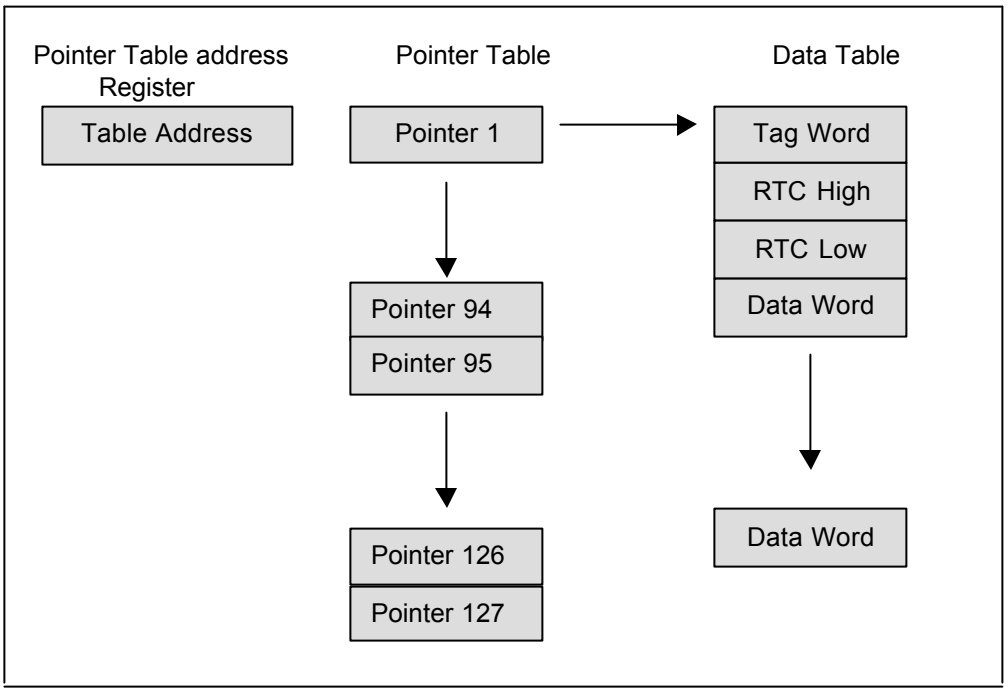
4.2.34 BLOCK "B" LAST ADDRESS Address: 28 R MT

This register contains the address of the last word in BLOCK "B" for the WORD MONITOR. The last address is calculated by the protocol chip. It is not necessarily equal to the BLOCK "B" end address in register 17. This is because any one of up to four words associated with the last in- coming word in the block could be stored in register 17. In order to keep all the words together, they are stored contiguously and the last address in BLOCK "B" is stored in this register; therefore, four addresses must always be reserved after the address in register 17 to accommodate this situation.

4.3.0 RT DATA TABLES

The data words associated with transmit/ receive/ broadcast messages and mode commands are stored in data tables. The data table addresses are stored in a POINTER TABLE located in the RAM. The data tables themselves can be individually allocated to either the RAM or the I/ O space. Data tables mapped to the I/ O space can be used for discrete I/ O without requiring host intervention. The mapping scheme is illustrated in the following diagram:

REMOTE TERMINAL MEMORY ORGANIZATION



The T/R bit subaddress and word count fields in the Command word are used to index into the Pointer table as defined below:

INDEX	T/R	SUBADDRESS	MODE CODE	COMMAND TYPE
0		NOT USED		
1 - 30	0	1 – 30		RECEIVE/BROADCAST
31	0	31 (Note 2)		
32		NOT USED		
33 - 62	1	1 – 30		TRANSMIT
63	1	31 (Note 2)		TRANSMIT
64 - 95	X	0, 31(Note 2)	0 - 31	MODE CODE
96		NOT USED		
97 - 126	0	1 – 30		BROADCAST
127	0	31 (Note 2)		BROADCAST

Note 1: Separate Broadcast Pointers are activated when BIT "0" of CONFIGURATION REGISTER 1 is a "1" (See this register for details).

Note 2: When 3818A/ 1553A protocol option is enabled, subaddress 31 is an extend subaddress, **Not a Mode Code Flag.**

4.3.1 MESSAGE ILLEGALITY

Commands are illegalized by setting the address field of the corresponding data table pointer to 0. When the protocol chip receives an illegal command, it responds with ME= 1 in the status; in addition, data transmission and storage are suppressed. All undefined mode commands are ignored.

4.3.2 REMOTE TERMINAL DATA TABLE TAG WORD

The data table's first word can be defined to be either a data word or a TAG WORD (see the NTAG field in the CONTROL register) which defines the table's status and associated options. The TAG WORD has the following format:

15	14	13	12	11	10	9	8
UPDATE	SSFENA	BCST	X	PULSE3	PULSE2	PULSE1	PULSE0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LOCK	INVALID	OVRWRT	WCNT4	WCNT3	WCNT2	WCNT1	WCNT0

UPDATE **Bits: 15**

1= Indicates that the table was updated with data by the CPU or a bus message. The CPU should set this bit after writing to the table and reset the bit after reading the table.

SSFENA **Bits: 14**

1= Enables setting the subsystem flag in the status word whenever the RT transmits stale data or overwrites received data (i. e., whenever data is transmitted from a table with UPD= 0, or is stored into a table with UPD= 1).

BCST **Bits: 13**

1= Indicates that the table contains data from a valid broadcast message.

0= Indicates that the table contains data from a non- broadcast message.

PULSE(3: 0) **Bits: 11:8**

This field defines which pin should be pulsed at the end of a valid message which accesses the data table (see CMDO in the CONTROL register for details). The field is defined as follows:

PULSE FIELD VALUE	PULSED OUTPUT PIN
0	NO PULSE
1 – 8	I/O DATA(7..0)
14	PLSCMD PIN

LOCK **Bits: 7**

1= Indicates that the protocol chip is currently using the table for a message, either writing receive data or reading transmit data.

INVALID **Bits: 6**

1= Indicates that the table contains invalid data.

OVW **Bits: 5**

1= Indicates that data received from the Mil Bus caused the data to be overwritten before its previous contents were read by the host or that the host did not update the data since the last transmission (i. e., whenever data is transmitted from a table with UPD= 0, or is stored into a table with UPD= 1). This bit is similar to the subsystem flag returned to the Bus Controller when SSFENA= 1.

WCNT(4: 0) **Bits: 4-0**

This field contains the word count/ mode code in the command which referenced the data table.

4.3.3 DATA TABLE POINTER WORD

The Data Table Pointer Word has the following format:

15	14	13	12	11	10	9	8
INTREQ	RTCENA 1	ADDR 13	ADDR 12	ADDR 11	ADDR 10	ADDR 09	ADDR 08
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ADDR 07	ADDR 06	ADDR 05	ADDR 04	ADDR 03	ADDR 02	ADDR 01	RTCENA 0

INTREQ **Bits: 15**

1= Specifies that an interrupt shall be issued after the completion of a message which accesses the data table and a header pushed on to the Fifo,.

0= No interrupt issued; nothing pushed on to Fifo.

ADDR(13: 1) **Bits: 13:1**

Defines the location of the RTU data table which MUST be in the lower 8K word address space. The RTU data tables always begin on word boundaries. The least significant bit of the word address is bit 1 in the Pointer word. If the data table address field is set to 0, the command associated with the pointer is illegalized.
with the pointer is illegalized.

RTCENA **Bits: 14,0**

Real Time Clock Time Tag Message Oprions. The Time_ Tag_ Transmit option will cause the first two transmitted words of the associated Transmit Command to contain the time tag, MSW first.

TIME TAG MODE	14	0
NO TIME TAG ON MESSAGES	0	0
NO TIME TAG ON MESSAGES	0	1
TIME TAG MESSAGES; DON'T TRANSMIT TIME TAG WITH TRANSMIT COMMAND	1	0
TIME TAG MESSAGES; TRANSMIT TIME TAG WITH TRANSMIT COMMAND	1	1

Note: If the No Time Tag option is used, data words occupy the time tag positions.

4.3.4 RT DATA TABLE BUFFERING SCHEME

Since the host and the NHi_ ET can access data tables asynchronously, data integrity must be ensured by a suitable buffering scheme. The method employed by the ET assumes that there are two pointer tables; one specifies data tables accessed by the ET and the other tables accessed by the host. The host's pointer table can reside anywhere in its memory space since it is never accessed by the ET. Data buffers are switched by the host exchanging pointers as explained below.

4.3.4.1 RT RAM ACCESS

When the ET wants to read or write to a data table, it fetches the corresponding data table pointer from its pointer table. It then sets the LOCK bit in the data table's TAG WORD to 1 and proceeds with the update. At the completion of the update, the ET sets the LOCK bit to 0 and also sets the UPDATE bit in the TAG WORD to 1 if it wrote to the data table or 0 if it read the data table. If the condition of the UPDATE bit at the start of the ET access indicates that the host has not read from or written to the data table since the last ET access to that table, the ET sets the OVRWRT bit in the TAG WORD to 1 to tell the host stale data has been transmitted by the ET or data has been overwritten by the ET.

Since the ET may fetch a data table pointer while the host is in the process of exchanging the corresponding pointers, there is a possibility that the ET's pointer will point to the table used by the host. In order to avoid this potential conflict, the host should check the LOCK bit in its data table tag word AFTER exchanging the pointers but BEFORE reading the data. If LOCK= 1, the host should wait until the protocol chip sets LOCK= 0.

NOTE: The LOCK bit is ALWAYS set in the TAG WORD of the data table accessed by the ET, irrespective of when the pointers are exchanged by the host. This is guaranteed because the ET reads the data table's pointer and sets the LOCK bit in the TAG WORD using a read - read - modify - write sequence which cannot be interrupted by the host (i. e., read POINTER - read TAG WORD - modify LOCK bit - write back TAG WORD with LOCK bit modified).

4.3.4.2 HOST RAM ACCESS

RECEIVE DATA TABLE

When the host wants to read the data in a RECEIVE data table, it **FIRST EXCHANGES** the pointer in its pointer table with the corresponding pointer in the ET's table. Then, the host reads the LOCK bit in the TAG WORD. If the LOCK bit is 0, the host proceeds with its access. If, however, the LOCK bit is 1, this informs the host that the ET is accessing that data table. The host should then delay its access until the LOCK bit has been set to 0 by the ET. When the host finishes accessing the RECEIVE data table, it should clear the UPD bit in the data table's TAG WORD to 0. This will tell the ET the host has taken the data.

TRANSMIT DATA TABLE

When the host wants to write data to a TRANSMIT data table, the apparent method would be to load the table with data then exchange corresponding pointers. There is a subtle problem with this approach. If the host had, within a short period of time prior to this exchange previously loaded and exchanged these same pointers while the ET had been transmitting data from that data table , the LOCK bit could still be set in the table the the host was loading during the second sequence. This is possible because it can take up to 640us to transmit a message, the LOCK bit being set for the entire time. This could cause new data to be mixed with old data and transmitted. Avoiding this potential problem is quite simple.

When the host wants to access a TRANSMIT data table, it first reads the LOCK bit in the TAG WORD of the table belonging to the host. If the LOCK bit is 0, the host proceeds with its access and loads the data table . If, however, the LOCK bit is 1, this informs the host that the ET is still accessing that data table. The host should then delay its access until the LOCK bit has been set to 0 by the ET. When the host finishes updating its TRANSMIT data table, it should set the UPD bit in the data table's TAG WORD to 1 and then exchange corresponding pointers. This will ensure that updated data for transmission is made available to the ET as soon as possible and inform the ET that it will be transmitting fresh data.

Since the host can change its table of pointers at any time, the above mapping scheme can be used to achieve any desired depth of buffering by simply employing a "round- robin" of pointers.

4.3.4.3 READ- MODIFY- WRITE

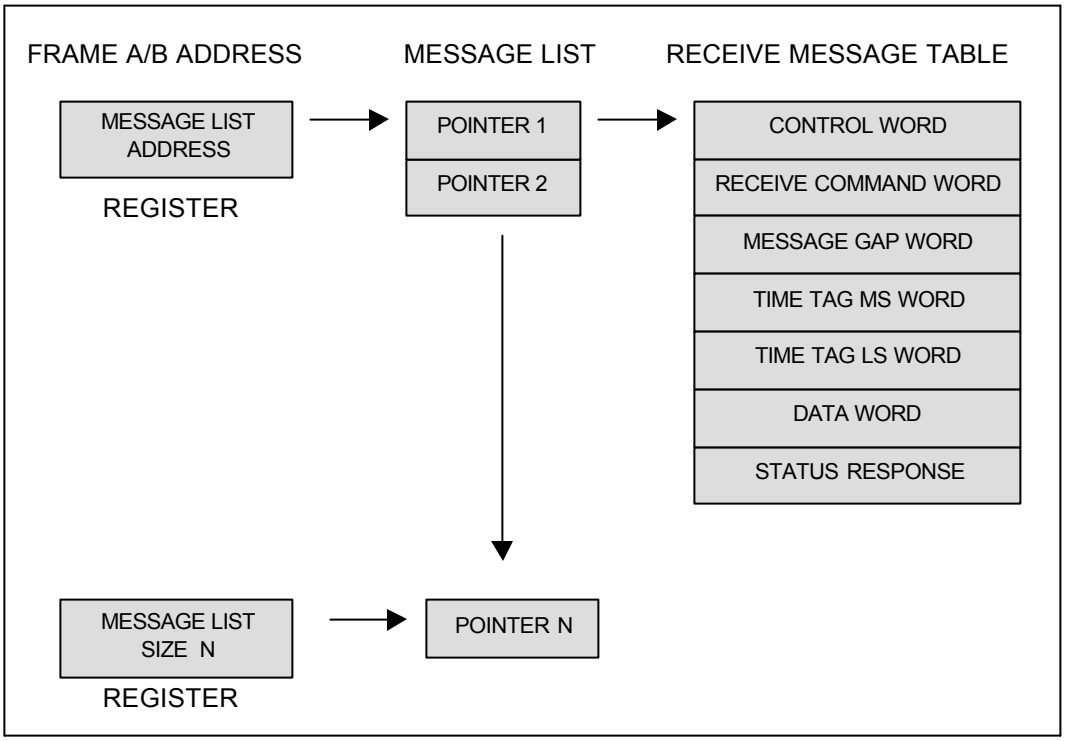
The host Read- Modify- Write cycle is used to support CPUs similar to the Motorola 680X0 where certain instructions (eg:, test and set) require two contiguous accesses to memory. Such accesses are unique in that the address remains active for both cycles.

4.4.0 BUS CONTROLLER MESSAGE LISTS AND DATA TABLES

The BC is organized and controlled by message lists. Each message list contains the addresses of data tables associated with the list. A message list can contain up to a maximum of 1023 16 bit addresses. The number of message lists and data tables is limited only by the size of the ram. The message list mapping scheme is illustrated in the following diagram.. FRAME A/ B POINTER and FRAME A/ B LENGTH registers have been discussed in a previous section. See details.

The CPU loads each message list with number of message table pointers determined by a given scenario requirement. The list is activated by placing its address in the FRAME A or FRAME B POINTER register and the number of pointers in the list in the corresponding FRAME LENGTH register.

BUS CONTROLLER MEMORY ORGANIZATION



4.4.1 BC CONTROL WORD

The BC CONTROL WORD contains information which is specific to the message in its data table. The word has certain bits which are controlled by the CPU and others are controlled by the ET. This dual functionality provides a more a friendly and uncomplicated user inter face. Bits(9: 0) are defined by the CPU and provide message setup detail, while Bits(15: 10) are controlled by the ET and report operation information.

15	14	13	12	11	10	9	8
EOM	RETRY	STATSET	ERROR	NORESP	SOM	LCLRTRY 1	LCLRTRY 0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUS	TFRSVINS	SRQ	BCST	BUSY3	SSF	MSGERR	RT-RT

EOM **Bits: 15**

1= **Set by the ET.** Message has been transmitted and completed.

RETRY **Bits: 14**

1= **Set by the ET.** A Retry has been attempted for this message. See LOCAL and GLOBAL retry setups for the scenario.

STATSET **Bits: 13**

1= **Set by the ET.** The Status word(s) returned by the RT had bits set that aren't "Don't Cares" or had the wrong RT address .

ERROR **Bits: 12**

1= **Set by the ET.** The message returned by the RT contained an error.

NORESP **Bits: 11**

1= **Set by the ET.** The RT DID NOT respond with a Status or data as expected.

SOM **Bits: 10**

1= **Set by the ET.** The message is currently active. The ET could be transmitting, waiting for a response or receiving.

LOCAL_ RETRY **Bits: 9,8**

Set by the CPU. Bits(9: 8) define a local specific retry scenario. If the BC control word defines no retry as the option for the message, then the global retry is enabled. If the global retry is defined as no retry, then their will not be a retry for the message.

LOCAL RETRY OPTIONS	9	8
NO RETRY	0	0
RETRY ACTIVE BUS	0	1
RETRY ALTERNATE BUS	1	0
RETRY ALTERNATE BUS, THEN ACTIVE BUS	1	1

BUS **Bits: 7**

0= **Set by CPU & ET.** Message will use BUS A.

1= **Set by CPU** & ET Message will use BUS B.

This bit defines the bus that the message will use. This bus setting, however, can be overridden by the settings in CONFIGURATION 1. See this register for specific details.

TFRSVINS **Bits: 6**

0= **Set by the CPU.** The value of the Terminal Flag bit, the Reserved bits, and the **I**nstrumentation bit are treated as "Don't Care", and will NOT cause the STATSET bit to be set no matter what their value in the returned Status word.

1= **Set by the CPU** The STATSET bit will be set if the Terminal Flag bit, or one of the Reservedbits or the instrumentation bit is set in the returned Status word.

SRQ **Bits: 5**

0= **Set by the CPU.** The value of the Service Request bit is treated as "Don't Care", and will NOT cause the STASET bit to be set no matter what its value in the returned Status word.

1= **Set by the CPU.** The STATSET bit will be set if the Service Request bit is set in the returned Status word.

BCST **Bits: 4**

Set by the CPU. The operation of this bit is determined by bit 5 of CONFIGURATION REGISTER 3. This bit will either be used as a mask or an xor flag for the Broadcast Received bit in the returned status word.

CONFIGURATION REG 3 Bit 05 = 0 (BCST bit is an Xor Flag):
If the Broadcast Received bit in the returned status word **DOES NOT** equal the BCST bit of the BC CONTROL WORD, then the STATSET bit in the BC CONTROL WORD will be set.

CONFIGURATION REG 3 Bit 05 = 1 (BCST bit is a Mask):

0= The value of the Broadcast Received bit is treated as "Don't Care", and will NOT cause the STATSET bit to be set no matter what its value in the returned Status word.

1= The STATSET bit will be set if the Broadcast Received bit is set in the returned Status word.

BUSY **Bits: 3**

0= **Set by the CPU.** The value of the Busy bit is treated as "Don't Care", and will NOT cause the STATSET bit to be set no matter what its value in the returned Status word.

1= **Set by the CPU.** The STATSET bit will be set if the Busy bit is set in the returned Status word.

SSF **Bits: 2**

0= **Set by the CPU.** The value of the Subsystem Flag bit is treated as "Don't Care", and will NOT cause the STATSET bit to be set no matter what its value in the returned Status word.

1= **Set by the CPU.** The STATSET bit will be set if the Subsystem Flag bit is set in the returned Status word.

MSGERR **Bits: 1**

0= **Set by the CPU.** The value of the Message Error bit is treated as "Don't Care", and will NOT cause the STATSET bit to be set no matter what its value in the returned Status word.

1= **Set by the CPU.** The STATSET bit will be set if the Message Error bit is set in the returned Status word.

RT_ RT **Bits: 0**

0= **Set by the CPU.** Message is **NOT** an RT to RT command.

1= **Set by the CPU.** Message **IS** an RT to RT command.

Note: Status bits treated as "Don't Cares" will NOT cause a STAT_SET.

4.4.2 BC COMMAND WORD

This is the 16 bit Command word that defines the message type as defined by the Mil-Bus.

4.4.3 BC MESSAGE GAP WORD

The BC MESSAGE GAP WORD contains additional information which is specific to the message in its data table. The word sets the gap time delay to the start of the next message in the list, end of message stop conditions, the message interrupt and a NOP feature. The resolution of the Gap counter is 1us and the maximum gap is 4ms. NOP's can be used to extend the inter-message gap beyond the 4ms maximum which is allowed by the Gap field.

15	14	13	12	11	10	9	8
NOP	EOM_INT	STATSTP	ERRSTP	GAP11	GAP10	GAP09	GAP08
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
GAP07	GAP06	GAP05	GAP04	GAP03	GAP02	GAP01	GAP00

NOP **Bits: 15**

0= Send message as defined.

1= Do NOT send message. Start Gap Timer and wait till time out, then GoTo next message.

Note: If the last message in a frame is a NOP(bit 15 = '1'), the E-O-F interrupt is disabled and any interrupts associated with that NOP message are disabled.

EOM_INT **Bits: 14**

0= **No** interrupt at End- of- Message.

1= **Interrupt** at End- of- Message.

STATSTP **Bits: 13**

0= Ignore a Status Set condition in the returned RT status word for the message.

1= Halt and go off- line if a Status Set condition is detected in the returned RT status word for the message.

ERRSTP **Bits: 12**

0= Ignore the Error condition in the message.

1= Halt and go off- line if an Error is detected in the message.

GAP **Bits: 11: 0**

This field sets the inter-message Gap Time delay. The resolution is 1us and the range of the delay is 0 to 4ms. The start of the next message in the list is delayed by this interval.

4.4.4 BC TIME TAG MS WORD

This word contains the upper 16 bits of the 32 bit Time Tag.

4.4.5 BC TIME TAG LS WORD

This word contains the lower 16 bits of the 32 bit Time Tag.

4.4.6 BC DATA WORD

This space contains from 1 to 32 data words which are associated with the message.

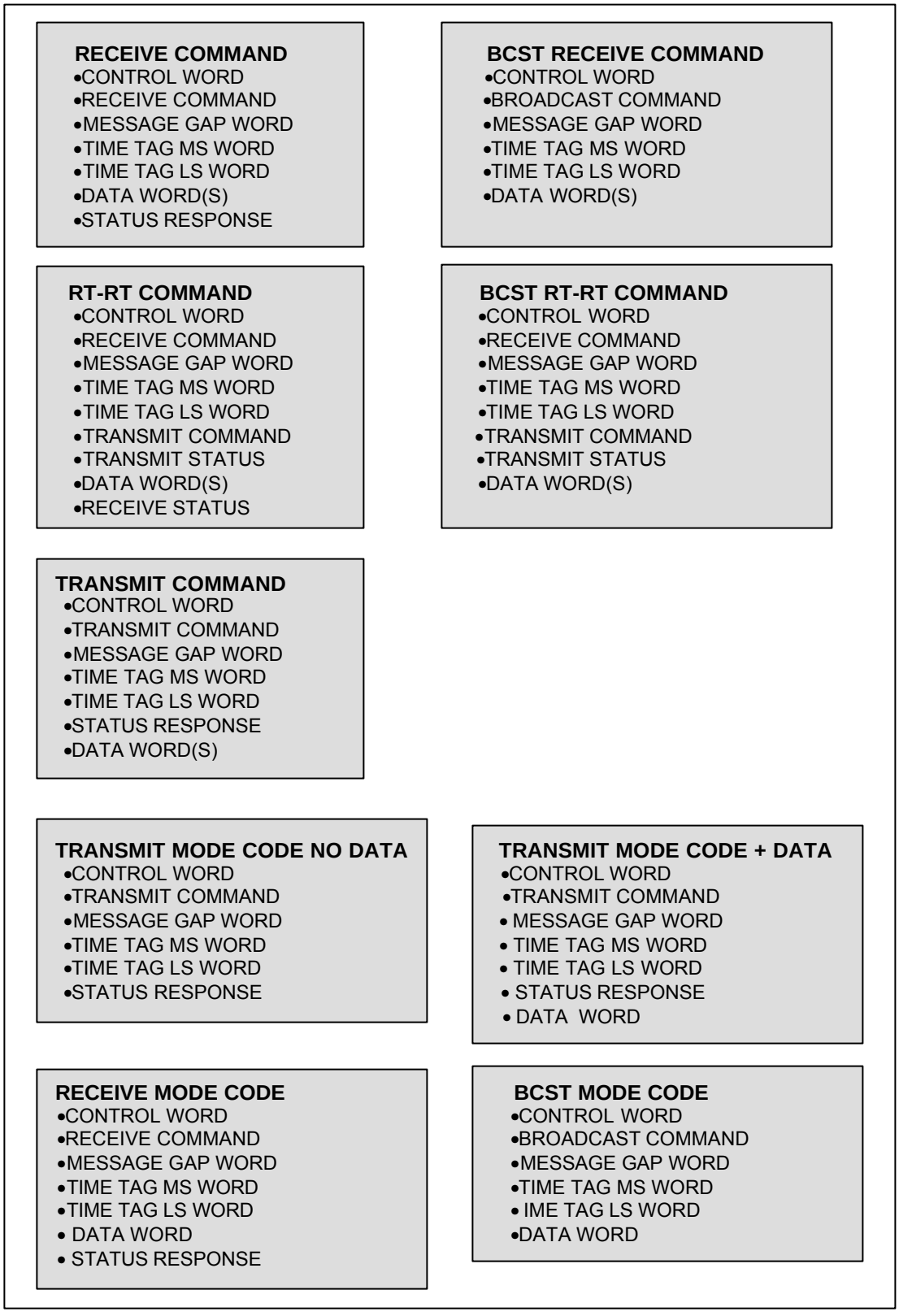
4.4.7 BC STATUS RESPONSE

This space contains the Status response from the RT

4.4.8 BUS CONTROLLER MESSAGES

The following table illustrates all the message table formats used in the Bus Controller. Notice that all the components of a message table are located in contiguous ram locations. This reduces CPU overhead and simplifies software development.

BUS CONTROLLER MESSAGE TABLE FORMATS



4.5.0 MESSAGE MONITOR MESSAGE LISTS AND DATA TABLES

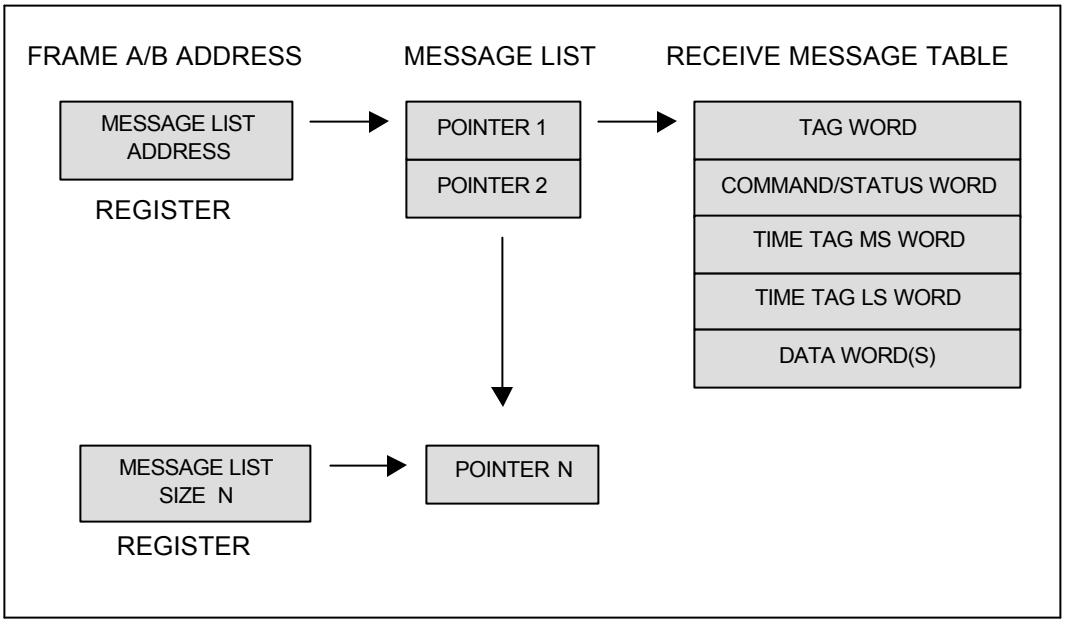
The Message Monitor is organized and controlled by message lists, very similar to the BC mode. Each message list contains the addresses of data tables associated with the list. A message list can contain up to a maximum of 1024 16 bit addresses. The number of message lists and data tables is limited only by the size of the ram. The message list mapping scheme used by the monitor is illustrated in the figure. FRAME A/ B POINTER and FRAME A/ B LENGTH registers have been discussed in a previous section. See details.

The CPU loads the first pointer in each message list. The list is activated by placing its address in the FRAME A or FRAME B POINTER register and the list length in the corresponding FRAME LENGTH register. The ET will calculate all succeeding pointers in the list as it creates message tables . When the number of pointers is equal to the list length the frame terminates.

Messages can be filtered by RT address. This is accomplished with the two MONITOR ADDRESS FILTER REGISTERS. See them for details.

The MESSAGE MONITOR begins storing a message when it detects a command sync, providing the RT address has been enabled in the MONITOR ADDRESS FILTER REGISTERS, and stops storing the message when a gap is detected.

MESSAGE MONITOR MEMORY ORGANIZATION



4.5.1 MESSAGE MONITOR TAG WORD

The MESSAGE MONITOR TAG WORD contains information which is specific to the message in its data table. The ET loads these bits as the message is processed.

The TAG WORD is optional and it can be suppressed from the message table. See Configuration register 3 for details.

15	14	13	12	11	10	9	8
EOM	0	WRDCNT5	WRDCNT4	WRDCNT3	WRDCNT2	WRDCNT1	WRDCNT0
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUS	OVRLAP	SOM	SYNCERR	DATAERR	CMD2ERR	CMD1ERR	RT-RT

Note: Bit 14 is reserved and always reads 0.

EOM **Bits: 15**
1= Complete message has been stored in the ram.

WRDCNT **Bits: 13: 8**
This six bit field represents the total number of words in the message table. This includes: Tag word, Command/Status word, two Time Tag words and Data words.

BUS **Bits: 7**
0= Message was received on bus "A".
1= Message was received on bus "B".

OVRLAP **Bits: 6**
1= A message was detected on the alternate bus before the message on the current bus was completed. The monitor aborts processing the current message, switches to the alternate bus, and begins processing the new message.

SOM **Bits: 5**
1= Message is currently active and being stored in the ram.

SYNCERR **Bits: 4**
1= A contiguous data word was received with a command sync.

DATAERR **Bits: 3**
1= A data word contained an error (encoding, parity, bit count, etc).

CMD2ERR **Bits: 2**
1= The second command word in an RT- RT command contained an error (encoding, parity, bit count, etc).

CMD1ERR **Bits: 1**
1= The command word or the first command word in an RT- RT command contained an error (encoding, parity, bit count, etc).

RT- RT **Bits: 0**
1 The command is an RT- RT message.

4.5.2 MESSAGE MONITOR COMMAND - STATUS WORD
This is the command or status word that triggered the message monitor to start storing the message.

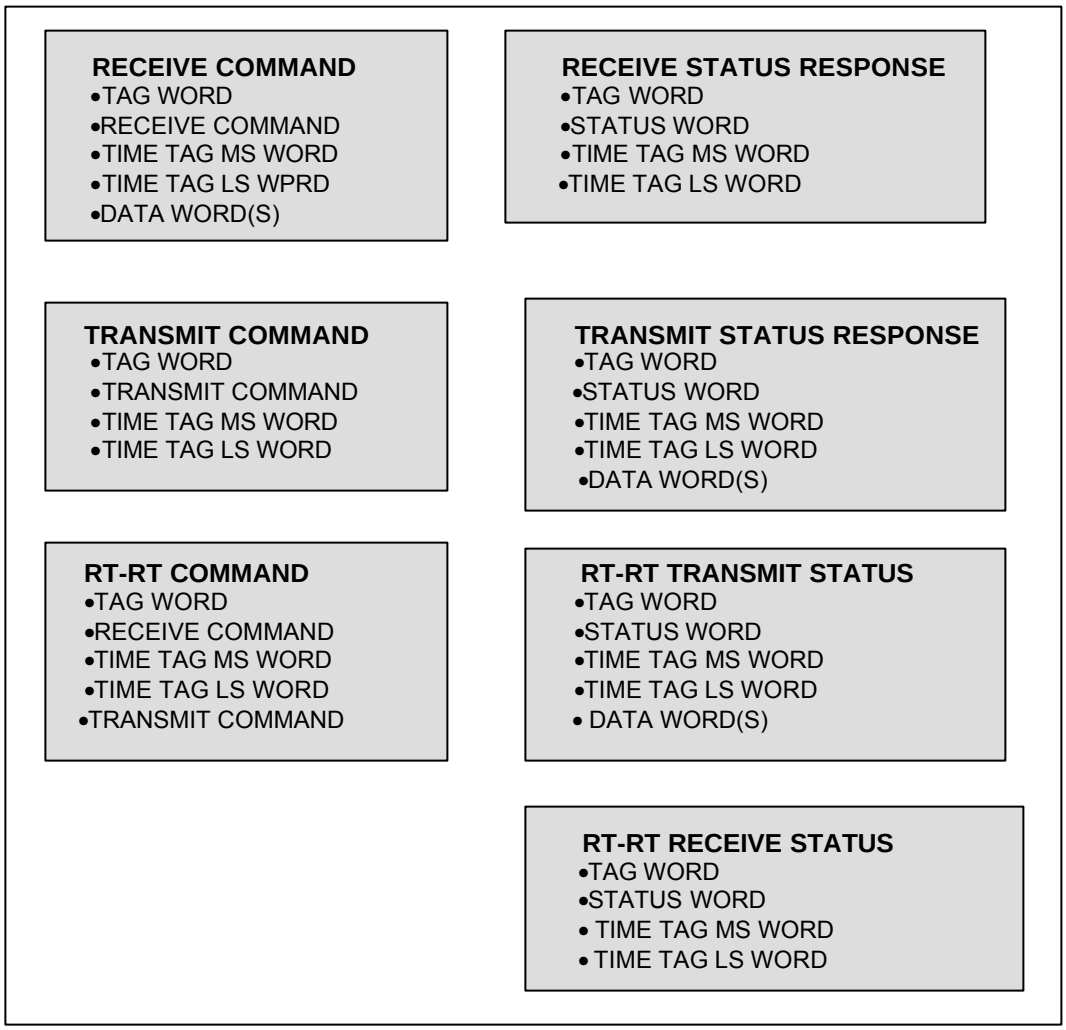
4.5.3 MESSAGE MONITOR TIME TAG MS WORD
This word contains the upper 16 bits of the 32 bit Time Tag.

4.5.4 MESSAGE MONITOR TIME TAG LS WORD
This word contains the lower 16 bits of the 32 bit Time Tag.

Note: The two TIME TAG WORDS are optional and they can be suppressed from the message table. See the Configuration register 3 for details**.**

4.5.5 MESSAGE MONITOR DATA WORDS
This space contains from 1 to 32 data words which may be associated with the message.

4.5.6 MESSAGE MONITOR MESSAGE TABLE FORMATS



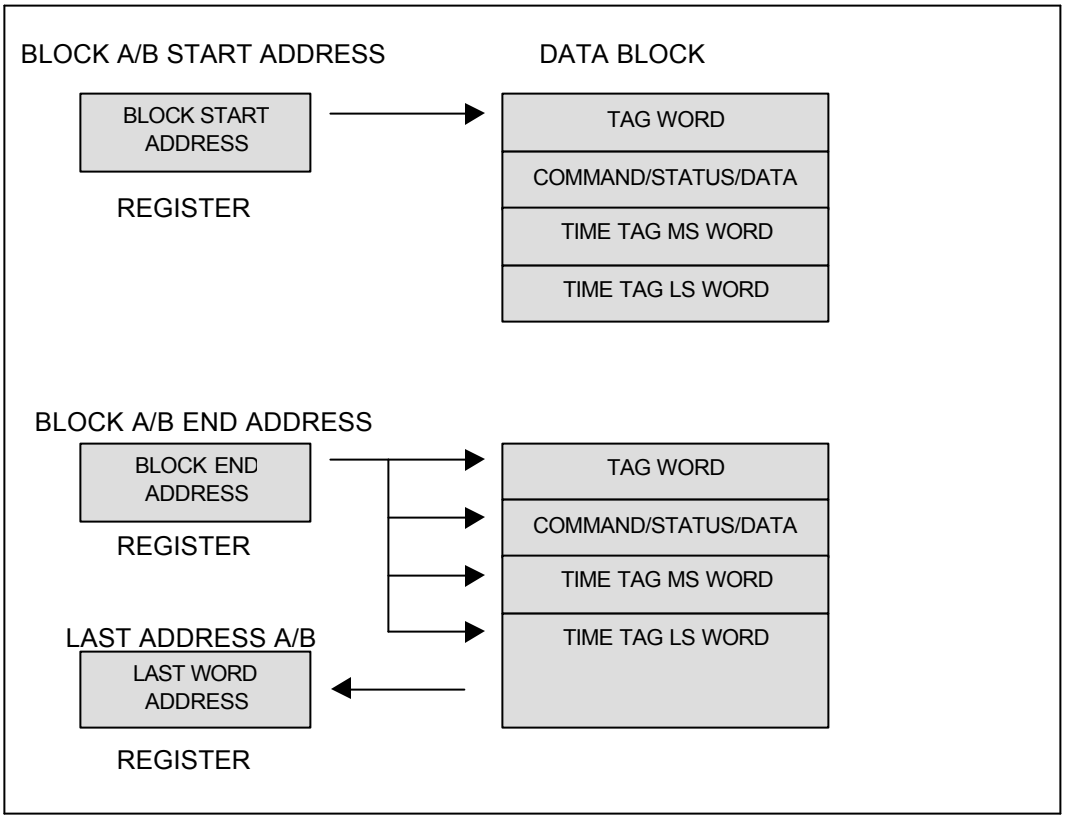
4.6.0 WORD MONITOR DATA TABLES

The Word Monitor is organized and controlled by two registers, the BLOCK A/ B START and the BLOCK A/ B END registers. The data table mapping scheme used by the monitor is illustrated the figure.

The Block Start register contains the address of the start of the data block, while the Block End register contains the end address of the data block; therefore, the amount of ram used by the data block is defined by these two registers. Notice, however, that the Block End register can contain the address of any one of the four words associated with the last word monitored in the block. This is a result of keeping the last four words in contiguous ram locations. The last address register, though, always contains the address of the last word in the data block. The address in this register is calculated by the ET and place in the Last Address register which is read only; therefore, when defining the ram space for a data block in the word monitor, always leave the next four locations after the block end address open. This will provide the reserve memory required to keep all the data in the block contiguous.

All the data in a data block is stored in consecutive addresses, starting with the user supplied block start address and ending with the monitor calculated last word address.

WORD MONITOR MEMORY ORGANIZATION



4.6.1 WORD MONITOR TAG WORD

The WORD MONITOR TAG WORD contains information which is specific to the current word taken from the bus and stored in the data table. The ET loads these bits as the word is processed.

The TAG WORD is optional and it can be suppressed from the message table. See the Configuration register 3 for details.

15	14	13	12	11	10	9	8
GAP07	GAP06	GAP05	GAP04	GAP03	GAP02	GAP01	GAP00
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUS	OVRLAP	0	0	BCST	SYNC	ERROR	GAPDET

Note: Reserved bits 5 and 4 always read 0

GAP **Bits: 15: 8**

This field contains the time interval in microseconds between the current word received and the preceding word received. The resolution of the time interval is 0.5us. If the gap is greater than or equal to 127.5us, then this field will hold at 127.5us. This field is valid only if bit 0 is a "1".

BUS **Bits: 7**

0= Word was received on bus "A".
1= Word was received on bus "B".

OVRLAP **Bits: 6**

1= A message was detected simultaneously on both busses . The Bus Monitor switches to the most current bus.

BCST **Bits: 3**

0 = Broadcast address **NOT** detected in the received word.
1= Broadcast address **WAS** detected in the received word.

SYNC **Bits: 2**

0= The received word contained a Data sync.
1= The received word contained a Command sync.

ERROR **Bits: 1**

0= Bus word had no errors.
1= Bus word contained errors - encoding, parity, bit count, etc.

GAPDET **Bits: 0**

0= This word was contiguous with the previous bus word. Ignore bits (15 - 8).
1= There was a gap between this word and the previous bus word. The gap time is recorded in bits (15 - 8).

4.6.2 WORD MONITOR COMMAND / STATUS - DATA

This is the word read from the bus.

4.6.3 WORD MONITOR TIME TAG MS WORD

This word contains the upper 16 bits of the 32 bit Time Tag.

4.6.4 WORD MONITOR TIME TAG LS WORD

This word contains the lower 16 bits of the 32 bit Time Tag.
Note: Time Tag is optional. See Configuration register 3 for details.

5.0.0 REMOTE TERMINAL MODE CODE OPERATION

5.1.0 GENERAL

This section defines the operation of the NHi- ET when operating as an RT during reception of all the mode commands. The following terms are used in this section:

VALID COMMAND
A command meeting the criteria established by the 1553B standard in paragraph 4.4.1.1.

INVALID COMMAND
A command NOT meeting the criteria established by the 1553B standard in paragraph 4.4.1.1.

UNIMPLEMENTED COMMAND
A command not implemented by the NHi- ET.

UNDEFINED MODE COMMAND
These commands are ignored by the NHi- ET.

The following general response characteristics apply to the NHi- ET when operating on the bus:

RECEIPT OF AN INVALID COMMAND
There is no response and the command is ignored.
RECEIPT OF AN UNIMPLEMENTED COMMAND
There is no response and the command is ignored.
RECEIPT OF AN UNDEFINED COMMAND
There is no response and the command is ignored.

The following abbreviations are used in this discussion:

LSW = LAST STATUS WORD
CDR = CONDITION REGISTER
TW = TAG WORD IN DATA TABLE
ME = MESSAGE ERROR BIT
BCR = BROADCAST BIT

Additional information about each mode code is available in the Interrupt Vector register and the Auxilliary Vector register if it is set to be interrupt driven (see Data Table Pointer word, Interrupt Vector register and Auxiliary Vector register).

5.2.0 TABLE OF RT MODE CODE RESPONSES

5.2.1 DYNAMIC BUS CONTROL (00000; T/ R= 1)

Responds with status except if broadcast
Bits set: *MDCD to "0" in CDR

COMMAND+ DATA WORD
No status response
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW
INV to "1" in TW

T/ R= 0
UNIMPLEMENTED COMMAND

BROADCAST
UNIMPLEMENTED COMMAND

5.2.2 SYNCHRONIZE WITHOUT DATA (00001; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast
Bits set: *MDCD to "0" in CDR
If broadcast- BCR, BCST in LSW & TW to "1"

COMMAND+ DATA WORD
No status response
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW
INV to "1" in TW

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.3 TRANSMIT LAST STATUS WORD (00010; T/ R= 1)

VALID COMMAND
Responds with last status except if broadcast. Status NOT updated.
Bits set: *MDCD to "0" in CDR.

COMMAND+ DATA WORD
Status not cleared. No status response.
Bits set: *MDCD to "0" in CDR. INV to "1" in TW.
ME to "1" in LSW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.4 INITIATE SELF TEST (00011; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast.
Bits set: *MDCD to "0" in CDR.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.5 TRANSMITTER SHUTDOWN (00100; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast. Transmitter on alternate bus inhibited.
Alternate bus transmitter re- enabled by: Reset mode code, Override Transmitter Shutdown mode code, resetting RT, or power up. Bits set: *MDCD to "0" and alt bus (A) (B) XEN to "0" in CDR.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.6 OVERRIDE TRANSMITTER SHUTDOWN (00101; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast. Transmitter on alternate bus enabled.
Bits set: *MDCD to "0" and alt bus (A) (B) XEN to "1" in CDR.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.7 INHIBIT TERMINAL FLAG (00110; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast.
Bits set: *MDCD to "0" and TFE to "0" in CDR. Terminal Flag inhibited in LSW.

If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.8 OVERRIDE INHIBIT TERMINAL FLAG (00111; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast.
Bits set: *MDCD to "0" and TFE to "1" in CDR. Terminal Flag enabled in LSW.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.9 RESET REMOTE TERMINAL (01000; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast. Both Transmitters enabled and Terminal
Flag enabled. Pointer base address register set to 2048 dec. External terminal address loaded.
Bits set: BUSY to “1” in LSW after Status word transmission.
If broadcast- BCR, BCST in LSW & TW set to "1".

COMMAND+ DATA WORD
No status response.
Bits set:
INV to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.10 RESERVED MODE CODES (01001- 01111; T/ R= 1)

VALID COMMAND
Responds with status except if broadcast.
Bits set: *MDCD to "0" in CDR.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

BROADCAST COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME and BCR set to "1" in LSW.
INV and BCST set to "1" in TW.

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.11 TRANSMIT VECTOR WORD (10000; T/ R= 1)

VALID COMMAND
Responds with status followed by vector word except if broadcast.
Bits set: *MDCD to "0" in CDR.

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

BROADCAST COMMAND
UNIMPLEMENTED COMMAND

BROADCAST COMMAND+ DATA WORD

UNIMPLEMENTED COMMAND

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.12 SYNCHRONIZE WITH DATA WORD (10001; T/ R= 0)

VALID COMMAND
Responds with status except if broadcast. Data word stored into RAM. Data word will update lower 16 bits of real time clock depending on the configuration of the RTC CONTROL REGISTER.

Bits set: *MDCD to "0" in CDR.
If broadcast- BCR, BCST in LSW & TW to "1".

COMMAND NO DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.

COMMAND + EXTRA DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV set to "1" in TW.

BROADCAST + EXTRA DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME and BCR to "1" in LSW.
INV and BCST to "1" in TW.

T/ R= 1
UNIMPLEMENTED COMMAND

T/ R= 1 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.13 TRANSMIT LAST COMMAND (10010; T/ R= 1)

VALID COMMAND
Responds with status followed by LAST VALID COMMAND word except if broadcast.
Status and command registers NOT updated.
Bits set: *MDCD to "0" in CDR.

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR.
INV to "1" in TW.
ME to "1" in LSW.

BROADCAST COMMAND
UNIMPLEMENTED COMMAND

BROADCAST COMMAND+ DATA WORD

UNIMPLEMENTED COMMAND

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.14 TRANSMIT BIT WORD (10011; T/ R= 1)

VALID COMMAND
Responds with status followed by BIT word.
Bits set: *MDCD to "0" in CDR.

COMMAND+ DATA WORD
No status response.
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

BROADCAST COMMAND
UNIMPLEMENTED COMMAND

BROADCAST COMMAND+ DATA WORD
UNIMPLEMENTED COMMAND

T/ R= 0
UNIMPLEMENTED COMMAND

T/ R= 0 AND BROADCAST
UNIMPLEMENTED COMMAND

5.2.15 SELECTED TRANSMITTER SHUTDOWN (10100; T/ R= 0)

VALID COMMAND
Responds with status except if broadcast.
Bits set:
If broadcast- BCR, BCST to "1" in TW & LSW.

COMMAND + EXTRA DATA WORD
No response
Bits set:

INV to "1" in TW.

COMMAND WITHOUT DATA WORD
No response
Bits set:

BROADCAST WITH EXTRA DATA WORD
No response
Bits set:
INV & BCST to "1" in TW

BROADCAST WITHOUT DATA WORD
No response
Bits set:

5.2.16 OVERRIDE SELECTED TRANSMITTER SHUTDOWN (10101; T/ R= 0)

VALID COMMAND
Responds with status except if broadcast.
Bits set:
If broadcast- BCR, BCST to "1" in TW & LSW.

COMMAND + EXTRA DATA WORD
No response
Bits set:
INV to "1" in TW.

COMMAND WITHOUT DATA WORD
No response
Bits set:

BROADCAST WITH EXTRA DATA WORD
No response
Bits set:
INV & BCST to "1" in TW

BROADCAST WITHOUT DATA WORD
No response
Bits set:

5.2.17 RESERVED MODE CODES (10110- 11111; T/ R= 1)

VALID COMMAND
Responds with status and data word.
Bits set: *MDCD to "0" in CDR.

COMMAND + DATA WORD
No response
Bits set: *MDCD to "0" in CDR. ME to "1" in LSW.
INV to "1" in TW.

BROADCAST COMMAND
UNIMPLEMENTED COMMAND

BROADCAST COMMAND+ DATA WORD
UNIMPLEMENTED COMMAND

5.2.18 RESERVED MODE CODES (10110- 11111; T/ R= 0)

VALID COMMAND
Responds with status except if broadcast.
Bits set:
If broadcast- BCR, BCST to "1" in TW & LSW.

COMMAND + EXTRA DATA WORD
No response
Bits set:
INV to "1" in TW.

COMMAND WITHOUT DATA WORD
No response
Bits set:
BROADCAST WITH EXTRA DATA WORD

No response
Bits set:
INV & BCST to "1" in TW

BROADCAST WITHOUT DATA WORD
No response
Bits set:

6.0.0 INITIALIZATION

There are several types of initialization that can set up the NHi- ET parameters.

6.1.0 INTERNAL INITIALIZATION

There are two methods of initializing the NHi ET. Each will produce the same results. They are: Hardware(MRST) and Software (writing to address 15, data not used). After these resets have been performed, the NHi ET is set to the RT mode and all internal state machines are reset.

The hardwire terminal address is loaded when a hardware or software reset occurs. The hardwire address is connected to I/ O DAT(5: 0) pins. I/ O DAT(4: 0) are used for the address and I/ O DAT5 is used to set odd parity in the address. The address is wired using external 4.7K pull-down resistors to set a low and internal 64K pull- up resistors to set a high.

The following table summarizes the condition of internal registers after a reset has been performed.
Note: All register bit set to 0 at reset except as noted.

REGISTER RESET TABLE

ADDR	REGISTER	BITS=1	COMMENTS
0	CONTROL	HWD	
1	POINTER TABLE ADDRESS		LOADED WITH 4096 DEC WORD
2	BASIC STATUS*		HARDWIRE ADDRESS LOADED
21	CONFIGURATION 3		
3	INTERRUPT MASK	ALL	ALL INTERRUPTS ARE MASKED
4	INTERRUPT VECTOR		
3	INTERRUPT REQUEST		
4	AUXILLIARY VECTOR		UNDEFINED
5,6	RTC; HIGH.LOW		NOT AFFECTED BY RESET
7	RTC CONTROL		
8	FIFO		RESET ONLY BY MRST
11	LAST COMMAND		UNDEFINED
12	LAST STATUS*		HARDWARE ADDRESS LOADED
18	ENCODER STATUS		
19	CONDITION	9,11-13	
13	FRAME “A” POINTER		LOADED WITH 2058 DEC
16	FRAME “B” POINTER		LOADED WITH 4096 DEC
14	FRAME "A" LENGTH		CLEARED
17	FRAME "B" LENGTH		CLEARED
	MSG MONITOR ADDR FILTER 1		ALL ADDRESSES UNMASKED
	MSG MONITOR ADDR FILTER 2		ALL ADDRESSES UNMASKED

* See RESET BUSY TABLE

RESET FUNCTION TABLE

RESET TYPE	REGISTERS	STATE MACHINES
MODE CODE	RESET	RESET
MRST(HARDWARE)	RESET	RESET
SOFTWARE	NO CHANGE	RESET

RESET BUSY TABLE

This table defines the state of the Busy bit in the Basic Status and Last Status registers.

RESET TYPE	RTCC REG BIT 6	CONFIG REG1 BIT1	BUSY BIT
MRST(HARDWARE)	X	X	SET TO "1"
SOFTWARE	0	0	SET TO "1"
SOFTWARE	X	1	NO CHANGE
SOFTWARE	1	X	NOCHANGE
MODE CODE	X	X	SET TO "1"
MODE CODE	X	X	SET TO "1"
MODE CODE	X	X	SET TO "1"

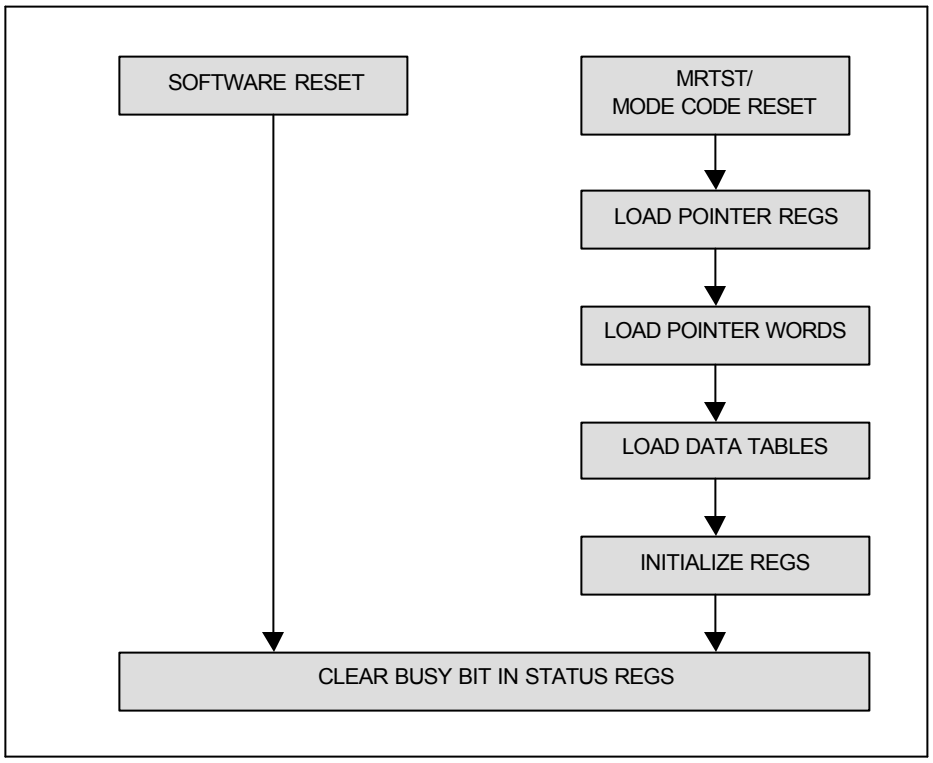
6.2.0 HOST INITIALIZATION OF NHi-ET

The host will usually want to initialize the ET at power up or at any other time it feels the procedure is necessary. At power up the host must initialize the registers discussed in the preceding section if the default settings of the internal initialization are not suitable. In addition the host must initialize the RAM, the pointer tables, and the data table tag words for each command type and subaddress.

The host initialization of the registers is also required only after a hardware reset. Since the RAM is not affected by any resets, it does not have to be re- initialized unless data has been lost or corrupted, therefore the pointer tables will remain intact.

The following flow diagram is a suggested method of host initialization of the NHi- ET in the RT mode as a function of the type of reset which has occurred.

TYPICAL NHi-ET INITIALIZATION PROCEDURE BY CPU



Note: If bit 6 in the RTCC or bit 1 in Configuration reg 1 is set to "1", the Software reset wil NOT set the busy bit in the Status regs.

7.0.0 INTERRUPT HANDLING

When an interrupt request is received by the NHi-ET, the *IRQ line goes low and header information about the message that caused the interrupt is pushed on an internal FIFO. If another interrupt request is received before the CPU performs an acknowledge, it's header information is also pushed onto the FIFO. I n this manner, there is no danger of losing interrupt vectors or header information due to receiving multiple interrupt requests before an acknowledge by the CPU takes place. The FIFO can hold header information for six interrupt messages. If an interrupt request occurs when the FIFO is full, a vector indicating FIFO overflow is first pushed onto the FIFO and then the header information for the message which caused the overflow is pushed onto the FIFO. As a result, the header information from the two oldest messages is lost.

If the FIFO is in the revolving mode, the FIFO will store seven interrupts. When another interrupt is issued and the FIFO contains seven previous headers, the new header is pushed onto the FIFO and the oldest header is lost.

7.1.0 HARDWARE INTERRUPT ACKNOWLEDGE

To acknowledge an interrupt in hardware, the *INTACK line is taken low, the *HCS line held high and the *INTPI line is held low. This pops the interrupt header information off the FIFO and into the IVR and AVR. The *IRQ line will go high if the FIFO is empty, but remain low if there are additional interrupt headers on the FIFO. If the NHi-ET is in the RT mode, the IVR will be outputted on the upper and lower byte of the CPU data bus. If the *INTPI line is high, then *INTACK is ignored.

The IVR and AVR can be read from address 4 after performing the hardware interrupt.

If there are more interrupt headers on the FIFO, indicated by the *IRQ remaining low after the interrupt acknowledge, the procedure is repeated until the FIFO is empty. An empty FIFO is indicated by the *IRQ line returning high after an interrupt acknowledge and bit 8 in the AVR will be a "1".

7.2.0 SOFTWARE INTERRUPT ACKNOWLEDGE

If the host CPU does not support a hardware interrupt acknowledge, a software acknowledge can be performed by reading address 8. This read pops the interrupt header information off the FIFO and into the IVR and AVR and places their contents on the CPU data bus. The *IRQ line will go high if the FIFO is empty and remain low if there are additional interrupt headers on the FIFO.

If there are more interrupt headers on the FIFO, indicated by the *IRQ remaining low after the interrupt acknowledge, the procedure is repeated until the FIFO is empty. An empty FIFO is indicated by the *IRQ line returning high after an interrupt acknowledge and the MSB in the AVR will be a "1".

8.0.0 TIPS - HINTS 'N TRICKS

This section will help the USER apply the ET to a system and implement its various features.

8.1.0 BUS CONTROLLER APPLICATIONS

The NHi-ET Bus Controller is flexible, powerful, and very easy to use. The number of operations required to initialize the device and to examine results of a data message transfer has been minimized.

The BC function of the NHi-ET employs registers embedded in the protocol chip and its internal ram to perform its various tasks. These tasks include:

Initiating Message Transfers
Diagnose RT Responses
Take Appropriate Action on Error Conditions
Data Storage

Message lists, containing the addresses of specific messages, are used to develop specific operational scenarios. The number of message lists and message tables is limited only by the size of the ram. A list can have up to 1024 messages. A message list is activated by placing its address in one of the two FRAME START REGISTERS and the list length in one of the FRAME LENGTH REGISTERS.

8.1.1 BC REGISTERS

This a brief description of the BC registers and their role. Specific bit functions are given in the address map section of this manual. Only the functions pertinent to the BC are described here. The following registers are used in the BC function:

8.1.1.1 CONFIGURATION REG 2 Address: 4

Complete EOM and continue.
Goto next message.
Goto EOF and continue.
Stop at end of current message.
Stop at end of current frame.
Abort - Go off line.
Go default frame.
Note: The first three Functions only apply to a bus jam condition.

8.1.1.2 CONFIGURATION REG 1 Address: 9

Mode select - RT, BC, MT.
Start BC.
Select default frame A or B.
Select default bus A or B.
Force bus A or B.

8.1.1.3 FRAME "A" POINTER Address: 13

Holds address of a selected BC message list.

8.1.1.4 FRAME "A" LENGTH Address: 14

Holds number of messages in frame A message list.
End of frame A options - Stop, Repeat , Go Alternate.
Stop on frame error.
Stop on status set.
Interrupt at end of frame.

8.1.1.5 FRAME "B" POINTER Address: 16

Holds address of a selected BC message list

8.1.1.6 FRAME "B" LENGTH Address: 17

Holds number of messages in frame B message list
.End of frame B options - Stop, Repeat, Go Alternate.
Stop on frame error.
Stop on status set.
Interrupt at end of frame.

8.1.1.7 CONDITION REGISTER Address 19

Bus A jammed.
Bus B jammed.
Current BC frame A or B.
End of frame A.
End of frame B.
Current frame busy.

8.1.1.8 END OF FRAME GAP Address: 20

End of frame delay before start of next frame - 64us resolution, 16 bits.

8.1.1.9 CONFIGURATION REG 3 Address: 21

Sets bus jam threshold - the number of extra words(0- 31) a message can have before a bus jam is declared.
Defines response to BCST bit if set in status word.
Sets global retry options.

8.1.2 BC RAM

The BC ram is used to store message lists and message tables. A message list is a block of consecutive memory locations, each of which contains the address of a message table. This list can contain any number of message table addresses up to a maximum of 1024. Many message lists can be stored in the ram, limited only by ram size, each one containing a different scenario. A particular message table may be a member of any number of message lists and occupy the same or different positions in each list.

8.1.3 BC MESSAGE TABLE

A message table is a block of consecutive memory which contains all the components of one message. These components are:
BC Control word.
BC Command word.
Message gap word.
32 bit time tag(two words).
2nd Command word (RT- RT only).
Data words.
Status word.
2nd Status word(RT- RT only).

In the case of an RT- RT transfer, the second command and status words are also in the message block as shown. The message block is designed to contain all the information associated with the message. Parts of the message, the command word for example, are not segregated into a separate ram area which would complicate reading or writing a message block by the CPU. The unified message block approach used in the NHi- ET provides faster access to a message block by the CPU and minimizes overhead.

8.1.4 BC CONTROL WORD

The BC control word is used by the CPU to define parameters in the message and it is used by the Protocol chip to report information back to the CPU. The following parameters are loaded into the BC control word by the CPU and used by the protocol chip for a message:
Status bit analysis.
Message bus - A or B.
Local retry actio in the event of an error. If there is no local retry, then the global retry action set in configuration register 2 is used if an error occurs.
The following information about that message is loaded into the BC control word by the protocol chip and used by the CPU:
Start of message flag.
No response.
Error.
Status bit set.
Retry attempted.
End of message flag.

8.1.5 BC COMMAND WORD

The command word is any of the 1553 valid commands. This word defines the type of data transfer in the message BC to RT, RT to BC, RT to RT, or Mode code.

8.1.6 MESSAGE GAP WORD

The message gap word defines delay between the end of the current message and the start of the next message in the list. The maximum intermessage gap is 4ms with a resolution 1us. The minimum inter message gap is approximately 8us when the delay is set to 0. The following other parameters are set in the message gap word:
Message stop on error.
Message stop on status set.
Interrupt on end of message.
NO- OP.

If the NO- OP is set in a message, the message is ignored except for its inter message delay. This feature can be used to extend the maximum intermessage gap from 4ms to any length, just by putting a series of NO- OP messages between two operational messages.

8.1.7 32 BIT TIME TAG(2 WORDS)

The internal time tag resolution is selectable as 1,2,4,8,16,32 or 64us.

8.1.8 BC INTERRUPTS

The following conditions will cause an interrupt to the CPU if they are not masked by the interrupt mask register.

PRIORITY	INTERRUPT	COMMENT
0	END OF MESSAGE	ENABLED IN MSG GAP WORD
1	END OF FRAME	ENABLED IN REGS 14 & 17
2	ERROR	ERRORS IN RT RESPONSE
3	RETRY	ERROR CAUSED RETRY
4	FIFO OVERFLOW	FIFO IS FULL
5	STATUS SET	STATUS BIT SET OR STATUS ADDRESS ERROR
6	NO RESPONSE	RT DID NOT RESPOND
7	FAILSAFE TIMEOUT	FAILSAFE TIMEOUT IN NHi-ET ENCODER

The following header information about the BC message that caused the interrupt is pushed on to the Fifo:

Priority
Message number in frame
Frame A or B
Bus A or B

8.1.9 BUS CONTROLLER EXAMPLE

Now, lets see how we would put the BC into operation!

The following sequence is one approach:

Define various message tables and load them into ram.
Define various message lists and load them withn the addresses of the previously defined message tables.

At this point, we have the ram loaded with individual messages tables and a number of message lists. Each message list contains the addresses of several(up to 1024) message tables.

Load the address of the first message list to be used into register 13, Frame A.
Load the total number of messages in that list into register 14, Frame length A.
Load the address of the second message list to be used into register 16, Frame B.
Load the total number of messages in that list into register 17, Frame length B.
Set the mode to BC in register 9.
Set the default frame to A in register 9.
Set the "Default bus" in register 9.
Set "Go alternate frame" in register 14.
Set "int on EOF" in register 14.
Set "Stop at EOF" in register 17.
Set "int on EOF" in register 17.
Set "Start BC" in register 9.

Based on these setup conditions, the following events will occur:

Frame A will execute the message list it contains. At the end of frame A, a frame gap delay will occur if a number other than 0 was loaded into register 20. At the end of the frame gap delay, The alternate frame, in this case B, will execute the message list it contains. At the end of frame B, the sequence will end and the BC will go off line.

Note: The alternate frame is the frame which is NOT the current frame. If the current frame is B, then the alternate frame is A. The alternate bus is the bus which is NOT the active bus.

This method is applicable if each message list contains many messages, tens or hundreds; however, each message list may only contain 10 or 20 messages and we would like to run a number of the small message lists automatically.

For the case of running many small message lists automatically, the approach is slightly different.

Lets say we want to run 10 different message lists automatically.

First create a super message list which contains all the addresses of the 10 smaller sub message lists. If we want to know when each of the sublists is completed, set the last message in each sub list to fire the interrupt. If we want to examine some of the data before the next sub list begins, set the inter message gap on the last message of the sub list to what ever delay is required.
Then:
Load the address of the super message list into register 13, Frame A.
Load the total number of messages in the super list into register 14, Frame length A.
Set the mode to BC in register 9.
Set the default frame to A in register 9.
Set the "Default bus" in register 9.
Set "int on EOF" in register 14.
Set "Stop at EOF" in register 14.
Set "Start BC" in register 9.
All the messages in the super list will be executed and an interrupt will be issued at the end of each sub list. An intermessage gap of some defined value will separate each sub list of messages.

Instead of stopping at the end of frame A, we could have elected to "Go alternate frame", in which case, frame B would then execute what ever message lists were in frame B. If we then selected the "Go alternate frame" option at the end of frame B, frame A would be executed.

Note: The default frame is the frame selected in configuration register 1. The default bus is the bus selected in the BC control word.

8.1.10 SAMPLE BUS CONTROLLER MEMORY MAP

BC Registers

REGISTERS	DATA(hex)
CONFIGURATION 1	0900
CONFIGURATION 2	0000
CONFIGURATION 3	0003
INTERRUPT MASK	0000
FRAME A MSG TABLE ADDRESS	0100
FRAME A LENGTH	0004

FRAME "A" MESSAGE LIST

ADDRESS	DESCRIPTION	DATA(hex)
0100	MSG 1 ADDRESS(BC-RT 2 WORDS)	0200
0101	MSG 2 ADDRESS(RT-BC 4 WORDS)	0220
0102	MSG 3(RT-RT 6 WORDS)	0240
0103	MSG 4(BROADCAST 1 WORD)	0260

BC FRAME "A" MESSAGE TABLES

MESSAGE 1

ADDRESS	DESCRIPTION	DATA(hex)
0200	CONTROL WORD	0000
0201	RECEIVE COMMAND WORD	0822
0202	MESSAGE GAP WORD	0010
0203	TIME TAG MS WORD	****
0204	TIME TAG LS WORD	****
0205	DATA WORD 1	1234
0206	DATA WORD 2	5678
0207	STATUS WORD	0800

MESSAGE 2

ADDRESS	DESCRIPTION	DATA(hex)
0220	CONTROL WORD	0080
0221	RECEIVE COMMAND WORD	0C24
0222	MESSAGE GAP WORD	4010
0223	TIME TAG MS WORD	****
0224	TIME TAG LS WORD	****
0225	STATUS WORD	0800
0226	DATA WORD 1	ABCD
0227	DATA WORD 2	FADE
0228	DATA WORD 3	BAD1
0229	DATA WORD 4	BEAD

MESSAGE 3

ADDRESS	DESCRIPTION	DATA(hex)
0240	CONTROL WORD	0182
0241	RECEIVE COMMAND WORD	0826
0242	MESSAGE GAP WORD	4010
0243	TIME TAG MS WORD	****
0244	TIME TAG LS WORD	****
0245	TRANSMIT COMMAND WORD	1426
0246	TRANSMIT STATUS WORD 1	1000
0247	DATA WORD 1	1111
0248	DATA WORD 2	2222
0249	DATA WORD 3	3333
024A	DATA WORD 4	4444
024B	DATA WORD 5	5555
024C	DATA WORD 6	6666
024D	RECEIVE STATUS WORD 4	0800

MESSAGE 4

ADDRESS	DESCRIPTION	DATA(hex)
0260	CONTROL WORD	0102
0261	RECEIVE COMMAND WORD	F821
0262	MESSSAGE GAP WORD	050
0263	TIME TAG MS WORD	****
0264	TIME TAG LS WORD	****
0265	DATA WORD	BEEF

8.2.0 MESSAGE MONITOR APPLICATIONS

The NHi-ET Message Monitor operation is very similar to that of the Bus Controller . The number of operations required to initialize the device and to examine results of a data message transfer has been minimized. Each message can have an associated tag word and/ or an associated 32 bit time tag.

The Message Monitor function of the NHi-ET employs registers embedded in the protocol chip and its internal ram to perform its various tasks. These tasks include:
Store Bus Messages
Diagnose RT Responses
Data Storage

A Message list, containing the addresses of message tables, is used to keep track of the message which have been monitored and stored.. The number of message lists and message tables is limited only by the size of the ram. A list can have up to 1024 messages. A message list is activated by placing its address in one of the two FRAME START REGISTERS and the list length in one of the FRAME LENGTH REGISTERS.

The CPU defines the address of the first message table in a message list. After the first message table has been stored in ram, the ET calculates the address in ram where the next message table will be stored and writes it into the next contiguous address in the message list. This process continues, the ET calculating and writing succeeding message table addresses until the frame length has been reached. The Message Monitor then performs the indicated End- of_ Frame action.

8.2.1 MESSAGE MONITOR REGISTERS

This a brief description of the Message Monitor registers and their role. Specific bit functions are given in the address map section of this manual. Only the functions pertinent to the Message Monitor are described here.

8.2.1.1 CONFIGURATION REG 2 **Address: 4**

Stop at end of current message.
Stop at end of current frame.
Abort - Go off line.
Go default frame.

8.2.1.2 CONFIGURATION REG 1 **Address: 9**

Mode select - RT, BC, MT.
Start MT.
Select default frame A or B.
Select Message or Word monitor.

8.2.1.3 FRAME "A" POINTER Address: 13

Holds address of a selected MT message list.

8.2.1.4 FRAME "A" LENGTH Address: 14

Holds number of messages in frame A message list.
End of frame A options - Stop, Repeat , Go Alternate.
Interrupt at end of frame.

8.2.1.5 FRAME "B" POINTER Address: 16

Holds address of a selected MT message list.

8.2.1.6 FRAME "B" LENGTH Address: 17

Holds number of messages in frame B message list.
End of frame B options - Stop, Repeat, Go Alternate.
Interrupt at end of frame.

8.2.1.7 CONDITION REGISTER Address: 19

Current BC frame A or B.
End of frame A.
End of frame B.
Current frame busy.

8.2.1.8 CONFIGURATION REGISTER 3 Address: 21

Enable tag word option.
Enable time tag option.

8.2.1.9 ADDRESS FILTER (0- 15) Address: 22

Masks terminal addresses 0 - 15.
A "1" will mask the address.

8.2.1.10 ADDRESS FILTER (16- 31) Address: 26

Masks terminal addresses 16 - 31.
A "1" will mask the address.

8.2.2 MESSAGE MONITOR RAM

The Message Monitor ram is used to store message lists and message tables in much the same manner as the BC. A message list is a block of consecutive memory locations, each of which contains the address of a message table. This list can contain any number of message table addresses up to a maximum of 1024. Many message lists can be stored in the ram, limited only by ram size, each one containing a different scenario.

8.2.3 MESSAGE MONITOR MESSAGE TABLE

A message table is a block of consecutive memory which contains all the components of one message. These components include:
Tag word control word.
Command word or Status word.
32 bit time tag(two words).
Data words.

The format of a Message Table is a function of the type of command stored in ram. These formats are illustrated in a later section of this manual. The unified message block approach again used in the Message monitor to provide faster access to a message block by the CPU and minimizes overhead.

8.2.4 MESSAGE MONITOR TAG WORD

The ET loads the Message monitor tag word with information about the message received. This information is used by the CPU to analyze the message. The following parameters are loaded into the Monitor tag word by the ET:
Start of message flag.
Data error - Data word(s) contained error(s).
Command 1 error - Comand word contained error(s).
Command 2 error - Second command word of an RT- RT command contained error(s).
Over lap - A message was detected on the alternate bus before the current message was Completed. The Message monitor aborts processing the current message, switches to the alternate bus and begins processing the new message.
Sync error - A contiguous data word had a command sync.
Bus A- B.
RT- RTcommand - Message was an RT- RT command.
End of message flag.

8.2.5 COMMAND WORD OR STATUS WORD

This is the Command or Status word associated with the message.

8.2.6 DATA WORD(S)

These are the data word(s) associated with the message.

8.2.7 MESSAGE MONITOR EXAMPLE

Setting up and Implementing the MESSAGE monitor!

Put address of message list in FRAME REG.
Put address of 1st message table in 1st location of message list.
Put message list size in FRAME LENGTH REG.
Set end of MESSAGE LIST options in FRAME LENGTH REG.
Select end of MESSAGE LIST interrupt in FRAME LENGTH REG.
Set the address Masks in REG's 22 and 26.
Select tag word option in REG 21.
Select time tag option in REG 21.
Select MESSAGE monitor in REG 9.
Start monitor in REG 9.

The monitor now collects messages until the message list size is reached. The selected End- of-Message List option is executed and an interrupt is issued if enabled.

The CPU can now read the entire message list. Each word in the message list is the address of a data block stored in ram. The first word in the block is the Tag Word associated with that message.

8.3.0 WORD MONITOR APPLICATIONS

The NHi-ET Word Monitor operation collects data on a word- by- word basis . Each word that is received and stored in the ram can have an associated tag word and/ or an associated 32 bit time tagged The number of operations required to initialize the device and to examine results of a data message transfer has been minimized.

The Word Monitor function of the NHi-ET employs registers embedded in the protocol chip and its internal ram to perform its various tasks. These tasks include:
Store Bus words
Diagnose RT Responses
Data Storage

A data block, containing the bus word ,tag word and time tag, is used to store the monitored data.. The begining of the data block is determined by the address placed in the BLOCK START REGISTER and the end of the data block is determined by the address in the BLOCK END REGISTER.

The BLOCK END REGISTER works inconjunction with the LAST ADDRESS REGISTER. Since a bus word can occupy up to four words in ram, tag word + 2 time tag words, the address in the BLOCK END REGISTER could occur on any of these four words. In order to keep the four words of the last ram entry contiguous in ram, the Word monitor uses the LAST ADDRESS REGISTER to store the address of the last word in the data block. The ET calculates this address and puts it in the LAST ADDRESS REGISTER. This address could be identical to that in the BLOCK END REGISTER or up to three addresses greater; therefore, always reserve the four addresses after the address in the BLOCK END REGISTER for this contingency.

8.3.1 WORD MONITOR REGISTERS

This a brief description of the Message Monitor registers and their role. Specific bit functions are given in the address map section of this manual. Only the functions pertinent to the Message Monitor are described here.

CONFIGURATION REG 2 **Address: 4**

Stop at end of current message.
Stop at end of current frame.
Abort - Go off line.
Go default frame .

8.3.1.2 CONFIGURATION REG 1 Address: 9

Mode select - RT, BC, MT.
Start MT.
Select default frame A or B.
Select Message or Word monitor.

8.3.1.3 BLOCK “A” START Address: 13

Holds address of a selected MT message list.

8.3.1.4 BLOCK “A” END Address: 14

Holds number of messages in frame A message list.
End of frame A options - Stop, Repeat , Go Alternate.
Interrupt at end of frame.

8.3.1.5 BLOCK “B” START Address: 16

Holds address of a selected MT message list.

8.3.1.6 BLOCK “B” END Address: 17

Holds number of messages in frame B message list.
End of frame B options - Stop, Repeat , Go Alternate.
Interrupt at end of frame.

8.3.1.7 CONDITION REGISTER Address: 19

Current BC frame A or B.
End of frame A.
End of frame B.
Current frame busy.

8.3.1.8 WORD MONITOR END OF BLOCK OPTIONS Address: 20

End of block A options - Stop, Repeat, Go Alternate.
Interrupt at end of block A.
End of block B options - Stop, Repeat, Go Alternate.
Interrupt at end of block B.

8.3.1.9 CONFIGURATION REGISTER 3 Address: 21

Enable tag word option.
Enable time tag option.
Enable time tag command syncs only option.

8.3.2 WORD MONITOR EXAMPLE

Setting up and Implementing the WORD monitor!

Set address of data block in BLOCK A/ B REG.
Set end of data block address in BLOCK A/ B END REG.
Set end of DATA BLOCK options in REG 20.
Select end of block interrupt in REG 20.
Select tag word option in REG 21.
Select time tag option in REG 21.
Select word monitor in REG 9.
Start monitor in REG 9.

The monitor now collects data until the end of block address is reached. The address of the last word in the block is stored in the Last Word Address register, the selected End- of- Block option is executed and an end of block interrupt is issued if enabled in reg 20.

The CPU can now read the entire data block and use the Tag Words and the Time Tags to analyze the data.

8.4.0 SIMULTANEOUS MONITOR AND REMOTE TERMINAL

The ET can operate as a simultaneous monitor and remote terminal. This mode is activated by setting bits 8 and 9 of Configuration register 1 to a "1". In this mode the ET will respond as a remote terminal to the address set in the Basic Status register. This address is set either by the hardwire address or software.

The ET will respond to all addresses except that in the Basic Status register as a monitor, word monitor or message monitor depending on the monitor mode selected. In the message monitor mode, the ET will only respond to terminal addresses which **have not** been masked in registers 22 and 26.

When the ET receives a message with the address in the Basic Status register, it will become a fully operational Remote Terminal for that message.

8.4.1 SIMULTANEOUS MODE INTERRUPT HANDLING

In this dual mode of operation, all the interrupts for the Remote terminal and Monitor operation remain valid; therefore both RT and MT messages can set interrupts and push headers on the FIFO during the dual mode operation.

When an interrupt is pushed on the FIFO, the priority level determines the circumstances that caused it. See the AVR and IVR descriptions for details. The following table describes the type of message which issued the interrupt in the Simultaneous mode:

PRIORITY LEVEL	MESSAGE TYPE
1,2,3,7	REMOTE TERMINAL
5	MONITOR
4	FIFO OVER FLOW

8.5.0 PC BOARD CONSIDERATIONS AND GUIDE LINES

There are a few guide lines which should be observed when mounting the ET and its coupling transformer on a PC board. The following considerations will prevent layout problems on the board:

The width of the two land traces for each Bus from the ET to the transformer must be as wide as possible(0.1in min width).

The length of the two land traces for each Bus from the ET to the transformer must be as short as possible(0.5in max length).

The two land traces for each Bus from the ET to the transformer must be balanced in length and width.

There should be no ground plane or power plane under transformer or the land traces connecting the transformer to the ET.

The center tap of the transformer primary must be connected to ground with a heavy short land trace.

The center tap of the transformer secondary should be left floating.

All the power and ground pins on the ET must be connected.

A 0.1uf capacitor should be connected from each power pin on the ET to ground.

9.0.0 PIN FUNCTIONAL DESCRIPTION

The NHi-ET I/ O pins are divided into 5 families:

General purpose signals
Host interface signals
I/ O bus interface signals
Mil Bus interface signals
Power

9.1.0 GENERAL PURPOSE SIGNALS

MRST_ L Master Reset (active low, input).
Initializes all registers and state machines. ET reads hardwire terminal address. Reset pulse width is 300ns min. The reset recovery time is 12us max after the rising edge of the reset pulse.

CLK_ H Terminal Clock from 10 Mhz oscillator (input).

9.2.0 HOST INTERFACE SIGNALS

H_ DAT (15: 0) Host Data bus (bi- directional).

H_ ADR (16/14: 1) Host Address bus (input).

HCS_ L Chip Select (active low, input).
Selects the NHi-ET. The falling edge of HCS_ L is used to latch the host address and indicates the start of a host memory cycle. The rising edge terminates the current cycle. During a host read- modify- write cycle. This signal must remain active from the beginning to the end of an access cycle. **NOTE: The host should not hold *HCS active for more than 5 microseconds, otherwise timing errors on the Mil-Std Data bus may occur.**

HWRL_ L Host Write Lower Byte (active low, input).

HWRH_ L Host Write Upper Byte (active low, input).

HRD_ L Host Read (active low, input).

DACK_ L Host Data Transfer Acknowledge (active low, open drain output, 5K internal pull up).
Indicates to the host that a data transfer has been completed. When the host reads data, it takes HCS_L low and the HRD_ L low. The ET will indicate that stable data is on the bus by outputting a low on DTACK_ L. When the Host writes data, it takes HCS_ L low and HWRL_ L and/ or HWRH_ L low. The ET then indicates that it has completed the write cycle by outputting a low on DTACK_ L.

IRQ_ L Host Interrupt Request (active low, open drain output, 5K internal pullup).
The IRQ_L will remain low until the Fifo is empty

INTACK_ L Host Interrupt Acknowledge (active low, input).
When HRD_ L= 0, INTACK_ L= 0, and HCS_ L= 1, an interrupt vector is popped from the FIFO, the IVR and AVR registers are updated, and the IVR is outputted onto both the lower and upper bytes of the host data bus, provided the INTPI_ L is low and the ET is in the RT mode.

INTPI_ L Interrupt Priority Input (active low, input).
This signal is used to daisy chain interrupt requests on the host bus. This signal must be active for the ET to output an interrupt vector.

INTPO_L_DSC Interrupt Priority Output, Disconnect Signal (output).
This pin has 2 possible functions, depending on the M1760 bit in the RTC CONTROL register.
If M1760= 0, then the signal is used to daisy chain interrupt requests on the host bus. When the ET requests an interrupt, this signal is output high; otherwise, this signal is equal to INTPI_ L.
If M1760= 1, then the pin is set to "1" when the store is disconnected (see EXTERNAL TERMINAL ADDRESS BUFFER for details).

9.3.0 DISCRETE I/O BUS INTERFACE SIGNALS

I/ O_ RD_ L I/ O Read (active low, output).

I/ O_ WR_ L I/ O Write (active low, output).

I/ O_ ADR (2: 1) I/ O Address (outputs).
These three signals can be used to select 4 byte- wide input devices and 4 byte- wide output devices which reside on the I/ O Data bus.

I/ O_ DAT (7: 0) I/ O DATA bus (bi- directional).
This bus is used for messages that are mapped to I/O, discreet pulse message identifiers and setting the Hardwire RT address.

CMDS Command Strobe (active high, output; 100ns).
This strobe is used for two special I/ O operations. When the strobe is active during a write cycle (i. e., CMDS= 1, I/ O WR_ L= 0), valid commands or pulses appear on the I/ O bus (see the CMDO bit in CONTROL register for details). When the strobe is active during a read cycle (i. e., CMDS= 1, I/ O RD_ L= 0), the EXTERNAL TERMINAL ADDRESS buffer is accessed.

PLSCMD_ BUS_ JAM Pulse Command or Bus Jam (active high, output; 100ns).
RT MODE:
Depends on the value of the CMDO bit in the CONTROL register. If CMDO= 0, then a pulse is issued whenever a bus message accesses a data table with PULSE (3: 0)= 14 (decimal) in its tag word. If CMDO= 1, then a pulse is issued whenever a valid broadcast command is received.
Note: The NTAG bit in the CONTROL register must be 0 to get a pulse output.

BC MODE:
This pin will go high and the BC will halt if a bus jam occurs. The CPU must then intervene to allow the BC to continue processing the frame (See configuration register 2 for details).

MDCDRST Mode Command Reset Pulse (active high, 400 nS pulse, output).
Pulsed high whenever the mode command "Reset" is received by the RT. Terminal must be in the RT mode for this pulse to be outputted.
mode.

SSF_ TF Subsystem Flag, Terminal Flag (active high, input).
Sets either the Subsystem Flag bit or the Terminal Flag bit in the STATUS register. The SSF_ TF bit in the CONTROL register determines which status bit will be set by this input (see CONTROL register for details).

9.4 MIL-BUS INTERFACE SIGNALS

BUS_ A, BUS_ A_ L BUS A signals (bi- directional).
Connected to a bus coupling transformer.

BUS_ B, BUS_ B_ L BUS B signals (bi- directional).
Connected to a bus coupling transformer.

TXINH_ A BUS A INHIBIT (input).
A logic high Inhibits the bus A transmitter.

TXINH_ B BUS B INHIBIT (input).
A logic high Inhibits the bus B transmitter.

10.0.0 ELECTRICAL CHARACTERISTICS

10.1.0 ABSOLUTE MAXIMUM RATINGS

PARAMETER	SYMBOL	MIN	MAX	UNITS	NOTE
SUPPLY VOLTAGE	Vcc	-0.3	+7.0	Volts	
INPUT VOLTAGE	Vin	-0.3	+6.7	Volts	1
INPUT CURRENT	Iin	-10		Microamps	2
INPUT ZAPPING	Vzap	2000		Volts	3
LATCH-UP TRIGGER	ILatch		200	Milliamps	4
THERMAL RESISTANCE	Tjc		4	DegC/W	
STORAGE TEMP.	Tstorage	-65	+150	Deg C	
LEAD TEMP.	TL		+300	Deg C	

Note 1: VCC referenced to ground
Note 2: Does not include current through internal 64K ohm pull- up/ down resistors.
Note 3: As defined for ESDS in Method 3015 0f MIL- STD- 883.
Note 4: The latch- up triggering current is the maximum current that will not cause latch- up on an I/ O BUFFER.

10.2.0 OPERATING CONDITIONS

PARAMETER	SYMBOL	MIN	MAX	UNITS
SUPPLY VOLTAGE	Vcc	4.7	5.5	Volts
STANDBY CURRENT	Iccstby		90	Milliamps
100% XMT CURRENT	Icc100		675	Milliamps
BUS LEVEL	Vpp	7.1		Volts
STUB LEVEL	Vpp	20	27	Volts
CASE TEMPERATURE	TC	-55	+125	Deg C

10.3.0 I/O TYPES & DESCRIPTIONS

I/O TYPES	DESCRIPTION
1	BI-DIRECTIONAL 3 STATE BUFFER 64K PULL UP
2	BI-DIRECTIONAL STATE BUFFER 64K PULL UP
3	TOTEM POLE OUTPUT BUFFER
4	TOTEM POLE OUTPUT BUFFER
5	OPEN DRAIN OUTPUT BUFFER
6	3 STATE OUTPUT BUFFER
7	INPUT BUFFER 64K PULL UP
8	INPUT BUFFER 64K PULL DOWN
9	INPUT BUFFER

I/O TYPE	SIGNAL NAME	I/O TYPE	SIGNAL NAME
1	H_DAT(15:0)	7	HWRH_L
2	I/O_DAT(7:0)	7	HWRL_L
3	DSC_INTPO_L	7	MRST_L
3	CMDS	7	INTPI_L
4	MDCDRST	7	INTACK_L
4	PLSCMD	7	SSF_TF
4	I/O_RD_L	7	H_ADR(14:1)
4	I/O_WR_L	7	HCS_L
5	DTACK_L	8	CLK10
5	IRQ_L	9	TXINH_A
6	I/O_ADR(2:1)	9	TXINH_B
7	HRD_L		

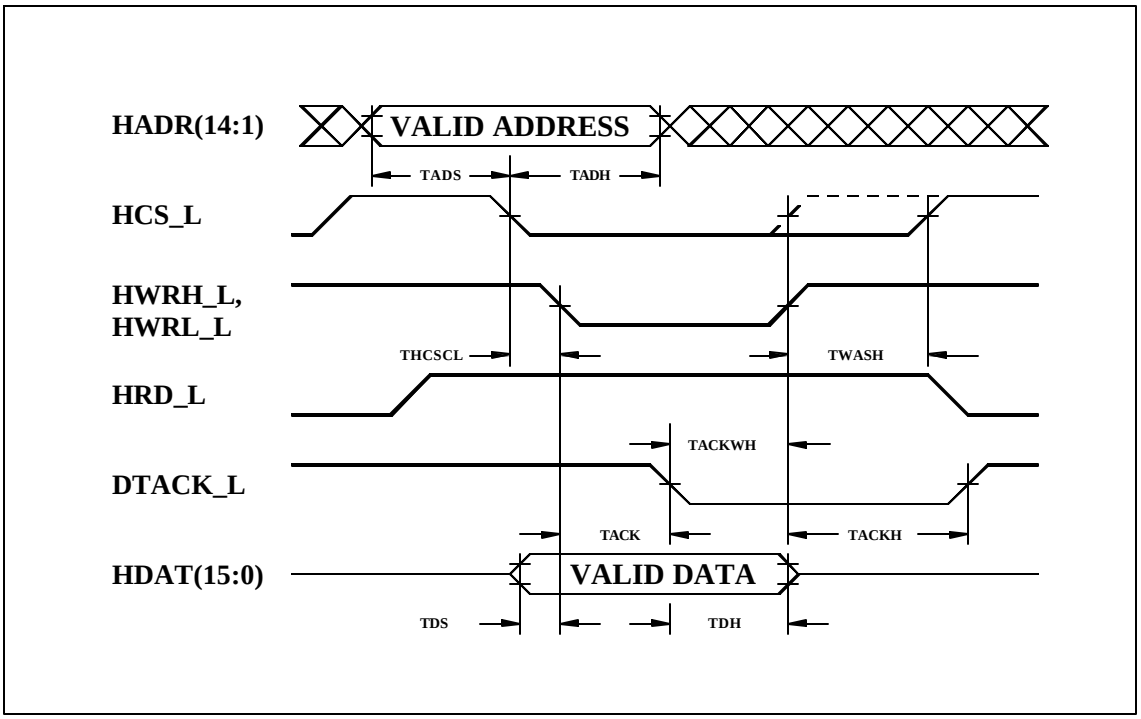
10.4.0 I/O ELECTRICAL CHARACTERISTICS

PARAMETER	I/O TYPE	CONDITION	MIN	MAX	UNITS
INPUT LOW VOLT	1,2,7,8,9			0.8	VOLTS
INPUT HIGH VOLT	1,2,7,8,9		2.0		VOLTS
OUTPUT LOW VOLT	1	IOL < 8.0 ma		0.4	VOLTS
	2	IOL < 4.0 ma		0.4	VOLTS
	3	IOL < 4.0 ma		0.4	VOLTS
	4	IOL < 6.0 ma		0.4	VOLTS
	5	IOL < 16.0 ma		0.4	VOLTS
	6	IOL < 8.0 ma		0.4	VOLTS
OUTPUT HIGH VOLT	1	IOH > -8.0 ma	2.4		VOLTS
	2	IOH > -4.0 ma	2.4		VOLTS
	3	IOH > -4.0 ma	2.4		VOLTS
	4	IOH > -6.0 ma	2.4		VOLTS
	6	IOH > - 8.0 ma	2.4		VOLTS
INPUT CAPACITANCE				10	PF

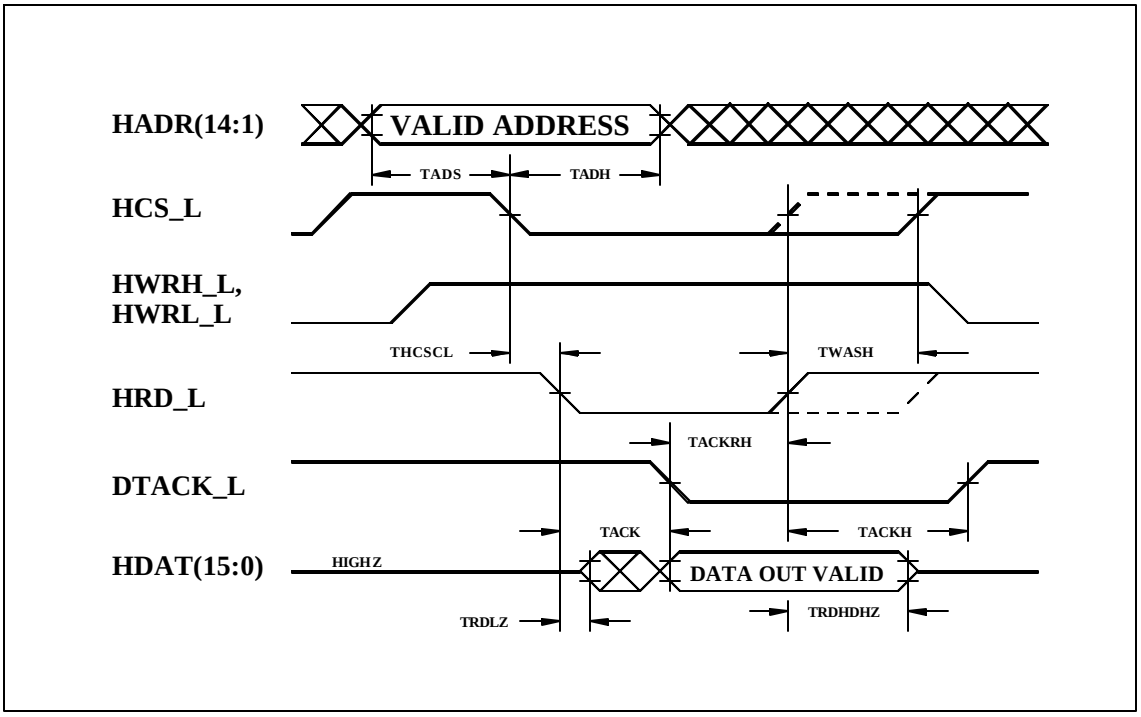
11.0.0 TIMING DIAGRAMS

The following diagrams and notes describe the timing of the address, data, and control lines.

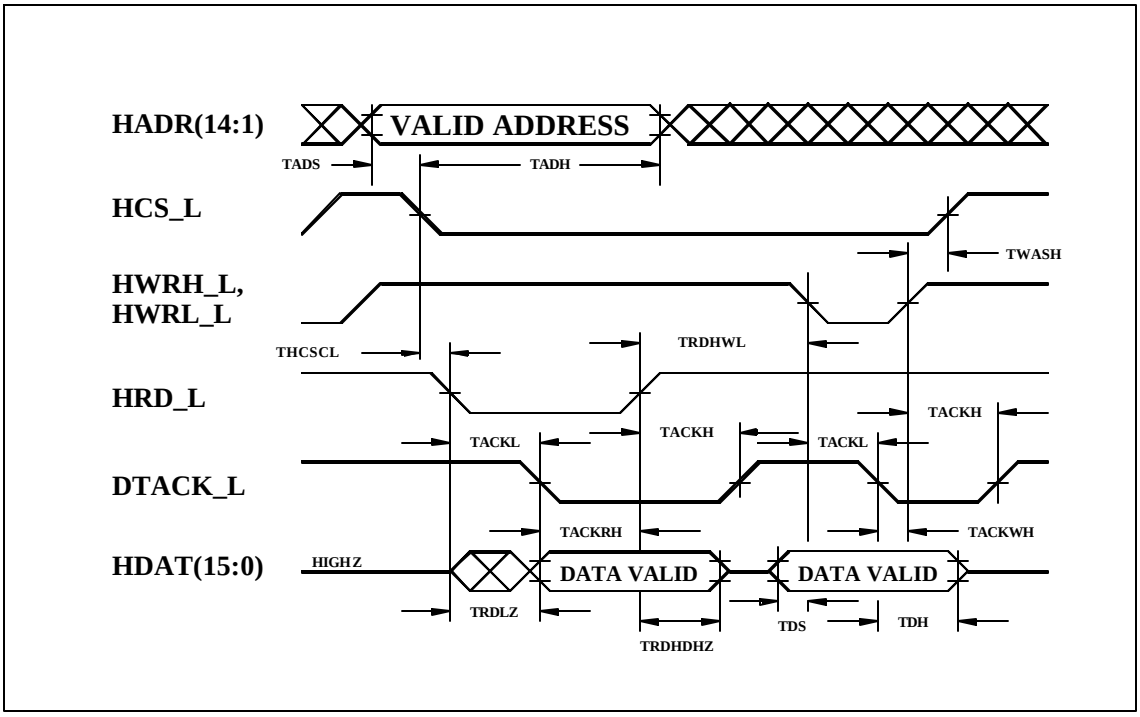
11.0.1 HOST WRITE CYCLE



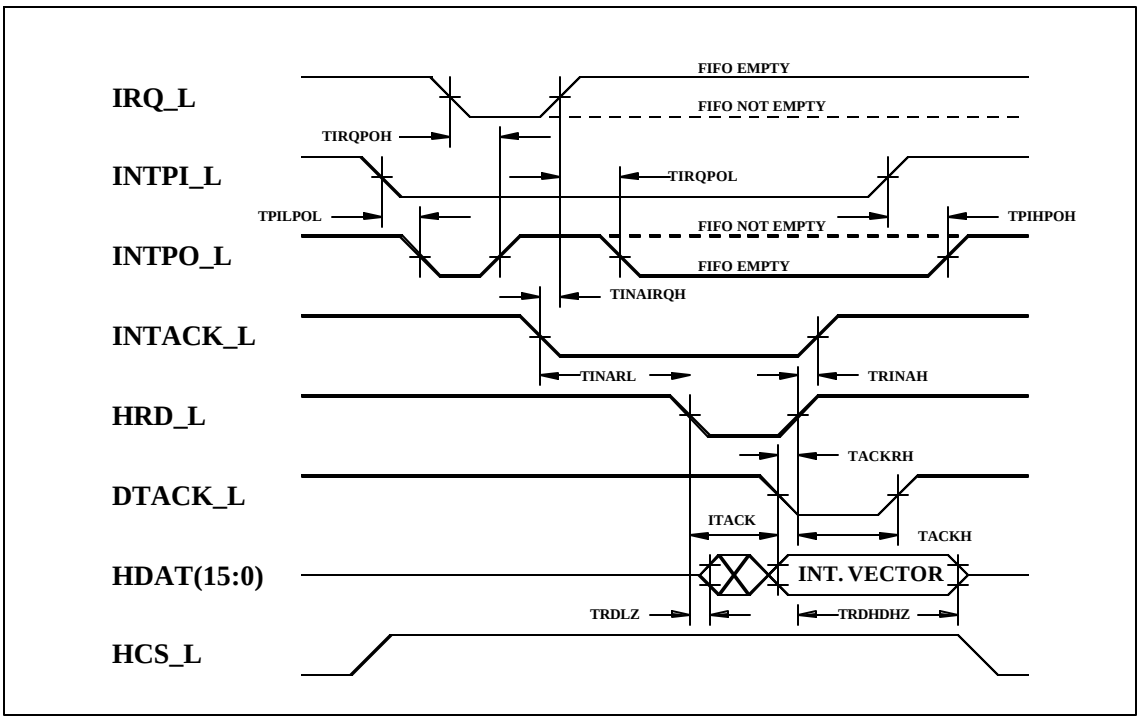
11.0.2 HOST READ CYCLE



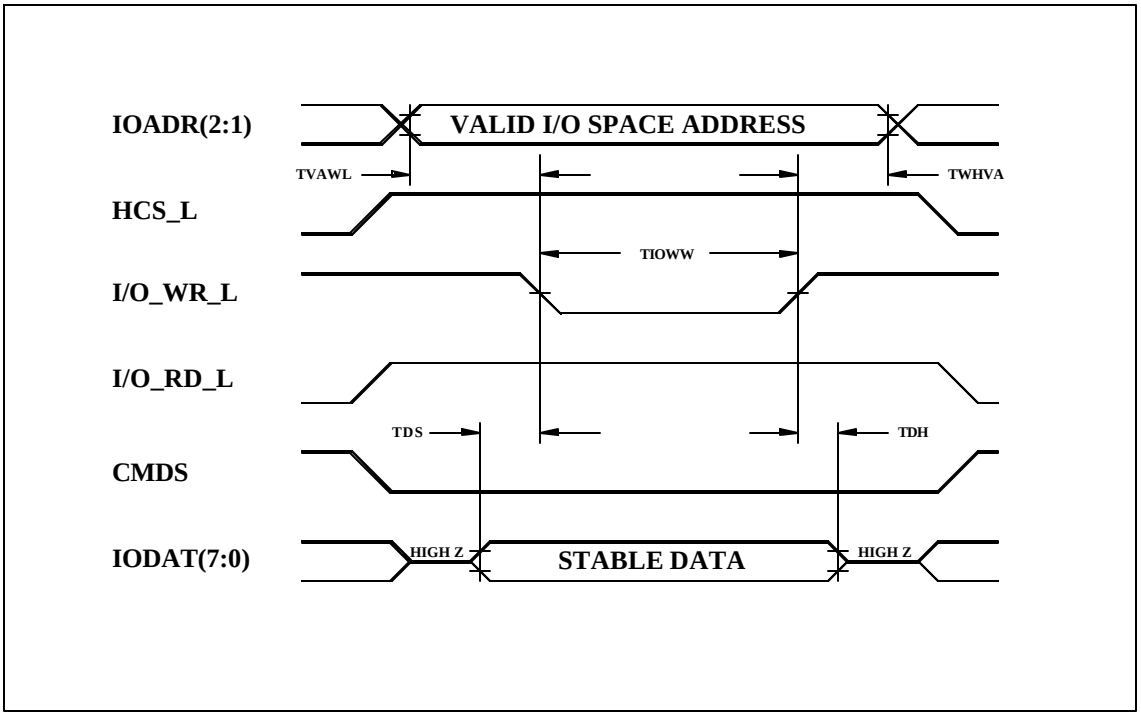
11.0.3 HOST READ- MODIFY- WRITE CYCLE



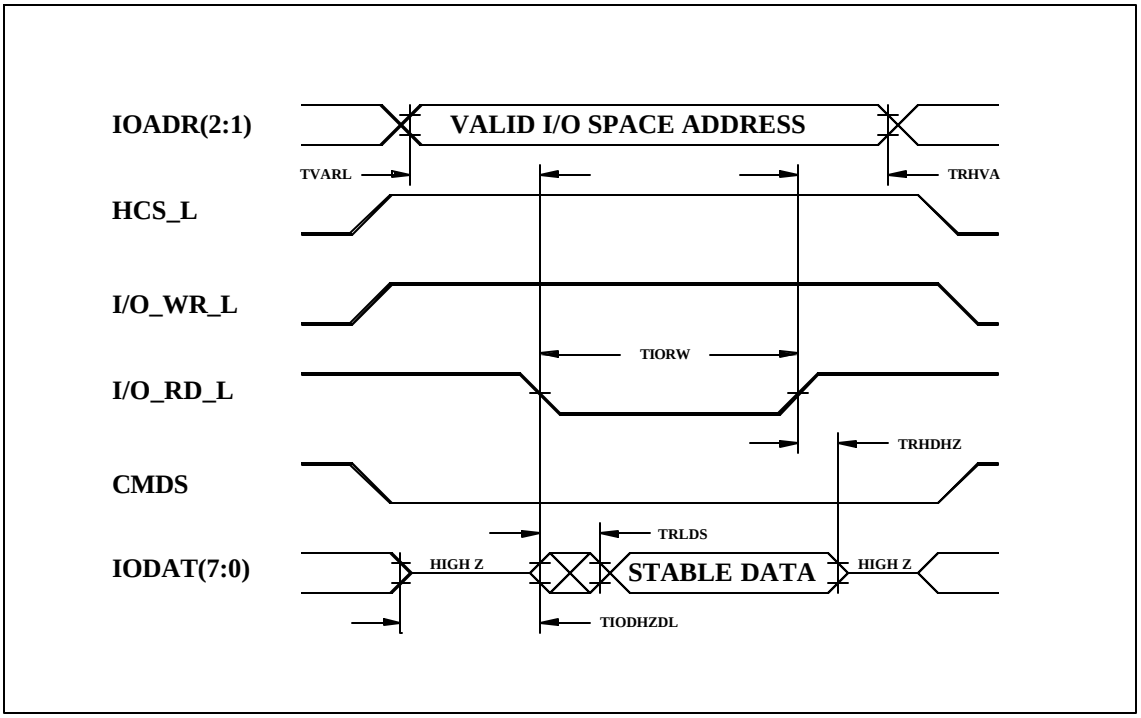
11.0.4 RT HARDWARE INTERRUPT ACKNOWLEDGE CYCLE



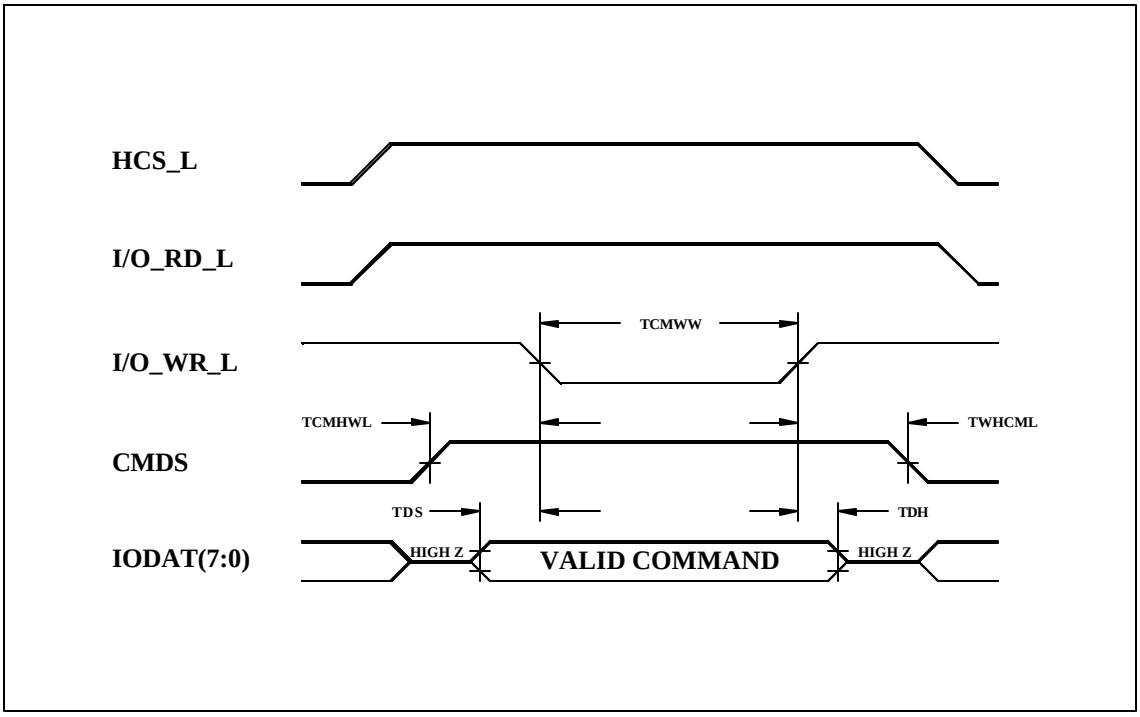
11.0.5 I/O WRITE CYCLE



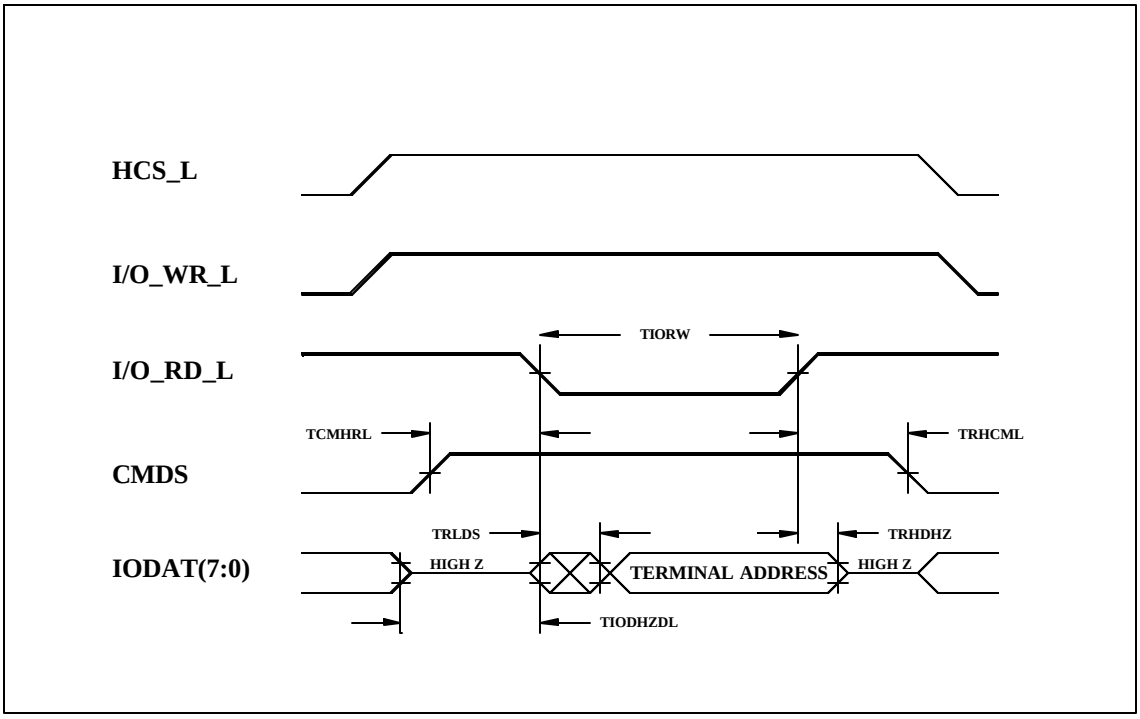
11.0.6 I/O READ CYCLE



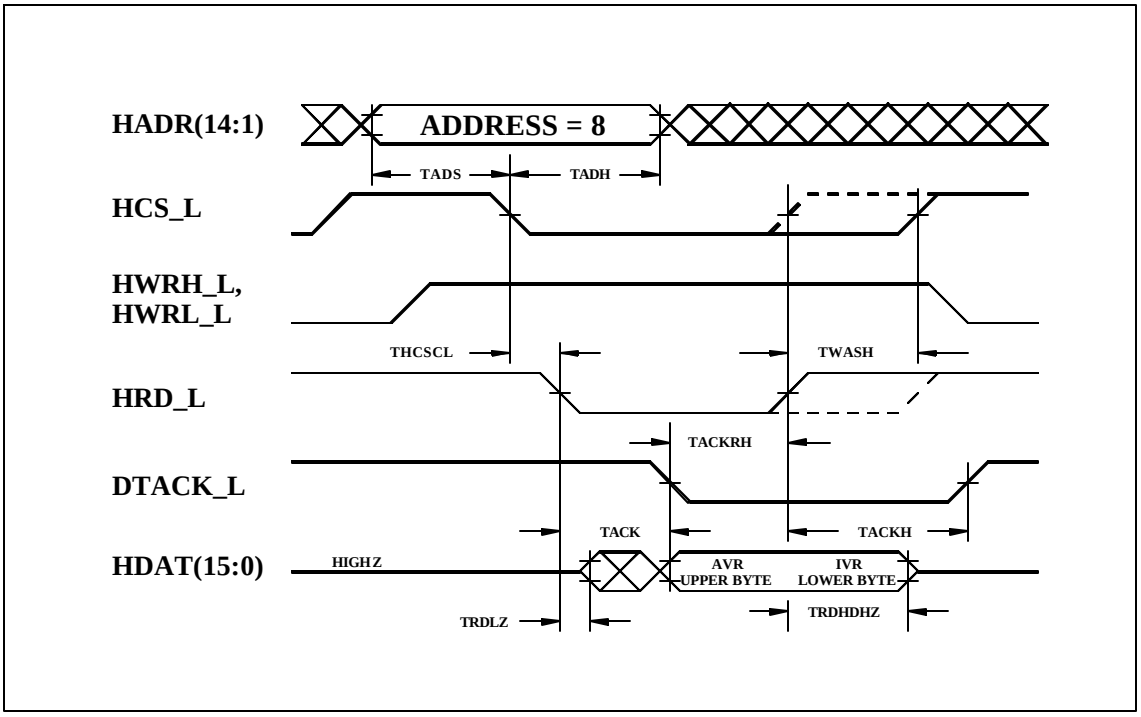
11.0.7 COMMAND WRITE CYCLE



11.0.8 TERMINAL ADDRESS READ CYCLE



11.0.9 SOFTWARE INTERRUPT ACKNOWLEDGE CYCLE



11.0.10 TIMING NOTES

The address is latched by the NHi- ET on the high- to- low transition of the *HCS line. TADS, TADH, and TASLC are referenced to the high- to- low transition of *HCS.

TACK is a function of the contending access performed by the NHi-ET (see host access table).

The low- to- high transition of HRD_L or I/ O RD_L or HCS_L terminates the read cycle.

The low- to- high transition of HWRH_L or HWRL_L or I/ O WR_L or HCS_L terminates the write cycle.

The DTACK_I line is tri- stated after delay TACKH. Its rise time is a function of the internal 5K ohm pull- up resistor and the external load.

While INTACK_L is low, INTPO_L will be affected by changes in IRQ_L.

ITACK starts after the falling edge of HRD_L and INTACK_L..

11.1.0 TIMING PARAMETER TABLES

11.1.1 HOST READ, WRITE, READ- MODIFY- WRITE TABLE and SOFTWARE INTERRUPT ACKNOWLEDGE

SYMBOL	PARAMETER	MIN(ns)	Max(ns)
TADS	ADDRESS SETUP TIME	0	-
TADH	ADDRESS HOLD TIME	200	-
THCSCL	HCS_L LOW TO COMMAND LOW	0	-
TACKWH	DATA ACKNOWLEDGE LOW TO WRITE HIGH	0	-
TWASH	HWRH,L_L HIGH TO HCS_L HIGH	0	-
TACKH	END OF CYCLE TO DATA ACKNOWLEDGE HIGH	0	30
TACKL(1)	NO CONTENTION	0	650
TACKL(2)	WITH CONTENTION	0	1500
TACKL(3)	WORST CASE ; ONCE AT START OF MESSAGE	0	3200
TDS	DATA SETUP TIME	0	75
TDH	DATA HOLD TIME	0	-
TACHRH	DATA ACKNOWLEDGE LOW TO READ HIGH	0	-
TRDLZ	HRD_L LOW TO DATA LOW Z	0	20
TRDHDHZ	HRD_L HIGH TO DATA HIGH Z	0	30
TRDHWL	HRD_L HIGH TO WRITE LOW	30	-

11.1.2 I/ O READ and TERMINAL ADDRESS READ TABLE

SYMBOL	PARAMETER	MIN(ns)	Max(ns)
TVARL	VALID ADDRESS TO I/O_RD_L LOW	50	-
TRHVA	ADDRESS VALID AFTER I/O_RD_L HIGH	50	-
TIORW	I/O_RD_L PULSE WIDTH	190	210
TRHDHZ	I/O_RD_L HIGH TO DATA HIGH Z	0	200
TRHCML	I/O_RD_L HIGH TO CMDS LOW	100	-
TCMHRL	CMDS HIGH TO I/O_RD_L LOW	100	-
TIODHZDL	I/O DATA BUS HIGH Z TO DATA ON BUS	100	-
TRLDS	I/O_RD_L LOW TO DATA STABLE	-	80

11.1.3 I/ O WRITE and COMMAND WRITE TABLE

SYMBOL	PARAMETER	MIN(ns)	Max(ns)
TVAWL	VALID ADDRESS TO I/O_WR_L LOW	50	-
TWHVA	ADDRESS VALID AFTER I/O_WR_L HIGH	50	-
TIOWW	I/O_WR_L PULSE WIDTH	90	110
TDS	DATA SETUP TIME	25	50
TDH	DATA HOLD TIME	40	80
TCMWW	COMMAND WRITE PULSE WIDTH	290	310
TCMHWL	CMDS HIGH TO I/O_WR_L LOW	40	-
TWHCML	I/O_WR_L HIGH TO CMDS LOW	100	-

11.1.4 RT HARDWARE INTERRUPT ACKNOWLEDGE CYCLE TABLE

SYMBOL	PARAMETER	MIN(ns)	Max(ns)
TIRQPOH	IRQ_L LOW TO INTPO_L HIGH	-	20
TPILPOL	INTPI_L LOW TO INTPO_L LOW	-	40
TPIHPOH	INTPI_L HIGH TO INTPO_L HIGH	-	40
TINAIRQH	INTACK_L LOW TO IRQ_L HIGH	0	200
TINAIRQL	INTACK_L HIGH TO NEXT IRQ_L LOW	0	200
TRINAH	HRD_L HUGH TO INTACK_L HIGH	0	-
ITACK	HRD_L LOW TO DTACK_L LOW	300	400
TACKRH	DTACK_L LOW TO HRD_L HIGH	0	-
TRDLZ	HRD_L LOW TO DATA IN LOW Z	0	20
TRDHDHZ	HRD_L HIGH TO DATA IN HIGH Z	0	30
TACKH	END OF CYCLE TO DTACK_L HIGH	0	30
TIRQPOL	IRQ_L HIGH TO INTPO_L LOW	10	20
TINARL	INTACK_L LOW TO HRD_L LOW	0	30

12.0.0 PIN FUNCTION TABLE

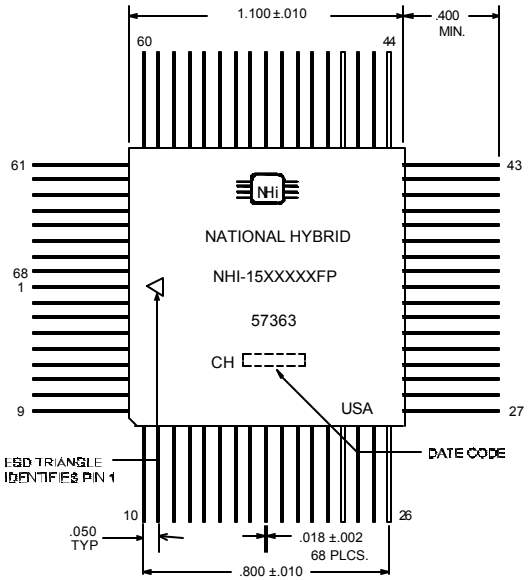
12.0.1 UNIVERSAL PIN FUNCTIONS FLAT PACK AND PIN GRID ARRAY PINS

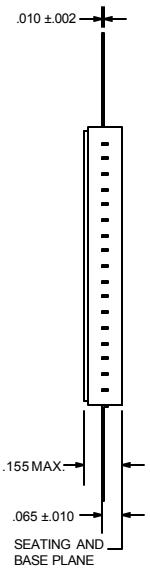
QFP	PGA	FUNCTION	QFP	PGA	FUNCTION
1	L10	I/O_DAT1	35	D2	H_ADR4
2	H10	I/O_DAT2	36	C1	H_ADR5
3	J10	I/O_DAT3	37	B1	H_ADR6
4	K10	I/O_DAT4	38	A2	H_ADR7
5	K9	I/O_DAT 5	39	K5	+5V
6	L9	I/O_DAT6	40	L5	+5V
7	L8	I/O_DAT7	41	J1	GND
8	G11	HCS_L	42	K1	GND
9	H11	SSF_TF	43	B2	GND
10	G10	MRST_L	44	A3	H_ADR8
11	J11	CLK10	45	B4	H_ADR9
12	K8	I/O_ADR1	46	A4	H_ADR10
13	L7	I/O_ADR2	47	B5	H_ADR11
14	K7	INTPO_L_DSC	48	A5	H_ADR12
15	L6	INTPI_L	49	B6	H_DAT0
16	K6	INTACK_L	50	A6	H_DAT1
17	G2	TXINH_A	51	A7	H_DAT2
18	B3	BUS_A	52	B7	H_DAT3
19	C2	BUS_A_L	53	A8	H_DAT4
20	C3	H_ADR13	54	A9	H_DAT5
21	L3	CMDS	55	A10	H_DAT6
22	K4	TXINH_B	56	B8	H_DAT7
23	J2	BUS_B	57	B9	H_DAT8
24	H1	BUS_B_L	58	B10	H_DAT9
25	K3	H_ADR14	59	B11	H_DAT10
26	L4	I/O_RD_L	60	C11	H_DAT11
27	H2	I/O_WR_L	61	C10	H_ADR1
28	K2	MDCDRST	62	D11	HRD_L
29	F2	PLSCMD/B_JAM	63	E11	HWRL_L
30	F1	IRQ_L	64	F11	HWRH_L
31	E1	DTACK_L	65	D10	H_DAT13
32	G1	H_DAT12	66	E10	H_DAT14
33	E2	H_ADR2	67	F10	H_DAT15
34	D1	H_ADR3	68	K11	I/O_DAT0
-	-	-	-	L2	N/C

Note: See individual parts listing for special pin functions.

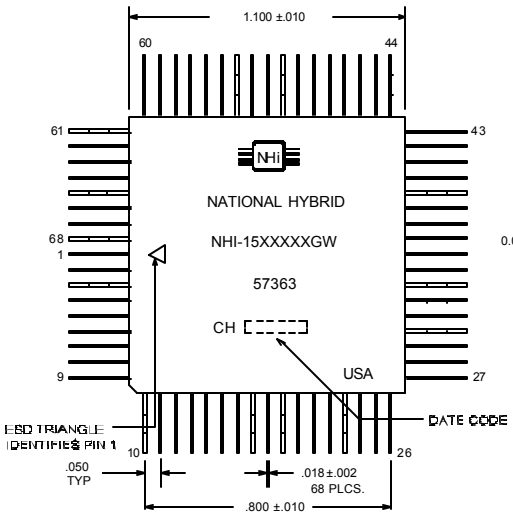
12.1.0 GENERIC PACKAGE OUTLINE DRAWINGS

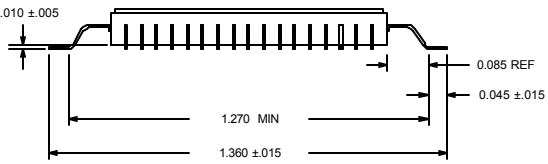
12.1.1 QUAD FLAT PACK UNFORMED LEAD



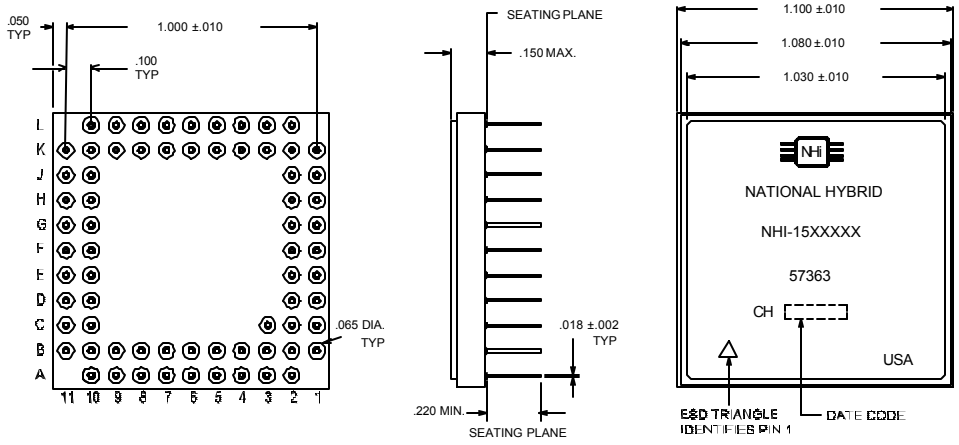


12.1.2 QUAD FLAT PACK GULL WING LEADS

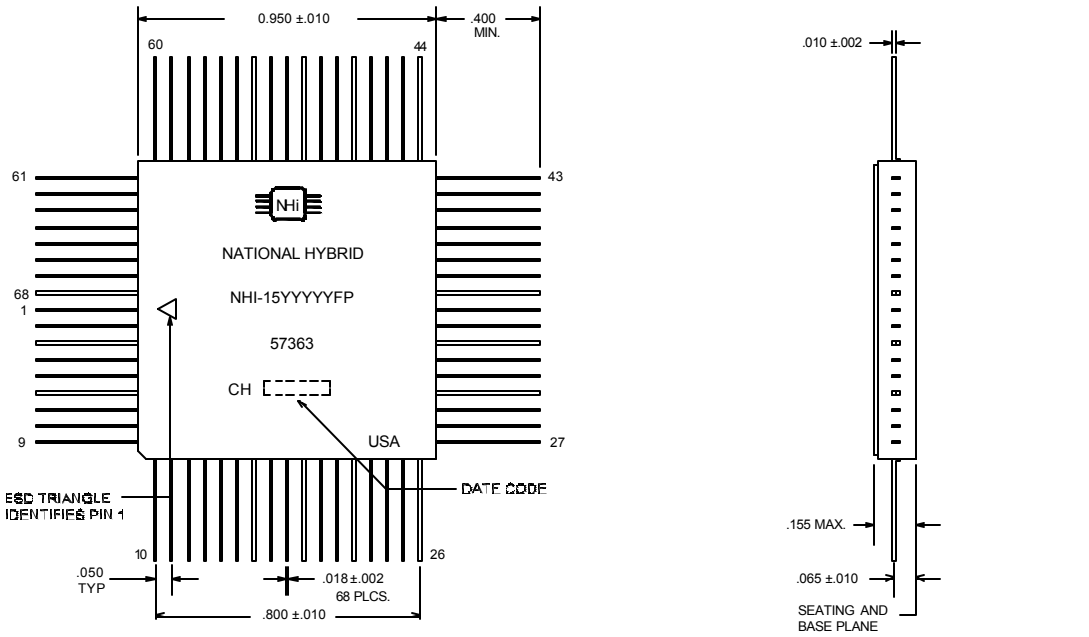




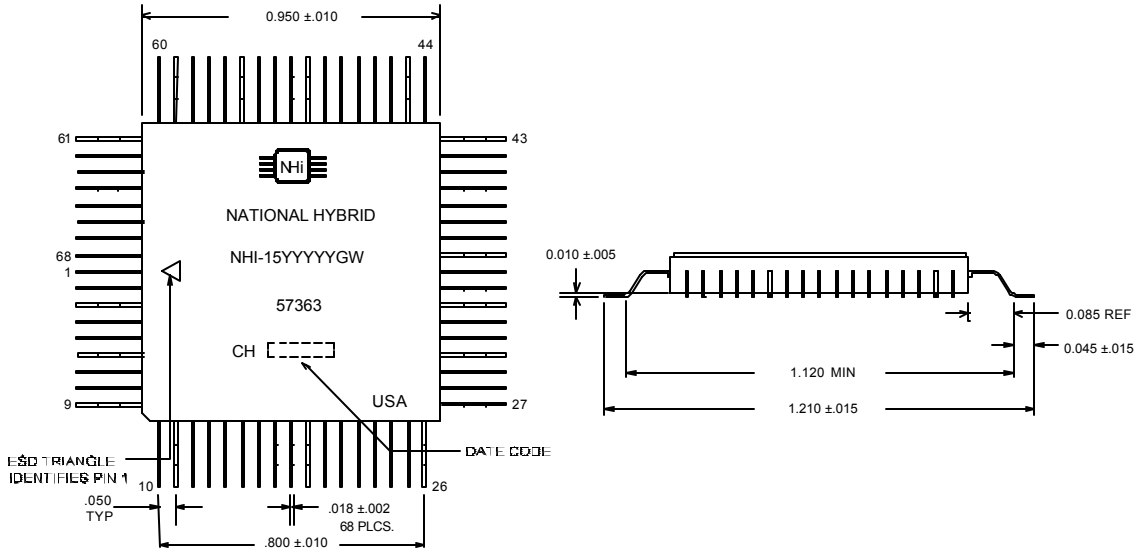
12.1.3 PIN GRID ARRAY



12.1.4 MICRO QUAD FLAT PACK UNFORMED LEADS



12.1.5 MICRO QUAD FLAT PACK GULL WING LEADS

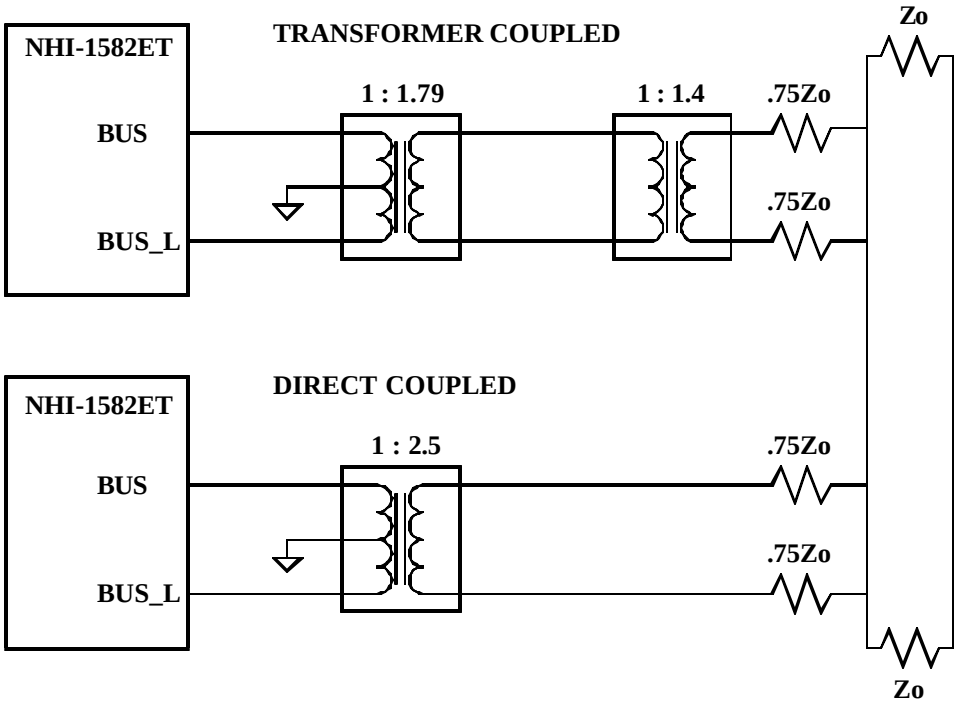


13.0.0 MATING TRANSFORMER REFERENCE

All the NHi-ET requires a coupling transformer with a turns ratio of 1: 2.5 for Direct Coupling, and a turns ratio of 1: 1.79 for Transformer Coupling to the Mil- Std Data Bus. Please contact Beta Transformer (www.bttc-beta.com) for a recommended transformer.

The center tap on the NHi-ET side of the coupling transformer must be grounded. The center tap on the bus side of the coupling transformer should be left floating.

The figure shows a typical transformer connection.

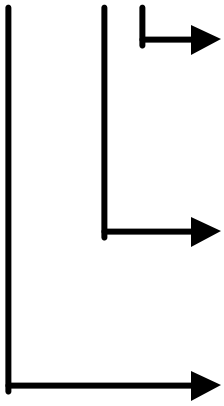


14.0.0 ORDERING INFORMATION

Unless otherwise specified, all terminals contain the following standard features:

- **Dual Redundant +5 Volt Only Operation**
- **Bus Controller, Bus Monitor, Remote Terminal**
- **NHi Monolithic Transceivers**
- **16K Word Internal Ram**
- **Multi Protocol Compliant**
- **Trapezoidal Output Waveform**
- **Package Outline 1.1 x 1.1 inches**
- **Package Pins Defined in Pin Function Table**

NHi-1582ETGW/ 883



Grade

883	Compliant to MIL- PRF- 38534 Class H
M	MIL- PRF- 38534 Table VIII Device Screening.
T	Industrial Grade, MIL Temp: -55 to +125° C
Blank	Industrial, -40 to +85° C

Package

Blank	Plug- In
GW	Gull Wing lead formed surface mount flatpack
FP	FlatPack

Device

82ET	See Standard Features List.
83ET	Sinewave Output Waveform.
98ET	External Time Tag Clock Input. External Time Tag Input replaces I/O_ADR2.
99ET	Sinewave Output Waveform, Time Tag Clock Input. External Time Tag Input replaces I/O_ADR2.
103ET	External Address Latch Input. External Address Latch Input replaces I/O_ADR1.
104ET	Sinewave Output Waveform, External Address Latch Input. External Address Latch Input replaces I/O_ADR1.
175ET	No Internal Transceivers. H_ADR13 Input replaces I/O_ADR1. H_ADR14 Input replaces I/O_ADR2. RXA Input replaces TXINH_A RXA_L input replaces H_ADR13 TXA output replaces BUS_A TXA_L output replaces BUS_A_L RXB Input replaces TXINH_B RXB_L input replaces H_ADR14 TXB output replaces BUS_B TXB_L output replaces BUS_B_L

Ordering Information Continued

Unless otherwise specified, all terminals contain the following features:

- **Dual Redundant +5 Volt Only Operation**
- **Bus Controller, Bus Monitor, Remote Terminal**
- **NHi Monolithic Transceivers**
- **16K Word Internal Ram**
- **Multi Protocol Compliant**
- **Trapezoidal Output Waveform**
- **Package Outline 1.1 x 1.1 inches**
- **Package Pins Defined in Pin Function Table**

282ET	64K Word Internal Ram. H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2.
283ET	Sinewave Output Waveform, 64K Word Internal Ram. H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2.
301ET	Package Outline 0.95 x .095 inches.
375ET	No Internal Transceivers- Use external transceivers; Package Outline 0.95 x .095 inches. TXA_H replaces I/O_ADR1. TXA_L replaces I/O_ADR2. RXA_H replaces BUS_A. RXA_L replaces TXINH_A. TXB_H replaces BUS_B. TXB_L replaces BUS_B_L. RXB_H replaces BUS_A_L. RXB_L replaces TXINH_B.
382ET	64K Word Internal Ram. Package Outline 0.95 x 0.95 inches H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2.
383ET	64K Word Internal Ram, Sinewave Output Waveform. Package Outline 0.95 x 0.95 inches. H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2.
398ET	64K Word Internal Ram Package Outline 0.95 x 0.95 inches. H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2. External Time Tag Input replaces I/O_WR_L.
399ET	64K Word Internal Ram, Sinewave Output Waveform. Package Outline 0.95 x 0.95 inches. H_ADR15 replaces I/O_ADR1. H_ADR16 replaces I/O_ADR2. External Time Tag Input replaces I/O_WR_L.

Ordering Information Continued

Unless otherwise specified, all terminals contain the following features:

- **Dual Redundant +5 Volt Only Operation**
- **Bus Controller, Bus Monitor, Remote Terminal**
- **NHi Monolithic Transceivers**
- **16K Word Internal Ram**
- **Multi Protocol Compliant**
- **Trapezoidal Output Waveform**
- **Package Outline 1.1 x 1.1 inches**
- **Package Pins Defined in Pin Function Table**

975ET No Internal Transceivers. Radiation Tolerant Terminal, Consult Factory for Details.
H_ADR13 Input replaces I/O_ADR1.
H_ADR14 Input replaces I/O_ADR2.
RXA Input replaces TXINH_A
RXA_L input replaces H_ADR13
TXA output replaces BUS_A
TXA_L output replaces BUS_A_L
RXB Input replaces TXINH_B
RXB_L input replaces H_ADR14
TXB output replaces BUS_B
TXB_L output replaces BUS_B_L

982ET Radiation Tolerant Terminal. Consult Factory for Details.

983ET Sinewave Output Waveform, Radiation Tolerant Terminal. Consult Factory for Details.

**** SMD Listing: DESC Drawing# 5962- 95558**
See QML- 38534 for NHi Qualification under Mil- PRF- 38534

DATA DEVICE CORPORATION
REGISTERED TO:
ISO 9001:2008, AS9100C:2009-01
EN9100:2009, JIS Q9100:2009
FILE NO. 10001296 ASH09

The first choice for more than 45 years—DDC

DDC is the world leader in the design and manufacture of high reliability data interface products, motion control, and solid-state power controllers for aerospace, defense, and industrial automation.

Inside the U.S. - Call Toll-Free 1-800-DDC-5757

Headquarters and Main Plant
105 Wilbur Place, Bohemia, NY 11716-2426
Tel: (631) 567-5600 Fax: (631) 567-7358
Toll-Free, Customer Service: 1-800-DDC-5757

Web site: www.ddc-web.com

Outside the U.S. - Call 1-631-567-5600

United Kingdom: DDC U.K., LTD
Mill Reef House, 9-14 Cheap Street, Newbury,
Berkshire RG14 5DD, England
Tel: +44 1635 811140 Fax: +44 1635 32264

France: DDC Electronique
10 Rue Carle-Hebert
92400 Courbevoie France
Tel: +33-1-41-16-3424 Fax: +33-1-41-16-3425

Germany: DDC Elektronik GmbH
Triebstrasse 3, D-80993 München, Germany
Tel: +49 (0) 89-15 00 12-11
Fax: +49 (0) 89-15 00 12-22

Japan: DDC Electronics K.K.
Dai-ichi Magami Bldg, 8F, 1-5, Koraku 1-chome,
Bunkyo-ku, Tokyo 112-0004, Japan
Tel: 81-3-3814-7688 Fax: 81-3-3814-7689
Web site: www.ddcjapan.co.jp

Asia: Data Device Corporation - RO Registered in Singapore
Blk-327 Hougang Ave 5 #05-164
Singapore 530327
Tel: +65 6489 4801