HOLT INC.
INTEGRATED CIRCUITS

HI-6300

MIL-STD-1553 / MIL-STD-1760
Protocol IP Core

February 2020

ADDITIONAL SUPPORTING DOCUMENTATION

Additional information is provided in the following:

- HI-1587 Datasheet (DS1587)
- AXI4-Lite IPIF v3.0 LogiCORETM IP Product Guide

Table of Contents

List of Figures

List of Tables

1. GENERAL DESCRIPTION

The HI-6300 MIL-STD-1553 IP Core provides a complete and compliant single or multi-function interface between a host processor and the Holt HI-1587 MIL-STD-1553 Bus Transceiver. HI-6300 includes Bus Controller (BC), Remote Terminal (RT), and Bus Monitor Terminal (MT) operating modes. Any combination of the contained 1553 modes can be enabled for concurrent operation.

The enabled terminals communicate with the MIL-STD-1553 buses through Holt's external dual bus transceiver, HI-1587 and an external MIL-STD-1553 transformer interface. The user allocates up to 64Kx22 of FPGA static RAM Blocks to suit application requirements. The IP Core presents the host with a simple memory mapped synchronous 16-bit parallel interface that is compatible with the Xilinx IPIF standard.

2. FEATURES

- Fully software compatible with Holt's existing hardware solutions: MAMBA^{TM} or HI-6130/31 families
- IP is based on fully validated IC solution
- Available DO-254 Certification Package supporting Design Assurance Level A
- Concurrent multi-terminal operation
- Synchronous Host Interface
- Built-in self-test feature
- Fully programmable Bus Controller with 28 op code instruction set
- Interface for up to 64K words static RAM with RAM Error Detection/Correction option.
- Independent time-tag counters for all terminals with 32-bit option for Bus Controller and 48-bit option for Monitor Terminal
- 64-Word Interrupt Log Buffer queues the most recent 32 interrupts.
- MIL-STD-1760 Boot mode to initialize RT with Busy Bit set without host intervention

3. BLOCK DIAGRAM

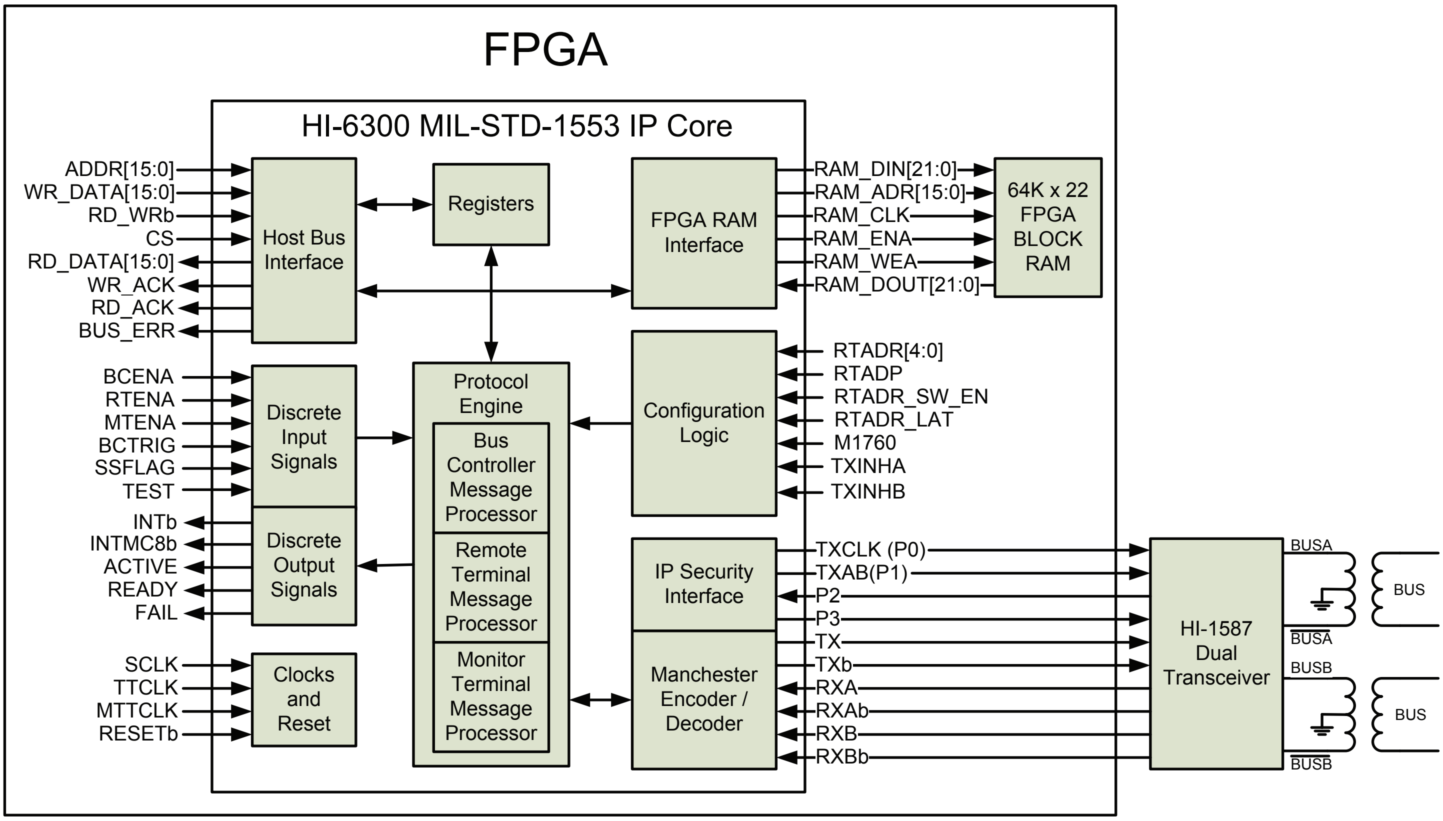


Figure 1. Block Diagram

4. FEATURE OVERVIEW

4.1. Bus Controller Operation

The HI-6300 IP Core is configurable to operate as a Bus Controller (BC). The BC is a programmable message-sequencing engine for control in MIL-STD-1553B applications. Programmed using a set of 28 instruction op codes, the BC greatly reduces the host's processing workload. The BC can optionally use a 16- or a 32-bit time base, clocked from a choice of six internally generated clocks, or an external time base clock. Special BC op codes manage all 32-bit time base functions.

The programmable HI-6300 Bus Controller autonomously supports multi-frame message scheduling, message retry schemes, storage of message data, asynchronous message insertion and status /error reporting to the host processor.

4.2. Remote Terminal Operation

The HI-6300 is configurable to operate as a Remote Terminal. The RT is modeled after the popular Holt MAMBA^{TM} Remote Terminal. RT message illegalization is achieved through a simple lookup table in shared RAM. The RT supports RAM buffer options include single, double and 2 circular buffer choices.

4.3. Monitor Terminal Operation

Message commands, terminal responses and message data are stored in internal RAM, using Simple Monitor Terminal (SMT) operation. The SMT records commands and data separately and can utilize 16- or 48-bit time tags with a range of clocking options.

4.4. Interrupts

Host interrupts can originate from the IP Core, including from any of the enabled terminal modes. A circular 64-word Interrupt Log Buffer retains interrupt information from the last 32 interrupts, while the hardware maintains a count of occurring interrupts since the previous host buffer service.

Hardware-assisted interrupt decoding provides quick identification of the interrupt source by terminal IP core: BC, RT, MT or the core. When a hardware interrupt occurs (e.g., Bus A Loopback Failure), a Pending Hardware Interrupt register bit explicitly identifies the interrupt source. For interrupts from BC, RT or MT, the three low-order bits in the same register identify the specific interrupt register (or registers) with pending interrupts: that is, the BC, RT or MT Pending Interrupt registers.

5. SIGNAL DESCRIPTIONS

Table 1. IP Core Host Interface Signals

Signal Name	Input/Output	Description
ADDR[15:0]	Inputs	Bus address for Read/Write operations.
WR_DATA[15:0]	Inputs	Write Data Bus - 16-bits wide.
$\overline{\text{RD\_WR}}$	Input	Read or Write bus cycle. High is a read. Low is a write.
CS	Input	Chip select bus for active Read/Write cycle. Asserts at the beginning of a valid cycle on the host interface. This signal, used in conjunction with $\overline{\text{RD\_WR}}$ signal.
RD_DATA[15:0]	Outputs	Read Data Bus - 16-bits wide. Read data will be valid at the rising edge of SCLK when RD_ACK is high.
WR_ACK	Output	Write Data Acknowledge - Acknowledges an IP write cycle and causes write control signal CS to de-assert.
BUS_ERR	Output	Module Error Asserted with WR_ACK or RD_ACK.

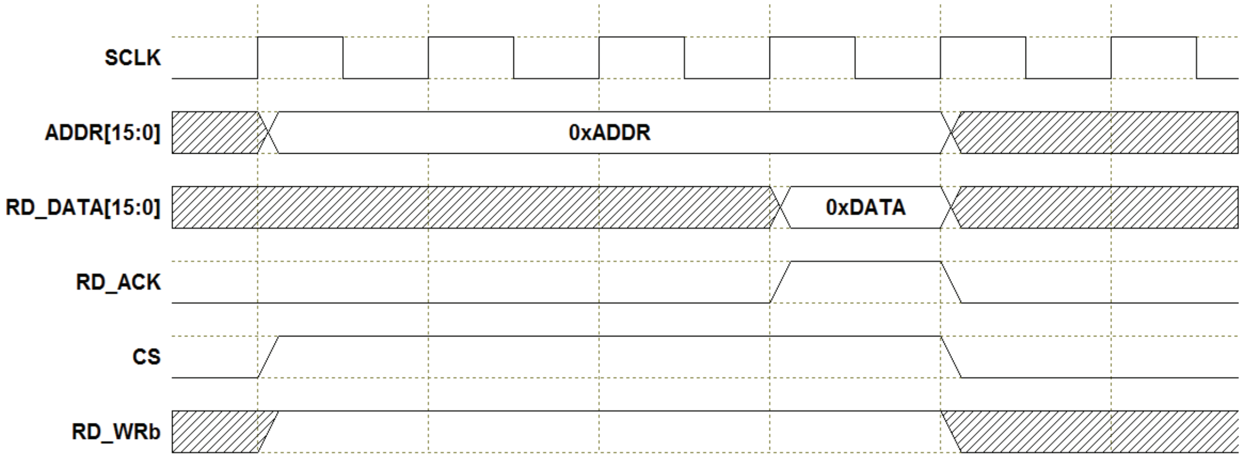


Figure 2. IP Host Interface Read Transaction

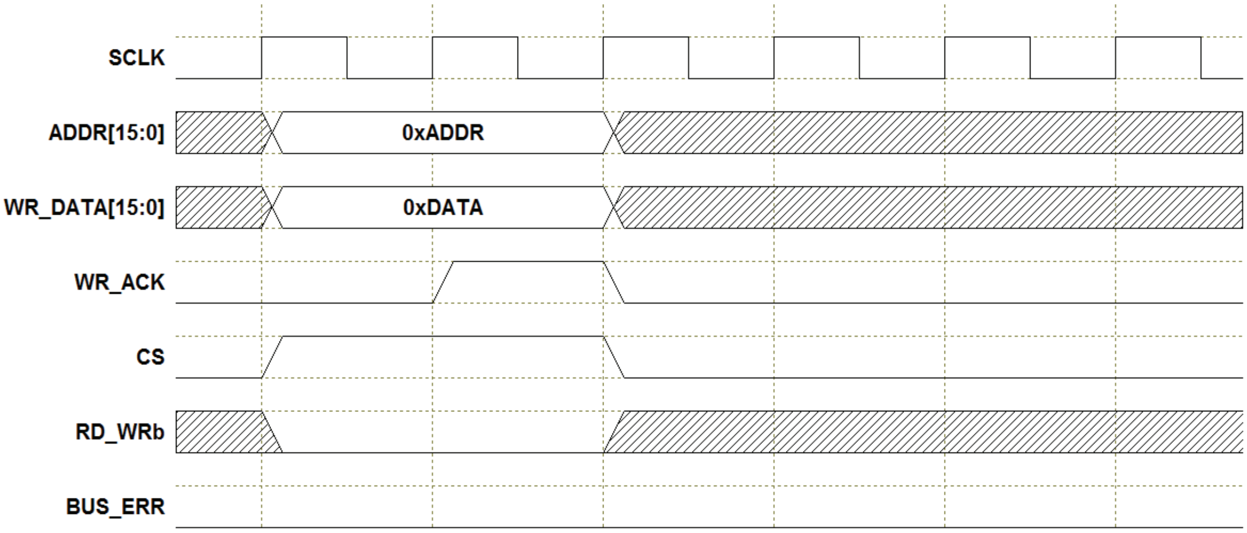


Figure 3. IP Host Interface Write Transaction

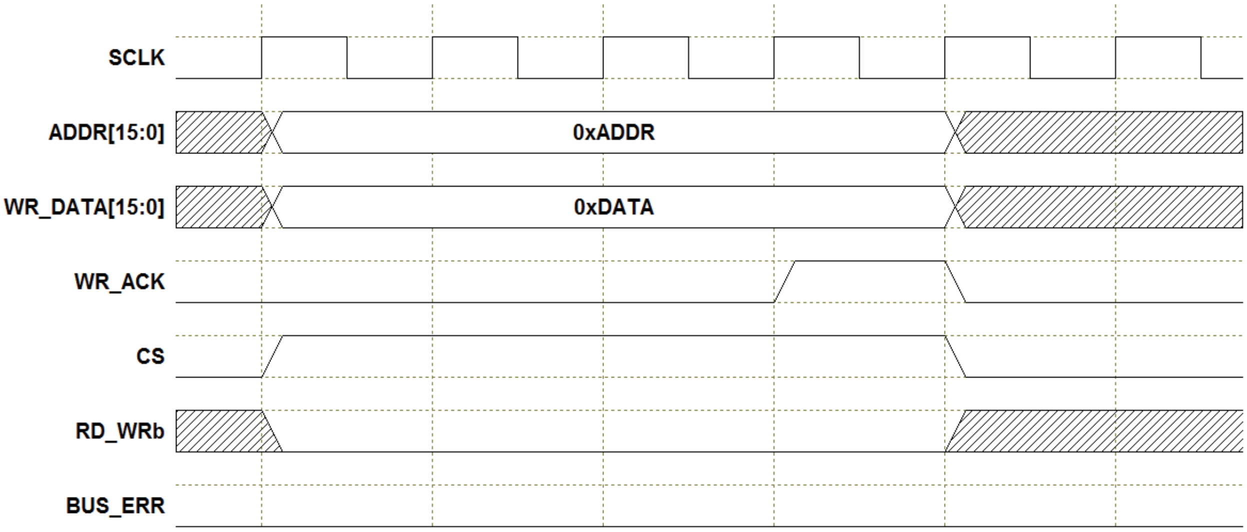


Figure 4. IP Host Interface Write Transaction (RT Descriptor Block Control Word)

Table 2. Clock and Reset Signals

Signal Name	Input/Output	Description
SCLK	Input	System clock input.
TTCLK	Inputs	Optional Time Tag Clock for Bus Controller and Remote Terminal.
MTTCLK	Input	Optional Time Tag Clock for Bus Monitor Terminal.
$\overline{\text{RESET}}$	Input	Master Reset − Active Low.

Table 3. IP Core Configuration Signals

Signal Name	Input/Output	Description
RTADR[4:0]	Inputs	RT Address Inputs
RTADP	Input	RT Address Parity
RTADR_SW_EN	Input	RT Address Software Programmable 0 = Disabled. RT address is not programmable by Host. 1 = Enabled. RT address is programmable by Host by writing “Remote Terminal Operational Status Register (0x0018)”. See RTADR_LAT and Table 4 below.
RTADR_LAT	Input	Latch RT Address 0 = Disable. RT address transparent with RTADR[4:0] and RTADP input signals. 1 = Enabled. RT address will be latched on rising edge of RTADR_LAT, $\overline{\text{RESET}}$, or RT soft reset. See RTADR_SW_EN and Table 4 below.
M1760	Input	Enable MIL-STD-1760 Mode. Come up as Remote Terminal Enabled with the busy bit set in the RT Status Word.
TXINHA	Input	Bus A Transmit Inhibited. This input is logically OR’ed with the corresponding bit in the “Master Configuration Register 1 (0x0000)” to enable or inhibit transmit on Bus A, affecting behavior for all enabled 1553 terminal modes.
TXINHB	Input	Bus B Transmit Inhibited. This input is logically OR’ed with the corresponding bit in the “Master Configuration Register 1 (0x0000)” to enable or inhibit transmit on Bus B, affecting behavior for all enabled 1553 terminal modes.

Table 4. RT_AD_LAT and RTA_SW_EN Truth Table

RTADR_LAT	RTADR_SW_EN	RT Operational Status Register (0x0018) Bit 9 (LOCK Bit)	Behavior
0	X	X	RT address transparent with RTADR[4:0] and RTADP input signals. Software writing to “Remote Terminal Operational Status Register (0x0018)” has no effect. LOCK Bit no effect.
1	0	X	RT address sampled on clock after rising edge of RTADR_LAT, on clock after rising edge of $\overline{\text{RESET}}$, or RT soft reset. Software writing to RT Address Bits in “Remote Terminal Operational Status Register (0x0018)” has no effect.
1	1	0	RT address sampled on clock after rising edge of RTADR_LAT, on clock after rising edge of $\overline{\text{RESET}}$, or RT soft reset. Software writing to RT Address Bits in “Remote Terminal Operational Status Register (0x0018)” will update the RT Address.
1	1	1	RT address sampled on clock after rising edge of RTADR_LAT, on clock after rising edge of $\overline{\text{RESET}}$, or RT soft reset. LOCK bit set; software writing to RT Address Bits in “Remote Terminal Operational Status Register (0x0018)” has no effect.

Table 5. IP Core Discrete Input Signals

Signal Name	Input/Output	Description
BCENA	Input	Bus Controller Enable.
RTENA	Input	Remote Terminal Enable.
MTENA	Input	Monitor Terminal Enable.
BCTRIG	Input	BC Trigger input, optional trigger for bus controller WTG Op Code.
SSFLAG	Input	Subsystem Flag for RT.
TEST	Input	Test Enable Input. The host asserts this signal to perform RAM self-test and loopback tests.

Table 6. IP Core Discrete Output Signals

Signal Name	Input/Output	Description
$\overline{\text{INT}}$	Output	Interrupt − Active Low. This signal is asserted each time an enabled interrupt event occurs.
$\overline{\text{INTMC8}}$	Output	Interrupt, Remote Terminal Mode code 8 − Active Low. Remote Terminal “Reset RT” mode command (MC8) received. This active low output is asserted at Status Word completion when the RT received a “Reset Remote Terminal” MC8 command.
ACTIVE	Output	Active message processing. This signal is high while an enabled BC or RT in the IP core is processing a 1553 message.
READY	Output	Module Ready for Configuration and Processing.
FAIL	Output	Module Failed IP Security Authorization.

Table 7. IP Core External RAM Interface Signals

Signal Name	Input/Output	Description
RAM_CLK	Output	Clock to RAM.
RAM_DIN[21:0]	Output	Write data to RAM.
RAM_ADR[15:0]	Output	RAM Address.
RAM_ENA	Output	RAM Cycle Enable.
RAM_WEA	Output	RAM Write Cycle.
RAM_DOUT[21:0]	Input	RAM Read Data.

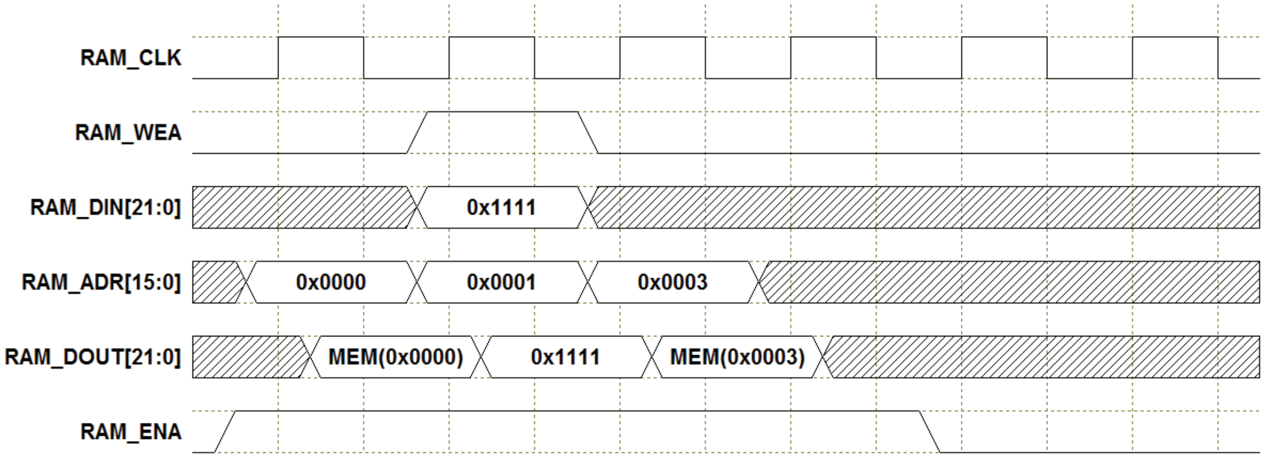


Figure 5. RAM Interface Timing

Table 8. IP Core External Transceiver Interface Signals

Signal Name	Input/Output	Description
TXCLK (P0)	Output	Transmit Clock.
TXAB (P1)	Output	Transmit select. Selects transmission on BUS A or BUS B. TXAB (P1) = 0 selects BUS A. TXAB (P1) = 1 selects BUS B.
P2	Input	IP Security Input (Connect to output P2 on HI-1587 IP Dongle Transceiver).
P3	Output	IP Security Output (Connect to input P3 on HI-1587 IP Dongle Transceiver).
TX	Output	MIL-STD-1553 Bus Transmit Positive Output.
$\overline{\text{TX}}$	Output	MIL-STD-1553 Bus Transmit Negative Output.
RXA	Input	MIL-STD-1553 Bus A Receive Positive Input.
$\overline{\text{RXA}}$	Input	MIL-STD-1553 Bus A Receive Negative Input.
RXB	Input	MIL-STD-1553 Bus B Receive Positive Input.
$\overline{\text{RXB}}$	Input	MIL-STD-1553 Bus B Receive Negative Input.

6. MEMORY MAP

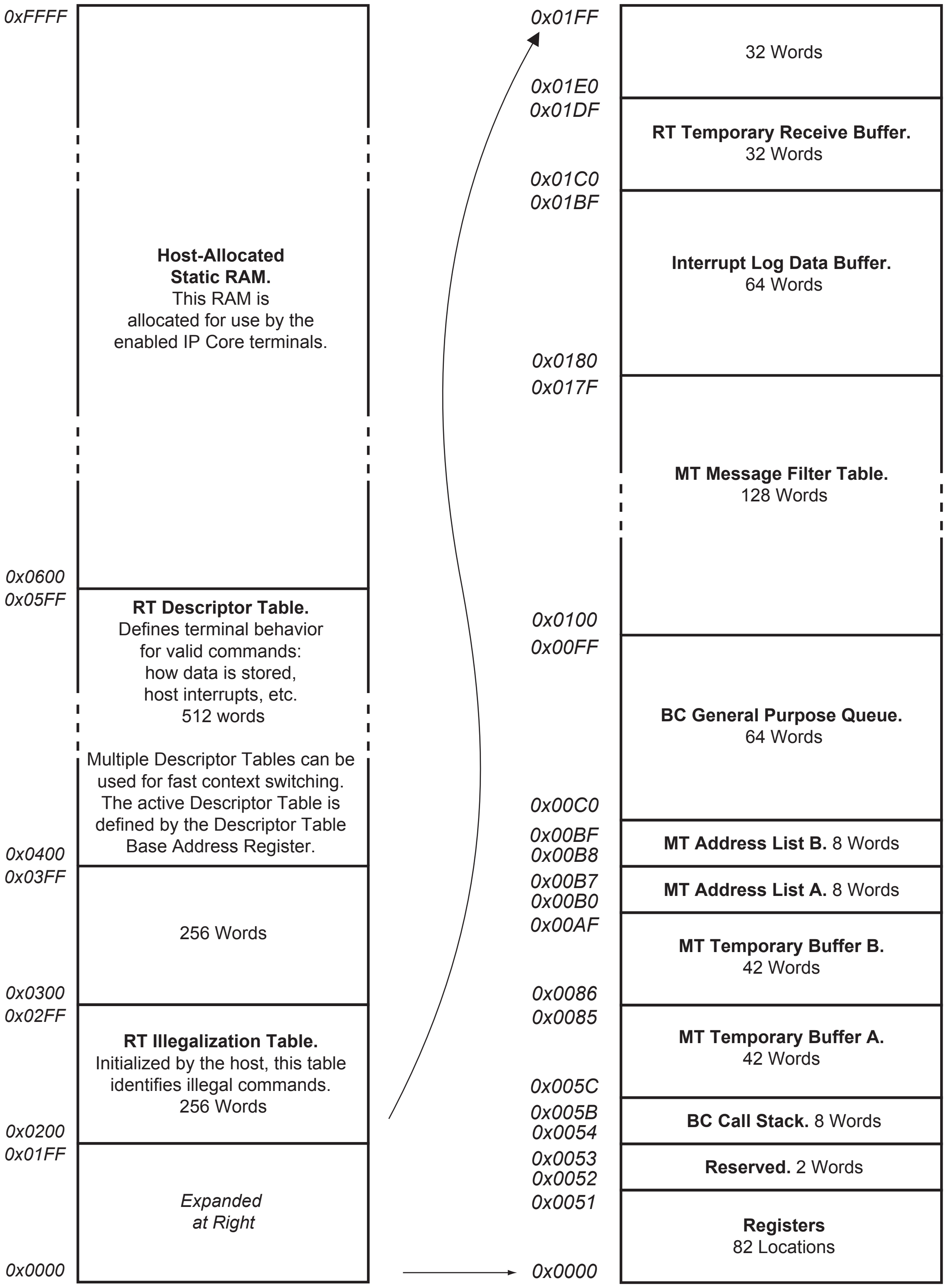

0xFFFF
Host-Allocated Static RAM. This RAM is allocated for use by the enabled IP Core terminals.
0x0600
0x05FF
RT Descriptor Table. Defines terminal behavior for valid commands: how data is stored, host interrupts, etc. 512 words
Multiple Descriptor Tables can be used for fast context switching. The active Descriptor Table is defined by the Descriptor Table Base Address Register.
0x0400
0x03FF
256 Words
0x0300
0x02FF
RT Illegalization Table. Initialized by the host, this table identifies illegal commands. 256 Words
0x0200
0x01FF
Expanded at Right
0x0000
0x01FF
32 Words
0x01E0
0x01DF
RT Temporary Receive Buffer. 32 Words
0x01C0
0x01BF
Interrupt Log Data Buffer. 64 Words
0x0180
0x017F
MT Message Filter Table. 128 Words
0x0100
0x00FF
BC General Purpose Queue. 64 Words
0x00C0
0x00BF
0x00B8
MT Address List B. 8 Words
0x00B7
0x00B0
MT Address List A. 8 Words
0x00AF
MT Temporary Buffer B. 42 Words
0x0086
0x0085
MT Temporary Buffer A. 42 Words
0x005C
0x005B
0x0054
BC Call Stack. 8 Words
0x0053
0x0052
Reserved. 2 Words
0x0051
Registers 82 Locations
0x0000

Figure 6. Address Mapping for Registers and RAM

7. RAM STRUCTURES

Figure 6 shows a map for memory and registers. Application requirements dictate the specific RAM structures needed. These structures listed here are explained later.

7.1. Interrupt Log Data Buffer

The IP core maintains information from the last 32 interrupts in a 64-word circular buffer in RAM known as the Interrupt Log. Two 16-bit words characterize each interrupt; one word identifies the interrupt type (Interrupt Identification Word) and one word identifies the command that generated the interrupt (Interrupt Address Word). After reset, the Interrupt Log Address Register contains the fixed buffer start address of 0x0180. After each occurring interrupt, the IP core updates the register to point to the log address used for the next occurring interrupt.

7.2. Bus Controller (BC) Instruction List

BC message sequencing uses an instruction list, comprised of multiple 2-word entries. Each entry consists of an instruction word (op code plus execution condition code) followed by a parameter word. Some op codes execute unconditionally, but most are conditional, and execute only if the condition code specified in the instruction word tests true. This architecture provides flexibility for message frame scheduling and autonomous BC program execution, based on various conditions.

7.3. Bus Controller (BC) Msg Control / Status Stack

Referenced when a BC Instruction List parameter is an address pointer, the Message Control Status Stack consists of 8- or 16-word control blocks for individual MIL-STD-1553 messages. Within the message control/status block, execution parameters are provided including inter-message gap, message format, active bus, message command word(s), the address pointer to a message data block in RAM (when the command includes data) and the expected RT status response.

7.4. Bus Controller (BC) Call Stack

When the Bus Controller executes a “subroutine call” op code, the BC Subroutine Call Stack stores the list address for the next op code, to be executed upon return from subroutine. The BC call stack allows 8 levels of nesting.

7.5. Bus Controller (BC) General Purpose Queue

The Bus Controller General Purpose Queue provides a convenient way for the BC to convey information to the external host. Numerous BC op codes push various data onto the queue, including time tag count, data values, message Block Status Word or the data at specific RAM address locations.

7.6. Monitor Terminal Temporary Buffers A & B

The Monitor Terminal (MT) has one 42 word receive buffer for each bus. Received command words, status words and data words are temporarily stored in these buffers. At message completion, the recorded data is copied to Monitor Terminal buffer(s).

7.7. Monitor Terminal (MT) Address List

The monitor Address List contains user-written RAM addresses assigning buffer start and end addresses, and buffer utilization interrupt addresses. It also has IP core-maintained data buffer pointer(s). RAM is alocated for two 8 word lists.

7.8. Monitor Terminal (MT) Message Filter Table

This 128-word table is optionally used to selectively monitor MIL-STD-1553 messages based on each command's RT address and subaddress, and the transmit/receive bit status.

7.9. Monitor Terminal (MT) Data Buffers

For bus monitor applications, the SMT simple monitor stores command and data words in separate RAM buffers. The monitor Address List assigns RAM buffer boundaries.

7.10. RT Command Illegalization Table

Optional illegal command detection utilizes this table in RAM. The table can illegalize any logical combination of 11 command word bits comprising subaddress, the transmit/receive bit and word count (or mode code), plus broadcast vs non-broadcast status, resulting in a total of 4,096 possible combinations. Terminal response to an illegal command sets "Message Error" status and transmits Status Word only. If illegal command detection is not used (that is, no "illegal" entries in Illegalization Table), the terminal responds "in form" to all valid commands.

7.11. RT Descriptor Table

A host-initialized Descriptor Table in RAM defines how the terminal responds to valid commands. The table is comprised of 128 four-word descriptor blocks. Each of 32 subaddresses and 32 mode code values has one descriptor block for transmit and another for receive. The descriptor table defines message options (interrupt selections, data buffer mode, etc.) and contains pointers to allocated data storage in RAM.

7.12. RT Temporary Receive Buffer

The Remote Terminal temporarily stores command and data words during message transaction. At successful message completion, all data is transferred to assigned subaddress buffers. This strategy prevents valid data from being overwritten by incomplete or bad data from a later message ending in error.

7.13. RT Message Data Buffers

Subaddress transmit and receive commands transact from 1 to 32 data words. For each transmit or receive subaddress, the application allocates space in RAM for storing a Message Information Word, Time Tag Word and the transacted message data words.

7.14. RT Storage for Mode Code Commands

In addition to commands used for data transmit and receive, MIL-STD-1553 also defines "mode code commands" for command and control. These "mode commands" either transfer a single data word, or no data word at all. The user has the option for storing mode data words in RAM buffers assigned in the Descriptor Table, or stored within the RT Descriptor Table itself. The second option is often preferred for its simplicity.

8. HARDWARE FEATURES

8.1. Remote Terminal Address Inputs

The 5-bit Remote Terminal address for the RT is set using inputs RTADR[4:0]. The RTADP input should be set or reset to present matching odd parity. The state of the RT address and parity inputs is latched into the “Remote Terminal Operational Status Register (0x0018)” upon rising edge on the $\overline{\text{RESET}}$ master reset input. The RT Address and Parity are also software programmable (see “Table 3. IP Core Configuration Signals” and “Table 4. RT_AD_LAT and RTA_SW_EN Truth Table”).

8.2. Encoder and Decoders

The IP core uses separate Manchester II encoders and decoders for each bus. Decoders check for proper sync pulse and Manchester waveform, edge skew, correct number of bits and parity. Encoders are used for transmission. During transmit, each encoded word is looped back through the bus decoder for error checking. Receiver bus sampling is clocked at 50 MHz, providing excellent tolerance to zero-crossing distortion.

8.3. Transmit Time-out Fail-Safe Counter

BUS A and BUS B transmitters both have continuously running watchdog timers that prevent continuous transmission beyond 663µs. Configuration options can optionally set the Terminal Flag status bit in the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” and inhibit all further transmissions once watchdog timer timeout has occured.

8.4. MIL-STD-1760 Mode

A dedicated input is available to activate MIL-STD-1760 Mode. When operating in MIL-STD-1760 mode, the IP core responds to any valid command (with matching RT address) with the BUSY bit set in the status word. No data words will be transmitted and no interrupts or logging of data occurs during post-reset initialization when operating in MIL-STD-1760 mode. **NOTE: The IP core will respond with BUSY bit set following authentication with the IP dongle.**

9. REGISTER & MEMORY ADDRESSING

The HI-6300 has an internal address space of 64K 22-bit words (16-bit word oriented and 6 bits for RAM Error-Correcting Code, ECC). Registers occupy the low 80 locations in this address space. In this data sheet, register / RAM addresses are expressed as hexadecimal numbers, using the C programming convention where the prefix “0x” denotes a hexadecimal value; e.g., 0x00FF represents 00FF hex.

Figure 6 shows address mapping for registers and FPGA RAM space. All registers and some RAM structures have fixed addresses. Other RAM structures shown are relocatable; Each relocatable structure has a base address register. Figure 6 shows the default locations for relocatable structures. RAM allocations for unused MIL-STD-1553 functions can be reassigned as needed. For example, an application using just a Bus Monitor can reassign all BC and RT RAM for monitor needs. IP Core RAM and register address mapping is word oriented, rather than byte oriented.

10. REGISTER DEFINITIONS

Residing at the start of the memory address space, 82 addresses are reserved for registers. Register addresses overlay the shared RAM address space. Register bits are active high and bit 15 is the most significant.

Table 9 lists all IP core registers.

Table 9. Register Summary

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
0	0x0000	All	"Master Configuration Register 1 (0x0000)" on page 30	0x0000
1	0x0001	All	"Master Status and Reset Register (0x0001)" on page 36	0x0000
2	0x0002	RT	**When TEST input signal is logic 0**, this address is "Remote Terminal Current Command Register (0x0002)" on page 127 **When TEST input signal is logic 1**, this address is "Loopback Test Receive Data Register (0x0002)" on page 199	0x0000
3	0x0003	RT	"Remote Terminal Current Control Word Address Register (0x0003)" on page 127	0x0000
4	0x0004	All	"IP Core Configuration Register (0x0004)" on page 35	0x0000
5	0x0005	All	"IP Core Security Register (0x0005)" on page 36	0x0000
6	0x0006	All	"Hardware Pending Interrupt Register (0x0006)" on page 44	0x0000
7	0x0007	BC	"Bus Controller (BC) Pending Interrupt Register (0x0007)" on page 95	0x0000
8	0x0008	MT	"SMT Bus Monitor Pending Interrupt Register (0x0008)" on page 116	0x0000
9	0x0009	RT	"Remote Terminal (RT) Pending Interrupt Register (0x0009)" on page 135	0x0000
10	0x000A	All	"Interrupt Count & Log Address Register (0x000A)" on page 40	0x0180
11	0x000B	----	Reserved	0x0000
12	0x000C	----	Reserved	0x0000
13	0x000D	----	Reserved	0x0000
14	0x000E	----	Reserved	0x0000
15	0x000F	All	"Hardware Interrupt Enable Register (0x000F)" on page 44	0x6018
16	0x0010	BC	"Bus Controller (BC) Interrupt Enable Register (0x0010)" on page 95	0x0000
17	0x0011	MT	"SMT Bus Monitor Interrupt Enable Register (0x0011)" on page 116	0x0000
18	0x0012	RT	"Remote Terminal (RT) Interrupt Enable Register (0x0012)" on page 135	0x0000
19	0x0013	All	"Hardware Interrupt Output Enable Register (0x0013)" on page 44	0x6018
20	0x0014	BC	"Bus Controller (BC) Interrupt Output Enable Register (0x0014)" on page 95	0x0000
21	0x0015	MT	"SMT Bus Monitor Interrupt Output Enable Register (0x0015)" on page 116	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
22	0x0016	RT	“Remote Terminal (RT) Interrupt Output Enable Register (0x0016)” on page 135	0x0000
23	0x0017	RT	“Remote Terminal Configuration Register (0x0017)” on page 119	0x0000
24	0x0018	RT	“Remote Terminal Operational Status Register (0x0018)” on page 125	0x0000
25	0x0019	RT	“Remote Terminal Descriptor Table Base Address Register (0x0019)” on page 127	0x0400
26	0x001A	RT	“Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” on page 128	0x0000
27	0x001B	RT	**When TEST input signal is logic 0**, this address is “Remote Terminal Current Message Information Word Register (0x001B)” on page 129. **When TEST input signal is logic 1**, this address is “RAM Self-Test Fail Address Register (0x001B)” on page 199.	0x0000
28	0x001C	RT	“Remote Terminal Bus A Select Register (0x001C)” on page 130	0x0000
29	0x001D	RT	“Remote Terminal Bus B Select Register (0x001D)” on page 130	0x0000
30	0x001E	RT	“Remote Terminal Built-In Test (BIT) Word Register (0x001E)” on page 131	0x0000
31	0x001F	RT	**When TEST input signal is logic 0**, this address is “Remote Terminal Alternate Built-In Test (BIT) Word Register (0x001F)” on page 132. **When TEST input signal is logic 1**, this address is “Loopback Test Transmit Data Register (0x001F)” on page 199.	0x0000
32	0x0020	----	Reserved	0x0000
33	0x0021	----	Reserved	0x0000
34	0x0022	----	Reserved	0x0600
35	0x0023	----	Reserved	0x0000
36	0x0024	----	Reserved	0x0000
37	0x0025	----	Reserved	0x0000
38	0x0026	----	Reserved	0x0000
39	0x0027	----	Reserved	0x0000
40	0x0028	----	**When TEST input signal is logic 0**, this register has no function. **When TEST input signal is logic 1**, this address is “Self-Test Control Register (0x0028)” on page 195	0x0000
41	0x0029	MT	“SMT Configuration Register (0x0029)” on page 108	0x0801
42	0x002A	----	Reserved	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
43	0x002B	----	Reserved	0x0000
44	0x002C	----	Reserved	0x0000
45	0x002D	----	Reserved	0x0000
46	0x002E	----	Reserved	0x0000
47	0x002F	MT	“SMT Bus Monitor Address List Start Address Register (0x002F)” on page 111	0x00B0
48	0x0030	MT	“SMT Next Message Command Buffer Address (0x0030)” on page 111	0x0000
49	0x0031	MT	“SMT Last Message Command Buffer Address (0x0031)” on page 112	0x0000
50	0x0032	BC	“BC (Bus Controller) Configuration Register (0x0032)” on page 78	0x0000
51	0x0033	BC	“Start Address Register for Bus Controller (BC) Instruction List (0x0033)” on page 87	0x0000
52	0x0034	BC	“Bus Controller (BC) Instruction List Pointer (0x0034)” on page 87	0x0000
53	0x0035	BC	“Bus Controller (BC) Frame Time Remaining Register (0x0035)” on page 88	0x0000
54	0x0036	BC	“Bus Controller (BC) Time To Next Message Register (0x0036)” on page 88	0x0000
55	0x0037	BC	“Bus Controller (BC) Condition Code Register (Read 0x0037)” on page 88 “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)” on page 91	0x0000
56	0x0038	BC	“Bus Controller (BC) General Purpose Queue Pointer Register (0x0038)” on page 91	0x00C0
57	0x0039	All	“Time Tag Counter Configuration Register (0x0039)” on page 50	0x0000
58	0x003A	MT	“SMT Bus Monitor Time Tag Count Register (0x003A)” on page 112	0x0000
59	0x003B	MT	“SMT Bus Monitor Time Tag Count Mid Register (0x003B)” on page 112	0x0000
60	0x003C	MT	“SMT Bus Monitor Time Tag Count High Register (0x003C)” on page 112	0x0000
61	0x003D	MT	“SMT Bus Monitor Time Tag Utility Register (0x003D)” on page 113	0x0000
62	0x003E	MT	“SMT Bus Monitor Time Tag Utility Mid Register (0x003E)” on page 113	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
63	0x003F	MT	"SMT Bus Monitor Time Tag Utility High Register (0x003F)" on page 113	0x0000
64	0x0040	MT	"SMT Bus Monitor Time Tag Match Register (0x0040)" on page 114	0x0000
65	0x0041	MT	"SMT Bus Monitor Time Tag Match Mid Register (0x0041)" on page 114	0x0000
66	0x0042	MT	"SMT Bus Monitor Time Tag Match High Register (0x0042)" on page 114	0x0000
67	0x0043	BC	"Bus Controller (BC) Time Tag Counter (0x0043)" on page 92	0x0000
68	0x0044	BC	"Bus Controller (BC) Time Tag Counter High (0x0044)" on page 92	0x0000
69	0x0045	BC	"Bus Controller (BC) Time Tag Utility Register (0x0045)" on page 93	0x0000
70	0x0046	BC	"Bus Controller (BC) Time Tag Utility High Register (0x0046)" on page 93	0x0000
71	0x0047	BC	"Bus Controller (BC) Time Tag Match Register (0x0047)" on page 93	0x0000
72	0x0048	BC	"Bus Controller (BC) Time Tag Match High Register (0x0048)" on page 93	0x0000
73	0x0049	RT	"Remote Terminal Time Tag Counter Register (0x0049)" on page 132	0x0000
74	0x004A	RT	"Remote Terminal Time Tag Utility Register (0x004A)" on page 133	0x0000
75	0x004B	----	Reserved	0x0000
76	0x004C	----	Reserved	0x0000
77	0x004D	BC / RT	"Extended Configuration Register (0x004D)" on page 46	0x0000
78	0x004E	All	"Master Configuration Register 2 (0x004E)" on page 33	0x3100
79	0x004F	BC	BC Last Message Block Address	Undefined[1]

Note 1: RAM location 0x004F is not a register and does not have a defined reset value.

11. REGISTERS USED BY ALL IP CORE FUNCTIONS

11.1. Master Configuration Register 1 (0x0000)

TXINHA	TXINHB	BCSTRT	BCENA	Reserved	Reserved	Reserved	MTENA	Reserved	RTENA	Reserved	RTSTEX	BSDTXO	Reserved	IMTA	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

All bits in this 16-bit register are read-write and are fully maintained by the host. This register is cleared after $\overline{\text{RESET}}$ input signal, and is unaffected by assertion of the MTRESET, RTRESET bits in the “Master Status and Reset Register (0x0001)”.

When configuring registers and RAM following Reset, “Master Configuration Register 1 (0x0000)” should always be written **last** to ensure configuration and initialization is complete **before** starting terminal operation (i.e. this ensures BCSTART and RTSTEX bits are not set until after configuration is complete).

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	TXINHA	R/W	0	Transmit Inhibit Bus A. This bit is logically ORed with the TXINHA input signal. This register bit and the corresponding TXINHA signal globally affects all enabled 1553 IP cores (BC, MT, RT). This inhibit disables all transmission on Bus A.
14	TXINHB	R/W	0	Transmit Inhibit Bus B. This bit is logically ORed with the TXINHB input signal. This register bit and the corresponding TXINHB signal globally affects all enabled 1553 IP cores (BC, MT, RT). This inhibit disables all transmission on Bus B.
13	BCSTRT	R/W	0	Bus Controller Start. If the BCENA register bit is logic 1, a host write which sets this bit to 1 begins Bus Controller operation. When written to 1, this bit self-resets to 0. This bit always reads back a logic 0 state.
12	BCENA	R/W	0	Bus Controller Enable. If this bit is logic 0, Bus Controller operation is disabled. When the BCENA bit is logic 1, the Bus Controller IP core is enabled, but BC operation does not begin until BCSTRT bit 13 is set. If this bit becomes logic 0 while BC operation is underway, BC operation is immediately terminated without waiting for message completion.
11 − 9	Reserved	-	-	These bits are not used and read logic 0.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
8	MTENA	R/W	0	Bus Monitor Enable. If this bit equals logic 0, Bus Monitor operation is disabled. When the MTENA bit is logic 1, the Bus Monitor is enabled. Operation commences when the receiver first decodes MIL-STD-1553 activity meeting the “start record” criteria selected by bits 6:5 in the MT Configuration Register. If monitor operation is underway when the MTENA bit becomes logic 0, monitor operation stops after completion of any message already underway; monitor resumes when the MTENA bit is logic 1.
7	Reserved	-	-	This bit is not used and reads logic 0.
6	RTENA	R/W	0	Remote Terminal Enable. If this bit equals logic 0, RT operation is disabled. When this bit is logic 1, Remote Terminal is enabled, but operation is controlled by the state of the RTSTEX register bit.
5	Reserved	-	-	This bit is not used and reads logic 0.
4	RTSTEX	R/W	0	Remote Terminal Start Execution. If register bit 6 is logic 1, setting this bit begins Remote Terminal operation. Once running, resetting this bit (or the RTENA register bit) immediately stops RT operation.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
3	BSDTXO	R/W	0	Bus Shutdown Transmit Only. The bit only applies when Remote Terminal is enabled. The BSDTXO bit determines how a 1553 bus inhibit works when (a) the RTINHA or RTINHB bit is set in the “Remote Terminal Configuration Register (0x0017)”, or (b) the RT receives a valid “bus shutdown” mode code command, either MC4 or MC20: • When the BSDTXO bit is reset, logic 1 for an RTINHA or RTINHB bit in “Remote Terminal Configuration Register (0x0017)” (or a “bus shutdown” mode command with auto shutdown enabled) **inhibits both transmit and receive** on the selected bus. • When the BSDTXO bit is set, logic 1 for an RTINHA or RTINHB bit in “Remote Terminal Configuration Register (0x0017)” (or a “bus shutdown” mode command with auto shutdown enabled) **inhibits transmit only** on the selected bus; but receive functions are unaffected. Valid commands are heeded, but the RT transmits no responses. **NOT RECOMMENDED**. The RT automatically fulfills unconditional MC4 “bus shutdown” in accordance with the BSDTXO setting, as well as MC5 “override bus shutdown.” The AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” determines whether conditional MC20 “selected bus shutdown” and MC21 “override selected bus shutdown” are fulfilled automatically, or by host writes to the RTINHA or RTINHB bits in the “Remote Terminal Configuration Register (0x0017)”: • When the AUTOBSD bit is logic 1 in the “Remote Terminal Configuration Register (0x0017)” (see page 119), **automatic fulfillment is disabled** for MC20 “selected bus shutdown” and MC21 “override selected bus shutdown” mode commands. The host fulfills bus shutdown and override by writing the RTINHA and RTINHB bits in the “Remote Terminal Configuration Register (0x0017)”. • When the AUTOBSD bit is logic 0 in the “Remote Terminal Configuration Register (0x0017)”, **automatic fulfillment is enabled** for MC20 “selected bus shutdown” and MC21 “override selected bus shutdown” mode commands. When the received mode data word matches the value stored in the RT “Bus A (or B) Select” register, the RT automatically fulfills MC20 “selected bus shutdown” in accordance with the BSDTXO setting, as well as MC21 “override selected bus shutdown”. Auto-shutdown bypasses the RTINHA and RTINHB bits in the “Remote Terminal Configuration Register (0x0017)”, but the upper 4 bits in the RT’s BIT Word register indicate Tx and Rx bus shutdown status.
2	Reserved	-	-	This bit is not used and reads logic 0.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
1	IMTA	R/W	0	Indicate MT Activity. When this bit equals 0, the ACTIVE status output is not asserted for Bus Monitor activity, unless the monitored message involves another on-core terminal). When this bit equals 1, enabled Bus Monitor activity is logically-ORed with the activity of the other on-core terminals to determine ACTIVE status; the ACTIVE output is asserted during such Bus Monitor activity, whether or not the monitored message involves another on-core terminal.
0	Reserved	-	-	This bit is not used and reads logic 0.

11.2. Master Configuration Register 2 (0x004E)

TXTEST	TXTSTPND	BCESTAT	MTESTAT	REVID3	REVID2	REVID1	REVID0	MODE1760	Reserved	Reserved	LBFTF	WDTF	WDTXINH	DBCMSTAT	BCLBK	
0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	Reset
RW		R		RW												Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	TXTEST	R/W	0	Transmitter Timeout Protection Test. BUS A and BUS B transmitters both have continuously running watchdog timers that prevent continuous transmission beyond 663µs. This bit is used to test the transmitter timeout protection feature. The following host initiation sequence needs to occur to activate the test: • Assert the TEST signal. • Write logic “1” to TXTEST bit 15 (the test signal may now be negated). • On the next host access, write logic “0” to TXTEST bit 15. Following the host initiation sequence, bit 14 will be set and the next RT transmission or response will attempt to transmit 35 MIL-STD-1553 data words with incrementing data count, totaling 700µs. If the transmitter timeout protection is functioning correctly, the transmission will stop after 663µs. If enabled, this will also result in Loopback Fail Interrupt bits 12 or 11 being set in the “Hardware Pending Interrupt Register (0x0006)”. **Note:** This bit will be reset when read and always reads back logic “0”.
14	TXTSTPND	R/W	0	Transmitter Timeout Protection Test Pending. This bit is set when the host initiates a transmitter timeout protection test. When the test begins on the next transmission, this bit clears automatically.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13	BCESTAT	R	0	Bus Controller Enabled Status. This bit reflects the status of the BCENA signal and reads “1” if the BC is enabled. **Note:** The HI-1587 IP dongle transceiver **must also** provide BC activation permission, or this signal will read “0” and the BC will not be enabled, regardless of the status of the BCENA signal.
12	MTESTAT	R	0	Monitor Terminal Enabled Status. This bit reflects the status of the MTENA signal and reads “1” if the MT is enabled. **Note:** The HI-1587 IP dongle transceiver **must also** provide MT activation permission, or this signal will read “0” and the MT will not be enabled, regardless of the status of the MTENA signal.
11 − 8	REVID[3:0]	R/W	0001	Revision ID. Following Reset, these bits will read 0001, the current revision ID.
7	MODE1760	R/W	0	Mode 1760 Status. This bit is set when the MODE1760 signal was high at $\overline{\text{RESET}}$ rising edge, so the IP core is operating in 1760 mode (see “MIL-STD-1760: Busy Status Assertion After Power-Up”). The host may read this bit to confirm the IP core is operating in 1760 mode. Once in 1760 mode, the RT immediately responds to any valid command (with matching RT address) with BUSY bit set in the Status Word. No data words are transmitted. No interrupts or logging of data occurs. Such responses can occur during power-up RAM self-test or host-written register and RAM initialization. There are three ways to exit 1760 mode and clear this bit: 1. write “1” to this register bit, 2. write “1” to the RTSTEX bit 4 in “Master Configuration Register 1 (0x0000)”, or 3. drive the MODE1760 input signal to logic “0” state.
6 − 5	Reserved	R/W	0	These bits are not used and read logic 0.
4	LBFTF	R/W	0	RT Self-Test Loopback Fail Terminal Flag. If this bit is set, an RT BUSA or BUSB loopback failure will cause the Terminal Flag status bit 0 in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” to be set. **Note:** This bit is NOT set for Test Mode loopback failures utilizing the TEST input signal and registers “Loopback Test Transmit Data Register (0x001F)” and “Loopback Test Receive Data Register (0x0002)”.
3	WDTF	R/W	0	Watchdog Timer Transmitter Timeout Terminal Flag If this bit is set, watchdog timer transmitter timeout will cause the Terminal Flag status bit 0 in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” to be set.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
2	WDTXINH	R/W	0	Watchdog Timer Transmitter Timeout Transmit Inhibit If this bit is set, watchdog timer timeout on BUS A or BUS B will cause transmission to be inhibited on the corresponding bus. If timeout occurs on BUS A, RTINHA bit 13 will be set in “Remote Terminal Configuration Register (0x0017)”. If timeout occurs on BUS B, RTINHB bit 12 will be set in “Remote Terminal Configuration Register (0x0017)”. **Note:** If WDTXINH bit 2 is NOT set, then watchdog timer timeout will stop a transmission on BUS A or BUS B after 663µs for the next transmission only. WDTXINH bit 2 must be set to inhibit subsequent transmissions on BUSA or BUS B.
1	DBCMSTAT	R/W	0	Dynamic Bus Control Masked Status Bit Flag If this bit is set and the RT responds to a MC0 (Mode Code 0 Dynamic Bus Control), the Masked Status Bit 7 of the “BC Block Status Word” will be set if the RT Status Word DBCA bit 1 is set.
0	BCLBK	R/W	0	BC to RT/MT Digital Loopback Test Setting this bit enables BC to RT and MT digital loopback testing. No loopback activity is transmitted to the bus. See Sections 22.1.6 and 22.1.7 for more details on digital loopback testing.

11.3. IP Core Configuration Register (0x0004)

Not Used							RAM_SZE	Not Used							CLK_SEL	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R/W							RW	R/W							RW	Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This register is used to set the size of the RAM to undergo Self-Test (entire 64 KB or only 8 KB). It is also used to select the Clock Frequency (50 MHz or 100 MHz).

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15 − 9	Reserved	R/W	0	These bits are not used.
8	RAM_SZE	R/W	0	BIST Test Size Selection 0 = 64 K words (Default) 1 = 8 K words
7 − 1	Reserved	R/W	0	These bits are not used.
0	CLK_SEL	R/W	0	Clock Frequency Selection 0 = 100 MHz (Default) 1 = 50 MHz

11.4. IP Core Security Register (0x0005)

IPCONFIG [15:8]								Reserved	Reserved	Reserved	Reserved	ULK_STAT3	ULK_STAT2	ULK_STAT1	ULK_STST0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15 − 8	IPCONFIG	R	0	Factory Configuration Status Bits Read 0x07: BC/RT/MT Read 0x06: RT/MT Read 0x05: BC/RT Read 0x04: RT only
7 − 4	Reserved	R	0	These bits are not used.
3 − 0	ULK_STAT	R		Security Handshake Unlock Status. Bit 3: Unlock Complete Bit 2: Unlock Passed Bit 1: Unlock Failed Bit 0: Unlock in Progress Read 0x00: Clear and no unlock activity Read 0x01: Unlock In Progress Read 0x0a: Unlock Failed Read 0x0c: Unlock Passed

11.5. Master Status and Reset Register (0x0001)

READY	Reserved	Reserved	MTRESET	Reserved	RTRESET	BCMIP	BCACTIVE	MTMIP	Reserved	Reserved	RTMIP	Reserved	RTINH	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R			RW			R										Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register has a combination of read only and read-write bits. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of MTRESET, RTRESET register bits.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	READY	R	0	The READY output signal reflects the state of this register bit. READY is low when a soft reset caused by bit 12~10 assertion or built-in test is underway. Host access to IP core registers or RAM is locked out while READY is low. While READY = 0, any host read access returns the value in this register, regardless of address provided. When READY goes high, the host may access registers and RAM.
14 − 13	Reserved	R	0	These bits are not used.
12	MTRESET	R/W	0	Bus Monitor Reset. When written to logic 1, this bit initiates Bus Monitor reset. This bit remains high until reset is complete. While this bit remains high, the READY output signal and register bit 15 are held low, host RAM and register access is suspended. While READY = 0, any host read access returns the value in this register, regardless of address provided. Upon reset completion, this bit self-clears to logic 0, the READY signal goes high and host read/write access is restored.
11	Reserved	R	0	This bit is not used.
10	RTRESET	R/W	0	Remote Terminal Reset. When written to logic 1, this bit initiates RT reset by clearing the RTSTEX Start Execution bit in the “Master Configuration Register 1 (0x0000)”, then performing the RT soft reset actions described in “Reset and Initialization” on page 193. This bit remains high until reset is complete. While this bit remains high, the READY output signal and register bit 15 are held low, host RAM and register access is suspended. While READY = 0, any host read access returns the value in this register, regardless of address provided. Upon reset completion, this bit self-clears to logic 0, the READY signal goes high and host read/write access is restored.
9	BCMIP	R	0	BC Message in Process. This bit is high when the BC is processing a MIL-STD-1553 message. Falling edge occurs at message completion, after register and RAM buffer updates.
8	BCACTIVE	R	0	BC Active. This bit is high when the BC is enabled and running. It will read logic 1 during MIL-STD-1553 message processing and during programmed delays.
7	MTMIP	R	0	Bus Monitor Message in Process. This bit is set when a valid MIL-STD-1553 command is decoded, and is reset upon monitored message completion.
6 − 5	Reserved	R	0	These bits are not used.
4	RTMIP	R	0	Remote Terminal Message in Process. This bit is set when a valid MIL-STD-1553 command is decoded for the RT, and is reset upon message completion.
3	Reserved	R	0	This bit is not used.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
2	RTINH	R	0	Remote Terminal Bus Inhibited. This bit is high when one bus is inhibited for RT due to execution of a “bus shutdown” mode code command. The shut-down bus is identified in the RT BIT (built-in test) Word Register (see page 131). Shut-down can be ended by “bus shutdown override” mode code command, $\overline{\text{RESET}}$ reset or setting the RTRESET bit in this register.
1 − 0	Reserved	R	0	These bits are not used.

11.6. Overview of Interrupts

For interrupt management, the host accesses up to thirteen registers and a 64-word circular Interrupt Log Buffer in RAM. The log buffer and the Interrupt Count & Log Address Register are utilized in any system design involving interrupts. In addition, there are four 3-register groups, identified by terminal function. One 3-register group is for Hardware Interrupts; this register triplet is always active. The other 3-register groups are only active when the corresponding terminal functions are enabled; these are the interrupt register triplets used for Bus Controller, Bus Monitor and Remote Terminal interrupts.

Each interrupt register triplet for BC, SMT, RT or Hardware consists of

- An Interrupt Enable Register to enable and disable interrupt event recognition
- A Pending Interrupt Register to capture the occurrence of enabled interrupt events
- An Interrupt Output Enable Register selectively enables $\overline{\text{INT}}$ output to host when enabled interrupts occur

Within each register triplet, corresponding register bits are mapped to the same interrupt-causing event. Initialize the Interrupt Enable Register to select interrupt-causing events heeded by the HI-6300; most applications utilize just a subset of the available interrupt options. Interrupt-causing events are ignored if their corresponding bits are reset in the Interrupt Enable Register. Setting an Interrupt Enable register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

The next datasheet sections describe interrupt features, namely the Interrupt Log Buffer, the Interrupt Count & Log Address Register and the Hardware Interrupt register triplet.

11.7. Hardware Interrupt Behavior

Behavior described here for Hardware Interrupts closely resembles the behavior for the BC, RT and SMT interrupt register triplets, described later in the corresponding sections of this datasheet.

When an enabled hardware interrupt event occurs, the Interrupt Log Buffer is updated and a bit is set in the “Hardware Pending Interrupt Register (0x0006)”. This action takes place only if the bit for the interrupt-causing event was already set in the “Hardware Interrupt Enable Register (0x000F)”. The host can poll the “Hardware Pending Interrupt Register (0x0006)” to detect occurrence of hardware interrupts, indicated by non-zero value. When the host reads the “Hardware Pending Interrupt Register (0x0006)”, it automatically clears to 0x0000.

When an enabled hardware interrupt event occurs, if the corresponding bit is also set in the “Hardware Interrupt Output Enable Register (0x0013)”, the $\overline{\text{INT}}$ output is asserted to alert the host. Thus, the “Hardware Interrupt Output Enable Register (0x0013)” establishes two interrupt priority levels for hardware events: high priority interrupts generate an $\overline{\text{INT}}$ signal output, while low priority interrupts do not. The host detects low priority interrupts by polling the “Hardware Pending Interrupt Register (0x0006)”.

A single $\overline{\text{INT}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the Hardware, BC, RT or MT Interrupt Output Enable registers. Multiple interrupt-causing events can occur simultaneously, so each $\overline{\text{INT}}$ output assertion can result from one or more interrupt conditions.

When the host receives an $\overline{\text{INT}}$ signal from the IP core, it identifies the event (or events) that triggered the interrupt. The host has two options: (a) go to the Interrupt Log Buffer (using the method described in “11.8. Interrupt Count & Log Address Register (0x000A)” and “11.9. Interrupt Log Buffer”), or (b) use a hardware-assisted scheme using the three low order bits in the “Hardware Pending Interrupt Register (0x0006)” to identify new interrupt(s).

For the second method, the host reads the “Hardware Pending Interrupt Register (0x0006)”. While bits 15-3 in this register identify hardware interrupt conditions, the three low-order register bits indicate zero vs. non-zero status for the BC, RT and MT Pending Interrupt Registers. If any of these bits is logic 1, the corresponding Pending Interrupt Register has one or more interrupt flags set. Any combination of these three bits may be set, or all three bits may be zero, if only hardware interrupt(s) occurred. When the host reads any of the four Pending Interrupt registers, the read access self-resets the register to 0x0000. Thus, the host should retain the read value from the “Hardware Pending Interrupt Register (0x0006)” when 1 or more bits are non-zero in the bit 2-0 range. These bits indicate zero vs. non-zero status for the BC, RT and MT Pending Interrupt Registers:

- When bits 2-0 in the “Hardware Pending Interrupt Register (0x0006)” read 000, there are no new interrupts in the BC, RT and MT Pending Interrupt Registers.
- When BCIP (BC Interrupt Pending) **bit 0** is set in the “Hardware Pending Interrupt Register (0x0006)”, the “Bus Controller (BC) Pending Interrupt Register (0x0007)” contains a nonzero value. The host can read the “Bus Controller (BC) Pending Interrupt Register (0x0007)” Register to identify the specific bus controller interrupt event(s).
- When MTIP (SMT Interrupt Pending) **bit 1** is set in the “Hardware Pending Interrupt Register (0x0006)”, the “SMT Bus Monitor Pending Interrupt Register (0x0008)” contains a nonzero value. The host can read this register to identify the specific bus monitor interrupt event(s).
- When RTIP (RT Interrupt Pending) **bit 2** is set in the “Hardware Pending Interrupt Register (0x0006)”, the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” contains a nonzero value. The host can read the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” to identify specific RT interrupt event(s).

When polling the Pending Interrupt registers to identify low priority interrupts that do not assert the $\overline{\text{INT}}$ output, the same decoding method can be applied. A single read of the “Hardware Pending Interrupt Register (0x0006)” reveals zero vs. non-zero status of all Pending Interrupt registers.

Alternately, the host can poll the “Interrupt Count & Log Address Register (0x000A)” to identify low priority interrupts that do not assert the $\overline{\text{INT}}$ output. Bits 15:9 in this register contain a 7-bit count value indicating the number of interrupts logged (0 - 127) since the “Interrupt Count & Log Address Register (0x000A)” was last read. Although the “Interrupt Log Buffer” only holds data from the last 32 interrupts, register bits 15:9 count beyond 32 for buffer overrun detection.

Counting stops at 127. Register bits 15:9 are reset automatically when the host reads the “Interrupt Count & Log Address Register (0x000A)”.

11.8. Interrupt Count & Log Address Register (0x000A)

MSB Interrupt Count							MSB Interrupt Log Address									
0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is read-only and is fully maintained by logic. The register contains 0x0180 after error-free $\overline{\text{RESET}}$ signal master reset. It is not affected by assertion of MTRESET, RTRESET bits in the “Master Status and Reset Register (0x0001)”. **Note:** Four bits in the “Hardware Interrupt Enable Register (0x000F)” come out of $\overline{\text{RESET}}$ master reset fully enabled (see 11.10.1). If error occurs after reset to trigger one of these 4 interrupts, the post-reset value in the register will not be 0x0180. The upper bits will reflect 1 to 4 interrupts have occurred (count left-shifted 9 places) and the lower bits (ranging from 1 to 4 interrupts) will reflect an even pointer address of 0x182, 0x184, 0x186 or 0x188.

The value in Interrupt Log Address Register bits 8:0 is a 9-bit address pointer to the circular 64-word “Interrupt Log Buffer”, located in RAM. Register bits 8:6 are always 1-1-0 so the 9-bit address pointer ranges from 0x0180 to 0x01BE. This pointer indicates the storage address for two information words that will be stored for the next-occurring interrupt. The value is always even since two words are stored for each interrupt.

Upper register bits 15:9 contain a 7-bit count value for the number of interrupts logged (0 − 127) since the Interrupt Count & Log Address Register was last read. Although the circular “Interrupt Log Buffer” only retains data from the last 32 interrupts, counting continues beyond 32 so the host can detect circular buffer overrun. Bits 15:9 stop incrementing at full count (127 interrupts) and automatically reset to zero when the host reads this register.

After $\overline{\text{RESET}}$ master reset, the HI-6300 initializes this register to 0x0180, an interrupt count of zero and Interrupt Log Buffer address of 0x180. After reset, the first interrupt stores words at buffer addresses 0x0180 and 0x0181. Subsequent interrupts store word pairs at sequential addresses. Information words for the 32nd interrupt are stored in last two buffer addresses 0x01BE and 0x01BF, and the Interrupt Log Address “rolls over” to read 0x0180, where interrupt information for the 33rd post-reset interrupt will be stored.

11.9. Interrupt Log Buffer

Shown in Figure 7, the Interrupt Log Buffer is a circular 64-word buffer in RAM, residing at address range 0x0180 to 0x01BF. IP Core logic stores two information words in the buffer for each enabled interrupt that occurs, so buffer size dictates storage for up to 32 interrupt events. After the 32nd, 64th, 96th,... interrupt occurs, the buffer address pointer (bits 8:0 in register 0x000A) “wraps around” to buffer start address 0x0180 and subsequent interrupts overwrite previously stored interrupt information (see “Interrupt Count & Log Address Register (0x000A)” on page 40).

Interrupt logic stores two words in the Interrupt Log Buffer for each enabled interrupt that occurs: an Interrupt Identification Word and an Interrupt Address Word. The Interrupt Identification Word (IIW) identifies the occurring interrupt type (BC, MT or RT) using the same format as the three low order bits in the “Hardware Pending Interrupt Register (0x0006)”, and the interrupt itself by matching bits [15:3] to the applicable Pending Interrupt Register. More than one bit may be asserted in an Interrupt Identification Word. For example, IBR (interrupt broadcast received) and MERR (interrupt message error) can occur for the same RT message. One assertion of the INT output signal alerts the host when concurrent message interrupts occur.

The log buffer Interrupt Address Word varies, depending on the interrupt type. Hardware interrupts are not directly linked with command or message processing. Hardware interrupts write an Interrupt Address Word value of 0x0000, except when a RAMERR or UNCRE (ECC Correction or ECC Uncorrectable) interrupt occurs (see Table 10 below). For BC or SMT interrupts, the Interrupt Address Word (IAW) identifies the message in which the interrupt occurred. For RT interrupts, the Interrupt Address Word (IAW) identifies the start location of the Descriptor Table entry for the message in which the interrupt occurred:

Table 10. Format of IIW and IAW in Interrupt Log Buffer

Interrupt Type	Interrupt Identification Word (IIW)	Interrupt Address Word (IAW)
Hardware	Matches format of Hardware Pending Interrupt Register 0x0006 on page 44	Always 0x0000. **EXCEPTION:** When RAMERR or UNCRE interrupts occur (ECC Correction or ECC Uncorrectable error), the IAW will contain the address of the location in RAM where the error occured.
Bus Controller (BC)	Bits 2:0 will match bits 2:0 of the “Hardware Pending Interrupt Register (0x0006)” on page 44, indicating the interrupt type. For a BC interrupt, bit 0 will be set. Bits 15:3 will match bits 15:3 of the “Bus Controller (BC) Pending Interrupt Register (0x0007)” on page 95, indicating which interrupt occured.	A BC Control/Status Block address, points to the Block Status Word of the message in which interrupt occurred
Simple Bus Monitor (SMT)	Bits 2:0 will match bits 2:0 of the “Hardware Pending Interrupt Register (0x0006)” on page 44, indicating the interrupt type. For a MT interrupt, bit 1 will be set. Bits 15:3 will match bits 15:3 of the “SMT Bus Monitor Pending Interrupt Register (0x0008)” on page 116, indicating which interrupt occured.	An SMT Command Buffer address, points to the Block Status Word of the message in which interrupt occurred
Remote Terminal RT	Bits 2:0 will match bits 2:0 of the “Hardware Pending Interrupt Register (0x0006)” on page 44, indicating the interrupt type. For a RT interrupt, bit 2 will be set. Bits 15:3 will match bits 15:3 of the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” on page 135, indicating which interrupt occured. **EXCEPTION:** If INTBUSY bit 2 is set in “Extended Configuration Register (0x004D)”, then bit 9 of RT Interrupt Information Word serves as WASBUSY status flag, asserted if terminal was BUSY when message interrupt occurred. See “Extended Configuration Register (0x004D)” on page 46 **Note:** Busy status is not an interrupt causing event.	RT Descriptor Table address pointing to the start location of the Descriptor Table entry of the message in which interrupt occurred

For a given terminal (BC, SMT, or RT) multiple interrupts can be enabled, and two or more interrupts can occur in a single message. There will be a single 2-word Log Buffer update and the Interrupt Information Word will have one bit set for each occurring interrupt. Simultaneous interrupts for one terminal (having interrupt output enabled) are logically-ORed, resulting in a single assertion of the $\overline{\text{INT}}$ output to the host.

When operating with two or more enabled terminal IP cores (BC, SMT, or RT), simultaneous interrupts can occur in the same message for multiple terminals. Each terminal IP core with occurring interrupt(s) will have its own 2-word Log Buffer update. Simultaneous interrupts for multiple terminals (having interrupt output enabled) are logically-ORed, resulting in a single assertion of the $\overline{\text{INT}}$ output to the host.

In later data sheet sections, definitions are provided for interrupt register triplets used by the BC (bus controller), SMT (simple monitor terminal) and RT (remote terminal).

0x01BF	INTERRUPT 32	Interrupt Address Word
0x01BE	INTERRUPT 32	Interrupt Information Word
0x01BD	INTERRUPT 31	Interrupt Address Word
0x01BC	INTERRUPT 31	Interrupt Information Word
0x01BB	INTERRUPT 30	Interrupt Address Word
0x01BA	INTERRUPT 30	Interrupt Information Word
0x01B9	INTERRUPT 29	Interrupt Address Word
0x01B8	INTERRUPT 29	Interrupt Information Word
0x01B7	INTERRUPT 28	Interrupt Address Word
0x01B6	INTERRUPT 28	Interrupt Information Word
0x01B5	INTERRUPT 27	Interrupt Address Word
0x01B4	INTERRUPT 27	Interrupt Information Word
	.	
	.	
	.	
	.	
	.	
	.	
	.	
	.	
0x018B	INTERRUPT 6	Interrupt Address Word
0x018A	INTERRUPT 6	Interrupt Information Word
0x0189	INTERRUPT 5	Interrupt Address Word
0x0188	INTERRUPT 5	Interrupt Information Word
0x0187	INTERRUPT 4	Interrupt Address Word
0x0186	INTERRUPT 4	Interrupt Information Word
0x0185	INTERRUPT 3	Interrupt Address Word
0x0184	INTERRUPT 3	Interrupt Information Word
0x0183	INTERRUPT 2	Interrupt Address Word
0x0182	INTERRUPT 2	Interrupt Information Word
0x0181	INTERRUPT 1	Interrupt Address Word
0x0180	INTERRUPT 1	Interrupt Information Word

← The Interrupt Log Address Register points to this address after Interrupt 31 event occurs. Upon Interrupt 32 completion, IP Core logic reinitializes the log address pointer to 0x0180 before Interrupt 33 is processed.

← Interrupt Log Address Register is initialized by device logic to point to this address after hardware reset ($\overline{\text{RESET}}$) or software reset

Figure 7. Fixed Address Mapping for Interrupt Log Buffer

11.10. Hardware Interrupt Registers

11.10.1. Hardware Interrupt Enable Register (0x000F)

RAMERR	UNCRE	WDTX	LBFA	LBFB	MTTTRO	BCTTRO	Reserved	RTTTM	MTTTM	BCTTM	Reserved	RTAPF	Reserved	Reserved	Reserved	
0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	Reset
RW											R					Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

** Bits 4 - 0 are read-only and cannot be modified by host.*

11.10.2. Hardware Pending Interrupt Register (0x0006)

RAMERR	UNCRE	WDTX	LBFA	LBFB	MTTTRO	BCTTRO	Reserved	RTTTM	MTTTM	BCTTM	Reserved	RTAPF	RTIP	MTIP	BCIP	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R													R *			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

** Bits 2 - 0 are set for pending interrupts from RT, MT or BC*

11.10.3. Hardware Interrupt Output Enable Register (0x0013)

RAMERR	UNCRE	WDTX	LBFA	LBFB	MTTTRO	BCTTRO	Reserved	RTTTM	MTTTM	BCTTM	Reserved	RTAPF	Reserved	Reserved	Reserved	
0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	Reset
RW											R					Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

** Bits 4 - 0 are read-only and cannot be set by host.*

These three registers govern hardware interrupt behavior. As explained earlier, bits 2-0 in the Hardware Pending Interrupt Register are set whenever interrupt bits are set in the other three pending interrupt registers (RT, MT and BC). The table below first describes the common bits 15-3 in all three registers and then describes register-to-register differences for bits 2-0.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	RAMERR	R	0	ECC Correction. This bit is set when a single bit error in RAM is detected and corrected.
14	UNCRE	R	0	ECC Uncorrectable. This bit will be set to logic “1” when two or more bit errors in the same word in RAM are detected. Note that more than one RAM data error per word is **not correctable**.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13	WDTX			Transmitter Watchdog Timeout BUSA and BUSB transmitters each have a watchdog timer to prevent continuous transmission beyond 663µs. If WDTX is set in the “Hardware Interrupt Enable Register (0x000F)” , the corresponding bit will be set in the “Hardware Pending Interrupt Register (0x0006)” and the Interrupt Log Buffer is updated. If bit 13 is set in the “Hardware Interrupt Output Enable Register (0x0013)”, an interrupt is generated at the $\overline{\text{INT}}$ signal.
12 11	LBFA LBFB	R/W	0	Loopback Fail Bus A Interrupt (LBFA) − Valid only for RT Loopback Fail Bus B Interrupt (LBFB) − Valid only for RT For all words transmitted by the RT, the IP core checks MIL-STD-1553 word validity for the subsequently received/decoded word detected on the bus. This includes sync, encoding, bit count and parity checking. The last word in each message transmitted by the RT is also checked for data matching. The LBFA bit is set each time loop-back detects an invalid or mismatched word on Bus A. The LBFB bit is set each time loop-back detects an invalid or mismatched word on Bus B. **NOTE:** Once LBFA or LBFB are set by a loopback failure on Bus A or Bus B respectively, they remain set (persistent) until cleared by reading the “Hardware Pending Interrupt Register (0x0006)”.
10	MTTTRO	R/W	0	MT Time Tag Counter Rollover. The Bus Monitor time tag counter rolled over from full count to zero. Depending on options selected in the “Time Tag Counter Configuration Register (0x0039)”, the MT time count may be either 16 or 48 bits.
9	BCTTRO	R/W	0	BC Time Tag Counter Rollover. The Bus Controller time tag counter rolled over from full count to zero. Depending on options selected in the “Time Tag Counter Configuration Register (0x0039)”, the BC time count may be either 16 or 32 bits.
8	Reserved	R	0	This bit is not used.
7	RTTTM	R/W	0	RT Time Tag Match. The 16-bit Remote Terminal time tag counter incremented to a count matching the contained value in the RT Time Tag Reload / Match Register.
6	MTTTM	R/W	0	MT Time Tag Match. The Bus Monitor time tag counter incremented to a count matching the contained value in the MT Time Tag Match Registers.
5	BCTTM	R/W	0	BC Time Tag Match. The Bus Controller time tag counter incremented to a count matching the contained value in the BC Time Tag Match Register(s).
4	Reserved	R	1	This bit is not used.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
3	RTAPF	R	1	RT Terminal Address Parity Fail Interrupt. The Remote Terminal address and parity bits (latched into the RT Operational Status Register (see page 125) at rising edge of $\overline{\text{RESET}}$) do not exhibit odd parity (do not have an odd number of bits having logic 1 state).
For the Hardware Interrupt Enable Register and the Hardware Interrupt Output Enable Register only				
2 − 0	Reserved			Bits 2-0 cannot be written and read back 000.
For the Hardware Pending Interrupt Register only				
2	RTIP	R	0	RT Interrupt Pending. When this bit is high, one or more bits are set in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”. The host can read that register to determine the RT interrupt event(s).
1	MTIP	R	0	MT Interrupt Pending. When this bit is high, one or more bits are set in the “SMT Bus Monitor Pending Interrupt Register (0x0008)”. The host can read that register to determine the MT interrupt event(s).
0	BCIP	R	0	BC Interrupt Pending. When this bit is high, one or more bits are set in the BC Pending Interrupt Register. The host can read that register (0x0007) to determine the BC interrupt event(s).

11.11. Extended Configuration Register (0x004D)

Reserved - DO NOT USE												CIRBCST	INTBUSY	DPBTOFF	INTOFF	
X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	Reset
R/W																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This register contains four bits which provide global options for RT operation.

Bit No.	*Mnemonic*	*Function*
15 − 4	Reserved	Do not use. Setting these bits may cause unpredictable behavior.

Bit No.	*Mnemonic*	*Function*
3	CIRBCST	Circular Buffer Mode Gap Error / Broadcast Flag. When CIRBCST = 0, bit 13 GAP/BCAST in Receive and Transmit “Subaddress Message Information Words” on page 162 and Receive and Transmit “Mode Command Message Information Words” on page 166 is a **Gap Error flag**. When CIRBCST = 1, bit 13 GAP/BCAST in Receive and Transmit “Subaddress Message Information Words” on page 162 and Receive and Transmit “Mode Command Message Information Words” on page 166 is a **Broadcast flag**.
2	INTBUSY	Report terminal Busy status for RT message interrupts; set WASBUSY flag bit 9 in log buffer Interrupt Identification Word. See “Interrupt Log Buffer” on page 40. Setting INTBUSY = 1 causes the WASBUSY status bit 9 to be set in the RT Interrupt Identification Word (IIW) when an enabled interrupt event occurs with RT Busy status. Therefore, RT subaddresses or mode codes with enabled message interrupts can optionally report when Busy status applied during those events. For example, assume the RT Rx subaddress 1 Interrupt When Accessed event (IWA) is enabled. If RT subaddress 1 sees a legal, valid receive message but the RT is Busy, the IWA and WASBUSY flags are both set in the Interrupt Log Buffer’s written Interrupt Identification Word. If RT is not Busy, only the IWA flag is set in the written IIW. When INTBUSY is reset, terminal Busy status for interrupts is not reported; the WASBUSY status bit in the logged RT Interrupt Identification Word is always 0. **Note 1:** The Interrupt Log Buffer is updated only when enabled interrupt events occur. By itself, RT Busy status is not an interrupt-causing event. **Note 2:** INTBUSY bit 2 does not affect the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” on page 135; register bit 9 is reserved. Only WASBUSY bit 9 in the Interrupt Log Buffer Interrupt Identification Word (IIW) is affected.

Bit No.	*Mnemonic*	*Function*
1	DPBTOFF	Bit 1 DPBTOFF disables ping pong DPB pointer toggle when the received valid command is illegal, or when a message occurs with Busy status. When using ping pong buffers, the DPB buffer pointer never toggles for valid, legal messages ending in error. Setting DPBTOFF = 1 also disables ping pong DPB pointer toggle when the received valid command is illegal, or when a message occurs with Busy status. For RT subaddresses using ping pong data buffers (see “Ping-Pong Data Buffering” on page 170), the IP core alternates message data storage between Data Buffer A and Data Buffer B, on a message-by-message basis. The Descriptor Table Control Word DPB bit 10 indicates the data buffer to be used by the next-occurring message to this subaddress (see Section “18.4. Descriptor Table” on page 145). When the DPB bit is logic 0, the next message uses Data Buffer A; when DPB is logic 1, the next message uses Data Buffer B. Set DPBTOFF = 1 to prevent toggle of the Control Word DPB bit for Illegal or RT Busy messages, as well as valid, legal messages ending in error. The DPB pointer therefore remains static until the next successful message is received, which overwrites the Message Information Word and Time Tag Word in the current ping pong buffer location. Note that receive and transmit subaddresses may have both legal and illegal word counts, dictated by the Command Illegalization Table. For such subaddresses, DPB toggle only occurs when a supported legal word count message is transacted. When the DPBTOFF option bit is set to modify behavior for ping pong buffers, DPB toggle is disabled for valid messages that are illegal, or legal messages resulting in RT Busy or Message Error status. Important note: message data words in the “next-used” (designated active) buffer are NOT altered for incomplete (illegal, Message Error or RT Busy) messages. **However the buffer Message Information and Time Tag Words are updated in that message data buffer so the host can detect when such messages occur.** Bits 10:8 in the buffer Message Information Word indicate Message Error, Busy and/or Illegal status. **When any of these 3 bits are set, the accompanying data should always be disregarded (whether or not the DPBTOFF option is used).** To maintain data integrity, the primary benefit of DPBTOFF = 1 is that the complemented DPB pointer always indicates the last-transacted “good” data set. For example if DPB is logic 0, the last successful message used Data Buffer B. The default condition after power-on reset in register 0x4D contains 0x0000. Thus configuration bit DPBTOFF is logic-0. For this case, the Control Word DPB bit toggles after completion of error-free messages (expected), but also illegal commands and messages resulting in Message Error or Busy status.

Bit No.	Mnemonic	Function
0	INTOFF	Suppress $\overline{\text{INT}}$ interrupt signal assertion for enabled RT message interrupts when the command is illegal or the message results in RT Busy status. RT Interrupt Registers are described on page 135. Globally enable RT interrupt types (MERR, IWA, IBR) by setting bits in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”. Enable these interrupts for individual receive or transmit subaddresses (or mode commands) by setting bits in their Descriptor Table Control Words. When an enabled RT interrupt event occurs, the corresponding “type” bit is set in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” and “Interrupt Log Buffer” on page 40 is updated. In addition, the $\overline{\text{INT}}$ output signal is normally asserted (low) if the corresponding bit is also set in the “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)”. This INTOFF option bit modifies interrupt output behavior. The INTOFF option bit prevents nuisance $\overline{\text{INT}}$ signal assertion when the command is illegal or when the message occurs with RT Busy status. Note that receive and transmit subaddresses may have both legal and illegal word counts, dictated by the Command Illegalization Table. For such subaddresses, an interrupt is only generated when a supported legal word count message is transacted.

11.12. Time Tag Counter Configuration

Each IP core (RT, BC or MT) has an independent time tag counter used for time-stamping messages. In the “Time Tag Counter Configuration Register (0x0039)”, bits 2-0 select the clock source for the RT and time tag counter. The same clock source is shared by the Bus Controller. The host controls the free-running RT time tag counter using bit pairs 9-8 in the “Time Tag Counter Configuration Register (0x0039)”. Here is a summary of host-initiated operations involving the RT time tag counter:

a. Clearing the 16-bit RT time tag count to 0x0000.

b. Copying the RT Time Tag Utility Register value (see page 133) into the 16-bit RT time tag counter.

c. Copying the current 16-bit RT time tag count value into the RT Time Tag Utility Register.

The bus controller (BC) can operate using either a 16- or 32-bit time tag counter, selected using register bit 3, BCTT32, in the “Time Tag Counter Configuration Register (0x0039)”. The BC time tag counter clock source is selected using register bits 2-0. This common clock source is shared by the BC, and RT. Bit pair 13-12 is used for clearing BC time tag counter, loading the counter with a 16- or 32-bit value contained in the “Bus Controller (BC) Time Tag Utility Register (0x0045)” on page 93, or writing the current 16- or 32-bit BC time tag counter value to the “Bus Controller (BC) Time Tag Utility Register (0x0045)”.

The free-running BC time tag counter can be reset to zero, loaded with an arbitrary value, or the current count can be captured. In 32-bit time tag mode, the full count is captured by simultaneously loading two utility registers. Writing bits 13-12 in the Time Tag Counter Configuration Register initiates these operations. Here is a summary of host-initiated operations involving the BC time tag counter:

a. Clearing a 16- or 32-bit BC time tag count, whichever is enabled.

b. When 16-bit BC time tag count is enabled,

- loading the 16-bit BC time tag counter with the 16-bit value contained in the “Bus Controller (BC) Time Tag Utility Register (0x0045)”
- capturing the current 16-bit BC time tag counter value to the “Bus Controller (BC) Time Tag Utility Register (0x0045)”

c. When 32-bit BC time tag count is enabled,

   - loading the 32-bit BC time tag counter with the 32-bit value contained in the register pair “Bus Controller (BC) Time Tag Utility Register (0x0045)” and “Bus Controller (BC) Time Tag Utility High Register (0x0046)”.
   - capturing the current 32-bit BC time tag counter value to the register pair “Bus Controller (BC) Time Tag Utility Register (0x0045)” and “Bus Controller (BC) Time Tag Utility High Register (0x0046)”

The bus monitor (MT) can operate using either a 16- or 48-bit time tag counter, selected using MT Configuration Register bits 1-0. When using 16-bit resolution, one register is adequate for holding time tag values. When using 48-bit time tag count resolution, three 16-bit registers are needed for each stored time tag count. The MT time tag counter clock source is selected using bits 7-5 in the “Time Tag Counter Configuration Register (0x0039)”. The MT time tag clock source is separate from the source shared by the BC and RT.

The free-running MT time tag counter can be reset to zero, loaded with an arbitrary value, or the current count can be captured. In 48-bit time tag mode, the full count is captured by simultaneously loading three utility registers. Writing bits 15-14 in the “Time Tag Counter Configuration Register (0x0039)” initiates these operations.

Here is a summary of host-initiated operations involving the MT time tag counter:

a. Clearing a 16- or 48-bit MT time tag count, whichever is enabled.

b. When 16-bit MT time tag count is enabled,

   - loading the 16-bit MT time tag counter with the 16-bit value contained in the “SMT Bus Monitor Time Tag Utility Register (0x003D)”
   - capturing the current 16-bit MT time tag counter value to the “SMT Bus Monitor Time Tag Utility Register (0x003D)”

c. When 48-bit MT time tag count is enabled,

   - loading the 48-bit MT time tag counter with the 48-bit value contained in the register triplet “SMT Bus Monitor Time Tag Utility Register (0x003D)”, “SMT Bus Monitor Time Tag Utility Mid Register (0x003E)” and “SMT Bus Monitor Time Tag Utility High Register (0x003F)”.
   - capturing the current 48-bit MT time tag counter value to the register triplet “SMT Bus Monitor Time Tag Utility Register (0x003D)”, “SMT Bus Monitor Time Tag Utility Mid Register (0x003E)” and “SMT Bus Monitor Time Tag Utility High Register (0x003F)” when 48-bit MT time tag count is enabled

Host interrupts can be generated when any of the four time tag counters in the IP core reach preset values contained in Time Tag Match registers. Refer to the Section 11.6.

11.13. Time Tag Counter Configuration Register (0x0039)

MTTTA1	MTTTA0	BCTTA1	BCTTA0	Reserved	Reserved	RTTTA1	RTTTA0	MTTCK3	MTTCK2	MTTCK1	MTTCK0	BCTT32	TTCK2	TTCK1	TTCK0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit read-write register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of MTRESET or RTRESET register bits.

When written, register bits 15-8 work in pairs to initiate a particular action, such as clearing or loading one of these counters. When written, register bits 15-8 self reset to zero after initiating the assigned action. Thus, bits 15-8 always read logic 0. Register bits 7-0 are used for configuring the various time tag counters in the IP core. These bits will read back the last value written by the host.

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>15
14</td><td>MTTTA1
MTTTA0</td><td>R/W</td><td>0</td><td>
MT Time Tag Action Bits 1-0.
After performing the action below, these host-written bits self reset to 00:
<table>
<tr><th>Bits 15-14</th><th>Action</th></tr>
<tr><td>00</td><td>Do Nothing.</td></tr>
<tr><td>01</td><td>Reset 16- or 48-bit MT Time Tag count to zero.</td></tr>
<tr><td>10</td><td>Load the 16- or 48-bit value from SMT Time Tag Utility Register(s) into the SMT Time Tag Count register(s).</td></tr>
<tr><td>11</td><td>Capture current 16- or 48-bit value in the SMT Time Tag Count Register(s) into the SMT Time Tag Utility Register(s).</td></tr>
</table>
If the MT is using 16-bit time tag, the SMT Time Tag Counter uses a single register address, 0x003A. The SMT Time Tag Utility Register used for the load and capture operations is register address 0x003D.
If the MT is using 48-bit time tag, the SMT Time Tag Counter uses three register addresses. The High-Mid-Low words are found at 0x003C, 0x003B and 0x003A respectively. The triplet of SMT Time Tag Utility Register Triplet used for the load and capture operations is located at register addresses 0x003F (High), 0x003E (Mid) and 0x003D (Low).
Bits 1-0 in the “SMT Configuration Register (0x0029)” select SMT 16-bit or 48-bit time tag counting:
<table>
<tr><th>SMT Configuration Register, bits 1-0</th><th>Time Tag Mode</th></tr>
<tr><td>01</td><td>16-bit time tag</td></tr>
<tr><td>11</td><td>48-bit time tag</td></tr>
</table>
When the MT is operating with 48-bit time tag, the recorded Command Buffer entry for each 1553 message has eight 16-bit words. When operating with 16-bit time tag, the recorded Command Buffer entry for each 1553 message has four 16-bit words.
</td></tr>
</table>

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th colspan="2">Function</th></tr>
<tr><td rowspan="7">13
12</td><td rowspan="7">BCTTA1
BCTTA0</td><td rowspan="7">R/W</td><td rowspan="7">0</td><td colspan="2">BC Time Tag Action Bits 1-0.
After performing the action below, these host-written bits self reset to 00:</td></tr>
<tr><td><b><i>Bits 13-12</i></b></td><td><b><i>Action</i></b></td></tr>
<tr><td>00</td><td>Do Nothing.</td></tr>
<tr><td>01</td><td>Reset 16- or 32-bit BC Time Tag count to zero.</td></tr>
<tr><td>10</td><td>Load the 16- or 32-bit value from BC Time Tag Utility Register(s) into the BC Time Tag Count Register(s).</td></tr>
<tr><td>11</td><td>Load the 16- or 32-bit value from the BC Time Tag Count Register(s) into the BC Time Tag Utility Register(s).</td></tr>
<tr><td colspan="2">If BCTT32 register bit 3 equals 0, the BC is using 16-bit time tag. The BC Time Tag Counter uses a single register address, 0x0043. The “Bus Controller (BC) Time Tag Utility Register (0x0045)” is used for these operations.
If BCTT32 register bit 3 equals 1 the BC is using 32-bit time tagging, so the BC Time Tag Counter requires two 16-bit register addresses. The High and Low BC time tag counter words are found at 0x0044 and 0x0043 respectively. The pair of BC Time Tag Utility Registers used for timer operations are located at register addresses 0x0046 (High word) and 0x0045 (Low word).</td></tr>
<tr><td>11 − 10</td><td>Reserved</td><td>R</td><td>0</td><td colspan="2">These bits are not used.</td></tr>
<tr><td rowspan="5">9
8</td><td rowspan="5">RTTTA1
RTTTA0</td><td rowspan="5">R/W</td><td rowspan="5">0</td><td colspan="2">RT Time Tag Action Bits 1-0.
After performing the RT time tag counter action below, these host-written bits self reset to 00:</td></tr>
<tr><td><b><i>Bits 9-8</i></b></td><td><b><i>Action</i></b></td></tr>
<tr><td>00 or 11</td><td>Do Nothing.</td></tr>
<tr><td>01</td><td>Reset 16-bit RT Time Tag count to zero.</td></tr>
<tr><td>10</td><td>Load the 16-bit value from the RT Time Tag Utility Register (0x004A) into the RT Time Tag Counter (0x0049).</td></tr>
<tr><td rowspan="3">7
6
5
4</td><td rowspan="3">MTTCK3
MTTCK2
MTTCK1
MTTCK0</td><td rowspan="3">R/W</td><td rowspan="3">0</td><td colspan="2">MT Time Tag Clock Selection Bits 7-4.
These bits select the clock source for the MT Time Tag Counters from the following options:</td></tr>
<tr><td><b><i>Bits 7-6-5-4</i></b></td><td><b><i>MT Time Tag Counter Clock Source</i></b></td></tr>
<tr><td>0-0-0-0
0-0-0-1
0-0-1-0
0-0-1-1
0-1-0-0
0-1-0-1
0-1-1-0
0-1-1-1
1-x-x-x</td><td>Time Tag Counter Disabled.
External clock provided at the MTTCLK input signal.
Internally generated 2μs clock.
Internally generated 4μs clock.
Internally generated 8μs clock.
Internally generated 16μs clock.
Internally generated 32μs clock.
Internally generated 64μs clock.
Internally generated 100ns clock.</td></tr>
</table>

Bit No.	Mnemonic	R/W	Reset	Function	
3	BCTT32	R/W	0	BC Time Tag 32-Bit Count Enable. When the BCTT32 bit equals 0, the BC time tag counter is 16 bits. When the BCTT32 bit equals 1, the BC time tag counter is 32 bits.	
2 1 0	TTCK2 TTCK1 TTCK0	R/W	0	BC and RT Time Tag Clock Selection Bits 2-0. These three bits select the clock source for the BC and RT Time Tag Counters from the following options:	
				Bits 2-1-0	***Time Tag Counter Clock Source***
				0-0-0	Time Tag Counter Disabled.
				0-0-1	External clock provided at TTCLK input signal.
				0-1-0	Internally generated 2μs clock.
				0-1-1	Internally generated 4μs clock.
				1-0-0	Internally generated 8μs clock.
				1-0-1	Internally generated 16μs clock.
				1-1-0	Internally generated 32μs clock.
				1-1-1	Internally generated 64μs clock.

12. BUS CONTROLLER − CONFIGURATION AND OPERATION

The HI-6300 can operate as an autonomous MIL-STD-1553 Bus Controller (BC), requiring minimal host processor support. All MIL-STD-1553B error-checking is automatically performed, including RT response time, Manchester II encoding, sync type, bit count, word parity, word count, responding RT address, and detection of the full range of possible error conditions encountered during BC operation. The IP core implements all MIL-STD-1553B message formats. Message format is configurable on a message by message basis. Each message is individually programmable for command type. Individual messages can be programmed for automatic retries on either bus, and interrupt requests may be enabled or disabled.

The Bus Controller provides a flexible means for scheduling major and minor frames, allowing insertion of asynchronous messages during frame execution. Upon error, individual messages can be programmed for one or two automatic retries, and the BC can switch buses before retry occurs. Message data is separated from control and status data, to serve the needs for double buffering in RAM and bulk data transfers.

Before Bus Controller operation can begin, the BCENA bit 12 in “Master Configuration Register 1 (0x0000)” on page 30 must be logic 1 to allow BC operation. All Bus Controller operational registers must be properly configured (see “Bus Controller Register Description” on page 78). The BC Instruction List in RAM must be initialized to define message sequencing and conditional execution, and finally the host must assert BCSTRT bit 13 in the “Master Configuration Register 1 (0x0000)” to initiate execution of Instruction List op codes. The following pages provide the necessary details for successful Bus Controller operation.

Figure 8 shows the registers and RAM resources utilized by the Bus Controller. All Bus Controller registers are fully described in “Bus Controller Register Description” on page 78.

Initial control of BC message sequencing involves the “Bus Controller (BC) Instruction List Pointer (0x0034)”. Before BC execution begins, the instruction list starting address is copied from the “Start Address Register for Bus Controller (BC) Instruction List (0x0033)”. Once message sequencing is underway, the “Bus Controller (BC) Instruction List Pointer (0x0034)” is updated by the BC control logic.

The BC Instruction List in RAM comprises a series of 2-word entries, an instruction op code followed by a parameter word. While processing messages, the BC control logic fetches and executes the instruction op code referenced by the “Bus Controller (BC) Instruction List Pointer (0x0034)” from the BC Instruction List. The pointer parameter, referencing the first word in the Message Control/Status Block, must have the form 0xHHH8 or 0xHHH0, where each H represents a hex character, 0-9 or A-F. If the individual message is RT-to-RT, the address must have the form 0xHHH0.

Each op code word in the BC Instruction List has the format:

Odd Parity																
X	X	X	X	X	X	0	1	0	1	0	X	X	X	X	X	
P	Op Code Field					Validation Field					Condition Code					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Bus Controller execution stops immediately if the BC logic fetches an op code word having one or more of these error conditions:

- Bit 15 contains even parity
- Bits 14-10 contain an undefined op code
- Validation field bits 9-5 do not equal 01010

If enabled in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, a BCTRAP interrupt occurs when execution stops because of an illegal op code.

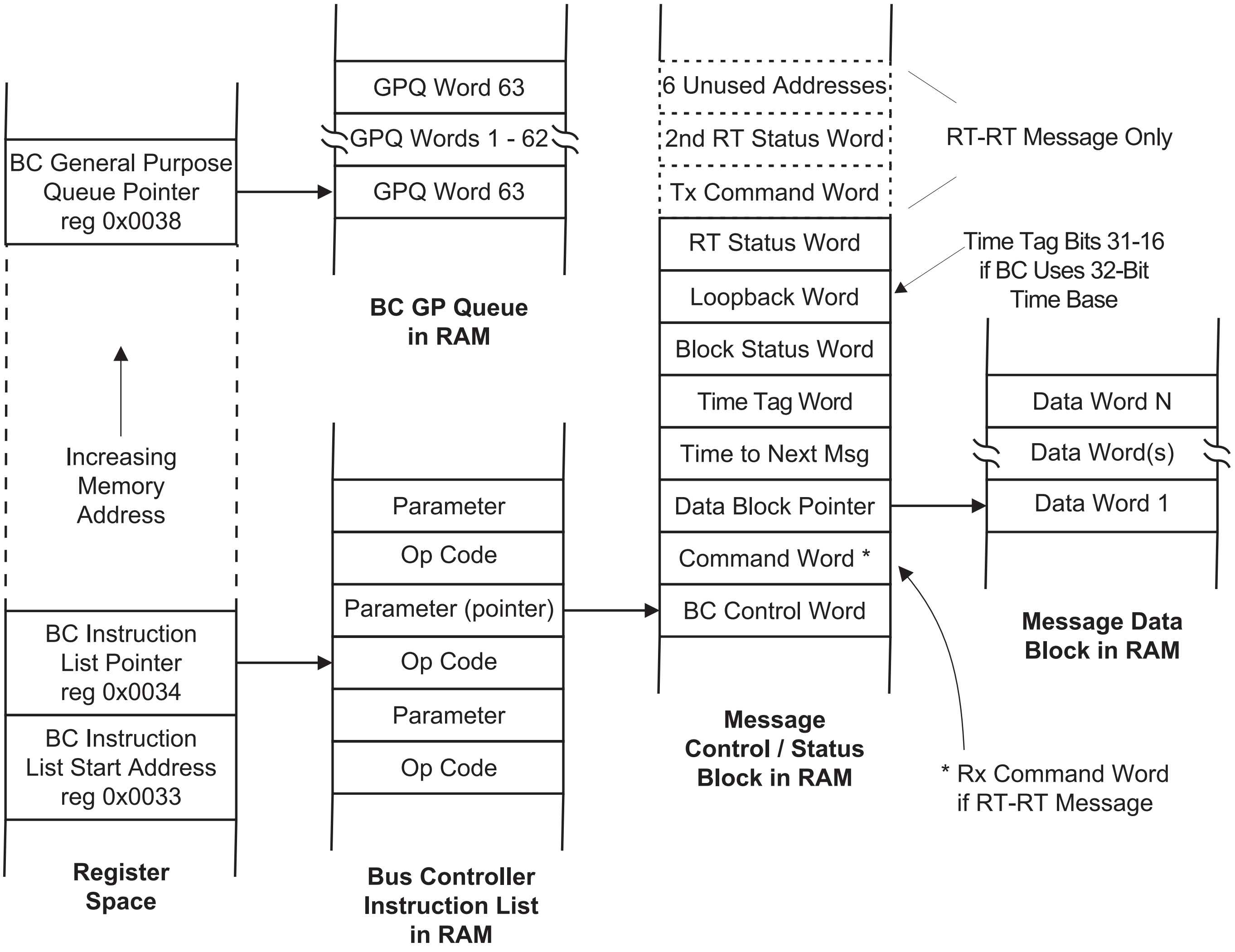


Figure 8. Bus Controller Message Sequence Structures

12.1. Bus Controller Condition Codes

For most op codes, execution is conditional, depending on evaluation of the Condition Code field. Condition Code bits 3-0 define the specific condition. Condition Code bit 4 identifies the required outcome for op code execution, true or false, following evaluation of Condition Code bits 3-0. The host has read-only access to the BC condition codes by reading the “Bus Controller (BC) Condition Code Register (Read 0x0037)”.

Eight of the condition codes (1000 through 1111) are set or cleared based on the outcome of the most recent message. The remaining eight codes are General Purpose Condition Codes, GP0 through GP7. Three processes affect values of the General Purpose Condition Code bits: (a) they may be toggled, set or cleared when the BC logic executes a FLG (GP Flags Bits) op code. (b) they may be toggled, set or cleared when the host writes the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. (c) only GP0 and GP1 may be set or cleared when the BC logic executes a CMT (Compare Message Timer) op code, or a CFT (Compare Frame Timer) op code. The sixteen BC Condition Codes are summarized in Table 11.

Table 11. Bus Controller Condition Code Table

Code Bit 3-0	Condition Code *Bit 4 = 0*	Inverse *Bit 4 = 1*	Function
0x0	LT / GP0	GT-EQ / $\overline{GP0}$	Less Than or GP0 Flag. This flag may be toggled, set or cleared when the BC logic executes a FLG (General Purpose Flags Bits) op code, or when the host writes the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. This flag is also set or cleared based on the evaluation of a CMT (Compare Message Timer) or CFT (Compare Frame Timer) instruction op code:
	LT / GP0 = 1	GT-EQ / $\overline{GP0}$ = 0	If CMT parameter < BC Time to Next Message Reg If CFT parameter < BC Frame Time Remaining Reg
	LT / GP0 = 0	GT-EQ / $\overline{GP0}$ = 1	If CMT parameter = BC Time to Next Message Reg If CFT parameter = BC Frame Time Remaining Reg
	LT / GP0 = 0	GT-EQ / $\overline{GP0}$ = 1	If CMT parameter > BC Time to Next Message Reg If CFT parameter > BC Frame Time Remaining Reg
	LT / GP0 = 1 or 0 based on compared values	GT-EQ / $\overline{GP0}$ = 0 or 1 based on compared values	If CMT parameter ≠ BC Time to Next Message Reg If CFT parameter ≠ BC Frame Time Remaining Reg
0x1	EQ / GP1	NE / $\overline{GP1}$	Equal To or GP1 Flag. This flag may be toggled, set or cleared when the BC logic executes a FLG (General Purpose Flags Bits) op code, or when the host writes the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. This flag is also set or cleared based on the evaluation of a CMT (Compare Message Timer) or CFT (Compare Frame Timer) instruction op code:
	EQ / GP1 = 0	NE / $\overline{GP1}$ = 1	If CMT parameter < BC Time to Next Message Reg If CFT parameter < BC Frame Time Remaining Reg
	EQ / GP1 = 1	NE / $\overline{GP1}$ = 0	If CMT parameter = BC Time to Next Message Reg If CFT parameter = BC Frame Time Remaining Reg
	EQ / GP1 = 0	NE / $\overline{GP1}$ = 1	If CMT parameter > BC Time to Next Message Reg If CFT parameter > BC Frame Time Remaining Reg
	EQ / GP1 = 0	NE / $\overline{GP1}$ = 1	If CMT parameter ≠ BC Time to Next Message Reg If CFT parameter ≠ BC Frame Time Remaining Reg

Code Bit 3-0	**Condition Code** *Bit 4 = 0*	**Inverse** *Bit 4 = 1*	**Function**
0x2	GP2	$\overline{\text{GP2}}$	General Purpose Flag Bits 2-7. These flags may be toggled, set or cleared when the BC logic executes a FLG (General Purpose Flags Bits) op code, or when the host writes the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”.
0x3	GP3	$\overline{\text{GP3}}$	
0x4	GP4	$\overline{\text{GP4}}$	
0x5	GP5	$\overline{\text{GP5}}$	
0x6	GP6	$\overline{\text{GP6}}$	
0x7	GP7	$\overline{\text{GP7}}$	
0x8	NORESP	RESP	No Response Flag. This flag is set when an RT failed to respond to a command, or responded later than the BC No Response Timeout programmed using bits 15-14 in the “BC (Bus Controller) Configuration Register (0x0032)”.
0x9	FMTERR	$\overline{\text{FMTERR}}$	Format Error Flag. This flag is set when the received response to the last message had one or more violations to MIL-STD-1553B validation criteria, including problems with sync, encoding, bit count, parity, or word count. This flag is also set when the received RT Status Word response from the last message contained an incorrect RT address field.
0xA	GOODBLOCK	$\overline{\text{GOODBLOCK}}$	Good Data Block Transfer. Reflecting status for the last 1553 message, this flag is set after completion of error-free RT-to-BC transfers, RT-to-RT transfers, or transmit mode code commands with data. This flag is reset after invalid messages, or after completion of BC-to-RT transfers, receive mode code commands with data, or mode code commands without data. This flag may be used to determine when the transmit aspect of an RT-to-RT transfer is error-free.
0xB	MSKSTATSET	$\overline{\text{MSKSTATSET}}$	Masked Status Set. Reflecting status for the last 1553 message, this flag is set when one or both of the following conditions occurred: • In the message BC Control Word, at least one of the Status Mask bits 14-9 are logic 0, but the corresponding bit or bits are set in the received RT Status Word. When the Reserved Bits Mask (message BC Control Word bit 9) is logic 0, this flag is set when any or all of the three reserved bits are set in the received RT Status Word. • In the “BC (Bus Controller) Configuration Register (0x0032)”, the BCR (broadcast command received) Mask Enable bit 0 is logic 1. The Mask Broadcast bit 5 in the message BC Control Word is logic 0, and the BCR (Broadcast Command Received) bit 4 is logic 1 in the received RT Status Word.

Code Bit 3-0	Condition Code *Bit 4 = 0*	Inverse *Bit 4 = 1*	Function
0xC	BADMSG	GOODMSG	Bad Message. Reflecting status for the last 1553 message, the Bad Message flag is set for Format Error, No Response error, or Loopback error. A Status Set condition has no effect on the Bad Message condition code.
0xD	1RETRY	$\overline{\text{1RETRY}}$	1 Retry. If Condition Code bits 3:0 = 0xD and bit 4 = 0, one or two message retries is indicated. If Condition Code bits 3:0 = 0xD and bit 4 = 1, zero message retries is indicated.
0xE	2RETRY	Undefined	2 Retries. If Condition Code bits 3:0 = 0xE and bit 4 = 0, two message retries is indicated. Condition Code bits 3:0 = 0xE and bit 4 = 1 is undefined.
0xF	ALWAYS	NEVER	Always. The ALWAYS bit is set (Condition Code bit 4 = 0) to designate an instruction op code as unconditional. The NEVER bit is set (Condition Code bit 4 = 1) to designate an instruction op code as NOP (no operation).

12.2. Bus Controller Instruction Op Codes

In the BC Instruction List, each op code word contains a 5-bit instruction field that spans bits 14-10. The instruction op codes are described in Table 12. Four instructions execute unconditionally, without evaluating the condition code test. For these instructions, the Condition Code field is “don’t care”.

The four unconditional instruction op codes are:

1. CMT (Compare Message Timer)
2. CFT (Compare Frame Timer)
3. FLG (General Purpose Flag Bits)
4. XQF (Execute and Flip)

All other instruction op codes execute conditionally. They execute only if evaluation of the Condition Code tests true, logic 1. If the Condition Code field tests false, the BC Instruction List Pointer skips to the next instruction op code in the BC Instruction List.

Many instruction op codes utilize the following parameter word in the BC Instruction List. Depending on the op code, the parameter may be a RAM address, a time value, an interrupt bit pattern, an argument that sets or clears General Purpose Flag bits, or an immediate value. For some op codes, the parameter word is not used and is therefore “don’t care.” For an XEQ (execute message) instruction, the parameter is a RAM address pointer referencing the start of the message Control/Status Block.

These instructions control program execution: Halt, Jump, Subroutine Call and Subroutine Return. Subroutine calls can be nested 8 levels deep. If BC Call Stack overflow or underflow occurs, IP core logic generates a CSTKERR (Call Stack Pointer Error) interrupt, if enabled. Other host interrupts are generated under program control using the Interrupt Request instruction. In this case, a 4-bit user-defined interrupt code is written to the BC Interrupt Request Bits 3-0 in the "Bus Controller (BC) Pending Interrupt Register (0x0007)".

Other instructions perform various duties: set, reset or toggle General Purpose Flag bits; load the Time Tag counter; load the Frame Time counter; begin a new BC frame; wait for external trigger, then start a new BC frame; evaluate remaining Frame Time; or evaluate time to next message.

Table 12. Bus Controller Instruction Op Codes

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
XEQ	Execute Message *Conditional*	0x01	RAM Address for Message Control/ Status Block	If the Condition Code evaluates True, execute the message at the parameter-specified Message Control Status Block address. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. At the start of XEQ message execution, if the fourth word in the Message Control/Status Block is nonzero, it is copied to the BC Time to Next Message Register, and message timer begins decrementing. The BC message sequencer does not fetch the next instruction op code until the message timer reaches zero. Regarding Condition Codes used with XEQ: • If using LT, GT-EQ, EQ and NE (which are only modified by the IP core upon completion of CMT or CFT op codes) the host must not change the value of the shared function GP0 or GP1 flag bit during execution of the contingent message. • If using GP Flag Bit status (GP0 through GP7) to enable a message, host must not alter the tested GP Flag bit during execution of a contingent message. • The ALWAYS and NEVER Condition Codes may be used with XEQ. The following Condition Codes must not be used with XEQ: BADMSG, RETRY1 or RETRY2, NORESP, MSKSTATSET, FMTERR or GOODBLOCK.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
XQG	Execute Message and Go *Conditional*	0x16	RAM Address for Message Control/ Status Block	If the Condition Code evaluates True, execute the message at the parameter-specified Message Control Status Block address. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. Regarding Condition Codes used with XQG: • If using LT, GT-EQ, EQ and NE (which are only modified by the IP core upon completion of CMT or CFT op codes) the host must not change the value of the shared function GP0 or GP1 flag bit during execution of the contingent message. • If using GP Flag Bit status (GP0 through GP7) to enable a message, host must not alter the tested GP Flag bit during execution of a contingent message. • The ALWAYS and NEVER Condition Codes may be used with XQG. The following Condition Codes must not be used with XQG: BADMSG, RETRY1 or RETRY2, NORESP, MSKSTATSET, FMTERR or GOODBLOCK. At the start of XQG message execution, if the fourth (Time to Next Message) word in the Message Control/Status Block is nonzero, it is copied to the “Bus Controller (BC) Time To Next Message Register (0x0036)”, and this message timer begins countdown. Completion of an XQG message may occur while message timer countdown continues. Unlike XEQ, the XQG op code does not wait for the decrementing message timer to hit 0 before fetching the next instruction op code. As long as op codes following XQG do not execute a 1553 message, each op code is performed after fetch. Upon reaching a following XEQ, XQG, XQF or XFG execute-message instruction, transaction of its 1553 message does not begin until Time to Next Message count reaches 0. Thus, programmed 1553 message timing is maintained, while allowing execution of non-message instruction op codes.
JMP	Jump *Conditional*	0x02	RAM Address within the BC Instruction List	If the Condition Code evaluates True, jump to the parameter-specified instruction op code in the BC Instruction List. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
CAL	Call Subroutine *Conditional*	0x03	RAM Address within the BC Instruction List	If the Condition Code evaluates True, push address of the next instruction op code onto the 8-level BC Call Stack, then jump to the parameter-specified instruction op code in the BC Instruction List. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
RTN	Return from Subroutine *Conditional*	0x04	Not Used (Don’t Care)	If the Condition Code evaluates True, pop the top address from the BC Call Stack, then jump to the popped instruction op code address in the BC Instruction List. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
INT	Interrupt Request *Conditional*	0x06	4-Bit Interrupt Pattern, 0x000N	If the Condition Code evaluates True, generate a host interrupt by writing the parameter-specified 4-bit value N to bits 8-5 in the “Bus Controller (BC) Pending Interrupt Register (0x0007)”. Otherwise, continue execution at the next op code in the BC Instruction List. Note: no interrupt is generated if N = 0
HLT	Halt *Conditional*	0x07	Not Used (Don’t Care)	If the Condition Code evaluates True, stop execution of the BC Instruction List until a new BC Start is issued by the host. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
DLY	Delay *Conditional*	0x08	Delay Time Value (1 μs per LSB resolution)	If the Condition Code evaluates True, initiate a delay equal to the parameter-specified value. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. If the Time-to-Next Message counter is in use, the DLY parameter has higher priority than the count for an unfinished Message Timer delay.
WFT	Wait until Frame Timer Equals 0 *Conditional*	0x09	Not Used (Don’t Care)	If the Condition Code evaluates True, stop BC Instruction List execution until the BC Frame Time Counter decrements to 0. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
CFT	Compare to Frame Timer *Conditional*	0x0A	Time Value (100 μs / LSB resolution)	Compare the parameter-specified Time Value to the “Bus Controller (BC) Frame Time Remaining Register (0x0035)”. Set and clear LT and EQ bits 1 and 0 in the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
CMT	Compare to Message Timer Unconditional	0x0B	Time Value (1 µs per LSB resolution)	Compare the parameter-specified Time Value to the “Bus Controller (BC) Time To Next Message Register (0x0036)”. Set and clear LT and EQ bits 1 and 0 in the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. When CMT is preceded by an XEQ or XQF instruction, the “Bus Controller (BC) Time To Next Message Register (0x0036)” value used for comparison is always 0x0000 because the message timer (initiated by XEQ or XQF) decrements to zero before fetching the CMP instruction. In this case, a CMT with non-zero parameter word always sets GT-EQ and NE and always resets LT and EQ flags. When CMT is preceded by an XEQ or XQF instruction, a CMT with parameter word 0x0000 always sets GT-EQ and EQ and always resets LT and NE flags. Further, the CMT op code will never set LT and NE, while clearing GT-EQ and EQ flags.
LTT	Load Time Tag Counter *Conditional*	0x0D	Time Value (Resolution is programmed by bits 2-0 in the “Time Tag Counter Configuration Register (0x0039)”)	If the Condition Code evaluates True, load the “Bus Controller (BC) Time Tag Counter (0x0043)” with the parameter-specified Time Value. This represents bits 15-0 when the BC operates with 32-bit time tag. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. Read LTH regarding LTT-LTH sequences.
LTH	Load Time Tag Counter High *Conditional* LTH only applies when 32-bit time base is enabled	0x18	Time Value (Resolution is programmed by bits 2-0 in the “Time Tag Counter Configuration Register (0x0039)”)	If the Condition Code evaluates True, load the “Bus Controller (BC) Time Tag Counter High (0x0044)” with the parameter-specified Time Value. This represents bits 31-16 when BC operates with 32-bit time tag. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. If LTH immediately follows an LTT instruction, all 32 counter bits are loaded simultaneously. If LTH is not preceded by an LTT instruction, time count bits 15-0 in “Bus Controller (BC) Time Tag Counter (0x0043)” will be cleared to 0x0000 when bits 31-16 are written to register “Bus Controller (BC) Time Tag Counter High (0x0044)”. This instruction is only allowed when BCTT32 bit 3 is logic 1 in the “Time Tag Counter Configuration Register (0x0039)”, selecting 32-bit time base operation for the BC and RT. If BCTT32 is logic 0, this instruction stops instruction list execution, and generates an illegal instruction BCTRAP interrupt, if enabled.
LFT	Load Frame Timer *Conditional*	0x0E	Time Value (100 µs / LSB resolution)	If the Condition Code evaluates True, load the “Bus Controller (BC) Frame Time Remaining Register (0x0035)” with the parameter-specified Time Value. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
SFT	Start Frame Timer *Conditional*	0x0F	Not Used (Don't Care)	If the Condition Code evaluates True, start decrementing the "Bus Controller (BC) Frame Time Remaining Register (0x0035)". Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
PTT	Push Time Tag Count *Conditional*	0x10	Not Used (Don't Care)	If the Condition Code evaluates True, push the value in the "Bus Controller (BC) Time Tag Counter (0x0043)" onto the "Bus Controller General Purpose Queue". Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
PTH	Push Time Tag Count High *Conditional* PTH only applies when 32-bit time base is enabled	0x19	Not Used (Don't Care)	If the Condition Code evaluates True, push the value in the "Bus Controller (BC) Time Tag Counter High (0x0044)" onto the "Bus Controller General Purpose Queue". This represents bits 31-16 when BC operates with 32-bit time tag. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. This instruction is only allowed when BCTT32 bit 3 is logic 1 in the "Time Tag Counter Configuration Register (0x0039)", selecting 32-bit time base operation for the BC and RT. If BCTT32 is logic 0, this instruction stops instruction list execution, and generates an illegal instruction BCTRAP interrupt, if enabled.
PTB	Push Time Tag Count Both *Conditional* PTB only applies when 32-bit time base is enabled	0x1A	Not Used (Don't Care)	If the Condition Code evaluates True, push the value in the "Bus Controller (BC) Time Tag Counter High (0x0044)" and then push the value in the "Bus Controller (BC) Time Tag Counter (0x0043)" onto the "Bus Controller General Purpose Queue". (Both words are fetched simultaneously but pushed serially.) Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. This instruction is only allowed when BCTT32 bit 3 is logic 1 in the "Time Tag Counter Configuration Register (0x0039)", selecting 32-bit time base operation for the BC and RT. If BCTT32 is logic 0, this instruction stops instruction list execution, and generates an illegal instruction BCTRAP interrupt, if enabled.
PBS	Push Block Status Word *Conditional*	0x11	Not Used (Don't Care)	If the Condition Code evaluates True, push the value of the Block Status Word from the most recent message onto the "Bus Controller General Purpose Queue". Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
PSI	Push Immediate Value *Conditional*	0x12	Immediate Value	If the Condition Code evaluates True, push the parameter-specified immediate value onto the "Bus Controller General Purpose Queue". Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
PSM	Push Indirect *Conditional*	0x13	Memory Address	If the Condition Code evaluates True, push the value stored at the parameter-specified address onto the “Bus Controller General Purpose Queue”. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
WTG	Wait for External Trigger *Conditional*	0x14	Not Used (Don’t Care)	If the Condition Code evaluates True, wait for a rising edge (logic 0 to 1 transition) on the BCTRIG signal before continuing execution at the next op code in the BC Instruction List Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.
XQF	Execute and Flip *Unconditional*	0x15	RAM Address for Message Control/ Status Block	Unconditionally execute the message at the parameter-specified Message Control/Status Block Address. **At message completion**, if the Condition Code evaluates True, then toggle bit 4 of the Message Control/Status Block Address, and store the new Message Control/Status Block Address as the updated value of the parameter following the XQF instruction op code. As a result, the next time this address in the BC Instruction List is executed, the processed Message Control/ Status Block resides at the updated address (old address XOR 0x0010) instead of the old address. Otherwise (Condition Code Evaluates False) the value of the Message Control/Status Block Address parameter is not changed. At the start of XQF message execution, if the fourth word in the Message Control/Status Block is nonzero, it is copied to the “Bus Controller (BC) Time To Next Message Register (0x0036)”, and message timer begins decrementing. The BC message sequencer does not fetch the next instruction op code until this message timer reaches zero.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
XFG	Execute, Flip and Go *Unconditional*	0x17	RAM Address for Message Control/ Status Block	Unconditionally execute the message at the parameter-specified Message Control/Status Block Address. **At message completion**, if the Condition Code evaluates True, then toggle bit 4 of the Message Control/Status Block Address, and store the new Message Control/Status Block Address as the updated value of the parameter following the XFG instruction op code. As a result, the next time this address in the BC Instruction List is executed, the processed Message Control/ Status Block resides at the updated address (old address XOR 0x0010) instead of the old address. Otherwise (Condition Code Evaluates False) the value of the Message Control/Status Block Address parameter is not changed. At the start of XQG message execution, if the fourth (Time to Next Message) word in the Message Control/Status Block is nonzero, it is copied to the “Bus Controller (BC) Time To Next Message Register (0x0036)”, and this message timer begins countdown. Completion of an XQG message may occur while message timer countdown continues. Unlike XQF, the XFG op code does not wait for the decrementing message timer to hit 0 before fetching the next instruction op code. As long as op codes following XFG do not execute a 1553 message, each op code is performed after fetch. Upon reaching a following XEQ, XQG, XQF or XFG execute-message instruction, transaction of its 1553 message does not begin until Time to Next Message count reaches 0. Thus, programmed 1553 message timing is maintained, while allowing execution of non-message instruction op codes.
WMP	Write Immediate Value to WMI Memory Pointer *Conditional*	0x1B	Immediate Value	If the Condition Code evaluates True, write the parameter-specified immediate value to the dedicated WMI memory pointer (a RAM location not accessible by the host). Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. Immediate value must exceed 0x4F or WMP instruction has no effect. After reset, the default WMI memory pointer value is 0x0050.
WMI	Write Immediate Value to Memory *Conditional*	0x1C	Immediate Value	If the Condition Code evaluates True, write the parameter-specified immediate value to 0x0050 or the memory address specified by the last WMP instruction performed. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List.

Name	*Instruction*	*Op Code*	*Parameter*	*Function*
DSZ	Decrement RAM Specified by Memory Address, Skip the Next Instruction if Zero *Conditional*	0x1D	Memory Address	If the Condition Code evaluates True, the memory address specified by the parameter word is decremented. If the new value is non-zero, the next instruction is executed. If the decremented value is zero, the next instruction is skipped. Otherwise (Condition Code Evaluates False), continue execution at the next op code in the BC Instruction List. The primary purpose of DSZ is N-iteration repeating execution loops. N is initialized with a WMI op code, and the instruction following DSZ is a JMP to top-of-loop.
FLG	General Purpose Flag Bits *Unconditional*	0x0C	Word value sets, clears or toggles General Purpose Flag Bits	The parameter word value is used to set, clear, or toggle the lower byte in the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. The upper and lower bytes in the parameter word provide 2-bit arguments that modify each of the eight GP flag bits, as illustrated below.

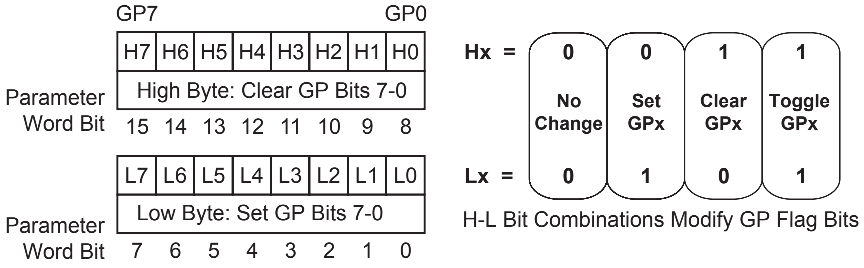


Figure 9. Bus Controller Flag Operation

12.3. Bus Controller General Purpose Queue

The BC architecture includes a General Purpose Queue, a 64-word circular buffer which the BC can use to convey information to the external BC host. Various BC instruction op codes push data values onto the queue, such as the Block Status Word for the last message, Time Tag Counter values, immediate data values, or values stored in specific RAM addresses.

The “Bus Controller (BC) General Purpose Queue Pointer Register (0x0038)” is initialized with the default starting address 0x00C0 after reset. The queue is relocatable, so the host may overwrite the default base address. Updated by the BC logic each time a data word is pushed onto the queue, the pointer in register 0x0038 always points to the next storage address in the queue to be written. The address pointer rolls over every 64th word written. If the BCGPQ bit 13 is logic 1 In the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, a BC interrupt is generated when the General Purpose Queue Pointer rolls over from its ending address to its base address.

12.4. Bus Controller Message Control / Status Blocks

In the BC Instruction List, each occurrence of the “execute message” instructions, XEQ, XQG, XQF and XFG, references a MIL-STD-1553B message. The op code word is followed by the parameter word, a memory pointer indicating the RAM start address for a corresponding Message Control/Status Block. The pointer address indicates the first word in the Message Control/Status Block, the BC Control Word. Figure 10 illustrates this relationship.

The HI-6300 is fully compatible with all MIL-STD-1553B message formats. For most MIL-STD-1553 messages, the corresponding Message Control/Status Block contains 8 words:

- **BC Control Word.** This word contains flags that select message format, choose the active bus, enable message retry and end-of-message interrupt, indicate expected RT status word flags, etc.
- **MIL-STD-1553 Command Word.** When message is RT-to-RT, this is the Receive Command Word.
- **Data Block Pointer.** For subaddress commands and mode code commands with data, this word identifies the start address of the Message Data Block in RAM. For mode commands without data, this word is not used.
- **Time-to-Next Message.** The time count loaded here begins decrementing at start of message. When value exceeds message execution time, it paces delivery of the next message.
- **Time Tag Word.** The current value of the internal time tag count is written to the Time Tag Word at Start-of-Message and again at the End-of-Message. When the BC uses a 16-bit time base, this location contains the complete time count. When the BC uses 32-bit time base, this word contains time bits 15-0 and block word 7 contains time bits 31-16 (instead of Loopback Word).
- **Block Status Word.** This word contains various message result flags.
- **Loopback Word,** containing the last word transmitted by the BC (16-bit time base only) or
  Time Tag Bits 31-16 (32-bit time base only)
- **RT Status Word** received.
  This is the **Transmit RT Status Word** when message is RT-to-RT.

When the message is RT-to-RT, the Message Control/Status Block contains 8 additional words:

- **Transmit Command Word.**
- **Receive RT Status Word.**
- Six unused word locations, to maintain 8 or 16 words per Message Control/Status Block.

Figure 10 shows the range of Message Control/Status Block variations. Selected words in the Message Control/Status Block are described next.

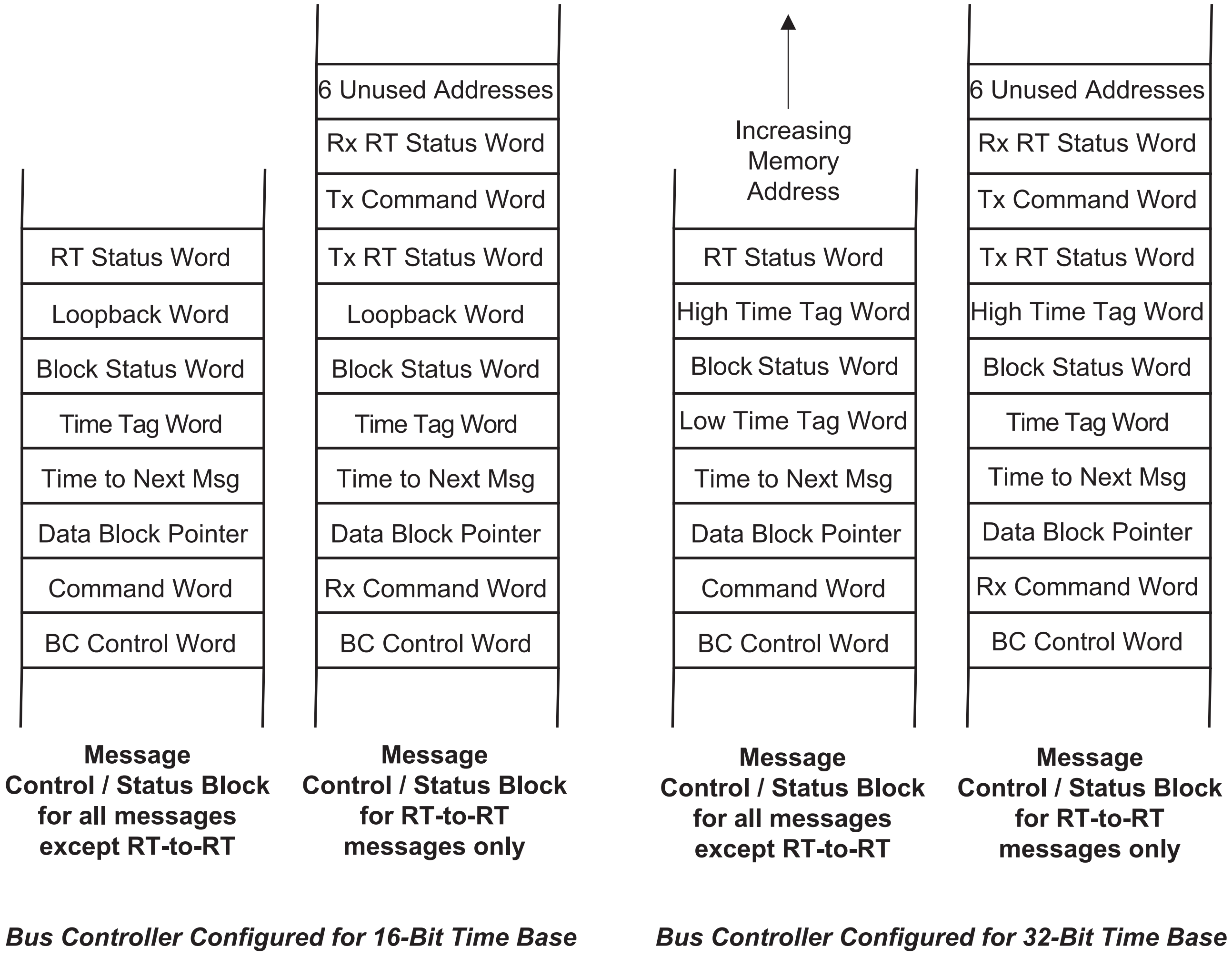

6 Unused Addresses
Rx RT Status Word
Tx Command Word
Tx RT Status Word
RT Status Word
Loopback Word
High Time Tag Word
Block Status Word
Low Time Tag Word
Time Tag Word
Time to Next Msg
Data Block Pointer
Command Word
Rx Command Word
BC Control Word
Increasing Memory Address
Message Control / Status Block for all messages except RT-to-RT
Message Control / Status Block for RT-to-RT messages only
Bus Controller Configured for 16-Bit Time Base
Bus Controller Configured for 32-Bit Time Base

Figure 10. Structure of Bus Controller Message Control / Status Blocks in RAM

12.4.1. BC Control Word

TXTTMC17	MEMASK	SRQMASK	BSYMASK	SSYSMASK	TFMASK	RSVMASK	RTRYENA	USEBUSA	SELFTST	MASKBCR	EOMINT	Reserved	MCODFMT	BCSTFMT	RTRTFMT	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The BC Control Word is the first word in each Message Control / Status Block. The BC Control Word is not transmitted on the MIL-STD-1553 bus. This word is initialized and maintained by the host to specify message attributes: message format, which bus to use, bit masks for the received RT Status Word, enabling interrupt at end-of-message, and enabling self test:

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	TXTTMC17	R/W	0	Transmit Time Tag for Synchronize Mode Code Command MC17. This option bit only applies when “BC (Bus Controller) Configuration Register (0x0032)” TTSYNEN bit 3 is logic 1. This bit affects only the “synchronize with data” mode code command, (mode code 0x11 or decimal 17). • If this Control Word bit is logic 0 (**or** if “BC (Bus Controller) Configuration Register (0x0032)” TTSYNEN bit 3 is logic 0) the BC transmits the value contained in the Message Data Block as the data word for a “synchronize” mode code command MC17. The transmitted word is fetched from the RAM address referenced by the Data Block Pointer. • If this Control Word bit **and** “BC (Bus Controller) Configuration Register (0x0032)” TTSYNEN bit 3 are both logic 1, the “synchronize” mode data word value originates from the 16-bit BC time base counter (low order 16 bits when using the 32-bit BC time base option). • If “BC (Bus Controller) Configuration Register (0x0032)” ETTSYN bit 2 is also logic 1, the transmitted time tag data word is always an even value; the least significant bit is always 0. • When ETTSYN bit 2 is logic 0, the data word least significant bit may be 0 or 1. The transmitted data value is saved in the message data block, at the address indicated by the Data Block Pointer word.
14	MEMASK	R/W	0	Message Error Bit Mask. If this BC Control Word bit is logic 0 **and** the Message Error bit 10 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the Message Error bit 10 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. **Note:** A “Status Set” condition results if one or more of these events occurs: • one or more of the mask bits 14-9 in the BC Control Word is logic 0 **and** the corresponding bit is logic 1 in the received RT Status Word, • the BCRME bit 0 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)” **and** the MASKBCR bit in the BC Control Word is logic 0 **and** the Broadcast Command Received bit 4 is logic 1 in the RT Status Word • the BCRME bit 0 is logic 0 in the “BC (Bus Controller) Configuration Register (0x0032)” **and** the MASKBCR bit in the BC Control Word differs from the Broadcast Command Received bit 4 in the RT Status Word • The received RT Status Word contains an RT Address field different from the RT Address field in the transmitted Command Word

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13	SRQMASK	R/W	0	Service Request Bit Mask. If this BC Control Word bit is logic 0 **and** the Service Request bit 8 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the Service Request bit 8 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. See “Note” at end of MEMASK bit description above.
12	BSYMASK	R/W	0	Busy Bit Mask. If this BC Control Word bit is logic 0 **and** the Busy bit 3 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the Busy bit 3 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. See “Note” at end of MEMASK bit description above.
11	SSYSMASK	R/W	0	Subsystem Flag Bit Mask. If this BC Control Word bit is logic 0 **and** the Subsystem Flag bit 2 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the Subsystem Flag bit 2 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. See “Note” at end of MEMASK bit description above.
10	TFMASK	R/W	0	Terminal Flag Bit Mask. If this BC Control Word bit is logic 0 **and** the Terminal Flag bit 0 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the Terminal Flag bit 0 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. See “Note” at end of MEMASK bit description above.
9	RSVMASK	R/W	0	Reserved Bits Mask. If this BC Control Word bit is logic 0 **and** one or more of the three Reserved bits 7-5 is logic 1 in the received RT Status Word, a “Status Set” condition will result. If this BC Control Word bit is logic 1, the three Reserved bits 7-5 in the received RT Status Word have no effect on the outcome for a “Status Set” condition. See “Note” at end of MEMASK bit description above.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
8	RTRYENA	R/W	0	Retry Enabled. If this Control Word bit is logic 1 **and** BCRE (BC Retry Enable) bit 12 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”, the BC will retry a message if RT response timeout or Format Error occurs. If this Control Word bit is logic 1 **and** BCRE (BC Retry Enable) bit 12 is logic 1 **and** BCRSB (BC Retry If Status Word Bits Set) bit 8 is also logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”, the BC will retry a message when a “Status Set” condition occurs. See “Note” at end of MEMASK bit 14 description above.
7	USEBUSA	R/W	0	Use Bus A / $\overline{\text{Use Bus B}}$. If this Control Word bit is logic 1, the BC transmits the command on Bus A. If this Control Word bit is logic 0, the BC transmits the command on Bus B.
6	SELFTST	R/W	0	Self-Test Message Off-Line. If this Control Word bit is logic 1, transmission of this message onto the 1553 bus is inhibited. Instead the digitally-encoded Command Word is looped back into the receive decoder for the selected bus. This tests both the encoding and decoding signal paths. Upon message completion, Loop Test Fail bit 8 in the Block Status Word indicates the self-test result. If BC is configured for 16-bit time base, the received Loopback Word is stored in the Message Control/Status Block. If the BC is configured for 32-bit time base, time tag bits 31-16 are stored in the Loopback Word location. See Section “22.1.7. Programmed BC-Mode Digital Loopback Testing (Off-Line)” on page 201.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
5	MASKBCR	R/W	0	Mask Broadcast Command Received Bit. **If the BCRME bit 0 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”**, then this Control Word bit is a “mask bit” like bits 14-9, acting upon the BCR Broadcast Command Received bit 4 in the received RT Status Word: • If this MASKBCR Control Word bit is logic 0 **and** the Broadcast Command Received bit 4 is logic 1 in the received RT Status Word, a “Status Set” condition will result. • If this MASKBCR Control Word bit is logic 1, the Broadcast Command Received bit 4 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. **If the BCRME bit 0 is logic 0 in the “BC (Bus Controller) Configuration Register (0x0032)”**, then this MASKBCR Control Word bit reflects the expected state of the Broadcast Command Received bit 4 in the received RT Status Word: • If this MASKBCR Control Word bit **does not match** the logic state of the Broadcast Command Received bit 4 in the received RT Status Word, a “Status Set” condition will result. • If this MASKBCR Control Word bit **matches** the logic state of the Broadcast Command Received bit 4 in the received RT Status Word, then the Broadcast Command Received bit 4 in the received RT Status Word has no effect on the outcome for a “Status Set” condition. See Table 13.
4	EOMINT	R/W	0	End of Message Interrupt. If the BCEOM bit 3 is logic 1 in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, an EOM interrupt will occur upon message completion, if this Control Word bit is logic 1.
3	Reserved	R/W	0	This bit is not used.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
2	MCODFMT	R/W	0	Mode Code Message Format.
1	BCSTFMT			Broadcast Message Format.
0	RTRTFMT			RT to RT Message Format. The combination of these three Message Format bits selects the MIL-STD-1553B message type:

Mode Code Bit 2	***Broadcast Bit 1***	***RT-RT Bit 0***	***Message Type***
0	0	0	BC-to-RT if the $T/\overline{R}$ bit * equals logic 0 RT-to-BC if the $T/\overline{R}$ bit * equals logic 1
0	0	1	RT-to-RT
0	1	0	Broadcast BC-to-RT
0	1	1	Broadcast RT-to-RT
1	0	0	Mode Code Command
1	0	1	Do Not Use
1	1	0	Broadcast Mode Code Command
1	1	1	Do Not Use

* Transmit / $\overline{\text{Receive}}$ bit in Command Word

Note: Bit 0 must be logic 1 for RT-RT messages. This is the single control point enabling the 16-word Control Block that configures RT-RT messages.

12.4.2. Time to Next Message Word

The Bus Controller provides a programmable delay for Time to Next Message. This word in the Message Control / Status Block specifies the delay from the start of this message, to the start of the next message. The delay is programmable with 1 µs per LSB resolution, and has a maximum value of 65.535 milliseconds.

When the specified Time to Next Message value is less than the actual time required to transact the current message, the next message starts immediately upon completion current message, after the minimum inter-message gap time of 4µs. This gap corresponds to a bus “dead time” of 2µs.

12.4.3. Data Block Pointer

The Data Block Pointer in the Message Control / Status Block provides the starting address in RAM for storage of message data words or mode code data. For BC-to-RT (receive) commands, this pointer contains the RAM location for the first data word transmitted by the BC.

For RT-to-BC (transmit) commands or RT-to-RT commands, this pointer contains the RAM location for storing the first data word transmitted by the RT (and received by the BC).

12.4.4. BC Block Status Word

EOM	SOM	BID	MBCE	STATSET	FE	TM	LBE	MSTATSET	RTRY1	RTRY0	GDB	WAG	LE	SE	IWE	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The Block Status Word in the Message Control / Status Block provides information regarding message status (in-process or completed), the bus it was transmitted on, whether errors occurred during the message, and the type of occurring errors. This word is written into RAM by the IP core after message completion. Because it resides in RAM, the host has read-write access, although this word is usually treated as read-only by the host.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	EOM	R/W	0	End of Message. This bit is set upon completion of a BC message, whether or not errors occurred. When EOM is set, the current value of the Time Tag Word(s) is (are) written to the corresponding Time Tag Word(s) in the BC Message Control/Status Block.
14	SOM	R/W	0	Start of Message. This bit is set at the start of a BC message and cleared at the end of the message. When SOM is set (and reset), the current value of the Time Tag Word(s) is (are) written to the corresponding Time Tag Word(s) in the BC Message Control/Status Block.
13	BID	R/W	0	Bus ID (Bus B / $\overline{\text{Bus A}}$). This bit is logic 1 if the BC message was transacted on Bus B. This bIt is logic 0 if the BC message was transacted on Bus A.
12	MBCE	R/W	0	Message Block Coding Error. A programming test aid, this bit only applies when CHKFMT bit 13 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”. Each message BC Control Word has flags that indicate format: mode code command (bit 2), RT-RT message (bit 1) and broadcast message (bit 0). When CHKFMT is 1, these flags are compared to the message Command Word(s) that follow the Control Word in the Message Block. When message format mismatch occurs (a BC programming error), this MBCE bit is set in the message Block Status Word. See CHKFMT description in “BC (Bus Controller) Configuration Register (0x0032)” on page 78.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
11	STATSET	R/W	0	Status Set. This bit is not affected by the values of mask bits 14-9 in the “BC Control Word” for the message. This bit is logic 1 when the received RT Status Word contains an unexpected bit value in the bit 10-0 range. The expected value is usually logic 0 for bits 10-0 in the received RT Status Word. Exception: the expected value for Broadcast Command Received bit 4 in the received RT Status Word is logic 1 when BCRME bit 0 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”, **and** MASKBCR bit in the “BC Control Word” bit is logic 0
10	FE	R/W	0	Format Error. This bit is logic 1 when a received RT response violates MIL-STD-1553 message validation criteria. This includes sync, word count, encoding, bit count or parity errors. Word bits 2-0 provide additional information. This flag is also set when the received RT Status Word response from the last message contained an incorrect RT address field.
9	TM	R/W	0	No Response Timeout Error. This bit is logic 1 when an RT fails to respond, or responds later than the BC No Response Timeout interval specified by bits 15-14 in the “BC (Bus Controller) Configuration Register (0x0032)”.
8	LBE	R/W	0	Loopback Error. The BC evaluates its own 1553 message transmissions. The received version of each word transmitted by the BC is checked for 1553 validity (sync, encoding, bit count and/or parity error). In addition, for each message transacted, the received image for the last word transmitted by the BC is evaluated for data match. This bit is logic 1 when the received version for one or more words transmitted by the BC fails 1553 “word validity” criteria, and/or the received version for the last word transmitted by the BC does not match the Manchester II word transmitted by the BC.
7	MSTATSET	R/W	0	Masked Status Set. This bit is logic 1 when any of the following conditions occurs: • One or more of the mask bits 14-9 in the BC Control Word is logic 0 **and** the corresponding bit is logic 1 in the received RT Status Word. • Or the BCRME bit 0 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)” **and** the MASKBCR bit in the BC Control Word is logic 0 **and** the Broadcast Command Received bit 4 is logic 1 in the RT Status Word. • Or the BCRME bit 0 is logic 0 in the “BC (Bus Controller) Configuration Register (0x0032)” **and** the MASKBCR bit in the BC Control Word **differs from** the Broadcast Command Received bit 4 in the RT Status Word. • Or the received RT Status Word contains an RT Address field different from the RT Address field in the transmitted Command Word.

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th colspan="3"><i>Function</i></th></tr>
<tr><td rowspan="3">6
5</td><td rowspan="3">RTRY1
RTRY0</td><td rowspan="3">R/W</td><td rowspan="3">0</td><td colspan="3">Retry Count 1 and Retry Count 0
If BCRE (BC Retry Enable) bit 12 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)” and RTRYENA bit 8 is logic 1 in the “BC Control Word” for this message, the BC will retry the message if RT response timeout or Format Error occurs.
Also, if BCRE (BC Retry Enable) bit 12 is logic 1 and BCRSB (BC Retry if Status Word Bits Set) bit 8 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)” and RTRYENA bit 8 is logic 1 in the “BC Control Word” for this message, the BC will retry the message when a “Status Set” condition occurs. See “MSTATSET” bit 7 description above.
The combination of these two bits indicates the number of times this message was retried:</td></tr>
<tr><td><b><i>RTRY1 Bit 6</i></b></td><td><b><i>RTRY0 Bit 5</i></b></td><td><b><i>Number of Retries</i></b></td></tr>
<tr><td>0
0
1
1</td><td>0
1
0
1</td><td>0
1
2
Not Used</td></tr>
<tr><td>4</td><td>GDB</td><td>R/W</td><td>0</td><td colspan="3">Good Transmit Data Block Transfer.
This bit is set to logic 1 upon successful completion of an error-free RT-to-BC message, RT-to-RT message, or transmit mode code message with data. This bit always resets to logic 0 for any BC-to-RT message, mode code message without data, or any incomplete or invalid message.
This bit may be used for determining when the transmit portion of an RT-to-RT message is error-free. If this bit and Error Flag bit 12 are both set to logic 1 in the Block Status Word for an RT-to-RT message, the transmitting RT responded correctly but error occurred in the receiving RT portion of the message.</td></tr>
<tr><td>3</td><td>WAG</td><td>R/W</td><td>0</td><td colspan="3">Wrong RT Address and/or No Gap.
This bit is logic 1 when one or both of the following conditions occur
• the RT address field within a received RT Status Word does not match the RT address field in the Command Word transmitted by the BC
• the BCGCE BC Gap Check Enable bit 1 in the “BC (Bus Controller) Configuration Register (0x0032)” and the RT responds with response time less than 4 µs per MIL-STD-1553B, mid-parity bit to mid-sync, (2 µs bus “dead time”).</td></tr>
<tr><td>2</td><td>LE</td><td>R/W</td><td>0</td><td colspan="3">Word Count (Length) Error.
This bit is logic 1 when an RT-to-BC message, RT-to-RT message, or transmit mode code message with data is transacted with the wrong number of data words.
This bit always resets to logic 0 for BC-to-RT messages, receive mode code messages, or transmit mode code messages without data.</td></tr>
<tr><td>1</td><td>SE</td><td>R/W</td><td>0</td><td colspan="3">Sync Error.
This bit is logic 1 when an RT responds with Data Sync in its Status Word, or with Command/Status Sync in a Data Word.</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
0	IWE	R/W	0	Invalid Word Error. This bit is logic 1 when an RT response in one or more words having at least one of the following errors: sync encoding error, Manchester II encoding error, bit count error, parity error.

13. BUS CONTROLLER REGISTER DESCRIPTION

13.1. BC (Bus Controller) Configuration Register (0x0032)

BCTO1	BCTO0	CHKFMT	BCRE	BC2RE	BCR1A	BCR2A	BCRSB	MENDV	BSYNDV	BCGTE	BCWDTE	TTSYNEN	ETTSYN	BCGCE	BCRME	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th><i>Function</i></th></tr>
<tr><td>15 − 14</td><td>BCTO1:0</td><td>R/W</td><td>0</td><td>
BC Time Out Select.
This 2-bit field selects the BC “no response” time-out delay from four available selections. Excluding RT-RT commands, response delay is measured from command word mid-parity bit to status word mid-sync:
<table>
<tr><th><i>Bit 15:14</i></th><th><i>Bus Dead Time</i></th><th><i>Time Out (excludes RT-RT)</i></th></tr>
<tr><td>00</td><td>15µs</td><td>17µs</td></tr>
<tr><td>01</td><td>20µs</td><td>22µs</td></tr>
<tr><td>10</td><td>58µs</td><td>60µs</td></tr>
<tr><td>11</td><td>138µs</td><td>140µs</td></tr>
</table>
For RT-RT commands, time out delay is measured per Figure 8 in the RT Validation Test Plan, SAE AS4111. That is, from mid-parity of the receive command to mid-sync of the first received data word. This adds 42µs for the embedded parity half-bit, transmit command word, transmit-RT status word and data half-sync within this interval:
<table>
<tr><th><i>Bit 15:14</i></th><th><i>TxRT Bus Dead Time</i></th><th><i>RT-RT Time Out*</i></th></tr>
<tr><td>00</td><td>18µs</td><td>60µs</td></tr>
<tr><td>01</td><td>23µs</td><td>65µs</td></tr>
<tr><td>10</td><td>58µs</td><td>100µs</td></tr>
<tr><td>11</td><td>138µs</td><td>180µs</td></tr>
</table>
<b>*Note:</b> per RT Validation Test Plan, Fig. 8.
All time out select values have –100ns / +500ns tolerance.
</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13	CHKFMT	R/W	0	Check Message Format. When this bit equals logic 1, “BC Control Word” message format bits 2-0 (mode command, RT-RT message and broadcast flags) are compared to values for the message Command Word(s) that follow the Control Word in the Message Block. This is provided as a BC program development aid. When the CHKFMT bit is logic 1 and message format mismatch occurs, the MBCE Message Block Coding Error bit 12 is set in the message Block Status Word (see Section “12.4.4. BC Block Status Word” on page 74). If enabled, BC Trap interrupt is generated (see Section “13.15.3. Bus Controller (BC) Interrupt Output Enable Register (0x0014)” on page 95). When the CHKFMT bit is logic 0, no format checking occurs between “BC Control Word” bits 2-0 and the MIL-STD-1553 message Command Word(s). Even with message format checking disabled, “BC Control Word” bit 0 **must be** logic 1 for RT-RT messages. *This is the single control point enabling the 16-word BC Control Block that configures an RT-to-RT message*.
12	BCRE	R/W	0	BC Retry Enable. If bit 12 equals logic 0, command retries are disabled for all messages. If bit 12 equals logic 1, command retries can be enabled on an individual message basis by setting bit 8 in the “BC Control Word”.
11	BC2RE	R/W	0	BC Second Retry Enable. If retries are enabled (register bit 12 equals 1) this bit selects the number of retries performed, If bit 11 equals logic 0, a single retry is performed. If bit 11 equals logic 1, up to two retries are performed.
10	BCR1A	R/W	0	BC First Retry Use Alternate Bus. If retries are enabled (register bit 12 equals 1) this bit selects the bus used for the first retry. If bit 10 equals 0, the first retry is performed on the same bus from which the message was originally transmitted. If bit 10 equals 1, first retry is performed on the alternate bus from which the message was originally transmitted.
9	BCR2A	R/W	0	BC Second Retry Use Alternate Bus. If first and second retries are enabled (register bits 12:11 equal 1-1) this bit selects the bus used for the second retry. If bit 9 equals 0, the second retry is performed on the same bus where the message was originally transmitted. If bit 9 equals 1, second retry is performed on the alternate bus from where the message was originally transmitted.

<table>
<tr><th><em>Bit No.</em></th><th><em>Mnemonic</em></th><th><em>R/W</em></th><th><em>Reset</em></th><th colspan="3"><em>Function</em></th></tr>
<tr><td>8</td><td>BCRSB</td><td>R/W</td><td>0</td><td colspan="3">BC Retry If Unmasked Status Word Bit Set.
This bit affects operation of BC retries. If bit 8 equals logic 0, the BC will not retry messages because of status bit(s) set in the RT Status Word, or Status Word with non-matching RT address field.
If retries are enabled and bit 8 equals logic 1, the BC will retry messages for these RT status word results:
1. One or more “BC Control Word” mask bits 14-9 is logic 0 (bits are not masked) and the corresponding bit is logic 1 in the received RT Status Word.
2. The BCR Mask Enable bit 0 equals 0 in the “BC (Bus Controller) Configuration Register (0x0032)”. The Broadcast Command Received (BCR) bit in the received RT Status Word differs from the Mask BCR bit (bit 5) in the BC Control Word.
3. Received Status Word RT address does not match Command Word RT address.</td></tr>
<tr><td rowspan="5">7</td><td rowspan="5">MENDV</td><td rowspan="5">R/W</td><td rowspan="5">0</td><td colspan="3">Message Error Status, No Data is Valid
Bit 7 affects BC validation of RT responses to transmit commands when the Message Error (ME) bit is asserted in the received RT Status Word.
<strong>When the MENDV bit equals logic 0:</strong>
• When an RT responds Message Error status to a transmit command, the response is valid only if the Status Word is followed by the commanded number of data words. Message result:</td></tr>
<tr><td><strong>Condition</strong></td><td><strong>Block Status Word</strong></td><td><strong>Condition Code Register</strong></td></tr>
<tr><td>Commanded number of data words</td><td>Good Data Block (GDB) bit = 1</td><td>Good Data Block Transfer (GDBT) bit = 1</td></tr>
<tr><td rowspan="2">If MEMASK = 1 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
</table>

<table>
<tr><th><em>Bit No.</em></th><th><em>Mnemonic</em></th><th><em>R/W</em></th><th><em>Reset</em></th><th colspan="3"><em>Function</em></th></tr>
<tr><td rowspan="11">7</td><td rowspan="11">MENDV (continued)</td><td rowspan="11">R/W</td><td rowspan="11">0</td><td colspan="3">• If the RT responds Message Error status to a transmit command with the wrong number of data words or no data words, here is the message result:</td></tr>
<tr><td><strong>Condition</strong></td><td><strong>Block Status Word</strong></td><td><strong>Condition Code Register</strong></td></tr>
<tr><td>Wrong number of (or no) data words</td><td>Format Error (FE) bit = 1</td><td>Format Error (FMTERR) bit = 1</td></tr>
<tr><td rowspan="3">If MEMASK = 1 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Word Count Error (LE) bit = 1.</td><td>Bad Message (BADMSG) bit = 1</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
<tr><td colspan="3"><strong>When the MENDV bit equals logic 1:</strong>
• When an RT responds Message Error status to a transmit command, the response is valid with the commanded number of data words, or ME status with no data words. Here is the message result:</td></tr>
<tr><td><strong>Condition</strong></td><td><strong>Block Status Word</strong></td><td><strong>Condition Code Register</strong></td></tr>
<tr><td>Commanded # of data words only, “Good Data” is not shown if no data</td><td>Good Data Block (GDB) bit = 1</td><td>Good Data Block Transfer (GDBT) bit = 1</td></tr>
<tr><td rowspan="2">If MEMASK = 0 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
</table>

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>7</td><td>MENDV
(continued)</td><td>R/W</td><td>0</td><td>
• If the RT responds Message Error status to a transmit command with the wrong number of data words (not 0), here is the message result:
<table>
<tr><th>Condition</th><th>Block Status Word</th><th>Condition Code Register</th></tr>
<tr><td>Wrong number of data words</td><td>Format Error (FE) bit = 1</td><td>Format Error (FMTERR) bit = 1</td></tr>
<tr><td rowspan="3">If MEMASK = 0 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Word Count Error (LE) bit = 1.</td><td>Bad Message (BADMSG) bit = 1</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
</table>
The only 3 cases when an RT transmits Message Error status onto the bus:
1. An RT using “illegal command detection” receives an illegal command that otherwise meets all other validation requirements. The RT responds with Status Word only, with Message Error bit set. No data words are sent.
2. An RT receives a “transmit status” mode command (MC2). The previous valid command for the RT had Message Error status. The RT responds with Status Word only, with Message Error bit set. No data words are sent.
3. An RT receives a “transmit last command” mode command (MC18 decimal). The previous valid command for the RT set Message Error status. The RT responds with Status Word (with Message Error bit set) and one data word, the previous Command Word.
<b>In summary, Message Error status never occurs with more than one data word, and only occurs with one data word for the “transmit last command” mode code.</b>
Besides illegal command detection, there is just one situation where Message Error status occurs, but Status transmission is suppressed: The RT detects a valid receive command having correct RT address, but an invalid word is detected in the accompanying data words, or a gap occurs between words. <b>In this situation, Message Error status is set but the RT suppresses its Status Word transmission.</b> This suppressed ME status is only seen by the BC if retrieved by a following “transmit status’’ or “transmit last command” mode command.
</td></tr>
</table>

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th colspan="3"><i>Function</i></th></tr>
<tr><td rowspan="16">6</td><td rowspan="16">BSYNDV</td><td rowspan="16">R/W</td><td rowspan="16">0</td><td colspan="3">Busy Status, No Data, is Valid
Bit 6 affects BC validation of RT responses to transmit commands when the Busy bit is asserted in the received RT Status Word.
<b>When the BSNDV bit equals logic 0:</b>
When an RT responds Busy status to a transmit command, the response is valid only if the Status Word is followed by the commanded number of data words. Here is the message result:</td></tr>
<tr><td><b>Condition</b></td><td><b>Block Status Word</b></td><td><b>Condition Code Register</b></td></tr>
<tr><td>Commanded number of data words</td><td>Good Data Block (GDB) bit = 1</td><td>Good Data Block Transfer (GDBT) bit = 1</td></tr>
<tr><td rowspan="2">If BSYMASK = 1 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
<tr><td colspan="3">• If the RT responds Busy status to a transmit command with the wrong number of data words or no data words, here is the message result:</td></tr>
<tr><td><b>Condition</b></td><td><b>Block Status Word</b></td><td><b>Condition Code Register</b></td></tr>
<tr><td>Wrong number of (or no) data words</td><td>Format Error (FE) bit = 1</td><td>Format Error (FMTERR) bit = 1</td></tr>
<tr><td rowspan="3">If BSYMASK = 1 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Word Count Error (LE) bit = 1.</td><td>Bad Message (BADMSG) bit = 1</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
<tr><td colspan="3"><b>When the BSNDV bit equals logic 1:</b>
• When an RT responds to a transmit command with Busy status, the response is valid when accompanied by the commanded number of data words, or accompanied by no data words. Here is the message result:</td></tr>
<tr><td><b>Condition</b></td><td><b>Block Status Word</b></td><td><b>Condition Code Register</b></td></tr>
<tr><td>Commanded # of data words only, “Good Data” is not shown if no data</td><td>Good Data Block (GDB) bit = 1</td><td>Good Data Block Transfer (GDBT) bit = 1</td></tr>
<tr><td rowspan="2">If BSYMASK = 0 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
</table>

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>6</td><td>BSYNDV (continued)</td><td>R/W</td><td>0</td><td>
<ul><li>If the RT responds to a transmit command with Busy status and the wrong number of data words (but not 0), here is the message result:</li></ul>
<table>
<tr><th>Condition</th><th>Block Status Word</th><th>Condition Code Register</th></tr>
<tr><td>Wrong number of data words</td><td>Format Error (FE) bit = 1</td><td>Format Error (FMTERR) bit = 1</td></tr>
<tr><td rowspan="3">If BSYMASK = 0 in BC Control Word</td><td>Masked Status Set (MSTATSET) bit = 1.</td><td>Masked Status Set (MSTATSET) bit = 1.</td></tr>
<tr><td>Word Count Error (LE) bit = 1.</td><td>Bad Message (BADMSG) bit = 1</td></tr>
<tr><td>Status Set (STATSET) bit = 1.</td><td>----------</td></tr>
</table>
<p>A busy RT is one that is functional, but cannot send or receive data when commanded by the Bus Controller. An RT that is busy sets the Busy bit in its Status Word responses. In response to transmit commands, the busy terminal has no words to transmit, so only the Status Word is transmitted. In the case of the “transmit vector word” and transmit BIT word” mode code commands, even if the data is available to the terminal, it is prohibited to send the data word if the Busy bit is set in the Status Word.</p>
<p>There is just one defined situation in which an RT transmits Busy status with one (and only one) data word: An RT receives a “transmit last command” mode command (MC18 decimal). When the previous valid command for the RT had Busy status, the RT responds with last message status condition (with Busy bit set) and one data word, the previous Command Word.</p>
</td></tr>
<tr><td>5</td><td>BCGTE</td><td>R/W</td><td>0</td><td>
<p>BC Message Gap Timer Enable.</p>
<p>If bit 5 is logic 0, the BC does not add delay between 1553 messages. Message timing is paced by the time required for the BC to complete message post processing. In this case, the minimum inter-message gap will be used, with a bus “dead time” of approximately 6 to 9µs.</p>
<p>If bit 5 is logic 1, the BC Message Timer is enabled. The “Time to Next Message” value from the Message Control Block is decremented at 1µs rate. When the count decrements from 1 to 0, the next message starts. If the specified message gap time is less than the time needed for the current message, the next message will start immediately after completion of the current message. In this case, the minimum inter-message gap will be used, with a bus “dead time” of approximately 6 to 9µs. This allows the BC to implement minor frame cycle times without host processor intervention.</p>
</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
4	BCWDTE	R/W	0	BC Watchdog Timer (WDT) Enabled. When this bit is logic 1, if the BC Watchdog Timer interrupt bit is set in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, the BC sets the BCWDT bit in the “Bus Controller (BC) Pending Interrupt Register (0x0007)” when the BC Frame Timer decrements from 1 to 0. When using this feature, it is necessary to periodically reload the Frame Time register by means of the LFT Load Frame Time op code in the BC message sequence control block. The loaded value must allow for the worst-case frame time (including message retries). The WDT provides a failsafe recovery from faults such as message sequence stuck in repetitive loop without LFT reload, or execution jumping to the wrong sequence.
3	TTSYNEN	R/W	0	BC Time Tag Synchronization Enable This option bit only affects the “synchronize with data” mode code command, MC17. When this bit is logic 1, the source of the mode data issued with the “synchronize” mode command is determined by message “BC Control Word” TXTTMC17 bit 15 (see Section 12.4.1) • If “BC Control Word” TXTTMC17 bit 15 = logic 0, the “synchronize” mode data word originates from the message data block, at the RAM address indicated by the Data Block Pointer word. • If “BC Control Word” TXTTMC17 bit 15 = logic 1, the “synchronize” mode data word value originates from the 16-bit BC time base counter (or the low order 16 bits, when using the 32-bit BC time base option). The transmitted data value is saved in the message data block, at the address indicated by the Data Block Pointer word. If the TTSYNEN bit is logic 0, regardless of the state of message Control Word bit 15, the “synchronize with data” mode command (MC17) is always issued with mode data originating from the message data block, at the RAM address indicated by the Data Block Pointer word.
2	ETTSYN	R/W	0	Even Time Tag Sync. This bit only applies when TTSYNEN bit 3 is logic 1 **and** “BC Control Word” TXTTMC17 bit 15 is also logic 1, selecting “synchronize” mode data origin as time base counter. In this case, if ETTSYN is logic 1, the transmitted time tag data word is always even; the low order bit is always 0. When the ETTSYN bit is logic 0, the data word may be even or odd; the LSB may be 0 or 1.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
1	BCGCE	R/W	0	BC Gap Check Enable. When this bit is logic 1, the BC verifies that any transmission by a Remote Terminal on the bus is preceded by an inter-word gap of at least 4µs. When this minimum gap time is violated, the BC declares the RT Status Word invalid, and the No Gap error bit is set in the Block Status Word. Gap detection measures the time span from mid-parity of last received word, to mid-sync of the following word (defined here as detection of a properly encoded sync plus two bits). Measured gap time is adjusted to evaluate just the interval of interest, corresponding to a bus "dead time" (end-of-parity to start-of-sync) of 2µs minimum. When this bit is logic 0 **(strongly recommended)** the BC does not check for minimum bus "dead time" prior to start of transmission by a IP core on the MIL-STD-1553 bus.
0	BCRME	R/W	0	BCR Mask Enable. This bit selects the function of "BC Control Word", Mask Broadcast, when evaluating the BCR (Broadcast Command Received) bit in RT status words. **If BCR Mask Enable bit 0 is logic 1**, then RT status word BCR bit masking is enabled. The state of the Mask BCR bit in each "BC Control Word" selectively allows or disallows BCR status testing by the BC: • When "Mask BCR" is logic 1 in a message's "BC Control Word" (disabling BCR status bit test), the value of the RT status word BCR bit is "don't care" in terms of affecting the occurrence of a "Status Set" condition. • When "Mask BCR" is logic 0 in the message's "BC Control Word", "Status Set" occurs when the BCR bit in the received RT Status Word is logic 1. • While broadcast commands never result in transmitted RT status, the "Mask BCR" bit should be set in BC Control Words for "transmit status" or "transmit last command" mode commands immediately following broadcast messages. Setting the "Mask BCR" bit of the message's "BC Control Word" to logic 1 indicates the expected value of the BCR bit in the received RT Status Word. **If BCR Mask Enable bit 0 is logic 0**, the "Mask BCR" bit in a message's "BC Control Word" indicates the expected state of the BCR bit in the received RT status word. In this situation, whenever the BCR bit in the received RT status word differs from the state of the "Mask BCR" bit in the "BC Control Word", a "Status Set" condition occurs and the BC generates a Status Set interrupt, if enabled. Table 13 summarizes the effects of BC configuration, message Control Word and RT Status Word on "Status Set" bit in the Block Status Word.

Table 13. Effect of “Broadcast Command Received” RT Status Bit on “Status Set” Condition

BC Configuration Register 0x0032 “BCR Mask Enable” bit 0	Message Control Word “Mask BCR” bit 5	Received BCR bit 4 in the Remote Terminal Status Word	Resultant Block Status Word “Status Set” bit 11
0 0 0 0	0 0 1 1	0 1 0 1	0 1 1 0
1 1 1	0 0 1	0 1 X	0 1 0

13.2. Start Address Register for Bus Controller (BC) Instruction List (0x0033)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the host. This register is cleared after $\overline{\text{RESET}}$ signal master reset. This register is initialized with the base address of the re-locatable BC Instruction List in IP core RAM.

13.3. Bus Controller (BC) Instruction List Pointer (0x0034)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Only and is fully maintained by the IP core. When the BC is running, the user programming changes this pointer value indirectly, by executing a JMP op code. When the bus controller is enabled, setting BCSTRT bit 13 in the “Master Configuration Register 1 (0x0000)” begins bus controller operation. The IP core copies the Instruction List base address from register 0x0033 into this register. This pointer references pairs of words in the BC instruction list. Each word pair is comprised of an op code word followed by a parameter word. Pointer update occurs just before execution of the next BC instruction list op code, after execution of the prior op code, and evaluation of its result-dependent outcome.

13.4. Bus Controller (BC) Frame Time Remaining Register (0x0035)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write. A value is written to this register upon execution of the BC instruction list op code, “Load Frame Timer” (LFT). Time remaining value begins decrementing upon execution of the Start Frame Timer (SFT) instruction op code. The parameter word accompanying the op code word is the desired time value, expressed with a resolution of 100 µs per LSB, with a maximum value of 6.5535 sec.

13.5. Bus Controller (BC) Time To Next Message Register (0x0036)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Only. This programmable time-to-next message timer is loaded on a message-by-message basis, with values from word 4 in each Message Control / Status Block. The BC time-to-next message is defined as the time from the start of the current message to the start of the next message, i.e., mid-sync zero crossing to the next mid-sync zero crossing. This timer provides a 1 µs per LSB resolution, with a maximum value of 65.535 ms.

13.6. Bus Controller (BC) Condition Code Register (Read 0x0037)

BCRUN	RETRY1	RETRY0	BADMSG	MSTATSET	GDBT	FMTERR	NORESP	GPF7	GPF6	GPF5	GPF4	GPF3	GPF2	EQ / GP1	LT / GP0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Sharing the same register address as the Write-Only “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”, this 16-bit register is Read-Only. Bit 15 indicates BC run/stop status. With this exception, the upper 8 bits indicate results from the last message processed by the Bus Controller. The lower 8 bits of this register are general purpose flag bits, which may be set, cleared, or toggled by the host using the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”, or by the IP core by means of the General Purpose Flag Bits (FLG) instruction op code. Further, bits 1-0 can be set or cleared by the IP core BC logic by execution of two BC instruction op codes: Compare to Frame Timer (CFT) and Compare to Message Timer (CMT).

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>15</td><td>BC RUN</td><td>R</td><td>0</td><td>BC Run / $\overline{\text{Stop}}$
This is a status bit, not a condition code. This bit indicates whether the BC is running or stopped. The bit is set to logic 1 when the BCSTRT bit 13 in the “Master Configuration Register 1 (0x0000)” is asserted and the BCENA bit 12 in the “Master Configuration Register 1 (0x0000)” is logic 1.
Once set, this bit resets to logic 0 if the BCENA register bit is reset to logic 0 by the host, or if the BC executes the HLT instruction, or if the BC executes an illegal op code. The “illegal op code” case will generate a BCTRAP interrupt, if enabled (see Section “13.15.3. Bus Controller (BC) Interrupt Output Enable Register (0x0014)” on page 95).</td></tr>
<tr><td>14
13</td><td>RETRY 1
RETRY 0</td><td>R</td><td>0</td><td>Message Retry Status Bits.
Bits 14-13 indicate the retry status of the most recent message the number of times message was retried:
<table>
<tr><th>Bit 14:13</th><th>Number of Message Re-tries</th></tr>
<tr><td>0-0
0-1
1-0
1-1</td><td>0
1
2
not used</td></tr>
</table></td></tr>
<tr><td>12</td><td>BADMSG</td><td>R</td><td>0</td><td>Bad Message.
This bit is logic 1 to indicate message format error, loopback test failure, or no response error for the last message.</td></tr>
<tr><td>11</td><td>MSTATSET</td><td>R</td><td>0</td><td>Masked Status Set.
This bit is set if one or more of the following conditions occurred during the last message:
• One or more of the Status Mask bits 14-9 are logic 0 in the “BC Control Word”, and the corresponding bits are logic 1 in the received RT Status Word. When BC Control Word “Reserved Bits Mask” bit 9 is logic 0, this Masked Status Set bit is set if any of the three Reserved bits are set in the received RT Status Word.
• The BCR Mask Enable bit 0 is logic 0 in the “BC (Bus Controller) Configuration Register (0x0032)” Opposite logic states occur for the Mask BCR bit in the message BC Control Word and the Broadcast Command Received (BCR) bit in the received RT Status Word.
• The BCR Mask Enable bit 0 is logic 1 in the “BC (Bus Controller) Configuration Register (0x0032)”. Opposite logic states prevail for the Mask BCR bit in the message BC Control Word and the Broadcast Command Received (BCR) bit in the received RT Status Word.
Table 13 on page 87 shows how the Broadcast Command Received (BCR) bit in the received RT Status Word affects Masked Status Set.</td></tr>
</table>

Bit No.	Mnemonic	R/W	Reset	Function
10	GDBT	R	0	Good Data Block Transfer. Indicating status for the last transmit-data message, this bit is set to logic 1 after completion of an error-free RT-to-BC transfer, RT-to-RT transfer, or transmit mode command with data. This bit is reset to logic 0 following any message in which error occurs. This bit is always logic 0 after completion of a BC-to-RT transfer, a receive mode command with data, or any mode command without data. This bit can be used to determine when the transmit portion of an RT-to-RT message was error-free: A Block Status Word for an RT-to-RT message having both the Error Flag and Good Data Block Transfer bits set indicates that the transmitting RT responded correctly, but an error was detected in the receiving RT portion of the message. This bit is not affected by the IP core loop back function.
9	FMTERR	R	0	Format Error. This bit is logic 1 when the received data from the most recent message contains one or more violations of the MIL-STD-1553 message validation criteria, including sync, encoding, parity, bit count or word count errors. This bit is also set if the received Status Word from the responding RT contains incorrect RT address bits 15:10.
8	NORESP	R	0	No Response Error. This bit is set to logic 1 when an RT fails to respond to a command, or responds later than the BC No Response Timeout time. The No Response Timeout delay is programmed using BC Timeout Select bits 15-14 in the “BC (Bus Controller) Configuration Register (0x0032)”.
7 − 2	GP7 − GP2	R	0	General Purpose Flags 7-2. Interpretation of these flag bits is user defined. These bits are set cleared or toggled by the host, through use of the “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”, or by the BC, through use of the FLG instruction op code.
1	EQ / GP1	R	0	Equal / General Purpose Flag 1. This flag bit is manipulated using the same methods as General Purpose Flags 7-2, or may be set or cleared by two BC instruction op codes, Compare to Frame Time Counter (CFT) or Compare to Message Time Counter (CMT).
0	LT / GP0	R	0	Less Than / General Purpose Flag 0 This flag bit is manipulated using the same methods as General Purpose Flags 7-2, or may be set or cleared by two BC instruction op codes, Compare to Frame Time Counter (CFT) or Compared to Message Time Counter (CMT).

13.7. Bus Controller (BC) General Purpose Flag Register (Write 0x0037)

CLR GPF7	CLR GPF6	CLR GPF5	CLR GPF4	CLR GPF3	CLR GPF2	CLR GPF1	CLR GPF0	SET GPF7	SET GPF6	SET GPF5	SET GPF4	SET GPF3	SET GPF2	SET GPF1	SET GPF0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
W																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Sharing the same register address as the Read-Only “Bus Controller (BC) Condition Code Register (Read 0x0037)”, this 16-bit register is Write-Only. This register is written by the host to set, clear or toggle any combination of the 8 general-purpose flags 7-0 in the “Bus Controller (BC) Condition Code Register (Read 0x0037)”. When this register is written, general-purpose flags are modified. Reading register address 0x0037 returns the value in the BC Condition Code Register, containing the modified GP flag bits.

Each general-purpose flag in the BC Condition Code Register is mirrored twice in the General Purpose Flag Register, once in the upper byte and once in the lower byte. Bits asserted in the lower byte set the corresponding GP flag bits in the BC Condition Code Register to 1. Bits asserted in the upper byte clear the corresponding GP flag bits in the BC Condition Code Register to 0. Bits asserted in both the lower and upper bytes for a specific GP flag toggles (inverts) the corresponding GP flag bit in the BC Condition Code Register. When both bits are written to logic 0 state for a specific GP flag bit, no change occurs for that GP flag bit. The FLG instruction op code operates similarly, as shown in the diagram in Figure 9. **Writes to this register have no effect unless the BC is already running.** The BCENA bit 12 in “Master Configuration Register 1 (0x0000)” must have previously been written high.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15 − 8	CLEAR GP7 − GP0	W	0	Clear General Purpose Flag 7-0. Bits asserted in the upper byte clear the corresponding GP flag bits in the BC Condition Code Register to 0.
7 − 0	SET GP7 − GP0	W	0	Set General Purpose Flag 7-0. Bits asserted in the lower byte set the corresponding GP flag bits in the BC Condition Code Register to 1. Bits asserted in both the lower and upper bytes for GPx toggles that GP flag bit in the BC Condition Code Register.

13.8. Bus Controller (BC) General Purpose Queue Pointer Register (0x0038)

MSB			General Purpose Queue RAM Address												LSB	
0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	Reset
R	RW															Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is a combination of Read-Only and Read-Write bits. This register contains 0x00C0 after $\overline{\text{RESET}}$ signal master reset. The initialized value represents the base address for the 64-word BC General Purpose Queue. The host can overwrite the default 0x00C0 value, but low order bits 5-0 and bit 15 must equal logic 0 for the initialized value.

The general purpose queue provides a way for the Bus Controller message sequencer to convey various information to the external host. The BC instruction set includes op codes that push data values onto this queue, including immediate data values, the Block Status Word from the most recent message, the Time Tag Register count, or the contents of a specified memory address.

The General Purpose Queue is implemented as a 64-word circular buffer. **This register always points to the next queue address to be written,** the address following the last queue location written by the Bus Controller. This queue pointer rolls over from bits 5:0 = 11111 to 00000, every 64th word written. (Bits 15:6 are static.) If enabled in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, the BCGPQ interrupt will be generated each time queue pointer rollover occurs.

13.9. Bus Controller (BC) Time Tag Counter (0x0043)

MSB Register Value LSB

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access

13.10. Bus Controller (BC) Time Tag Counter High (0x0044)

MSB Register Value LSB

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access

These registers are read-only and are cleared after $\overline{\text{RESET}}$ signal Master Reset. The Bus Controller can be configured for either 16- or 32-bit time base counting in the “Time Tag Counter Configuration Register (0x0039)”. When configured for 16-bit time base operation, register 0x0043 contains the entire 16-bit count. When configured for 32-bit time base operation, count bits 31-16 reside in register 0x0044 while register 0x0043 contains bits 15-0.

For programmed bus controller action, instruction op codes are provided for loading a time tag count value, or pushing the current time tag count onto the BC General Purpose Queue. If configured for 32-bit time base operation, separate op codes are provided for loading the upper or lower words individually, or pushing the individual words or simultaneously pushing both words onto the BC General Purpose Queue:

Configuration	*Op Code*	*Description*
16-Bit Time Base	LTT LTH PTT PTH PTB	Load Time Tag Count (parameter) into register 0x0043. Not used for 16-bit time base. Push Time Tag Count from register 0x0043 onto BC GP Queue. Not used for 16-bit time base. Not used for 16-bit time base.
32-Bit Time Base	LTT LTH PTT PTH PTB	Load Low Time Tag Count (parameter) into register 0x0043. Load High Time Tag Count (parameter) into register 0x0044. Push Low Time Tag Count from register 0x0043 onto BC GP Queue. Push High Time Tag Count from register 0x0044 onto BC GP Queue. Push Low and High Time Tag Counts from register 0x0043 and 0x0044 onto BC GP Queue (simultaneous 32-bit count capture)

The host can bypass BC Instruction List execution to exercise direct control over the BC Time Tag counter. By writing bits 13-12 in the “Time Tag Counter Configuration Register (0x0039)”, the host can clear time tag count to zero, or load the current value contained in the BC Time Tag Utility Register(s) into the BC Time Tag counter(s). Finally, the BC Time Tag Match Register(s) provide capability for host interrupts when the time tag count reaches any predetermined 16- or 32-bit value.

13.11. Bus Controller (BC) Time Tag Utility Register (0x0045)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.12. Bus Controller (BC) Time Tag Utility High Register (0x0046)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These registers are read-write and are cleared after $\overline{\text{RESET}}$ signal Master Reset. This utility register pair is used for simultaneously loading a 16- or 32-bit value into the BC Time Tag Counter. When loading, the value contained in utility register 0x0045 is copied into “Bus Controller (BC) Time Tag Counter (0x0043)”. If the BC is configured for 16-bit time base, register 0x0043 contains the entire 16-bit count. If configured for 32-bit time base operation, count bits 31-16 are simultaneously copied from utility register 0x0046 into “Bus Controller (BC) Time Tag Counter High (0x0044)”. Please refer to the description for bits 13-12 in the “Time Tag Counter Configuration Register (0x0039)” on page 50.

13.13. Bus Controller (BC) Time Tag Match Register (0x0047)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.14. Bus Controller (BC) Time Tag Match High Register (0x0048)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These registers are read-write and are cleared after $\overline{\text{RESET}}$ signal Master Reset. When the BCTTM bit 5 is logic 1 in the “Hardware Interrupt Enable Register (0x000F)”, an interrupt occurs when the BC time tag count matches the value stored in this register pair. If the BC is configured for 16-bit time base, match register 0x0047 is compared to time base count register 0x0043 for match determination. If configured for 32-bit time base operation, count bits 31-16 in match register 0x0048 is also compared to “Bus Controller (BC) Time Tag Counter High (0x0044)” for match determination. Please refer to the description for BCTTM bit 5 in the Hardware Interrupt Registers in Section 11.10.

13.15. Bus Controller Interrupt Registers and Their Use

Section 11.7 on page 39 through Section 11.9 describe how the host uses three Hardware Interrupt registers, the Interrupt Log Buffer and the Interrupt Count & Log Address Register to manage interrupts. When the Bus Controller is enabled, three additional registers are dedicated to Bus Controller interrupts. Comparable to the Hardware Interrupt register triplet, the Bus Controller has

- A “Bus Controller (BC) Interrupt Enable Register (0x0010)” to enable and disable interrupts
- A “Bus Controller (BC) Pending Interrupt Register (0x0007)” to capture the occurrence of enabled interrupts

- A “Bus Controller (BC) Interrupt Output Enable Register (0x0014)” to enable $\overline{\text{INT}}$ output to host, for pending enabled interrupts

Each individual bit in all three registers is mapped to the same interrupt-causing event when the corresponding interrupt condition is enabled. Numerous interrupt options are available for the BC. At initialization, bits are set in the “Bus Controller (BC) Interrupt Enable Register (0x0010)” to identify the interrupt-causing events for the BC which are heeded by the IP core. Most Bus Controller applications only use a subset of available BC interrupt options. Interrupt-causing events are ignored when their corresponding bits are reset in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”. Setting an Interrupt Enable Register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

Whenever a Bus Controller interrupt event occurs (and the corresponding bit is already set in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”), these actions occur:

- The Interrupt Log Buffer is updated.
- A bit corresponding to the interrupt type is set in the “Bus Controller (BC) Pending Interrupt Register (0x0007)”. The type bit is logically-ORed with the preexisting register value, retaining bits for prior, unserviced BC interrupts.
- BC Interrupt Pending (BCIP) bit 0 is set in the “Hardware Pending Interrupt Register (0x0006)”. The BCIP bit is logically-ORed with the preexisting register value, retaining bits for unserviced hardware interrupts and the preexisting status of the MTIP and RTIP (Bus Monitor and RT) interrupt pending bits.
- If the matching bit is already set in the “Bus Controller (BC) Interrupt Output Enable Register (0x0014)”, an $\overline{\text{INT}}$ output occurs.

If the matching bit in the “Bus Controller (BC) Interrupt Output Enable Register (0x0014)” was not already set (i.e., low priority polled interrupt), the host can poll the “Bus Controller (BC) Pending Interrupt Register (0x0007)” to detect the occurrence of BC interrupts, indicated by non-zero value. Reading the “Bus Controller (BC) Pending Interrupt Register (0x0007)” automatically clears it to 0x0000.

A single $\overline{\text{INT}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the four Interrupt Output Enable Registers (hardware, BC, RT and SMT). Multiple interrupt-causing events can occur simultaneously, so single or simultaneous interrupt events can assert the $\overline{\text{INT}}$ host interrupt output.

When the host receives an $\overline{\text{INT}}$ signal from the IP core, it identifies the event(s) that triggered the interrupt. Section 11.7 describes two methods for identifying the interrupt source(s). One scheme uses the three low order bits in the “Hardware Pending Interrupt Register (0x0006)” to indicate when BC, RT and SMT interrupts occur. When BCIP (BC Interrupt Pending) bit 0 is set in the “Hardware Pending Interrupt Register (0x0006)”, the “Bus Controller (BC) Pending Interrupt Register (0x0007)” contains a nonzero value and may be read next to identify the specific BC interrupt event(s). Or, the host can directly interrogate the Interrupt Count & Log Address Register, followed by the Interrupt Log Buffer. Data sheet section 11.7 has a detailed description.

13.15.1. Bus Controller (BC) Interrupt Enable Register (0x0010)

BCWDT	SELMSG	BCGPQ	BCRETRY	CSTKERR	BCTRAP	STATSET	BCIRQ3	BCIRQ2	BCIRQ1	BCIRQ0	BCMERR	BCEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.15.2. Bus Controller (BC) Pending Interrupt Register (0x0007)

BCWDT	SELMSG	BCGPQ	BCRETRY	CSTKERR	BCTRAP	STATSET	BCIRQ3	BCIRQ2	BCIRQ1	BCIRQ0	BCMERR	BCEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.15.3. Bus Controller (BC) Interrupt Output Enable Register (0x0014)

BCWDT	SELMSG	BCGPQ	BCRETRY	CSTKERR	BCTRAP	STATSET	BCIRQ3	BCIRQ2	BCIRQ1	BCIRQ0	BCMERR	BCEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Three registers govern BC interrupt behavior: the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, the “Bus Controller (BC) Pending Interrupt Register (0x0007)” and the “Bus Controller (BC) Interrupt Output Enable Register (0x0014)”. When a bit is set in the “Bus Controller (BC) Interrupt Enable Register (0x0010)”, the corresponding BC interrupt is enabled. When a bit is reset in this register, the corresponding interrupt event is unconditionally disregarded. Setting a register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

When an enabled BC interrupt event occurs, the corresponding bit is set in the “Bus Controller (BC) Pending Interrupt Register (0x0007)” and the Interrupt Log Buffer is updated. To simplify interrupt decoding, BCIP bit 0 in the “Hardware Pending Interrupt Register (0x0006)” is also set whenever a message sets at least one bit in the “Bus Controller (BC) Pending Interrupt Register (0x0007)”.

If the corresponding bit is set in the “Bus Controller (BC) Interrupt Output Enable Register (0x0014)”, the $\overline{\text{INT}}$ output is asserted at message completion. The “Bus Controller (BC) Interrupt Output Enable Register (0x0014)” establishes two priority levels: high priority interrupts generate an $\overline{\text{INT}}$ output while low priority interrupts do not. Both priority levels update the “Bus Controller (BC) Pending Interrupt Register (0x0007)” and Interrupt Log Buffer. The host can detect low priority (masked) interrupts by polling Pending Interrupt registers.

The table below describes common bits in all three BC interrupt registers.

Bit No.	*Mnemonic*	*Function*
15	BCWDT	BC Watchdog Timer Interrupt. The BC Frame Timer expired.
14	SELMSG	BC Selected Message Interrupt. The completion of a MIL-STD-1553 message that has bit 4 set (EOM) in the message block Control Word.
13	BCGPQ	BC General Purpose Queue Rollover Interrupt. The 64-word circular BC General Purpose Queue Pointer rolled over to its base address value.
12	BCRETRY	BC Retry Interrupt. The occurrence of a retried message by the BC. If enabled, the interrupt will occur after the last enabled message retry (one or two) regardless of the outcome, successful or unsuccessful.
11	CSTKERR	BC Call Stack Pointer Error Interrupt. The BC subroutine stack depth was violated due to an overflow or underflow condition. Call stack level is incremented each time a BC “subroutine call” op code (CAL) is executed. Call stack level is decremented each time a BC “subroutine return” op code (RTN) is executed. The allowed range for the call stack level is 0-7. An interrupt occurs when a CAL op code executes when stack level is 7. An interrupt also occurs when a RTN op code executes when stack level is 0.
10	BCTRAP	BC Trap Interrupt. Two conditions can assert this interrupt: The BC fetched an illegal op code. The BC operation stops when the current 1553 message is complete. An illegal op code is either undefined, fails parity check, and/or has the wrong value for bits 9-5. When this occurs, BCRUN bit 15 resets to logic 0 in the “Bus Controller (BC) Condition Code Register (Read 0x0037)” and “Bus Controller (BC) General Purpose Flag Register (Write 0x0037)”. When the CHKFMT bit 13 is set in the “BC (Bus Controller) Configuration Register (0x0032)”, the “BC Control Word” message format bits 2-0 (mode command, RT-RT and broadcast message flags) are compared to the stored value(s) for the message Command Word(s) following the Control Word in the Message Block. When mismatch occurs between Control Word format bits and Command Word(s), the BCTRAP interrupt is asserted, if enabled. BC instruction list execution continues, so this condition can be differentiated from illegal op code because BCRUN bit 15 remains high in the BC Condition Code and GP Flag register, 0x0037. For mode command or broadcast mismatch, the stored message block Command Word(s) are transmitted. For RT-RT format mismatch, RT-RT bit 1 in the Control Word has priority, determining whether the message block is treated as an 8- or 16-word entity. For RT-RT format mismatch, **message failure is likely for this message or the next message block**, since the message block boundary is misplaced. The CHKFMT option bit detects BC programming problems in the development phase. The option is normally disabled in the field.

Bit No.	*Mnemonic*	*Function*
9	STATSET	BC Status Set Interrupt. The BC received an RT Status Word containing the wrong RT address field, or having an unexpected bit value for at least one of the eight non-reserved status bits. The expected value for these bits (excluding the BCR bit) is usually 0. The BCR (broadcast command received) bit can have an expected value of 1 when BCR Mask Enable bit in the “BC (Bus Controller) Configuration Register (0x0032)” is logic 0. In this case, the Mask Broadcast bit in the message block Control Word shows the expected value of the Status Word BCR bit. If the Control Word’s Mask Broadcast bit is logic 1, the expected value of BCR in the RT Status Word is logic 1.
8 − 5	BCINT3:0	BC Interrupt Request Bits 3-0. When this 4-bit field is nonzero, the BC executed an INT op code. The value of bits 8:5 will equal the value of the 4 LSBs in the parameter associated with the INT op code. The user may define the 4- bit pattern to suit application requirements.
4	BCMERR	BC Message Error Interrupt. Any one of the following five conditions will assert this interrupt if enabled 1. A non-broadcast message ended with RT Status Word containing the ME Message Error status bit set. 2. RT response time-out. 3. BC loopback failure. 4. Incorrect address in RT status word. 5. Minimum gap time violated (if enabled).
3	BCEOM	BC End of Message Interrupt. The successful completion of a message, regardless of validity.
2 − 0	Reserved	Bits 2-0 cannot be written, and read back 000.

14. SIMPLE MONITOR TERMINAL (SMT)

The HI-6300 can operate as an autonomous MIL-STD-1553 Bus Monitor, requiring minimal host support.

14.1. Overview

Simple Monitor Terminal (SMT) has its own dedicated Time Tag counter, and can use either a 16- or 48-bit Time Tag scheme. The SMT monitor utilizes two circular buffers in RAM: a Command Buffer and a Data Buffer. Each recorded MIL-STD-1553 message appends a fixed length entry into the Command Buffer and a variable length entry into the Data Buffer.

The SMT message records a fixed length "message block" in the Command Buffer for each MIL-STD-1553 message. The advantage of fixed length Command Buffer message blocks is that the host can quickly jump to the block start address for any message.

The number of words added to the Data Buffer for each message depends on the MIL-STD-1553 message type, ranging from zero (broadcast mode command without data) to 35 words (for a 32 data word RT-RT command).

Both circular buffers are fully utilized for recording message data. The SMT monitor allows selective monitoring of MIL-STD-1553 messages, based on the address, subaddress and T/$\overline{R}$ status in each monitored Command Word, or can monitor all messages, when preferred. The SMT monitor offers flexible interrupt options.

In "Master Configuration Register 1 (0x0000)", MTENA bit 8 enables the SMT monitor. If "Master Configuration Register 1 (0x0000)" bit 8 equals logic 0, Bus Monitor operation is disabled. When "Master Configuration Register 1 (0x0000)" MTENA bit 8 is logic 1, the Bus Monitor is enabled. Operation commences when the receiver first decodes MIL-STD-1553 activity meeting the "start record" criteria selected by bits 6-5 in the "SMT Configuration Register (0x0029)". If monitor operation is underway when "Master Configuration Register 1 (0x0000)" MTENA bit 8 becomes logic 0, monitor operation stops after completion of any message already underway.

The HI-6300 is configured for SMT operation by writing bits 1-0 in the "SMT Configuration Register (0x0029)".

When "SMT Configuration Register (0x0029)" bits 1-0 equal 01, the SMT operates with 16-bit Time Tag resolution and each recorded MIL-STD-1553 message adds a four word entry in the Circular Command Buffer. This is summarized in Table 14.

Table 14. Message Block in Circular Command Buffer for SMT Monitor using 16-bit Time Tag

Message Word Block	***Word Name***	***Word Function when using 16-bit time tag***
Word 3	Message Command Word	Message Command Word. The MIL-STD-1553 Command Word that initiated the message. For an RT-RT message, Receive Command Word 1 is stored here; Transmit Command Word 2 is the first stored word in the Message Data Block.
Word 2	Data Block Pointer	Starting address in the Data Buffer for the corresponding message data block.
Word 1	Message Time Stamp Bits 15 ~ 0	Sixteen bit message time stamp. Word 0 is the first word in the Command Buffer entry for each message.

Message Word Block	***Word Name***	***Word Function when using 16-bit time tag***
Word 0	Block Status Word	Message Block Status Word, defined in Section 14.2. Word 0 is the first word in the Command Buffer entry for each message.

When "SMT Configuration Register (0x0029)" bits 1-0 equal 11, the Simple Message Monitor operates with 48-bit Time Tag resolution. Each MIL-STD-1553 message adds an 8-word entry in the Circular Command Buffer. This is summarized in Table 15. The expanded message block accommodates two additional Time Tag words, a Message Length word and a Response Time word not found when using 16-bit Time tag resolution.

Table 15. Message Block in Circular Command Buffer for SMT Monitor using 48-bit Time Tag

Message Word Block	***Word Name***	***Word Function when using 16-bit time tag***
Word 7	Message Command Word	Message Command Word. The MIL-STD-1553 Command Word that initiated the message. For an RT-RT message, Receive Command Word 1 is stored here; Transmit Command Word 2 is the first stored word in the Message Data Block.
Word 6	Data Block Pointer	Starting address in the Data Buffer for the corresponding message data block.
Word 5	Message Length Word (bytes)	The Message Length Word indicates the total number of bytes of the 1553 message including all command words and status words. The range is 2 to 72 bytes, corresponding to 1 to 36 16-bit words stored. **Note:** Since the Message Command Word is stored within the Message Block itself, the number of bytes to read from the Message Data Block should be reduced by 2 bytes. That is, the number of bytes to read from the Message Data Block = Message Length Word − 2.
Word 4	Response Time Word	The Response Time Word contains two 8-bit fields: • Bits 15 ~ 8 contains GAP2 • Bits 7 ~ 0 contains GAP1 All GAP values are measured from mid-parity zero crossing of the preceding word, to the mid-sync zero crossing of the Status Word (the gap "dead time" interval plus 2 μs). Time resolution is 100 ns per LSB, so the maximum indicated gap time for GAP1 or GAP2 is 25.5 μs. For RT-RT messages, the GAP1 byte indicates transmit RT response time, and the GAP2 byte indicates received RT response time. For all other messages, the GAP1 byte indicates the only RT response time, and the GAP2 byte reads 0x00.
Word 3	Block Status Word	Message Block Status Word, defined in Section 14.2.

Message Word Block	***Word Name***	***Word Function when using 16-bit time tag***
Word 2	Message Time Stamp Bits 47 ~ 32	Upper 16-bit word of message 48-bit time stamp.
Word 1	Message Time Stamp Bits 31 ~ 16	Middle 16-bit word of message 48-bit time stamp.
Word 0	Message Time Stamp Bits 15 ~ 0	Lower 16-bit word of message 48-bit time stamp. Word 0 is the first word in the Command Buffer entry for each message.

The Circular Command Buffer address range is bounded by the values in Address List Words 0 and 2. The Circular Data Buffer address range is bounded by the values in Address List Words 4 and 6. The “Next Address” Words 1 and 5 must be initialized by the host for the first data written after reset, usually to match the Word 0 and Word 4 values respectively. Thereafter, these values are maintained by the IP core each time a new MIL-STD-1553 message is recorded.

Two optional Buffer address interrupts are offered. When enabled, a Command or Data Buffer Address Interrupt occurs whenever the matching RAM address in the Buffer is written. The Address List contains the address values for these optional “buffer utilization” interrupts.

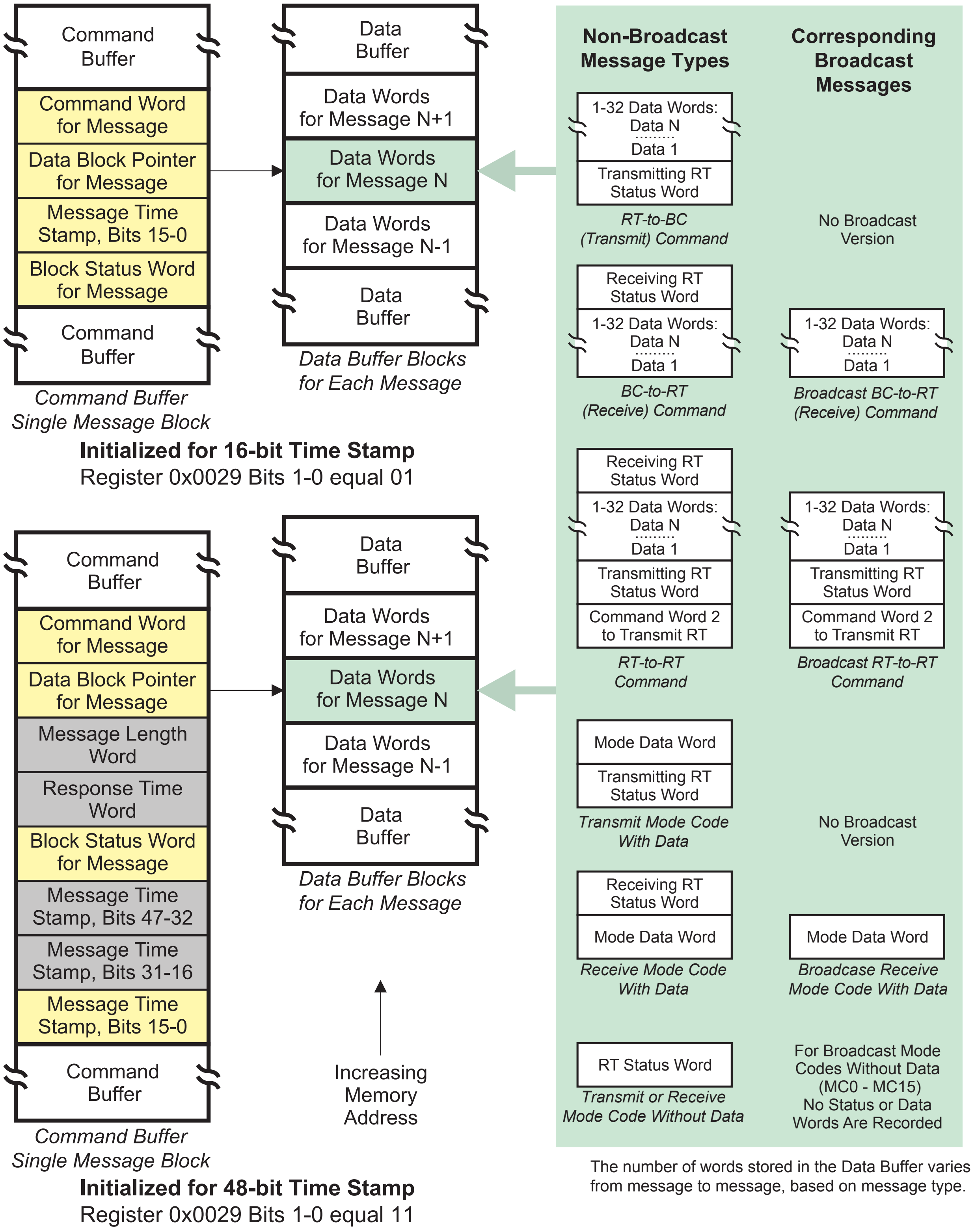
For SMT, the 8-word Monitor Address List is defined in Table 16.

Table 16. Monitor Address List for SMT

Address List Word	***Word Name***	***Description***
Word 7	Data Buffer Interrupt Address	Host initialized with a RAM address value if this interrupt is enabled. If enabled, an interrupt occurs when the matching RAM address is written. Address must occur within the range bounded by Words 4 and 6.
Word 6	Data Buffer End Address	Host initialized, defines SMT Data Buffer upper (rollover) address.
Word 5	Data Buffer Next Address	**Must be host initialized**, usually to match SMT Data Buffer Start Address. **Updated by IP core** each time a new MIL-STD-1553 message is recorded. This value advances through the address range in circular buffer fashion.
Word 4	Data Buffer Start Address	Host initialized, defines SMT Data Buffer lower address boundary.
Word 3	Command Buffer Interrupt Address	Host initialized with a RAM address value if this interrupt is enabled. If enabled, an interrupt occurs when the matching RAM address is written. Address must occur within the range bounded by Words 0 and 2.

Address List Word	*Word Name*	*Description*
Word 2	Command Buffer End Address	Host initialized, defines SMT Circular Command Buffer upper (rollover) address.
Word 1	Command Buffer Next Address	**Must be host initialized**, usually to match Command Buffer Start Address. **Updated by IP core** each time a new MIL-STD-1553 message is recorded. This value advances through the address range in circular buffer fashion.
Word 0	Command Buffer Start Address	Host initialized, defines SMT Circular Command Buffer lower address boundary. Word 0 occurs at the Address List base address in register 0x002F.

For each monitored MIL-STD-1553 command, the written Command Buffer entry is fixed at 4 or 8 words, depending on selected Time Tag resolution. Depending on MIL-STD-1553 message type, the written Data Buffer entry varies in length, ranging from zero words (for broadcast mode code commands without data) to 35 words (for an RT-to-RT message with 32 data words). Simple Monitor Terminal Data Storage is summarized in Figure 11.


Command Buffer
Command Word for Message
Data Block Pointer for Message
Message Time Stamp, Bits 15-0
Block Status Word for Message
Command Buffer
Command Buffer Single Message Block
Data Buffer
Data Words for Message N+1
Data Words for Message N
Data Words for Message N-1
Data Buffer
Data Buffer Blocks for Each Message
Initialized for 16-bit Time Stamp
Register 0x0029 Bits 1-0 equal 01
Command Buffer
Command Word for Message
Data Block Pointer for Message
Message Length Word
Response Time Word
Block Status Word for Message
Message Time Stamp, Bits 47-32
Message Time Stamp, Bits 31-16
Message Time Stamp, Bits 15-0
Command Buffer
Command Buffer Single Message Block
Data Buffer
Data Words for Message N+1
Data Words for Message N
Data Words for Message N-1
Data Buffer
Data Buffer Blocks for Each Message
Increasing Memory Address
Initialized for 48-bit Time Stamp
Register 0x0029 Bits 1-0 equal 11
Non-Broadcast Message Types
Corresponding Broadcast Messages
1-32 Data Words: Data N .... Data 1
Transmitting RT Status Word
RT-to-BC (Transmit) Command
No Broadcast Version
Receiving RT Status Word
1-32 Data Words: Data N .... Data 1
BC-to-RT (Receive) Command
1-32 Data Words: Data N .... Data 1
Broadcast BC-to-RT (Receive) Command
Receiving RT Status Word
1-32 Data Words: Data N .... Data 1
Transmitting RT Status Word
Command Word 2 to Transmit RT
RT-to-RT Command
1-32 Data Words: Data N .... Data 1
Transmitting RT Status Word
Command Word 2 to Transmit RT
Broadcast RT-to-RT Command
Mode Data Word
Transmitting RT Status Word
Transmit Mode Code With Data
No Broadcast Version
Receiving RT Status Word
Mode Data Word
Receive Mode Code With Data
Mode Data Word
Broadcase Receive Mode Code With Data
RT Status Word
Transmit or Receive Mode Code Without Data
For Broadcast Mode Codes Without Data (MC0 - MC15) No Status or Data Words Are Recorded

The number of words stored in the Data Buffer varies from message to message, based on message type.

Figure 11. Simple Monitor Terminal (SMT) Data Storage

14.2. SMT Block Status Word (BSW) Description

The SMT bus monitor stores a Block Status Word in the Circular Command Buffer for each monitored MIL-STD-1553 message. This word provides information regarding message status, the bus on which the message occurred, whether errors occurred during the message, and the type of occurring errors. The Block Status Word for SMT is defined as follows:

Reserved	Reserved	BID	EO	RR	IGE	TM	Reserved	Reserved	Reserved	LE	SE	WE	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The IP core offers an Extended Status reporting option, enabled when bit 2 in “SMT Configuration Register (0x0029)” is logic 1. When this option is enabled, addition status information is available in the SMT block Status Word.

EOM	SOM	BID	EO	RR	IGE	TM	GDB	DSR	SFS	LE	SE	WE	RRGSA	RRCW2	CWCE	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

SMT Block Status Word
with Extended Status Bits Enabled

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	EOM	R/W	0	End of Message. Bit 15 is set upon completion of a monitored message, whether or not errors occurred. When EOM is set, SOM bit 14 is concurrently reset.
14	SOM	R/W	0	Start of Message. Bit 14 is set to logic 1 approximately 3-4 µs after completion of a valid Command Word, and is reset to logic 0 at the end of the message. If the monitor uses message filtering, SOM is only set for monitored messages.
13	BID	R/W	0	Bus ID (Bus B / $\overline{\text{Bus A}}$). Bit 13 indicates the bus ID for the message. This bit is logic 0 for a message occurring on Bus A. This bit is logic 1 for a message occurring on Bus B.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
12	EO	R/W	0	Error Occurred Flag. This bit indicates a message error was encountered. This bit is set when one or more of the following conditions are true (logical-OR): • an unfinished message is superseded by another valid command • Bit 10 Illegal Gap Error is set • Bit 9 Response Timeout is set • Bit 5 Length (Word Count) Error is set • Bit 4 Sync Type Error is set • Bit 3 Invalid Word Error is set • Bit 2 RT-RT Gap / Sync / Address Error is set • Bit 1 RT-RT Command Word 2 Error is set (except as noted) • Bit 0 Command Word Content Error is set (except as noted) There are three exceptions where register bit 0 or 1 is set without affecting bit 12 state: **Bit 1 RT-RT Command Word 2 Errors that do not assert bit 12** • RT-RT Transmit Command Word 2 subaddress field equals 00000 or 11111 (mode code command indicated) • RT-RT Transmit Command Word 2 has the same RT Address as Receive Command Word 1 **Bit 0 Command Word Content Error that does not assert bit 12** • Undefined receive mode code 0~15 decimal.
11	RR	R/W	0	RT-to-RT Transfer When logic 1, bit 11 indicates an RT-to-RT message, beginning with two contiguous Command Words.
10	IGE	R/W	0	Illegal Gap Error When logic 1, bit 10 indicates an illegal gap occurred on the bus, other than Response Timeout.
9	TM	R/W	0	Response Timeout When logic 1, bit 9 indicates a response timeout occurred. This bit is set if an RT Status Word associated with this message failed to arrive within the response time interval specified by bits 15-14 in the “SMT Configuration Register (0x0029)”.
8	GDB	R/W	0	Good Data Block Transfer Bit 8 is set to logic 1 following completion of a valid, error-free message. This bit is reset to logic 0 following completion of a message in which error occurred. If an RT responds to a transmit command with Busy status and does not transmit the commanded data words, this is not considered a message error that causes GDB reset.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
7	DSR	R/W	0	Data Buffer Rollover Bit 7 is logic 1 to indicate that this message overran the monitor Data Buffer end address, causing the storage pointer to roll over to the base address.
6	SFS	R/W	0	Status Flag Set Bit 6 is logic 1 when a status bit was set in an RT Status Word response.
5	LE	R/W	0	Word Count (Length) Error Bit 5 indicates that the number of data words transmitted by the BC or RT differs from the Word Count specified in the Command Word. An RT Status Word with the Busy bit set will not cause Word Count Error. A transmit command with Response Timeout will not cause Word Count Error.
4	SE	R/W	0	Sync Type Error Bit 4 is logic 1 to indicate that a BC transmitted data sync with a Command Word, or a command / status sync occurred with Data Word, or an RT responded with data sync in its Status Word and/or command/ status sync in a Data Word.
3	WE	R/W	0	Invalid Word Error (WE) Bit 3 is logic 1 indicate on invalid word error occurred. This includes Manchester decoding errors in the sync pattern or word bits, or the wrong number of bits in the word, or parity error.
2	RRGSA	R/W	0	RT-to-RT Gap/Sync/Address Error (RRGSA) Bit 2 is logic 1 if one or more of the following RT-RT message conditions occur: • MT Gap Check is enabled (bit 12 equals 1 in “SMT Configuration Register (0x0029)”) and an RT Status Word is received having a response time less than 4µs, per MIL-STD-1553B (mid-parity to mid-sync). In other words, the bus “dead time” was less than 2µs. • One of the RTs responds with an invalid Status Word, having a sync error, a Manchester encoding error, bit count error and/or parity error • One of the RT Status Words contains an RT Address that differs from the RT Address in the corresponding Command Word.
1	RRCW2	R/W	0	RT-to-RT Command Word 2 Error (RRCW2) Bit 1 is logic 1 if an RT-to-RT message occurs (two contiguous Command Words) with one or more of the following illogical conditions: • Transmit Command Word 2 $T/\overline{R}$ bit equals 0 (receive) • Transmit Command Word 2 subaddress field equals 00000 or 11111 (mode command indicated) • Transmit Command Word 2 has the same RT Address as Receive Command Word 1 • Transmit Command Word 2 has sync error

Bit No.	Mnemonic	R/W	Reset	Function
0	CWCE	R/W	0	Command Word Content Error (CWCE) Bit 0 is logic 1 if a received Command Word violates one or more MIL-STD-1553B requirements: • A non-mode code broadcast transmit Command Word occurred. (non-mode has 5-bit subaddress field equal to decimal 1~30) • A receive mode code Command Word was received with mode code in the range of 0~15 decimal (undefined) • A broadcast transmit mode code command occurred having a mode code value for which broadcast is not allowed (mode code = decimal 0, 2, 16, 18 or 19)

14.3. SMT Message Filter Table

The Simple Monitor Terminal can select messages for monitoring through the use of a 128-word MT Filter Table, located at fixed RAM address 0x0100. When the table bit corresponding to a new message Command Word is logic 1, that message is ignored by the monitor. If the table bit is logic 0, that message is recorded.

After $\overline{\text{RESET}}$ master reset, 100% of MIL-STD-1553 messages are monitored, since the entire table address range 0x0100 through 0x017F inclusive is 0x0000. The result is that every valid Command Word, received on an idle bus, marks the start of a new MIL-STD-1553 message recorded by the monitor. The Message Filter Table is addressed using three fields in the received Command Word: the 5-bit RT Address field, the T/$\overline{\text{R}}$ Transmit/Receive bit, and the MSB of the 5-bit Subaddress field. This is illustrated in Figure 12.

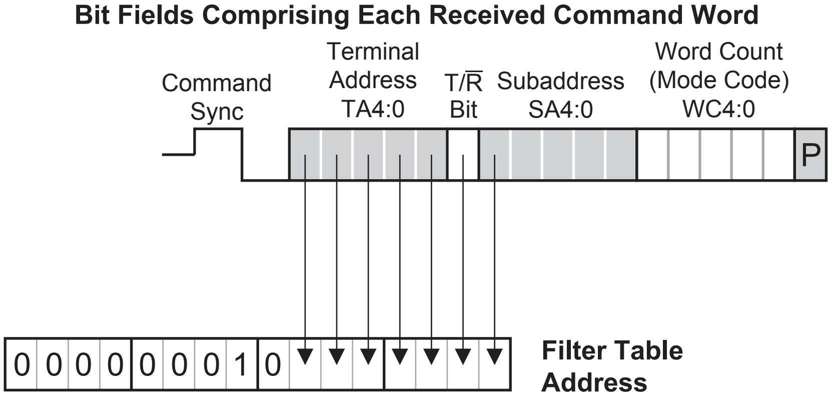


Figure 12. Deriving the Monitor Filter Table Address from the Received Command Word

Each RT Address from 0 to 31 decimal has four 16-bit table words: two words enable/disable individual Receive Subaddresses, two more words enable/disable individual Transmit Subaddresses. The first four table words apply to Subaddress 0 and are illustrated in Table 17. This 4-word pattern repeats for all 32 Subaddresses, 0-31 decimal.

Table 17. SMT Message Filter Table

Filter Table addresses 0x017C - 0x017F	**RT Address 31 Subaddresses (4 words)**															
Filter Table addresses 0x0178 - 0x017B	**RT Address 30 Subaddresses (4 words)**															
. . .	. . .															
Filter Table addresses 0x0108 - 0x010B	**RT Address 2 Subaddresses (4 words)**															
Filter Table addresses 0x0104 - 0x0107	**RT Address 1 Subaddresses (4 words)**															
Filter Table address 0x0103	**RT Address 0, Transmit Subaddresses 31 to 16**															
Word Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Transmit SA	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Filter Table address 0x0102	**RT Address 0, Transmit Subaddresses 15 to 0**															
Word Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Transmit SA	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Filter Table address 0x0101	**RT Address 0, Receive Subaddresses 31 to 16**															
Word Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Receive SA	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Filter Table address 0x0100	**RT Address 0, Receive Subaddresses 15 to 0**															
Word Bit #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Receive SA	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

A subaddress message is monitored when the corresponding word bit equals logic 0. The message is not monitored when the bit equals 1.

15. SIMPLE MONITOR TERMINAL (SMT) REGISTER DESCRIPTION

15.1. SMT Configuration Register (0x0029)

	MTTO1	MTTO0	Reserved	GCHK	Reserved	Reserved	MTTTB1	MTTTB0	Reserved	MTSRR1	MTSRR0	MTCRIW	Reserved	MTXMF	48BTT	Reserved
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
Host Access	RW		W	RW												R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th><i>Function</i></th></tr>
<tr><td>15 − 14</td><td>MTTO1:0</td><td>R/W</td><td>0</td><td>MT Time Out Select.
This 2-bit field selects the Monitor “no response” time-out delay from four available selections. Excluding RT-RT commands, the delay is measured from command word mid-parity bit to status word mid-sync.
<table>
<tr><th><b><i>Bit 15:14</i></b></th><th><b><i>Bus Dead Time</i></b></th><th><b><i>Time Out (excludes RT-RT)</i></b></th><th><b><i>RT-RT Time Out</i></b></th></tr>
<tr><td>00</td><td>16µs</td><td>18µs</td><td>61µs</td></tr>
<tr><td>01</td><td>21µs</td><td>23µs</td><td>66µs</td></tr>
<tr><td>10</td><td>80µs</td><td>82µs</td><td>122µs</td></tr>
<tr><td>11</td><td>138µs</td><td>140µs</td><td>180µs</td></tr>
</table>
For RT-RT commands, time out delay is measured per Figure 8 in the RT Validation Test Plan, SAE AS4111. That is, from mid-parity of the receive command to mid-sync of the first received data word. This adds 40µs for the embedded transmit command word and transmit-RT status word within this interval.</td></tr>
<tr><td>13</td><td>Reserved</td><td>W</td><td>0</td><td>Bit 13 is not used by the bus monitor operating in SMT mode.
Initialize this bit to logic 0.</td></tr>
<tr><td>12</td><td>GCHK</td><td>R/W</td><td>0</td><td>Gap Check.
When this bit equals 1, the monitor evaluates inter-message gaps and RT response times for a minimum preceding bus dead time of 2 µs. This dead time corresponds to an inter-message gap of 4µs, measured per MIL-STD-1553, from mid-parity zero crossing of the preceding word, to mid-sync zero crossing of the following word. A minimum gap time violation results in a Format Error in the Block Status Word for the message.
When this bit equals 0 (recommended), the monitor does not check for short inter-message gap times.</td></tr>
<tr><td>11</td><td>Reserved</td><td>R/W</td><td>1</td><td>Bit 11 is not used.</td></tr>
<tr><td>10</td><td>Reserved</td><td>R/W</td><td>0</td><td>Bit 10 is not used.</td></tr>
</table>

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th colspan="2">Function</th></tr>
<tr><td rowspan="7">9 − 8</td><td rowspan="7">MTTTB1:0</td><td rowspan="7">R/W</td><td rowspan="7">0</td><td colspan="2">Monitor Time Tag Message Bit Select.
This 2-bit field selects the bit within the MIL-STD-1553 message where time stamp occurs. Time stamp occurs at mid-bit transition:</td></tr>
<tr><td><b><i>Bit 9:8</i></b></td><td><b><i>Time Tag Event</i></b></td></tr>
<tr><td>00</td><td>Last Bit of Last Word in Message</td></tr>
<tr><td>01</td><td>First Bit of First (Command) Word in Message</td></tr>
<tr><td>10</td><td>Last Bit of First (Command) Word in Message</td></tr>
<tr><td>11</td><td>Time tag disabled, stores time tag = 0</td></tr>
<tr><td colspan="2">For options 00 and 10, the “Last Bit” precedes the word’s parity bit. For option 01, the “First Bit” occurs 0.5μs after command sync. While “First Word” generally denotes a command word, message recording can begin with a data word when register bit 5 equals 1.</td></tr>
<tr><td>7</td><td>Reserved</td><td>R/W</td><td>0</td><td colspan="2">Bit 7 is not used by the bus monitor.
Initialize this bit to logic 0.</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
6 − 5	MTSRR1:0	R/W	0	MT Start-Record Requirement 1:0. When register bits 6-5 equal 00, the MT starts recording a new MIL-STD-1553 message when a **properly encoded, complete MIL-STD-1553 word with command sync** is decoded: The command sync is followed by 16 properly encoded data bits plus a 17th parity bit expressing odd parity. No data is recorded until this condition is met. This is the usual setting. (default setting) When register bits 6-5 equal 01, the MT starts recording a new MIL-STD-1553 message when a **properly encoded, complete MIL-STD-1553 word with command sync or data sync** is decoded. The properly encoded command sync (or data sync) is followed by 16 properly encoded data bits plus a 17th parity bit expressing odd parity. If recording begins with data sync, the Sync Error flag will be set in the Block Status Word. When register bits 6-5 equal 10, the MT starts recording a new MIL-STD-1553 message upon detection of a **properly encoded command sync with two contiguous data bits**. If the properly encoded command sync with two contiguous data bits does not result in a valid command word, the Invalid Word Error is set in the Block Status Word. This selection begins recording for complete MIL-STD-1553 command words as well as for command word fragments, or command words with bad parity. Under some circumstances, this record option might be helpful for debugging MIL-STD-1553 communication failure. When register bits 6-5 equal 11, the MT starts recording new bus activity upon detection of **any properly encoded sync (command or data) with two contiguous data bits**. This selection begins recording for complete MIL-STD-1553 command or data words as well as for word fragments, or words with bad parity. If the properly encoded sync with two contiguous data bits does not result in a valid Manchester II word, the Invalid Word Error is set in the Block Status Word. If recording begins with data sync, the Sync Error flag will be set in the Block Status Word. Under some circumstances, this record option might be helpful for debugging MIL-STD-1553 communication failure.
4	MTCRIW	R/W	0	MT Continue Recording After Invalid Word. When bit 4 equals 0, the MT stops recording an incomplete message when an invalid MIL-STD-1553 word is decoded. The invalid word is not stored, and the MT awaits word detection per register bits 6-5 before the next MIL-STD-1553 message is recorded. (default) When bit 4 equals 1, the MT continues recording an incomplete message when an invalid MIL-STD-1553 word is decoded. The invalid word is stored and the MT continues monitoring the message until completion or time-out occurs.
3	Reserved	R/W	0	Bit 3 is not used by the bus monitor. Initialize this bit to logic 0.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
2	MTXMF	R/W	0	Extended Message Flag Enable. Usually register bit 2 is set to logic 1 to enable expanded status/error flags, occupying the “reserved” bit positions in the SMT Block Status Word. When register bit 2 equals 0, the recorded status/error flags are limited (see “SMT Block Status Word (BSW) Description”).
1	48BTT	R/W	0	48-Bit Time Tag / $\overline{\text{16-Bit Time Tag}}$ When register bit 1 equals 0, the SMT time tag counter operates with 16-bit resolution and the recorded entry for each MIL-STD-1553 message in the Command Buffer is four 16-bit words. When register bit 1 equals 1, the SMT time tag counter operates with 48-bit resolution. To record the 48-bit time count, the entry for each MIL-STD-1553 message in the Command Buffer is eight 16-bit words. Two of the words added are used for Response Time and Message Length words. See Section 15.5.
0	Reserved	R	1	This bit is not used and always reads logic “1”.

15.2. SMT Bus Monitor Address List Start Address Register (0x002F)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the host. After $\overline{\text{RESET}}$ signal master reset, this register is initialized with 0x00B0, the default base address of the MT Address Table in IP core RAM. The host can overwrite the default base address. Base address 0x00B8 provides an alocated location for a second MT Address Table (see “Figure 6. Address Mapping for Registers and RAM”). This register is not affected by MT soft reset, when the MTRESET bit is asserted in the “Master Status and Reset Register (0x0001)”. The Address List for SMT mode is summarized in Table 16 on page 100.

15.3. SMT Next Message Command Buffer Address (0x0030)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is read-only and is updated by the MT upon completion of a monitored MIL-STD-1553 message. This register is cleared after $\overline{\text{RESET}}$ signal master reset or by MT soft reset, when the MTRESET bit is asserted in the “Master Status and Reset Register (0x0001)”. This register contains the address for the first word to be stored in MT Command Buffer, for the next MIL-STD-1553 message. After the first post-reset message is logged, this register mirrors the value contained in SMT Address List word 1 (see Table 16 on page 100).

The MT logic only updates this “next message address” register after message completion. Therefore, after reset or after the host has changed the “SMT Bus Monitor Address List Start Address Register (0x002F)”, this register does not contain a pointer address until processing for the next message is completed. If the read value equals zero, the "next message address" is the Command Buffer starting address.

15.4. SMT Last Message Command Buffer Address (0x0031)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is read-only and is updated by the MT upon completion of a monitored MIL-STD-1553 message. This register is cleared after $\overline{\text{RESET}}$ signal master reset or by MT soft reset, when the MTRESET bit is asserted in the "Master Status and Reset Register (0x0001)".

This register contains the RAM address for the first word stored in the Circular Command Buffer for the last completed MIL-STD-1553 message.

15.5. SMT Bus Monitor Time Tag Count Register (0x003A)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

When MT Configuration Register bits 1-0 equal 01, the Simple Message Monitor operates with 16-bit Time Tag resolution and register 0x003A contains the full 16-bit Time Tag count.

When MT Configuration Register bits 1-0 equal 11, the Simple Message Monitor operates with 48-bit Time Tag resolution and the full Time Tag Count requires the above register plus two additional registers:

15.6. SMT Bus Monitor Time Tag Count Mid Register (0x003B)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

15.7. SMT Bus Monitor Time Tag Count High Register (0x003C)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

When configured for 48-bit time base operation, count bits 47-17 reside in register 0x003C, count bits 31-16 reside in register 0x003B while register 0x003A contains bits 15-0.

The host cannot directly write these registers but uses other methods to control or read Time Tag count. By writing bits 15-14 in the "Time Tag Counter Configuration Register (0x0039)", the host can clear time tag count to zero, copy the current time count to the SMT Time Tag Utility Register(s), or load the current value contained in the SMT Time Tag Utility Register(s) into the SMT Time Tag counter(s). Finally, the SMT Time Tag Match Register(s) provide capability for host interrupts when the time tag count reaches any predetermined 16- or 48-bit value. For further information, refer to the description of the “Time Tag Counter Configuration Register (0x0039)”.

15.8. SMT Bus Monitor Time Tag Utility Register (0x003D)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

When “SMT Configuration Register (0x0029)” bits 1-0 equal 01, the Simple Message Monitor operates with 16-bit Time Tag resolution and register 0x003D is the only Time Tag Utility register needed.

When “SMT Configuration Register (0x0029)” bits 1-0 equal 11, the Simple Message Monitor operates with 48-bit Time Tag resolution and “utility” operations require the above register plus two additional registers:

15.9. SMT Bus Monitor Time Tag Utility Mid Register (0x003E)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

15.10. SMT Bus Monitor Time Tag Utility High Register (0x003F)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These registers are read-write and are cleared after $\overline{\text{RESET}}$ signal Master Reset. This utility register triplet is used for simultaneously loading or reading a 16- or 48-bit value into or from the SMT Time Tag Counter. Please refer to the description for bits 15-14 in the “Time Tag Counter Configuration Register (0x0039)”.

Loading a 16-bit or 48-bit value into the SMT Time Tag Count Register(s)

When loading or clearing Time Tag count, the 16-bit value in ”SMT Bus Monitor Time Tag Utility Register (0x003D)” is copied into ”SMT Bus Monitor Time Tag Count Register (0x003A)”. If configured for 48-bit time stamp operation, count bits 47-17 and count bits 31-16 are simultaneously copied from Time Tag Utility Registers 0x003F and 0x003E into SMT Time Tag Count Registers 0x003C and 0x003B respectively.

Capturing a 16-bit or 48-bit value from the SMT Time Tag Count Register(s)

When capturing Time Tag count, the 16-bit value in ”SMT Bus Monitor Time Tag Count Register (0x003A)” is copied into “SMT Bus Monitor Time Tag Utility Register (0x003D)”. If configured for 48-bit time stamp operation, count bits 47-17 and count bits 31-16 in SMT Time Tag Count Registers 0x003C and 0x003B are simultaneously copied into Time Tag Utility Registers 0x003F and 0x003E respectively.

15.11. SMT Bus Monitor Time Tag Match Register (0x0040)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
							RW									Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

When “SMT Configuration Register (0x0029)” bits 1-0 equal 01, the Simple Message Monitor operates with 16-bit Time Tag resolution and register 0x0040 is the only Time Tag Match register needed.

When “SMT Configuration Register (0x0029)” bits 1-0 equal 11, the Simple Message Monitor operates with 48-bit Time Tag resolution and time tag matching operations require the above register plus two additional registers:

15.12. SMT Bus Monitor Time Tag Match Mid Register (0x0041)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
							RW									Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

15.13. SMT Bus Monitor Time Tag Match High Register (0x0042)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
							RW									Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These registers are read-write and are cleared after $\overline{\text{RESET}}$ signal Master Reset. When the MTTTM bit 6 is logic 1 in the “Hardware Interrupt Enable Register (0x000F)”, an interrupt occurs when the MT time tag count matches the value stored in this register triplet. If the MT is configured for 16-bit time tag, Time Tag Match Register 0x0040 is compared to Time Tag Count register 0x003A for match determination.

If configured for 48-bit time tag operation, count bits 47-17 and 31-16 in Time Tag Match Registers 0x0042 and 0x0041 are also compared to MT Time Tag Count Registers 0x003C and 0x003B for 48-bit match determination.

Please refer to the description for MTTTM bit 6 in the “Hardware Interrupt Registers” on page 44.

15.14. SMT Bus Monitor Interrupt Registers and Their Use

Section 11.7 on page 39 through Section 11.9 describe how the host uses three Hardware Interrupt registers, the Interrupt Log Buffer and the Interrupt Count & Log Address Register to manage interrupts. When the SMT is enabled, three additional registers are dedicated to SMT interrupts. Comparable to the Hardware Interrupt register triplet, the SMT has

- An “SMT Bus Monitor Interrupt Enable Register (0x0011)” to enable and disable interrupts
- An “SMT Bus Monitor Pending Interrupt Register (0x0008)” to capture the occurrence of enabled interrupts
- An “SMT Bus Monitor Interrupt Output Enable Register (0x0015)” to enable $\overline{\text{INT}}$ output to host, for pending enabled interrupts

Each individual bit in all three registers is mapped to the same interrupt-causing event when the corresponding interrupt condition is enabled. Numerous interrupt options are available for the SMT. At initialization, bits are set in the “SMT Bus Monitor Interrupt Enable Register (0x0011)” to identify the interrupt-causing events for the SMT which are heeded by the IP core. Most SMT applications only use a subset of available SMT interrupt options. Interrupt-causing events are ignored when their corresponding bits are reset in the “SMT Bus Monitor Interrupt Enable Register (0x0011)” . Setting an Interrupt Enable Register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

Whenever an SMT interrupt event occurs (and the corresponding bit is already set in the “SMT Bus Monitor Interrupt Enable Register (0x0011)”), these actions occur:

- The Interrupt Log Buffer is updated.
- A bit corresponding to the interrupt type is set in the “SMT Bus Monitor Pending Interrupt Register (0x0008)”. The type bit is logically-ORed with the preexisting register value, retaining bits for prior, unserviced SMT interrupts.
- MT Interrupt Pending (MTIP) bit 1 is set in the “Hardware Pending Interrupt Register (0x0006)”. The MTIP bit is logically-ORed with the preexisting register value, retaining bits for unserviced hardware interrupts and the pre-existing status of the BCIP and RTIP (Bus Controller and RT) interrupt pending bits.
- If the matching bit is already set in the “SMT Bus Monitor Interrupt Output Enable Register (0x0015)”, an $\overline{\text{INT}}$ output occurs.

If the matching bit in the “SMT Bus Monitor Interrupt Output Enable Register (0x0015)” was not already set (i.e., low priority polled interrupt), the host can poll the “SMT Bus Monitor Pending Interrupt Register (0x0008)” to detect the occurrence of SMT interrupts, indicated by non-zero value. Reading the “SMT Bus Monitor Pending Interrupt Register (0x0008)” automatically clears it to 0x0000.

A single $\overline{\text{INT}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the four Interrupt Output Enable registers (hardware, BC, RT and SMT). Multiple interrupt-causing events can occur simultaneously, so single or simultaneous interrupt events can assert the $\overline{\text{INT}}$ host interrupt output.

When the host receives an $\overline{\text{INT}}$ signal from the IP core, it identifies the event(s) that triggered the interrupt. Section 11.7 describes two methods for identifying the interrupt source(s). One scheme uses the three low order bits in the “Hardware Pending Interrupt Register (0x0006)” to indicate when BC, RT and SMT interrupts occur. When MT Interrupt Pending (MTIP) bit 1 is set in the “Hardware Pending Interrupt Register (0x0006)”, the “SMT Bus Monitor Pending Interrupt Register (0x0008)” contains a nonzero value and may be read next to identify the specific SMT interrupt event(s). Or, the host can directly interrogate the Interrupt Count & Log Address Register, followed by the Interrupt Log Buffer. Section “11.7. Hardware Interrupt Behavior” has a detailed description.

15.14.1. SMT Bus Monitor Interrupt Enable Register (0x0011)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	CBUFRO	DBUFRO	CBUFMAT	DBUFMAT	MTMERR	MTEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

15.14.2. SMT Bus Monitor Pending Interrupt Register (0x0008)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	CBUFRO	DBUFRO	CBUFMAT	DBUFMAT	MTMERR	MTEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

15.14.3. SMT Bus Monitor Interrupt Output Enable Register (0x0015)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	CBUFRO	DBUFRO	CBUFMAT	DBUFMAT	MTMERR	MTEOM	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Three registers govern SMT interrupt behavior: the SMT Interrupt Enable Register, the SMT Pending Interrupt Register and the SMT Interrupt Output Enable Register. When a bit is set in the SMT Interrupt Enable Register, the corresponding SMT interrupt is enabled. When a bit is reset in this register, the corresponding interrupt event is unconditionally disregarded. Setting a register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

When an enabled SMT interrupt event occurs, the corresponding bit is set in the SMT Pending Interrupt Register and the Interrupt Log Buffer is updated. To simplify interrupt decoding, MTIP bit 1 in the "Hardware Pending Interrupt Register (0x0006)" is also set whenever one or more bits are set in the SMT Pending Interrupt Register.

If the corresponding bit is already set in the SMT Interrupt Output Enable Register, the $\overline{\text{INT}}$ output signal is asserted at Pending Interrupt Register assertion. The SMT Interrupt Output Enable Register establishes two priority levels: high priority interrupts generate an $\overline{\text{INT}}$ output while low priority interrupts do not. Both priority levels update the SMT Pending Interrupt Register and the Interrupt Log Buffer. The host detects low priority (masked) interrupts by polling SMT Pending Interrupt Register.

The table below describes the bit descriptions shared by all three SMT interrupt registers.

Bit No.	*Mnemonic*	*Function*
15 − 9	Reserved	These bits are not used in SMT monitor mode. They should be initialized logic 0 in the “SMT Bus Monitor Interrupt Enable Register (0x0011)”. These bits will always read logic 0 in the “SMT Bus Monitor Pending Interrupt Register (0x0008)”.
8	CBUFRO	Command Buffer Rollover Interrupt. The Command Buffer Pointer value (Word 1 in the SMT Address List) has rolled over to the Command Buffer Start Address (Word 0 in the SMT Address List).
7	DBUFRO	Data Buffer Rollover Interrupt. The Data Buffer Pointer value (Word 5 in the SMT Address List) has rolled over to the Data Buffer Start Address (Word 4 in the SMT Address List).
6	CBUFMAT	Command Buffer Address Match Interrupt. The Command Buffer Pointer value (Word 1 in the SMT Address List) has reached the Command Buffer Address Match value in Word 3 of the SMT Address List.
5	DBUFMAT	Data Buffer Address Match Interrupt. The Data Buffer Pointer value (Word 5 in the SMT Address List) has reached the Data Buffer Address Match value in Word 7 of the SMT Address List.
4	SMTMERR	SMT Message Error Interrupt. A non-broadcast MIL-STD-1553 message ended with an RT Status Word containing the ME Message Error status bit set.
3	SMTEOM	SMT End of Message Interrupt. Successful completion of a MIL-STD-1553 message, regardless of validity.
2 − 0	Reserved	Bits 2-0 cannot be written, and read back 000.

16. REMOTE TERMINAL − OVERVIEW

The HI-6300 can operate as a MIL-STD-1553 Remote Terminal, requiring minimal host support. When the Remote Terminal (RT) is enabled, its configuration and operation is nearly identical to the Holt HI-6135 integrated circuit. The Remote Terminal has its own Descriptor Table, command Illegalization Table and host interrupt configuration.

The following signal signals are provided for the Remote Terminal:

- RT Terminal Address 4 - 0 input signals
- RT Terminal Address Parity input signal
- RT Subsystem Fail input signals
- RT Mode Code 8 (Reset Remote Terminal) output signals

By writing the “Master Status and Reset Register (0x0001)” , the Remote Terminal can be independently reset using “soft reset”. The RT can be configured to automatically assert soft reset when a valid “Reset Remote Terminal” mode code command is received.

In this section of the data sheet, the Remote Terminal registers are described first, followed by the details for configuring and operating the RT.

17. REMOTE TERMINAL REGISTERS

17.1. Remote Terminal Configuration Register (0x0017)

RTTO1	RTTO0	RTINHA	RTINHB	BCSTINV	DBCENA	UMCINV	NOTICE2	SMCP	TRXDB	ALTBITW	AUTOBSD	MC17OP1	MC17OP0	MC16OPT	MC8OPT	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th><i>Function</i></th></tr>
<tr><td>15 − 14</td><td>RTTO1:0</td><td>R/W</td><td>0</td><td>RT-RT Time Out Select.
This 2-bit field selects the “no response” time-out delay for RT-to-RT receive commands from four available selections:
<table>
<tr><th><i>Bit 15:14</i></th><th><i>Bus Dead Time</i></th><th><i>RT-RT Time Out</i></th></tr>
<tr><td>00</td><td>15μs</td><td>57μs</td></tr>
<tr><td>01</td><td>20μs</td><td>62μs</td></tr>
<tr><td>10</td><td>58μs</td><td>100μs</td></tr>
<tr><td>11</td><td>138μs</td><td>180μs</td></tr>
</table>
For RT-RT commands, time out delay is measured per Figure 8 in the RT Validation Test Plan, SAE AS4111. That is, from mid-parity of the receive command to mid-sync of the first received data word. This interval includes 20μs each for the embedded transmit command word and transmit-RT status word within this span.</td></tr>
<tr><td>13</td><td>RTINHA</td><td>R/W</td><td>0</td><td>RT Bus A Inhibit.
If this bit is logic 1, Bus A is inhibited, as defined by the BSDTXO bit in “Master Configuration Register 1 (0x0000)”. The BSDTXO bit offers two options: inhibit transmit and receive, or inhibit only transmit.
Note: If this bit is logic 0, Bus A is not inhibited here but its operation may otherwise be globally inhibited by logic 1 at the TXINHA signal, or logic 1 at the TXINHA bit in the “Master Status and Reset Register (0x0001)”.</td></tr>
<tr><td>12</td><td>RTINHB</td><td>R/W</td><td>0</td><td>RT Bus B Inhibit.
If this bit is logic 1, Bus B is inhibited, as defined by the BSDTXO bit in “Master Configuration Register 1 (0x0000)”. The BSDTXO bit offers two options: inhibit transmit and receive, or inhibit only transmit.
Note: If this bit is logic 0, Bus B is not inhibited here but its operation may otherwise be globally inhibited by logic 1 at the TXINHB signal, or logic 1 at the TXINHB bit in the “Master Status and Reset Register (0x0001)”.</td></tr>
<tr><td>11</td><td>BCSTINV</td><td>R/W</td><td>0</td><td>Broadcast Commands Invalid.
If this bit is high, commands addressed to RT address 31 are treated as invalid: There is no terminal recognition of commands to RT address 31; there is no RT command response, and no status updating for the benefit of following “transmit status” or “transmit last command” mode commands. If this bit is low, commands addressed to RT address 31 are treated as valid broadcast commands.</td></tr>
</table>

Bit No.	Mnemonic	R/W	Reset	Function
10	DBCENA	R/W	0	Dynamic Bus Control Enable. When this bit is logic 1, the RT accepts bus control after receiving a valid legal “dynamic bus control” mode command. After automatically responding an RT Status Word with DBCA status bit 1 set, the RTSTEX bit in the “Master Configuration Register 1 (0x0000)” resets to 0 and the BCSTRT bit in the “Master Configuration Register 1 (0x0000)” is set to logic 1. If automatic dynamic bus control is allowed, the BCENA register bit in the “Master Configuration Register 1 (0x0000)” must be logic 1, and the bus controller should be fully initialized in advance to permit immediate operation. To assert host control over BC switch-in timing, keep the BCENA register bit in the “Master Configuration Register 1 (0x0000)” reset to 0; this option’s automatic BCSTART bit-set then does nothing. When ready to commence BC operation, the host first sets the BCENA register bit, then the BCSTART bit in the “Master Configuration Register 1 (0x0000)” to start BC. When this bit is logic 0, the host must perform all actions necessary to change from RT to Bus Controller mode. Typically the RT Descriptor Table is configured to generate an IWA interrupt to alert the host upon receiving a valid legal Dynamic Bus Control mode code command, MC0.
9	UMCINV	R/W	0	Undefined Mode Codes Invalid. This bit determines whether the RT treats undefined mode code commands as valid (default) or invalid commands. This bit applies only to the following undefined mode code commands: • Mode Codes 0 through 15 with T/R bit = 0 • Mode Codes16, 18 and19 with T/R bit = 0 • Mode Codes 17, 20 and 21 with T/R bit = 1 If this bit is low (default state after $\overline{\text{RESET}}$ signal reset) undefined mode code commands are considered valid, and RT response is based on individual mode command settings in the Illegalization Table: If mode command is legal, the RT “responds in form” and updates status. If a mode command is illegal, the RT asserts Message Error status and (if non-broadcast) transmits only its Status Word without associated data word. If this bit is high, undefined mode code commands are treated as invalid: There is no RT recognition of an invalid command, no RT command response, and no status updating for the benefit of following “transmit status” or “transmit last command” mode commands.
8	NOTICE2	R/W	0	Notice 2 Broadcast Data Storage. If this bit is high, the terminal stores data associated with broadcast commands separately from data associated with non-broadcast commands to meet the requirements of MIL-STD-1553B Notice 2. If this bit is low, broadcast command data is stored in the same buffer with data from non-broadcast commands.
7	SMCP	R/W	0	Simplified Mode Command Processing. When this bit is asserted, the remote terminal applies “Simplified Mode Command Processing” for all valid mode code commands, as described in Section 20.5 on page 191.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
6	TRXDB	R/W	0	Temporary Receive Data Buffer. When this bit is asserted, the remote terminal enables a temporary receive data buffer used during receive commands. See Section 18.3 on page 145. When this bit is asserted, the RT stores received data words in a 32-word data buffer during message processing. Only after error-free message completion, are the buffered words written into the data buffer memory assigned to the specific subaddress in the RT Descriptor Table. This bit should only be modified when RTSTEX bit is low in “Master Configuration Register 1 (0x0000)” (see Section 11.1 on page 30). Changing the TRXDB bit when the RTSTEX configuration bit is logic-1 causes unpredictable results.
5	ALTBITW	R/W	0	Alternate BIT Word Enable. When this bit is logic 0, the remote terminal responds to a “transmit BIT word” mode command (MC19) by sending the word stored in its Built-In Test Word register. The Built-In Test Word register resides at address 0x001E. When this bit is logic 1, the remote terminal responds to a “transmit BIT word” mode command (MC19) by sending the word stored in its **Alternate** Built-In Test Word register. The Alternate Built-In Test Word register resides at address 0x001F. Using an Alternate Built-In Test Word register allows the user to fully define the BIT word, while the default Built-In Test Word register locations contain several predefined, IP core-controlled status bits.

Bit No.	Mnemonic	R/W	Reset	Function
4	AUTOBSD	R/W	0	Automatic Bus Shutdown Enable. ***Note:** Mode commands MC4 and MC5 are not affected by the AUTOBSD bit, but are included in this description to present a complete picture of IP Core response to bus shutdown mode commands.* The Bus Controller exercises “shutdown“ control over Remote Terminal connections to the inactive MIL-STD-1553 bus using the “transmitter shutdown” (MC4) or “selected transmitter shutdown” MC20 (decimal) mode code commands. These apply only to the inactive bus. The RT cannot shutdown the bus where the command is received. When the inactive bus transmitter is shutdown, the IP Core inhibits further transmission on that bus for the affected RT. Once shutdown, the transmitter can be reactivated by (a) an “override transmitter shutdown” (MC5) mode command, (b) an “override selected transmitter shutdown” (MC21) mode command, (c) a “reset remote terminal” (MC8) mode command, (d) asserting hardware $\overline{\text{RESET}}$ Master Reset input signal or (e) software reset initiated by setting the RTRESET bit in the “Master Status and Reset Register (0x0001)”. When the AUTOBSD bit is reset, the IP Core automatically performs bus shutdown and shutdown override in response to mode commands. When the AUTOBSD bit is set, the IP Core only transmits status; the host must perform bus shutdown and override duties by asserting control of the TXINHA and TXINHB bits in the “Master Configuration Register 1 (0x0000)”, or by controlling the input signals with the same function. Mode commands MC4 (”transmitter shutdown”) and MC5 (”override transmitter shutdown”) have unconditional shutdown or override response. When MC4 is received, the terminal fulfills shutdown for the inactive bus, disabling the transmitter and receiver, or transmitter only, depending on the state of the BSDTXO bit in “Master Configuration Register 1 (0x0000)”. The IP Core affirms shutdown status by setting the corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. When mode command MC5 is received, inactive bus transmit and receive is automatically reenabled by the IP Core; “shutdown override” status is affirmed by resetting the corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.

<table>
<tr><th>*Bit No.*</th><th>*Mnemonic*</th><th>*R/W*</th><th>*Reset*</th><th colspan="4">*Function*</th></tr>
<tr><td rowspan="3">4</td><td rowspan="3">AUTOBSD (continued)</td><td rowspan="3">R/W</td><td rowspan="3">0</td><td colspan="4">The “selected transmitter shutdown” (MC20) and “override selected transmitter shutdown” (MC21) mode commands act similarly to MC4 and MC5 respectively, except bus shutdown (or shutdown override) is conditional, based on the value of a mode data word received with the command. To act on a given bus, the received mode data word must match a predetermined “bus select” value. Bus shutdown (or shutdown override) can only act on the inactive bus, and only when the received mode data word matches the “bus select” value for that bus. When a MC20 mode data word matches the “bus select” value for the inactive bus, the terminal fulfills shutdown for the inactive bus, disabling the transmitter and receiver, or transmitter only, depending on the state of the BSDTXO bit in “Master Configuration Register 1 (0x0000)”. The IP Core affirms shutdown status by setting the corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. When a MC21 mode data word matches the “bus select” value for the inactive bus, the terminal fulfills shutdown override for the inactive bus, enabling the transmitter (and receiver, if BSDTXO bit in “Master Configuration Register 1 (0x0000)” is logic 0). The IP Core affirms override status by resetting the corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.
When the AUTOBSD bit equals zero, unique “bus select” values should be initialized by the host in the “Remote Terminal Bus A Select Register (0x001C)” and “Remote Terminal Bus B Select Register (0x001D)” for fulfillment of “selected transmitter” shutdown and override mode commands. When AUTOBSD equals zero, transmitter shutdown (or shutdown override) automatically occurs when the received mode data value matches the inactive bus “Bus Select” register.
Below shows IP Core response for “transmitter shutdown” and “override transmitter shutdown” mode code commands for different configuration options:</td></tr>
<tr><td colspan="4">**The AUTOBSD bit in RT Configuration Register 0x0017 is logic 0 or 1**</td></tr>
<tr><td>**MC4 (or MC5)** unconditional fulfillment</td><td>Inactive Bus Tx & Rx Disabled (Enabled). *(only Tx is disabled, if the* BSDTXO *config. bit = 1)*</td><td>Status Word transmitted, unless broadcast</td><td>In BIT Word Register, TXSD & RXSD bits updated. *(only TXSD bit updated, if the* BSDTXO *config. bit = 1)*</td></tr>
</table>

<table>
<tr><th>*Bit No.*</th><th>*Mnemonic*</th><th>*R/W*</th><th>*Reset*</th><th colspan="4">*Function*</th></tr>
<tr><td rowspan="6">4</td><td rowspan="6">AUTOBSD
(continued)</td><td rowspan="6">R/W</td><td rowspan="6">0</td><td colspan="4">**The AUTOBSD bit in RT Configuration Register 0x0017 is logic 0**</td></tr>
<tr><td>**MC20 (or MC21)** if mode data value matches “Bus Select” value</td><td>Inactive Bus Tx & Rx Disabled (Enabled). *(only Tx is disabled, if the* BSDTXO *config. bit = 1)*</td><td>Status Word transmitted, unless broadcast</td><td>In BIT Word Register, TXSD & RXSD bits updated. *(only TXSD bit updated, if the* BSDTXO *config. bit = 1)*</td></tr>
<tr><td>**MC20 (or MC21)** if mode data does NOT match “Bus Select” value</td><td>Inactive Bus Tx & Rx status not changed</td><td>Status Word transmitted, unless broadcast</td><td>In BIT Word Register, TXSD & RXSD bits are static</td></tr>
<tr><td colspan="4">**The AUTOBSD bit in RT Configuration Register 0x0017 is logic 1**</td></tr>
<tr><td>**MC20 (or MC21)** if mode data value matches “Bus Select” value</td><td>Inactive Bus Tx & Rx status NOT changed *(Host can modify RTINH bits 13,12 in RT Config. Reg. 0x0017)*</td><td>Status Word transmitted, unless broadcast</td><td>In BIT Word Register, TXSD & RXSD bits are static</td></tr>
<tr><td>**MC20 (or MC21)** if mode data does NOT match “Bus Select” value</td><td>Inactive Bus Tx & Rx status not changed</td><td>Status Word transmitted, unless broadcast</td><td>In BIT Word Register, TXSD & RXSD bits are static</td></tr>
<tr><td>3 − 2</td><td>MC17OP1:0</td><td>R/W</td><td>0</td><td colspan="4">MC17 Sync Option Bits 1:0
If register bits 3-2 equal 11, the data word received with a valid “synchronize” mode command (MC17) is unconditionally loaded into the “Remote Terminal Time Tag Counter Register (0x0049)”. For non-broadcast MC17 commands, the counter load occurs before status word transmission. If register bits 3-2 equal 00, the external host assumes responsibility for actions needed to perform “synchronize” duties upon reception of the valid MC17 “synchronize” mode code command, but status transmission automatically occurs.
The binary 01 and 10 combinations of register bits 3-2 support certain extended subaddress schemes. If bits 3-2 equal 01, the received data word is automatically loaded into the “Remote Terminal Time Tag Counter Register (0x0049)” if bit 0 of the received data word equals 0. If bits 3-2 equal 10, the received data word is automatically loaded into the Time-Tag counter if bit 0 of the received data word equals 1. For non-broadcast MC17 commands, the counter load occurs before status word transmission.</td></tr>
<tr><td>1</td><td>MC16OPT</td><td>R/W</td><td>0</td><td colspan="4">Host reset of “service request” status bit for mode code 16.
If this bit is logic 0, reception of a “transmit vector word” mode command (MC16) causes automatic reset of the Service Request status bit. The Service Request bit is reset in the Status Word Bits register before status word transmission begins. If this bit is logic 1, the external host assumes responsibility for resetting the Service Request bit in the Status Word Bits register.</td></tr>
</table>

Bit No.	Mnemonic	R/W	Reset	Function
0	MC8OPT	R/W	0	Automatic soft reset for mode code 8. If this bit is logic 0, reception of a “reset remote terminal” mode command (MC8) causes automatic assertion of SRESET software reset. If non-broadcast mode command, reset occurs after status word transmission is complete. If this bit is logic 1, the external host assumes responsibility for actions needed to perform terminal reset.

17.2. Remote Terminal Operational Status Register (0x0018)

RTADR4	RTADR3	RTADR2	RTADR1	RTADR0	RTADP	LOCK	Reserved	Reserved	Reserved	MCND	MCRD	MCTD	RTAPF	Reserved	Reserved	
These bits latch signals							0	0	0	0	0	0	0	0	0	Reset
RW (see LOCK bit 9)							R									Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

At rising edge on the $\overline{\text{RESET}}$ Master Reset input signal, register bits 15-9 capture the logic states (0 or 1) of the corresponding input signals having like names (if applicable). After reset, register bits 15-9 can be overwritten only if LOCK bit 9 is logic 0. If the register LOCK bit is logic 1, these bits are read-only.

Bits 8-0 are read-only; these bits are cleared after $\overline{\text{RESET}}$ signal master reset, but are unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

Bit No.	Mnemonic	R/W	Reset	Function
15 − 11 10	RTADR[4:0] RTADP	R/W	0	Remote Terminal Address bits 4-0. Remote Terminal Address Parity. These bits reflect the state of the input signals RTADR4 through RTADR0 that applied at the rising edge of the $\overline{\text{RESET}}$ master reset input signal (i.e. the active remote terminal address). The RTADP bit, when appended to the remote terminal address bits, provides odd parity. If the register LOCK bit is high, bits 15-10 are read-only. If the register LOCK bit is low, the host can overwrite these bits and change the terminal address and parity.
9	LOCK	R/W	0	Remote Terminal Address Lock. After reset, the host can overwrite bits 15-9 **only** if register LOCK bit 9 is logic 0. Refer to “Table 3. IP Core Configuration Signals” and “Table 4. RT_AD_LAT and RTA_SW_EN Truth Table” on page 18 for further details on software programming of RT Address. When the LOCK bit is high, the host cannot overwrite register bits 15-9. To restore host write capability for these bits, the $\overline{\text{RESET}}$ master reset input signal must first be asserted to restore register LOCK bit 9 to logic 0.
8 − 6	Reserved	R	0	These bits are not used.

<table>
<tr><th><em>Bit No.</em></th><th><em>Mnemonic</em></th><th><em>R/W</em></th><th><em>Reset</em></th><th colspan="3"><em>Function</em></th></tr>
<tr><td rowspan="7">5
4
3</td><td rowspan="7">MCND
MCRD
MCTD</td><td rowspan="7">R</td><td rowspan="7">0</td><td colspan="3">No Data Mode Command Flag.
Receive Data Mode Command Flag.
Transmit Data Mode Command Flag.
These three bits reflect the type of command stored in the RT Current Command Register:</td></tr>
<tr><td><strong><em>Current Command Type</em></strong></td><td><strong><em>Bits 5-4-3</em></strong></td><td><strong><em>Current Command Word</em></strong></td></tr>
<tr><td>Subaddress, not mode code</td><td>000</td><td>Subaddress, transmit or receive</td></tr>
<tr><td>Mode code, no data word</td><td>100</td><td>MC0 to MC15, $T/\overline{R}$ bit = 1</td></tr>
<tr><td>Mode code, received word</td><td>010</td><td>MC16 to MC31, $T/\overline{R}$ bit = 0</td></tr>
<tr><td>Mode code, transmit word</td><td>001</td><td>MC16 to MC31, $T/\overline{R}$ bit = 1</td></tr>
<tr><td>Mode code, undefined</td><td>111</td><td>MC0 to MC15, $T/\overline{R}$ bit = 0</td></tr>
<tr><td>2</td><td>RTAPF</td><td>R</td><td>0</td><td colspan="3">Remote Terminal Address Parity Fail.
This bit is set when RT address parity error occurs for the value contained in register bits 15-10. It is low when correct odd parity applies for bits 15-10.</td></tr>
<tr><td>1 − 0</td><td>Reserved</td><td>R</td><td>0</td><td colspan="3">These bits are not used.</td></tr>
</table>

17.3. Remote Terminal Current Command Register (0x0002)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is read-only and is fully maintained by the IP core. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

This register contains the last valid command word received by the Remote Terminal over either MIL-STD-1553 bus. This register is updated 5µs after the ACTIVE output is asserted.

17.4. Remote Terminal Current Control Word Address Register (0x0003)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is read-only and is fully maintained by the IP core. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

This register contains the address for the descriptor table Control Word corresponding to the current command stored in the Current Command Register, above. This register is updated 5 µs after the ACTIVE output is asserted for recognition of a valid command for the RT. Also see description for the “Current Message Information Word” register.

17.5. Remote Terminal Descriptor Table Base Address Register (0x0019)

MSB							Register Value								LSB	
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	Reset
R	RW						R									Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and contains the starting address for the Remote Terminal’s Descriptor Table. This register is initialized with default values after $\overline{\text{RESET}}$ signal master reset, or by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”. The post-reset register value is 0x0400. After initialization, this register is fully maintained by the host. Bit 15 and bits 8:0 cannot be set and will always read logic 0.

17.6. Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)

TXANDCLR	Reserved	Reserved	Reserved	Reserved	ME	INST	SVCREQ	Reserved	Reserved	Reserved	BCR	BUSY	SSYSF	DBCA	TF	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW	R					RW		R				RW		RW	RW	Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write. With the exception of bits 4 and 10, this register is maintained by the host. This register is cleared after $\overline{\text{RESET}}$ signal master reset, or by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

Register bits 14-10 and 7-4 are read-only. Most of these bits read back zero, except for bits 4 and 10, which are maintained by the IP core. The remaining bits in the register are Read-Write and are maintained by the host. All bits are active high. Register bits 10-0 are reflected in the outgoing MIL-STD-1553 RT status word. The RT status word reflects the state of host-written register bits until overwritten by the host, unless the Transmit and Clear function (bit 15) is enabled. When set, the Transmit and Clear bit resets itself and bits 9-5 and 3-0 after the next transmitted status word.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	TXANDCLR	R/W	0	Transmit and Clear. When this bit is set, it resets itself and bits 9-5 and 3-0 after the next transmitted status word. This bit does not affect operation of the Transmit Status Word and Transmit Last Command mode codes. Example: Transaction of a valid legal command with the INST and TXANDCLR bits asserted results in status word transmission with the Instrumentation bit set. If the following command is Transmit Status or Transmit Last Command mode code, the Instrumentation bit remains set.
14 − 11	Reserved	R	0	These bits are not used, cannot be written, always read back 0000.
10	ME	R	0	Message Error status bit. The IP core maintains this read-only bit, based on prior message results.
9	INST	R/W	0	Instrumentation status bit. The host maintains this read-write bit.
8	SVCREQ	R/W	0	Service Request status bit. The host maintains this read-write bit. If the RT’s Configuration Register MC16OPT bit equals 0, reception of a “transmit vector word” mode command (MC16) causes automatic reset of the SVCREQ status bit in this register; when this occurs, the register bit is reset before status word transmission begins.
7 − 5	Reserved	R	0	These bits are not used, cannot be written, always read back 000.
4	BCR	R	0	Broadcast Command Received status bit. The IP core maintains this read-only bit, based on prior message results.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
3	BUSY	R/W	0	Busy status bit. The host maintains this read-write bit. When set, the RT asserts its Busy bit in status response for all valid commands. Instead of enabling Busy for all commands, the host can assert Busy status for selected commands by asserting the Busy bit in descriptor table Control Words for the individual commands. When response to a command conveys Busy status, the RT suppresses transmission of data words that would normally accompany status for transmit commands. For messages transacted with Busy status, the WASBSY flag is asserted in the stored Message Information Word. If INTBUSY bit 2 is set in the “Extended Configuration Register (0x004D)”, the WASBUSY bit 9 is also enabled in the RT Interrupt Information Word. **Note:** Busy status alone is not an interrupt event. See “Extended Configuration Register (0x004D)” on page 46.
2	SSYSF	R/W	0	Subsystem Fail status bit. The host maintains this read-write bit. This register bit is logically ORed with the SSFLAG input signal. If either SSYSF register bit or SSFLAG input signal is asserted, the SSYSF Subsystem Flag status bit is set.
1	DBCA	R/W	0	Dynamic Bus Control Acceptance status bit. The host maintains this read-write bit. If the terminal is to acknowledge a Dynamic Bus Control, Mode Code 0 command, the host should set this bit to a “1”.
0	TF	R/W	0	Terminal Flag status bit. The host maintains this read-write bit. When this bit is asserted, the Terminal Flag status bit is set. If the Terminal Flag bit is set while responding to subaddress transmit commands or mode code commands 16-31 that normally transmit a data word, all data word transmission is suppressed.

17.7. Remote Terminal Current Message Information Word Register (0x001B)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Only and is fully maintained by the IP core. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”. This register contains the data buffer address (assigned in the terminal’s Descriptor Table) corresponding to the last decoded valid command’ for the Remote Terminal. This register is updated 5µs after the ACTIVE output is asserted.

The value in this register points to the command’s Message Information Word (or MIW) in the Descriptor Table. The value of the current command word itself is stored in the Current Command Register.

17.8. Remote Terminal Bus A Select Register (0x001C)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the IP core. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

The Bus A Select register is only used when the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 0. This AUTOBSD setting means the IP core automatically fulfills mode commands MC20 (decimal) “selected transmitter shutdown” or MC21 “override selected transmitter shutdown”. “Transmitter shutdown” or “shutdown override” can only occur for the inactive bus. If either mode command is received on Bus B, the inactive bus is Bus A. The IP core compares the received mode data word to the contents of the Bus A Select register to determine whether inactive Bus A is selected for “transmitter shutdown” or “transmitter shutdown override”. (Bus shutdown or shutdown override can only occur for the inactive bus.) If the data word matches the value stored in the Bus A Select register and AUTOBSD equals 0, the IP core automatically fulfills MC20 “transmitter shutdown” or MC21 “shutdown override” without host assistance: If the mode command received was MC20 (bus shutdown), the Transmit Shutdown A bit in the RT’s BIT (built-in test) Word Register is asserted. If mode command MC21 (override bus shutdown) was received, the Transmit Shutdown A bit in the BIT Word Register is negated.

17.9. Remote Terminal Bus B Select Register (0x001D)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the IP core. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is unaffected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

The Bus B Select register is only used when the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 0. This AUTOBSD setting means the IP core automatically fulfills mode commands MC20 (decimal) “selected transmitter shutdown” or MC21 “override selected transmitter shutdown”. “Transmitter shutdown” or “shutdown override” can only occur for the inactive bus. If either mode command is received on Bus A, the inactive bus is Bus B. The IP core compares the received mode data word to the contents of the Bus B Select register to determine whether inactive Bus B is selected for “transmitter shutdown” or “transmitter shutdown override”. (Bus shutdown or shutdown override can only occur for the inactive bus.) If the data word matches the value stored in the Bus B Select register and AUTOBSD equals 0, the IP core automatically fulfills MC20 “transmitter shutdown” or MC21 “shutdown override” without host assistance: If the mode command received was MC20 (bus shutdown), the Transmit Shutdown B bit in the RT’s BIT (built-in test) Word Register is asserted. If mode command MC21 (override bus shutdown) was received, the Transmit Shutdown B bit in the BIT Word Register is negated.

17.10. Remote Terminal Built-In Test (BIT) Word Register (0x001E)

TXASD	TXBSD	RXASD	RXBSD	User Assigned Bits						LBFA	LBFB	BMTF	RT	Reserved	TFBINH	
P	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R				RW								R				Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Bits 11-4 in this 16-bit register is read-write; the remaining bits are read-only. The ten assigned bits are written by the IP core when predetermined events occur. The host may overwrite the IP core-written bits 5 and 4. After RESET signal master reset, bits 13-12, 5-4 and 0 are reset. Bits 15-14 will be set if the corresponding TXINHA or TXINHB input signals are high. Bits 3-1 will be set if RT address parity error or post-RESET memory test failure occurred. These registers are not affected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”.

If the ALTBITW option bit in the “Remote Terminal Configuration Register (0x0017)” is zero when a valid “transmit BIT word” mode command (MC19) is received, the current value in this register is transmitted as the mode data word in the terminal response. The value is also copied to the Remote Terminal’s assigned data buffer for MC19, after mode command fulfillment.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15 14	TXASD TXBSD	R	0	Transmit Bus A Shutdown. Transmit Bus B Shutdown. These read-only bits are set when the corresponding bus transmitter was disabled by assertion of the bus TXINHA or TXINHB input signal, or by fulfillment of a “transmitter shutdown” mode command, either MC4 or MC20. Refer to the description for the BSDTXO bit in the “Master Configuration Register 1 (0x0000)” and the description for the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” for further information.
13 12	RXASD RXBSD	R	0	Receive Bus A Shutdown. Receive Bus B Shutdown. These read-only bits are set when the corresponding bus receiver was disabled concurrently with a bus transmitter by a “transmitter shutdown” mode command MC4 or MC20. Refer to the description for the BSDTXO bit in the “Master Configuration Register 1 (0x0000)” and the description for the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” for further information.
11 − 6	----------	R/W	0	User assigned bits.
5 4	LBFA LBFB	R/W	0	Bus A Loopback Fail. Bus B Loopback Fail. These bits are set if Bus A or Bus B loopback failure occurs during self-test (see Section “22.1.1. Self-Test Control Register (0x0028)” on page 195).

Bit No.	Mnemonic	R/W	Reset	Function
3	BMTF	R	0	BIST Memory Test Fail. This bit is set if error occurs during built-in self-test for IP core Random Access Memory (RAM) (see Section “22.1.1. Self-Test Control Register (0x0028)” on page 195).
2	RTAPF	R	0	RT Address Parity Fail. This bit is asserted when “Remote Terminal Operational Status Register (0x0018)” bits 15:10 reflect parity error. After $\overline{\text{RESET}}$ master reset, bits 15:10 in the RT’s Operational Status Register reflect input signal states.
1	Reserved	R	0	These bits are not used.
0	TFBINH	R	0	This bit is set when the Terminal Flag status bit is disabled while fulfilling an “inhibit terminal flag bit” mode code command (MC6). This bit is reset if terminal flag status bit disablement is cancelled later by an “override inhibit terminal flag bit” mode code command (MC7).

17.11. Remote Terminal Alternate Built-In Test (BIT) Word Register (0x001F)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the host. This register is cleared after $\overline{\text{RESET}}$ signal master reset. It is not affected by assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”. If the ALTBITW option bit in the “Remote Terminal Configuration Register (0x0017)” equals one when a valid “transmit BIT word” mode command (MC19) is received, the current value in this register is transmitted as the mode data word in the terminal response. The value is also copied to the assigned data buffer for MC19, after mode command fulfillment.

17.12. Remote Terminal Time Tag Counter Register (0x0049)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This register is read-only and is cleared after $\overline{\text{RESET}}$ signal Master Reset or assertion of RTRESET remote terminal software reset in the “Master Status and Reset Register (0x0001)”. Reads to this register address return the current value of the free running 16-bit Time Tag counter. Counter resolution is programmed by the TTCK2:0 bits in the “Time Tag Counter Configuration Register (0x0039)”. Options are: 2, 4, 8, 16, 32 and 64µs, or externally provided clock. The same clock source is shared by the RT and BC.

The IP core automatically resets the Time-Tag Counter when a “synchronize” mode command without data (MC1) is received. In addition, the host can reset, load or capture the Time Tag count at any time by asserting action bits in the “Time Tag Counter Configuration Register (0x0039)”. Load and capture operations utilize the “Remote Terminal Time Tag Utility Register (0x004A)”, described below.

The MC17OP1:0 bits in the "Remote Terminal Configuration Register (0x0017)" allows automatic loading of Time-Tag count using the data word received with a "synchronize with data" mode command, MC17. If both of these bits equal one, the data word received with a valid "synchronize" mode command (MC17) is unconditionally loaded into the Time-Tag counter. For non-broadcast MC17 commands, the counter load occurs before status word transmission. If both MC17OP1 and MC17OP0 bits equal 0, the external host assumes responsibility for actions needed to perform "synchronize" duties upon reception of the valid MC17 "synchronize" command, but status transmission occurs automatically.

The binary 01 and 10 combinations of these bits support certain extended subaddressing schemes. If the MC17OP1:0 bits equal 01, the received data word is automatically loaded into the Time-Tag counter if the low order bit of the received data word (bit 0) equals 0. If the MC17OP1:0 bits equal 10, the received data word is automatically loaded into the Time-Tag counter if the low order bit of the received data word (bit 0) equals 1. For non-broadcast MC17 commands, the counter is loaded before status word transmission.

17.13. Remote Terminal Time Tag Utility Register (0x004A)

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This 16-bit register is Read-Write and is fully maintained by the host. This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is not affected by assertion of RTRESET remote terminal software reset in the "Master Status and Reset Register (0x0001)". This register has two functions associated with the two free-running Remote Terminal Time Tag Counters:

17.13.1. RT Time Tag Counter Loading

When the RTTTA1-0 bits 9-8 in "Time Tag Counter Configuration Register (0x0039)" are written to 1-0, the value contained in the "Remote Terminal Time Tag Utility Register (0x004A)" is loaded into the "Remote Terminal Time Tag Counter Register (0x0049)".

17.13.2. RT Time Tag Count Match Interrupts

If the RTTTM interrupt is enabled in the "Hardware Interrupt Enable Register (0x000F)", then time tag "count match" interrupts are enabled. When enabled, the hardware RTTTM interrupt occurs when the free running "Remote Terminal Time Tag Counter Register (0x0049)" matches the value contained in the "Remote Terminal Time Tag Utility Register (0x004A)".

17.14. Remote Terminal Interrupt Registers and Their Use

Section 11.7 on page 39 through Section 11.9 describe how the host uses three Hardware Interrupt registers, the Interrupt Log Buffer and the Interrupt Count & Log Address Register to manage interrupts. Three additional registers are dedicated to the RT interrupts. Comparable to the Hardware Interrupt register triplet, the RT has

- A “Remote Terminal (RT) Interrupt Enable Register (0x0012)” to enable and disable interrupts
- A “Remote Terminal (RT) Pending Interrupt Register (0x0009)” to capture the occurrence of enabled interrupts
- A “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)” to enable $\overline{\text{INT}}$ output to host, for pending enabled interrupts

Each individual bit in all three registers is mapped to the same interrupt-causing event when the corresponding interrupt condition is enabled. Numerous interrupt options are available. At initialization, bits are set in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” to identify the interrupt-causing events which are heeded by the IP core. Most RT applications only use a subset of available interrupt options. Interrupt-causing events are ignored when their corresponding bits are reset in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”. Setting an Interrupt Enable register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

Whenever a RT interrupt event occurs (and the corresponding bit is already set in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”), these actions occur:

- The Interrupt Log Buffer is updated.
- A bit corresponding to the interrupt type is set in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”. The type bit is logically-ORed with the preexisting register value, retaining bits for prior, unserviced RT interrupts.
- RT Interrupt Pending (RTIP) bit 1 is set in the “Hardware Pending Interrupt Register (0x0006)”. The RTIP bit is logically-ORed with the preexisting register value, retaining bits for unserviced hardware interrupts and the preexisting status of the BCIP and MTIP (Bus Controller and MT) interrupt pending bits.
- If the matching bit is already set in the “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)”, an $\overline{\text{INT}}$ output occurs.

If the matching bit in the “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)” was not already set (i.e., low priority polled interrupt), the host can poll the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” to detect the occurrence of interrupts, indicated by non-zero value. Reading the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” automatically clears it to 0x0000.

A single $\overline{\text{INT}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the four Interrupt Output Enable registers (hardware, BC, RT and SMT). Multiple interrupt-causing events can occur simultaneously, so single or simultaneous interrupt events can assert the $\overline{\text{INT}}$ host interrupt output.

When the host receives an $\overline{\text{INT}}$ signal from the IP core, it identifies the event(s) that triggered the interrupt. Section 11.7 describes two methods for identifying the interrupt source(s). One scheme uses the three low order bits in the “Hardware Pending Interrupt Register (0x0006)” to indicate when BC, RT and SMT interrupts occur. When RT Interrupt Pending (RTIP) bit 1 is set in the “Hardware Pending Interrupt Register (0x0006)”, the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” contains a nonzero value and may be read next to identify the specific RT interrupt event(s). Or, the host can directly interrogate the Interrupt Count & Log Address Register, followed by the Interrupt Log Buffer. Section 11.7 has a detailed description.

17.14.1. Remote Terminal (RT) Interrupt Enable Register (0x0012)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RTMC8	IXEQZ	ILCMD	IBR	MERR	IWA	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

17.14.2. Remote Terminal (RT) Pending Interrupt Register (0x0009)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RTMC8	IXEQZ	ILCMD	IBR	MERR	IWA	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

17.14.3. Remote Terminal (RT) Interrupt Output Enable Register (0x0016)

Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RTMC8	IXEQZ	ILCMD	IBR	MERR	IWA	Reserved	Reserved	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

Three registers govern RT interrupt behavior: the RT Interrupt Enable Register, the RT Pending Interrupt Register and the RT Interrupt Output Enable Register. When a bit is set in the RT Interrupt Enable Register, the corresponding RT interrupt is enabled. When a bit is reset in this register, the corresponding interrupt event is unconditionally disregarded. Setting a register bit from 0 to 1 does not trigger interrupt recognition for events that occurred while the bit was zero.

When an enabled RT interrupt event occurs, the corresponding bit is set in the RT Pending Interrupt Register and the Interrupt Log Buffer is updated. To simplify interrupt decoding, RTIP bit 2 in the “Hardware Pending Interrupt Register (0x0006)” is also set whenever a message sets at least one bit in the RT Pending Interrupt Register.

If the corresponding bit is set in the RT Interrupt Output Enable Register, the $\overline{\text{INT}}$ output is asserted at message completion. The RT Interrupt Output Enable Register establishes two priority levels: high priority interrupts generate an $\overline{\text{INT}}$ output while low priority interrupts do not. Both priority levels update the Pending Interrupt Register and Interrupt Log Buffer. The host can detect low priority (masked) interrupts by polling Pending Interrupt registers. When the INTOFF bit 0 is set in “Extended Configuration Register (0x004D)” on page 46, $\overline{\text{INT}}$ signal assertion for enabled RT interrupt-causing events is suppressed when the command is illegal (Message Error response) or results in RT busy status.

When one or more bits are set in the RT Interrupt Enable Register, occurrence of an enabled RT interrupt-causing event triggers an “Interrupt Log Buffer” update. The Interrupt Identification Word (written to the “Interrupt Log Buffer” on page 40 for RT events) mirrors the RT Pending Interrupt register. **NOTE:** While bit 9 is reserved (always 0) in the RT Pending Interrupt register, bit 9 in the written log buffer Interrupt Identification Word (IIW) has a defined function (when INTBUSY bit 2 in “Extended Configuration Register (0x004D)” on page 46 is set, RT IIW bit 9 serves as WASBUSY status flag, asserted if the RT was Busy when the RT interrupt event occurred. RT Busy status is not an interrupt-causing event.).

The following table describes the shared bit descriptions used by all three RT interrupt registers:

Bit No.	*Mnemonic*	*Function*
15 − 10	Reserved	These bits are not used.
9	Reserved	Bit 9 cannot be written and reads back logic 0.
8	RTMC8	RT Mode Code 8 Command Interrupt. Remote terminal processed a valid MIL-STD-1553 “reset remote terminal” mode code command. An RTMC8 interrupt notifies the host when the Bus Controller commands remote terminal reset. If the RTMC8 bit is reset in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” when the RT decodes a valid “reset remote terminal” mode command, bit 0 in the “Remote Terminal Configuration Register (0x0017)” dictates whether the reset response is automatic, or host controlled. The event does not affect the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, the Interrupt Log or the $\overline{\text{INT}}$ output, but the $\overline{\text{INTMC8}}$ output signal is asserted to indicate that the RT needs reset. In the case of a broadcast command to RT31, $\overline{\text{INTMC8}}$ output signal will also be asserted.
7	IXEQZ	RT Index Equals Zero Interrupt. Index counts are used in multi-message bulk data transfers. “Index equals zero” occurs when the last expected message was transacted. Defined IXEQZ interrupt events comprise: (a) subaddresses using indexed buffer mode when the index decrements from 1 to 0, or (b) subaddresses using circular buffer modes when the pre-determined number of messages has been transacted.
6	ILCMD	RT Illegal Command Interrupt. The Remote Terminal encountered a valid illegal message, as defined in the RT Illegal Command Table. Illegal commands are detected when a new valid command word is decoded and the RT Illegalization Table bit corresponding to the received command is logic 1. (Table bits are logic 0 for legal commands.) The RT Illegalization Table contains nonzero values only when “illegal command detection” is being applied. When illegal commands are received, the RT responds by transmitting a status word with ME “message error” flag set; no data words are transmitted. If the ILCMD bit is reset in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” when a valid illegal command is decoded, the event does not affect the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, the Interrupt Log or the $\overline{\text{INT}}$ output.
5	IBR	RT Broadcast Command Received Interrupt. Broadcast commands are enabled for the RT and the terminal encountered a valid command addressed to RT31, the broadcast command address.
4	MERR	RT Message Error Status Interrupt. The Remote Terminal set its Message Error status flag while processing a valid MIL-STD-1553 message. Message errors are caused by Manchester encoding problems or protocol errors.

Bit No.	*Mnemonic*	*Function*
3	IWA	RT Interrupt When Accessed. The Remote Terminal processed a valid MIL-STD-1553 command having the IWA interrupt enabled in its RT Descriptor Table entry. IWA interrupts are used to notify the host each time certain command words are encountered.
2 − 0	Reserved	Bits 2-0 cannot be written, and read back 000.

18. REMOTE TERMINAL CONFIGURATION AND OPERATION

18.1. Command Responses

A brief review of MIL-STD-1553 commands and responses is appropriate here to establish terminology used in the rest of this data sheet. Shown in Figure 13, each command word is comprised of a sync field, three 5-bit data fields, a single bit denoting Transmit / $\overline{\text{Receive}}$ direction and ends with a parity bit. The hardware decoder uses the sync field to determine word type (command vs. data). Word validity is based on proper sync encoding, Manchester II encoding, correct bit count and correct odd parity for the 16 data bits. Once a valid word with command sync is found, the sync and parity are stripped before the command's 16 data bits are stored for further processing.

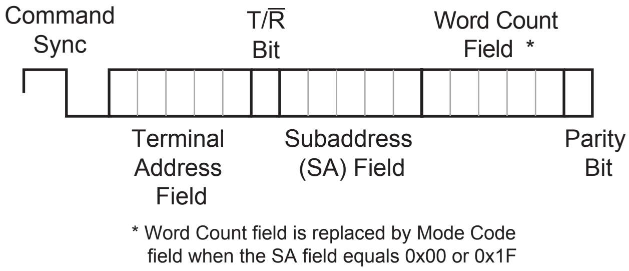


Figure 13. MIL-STD-1553 Command Word Structure

A "valid command" can be specifically addressed to the individual terminal (the command word's embedded Terminal Address field matches the terminal address latched in the Operational Status register) or can be a "broadcast command" addressed to all terminals. Broadcast commands are always addressed to RT address 31 (0x1F). In systems where broadcast commands are disallowed, RT31 is not used as a conventional terminal address. When set, the BCSTINV bit in the "Remote Terminal Configuration Register (0x0017)" renders RT31 commands as "invalid": broadcast commands are indistinguishable from commands addressed to other terminals. Invalid commands are simply disregarded.

When the command word's 5-bit SA (subaddress) field is in the range of 1 to 30 (0x01 to 0x1E) the command is considered a "subaddress command". The terminal will either receive or transmit data words, and "direction" is specified by the command's T/$\overline{R}$ bit. The number of data words transacted is specified in the 5-bit word count field, ranging from 1 to 32 words. Thirty-two data words is represented when the word count field equals 0x00.

When the command's 5-bit subaddress field equals 0 or 31 (0x1F) a "mode code" command is indicated; the low order five bits no longer specify a word count, instead they convey a mode code value. This data sheet refers to mode code commands by the mode code number. For example, a mode command with 5-bit mode code field of 0x10 is called MC16, and the full range of mode code values is MC0 through MC31 (decimal).

Mode codes MC16 through MC31 (0x10 through 0x1F) have a single associated data word. When the command T/$\overline{R}$ bit equals 0, the data word is contiguous with the command word and received by the RT. When the command's T/$\overline{R}$ bit equals 1, the data word is transmitted by the RT, following the terminal's transmitted status word.

Mode codes MC0 through MC15 (0x0F) do not have associated data words. For these 16 commands, the command T/$\overline{R}$ bit does not specify "direction". These commands must be transmitted with T/$\overline{R}$ bit equal to 1. If the T/$\overline{R}$ bit is 0, the mode command is "undefined".

Twenty-two mode commands are "undefined mode commands " in MIL-STD-1553B:

Mode Codes 0 through 15 with T/R bit = 0
Mode Codes16, 18 and19 with T/R bit = 0
Mode Codes 17, 20 and 21 with T/R bit = 1

The UMCINV bit in the "Remote Terminal Configuration Register (0x0017)" determines how these undefined mode commands are handled by the IP core. If the UMCINV configuration bit equals 1, the undefined mode commands are treated as invalid. They are not recognized by the IP core. There is no terminal response and status is not updated. If the UMCINV configuration bit equals 0, the 22 undefined mode commands are considered valid; this is the default condition following reset. For this case, terminal response depends on whether or not the application uses "illegal command detection."

If illegal command detection is not used, all Illegalization Table entries should be logic 0, including the 22 entries for these undefined commands. (The Illegalization Table is fully described in Section 18.2 on page 140. After $\overline{\text{RESET}}$ reset, all entries equal logic 0.) The terminal responds "in form", transmitting clear status (and a single mode data word if the command is MC17, MC20 or MC21 with T/$\overline{\text{R}}$ bit = 1). Terminal status is updated.

If illegal command detection applies, the Illegalization Table entries for these 22 undefined commands should be initialized to logic 1. In this case, the terminal will respond with status word only, with Message Error bit set. No mode data word is transmitted. Terminal status is updated.

Twenty-seven mode codes are considered "reserved" in MIL-STD-1553B:

Mode Codes 9 through 15 with T/R bit = 1
Mode Codes 22 through 31 with T/R bit = 1
Mode Codes 22 through 31 with T/R bit = 0

Treatment of these reserved mode commands depends on their respective Illegalization Table entries. As described above for undefined mode commands, response depends on whether or not illegal command detection applies.

Any mode commands not implemented in the terminal should be treated the same as reserved mode commands.

The important point is that "illegal command detection" should be universally applied (or not applied) when setting up a Remote Terminal application. Here are the two options:

Not using Illegal command detection. The Illegalization Table is left in its default state (all locations equal to $\overline{\text{RESET}}$ post-reset 0x0000). The terminal responds "in form" to all valid commands, whether legal or illegal.

Using illegal command detection. The Illegalization Table is initialized by the host to implement "illegal command detection". The host sets bits for all illegal commands. This generally includes the reserved and unimplemented mode commands, unimplemented subaddresses (or specific word counts, T/$\overline{\text{R}}$ bit states, and/or broadcast vs. non-broadcast status within subaddresses). Treatment for the undefined mode commands depends on UMCINV bit.

The host defines terminal response for all individual commands by initializing the Descriptor Table, fully described later. At this point, a few comments about the Descriptor Table are appropriate.

The command SA (subaddress) field has a range of 0 to 31 (0x1F). When SA is in the range 1 to 30 (0x1E), the command is a transmit or receive "subaddress command". The number of data words transmitted or received is expressed in the low order 5 bits. When SA equals 0 or 31 (0x1F) the command is a mode command and the mode code value is expressed in the low order 5 bits.

For each subaddress, separate table "descriptor blocks" for transmit and receive commands permit different data buffering to be applied. The host initializes the table so each transmit-subaddress and each receive-subaddress uses one of four methods for storing message data. During table initialization, memory is allocated in shared RAM for storing message data according to the application requirements. Each transmit-subaddress and receive-subaddress has one or more data pointers (depending on buffer method) addressing its reserved data buffer(s).

Each mode command also has its own table "descriptor block". Mode commands have either one data word or no associated data words. Descriptor words used as data pointers by "subaddress commands" are instead used for direct storage of transacted mode data words. Mode commands that transmit or receive mode data words have a dedicated storage address range in shared RAM, eliminating the need for descriptor table data pointers.

Each mode command with mode data word has its own fixed address for data storage. This includes reserved mode codes with data word. Thus the IP core can respond consistently for all mode commands; transmitted data values for "in form" responses (when "illegal command detection" is not used) can be predetermined, even for the reserved mode commands.

18.1.1. RT to RT Commands.

The MIL-STD-1553 standard allows for data word transmission from a specified transmitting terminal to a different receiving terminal. When broadcast commands are allowed, data transmission can be addressed to the broadcast terminal address, RT31. If broadcast is allowed, the host should initialize the BCSTINV (broadcast invalid) bit in the "Remote Terminal Configuration Register (0x0017)".

All RT to RT commands are characterized by a pair of contiguous command words: Command Word 1 is a receive command addressed to the intended receiving terminal, then Command Word 2 is a transmit command addressed to a single transmitting terminal. Command Word 2 cannot be broadcast address RT31. The IP core automatically detects and handles RT to RT commands, except when either command word contains a subaddress field equal to 0x0 or 0x1F. Either subaddress value indicates a mode code command; the IP core treats RT to RT commands with mode code as invalid. If either RT-RT command word is addressed to the terminal but contains subaddress 0x0 or 0x1F, the command is not recognized; there is no RT command response, and no status updating for the benefit of following "transmit status" or "transmit last command" mode commands.

When either RT-RT command word (with subaddress field not equal to 0x0 or 0x1F) is addressed to the terminal, but the other command word contains subaddress 0x0 or 0x1F, the RT-RT command is not recognized as valid. There is no RT command response, and no status updating for the benefit of following "transmit status" or "transmit last command" mode commands.

An RT-RT command pair where Command Word 1 is addressed to the terminal and Command Word 2 is addressed to a different terminal is considered an "RT-RT receive" command. When the message is transacted, the IP core sets the RTRT bit in the Receive Subaddress Message Information Word in the subaddress data buffer.

An RT-RT command pair where Command Word 2 is solely addressed to the terminal (not RT31) is considered an "RT-RT transmit" command. The Message Information Word does not distinguish the RT to RT transmit message from an ordinary RT to BC transmit command.

18.2. Command Illegalization Table

The following pages describe various structures residing in the RAM shared between the host and command processing logic. The host initializes these structures to control the terminal's response to received commands. The first structure described is the command Illegalization Table used for "illegal command detection".

Illegal command detection is an optional process. When illegal command detection is not used, the terminal "responds in form" to all valid commands: it sends Clear Status and transacts the number of data words defined in the received command. When illegal command detection is not used, the bus controller cannot tell whether the command is legal or illegal, from the terminal's transmitted response.

If illegal command detection is used, the terminal responds differently when an illegal command is detected. The terminal responds to illegal commands with "message error" status, transmitting only status word. Data word transmission is suppressed if the command type inherently includes transmitted data words. The terminal responds to each legal command with clear status and transacts the number of data words defined in the type of command received.

For consistency, apply illegal command detection to all illegal and unimplemented commands, and to all reserved or undefined mode code commands, or "respond in form" to all of these commands (illegal command detection disabled) by leaving the Illegalization Table in the all-cleared default state after RESET master reset

The IP core uses a 256-word "Illegalization Table" in shared RAM to distinguish between legal and illegal commands. After the (RESET) master reset input is negated, the IP core performs internal self test including a shared RAM test which leaves all memory locations fully reset. Once self test is complete, the READY output goes high to indicate readiness for host initialization. At this point, all entries in the Illegalization Table read logic 0, so by default, illegal command detection is not applied.

To apply illegal command detection, the host writes the Illegalization Table to set bits for all illegal command combinations. This typically includes any unimplemented subaddresses and/or word counts, undefined mode commands, reserved mode commands and any mode commands not implemented in the terminal design.

Once RTSTEX is set in the "Master Configuration Register 1 (0x0000)", terminal execution begins. Each time a valid command is received, a 1-bit entry (indexed using command word data bits) is fetched from the Illegalization Table:

If fetched Illegalization Table bit equals logic 0, the command is "legal"; the terminal responds "in form", transmitting clear status and transacting the number of data words defined for the message type. Terminal status is updated.

If fetched Illegalization Table bit equals logic 1, the command is "illegal"; the terminal responds with status word only, with Message Error bit set. No data words are transmitted. Terminal status is updated.

When illegal command detection is not applied, all table entries should read logic 0; the terminal responds "in form" to all valid commands.

The illegalization scheme allows any subset of command T/$\overline{\text{R}}$ bit, broadcast vs. non-broadcast status, subaddress and word count (or mode code number), for a total of 4,096 legal/illegal command combinations. Commands may be illegalized down to the word count level. For example, 10-word receive commands to a given subaddress may be legal, while 9-word receive commands to the same subaddress are illegal.

Broadcast receive commands are illegalized separately from non-broadcast receive commands. Transmit and receive commands for the same subaddress are illegalized separately. For mode commands, any combination of mode code number, T/$\overline{\text{R}}$ bit and broadcast/non-broadcast status can be legal or illegal.

The Illegalization Table is located in shared RAM within the fixed address range of 0x0200 to 0x02FF. See Figure 14. The table is comprised of 256 16-bit words. To cover the full range of 1 to 32 data words, each subaddress uses a pair of illegalization registers. The lower register (even memory address) covers word counts 0 to 15, using one bit per word count. As in command encoding, "0" denotes 32 data words. Bit 0 corresponds to 32 data words, bit 1 corresponds to 1 data word and bit 15 corresponds to 15 data words. The upper register (odd memory address) similarly covers word counts 16 to 31, using one bit per word count. Bit 0 corresponds to 16 data words, while bit 15 corresponds to 31 data words.

When a command's subaddress field equals 0 or 31 (0x1F), the command is a mode command. Table entries for mode commands use bits to represent mode code numbers, not word counts. The lower register (even memory address) covers mode codes 0 to 15, using one bit per mode code. Bit 0 corresponds to mode code 0, bit 15 corresponds to mode code 15. The upper register (odd memory address) similarly covers mode codes 16 to 31, using one bit per mode code. Bit 0 corresponds to mode code 16, bit 15 corresponds to mode code 31. There is no functional difference between SA0 mode commands and SA31 mode commands. Since either subaddress indicates a mode command, the subaddress 0 table words should match the subaddress 31 table words in each quadrant.

Table entries from 0x0242 to 0x027D do not have to be programmed. These correspond to broadcast transmit subaddress commands (undefined by MIL-STD-1553B) and are always invalid. There is no terminal response.

Addressing for the Illegalization Table is derived from the command word T/$\overline{\text{R}}$ bit, subaddress field, MSB of the Word Count (Mode Code) field and the command's broadcast vs. non-broadcast status as shown below in Figure 14.

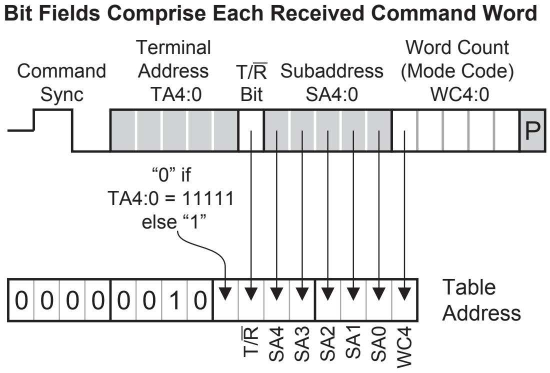


Figure 14. Deriving the Illegalization Table Address From the Received Command Word

Figure 16 on page 144 shows individual bit locations in the Illegalization Table for broadcast and non-broadcast variants of all mode commands defined by MIL-STD-1553B. Locations are also identified for reserved mode codes and undefined mode code commands.

The following examples illustrate how the Illegalization Table is initialized to distinguish between legal and illegal commands when "illegal command detection" is being used. Remember: If the terminal does not use illegal command detection, the table is left in its post-$\overline{\text{RESET}}$ reset state, with all table locations reset to 0x0000. In this case, all command responses are "in form".

For "subaddress commands" (ordinary receive commands or transmit commands) individual table bits correspond to word counts specified in the received command word. If a bit is 0, the corresponding word count is legal. If a bit is 1, the corresponding word count is illegal.

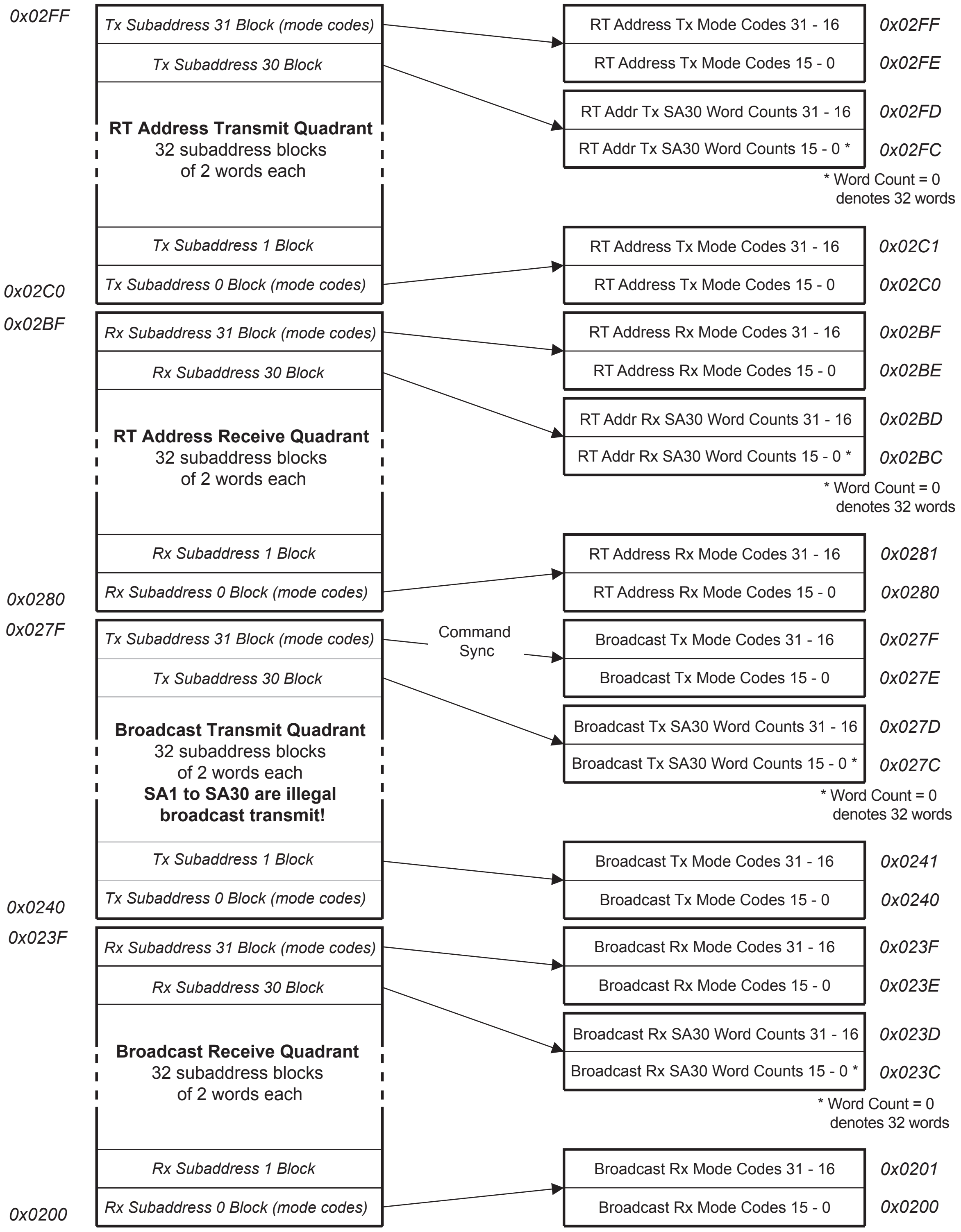
For example, transmit commands to RT subaddress 1 are controlled by the words at 0x02C2 and 0x02C3. In Figure 15, these words are located in the "RT Address Transmit" block. The word stored at 0x02C3 controls subaddress 1 transmit commands having word counts 16 to 31. The word stored at 0x02C2 controls subaddress 1 transmit commands having word counts 1 to 15 or 32. (Reminder: In MIL-STD-1553B, zero corresponds to 32 words.)

```
Word at 0x02C3 (Tx Subaddr 1) 31 to 16 words
Bit      15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
Words    31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

Word at 0x02C2 (Tx Subaddr 1) 15 to 1 & 32 words
Bit      15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
Words    15 14 13 12 11 10  9  8  7  6  5  4  3  2  1 32
```

If the word stored at 0x02C3 = 0xFFFF and the word stored at 0x02C2 = 0xFF0F, then commands with 4, 5, 6, or 7 data words are the only legal transmit commands for subaddress 1 and all other word counts are illegal. Receive commands and broadcast receive commands for Subaddresses1 through 30 are encoded similarly.

For "mode code commands" (characterized by command word subaddress field equal to 00000 or 11111 binary) individual table bits correspond to individual mode code values. Here "transmit" and "receive" simply indicate the state of the command word T/$\overline{R}$ bit. (For mode codes 0-15, the T/$\overline{R}$ bit does not indicate data direction since data is not transacted when fulfilling these commands).


0x02FF
Tx Subaddress 31 Block (mode codes)
Tx Subaddress 30 Block
RT Address Transmit Quadrant
32 subaddress blocks
of 2 words each
Tx Subaddress 1 Block
0x02C0
Tx Subaddress 0 Block (mode codes)
0x02BF
Rx Subaddress 31 Block (mode codes)
Rx Subaddress 30 Block
RT Address Receive Quadrant
32 subaddress blocks
of 2 words each
Rx Subaddress 1 Block
0x0280
Rx Subaddress 0 Block (mode codes)
0x027F
Tx Subaddress 31 Block (mode codes)
Tx Subaddress 30 Block
Broadcast Transmit Quadrant
32 subaddress blocks
of 2 words each
SA1 to SA30 are illegal
broadcast transmit!
Tx Subaddress 1 Block
0x0240
Tx Subaddress 0 Block (mode codes)
0x023F
Rx Subaddress 31 Block (mode codes)
Rx Subaddress 30 Block
Broadcast Receive Quadrant
32 subaddress blocks
of 2 words each
Rx Subaddress 1 Block
0x0200
Rx Subaddress 0 Block (mode codes)
RT Address Tx Mode Codes 31 - 16 0x02FF
RT Address Tx Mode Codes 15 - 0 0x02FE
RT Addr Tx SA30 Word Counts 31 - 16 0x02FD
RT Addr Tx SA30 Word Counts 15 - 0 * 0x02FC
* Word Count = 0
denotes 32 words
RT Address Tx Mode Codes 31 - 16 0x02C1
RT Address Tx Mode Codes 15 - 0 0x02C0
RT Address Rx Mode Codes 31 - 16 0x02BF
RT Address Rx Mode Codes 15 - 0 0x02BE
RT Addr Rx SA30 Word Counts 31 - 16 0x02BD
RT Addr Rx SA30 Word Counts 15 - 0 * 0x02BC
* Word Count = 0
denotes 32 words
RT Address Rx Mode Codes 31 - 16 0x0281
RT Address Rx Mode Codes 15 - 0 0x0280
Command
Sync
Broadcast Tx Mode Codes 31 - 16 0x027F
Broadcast Tx Mode Codes 15 - 0 0x027E
Broadcast Tx SA30 Word Counts 31 - 16 0x027D
Broadcast Tx SA30 Word Counts 15 - 0 * 0x027C
* Word Count = 0
denotes 32 words
Broadcast Tx Mode Codes 31 - 16 0x0241
Broadcast Tx Mode Codes 15 - 0 0x0240
Broadcast Rx Mode Codes 31 - 16 0x023F
Broadcast Rx Mode Codes 15 - 0 0x023E
Broadcast Rx SA30 Word Counts 31 - 16 0x023D
Broadcast Rx SA30 Word Counts 15 - 0 * 0x023C
* Word Count = 0
denotes 32 words
Broadcast Rx Mode Codes 31 - 16 0x0201
Broadcast Rx Mode Codes 15 - 0 0x0200

Illegalization Table Comprised of 32 2-word Blocks per Quadrant

Example Subaddress Blocks from Each Table Quadrant

Figure 15. Address Mapping for Illegalization Table

Note: Default start address is 0x0200.

Figure 16 summarizes the 16 Illegalization Table locations for mode commands. These locations are scattered throughout the overall Illegalization Table shown in Figure 13. Remember: the host must initialize all table locations corresponding to both subaddress 0 and subaddress 31 (11111 binary).

Consider an example in which all reserved and all undefined mode commands are illegal. If all RT defined transmit mode commands are legal except MC0 (”dynamic bus control”) the eight table entries for transmit mode commands would be:

```
0x02FF and 0x02C1 = 1111 1111 1111 0010 = 0xFFF2   Tx MC with data
0x02FE and 0x02C0 = 1111 1110 0000 0001 = 0xFE01   Tx MC without data
0x027F and 0x0241 = 1111 1111 1111 1111 = 0xFFFF   Br.Tx MC with data (all illegal)
0x027E and 0x0240 = 1111 1110 0000 0101 = 0xFE05   Br.Tx MC without data
```

The receive mode command words are encoded similarly. Continuing the same example where all reserved and all undefined mode commands are illegal: If all RT defined receive mode commands are legal, the eight table entries for receive mode commands would be:

```
0x02BF and 0x0281 = 1111 1111 1100 1101 = 0xFFCD   Rx MC with data
0x02BE and 0x0280 = 1111 1111 1111 1111 = 0xFFFF   Rx MC without data (all illegal)
0x023F and 0x0201 = 1111 1111 1100 1101 = 0xFFCD   Br.Rx MC with data
0x023E and 0x0200 = 1111 1111 1111 1111 = 0xFFFF   Br.Rx MC without data (all illegal)
```

RAM Address			Bit No.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02FF and 0x02C1	Tx MC31 - MC16	Transmit Mode Commands With Data	MC #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			Status	R	R	R	R	R	R	R	R	R	R	U	U	D	D	U	D
0x02FE and 0x02C0	Tx MC15 - MC0	Transmit Mode Commands Without Data	MC #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			Status	R	R	R	R	R	R	R	D	D	D	D	D	D	D	D	D
0x02BF and 0x0281	Rx MC31 - MC16	Receive Mode Commands With Data	MC #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			Status	R	R	R	R	R	R	R	R	R	R	D	D	U	U	D	U
0x02BE and 0x0280	Rx MC15 - MC0	Receive Mode Commands Without Data	MC #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			Status	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
0x027F and 0x0241	Br.Tx MC31 - MC16	Broadcast Transmit Mode Commands With Data	MC #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			Status	NB	NB	NB	NB	NB	NB	NB	NB	NB	NB	U	U	NB	NB	U	NB
0x027E and 0x0240	Br.Tx MC15 - MC0	Broadcast Transmit Mode Commands Without Data	MC #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			Status	R	R	R	R	R	R	R	D	D	D	D	D	D	NB	D	NB
0x023F and 0x0201	Br.Rx MC31 - MC16	Broadcast Receive Mode Commands With Data	MC #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			Status	R	R	R	R	R	R	R	R	R	R	D	D	U	U	D	U
0x023E and 0x0200	Br.Rx MC15 - MC0	Broadcast Receive Mode Commands Without Data	MC #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			Status	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

LEGEND
D = Defined Mode Command R = Reserved Mode Code
U = Undefined Mode Command NB = Broadcast Not Allowed

Figure 16. Summary of RT Illegalization Table Addresses for Mode Code Commands

18.3. Temporary Receive Data Buffer

The 32-word temporary receive data buffer resides in shared RAM in address space 0x01C0 to 0x01DF. The IP core optionally uses this buffer for temporary storage of receive data words until successful message completion. To enable the buffer, the host asserts the TRXDC bit in the “Remote Terminal Configuration Register (0x0017)”.

When enabled, the terminal stores received data words in the 32-word buffer during message processing. Upon error-free message completion, all buffered words are written in a burst to the data buffer memory assigned to the specific subaddress in the RT Descriptor Table.

When the TRXDB bit in the “Remote Terminal Configuration Register (0x0017)” is negated, the temporary receive data buffer is disabled. At 20us intervals, the terminal writes received data words to assigned subaddress data buffer memory as each word is received. If message error occurs during data reception, data integrity is lost; valid data from the prior receive message may be partially overwritten by data from a message ending in error. MIL-STD-1553 states that all received data from messages ending in error **should be disregarded.**

In a typical application, the temporary buffer is not directly accessed by the host, although there is no restriction preventing host data access. The host should never write data into the temporary buffer space.

18.4. Descriptor Table

The Descriptor Table, resides in shared RAM, at default address ranges 0x0400 to 0x05FF. This table is initialized by the host to define how the terminal processes valid commands. Descriptor Table settings for each command specify where message data is stored, how data is stored, whether host interrupts are generated, and other aspects essential to command processing. **Before initializing the RAM Descriptor Table, the RTENA bit must be set in the “Master Configuration Register 1 (0x0000)”.** Terminal execution does not begin until the RTSTEX bit is set in “Master Configuration Register 1 (0x0000)”.

Shown in Figure 17, the table consists of 128 consecutive “descriptor blocks”, each comprised of four 16-bit words. The table is organized into four quadrants.

The Receive Subaddress and Transmit Subaddress quadrants define response for commands having a subaddress field ranging from 1 to 30 (0x1E). These are simple N-data word receive or transmit commands, where N can range from 1 to 32 words. When the command T/$\overline{R}$ bit equals 0, the receive command quadrant applies. When the T/$\overline{R}$ bit equals 1, the transmit command quadrant applies.

Both subaddress quadrants are padded at top and bottom with unused Descriptor Blocks for subaddresses 0 and 31 (0x1F). The word space reserved for SA0 and SA31 aligns the table addressing, but values stored in these eight locations is not used. Command subaddresses 0 and 31 indicate mode commands. The response for commands containing either SA value is defined in the two mode command table quadrants. The Receive Mode Command quadrant applies when the command word T/$\overline{R}$ bit equals 0, while the Transmit Mode Command quadrants applies when T/$\overline{R}$ equals 1.

The term “Transmit Mode Command” is misleading. All defined mode commands with mode code less than 0x0F haveT/$\overline{R}$ bit equal to 1, yet none of these mode commands transmits a data word. They transmit only the terminal status word, just like receive commands. However, the RT responds to transmit mode commands with mode code 0x10 to 0x1F by transmitting a mode data word. Just three such transmit mode commands are defined.

Within the Receive and Transmit Mode Command quadrants, block addressing is based on the low order 5 bits in the command word, containing the mode code value. This is fundamentally different from the Subaddress quadrants in which block addressing is based on the 5-bit subaddress field. Figure 18 shows how to derive Control Word address from the received Command Word. The Control Word address for the last valid command can also be found in the “Remote Terminal Current Control Word Address Register (0x0003)”.

All 128 4-word Descriptor Blocks start with a Control Word. There are four Control Word variants based on command type: receive vs. transmit and mode vs. non-mode commands. All descriptor Control Words are initialized by the host to define basic command response. Each Control Word specifies the data buffer method and host interrupt for a specific subaddress or mode command.

Each subaddress has both a Receive Subaddress block and a Transmit Subaddress block. Receive and transmit commands to the same subaddress can be programmed to respond differently.

The function of the three remaining descriptor words (in each 4-word block) depends on which of the 4 data buffer methods are specified in the Control Word.

Indexed (or Single Buffer) Method where a predetermined number of messages is transacted using a single data buffer in shared RAM. Several host interrupt options are offered, including an interrupt generated when all N messages are successfully completed.

Double (or Ping-Pong) Buffer Method where successive messages alternate between two data buffers in shared RAM. Several host interrupt options are offered.

Circular Buffer Mode 1 where buffer boundaries determine when the bulk transfer is complete and message information and time-tag words are stored with message data in a common buffer. Several host interrupt options are offered, including an interrupt generated when the allocated data buffer is full.

Circular Buffer Mode 2 where the number of messages transacted defines bulk transfer progress, and message data words are stored contiguously in one buffer while message information and time-tag words are stored in a separate buffer. Several host interrupt options are offered, including an interrupt generated when all N messages are successfully completed.

The 4-word Descriptor Table entry for each command (its descriptor block) begins with a Control Word. There are four types of descriptor Control Word:

- Receive Subaddress Control Word
- Transmit Subaddress Control Word
- Receive Mode Command Control Word
- Transmit Mode Command Control Word

The descriptor Control Word is initialized by the host to select data buffer method and interrupt options. After a command is processed by the terminal, the IP core updates the command's descriptor Control Word. Updates will differ based on the chosen data buffer method. Reading the descriptor table can differ from other RAM accesses.

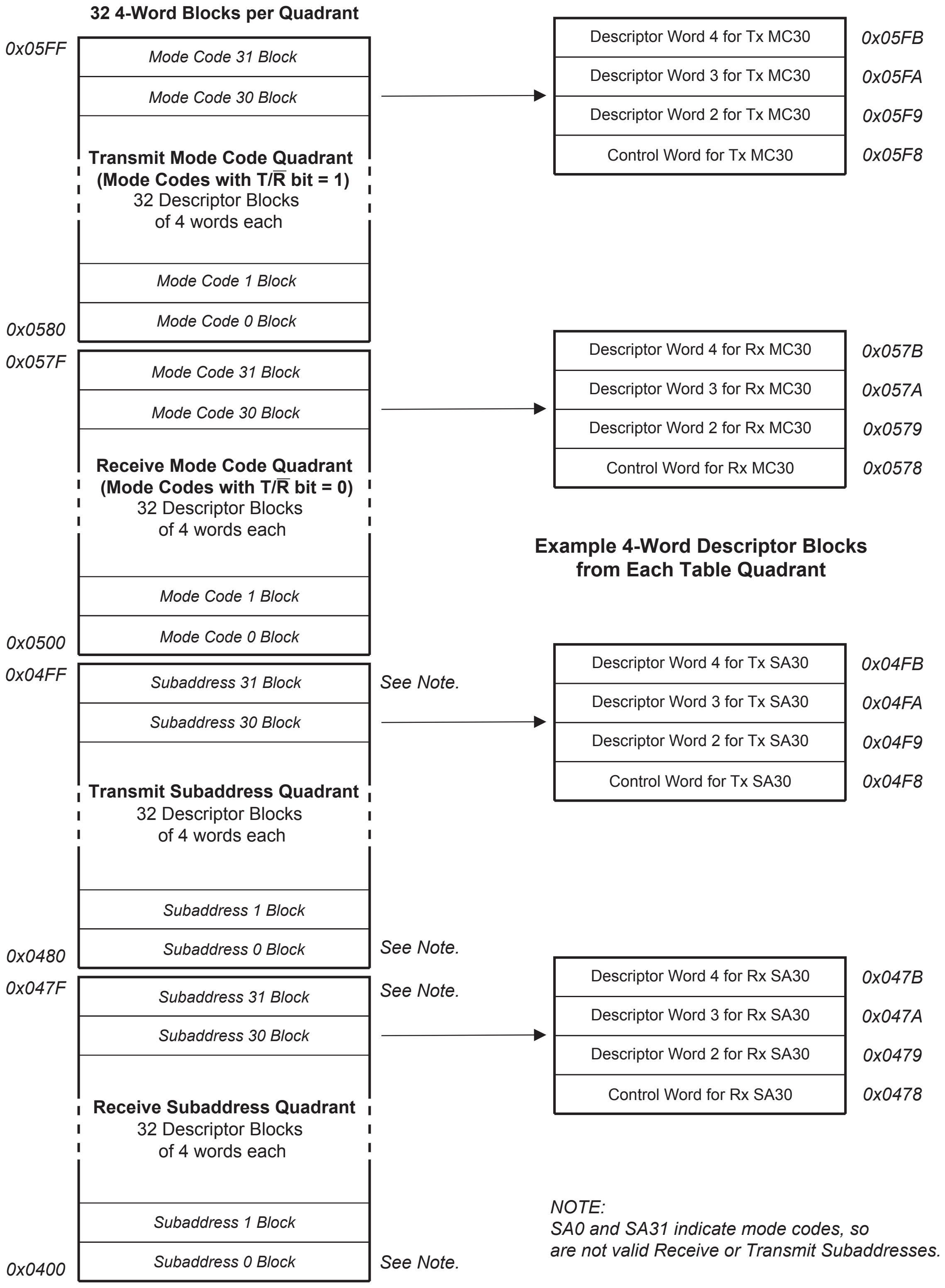

32 4-Word Blocks per Quadrant
0x05FF
Mode Code 31 Block
Mode Code 30 Block
Transmit Mode Code Quadrant
(Mode Codes with T/R̄ bit = 1)
32 Descriptor Blocks
of 4 words each
Mode Code 1 Block
Mode Code 0 Block
0x0580
0x057F
Mode Code 31 Block
Mode Code 30 Block
Receive Mode Code Quadrant
(Mode Codes with T/R̄ bit = 0)
32 Descriptor Blocks
of 4 words each
Mode Code 1 Block
Mode Code 0 Block
0x0500
0x04FF
Subaddress 31 Block
See Note.
Subaddress 30 Block
Transmit Subaddress Quadrant
32 Descriptor Blocks
of 4 words each
Subaddress 1 Block
Subaddress 0 Block
See Note.
0x0480
0x047F
Subaddress 31 Block
See Note.
Subaddress 30 Block
Receive Subaddress Quadrant
32 Descriptor Blocks
of 4 words each
Subaddress 1 Block
Subaddress 0 Block
See Note.
0x0400
Descriptor Word 4 for Tx MC30 0x05FB
Descriptor Word 3 for Tx MC30 0x05FA
Descriptor Word 2 for Tx MC30 0x05F9
Control Word for Tx MC30 0x05F8
Descriptor Word 4 for Rx MC30 0x057B
Descriptor Word 3 for Rx MC30 0x057A
Descriptor Word 2 for Rx MC30 0x0579
Control Word for Rx MC30 0x0578
Example 4-Word Descriptor Blocks
from Each Table Quadrant
Descriptor Word 4 for Tx SA30 0x04FB
Descriptor Word 3 for Tx SA30 0x04FA
Descriptor Word 2 for Tx SA30 0x04F9
Control Word for Tx SA30 0x04F8
Descriptor Word 4 for Rx SA30 0x047B
Descriptor Word 3 for Rx SA30 0x047A
Descriptor Word 2 for Rx SA30 0x0479
Control Word for Rx SA30 0x0478
NOTE:
SA0 and SA31 indicate mode codes, so
are not valid Receive or Transmit Subaddresses.

Figure 17. Address Mapping for RT Descriptor Table

Note: Assumes default table base address = 0x0400.
Before initializing the RT Descriptor Table, the RTENA bit must be set in Master Configuration Register 0x0000.

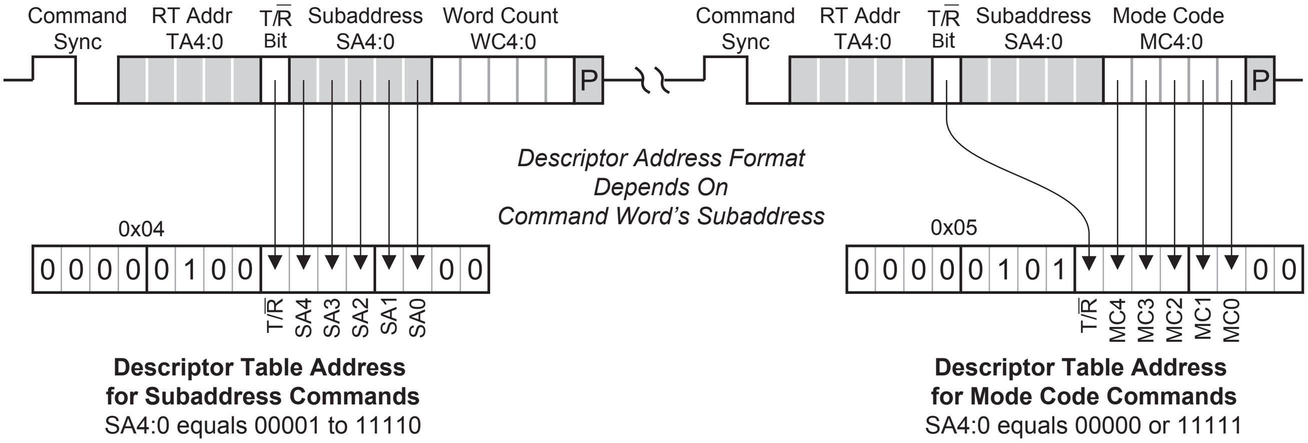


Figure 18. Deriving a Descriptor Table Control Word Address From Command Word
(assumes table base address = 0x0400)

18.4.1. Receive Subaddress Control Word

Receive Subaddress Control Words apply when a valid command word $T/\overline{R}$ bit equals zero (receive) and the subaddress field has a value in the range of 1 to 30 (0x1E). The descriptor Control Word defines terminal command response and interrupt behavior, and conveys activity status to the host. It is initialized by the host before terminal execution begins. If using ping-pong data buffers, Control Words should only be written when the RTENA bit is set in the "Master Configuration Register 1 (0x0000)". Failure to meet this requirement prevents automatic assertion of PPON bit 8 when PPEN bit 2 is set, and successive messages will repeatedly use the same buffer. Bits 8-11 cannot be written by the host; these bits are updated by the IP core during terminal execution, that is, when the "Master Configuration Register 1 (0x0000)" RTSTEX bit equals 1. The host can write bits 0-2 and 4-7 only when RSTEX equals zero; bits 3 and 12-15 can be written anytime. This register is cleared to 0x0000 by $\overline{\text{RESET}}$ master reset. Software reset (SRST) clears just the DBAC, DPB and BCAST bits. **Following any read cycle to the Control Word address, the DBAC bit is reset.**

	IXEQZ	IWA	IBR	MKBUSY	DBAC	DPB	BCAST	PPON	CIR2ZN3	CIR2ZN2	CIR2ZN1	CIR2ZN0	STOPP	PPEN	CIR2EN	CIR1EN	
	H	H	H	H	D1	D	D	D	H	H	H	H	H	H	H	H	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle

NOTE: '*Reset*' refers to bit value following Master Reset ($\overline{\text{RESET}}$). The bit value following software reset is unchanged unless specifically indicated by an "SR" value.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	IXEQZ	R/W	0	Interrupt When Index Equals Zero. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is high, assertion of this bit enables generation of an interrupt for (a) subaddresses using indexed buffer mode when the INDX value decrements from 1 to 0, or (b) subaddresses using a circular buffer mode when the pre-determined number of messages has been transacted. If enabled, upon completion of command processing that results in index = 0, an IXEQZ interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, output signal $\overline{\text{INT}}$ is asserted, and the interrupt is registered in the Interrupt Log.
14	IWA	R/W	0	Interrupt When Accessed. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IWA bit is high, assertion of this bit enables interrupt generation when the subaddress receives any valid receive command. If enabled, upon completion of command processing, an IWA interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, output signal $\overline{\text{INT}}$ is asserted, and the interrupt is registered in the Interrupt Log.
13	IBR	R/W	0	Interrupt Broadcast Received. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IBR bit is high, assertion of this bit enables interrupt generation when the subaddress receives a valid broadcast command. If enabled, upon completion of message processing an IBR interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, output signal $\overline{\text{INT}}$ is asserted, and the interrupt is registered in the Interrupt Log. This bit has no function if the BCSTINV bit is high in the “Remote Terminal Configuration Register (0x0017)”. In this case, commands to RT address 31 are not recognized as valid by the IP core.
12	MKBUSY	R/W	0	Make Busy. The host asserts the MKBUSY bit to respond with Busy status for commands to this receive subaddress. This bit is an alternative to globally applying Busy status for all valid commands, enabled from the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”. See that register description for additional information. When Busy is asserted, received data words are not stored and the DPB bit does not toggle after message completion.
11	DBAC	R	0 SR = 0	Descriptor Block Accessed. Internal IP core logic asserts the DBAC bit upon completion of message processing. The host may poll this bit to detect subaddress activity, instead of using host interrupts. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset, SRST software reset or a read cycle to this memory address.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
10	DPB	R	0 SR = 0	Data Pointer B. This status bit is maintained by the IP core and only applies in ping-pong buffer mode. This bit indicates the buffer to be used for the next occurring receive command to this subaddress. When the DPB bit is logic 0, the next message will use Data Pointer A; when DPB is logic 1, the next message uses Data Pointer B. In ping-pong buffer mode, the bit is inverted after each **error-free** message completion. **To also ensure the DPB bit is not altered after illegal commands or messages ending with Busy status, the DPBTOFF bit should be set in the "Extended Configuration Register (0x004D)" on page 46.** This ensures unsuccessful messages are not stored in the data buffer and are overwritten by subsequent successful messages. (see also "Ping-Pong Enable / Disable Handshake" on page 171). The DBP bit is reset to logic 0 by RESET master reset or SRST software reset; therefore the first message received after either reset will use Buffer A. This bit is "don't care" for indexed single-buffer mode or either circular buffer mode.
9	BCAST	R	0 SR = 0	Broadcast Command. IP Core logic sets this bit when a valid broadcast receive command is received at this subaddress. If IBR bit 13 and "Remote Terminal (RT) Interrupt Enable Register (0x0012)" IBR bit are both set, the output signal INT is asserted. This bit has no function if the BCSTINV bit is asserted in the "Remote Terminal Configuration Register (0x0017)"; in this case commands to RT address 31 are not recognized as valid by the IP core. This bit is reset to logic 0 by RESET master reset or SRST software reset.
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is controlled by the IP core and cannot be written by the host. It only applies if PPEN bit 2 was initialized to logic one by the host after reset, enabling ping-pong buffer mode for this subaddress. IP Core logic asserts this bit when it recognizes ping-pong is active for this subaddress. Before off-loading the receive data buffer for this subaddress, the host can ask the IP core to temporarily disable ping-pong by asserting STOPP bit 3. The IP core acknowledges ping-pong is disabled by negating PPON. The host can safely off-load the buffer without data collision while PPON is negated. After buffer servicing, the host asks the IP core to re-enable ping-pong by negating STOPP bit 3. The IP core acknowledges ping-pong is re-enabled by asserting PPON. If PPEN bit 2 is high and PPON bit 8 is low when new commands arrive for this subaddress, ping-pong is disabled. Each new message overwrites existing data in the buffer specified by DPB bit 10, and the DPB bit does not toggle after command completion.
7-4	CIR2ZN	R/W	0	Circular Mode 2 Zero Number. Used only in circular buffer mode 2, this 4-bit field is initialized with the number of trailing zeros in the initialized MIBA address. This is explained in Section 19.6, which fully describes circular buffer mode 2.
3	STOPP	R/W	0	Stop Ping-Pong Request. The host asserts this bit to suspend ping-pong buffering for this subaddress. The host resets this bit to ask the IP core to re-enable ping-pong. The IP core confirms recognition of ping-pong enable or disable status by writing PPON bit 8. Refer to Section 19.3, which fully describes ping-pong mode.

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th colspan="4">Function</th></tr>
<tr><td rowspan="6">2
1
0</td><td rowspan="6">PPEN
CIR2EN
CIR1EN</td><td rowspan="6">R/W
R/W
R/W</td><td rowspan="6">0
0
0</td><td colspan="4">Ping-Pong, Circular Buffer Mode 2 or Circular Buffer Mode 1 Enable.
The PPEN, CIR2EN and CIR1EN bits are initialized by the host to select buffer mode. The table below summarizes how buffer mode selection is encoded.
In the case of ping-pong, the host initializes the PPEN bit to logic one after reset to enable ping-pong buffering for this subaddress. The host asserts STOPP bit 3 to ask the IP core to temporarily disable ping-pong. Negating the STOPP bit asks the IP core to re-enable ping-pong. The IP core confirms ping-pong enable or disable state changes by writing the PPON bit. PPEN bit 2 should only be initialized or otherwise written when the RTENA bit is also set in the “Master Configuration Register 1 (0x0000)”.</td></tr>
<tr><td><b>PPEN</b></td><td><b>CIR2EN</b></td><td><b>CIR1EN</b></td><td><b>Buffer Mode</b></td></tr>
<tr><td>1</td><td>Don’t Care</td><td>Don’t Care</td><td>Ping-Pong</td></tr>
<tr><td>0</td><td>1</td><td>Don’t Care</td><td>Circular Mode 2</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Circular Mode 1</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Indexed Single Buffer</td></tr>
</table>

18.4.2. Transmit Subaddress Control Word

Transmit Subaddress Control Words apply when a valid command word $T/\overline{R}$ bit equals one (transmit) and the subaddress field has a value in the range of 1 to 30 (0x1E). The descriptor Control Word defines terminal command response and interrupt behavior, and conveys activity status to the host. It is initialized by the host before terminal execution begins. If using ping-pong data buffers, Control Words should only be written when the applicable RTENA bit is set in the “Master Configuration Register 1 (0x0000)”. Failure to meet this requirement prevents automatic assertion of PPON bit 8 when PPEN bit 2 is set, and successive messages will repeatedly use the same buffer. Bits 8-11 cannot be written by the host; these bits are updated by the IP core during terminal execution, that is, when the “Master Configuration Register 1 (0x0000)” RTSTEX bit equals 1. The host can write bits 0-2 and 4-7 only when RTSTEX equals zero; bits 3,12 and 14-15 can be written anytime. This register is cleared to 0x0000 by $\overline{\text{RESET}}$ master reset. Software reset (SRST) clears just the DBAC, DPB and BCAST bits. **Following any host read cycle to the Control Word address, the DBAC bit is reset.**

	IXEOZ	IWA		MKBUSY	DBAC	DPB	BCAST	PPON	CIR2ZN3	CIR2ZN2	CIR2ZN1	CIR2ZN0	STOPP	PPEN	CIR2EN	CIR1EN	
	H	H	X	H	D1	D	D	D	H	H	H	H	H	H	H	H	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle
X	Bit is not used, may be logic 0 or 1

NOTE: ‘*Reset*’ refers to bit value following Master Reset ($\overline{\text{RESET}}$). The bit value following software reset is unchanged unless specifically indicated by an “SR” value.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	IXEQZ	R/W	0	Interrupt When Index Equals Zero. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is high, assertion of this bit enables generation of an interrupt for (a) subaddresses using indexed buffer mode when the INDX value decrements from 1 to 0, or (b) subaddresses using a circular buffer mode when the pre-determined number of messages has been transacted. If enabled, upon completion of command processing that results in index = 0, an IXEQZ interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, output signal $\overline{\text{INT}}$ is asserted, and the interrupt is registered in the Interrupt Log.
14	IWA	R/W	0	Interrupt When Accessed. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IWA bit is high, assertion of this bit enables interrupt generation when the subaddress receives any valid transmit command. If enabled, upon completion of command processing, an IWA interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, output signal $\overline{\text{INT}}$ is asserted, and the interrupt is registered in the Interrupt Log.
13	-----	-----	0	Not Used
12	MKBUSY	R/W	0	Make Busy. The host asserts the MKBUSY bit to respond with Busy status for commands to this transmit subaddress. This bit is an alternative to globally applying Busy status for all valid commands, enabled from the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”. See that register description for additional information. When Busy is asserted, data words are not transmitted and the DPB bit does not toggle after message completion.
11	DBAC	R	0 SR = 0	Descriptor Block Accessed. Internal IP core logic asserts the DBAC bit upon completion of message processing. The host may poll this bit to detect subaddress activity, instead of using host interrupts. This bit is reset to logic zero by $\overline{\text{RESET}}$ master reset, SRST software reset or a read cycle to this memory address.
10	DPB	R	0 SR = 0	Data Pointer B. This status bit is maintained by the IP core and only applies in ping-pong buffer mode. This bit indicates the buffer to be used for the next occurring transmit command to this subaddress. When the DPB bit is logic 0, the next message will use Data Pointer A; when DPB is logic 1, the next message uses Data Pointer B. In ping-pong buffer mode, the bit is inverted after each error-free message completion. **To also ensure the DPB bit is not altered after illegal commands or messages ending with Busy status, the DPBTOFF bit should be set in the “Extended Configuration Register (0x004D)” on page 46.** This ensures unsuccessful messages are not stored in the data buffer and are overwritten by subsequent successful messages. (see also “Ping-Pong Enable / Disable Handshake” on page 171). The DBP bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset; therefore the first message received after either reset will use Buffer A. This bit is “don’t care” for indexed single-buffer mode or either circular buffer mode.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
9	BCAST	R	0 SR = 0	Broadcast Received. The IP core sets this bit when a broadcast-transmit command is received for this subaddress. Because non-mode broadcast-transmit commands are always illegal, the assertion of this bit in the Control Word by the IP core indicates an illegal command was received. Terminal response varies, depending on whether or not illegal command detection applies (any bits set in Illegalization Table). This bit has no function if the BCSTINV bit is asserted in the “Remote Terminal Configuration Register (0x0017)”; in this case commands to RT address 31 are not recognized as valid by the IP core. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset.
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is controlled by the IP core and should not be written by the host. It only applies if PPEN bit 2 was initialized to logic one by the host after reset, enabling ping-pong buffer mode for this subaddress. The RT asserts this bit when it recognizes ping-pong is active for this subaddress. Before loading the transmit data buffer for this subaddress, the host can ask the RT to temporarily disable ping-pong by asserting STOPP bit 3. The RT acknowledges ping-pong is disabled by negating PPON. The host can safely load the buffer without data collision while PPON is negated. After buffer servicing, the host asks the RT to re-enable ping-pong by negating STOPP bit 3. The RT acknowledges ping-pong is re-enabled by asserting PPON. If PPEN bit 2 is high and PPON bit 8 is low when new commands arrive for this subaddress, ping-pong is disabled. Each new message transmits data from the same buffer, specified by DPB bit 10, and the DPB bit does not toggle after command completion.
7-4	CIR2ZN	R/W	0	Circular Mode 2 Zero Number. Used only in circular buffer mode 2, this 4-bit field is initialized with the number of trailing zeros in the initialized MIBA address. This is explained in Section 19.6, which fully describes circular buffer mode 2.
3	STOPP	R/W	0	Stop Ping-Pong Request. The host asserts this bit to suspend ping-pong buffering for this subaddress. The host resets this bit to ask the RT to re-enable ping-pong. The RT confirms recognition of ping-pong enable or disable status by writing PPON bit 8. Refer to Section 19.3, which describes ping-pong mode in detail.

Bit No.	Mnemonic	R/W	Reset	Function
2 1 0	PPEN CIR2EN CIR1EN	R/W R/W R/W	0 0 0	Ping-Pong, Circular Buffer Mode 2 or Circular Buffer Mode 1 Enable. The PPEN, CIR2EN and CIR1EN bits are initialized by the host to select buffer mode. The table below summarizes how buffer mode selection is encoded. In the case of ping-pong, the host initializes the PPEN bit to logic one after reset to enable ping-pong buffering for this subaddress. The host asserts STOPP bit 3 to ask the IP core to temporarily disable ping-pong. Negating the STOPP bit asks the IP core to re-enable ping-pong. The IP core confirms ping-pong enable or disable state changes by writing the PPON bit. PPEN bit 2 should only be initialized or otherwise written when the RTENA bit is also set in the “Master Configuration Register 1 (0x0000)”.

PPEN	CIR2EN	CIR1EN	Buffer Mode
1	Don’t Care	Don’t Care	Ping-Pong
0	1	Don’t Care	Circular Mode 2
0	0	1	Circular Mode 1
0	0	0	Indexed Single Buffer

18.4.3. Data Buffer Options for Mode Code Commands

Data buffer options for mode code commands differ from options offered for subaddress commands. Mode commands cannot use either circular data buffer method, but may use double (ping-pong) buffering or single (indexed) buffering. Single message Index mode (INDX = 0) is suitable in many applications (see Section 19.4.1). An alternative called **Simplified Mode Command Processing** (SMCP) may be globally applied for all mode code commands (see Section 20.5).

To use single (indexed) buffer or double (ping-pong) buffer for mode commands, the SMCP bit in the “Remote Terminal Configuration Register (0x0017)” is logic 0. The Control Word PPEN bit for each mode command determines whether ping-pong or indexed buffering is used.

To use Simplified Mode Command Processing, the SMCP bit in the “Remote Terminal Configuration Register (0x0017)” is set to logic 1. The Control Word PPEN bit for mode commands is “don’t care” (no longer specifies index or ping-pong buffer mode) because Simplified Mode Command Processing stores mode command data and message information words directly within each mode command’s redefined Descriptor Table block. When SMCP is enabled, mode code command descriptor blocks (in the Descriptor Table) do not contain data pointers to reserved buffers elsewhere in the shared RAM. Instead, each 4-word descriptor block itself contains the message information word, the time-tag word and the data word transacted for each mode command (for mode codes 16-31 decimal).

When Simplified Mode Command Processing is used, the range of active bits is reduced in each receive or transmit mode command Control Word. Interrupt control and response is not affected by the SMCP option. Simplified Mode Command Processing is fully presented in the later data sheet section 20.5.

18.4.4. Receive Mode Control Word

Receive Mode Control Words apply when the command word $T/\overline{R}$ bit equals zero (receive) and the subaddress field has a value of 0 or 31 (0x1F). The descriptor Control Word defines terminal command response and interrupt behavior, and conveys activity status to the host. It is initialized by the host before terminal execution begins. If using ping-pong data buffers, Control Words should only be written when the RTENA bit is set in the “Master Configuration Register

1 (0x0000)". Failure to meet this requirement prevents automatic assertion of PPON bit 8 when PPEN bit 2 is set, and successive messages will repeatedly use the same buffer. Bits 8-11 cannot be written by the host; these bits are updated by the IP core during terminal execution, that is, when the "Master Configuration Register 1 (0x0000)" RTSTEX bit equals 1. The host can write bit 2 only when RTSTEX equals zero; bits 3 and 12-15 can be written anytime. This register is cleared to 0x0000 by $\overline{\text{RESET}}$ master reset. Software reset (SRST) clears just the DBAC, DPB and BCAST bits. **Following any read cycle to the Control Word address, the DBAC bit is reset.**

When single-message indexed buffering or ping-pong buffering is used instead of SMCP (Simplified Mode Code Processing), the receive mode Control Word looks like this:

	IXEQZ	IWA	IBR	MKBUSY	DBAC	DPB	BCAST	PPON					STOPP	PPEN			
	H	H	H	H	D1	D	D	D	X	X	X	X	H	H	X	X	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle
X	Bit is not used, may be logic 0 or 1

When SMCP applies, the number of active mode Control Word bits is reduced:

	IXEQZ	IWA	IBR	MKBUSY	DBAC		BCAST	PPON									
	H	H	H	H	D1	X	D	D	X	X	X	X	X	X	X	X	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle
X	Bit is not used, may be logic 0 or 1

NOTE: '*Reset*' refers to bit value following Master Reset ($\overline{\text{RESET}}$). The bit value following software reset is unchanged unless specifically indicated by an "SR" value.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	IXEQZ	R/W	0	Interrupt When Index Equals Zero. If the "Remote Terminal (RT) Interrupt Enable Register (0x0012)" IXEQZ bit is high, assertion of this bit enables generation of an interrupt for mode code commands using indexed buffer mode when the INDX value decrements from 1 to 0. Upon completion of command processing that results in INDX = 0, when IXEQZ interrupts are enabled, an IXEQZ interrupt is entered in the "Remote Terminal (RT) Pending Interrupt Register (0x0009)", the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log.
14	IWA	R/W	0	Interrupt When Accessed. If the "Remote Terminal (RT) Interrupt Enable Register (0x0012)" IWA bit is high, assertion of this bit enables interrupt generation at each instance of a valid mode code command. Upon completion of command processing, when IWA interrupts are enabled, an IWA interrupt is entered in the "Remote Terminal (RT) Pending Interrupt Register (0x0009)", the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13	IBR	R/W	0	Interrupt Broadcast Received. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IBR bit is high, assertion of this bit enables interrupt generation at each instance of a valid broadcast receive mode code command. Upon completion of command processing, when IBR interrupts are enabled, an IBR interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log. This bit has no function if the BCSTINV bit is high in the “Remote Terminal Configuration Register (0x0017)”. In this case, commands to RT address 31 are not recognized as valid by the IP core.
12	MKBUSY	R/W	0	Make Busy. The host asserts the MKBUSY bit to respond with Busy status for commands to this mode code. This bit is an alternative to globally applying Busy status for all valid commands, enabled from the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”. See that register description for additional information. When Busy is asserted, mode data words received with MC16-MC31 are not stored and the DPB bit does not toggle after message completion.
11	DBAC	R	0 SR = 0	Descriptor Block Accessed. Internal IP core logic asserts the DBAC bit upon completion of message processing. The host may poll this bit to detect mode command activity, instead of using host interrupts. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset, SRST software reset or a read cycle to this memory address.
10	DPB	R	0 SR = 0	Data Pointer B. This status bit is maintained by the IP core and only applies for mode commands using ping-pong buffer mode. This bit indicates the buffer to be used for the next occurring mode command. When the DPB bit is logic 0, the next message will use Data Pointer A; when DPB is logic 1, the next message uses Data Pointer B. In ping-pong buffer mode, the bit is inverted after each error-free message completion. **To also ensure the DPB bit is not altered after illegal commands or messages ending with Busy status, the DPBTOFF bit should be set in the “Extended Configuration Register (0x004D)” on page 46.** This ensures unsuccessful messages are not stored in the data buffer and are overwritten by subsequent successful messages. (see also “Ping-Pong Enable / Disable Handshake” on page 171). The DBP bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset; therefore the first message received after either reset will use Buffer A. This bit is “don’t care” for indexed single-buffer mode.
9	BCAST	R	0 SR = 0	Broadcast Received. IP Core logic sets this bit when a valid broadcast mode command is received having T/$\overline{\text{R}}$ bit = 0. This bit has no function if the BCSTINV bit is asserted in the “Remote Terminal Configuration Register (0x0017)”. In this case, RT address 31 commands are not recognized as valid by the IP core. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is read only and only applies for mode commands using ping-pong mode (PPEN bit 2 was initialized to logic 1 by the host after reset). The IP core asserts this bit when it recognizes ping-pong is active for this mode code. Before off-loading the receive data buffer for this mode code, the host can ask the IP core to temporarily disable ping-pong by asserting STOPP bit 3. The IP core acknowledges ping-pong is disabled by negating PPON. The host can safely load or off-load the buffer without data collision while PPON is negated. After buffer servicing, the host asks the IP core to re-enable ping-pong by negating STOPP bit 3. The IP core acknowledges ping-pong is re-enabled by asserting PPON. If PPEN bit 2 is high and PPON bit 8 is low when new commands arrive for this subaddress, ping-pong is disabled. Each new message overwrites existing data in the buffer specified by DPB bit 10, and the DPB bit does not toggle after command completion.
7-4	-----	-----	0	Not Used
3	STOPP	R/W	0	Stop Ping-Pong Request. The host asserts this bit to suspend ping-pong buffering for this mode code. The host resets this bit to ask the IP core to re-enable ping-pong. The IP core confirms recognition of ping-pong enable or disable status by writing PPON bit 8.
2	PPEN	R/W	0	Ping-Pong Buffer Enable. The PPEN bit is initialized by the host to select buffer mode. If this bit is high, ping-pong buffering is selected. If this bit is low, indexed single buffering is selected. After reset, the host initializes this bit to logic 1 to enable ping-pong buffering for this mode code. The host asserts STOPP bit 3 to ask the IP core to temporarily disable ping-pong. Negating the STOPP bit asks the IP core to re-enable ping-pong. The IP core confirms ping-pong enable or disable state changes by writing the PPON bit 8. PPEN bit 2 should only be initialized or otherwise written when the RTENA bit is also set in the “Master Configuration Register 1 (0x0000)”.
1,0	-----	-----	0	Not Used.

18.4.5. Transmit Mode Control Word

Transmit Mode Control Words apply when the command word T/$\overline{\text{R}}$ bit equals one (transmit) and the subaddress field has a value of 0 or 31 (0x1F). The descriptor Control Word defines terminal command response and interrupt behavior, and conveys activity status to the host. It is initialized by the host before terminal execution begins. If using ping-pong data buffers, Control Words should only be written when the RTENA bit is also set in the "Master Configuration Register 1 (0x0000)". Failure to meet this requirement prevents automatic assertion of PPON bit 8 when PPEN bit 2 is set, and successive messages will repeatedly use the same buffer. Bits 8-11 cannot be written by the host; these bits are updated by the IP core during terminal execution, that is, when the "Master Configuration Register 1 (0x0000)" RTSTEX bit equals 1. The host can write bit 2 only when RTSTEX equals zero; bits 3 and 12-15 can be written anytime. This register is cleared to 0x0000 by $\overline{\text{RESET}}$ master reset. Software reset (SRST) clears just the DBAC, DPB and BCAST bits. **Following any read cycle to the Control Word address, the DBAC bit is reset.**

When single-message indexed buffering or ping-pong buffering is used instead of SMCP (Simplified Mode Code Processing), the transmit mode Control Word looks like this:

	IXEQZ	IWA	IBR	MKBUSY	DBAC	DPB	BCAST	PPON					STOPP	PPEN			
	H	H	H	H	D1	D	D	D	X	X	X	X	H	H	X	X	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle
X	Bit is not used, may be logic 0 or 1

When SMCP applies, the number of active mode Control Word bits is reduced:

	IXEQZ	IWA	IBR	MKBUSY	DBAC		BCAST	PPON									
	H	H	H	H	D1	X	D	D	X	X	X	X	X	X	X	X	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

H	Bit maintained by host
D	Bit maintained by IP Core
D1	Bit set by IP Core, reset by host read cycle
X	Bit is not used, may be logic 0 or 1

NOTE: '*Reset*' refers to bit value following Master Reset ($\overline{\text{RESET}}$). The bit value following software reset is unchanged unless specifically indicated by an "SR" value.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
15	IXEQZ	R/W	0	Interrupt When Index Equals Zero. If the "Remote Terminal (RT) Interrupt Enable Register (0x0012)" IXEQZ bit is high, assertion of this bit enables generation of an interrupt for mode code commands using indexed buffer mode when the INDX value decrements from 1 to 0. Upon completion of command processing that results in INDX = 0, when IXEQZ interrupts are enabled, an IXEQZ interrupt is entered in the "Remote Terminal (RT) Pending Interrupt Register (0x0009)", the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
14	IWA	R/W	0	Interrupt When Accessed. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IWA bit is high, assertion of this bit enables interrupt generation at each instance of a valid mode code command. Upon completion of command processing, when IWA interrupts are enabled, an IWA interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log.
13	IBR	R/W	0	Interrupt Broadcast Received. If the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IBR bit is high, assertion of this bit enables interrupt generation at each instance of a valid broadcast transmit mode code command. Upon completion of command processing, when IBR interrupts are enabled, an IBR interrupt is entered in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”, the $\overline{\text{INT}}$ output signal is asserted, and the interrupt is registered in the Interrupt Log. This bit has no function if the BCSTINV bit is high in the “Remote Terminal Configuration Register (0x0017)”. In this case, commands to RT address 31 are not recognized as valid by the IP core.
12	MKBUSY	R/W	0	Make Busy. The host asserts the MKBUSY bit to respond with Busy status for commands to this mode code. This bit is an alternative to globally applying Busy status for all valid commands, enabled from the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”. See that register description for additional information. When Busy is asserted, mode data words are not transmitted with MC16-MC31, and the DPB bit does not toggle after message completion. The MKBUSY bit is not heeded if set in the Control Word for mode code command MC8 “reset remote terminal”. For this command only, Busy is inhibited for the status response transmitted before the reset process begins.
11	DBAC	R	0 SR = 0	Descriptor Block Accessed. Internal IP core logic asserts the DBAC bit upon completion of message processing. The host may poll this bit to detect mode command activity, instead of using host interrupts. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset, SRST software reset or a read cycle to this memory address.
10	DPB	R	0 SR = 0	Data Pointer B. This status bit is maintained by the IP core and only applies for mode commands using ping-pong buffer mode. This bit indicates the buffer to be used for the next occurring mode command. When the DPB bit is logic 0, the next message will use Data Pointer A; when DPB is logic 1, the next message uses Data Pointer B. In ping-pong buffer mode, the bit is inverted after each error-free message completion. **To also ensure the DPB bit is not altered after illegal commands or messages ending with Busy status, the DPBTOFF bit should be set in the “Extended Configuration Register (0x004D)” on page 46.** This ensures unsuccessful messages are not stored in the data buffer and are overwritten by subsequent successful messages. (see also “Ping-Pong Enable / Disable Handshake” on page 171). The DBP bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset; therefore the first message received after either reset will use Buffer A. This bit is “don’t care” for indexed single-buffer mode.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
9	BCAST	R	0 SR = 0	Broadcast Received. IP Core logic sets this bit when a valid broadcast mode command is received having T/$\overline{R}$ bit = 1. This bit has no function if the BCSTINV bit is asserted in the “Remote Terminal Configuration Register (0x0017)”. In this case, RT address 31 commands are not recognized as valid by the IP core. This bit is reset to logic 0 by $\overline{\text{RESET}}$ master reset or SRST software reset.
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is read only and only applies for mode commands using ping-pong mode (PPEN bit 2 was initialized to logic 1 by the host after reset). The IP core asserts this bit when it recognizes ping-pong is active for this mode code. Before loading the transmit data buffer for this mode code, the host can ask the IP core to temporarily disable ping-pong by asserting STOPP bit 3. The IP core acknowledges ping-pong is disabled by negating PPON. The host can safely load or off-load the buffer without data collision while PPON is negated. After buffer servicing, the host asks the IP core to re-enable ping-pong by negating STOPP bit 3. The IP core acknowledges ping-pong is re-enabled by asserting PPON. If PPEN bit 2 is asserted and PPON bit 8 is negated when a new command arrives for this mode code, ping-pong disable handshake is in effect: The IP core applies single-buffer index mode using Data Pointer A or Data Pointer B, per DPB bit 10. The DPB bit does not toggle after command completion.
7-4	-----	-----	0	Not Used
3	STOPP	R/W	0	Stop Ping-Pong Request. The host asserts this bit to suspend ping-pong buffering for this mode code. The host resets this bit to ask the IP core to re-enable ping-pong. The IP core confirms recognition of ping-pong enable or disable status by writing PPON bit 8.
2	PPEN	R/W	0	Ping-Pong Buffer Enable. The PPEN bit is initialized by the host to select buffer mode. If this bit is high, ping-pong buffering is selected. If this bit is low, indexed single buffering is selected. After reset, the host initializes this bit to logic 1 to enable ping-pong buffering for this mode code. The host asserts STOPP bit 3 to ask the IP core to temporarily disable ping-pong. Negating the STOPP bit asks the IP core to re-enable ping-pong. The IP core confirms ping-pong enable or disable state changes by writing the PPON bit 8. PPEN bit 2 should only be initialized or otherwise written when the RTENA bit is also set in the “Master Configuration Register 1 (0x0000)”.
1,0	-----	-----	0	Not Used.

19. REMOTE TERMINAL MESSAGE DATA BUFFERS

The memory structures described up to this point comprise not more than 2K words of the lower memory address space. The remaining memory is allocated by the host for message data storage, to fulfill application requirements. This section describes the remaining data structures in shared RAM that control (and result from) command processing.

By initializing the RT Descriptor Table, the host allocates memory space for storing data for each subaddress used in the Remote Terminal application. Each legal Receive Subaddress and each legal Transmit Subaddress are usually assigned unique buffer memory spaces. (Exception: To comply with the requirements for MIL-STD-1553 data wrap-around, it is convenient to assign the data wrap-around subaddress to use the same buffer space for both receive and transmit commands.)

As an option, data from broadcast receive commands can be stored separately from data resulting from non-broadcast receive commands. Each subaddress buffer can use any of four data storage methods offered.

Subaddress (non-mode) commands are transacted with one to 32 data words. These are stored in a data buffer in shared RAM. For receive commands, the IP core stores data received during message processing in the shared RAM buffer. Later, the host retrieves these data words from the buffer. In the case of transmit commands, the host has previously stored transmit data words in the transmit subaddress buffer. The IP core retrieves these data words for transmission while processing the transmit command.

For each complete message processed, the message data stored in the buffer is comprised of these elements:

1. Message Information Word.
2. Time-Tag Word.
3. One to 32 Data Words transmitted or received during message transaction (except no data word for mode code commands 0 - 15 decimal).

The Message Information word and Time-Tag word are generated by the IP core and stored in assigned buffer space to aid the host in further message processing. The Message Information word contains message type, word count and message error information. The 16-bit Time-Tag word contains the value in the IP core internal Time-Tag counter when the command is validated.

The host initializes the Descriptor Table entry for each subaddress or mode command to select one of four data buffering methods.

1. **Indexed (Single Buffer) Method (see 19.4).**

   A predetermined number of messages (N) is transacted using a single data buffer in shared RAM. Several host interrupt options are offered, including host interrupt when all N messages are successfully completed. This method also supports single-message mode when N is purposely initialized to zero.

2. **Double (or Ping-Pong) Buffer Method (see 19.3).**

   Successive messages alternate between two 34-word data buffers in shared RAM. Several host interrupt options are offered.

3. **Circular Buffer Mode 1 (see 19.5).**

   Buffer boundaries determine when the bulk transfer is complete. Message information and time-tag words are stored in the same buffer with data words. Several host interrupt options are offered, including host interrupt when the allocated data buffer is full.

4. **Circular Buffer Mode 2 (see 19.6).**

   The number of messages transacted defines bulk transfer progress. Message data words are stored contiguously in one buffer while message information and time-tag words are stored in a separate buffer. Several host interrupt options are offered, including host interrupt when all N messages are completed.

The data buffer options are summarized in Table 18.

Simplified Mode Command Processing.

This is a global option that applies for all mode code commands, when enabled. Mode commands have either one data word, or no data word. Instead of using data buffers for storing this limited mode command data, the message data is stored directly within the Descriptor Table. This option for mode commands is described in section 20.5.

Broadcast Data Separation

When the NOTICE2 option is enabled, data words resulting from broadcast receive commands will be stored separately from data resulting from non-broadcast receive commands when using indexed or ping-pong buffer modes. When NOTICE2 applies, all subaddresses using indexed or ping-pong modes must have an assigned 34-word broadcast data buffer in addition to the primary buffers listed above.

When using circular buffers with Notice 2, the user is responsible for separating buffer data stored by broadcast and non-broadcast messages. To make this possible, an option is offered that provides a BCAST status bit in the data buffer Message Information Word (MIW), saved in the data buffer each time a message is received. By examining the MIWs stored in the circular buffer, the host can differentiate broadcast from non-broadcast messages. See description of option bit 3 in the “Extended Configuration Register (0x004D)”, as well as Section 20.1 below.

Table 18. Summary of Data Buffer Modes.

Buffer Mode	Data Buffer(s) Number and Size	Message Info Word	Suitable for Mode Codes?	Primary Application
Indexed	One. Host defines size for N messages	Stored in same buffer as data	Yes, only single message mode	For transacting N (multiple) messages with optional host interrupt when done
Ping-Pong	Two 34-word buffers, one message each	Stored in same buffer as data	Yes	For transacting single messages, alternating between A and B buffers
Circular 1	One. Host defines size for N words	Stored in same buffer as data	No	For transacting messages until buffer is full / empty, optional interrupt when done
Circular 2	One. Host defines size for N messages, plus Msg Info Block	Stored in separate buffer (Msg info block)	No	For transacting N (multiple) messages with optional host interrupt when done. Data buffer holds contiguous pure data.

19.1. Subaddress Message Information Words

19.1.1. Receive Subaddress Command

For receive subaddress commands, the IP core stores the received data words plus two additional words. The IP core adds a receive subaddress Message Information Word and a Time-Tag Word to the received data words. The IP core stores the Message Information and Time-Tag words ahead of the data words associated with the receive command, as shown below. If message error occurs, the RT stores only the receive subaddress Message Information Word and Time-Tag Word. Once a message error is detected, the IP core sets the MERR bit in the receive subaddress Message Information Word. When this occurs, all data words are considered invalid. Whenever the receive subaddress Message Information Word MERR bit is set, the host should disregard the record’s data word(s).

Here is an example data structure for a 3-word receive command. Notice that the receive subaddress Data Pointer points to the data structure starting address, not the first data word. The data pointer is located in the receive subaddress command’s Descriptor Block, fully described later:

	Data Buffer Hex Address	Word Description	IP Core Writes Word ...
Data pointer equals 0x1500 →	0x1500	Message Information Word	After message completion
	0x1501	Time-Tag Word	“ “ “
	0x1502	Data Word 1	After message completion (See **Note**)
	0x1503	Data Word 2	“ “ “ “ “
	0x1504	Data Word 3	“ “ “ “ “

Note: The data words are written after message completion when the RT Configuration Register bit TRXDB is 1, otherwise written when received.

	TMOERR	IWDERR	GAP/BCAST	WCTERR	SYNERR	MERR	WASBSY	ILCMD	TXRTERR	RTRT	BUSID	WC4	WC3	WC2	WC1	WC0	
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

The following bits comprise the receive subaddress Message Information Word:

Bit No.	*Mnemonic*	*Function*
15	TMOERR	Time-Out Error. This bit is asserted for RT-RT receive messages when the transmitting terminal fails to start its status word and data transmission before time-out occurs, per RTTO[1:0] bits in the “Remote Terminal Configuration Register (0x0017)”.
14	IWDERR	Invalid Word Error. Assertion of this bit indicates Manchester error or parity error was observed in a received data word.
13	GAP/ BCAST	Gap Error / Broadcast Flag. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Gap Error flag. Assertion of this bit indicates bus activity was detected immediately after the last expected receive data word or that a gap occurred before all expected data words were received. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Broadcast flag, asserted when the received message was broadcast.
12	WCTERR	Word Count Error. This bit is asserted if command is received with less data words than the command word specifies. For example, a receive command for three data words is received with two contiguous data words.
11	SYNERR	Sync Error. This bit is asserted when an incorrect (command/status) sync type occurs in received data words.

Bit No.	*Mnemonic*	*Function*
10	MERR	Message Error. This bit is asserted when message error status change occurs during command processing. See bits 7 and 11-15 for details.
9	WASBSY	Was Busy. This bit is asserted when the terminal responds to the receive command with BUSY status, due to global BUSY bit set in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”, or command-specific MKBUSY bit set in the descriptor table Control Word. Received data words were buffered normally.
8	ILCMD	Illegal Command Received. This bit is asserted when the Illegalization Table bit corresponding to the received command is logic 1. The Illegalization Table should only contain nonzero values when “illegal command detection” is being applied. See Section 18.2 for further information.
7	TXRTERR	RT-RT Transmit Remote Terminal Error. This bit is set when the terminal decodes a valid RT-RT receive command, but one of four potential errors is detected in the second command word, CW2: (1) CW2 is addressed to broadcast address RT31. (2) the CW2 T/$\overline{\text{R}}$ bit equals 0, (3) the CW2 subaddress is a mode command indicator, 00000 or 11111, or (4) CW2 has same non-broadcast terminal address as receive command word CW1. The TXRTERR bit is also set when status word received from the transmitting terminal is invalid (e.g., parity error) or bits 15:11 in the status word reflect the wrong RT address (does not match CW2).
6	RTRT	Remote Terminal to Remote Terminal Transfer. Assertion of this bit indicates the receive command was an error-free RT-to-RT transfer.
5	BUSID	Bus Identification. If this bit equals zero, message was transacted on Bus A. If bit equals one, it was transacted on Bus B.
4-0	WC4:0	Word Count. This 5-bit field contains the word count extracted from the command word. Zero indicates 32 words.

19.1.2. Transmit Subaddress Command

The external host is responsible for organizing the data packet (i.e., storing N data words) in shared RAM and initializing the applicable data pointer. The host must allocate two memory locations at the starting address of the data record for IP core storage of the transmit subaddress Message Information Word and Time-Tag Word.

Here is an example data structure for a 3-word transmit command. Notice that the Data Pointer points to the data structure starting address, not the first data word. The data pointer is located in the transmit subaddress command’s Descriptor Block.

		Data Buffer Hex Address	Word Description	Word is Written By ...
Data pointer equals 0x1500	→	0x1500	Message Information Word	IP Core, after message completion
		0x1501	Time-Tag Word	“ “ “ “
		0x1502	Data Word 1	Host, prior to terminal’s data transmit
		0x1503	Data Word 2	“ “ “ “ “ “
		0x1504	Data Word 3	“ “ “ “ “ “

			GAP/BCAST	WCTERR		MERR	WASBSY	ILCMD		RTRT	BUSID	W4	W3	W2	W1	W0	
	X	X			X				X								
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	LSB

The following bits comprise the transmit subaddress Message Information Word.

Bit No.	*Mnemonic*	*Function*
15,14	-----	Not Used.
13	GAP/ BCAST	Gap Error / Broadcast Flag. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Gap Error flag. Assertion of this bit indicates bus activity was detected immediately after the transmit command word, when a gap was expected. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Broadcast flag, asserted when the received message was broadcast.
12	WCTERR	Word Count Error. This bit is asserted if command is received with unexpected data word(s).
11	-----	Not Used.
10	MERR	Message Error. This bit is asserted when message error status change occurs during command processing. See bits 12 and 13 for details.
9	WASBSY	Was Busy Status. This bit is asserted when the terminal responds to the transmit command with BUSY status, due to global BUSY bit set in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”, or command-specific MKBUSY bit set in the descriptor table Control Word. No data words were transmitted.
8	ILCMD	Illegal Command Received. This bit is asserted when the Illegalization Table bit corresponding to the received command equals one. The Illegalization Table should only contain nonzero values when “illegal command detection” is being applied. See Section 18.2 for further information.
7	-----	Not Used.
6	RTRT	Remote Terminal to Remote Terminal Transfer. Assertion of this bit indicates the transmit command was an error-free RT-to-RT transfer.

Bit No.	*Mnemonic*	*Function*
5	BUSID	Bus Identification. If this bit equals zero, message was transacted on Bus A. If bit equals one, it was transacted on Bus B.
4-0	WC4:0	Word Count. This 5-bit field contains the word count extracted from the command word. Zero indicates 32 words.

19.2. Mode Command Message Information Words

Mode command data structures in shared RAM are similar to those for subaddresses. Mode codes 0 through 15 (0x0F) do not have an associated data word, so data structures for these mode code values have just a Message Information Word and Time-Tag Word. The Message Information Word is stored at the memory address specified by the descriptor table Data Pointer. Mode codes 16 through 31 (0x10 through 0x1F) have one associated data word. The Message Information Word is stored at the memory address specified by the descriptor table Data Pointer, and the Time-Tag Word is stored in the following location. The data word is stored at the memory address specified by the Data Pointer plus two locations.

19.2.1. Receive Mode Command

The receive mode command data structure contains a Message Information Word, a Time-Tag Word and may contain one Data Word. If a receive mode command has a data word, the IP core may apply the data as defined by MIL-STD-1553, plus store the received single mode data word at the address specified by the Data Pointer, plus two locations. Refer to the Mode Code Command Summary in Table 20.

Here is an example data structure for a receive mode command with data (mode code values 0x10 through 0x1F). Notice that the Data Pointer points to the data structure starting address, not the mode data word. The data pointer is located in the receive mode command's Descriptor Block, fully described later:

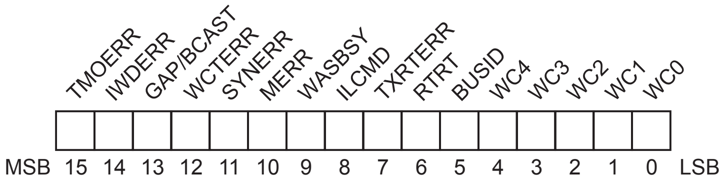
		Data Buffer Hex Address	**Word Description**	**Word is Written By ...**
Data pointer equals 0x1500	→	0x1500	Message Information Word	IP Core, after message completion
		0x1501	Time-Tag Word	“ “ “ “
		0x1502	Mode Data Word	“ “ “ “

Three receive mode commands with data are not defined under MIL-STD-1553B. These are MC16, MC18 and MC19 (mode codes 0x10, 0x12 and 0x13 respectively). However the IP core responds “in form” if illegal command detection is not used (corresponding bits in Illegalization Table are logic 0) **and** the UMCINV bit in the “Remote Terminal Configuration Register (0x0017)” is logic 0.

For mode code commands without data, the data structure contains only the Message Information Word and Time-Tag Word.

Here is an example data structure for a receive mode command without data (mode code values 0x00 through 0x0F). Note: None of these receive mode commands are defined under MIL-STD-1553B but the IP core responds “in form” if illegal command detection is not used (corresponding bits in Illegalization Table are logic 0) and the UMCINV bit in the “Remote Terminal Configuration Register (0x0017)” is logic 0. Notice that the data pointer points to the data structure starting address, the message information word. The data pointer is located in the receive mode command's Descriptor Block, fully described later:

		Data Buffer Hex Address	Word Description	Word is Written By ...
Data pointer equals 0x1500	→	0x1500	Message Information Word	IP Core, after message completion
		0x1501	Time-Tag Word	“ “ “ “



The following bits comprise the receive mode Message Information Word:

Bit No.	*Mnemonic*	*Function*
15	-----	Not Used.
14	IWDERR	Invalid Word Error. Assertion of this bit indicates Manchester error or parity error was observed in a received data word.
13	GAP/ BCAST	Gap Error / Broadcast Flag. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Gap Error flag. Assertion of this bit indicates bus activity was detected immediately after a received mode data word or that a gap occurred before the data word was received. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Broadcast flag, asserted when the received message was broadcast.
12	WCTERR	Word Count Error This bit is asserted if the command is received without expected mode data word, or with extra word.
11	SYNERR	Sync Error. This bit is asserted when incorrect (command/status) sync type occurs in received mode data word.
10	MERR	Message Error. This bit is asserted when message error status change occurs during command processing. See bits 11- 14 for details.
9	WASBSY	Was Busy Status. This bit is asserted when the terminal responds to the mode command with BUSY status, due to global BUSY bit set in the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”, or command-specific MKBUSY bit set in the descriptor table Control Word.

Bit No.	*Mnemonic*	*Function*
8	ILCMD	Illegal Command Received. This bit is asserted when the Illegalization Table bit corresponding to the received command equals one. The Illegalization Table should only contain nonzero values when “illegal command detection” is being applied. See Section 18.2 for further information.
7,6	-----	Not Used.
5	BUSID	Bus Identification. If this bit equals zero, message was transacted on Bus A. If bit equals one, it was transacted on Bus B.
4-0	MC4:0	Mode Code. This 5-bit field contains the mode code extracted from the command word.

19.2.2. Transmit Mode Command

The transmit mode command data structure contains a Message Information Word, a Time-Tag word and may contain one Data Word. For mode commands with associated data word (mode codes 16-31 decimal) the host is responsible for loading the Mode Command Data Table before transmit mode commands are received (e.g., Transmit Vector Word mode code). Two mode codes have internally generated data words: MC18 “Transmit Last Command” and MC19 “Transmit BIT Word”. For these, the IP core automatically transmits the data word then copies the transmitted data value to the stored data structure.

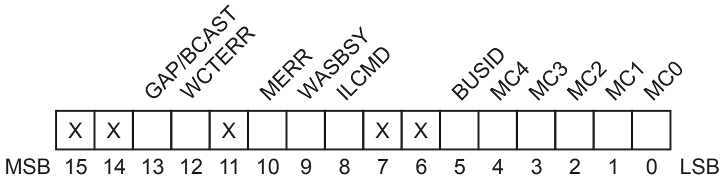
Here is an example data structure for a transmit mode command with data (mode code values 0x10 through 0x1F). This applies to MC16 “Transmit Vector Word”. Notice that the data pointer points to the data structure starting address, not the mode data word. The data pointer is located in the transmit mode command’s Descriptor Block, fully described later:

		Data Buffer Hex Address	**Word Description**	**Word is Written By ...**
Data pointer equals 0x1500	→	0x1500	Message Information Word	IP Core, after message completion
		0x1501	Time-Tag Word	“ “ “ “
		0x1502	Mode Data Word	Host, prior to terminal’s data transmit (except MC18, MC19 are written by the IP core after completion)

Three transmit mode commands with data are not defined under MIL-STD-1553B. These are MC17, MC20 and MC21 (mode codes 0x11, 0x14 and 0x15 respectively). However the IP core responds “in form” if illegal command detection is not used (corresponding bits in Illegalization Table are logic 0) and the UMCINV bit in the “Remote Terminal Configuration Register (0x0017)” is logic 0.

For mode code commands without data, the data structure contains only the Message Information Word and Time-Tag Word. Here is an example data structure for a transmit mode command without data (mode code values 0x00 through 0x0F). Again, the data pointer points to the data structure starting address. The data pointer is located in the transmit mode command’s Descriptor Block, fully described later:

		Data Buffer Hex Address	Word Description	Word is Written By ...
Data pointer equals 0x1500	→	0x1500	Message Information Word	IP Core, after message completion
		0x1501	Time-Tag Word	“ “ “ “



The following bits comprise the mode transmit Message Information Word:

Bit No.	*Mnemonic*	*Function*
15,14	-----	Not Used.
13	GAP/ BCAST	Gap Error / Broadcast Flag. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Gap Error flag. This bit is high when bus activity was detected immediately after the mode command word, when a gap was expected. When “Extended Configuration Register (0x004D)” on page 46 bit 3 is 0, this bit is a Broadcast flag, asserted when the received message was broadcast.
12	WCTERR	Word Count Error This bit is asserted if command is received with unexpected data word(s).
11	-----	Not Used.
10	MERR	Message Error. This bit is asserted when message error status change occurs during command processing. See bits 12-13 for details.
9	WASBSY	Was Busy Status. This bit is asserted when the terminal responds to the mode command with BUSY status, due to global BUSY bit set in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”, or command-specific MKBUSY bit set in the descriptor table Control Word. No mode data word was transmitted.
8	ILCMD	Illegal Command Received. This bit is asserted when the Illegalization Table bit corresponding to the received command is logic 1. The Illegalization Table should only contain nonzero values when “illegal command detection” is being applied. See Section 18.2 for further information.
7,6	-----	Not Used.
5	BUSID	Bus Identification. If this bit equals zero, message was transacted on Bus A. If bit equals one, it was transacted on Bus B.
4-0	MC4:0	Mode Code. This 5-bit field contains the mode code extracted from the command word.

19.3. Ping-Pong Data Buffering

19.3.1. Double Buffered (Ping-Pong) Mode

Ping-pong buffer mode is a method for storing message and time-tag information and data associated with messages. Each unique MIL-STD-1553 subaddress and mode code is assigned a pair of data buffers for transmit commands and a pair of data buffers for receive commands. The IP core retrieves buffer data for transmit commands, or stores buffer data for receive commands. During ping-pong operation, the IP core alternates message storage between Data Buffer A and Data Buffer B, on a message-by-message basis.

When a subaddress or mode command uses ping-pong data buffer mode, its 4-word descriptor block in the Descriptor Table is defined as follows:

Descriptor Word 1	Control Word
Descriptor Word 2	Data Pointer A
Descriptor Word 3	Data Pointer B
Descriptor Word 4	Broadcast Data Pointer

If Descriptor Word 1 is stored at memory address N, Descriptor Word 2 is stored at address N+1, and the other two words are stored at addresses N+2 and N+3.

Prior to starting terminal operation, enable ping-pong buffering for any subaddress (or mode code) by asserting the PPEN bit and negating the STOPP bit in the descriptor Control Word. When the IP core detects ping-pong is selected (PPEN = 1) and enabled (STOPP = 0), it asserts the Control Word PPON bit to confirm ping-pong is active.

During ping-pong operation, the RT determines the active data buffer at the beginning of message processing. The Control Word DPB bit indicates the data pointer to be used by the next command. DPB equals logic 0 means Data Pointer A is used next; DPB equals logic 1 means Data Pointer B is used next. For ping-pong, Data Pointers A and B are static values pointing to the first address in each buffer. At the conclusion of error-free message processing, the Control Word DPB bit is inverted so the next command “ping-pongs” to the other data buffer. Each new message to the subaddress or mode code overwrites message data and information words written previously. To assure data integrity, the DPB pointer should only toggle after completion of error-free messages. To cover the full set of conditions, set DPBTOFF bit 1 in “Extended Configuration Register (0x004D)” on page 46. When option bit DPBTOFF = 1, DPB pointer toggle is prevented after incomplete messages, illegal commands, and messages resulting in BUSY or MESSAGE ERROR status. (When option bit DPBTOFF = 0, the illegal and BUSY cases still cause DPB pointer toggle.)

Please note that a subaddress may contain both legal and illegal word counts. When DPBTOFF = 1, DPB pointer toggle only occurs for the expected (legal) word count(s).

Figure 19 is a general illustration of ping-pong buffer mode. Figure 20 shows a specific example.

The primary benefit of using the DPBTOFF = 1 option is always knowing where to find the most-recent valid data. When DPBTOFF = 1, the complemented DPB pointer always indicates the last-transacted “good” data set. For example, if DPB is logic 0, the last successful message used Data Buffer B.

(**Exception:** immediately following Master Reset, the entire memory range is cleared to zero, so neither buffer contains message data. After reset, the host typically initializes outgoing data for the first message occurring on each transmit subaddress, Buffer A. The Message Information and Time Tag Words will read 0x0000 until the first message is transacted. After reset, the first-used Buffer A for each receive subaddress will contain 0x0000 for the Message Information and Time Tag Words and all data locations, until the first message is transacted).

19.3.2. Ping-Pong Enable / Disable Handshake

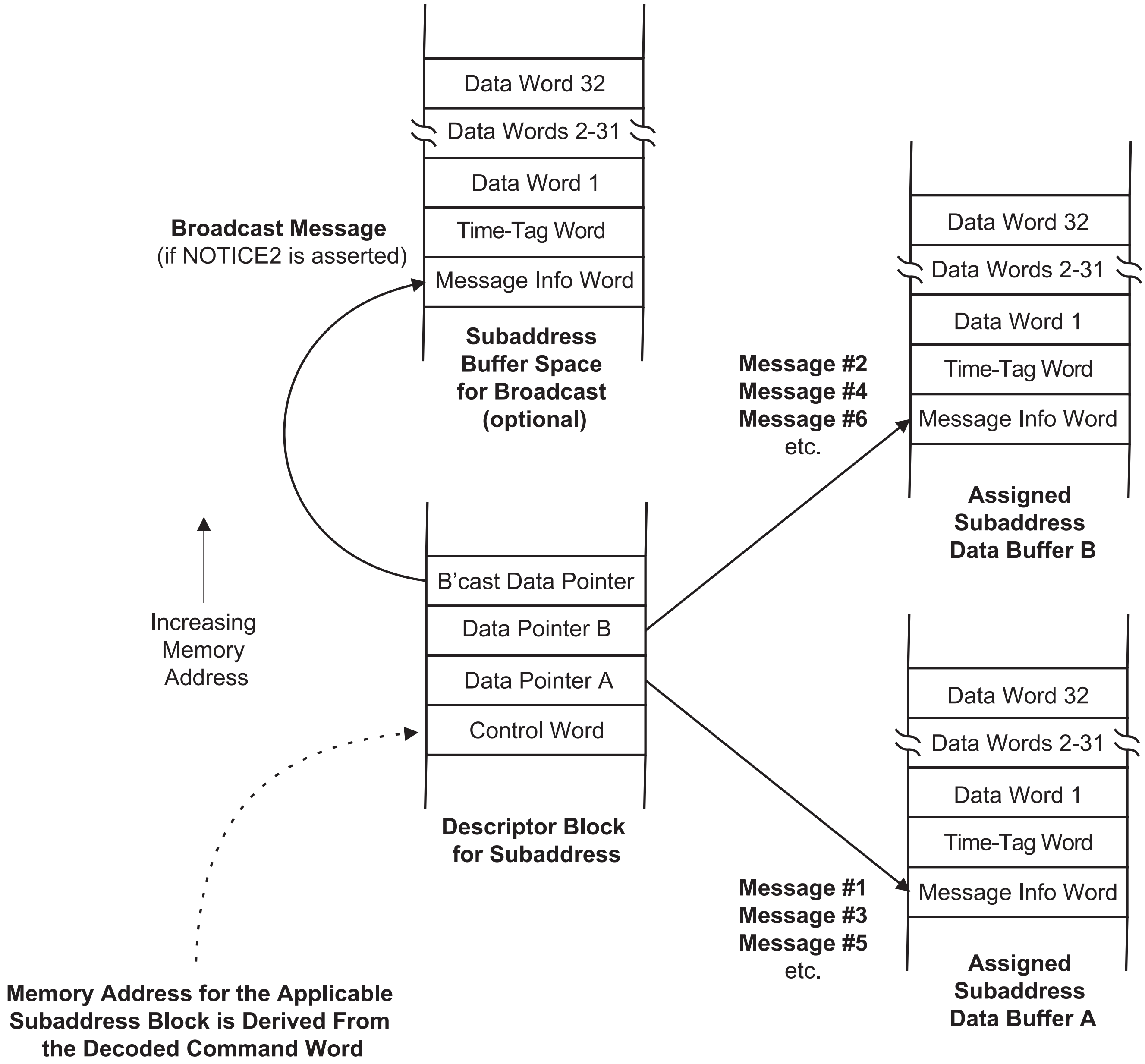
Because ping-pong messages and host buffer servicing are asynchronous, there is potential for “data collision”. Here is a data collision example: The host reads data from an earlier message while the IP core simultaneously writes new message data to the same buffer. The host reads a mix of new and old message data. Collisions can occur for both transmit and receive messages.

A handshake scheme lets the external host asynchronously service ping-pong data buffers without data collision. To off-load or load a subaddress (or mode code) buffer, the application software performs the following sequence:

a. Host asserts the Control Word STOPP bit to suspend ping-pong operation for the subaddress. When the IP core recognizes STOPP bit assertion, it negates the PPON bit to acknowledge ping-pong is disabled. While PPON remains low, the last written (or read) data buffer is protected against IP core updates. During this time, new messages use the active buffer indicated by the Control Word DPA bit. Recurring messages repeatedly use the same buffer until ping-pong resumes.

b. Host services the last-used data buffer. If the Control Word DPB bit equals logic 1, the last command used Buffer A. The host application software off-loads or loads inactive Buffer A while the remote terminal uses active Buffer B for new message(s). If the DPB bit equals logic 0, the last command used Buffer B. The host application software off-loads or loads inactive Buffer B while the remote terminal uses active Buffer A for any new messages. Each new receive message overwrites buffer contents from the last receive message. To avoid possible data loss, host buffer servicing should be timed for completion before a second message can occur.

c. Host negates the Control Word STOPP bit to resume ping-pong operation for the subaddress. When the RT recognizes the STOPP bit is reset, it sets the PPON bit to acknowledge ping-pong is again active. As long as PPON remains set, the IP core alternates between data buffers A and B for new messages.



Message processing alternates between Data Buffers A and B. Upon sucessful message completion, the DPB bit in Descriptor Control Word is updated so next message uses other buffer. Buffers are overwritten every other message.

Separate buffer for broadcast messages is optional. There is no alternate buffer for successive broadcast messages.

Figure 19. Illustration of Ping-Pong Buffer Mode

19.3.3. Broadcast Message Handling in Ping-Pong Mode

For MIL-STD-1553B Notice II compliance, a remote terminal should be capable of storing data from broadcast messages separately from non-broadcast message data. Some applications may not include this requirement. The standard does not stipulate where data separation should occur (e.g., within the RT or within the external host) so the IP core provides alternative strategies.

When the NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)” is 1 and the BCSTINV bit is 0, ping-pong mode subaddresses (or mode codes) will buffer data words from broadcast and non-broadcast messages separately. Broadcast message information and data are stored in the broadcast data buffer; non-broadcast message information and data are stored in ping-pong buffers A and B. Since there is just one broadcast data buffer, the NOTICE2 option treats broadcast messages as exceptions to normal ping-pong mode. When using the NOTICE2 option, broadcast data buffer servicing should have high priority, because a closely following broadcast message will overwrite the broadcast buffer.

Every mode command and subaddress (including transmit subaddresses) must have an assigned valid broadcast data pointer when NOTICE2 is asserted. When the NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)” is 1 and the BCSTINV bit is 0, reception of a broadcast-transmit message updates the Message Information and Time-Tag Words for the assigned broadcast buffer, but no data is transmitted on the bus. Since broadcast-transmit is not allowed, multiple transmit subaddresses may share a common “bit bucket” broadcast buffer. A two word buffer is sufficient for storing the MIW and Time-Tag Word.

When using ping-pong mode, there are two ways to handle broadcast messages, when broadcast is enabled:

Option 1 for Ping-Pong Mode Broadcast Messages:

This option isolates broadcast message information in the broadcast data buffer. If the descriptor Control Word IBR bit and “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IBR bit are both set, reception of broadcast messages generates an INT host interrupt. To prevent data loss, the broadcast data buffer must be serviced before the next broadcast message occurs. Broadcast messages do not affect non-broadcast message ping-pong; the Control Word DPB bit does not toggle after broadcast message completion.

Option 1 Setup: At initialization, host asserts the NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)” and sets the IBR (Interrupt Broadcast Received) bit in descriptor Control Word(s). The IBR bit is asserted in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”.

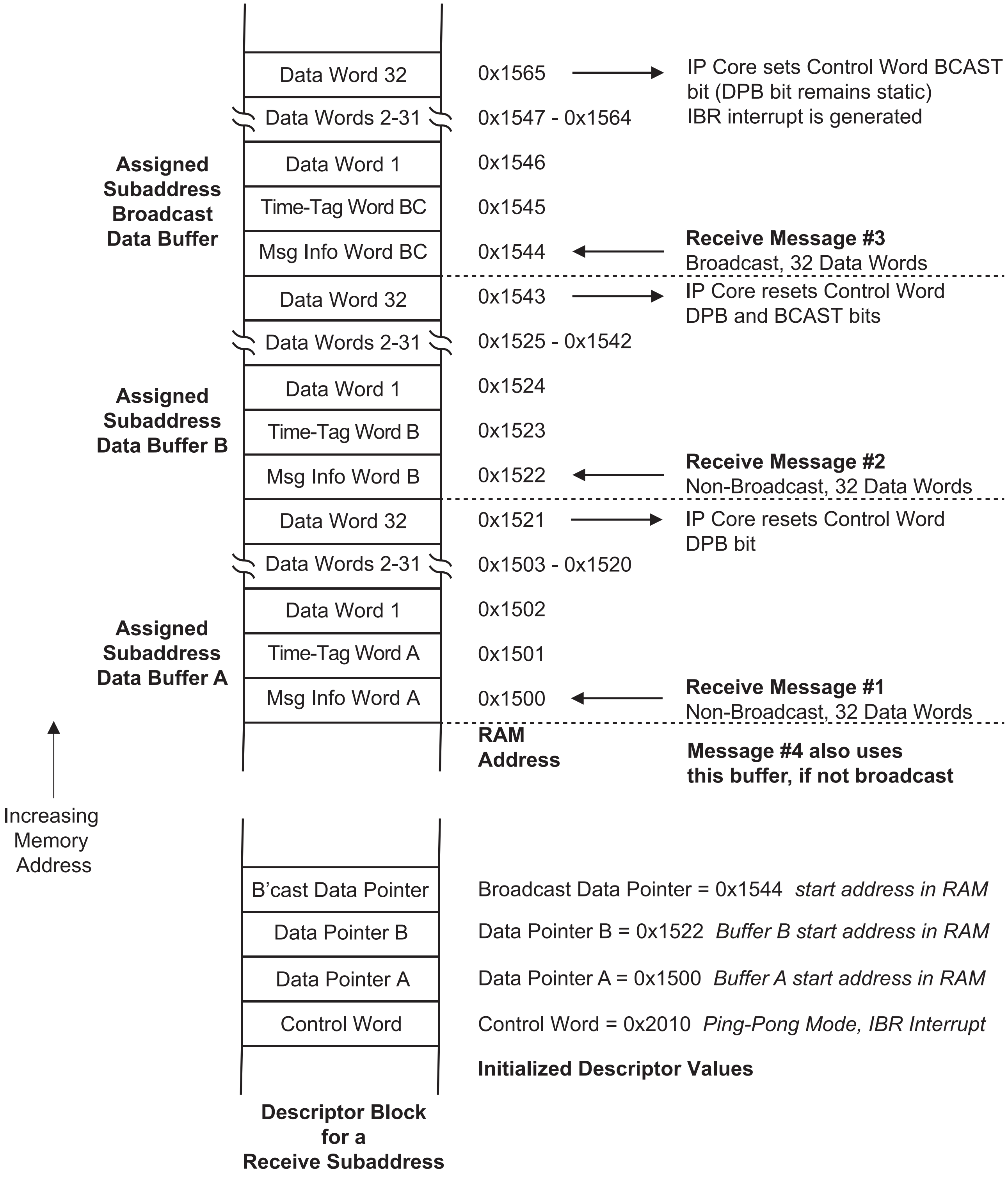
When a broadcast command is received, message information and data is stored in the broadcast data buffer and an $\overline{\text{INT}}$ interrupt is generated. The host must read the Interrupt Log to determine the originating subaddress (or mode code), then service the broadcast data buffer for that subaddress (or mode code) before another broadcast message to the same subaddress (or mode code) arrives.

Option 2 for Ping-Pong Mode Broadcast Messages:

The second alternative stores both broadcast and non-broadcast message information in the ping-pong data buffers A and B. IWA interrupts can signal arrival of any new message. The RT handles broadcast messages just like non-broadcast messages, except the Message Information Word BCAST bit is asserted to identify broadcast messages during host buffer servicing. All messages toggle the Control Word DPB bit in message post-processing. For Notice II compliance, separation of broadcast and non-broadcast data occurs within the host.

Option 2 Setup: At initialization, host negates the NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)”. If IWA interrupts are used, the host asserts the descriptor Control Word IWA (Interrupt When Accessed) bit 14 and the corresponding bit is asserted in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”. Using this option, the IBR interrupt is probably not used.

The host typically services the ping-pong data buffers A and B whenever a message is transacted. Using the setup above, this occurs whenever the subaddress IWA interrupt generates an $\overline{\text{INT}}$ interrupt output for the host. The host must read the Interrupt Log to determine the originating subaddress or mode code. The applicable data buffer is indicated by the DPB bit in the Receive Control Word. The Message Information Word BCAST bit is asserted if the message was broadcast.


Data Word 32
0x1565
IP Core sets Control Word BCAST bit (DPB bit remains static)
IBR interrupt is generated
Data Words 2-31
0x1547 - 0x1564
Assigned Subaddress Broadcast Data Buffer
Data Word 1
0x1546
Time-Tag Word BC
0x1545
Msg Info Word BC
0x1544
Receive Message #3
Broadcast, 32 Data Words
Data Word 32
0x1543
IP Core resets Control Word DPB and BCAST bits
Data Words 2-31
0x1525 - 0x1542
Assigned Subaddress Data Buffer B
Data Word 1
0x1524
Time-Tag Word B
0x1523
Msg Info Word B
0x1522
Receive Message #2
Non-Broadcast, 32 Data Words
Data Word 32
0x1521
IP Core resets Control Word DPB bit
Data Words 2-31
0x1503 - 0x1520
Assigned Subaddress Data Buffer A
Data Word 1
0x1502
Time-Tag Word A
0x1501
Msg Info Word A
0x1500
Receive Message #1
Non-Broadcast, 32 Data Words
RAM Address
Message #4 also uses this buffer, if not broadcast
Increasing Memory Address
B'cast Data Pointer
Broadcast Data Pointer = 0x1544 start address in RAM
Data Pointer B
Data Pointer B = 0x1522 Buffer B start address in RAM
Data Pointer A
Data Pointer A = 0x1500 Buffer A start address in RAM
Control Word
Control Word = 0x2010 Ping-Pong Mode, IBR Interrupt
Initialized Descriptor Values
Descriptor Block for a Receive Subaddress

Following reset (which resets Control Word DPB bit), the subaddress transacts 4 commands of 32 data words each. The NOTICE 2 option is enabled so the IP Core segregates data from broadcast and non-broadcast messages. Message #3 is a broadcast command, while the other three messages are non-broadcast. Notice that the broadcast message does not affect DPB bit, but the following message resets BCAST bit. The interspersed broadcast command does not affect alternation between Buffer A and Buffer B.

Figure 20. Ping-Pong Buffer Mode Example for a Receive Subaddress

19.4. Indexed Data Buffer Mode

Also called “single buffer mode”, indexed buffering is one method for storing message and time-tag information and data associated with messages. Buffer mode is selected for each subaddress or mode code in the Descriptor Table Control Words. Indexed mode is enabled when Control Word PPEN, CIR1EN and CIR2EN bits are all zero.

When a subaddress or mode command uses the indexed data buffer mode, its 4-word descriptor block in the Descriptor Table is defined as follows:

Descriptor Word 1	Control Word
Descriptor Word 2	Data Pointer A
Descriptor Word 3	INDX Index Word
Descriptor Word 4	Broadcast Data Pointer

If Descriptor Word 1 is stored at memory address N, Descriptor Word 2 is stored at address N+1, and the other two words are stored at addresses N+2 and N+3.

As the name implies, all message information and data is stored in a single buffer, indexed by descriptor word Data Pointer A. The descriptor Control Word DPB bit is “don’t care”. The host initializes the desired message count in descriptor INDX word. During message processing, the IP core retrieves or stores data words from the address specified by descriptor Data Pointer A, automatically incrementing the pointer address as words are read or stored. Data Pointer A is updated during command post-processing with the current buffer address unless the message index count in descriptor INDX (word 3 of descriptor block) decrements to zero upon completion of the message. Figure 21 is a general illustration of indexed single buffer mode. Figure 22 shows a specific example.

To set up a terminal subaddress to buffer multiple messages, the host writes the desired index count (INDX) to subaddress descriptor word 3. The initial INDX value ranges from zero to 3FF hex (1023) messages. The IP core decrements the INDX count each time an error-free message is transacted, and the data pointer is updated to the first memory address to be used for the next message. If INDX decrements from one to zero and Control Word IXEQZ bit 15 is asserted, the IXEQZ bit is set in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)”. If the corresponding bit in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” is asserted, an $\overline{\text{INT}}$ interrupt is generated when INDX decrements from one to zero.

INDX counter decrement does not occur if the command was illegalized or if INDX already equals zero. Once INDX equals zero, further commands will overwrite the last-written data buffer block and the data pointer value is not updated after successful message completion.

When using Index Mode with a non-zero INDX value, the host must remember the initial Data Pointer A address. The Data Pointer A word is not automatically reinitialized to the buffer start address when INDX decrements from 1 to 0.

19.4.1. Single Message Mode

When Index Mode is initialized with an INDX value of zero, the subaddress or mode code is operating in “Single Message Mode”. Here, the same data block is repeatedly over-read (for transmit data) or overwritten (for receive or broadcast data). The DPA pointer is not updated at the end of each message. The chief advantage of single message mode is simplicity. In comparison to other data buffering options, the single message buffer uses an absolute minimum amount of memory space. The IXEQZ interrupt cannot be used for this scheme (INDX is always zero) but IWA interrupts may be used. Single message mode is best suited to synchronous data transfer where the host processor can reliably read or write new message data prior to the start of the next message to the same subaddress or mode code.

19.4.2. Broadcast Message Handling in Index Mode

For MIL-STD-1553B Notice II compliance, a remote terminal should be capable of storing data from broadcast messages separately from non-broadcast message data. Some applications may not include this requirement. The standard does not stipulate where data separation should occur (e.g., within the RT or within the external host) so the IP core supports alternative strategies.

When the NOTICE2 bit is logic 1 in the “Remote Terminal Configuration Register (0x0017)”, broadcast message data is stored in a broadcast data buffer assigned for the subaddress or mode command. Each subaddress or mode command must have an assigned, valid non-zero broadcast buffer address. Non-broadcast message data is stored in Data Buffer A.

There are two ways to deal with broadcast messages in indexed buffer mode:

Option 1 for Index Mode Broadcast Messages:

The first alternative isolates broadcast message information in the broadcast data buffer. If the descriptor Control Word IBR bit and “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IBR bit are both set, reception of broadcast messages generates an $\overline{\text{INT}}$ interrupt to the host. The broadcast data buffer must be processed before another broadcast message arrives to prevent loss of data. Broadcast messages do not decrement the INDX register, and Data Pointer A is not updated in message post-processing. This scheme may be well suited for Single Message Mode (INDX = 0) when the host can reliably service either the broadcast data buffer or data buffer A before the next receive message arrives for the same subaddress (or mode code).

Option 1 Setup: At initialization, host asserts NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)” and sets the Control Word IBR (Interrupt Broadcast Received) bit for each index mode descriptor block. The IBR bit is also asserted in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”.

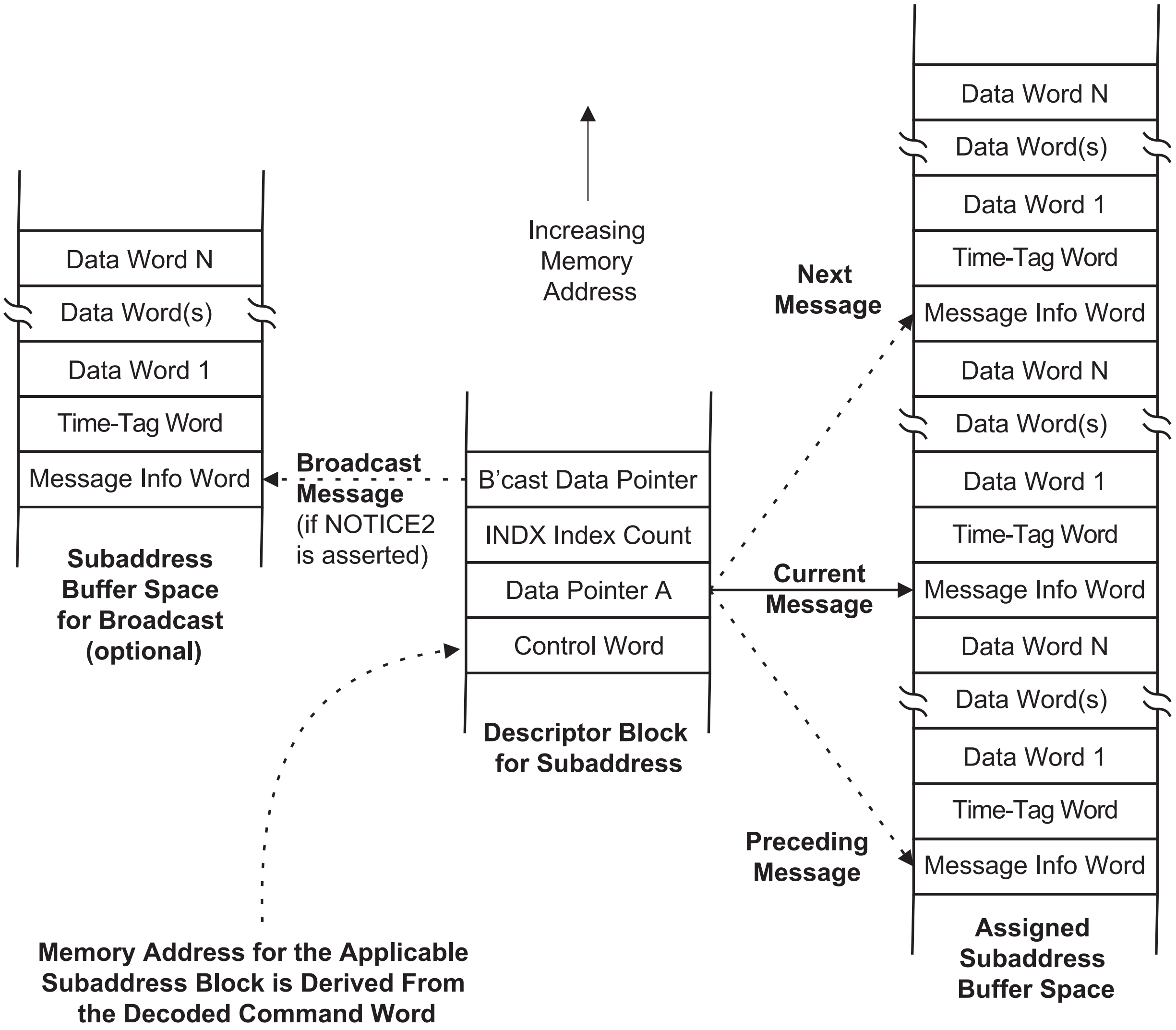
When a broadcast command is received, message information and data are stored in the broadcast data buffer. If descriptor Control Word IBR bit is set, an $\overline{\text{INT}}$ interrupt is generated. The host must read the Interrupt Log to determine the originating subaddress (or mode code) then service the broadcast data buffer for that subaddress (or mode code) before the next broadcast message to the same subaddress (or mode code) arrives.

Option 2 for Index Mode Broadcast Messages:

The second alternative stores both broadcast and non-broadcast message information in data buffer A. Optional IBR interrupts can signal arrival of broadcast messages. The RT handles broadcast messages just like non-broadcast messages, except the Message Information Word BCAST bit is asserted to identify broadcast messages during host buffer servicing. All messages decrement the INDX register and Data Pointer A is updated in message post-processing. This scheme is compatible with Single Message Mode or conventional N-message indexing. For Notice II compliance, separation of broadcast and non-broadcast data occurs within the host.

Option 2 Setup: At initialization, host negates the NOTICE2 bit in the “Remote Terminal Configuration Register (0x0017)”. If broadcast interrupts are used, the Control Word IBR (Interrupt Broadcast Received) bit is asserted at each desired index mode descriptor block. The IBR bit is also asserted in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”.

Using option 2, the host has several options for servicing data buffer A: (a) when INDX decrements from one to zero (using the IXEQZ interrupt), (b) when a broadcast message occurs (using the IBR interrupt) or (c) when any message arrives (using the IWA interrupt).



Upon successful message completion, if non-zero the INDX count in Descriptor Word 3 is decremented. If decremented result is non-zero, Data Pointer A is adjusted so next message is stored above just-completed message. If decremented INDX is zero, Data Pointer A remains static and IXEQZ interrupt occurs if enabled in Control Word.

Figure 21. Illustration of Indexed Buffer Mode

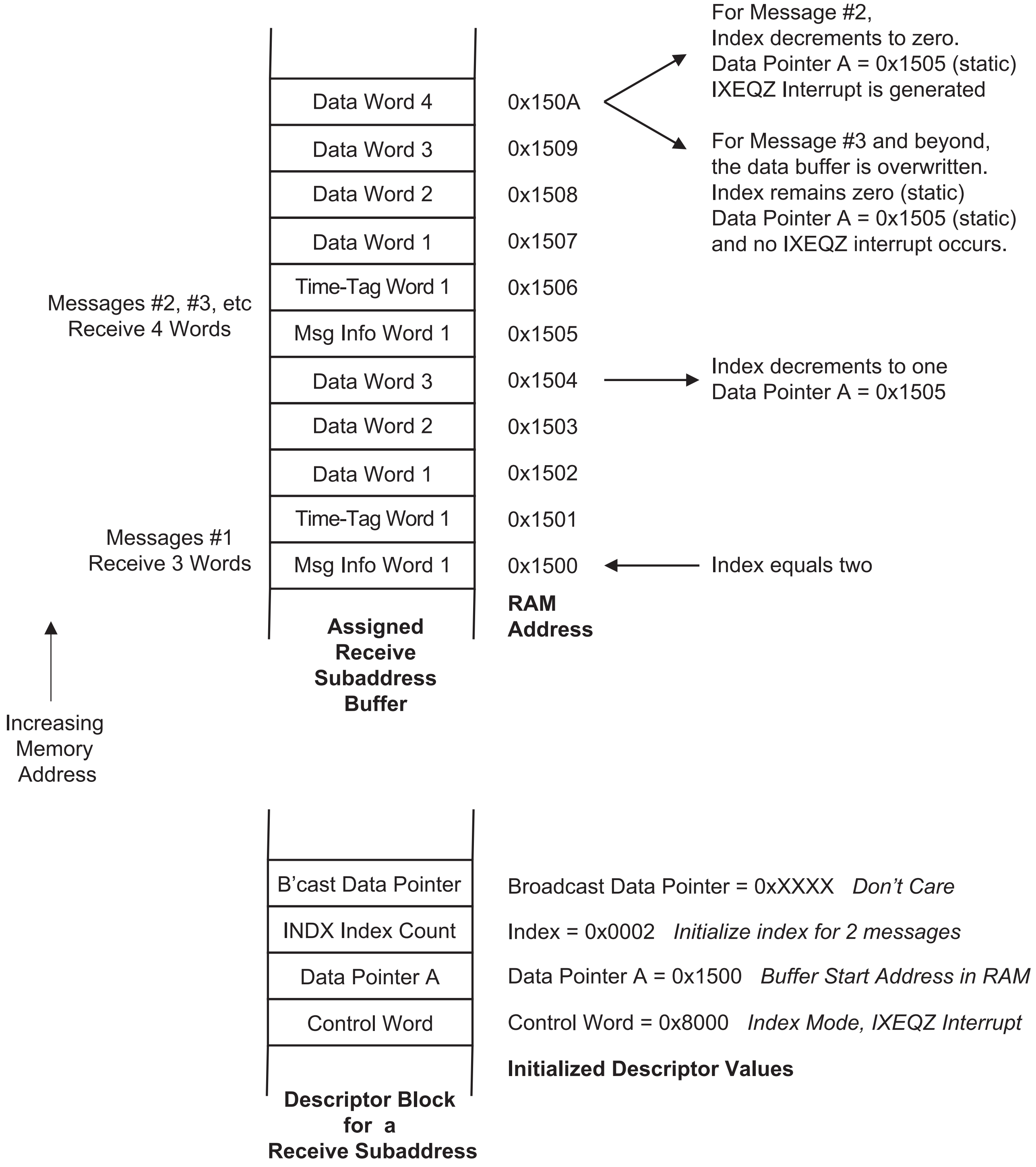


Figure 22. Indexed Buffer Mode Example for a Receive Subaddress (broadcast disabled)

19.5. Circular Buffer Mode 1

The IP core offers two circular data buffer modes as alternatives to ping-pong and indexed buffering. These circular buffer options only apply for subaddress commands, not mode code commands. Circular buffering simplifies software servicing of the remote terminal when implementing bulk data transfers. A circular buffer mode can be selected for any subaddress by properly initializing its descriptor Control Word. Circular Buffer Mode 1 is selected when descriptor Control Word PPEN and CIR2EN bits are both 0, and the CIR1EN bit is logic 1.

When a subaddress uses circular buffer mode 1, its four word block in the Descriptor Table is defined as follows:

Descriptor Word 1	Control Word
Descriptor Word 2	SA (Buffer Start Address)
Descriptor Word 3	CA (Buffer Current Address)
Descriptor Word 4	EA (Buffer End Address)

If Descriptor Word 1 is stored at memory address N, Descriptor Word 2 is stored at address N+1, and the other two words are stored at addresses N+2 and N+3.

Figure 23 provides a generalized illustration of Circular Buffer Mode 1, while Figure 24 shows a specific example. Circular Buffer Mode 1 uses a single user-defined buffer that merges all transmit or receive data, along with message information. Two words (Message Information and Time-Tag) are stored at the beginning of the block for each message, followed by the message data word(s). The Mode 1 buffer pointers roll over (are reset to their base addresses) when the allocated data buffer memory is full.

For each valid receive message, the IP core enters a Message Information word, Time-Tag word and data word(s) into the circular receive buffer. For each valid transmit message, the IP core enters a Message Information word and a Time-Tag word into reserved memory locations within the circular transmit buffer. The IP core automatically controls the wrap around of circular buffers.

Two pointers define circular buffer length: start of buffer (lowest address) and end of buffer (highest address). User specifies the start of buffer (SA) by writing the lowest address value into the second word of a unique subaddress descriptor block. The user defines the end of the buffer (EA) by writing the highest address value to the fourth word of that unique descriptor block. Both SA and EA remain static during message processing. The third word in the descriptor block identifies the current address CA (i.e., last accessed address plus one). The circular buffer wraps to the start address after completing a message that results in CA being greater than or equal to EA. If CA increments past EA during message processing, the IP core will access memory addresses greater than the EA value. Reserve 33 address locations past the EA address to accommodate a worst-case 32 data word message with a record starting at address = EA minus 1.

Each receive subaddress and transmit subaddress may have a unique circular buffer assignment. The RT decodes the command word T/$\overline{R}$ bit, subaddress field and word count / mode code field to select the unique command descriptor block containing the Control Word, SA pointer, CA pointer and EA pointer.

For receive messages, the IP core stores the Message Information word to the address specified by CA, the Time-Tag word into CA+1 and the data into the next “N” locations starting with CA+2. For transmit messages, the IP core stores the Message Information word to the address specified by CA and the Time-Tag word into CA+1. Retrieval of data for transmission starts at address CA+2. When entering multiple transmit command data packets into the circular buffer, delimit each data packet with two reserved memory locations. The IP core stores the Message Information word and Time-Tag word into the reserved locations when processing the command.

Message processing for all commands begins with the IP core reading the unique descriptor block for the subaddress or mode code specified by the T/$\overline{R}$ bit, subaddress and word count fields in the received command word.

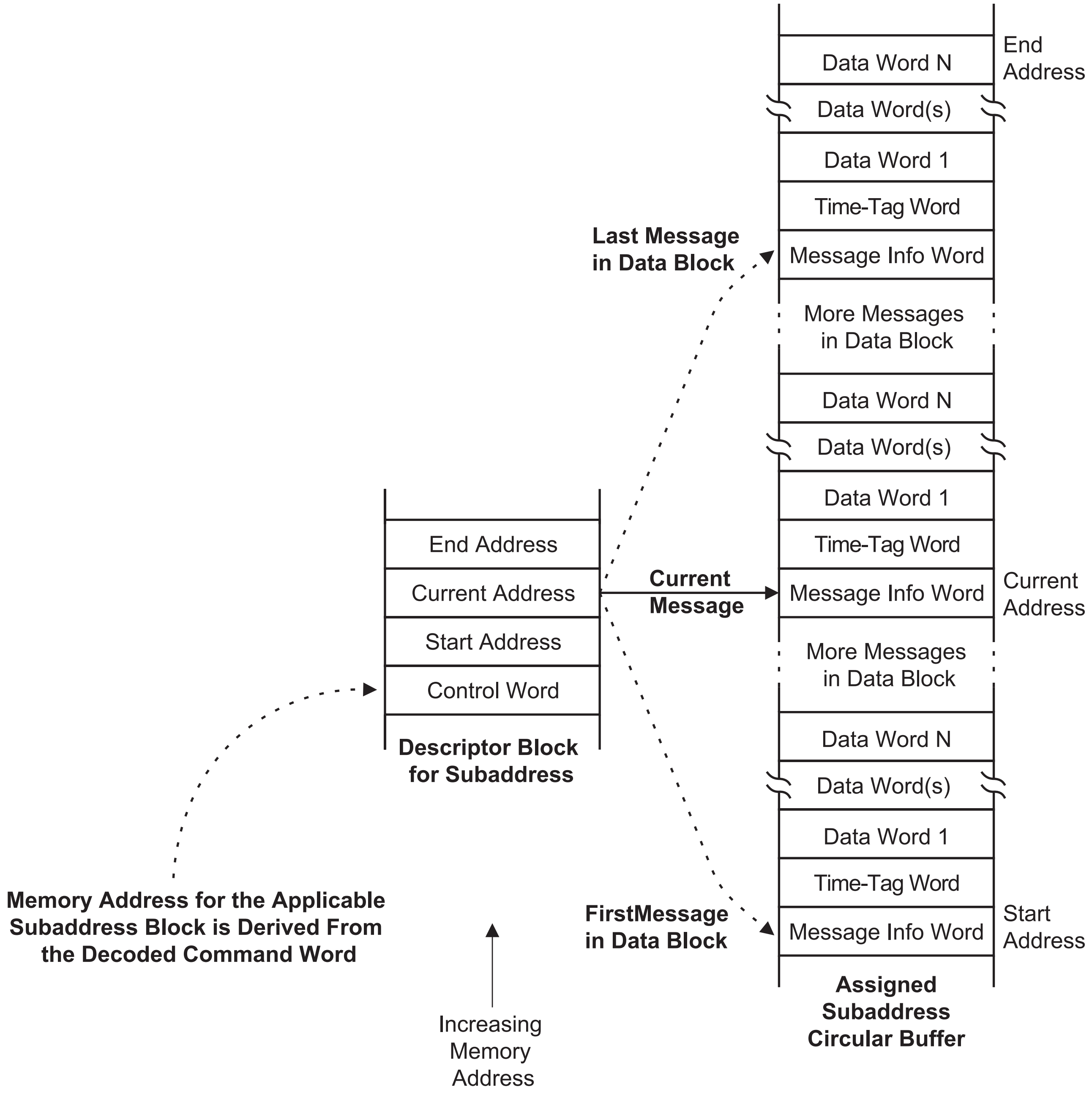
For receive messages, the IP core stores “N” received data words in the circular data buffer. The first data word received is stored at the location specified by the CA pointer +2. After message completion, the IP core stores the Message Information word and Time-Tag words to addresses CA and CA+1 respectively. If no errors were detected, the IP core updates descriptor CA register. If the next address location (last stored data word +1) is less than or equal to EA, CA is updated to (last stored address +1). If the next address location (last stored data word +1) is greater than EA, the data buffer is full (or empty); CA is updated to the SA value. If descriptor Control Word IXEQZ bit is asserted (and if “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is asserted) the IP core generates an interrupt to indicate full receive buffer by asserting the $\overline{\text{INT}}$ interrupt output.

Although all messages store Message Information and Time-Tag words, no data is stored if the message ended with error, or if the Busy status bit was set or if the command was illegal (example: illegalized word count). Such messages do not update CA, so the next message overwrites the same buffer space.

For transmit commands, the IP core begins transmission of data retrieving the first data word stored at address CA+2. (Reminder: addresses CA and CA+1 are reserved for the Message Information and Time-Tag words.) When message processing is complete, the IP core writes the Message Information and Time-Tag words into the buffer. If no errors were detected, the IP core updates descriptor CA register. If the next address location (last retrieved data word +1) is less than or equal to EA, CA is updated to (last retrieved address +1). If the next address location (last retrieved data word +1) is greater than EA, the transmit data buffer is empty; CA is updated to the SA value. If the descriptor Control Word IXEQZ bit is asserted (and if the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is asserted) the IP core indicates “transmit buffer empty” by asserting the $\overline{\text{INT}}$ interrupt output.

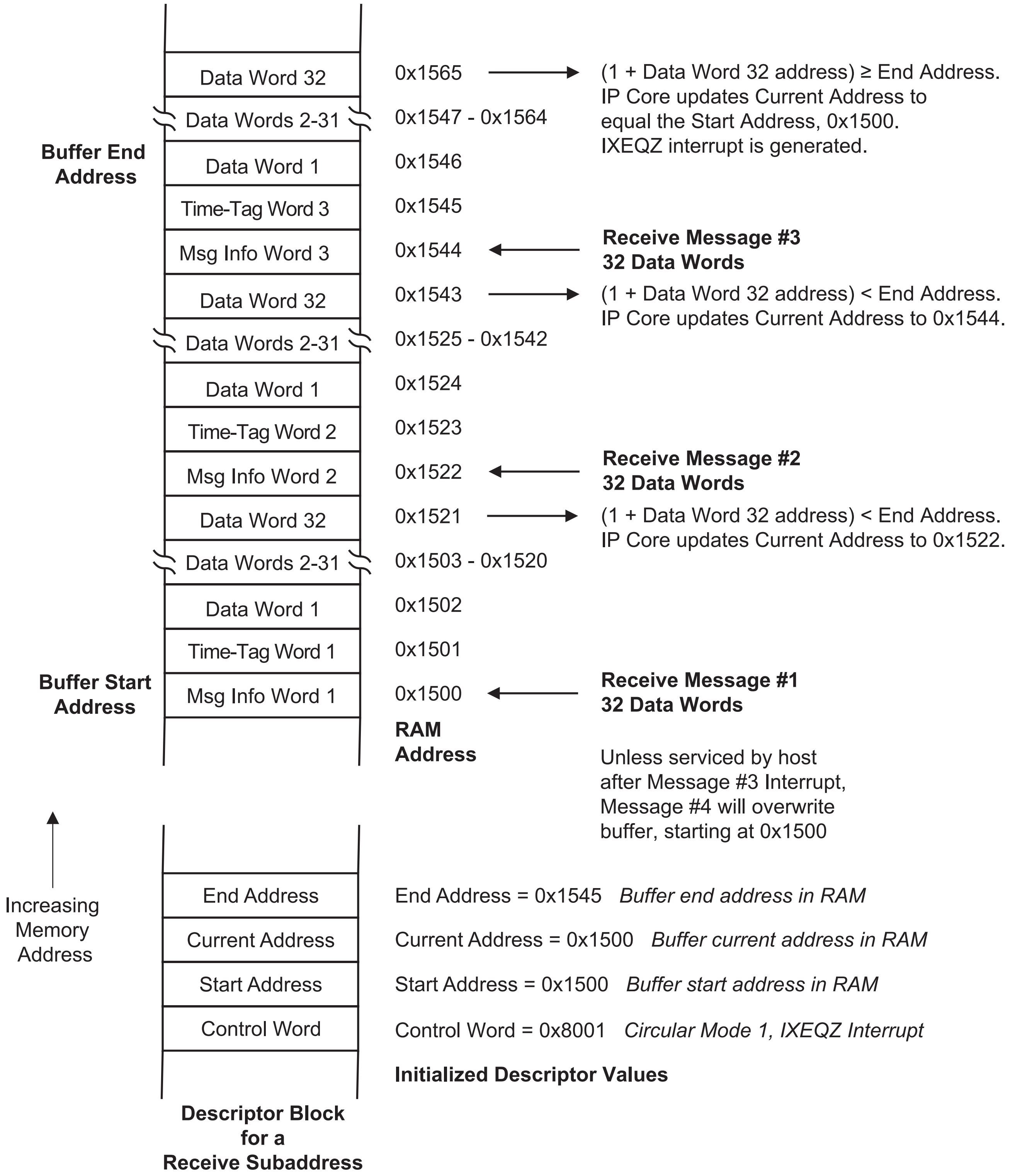
The IP Core does not segregate broadcast and non-broadcast data for this circular buffer mode, even when the NOTICE2 bit is set in the “Remote Terminal Configuration Register (0x0017)”. Data words from broadcast receive commands are stored in the same buffer with data from non-broadcast receive commands. However Notice 2 for MIL-STD-1553 does not state where data segregation should occur. It is acceptable for the host to separate broadcast and non-broadcast data when offloading the circular buffer. To choose this option, set bit 3 in “Extended Configuration Register (0x004D)”. This enables the BCAST (broadcast status) bit in the Message Information Word stored for each message. This flag reflects broadcast or non-broadcast status for each message in the buffer.

For transmit subaddresses using Circular Buffer Mode 1, occurrences of broadcast-transmit commands to RT31 do not result in bus transmission. However these messages update the Message Information Word addressed by the Current Address (CA) pointer (and following Time-Tag Word) but afterwards, the CA pointer remains unchanged. The next transmit command to the same subaddress, whether broadcast or not, overwrites the Message Information and Time-Tag Word locations written by the previous broadcast transmit command.



Descriptor block is initialized so Current Address equals buffer Start Address. After each successful message transaction, Current Address is adjusted to point past last data word accessed. If adjusted Current Address points past End Address, the Current Address is reinitialized to match Start Address and an optional interrupt is generated to notify host that the pre-determined data block was fully transacted.

Figure 23. Illustration of Circular Buffer Mode 1


Data Word 32
0x1565
(1 + Data Word 32 address) ≥ End Address.
IP Core updates Current Address to
equal the Start Address, 0x1500.
IXEQZ interrupt is generated.
Data Words 2-31
0x1547 - 0x1564
Buffer End Address
Data Word 1
0x1546
Time-Tag Word 3
0x1545
Msg Info Word 3
0x1544
Receive Message #3
32 Data Words
Data Word 32
0x1543
(1 + Data Word 32 address) < End Address.
IP Core updates Current Address to 0x1544.
Data Words 2-31
0x1525 - 0x1542
Data Word 1
0x1524
Time-Tag Word 2
0x1523
Msg Info Word 2
0x1522
Receive Message #2
32 Data Words
Data Word 32
0x1521
(1 + Data Word 32 address) < End Address.
IP Core updates Current Address to 0x1522.
Data Words 2-31
0x1503 - 0x1520
Data Word 1
0x1502
Time-Tag Word 1
0x1501
Buffer Start Address
Msg Info Word 1
0x1500
Receive Message #1
32 Data Words
RAM Address
Unless serviced by host
after Message #3 Interrupt,
Message #4 will overwrite
buffer, starting at 0x1500
Increasing Memory Address
End Address
End Address = 0x1545 Buffer end address in RAM
Current Address
Current Address = 0x1500 Buffer current address in RAM
Start Address
Start Address = 0x1500 Buffer start address in RAM
Control Word
Control Word = 0x8001 Circular Mode 1, IXEQZ Interrupt
Initialized Descriptor Values
Descriptor Block
for a
Receive Subaddress

Unlike Indexed mode, Data Block completion is based on Buffer Full / Buffer Empty, not number of messages.
Buffer size was purposely sized to yield remaining capacity after 2 full-count messages, to illustrate IP Core behavior.
The circular buffer should have a 33-word pad beyond its End Address to deal with buffer overrun without data loss.

Figure 24. Circular Buffer Mode 1 Example for a Receive Subaddress

19.6. Circular Buffer Mode 2

Circular Buffer Mode 2 segregates message data and message information in separate host-defined buffers. Separating data from message information simplifies the host software that loads or unloads the data to or from the buffer. After a predetermined number of messages has been transacted, buffer address pointers for data and message information are automatically reset to their base addresses. Figure 25 is a generalized illustration of Circular Buffer Mode 2, while Figure 26 shows a specific example.

Circular Buffer Mode 2 is selected when the Control Word PPEN bit is zero and the CIR2EN bit is logic 1. When the CIR2EN bit is high, the CIR1EN bit is don't care. The descriptor Control Word DPB bit is not used.

Any receive subaddress using circular buffer mode 2 has two circular buffers: a data storage buffer and a message information buffer. A separate buffer pair may be used for transmit commands to the same subaddress, if it also uses circular buffer mode 2. Each transmit and receive subaddress using circular buffer mode 2 may have unique data buffer and message info buffer assignments. Careful management (involving the bus controller) may allow buffer sharing, as long as multiple message sequences to a given subaddress are not interrupted by messages to other subaddresses that use the same buffer space.

When a subaddress uses circular buffer mode 2, its Descriptor Table 4-word block is defined as follows:

Descriptor Word 1	Control Word
Descriptor Word 2	SA (Buffer Start Address)
Descriptor Word 3	CA (Buffer Current Address)
Descriptor Word 4	MIBA (Message Info Buffer Addr)

If Descriptor Word 1 is stored at memory address N, Descriptor Word 2 is stored at address N+1, and the other two words are stored at addresses N+2 and N+3. The first word in the descriptor block is the Control Word. The second and third words in the descriptor are the Start Address (SA) and Current Address (CA) pointers. The Message Information Buffer Address (MIBA) points to the storage location for the Message Information Word from the next occurring message.

Each time a message is completed, the IP core writes a new Message Information Word and Time-Tag Word in the MIB (Message Information Buffer) at the MIBA address and following location, respectively. The MIBA pointer is not updated if message error occurred, if the Busy status bit was set, or if the command was illegalized (for example an illegal word count expressed in the command word.) For these situations, the Message Information and Time-Tag words are still written, but MIB updates for the following message will overwrite the just-written Message Information and Time-Tag word addresses.

For error-free receive messages, received data words are stored in the data buffer after message completion, starting at the CA address value. The CA value is then updated for next-message readiness.

After writing the two MIB words, the IP core updates the MIBA value to show the buffer address to be used by the next message. Until the predetermined number of error-free messages is transacted, the MIBA value is double-incremented at each update. Before updating the MIBA in Descriptor Word 4, the pre-existing MIBA value is incremented once then checked for 'full count," occurring when all N low-order address bits initialized to zero (explained below) become N "one" bits. Full count means the predetermined number of successful messages was completed. When this occurs, the CA and MIB pointers are automatically written to their initialized values by the IP core.

To preserve data integrity, the TRXDB bit should be set in the "Remote Terminal Configuration Register (0x0017)" to avoid storing incomplete data from messages resulting in error. With TRXDB asserted, the host is not bothered by message retries caused by errors. The Buffer Empty/Full interrupt (if enabled) is generated only upon successful transaction of the entire N-message data block.

To initialize Circular Buffer Mode 2, the host must know the number of messages to be transacted, always a power of two: 1, 2, 4, 8, 16, 32, 64, 128, 256 or 512 messages. The host writes descriptor Control Word bits 7:4 with an encoded 4-bit value to set the fixed number of messages to be transacted. This is illustrated in Table 19. The host initializes the descriptor block MIBA pointer with a Message Information Buffer starting address. Because the MIB stores two words for each message, the allocated MIB space should equal 2x the number of messages.

The initially-loaded MIB base address value is restricted. Some lower bits of the starting address must be zero so the IP core can restore the MIBA pointer to the initial MIB base address after the predetermined message count is transacted. As illustrated in Table 19, the required number of logic-0 bits depends on the message count. Initializing the MIBA base address with more trailing zeros than indicated is acceptable; initializing less trailing zeros will cause malfunction.

Allocated space in the data buffer (see column 3, Table 19) assumes each message has the maximum 32 data words. If messages contain less than 32 words, the data buffer size can be reduced. Since Circular Buffer Mode 2 counts messages, values in all remaining Table 19 columns remain valid when message word count is reduced.

The host may read the MIBA value to determine the number of messages that have occurred since initialization. By reading the initially-zeroed lower bits of the MIB Address, the host may determine the number of the next occurring message.

From Table 19, a block of 128 messages requires 8 trailing zeros in the initial MIBA address, for example, 0x0F00. After each message is completed, the MIBA value is updated (0x0F02, 0x0F04, etc.) The IP core detects message block completion when all required initially-zero trailing address bits equal 1 after MIBA is incremented once. In our example, MIBA would increment from 0x0FFE to 0x0FFF. When “full count” occurs, the IP core updates MIBA to the original value (e.g., 0x0F00) and copies the SA starting address value to CA current address register, ready for buffer service by the host. The IP core optionally generates a “buffer empty-full” interrupt for the host when block transfer is completed.

During block transfer, the host can read the MIBA value to determine the number of additional messages needed before the N-message data block is complete.

Message processing for all commands begins with the RT reading the unique descriptor block for the subaddress specified by the T/$\overline{\text{R}}$ bit, subaddress and word count fields in the received command word.

Table 19. Circular Buffer Mode 2 (Initialization factors based on message block size)

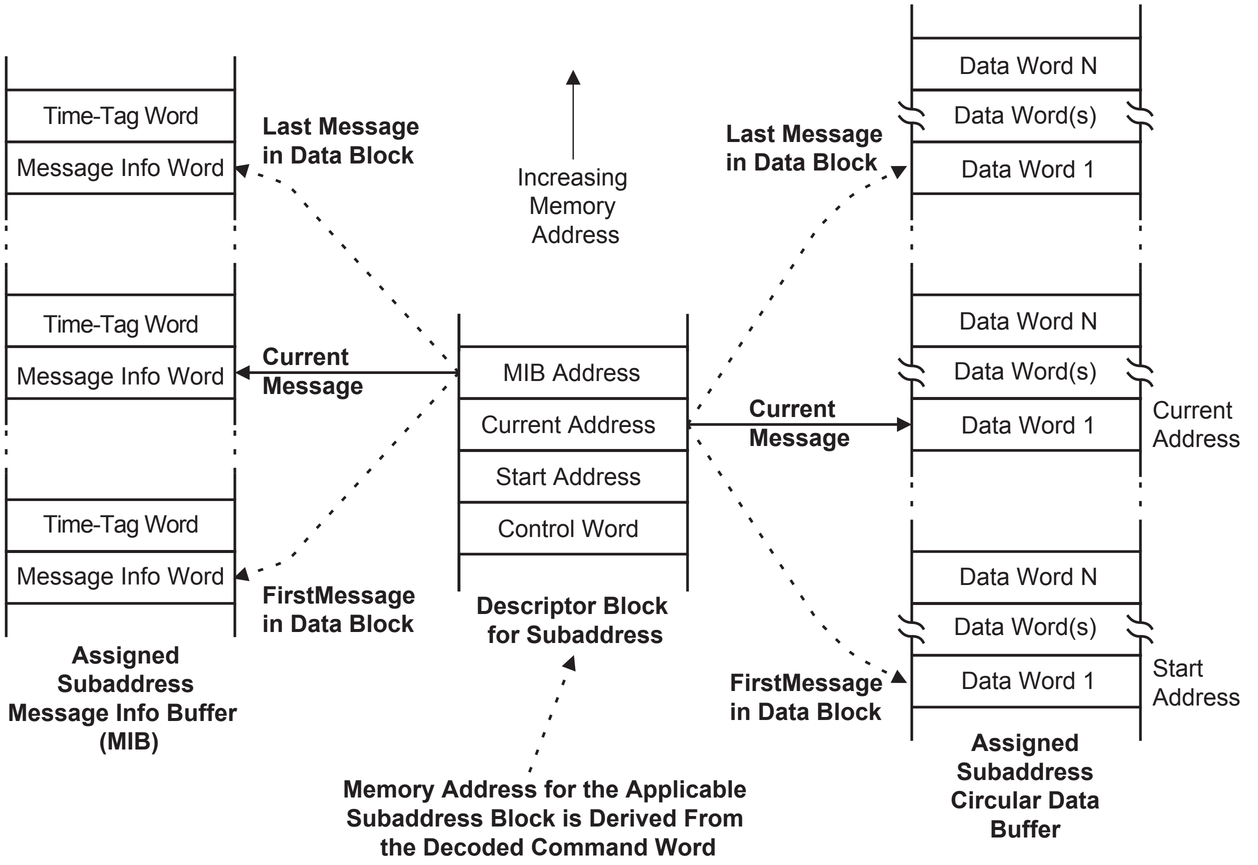
Number of Messages	Control Word Bits 7:4 CIR2ZN Field	Required Data Space if 32 Words / Msg	Required MIB Space, 2 Words / Msg	Initial Binary MIBA Value, Showing the Required Leading and Trailing Zeros
2	0010 (2)	64	4	00xxxxxxxxxxxx00
4	0011 (3)	128	8	00xxxxxxxxxxx000
8	0100 (4)	256	16	00xxxxxxxxxx0000
16	0101 (5)	512	32	00xxxxxxxxx00000
32	0110 (6)	1,024	64	00xxxxxxxx000000
64	0111 (7)	2,048	128	00xxxxxxx0000000
128	1000 (8)	4,096	256	00xxxxxx00000000
256	1001 (9)	8,192	512	00xxxxx000000000
512	1010 (A)	16,384	1,024	00xxxx0000000000

For receive subaddresses using Circular Buffer Mode 2, the IP core stores received data words in the circular data buffer. The first data word received for each message is stored at the location indicated by the CA pointer.   After the correct number of words is received (as specified in the command word) the IP core writes Message Information and Time-Tag words in the Message Information Buffer then updates the descriptor CA Current Address and MIBA Message Information pointers for next-message readiness. If the predetermined total number of messages has not yet been transacted, MIBA points to the next location in the message information buffer and CA points to the next location in the data buffer. If the completed message is the last message in the block, the CA current (data) address and MIBA message Information pointers are reinitialized to their base address values. (Control Word bits 7:4 tell the IP core how many MIBA lower bits to reset.) If the descriptor Control Word IXEQZ bit is asserted (and if the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is asserted) the IP core generates a Buffer Full / Empty interrupt, asserting the $\overline{\text{INT}}$ interrupt output.

For transmit subaddresses using Circular Buffer Mode 2, the IP core transmits data from the assigned RAM buffer, starting at the location specified by the CA pointer. The first data word transmitted is stored at the location specified by the CA pointer. After all data words are transmitted (as specified in the command word) the IP core writes Message Information and Time-Tag words in the Message Information Buffer then updates the descriptor CA Current Address and MIBA Message Information pointers for next-message readiness. If the predetermined total number of messages has not yet been transacted, MIBA points to the next location in the message information buffer and CA points to the next location in the data buffer. If the completed message is the last message in the block, the CA current (data) address and MIBA message Information pointers are reinitialized to their base address values. (Control Word bits 7:4 tell the IP core how many MIBA lower bits to reset.) If the descriptor Control Word IXEQZ bit is asserted (and if the “Remote Terminal (RT) Interrupt Enable Register (0x0012)” IXEQZ bit is asserted) the IP core generates a Buffer Full / Empty interrupt, asserting the $\overline{\text{INT}}$ interrupt output.

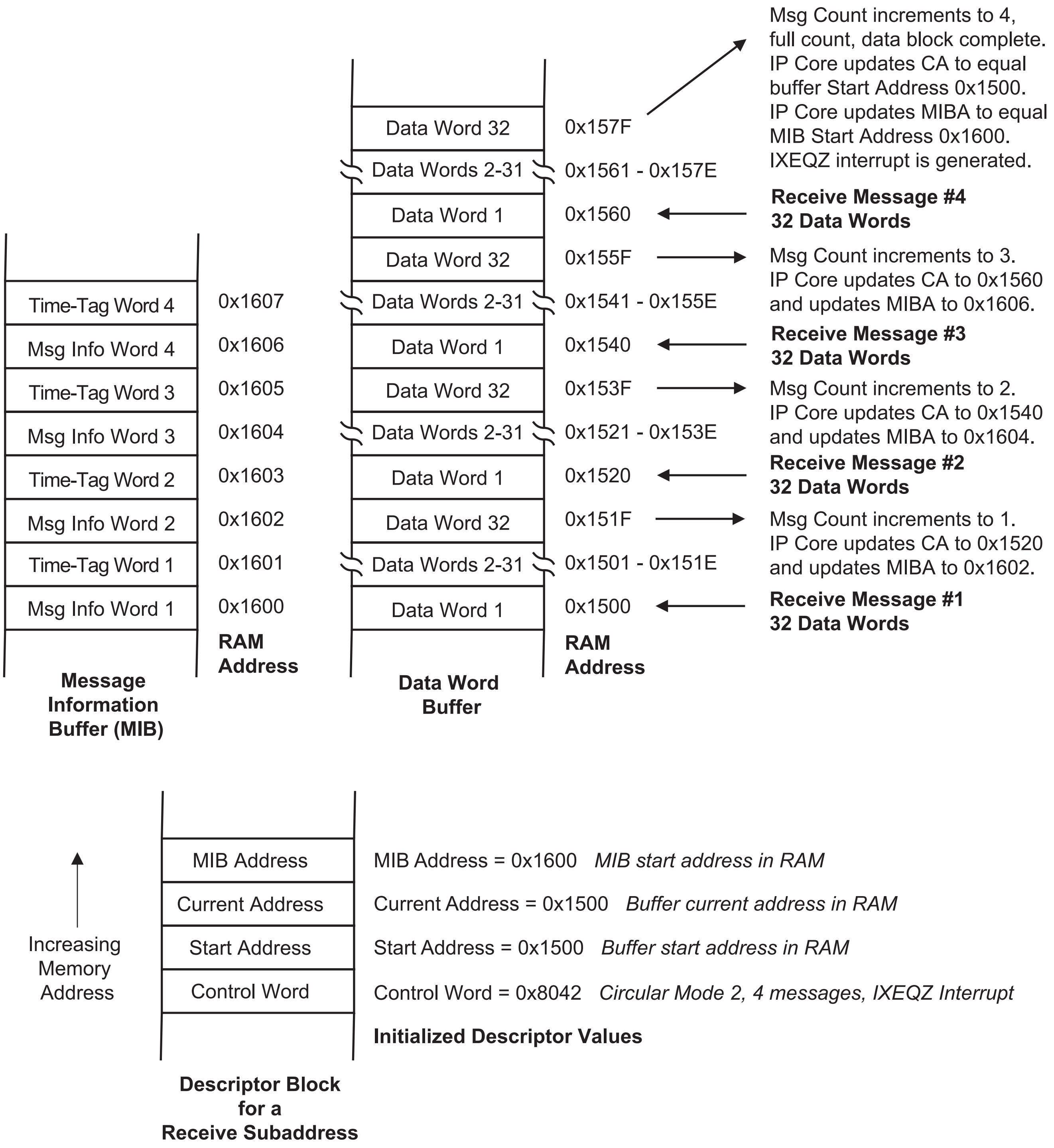
The IP Core does not segregate broadcast and non-broadcast data for this circular buffer mode, even when the NOTICE2 bit is set in the “Remote Terminal Configuration Register (0x0017)”. Data words from broadcast receive commands are stored in the same buffer with data from non-broadcast receive commands. However Notice 2 for MIL-STD-1553 does not state where data segregation should occur. It is acceptable for the host to separate broadcast and non-broadcast data when offloading the circular buffer. To choose this option, set bit 3 in “Extended Configuration Register (0x004D)”. This enables the BCAST (broadcast status) bit in the Message Information Word stored for each message. This flag reflects broadcast or non-broadcast status for each message in the buffer.

For transmit subaddresses using Circular Buffer Mode 2, occurrences of broadcast-transmit commands to RT31 do not result in bus transmission. However these messages update the Message Information Word addressed by the Message Information Block (MIB) pointer (and the following Time-Tag Word) but afterwards, the MIB and CA pointers remain unchanged. The next transmit command to the same subaddress, whether broadcast or not, overwrites the Message Information and Time-Tag Word locations written by the previous broadcast transmit command.



Segregated storage for data and message information simplifies host loading / offloading of buffered data. Descriptor MIB Address tracks number of messages. Full count occurs when N initialized 0-bits become N 1-bits. When full number of messages in block is transacted, an optional interrupt is generated to notify host.

Figure 25. Illustration of Circular Buffer Mode 2



Data Block completion is based on number of messages, not Buffer Full or Buffer Empty.
Example is set to successfully transact four 32 data word receive messages, then generate IXEQZ interrupt for host.
The data buffer requires minimal processing by host because message information words are stored separately in MIB.

Figure 26. Circular Buffer Mode 2 Example for a Receive Subaddress

20. REMOTE TERMINAL MODE COMMAND PROCESSING

20.1. General Considerations

The IP core provides decoding for all mode code combinations, consistent with MIL-STD-1553B requirements. Several mode command options are provided to suit any application requirement:

In the “Remote Terminal Configuration Register (0x0017)”, the option bit UMCINV (Undefined Mode Codes Invalid) globally defines whether undefined mode code commands are treated as valid (default) or invalid commands. This bit applies only to the following 22 mode code commands that are undefined in MIL-STD-1553B:

Mode Codes 0 through 15 with T/$\overline{\text{R}}$ bit = 0
Mode Codes16, 18 and19 withT/$\overline{\text{R}}$ bit = 0
Mode Codes 17, 20 and 21 with T/$\overline{\text{R}}$ bit = 1

If the UMCINV bit is low (default after $\overline{\text{RESET}}$ reset) undefined mode code commands are considered valid and RT response is based on individual mode command settings in the Illegalization Table: If the command’s illegalization table bit equals 0, the mode command is legal; the RT responds “in form” and updates status. If the command’s illegalization table bit equals 1 the mode command is illegal, the RT asserts Message Error status and (if non-broadcast) transmits only its Status Word without associated data word. Table 20 describes explicit terminal response for each mode code value and command T/$\overline{\text{R}}$ bit state, based on various option settings.

If UMCINV is asserted, the 22 undefined mode code commands are treated as invalid: There is no terminal recognition of the command. No command response occurs and status remains unchanged for the benefit of following “transmit status” or “transmit last command” mode commands.

If UMCINV is low, the IP core determines legal vs. illegal status of commands from the Illegalization Table. If the terminal does not use illegal command detection, the Illegalization Table should be left in its post-reset default state, all values equal logic 0. In this case, the terminal provides “in form” response to all valid commands. The terminal responds with clear status and a transmitted mode data word for mode commands 16-31 with T/$\overline{\text{R}}$ bit equals 1. Assigned data buffer locations can be initialized to provide predictable “in form” responses for all transmit mode codes 16-31. (If UMCINV is asserted, the terminal will not respond or update status for received mode codes 17, 20 and 21 with T/$\overline{\text{R}}$ = 1.)

To use illegal command detection, the host modifies the Illegalization Table to make illegal any combination subaddress and mode code commands. This may include undefined mode codes, reserved mode codes, and/or mode codes not implemented in the application.

20.2. Mode Command Interrupts

For mode commands, interrupt generation is programmed by the top three bits in the descriptor table Control Word. Notice that broadcast-transmit interrupts can be enabled for mode code values in the range of 0 - 15, but broadcast-transmit mode codes 16 - 31 are not allowed. When a mode command is received and the IWA interrupt bit is asserted in its descriptor Control Word, that command will generate a host interrupt if the IWA bit is high in the “Remote Terminal (RT) Interrupt Enable Register (0x0012)”. The IWA bit is asserted in the “Remote Terminal (RT) Pending Interrupt Register (0x0009)” and the $\overline{\text{INT}}$ interrupt output is asserted.

Before $\overline{\text{INT}}$ interrupt assertion, the IP core updates the Interrupt Log buffer, writing a new IIW Interrupt Information Word and a new IAW Interrupt Address Word. The IWA (interrupt when accessed) bit is asserted in the new IIW to indicate interrupt type. The IAW contains the Descriptor Table address for the mode command’s Control Word, based on mode code value and command word T/$\overline{\text{R}}$ bit state. The host reads the IAW to determine the command that caused the interrupt.

20.3. Mode Command Data Words

Mode commands having mode code values from 0 through 15 (decimal) do not have an associated data word. These are received as Command Word only, never having a contiguous data word. The terminal response to valid mode commands 0-15 always consists of Status Word only, assuming command was not broadcast.

Mode commands having mode code values from 16 through 31 (decimal) always have an associated data word. When the command word $T/\overline{R}$ bit equals 0, the terminal receives a data word, contiguously following the Command Word. When valid legal mode commands 16-31 arrive with $T/\overline{R}$ bit equal to 1, the terminal responds by transmitting its status word with a single data word.

When the SMCP option bit in the “Remote Terminal Configuration Register (0x0017)” is zero, individual data words for mode codes 16-31 decimal are stored in an indexed or ping-pong buffer assigned by the mode command’s Descriptor Table entry. Circular buffer methods are not available for mode code commands.

When the SMCP option bit in the “Remote Terminal Configuration Register (0x0017)” is asserted, individual data words for mode codes 16-31 decimal are stored within the Descriptor Table itself. This is explained next.

Table 20. Mode Code Command Summary

Command $T/\overline{R}$ bit	Mode Code		MIL-STD-1553 Defined Function	Associated Data Word	Broadcast Allowed	See Note
	Binary	Dec.				
0	00000 to 01111	0 to 15	Undefined mode commands 0 - 15 when $T/\overline{R}$ bit = 0	No	No	(1)
1	00000	0	Dynamic Bus Control	No	No	
1	00001	1	Synchronize (without data)	No	Yes	
1	00010	2	Transmit Status Word	No	No	
1	00011	3	Initiate Self-Test	No	Yes	
1	00100	4	Transmitter Shutdown	No	Yes	
1	00101	5	Override Transmitter Shutdown	No	Yes	
1	00110	6	Inhibit Terminal Flag	No	Yes	
1	00111	7	Override Inhibit Terminal Flag	No	Yes	
1	01000	8	Reset Remote Terminal	No	Yes	
1	01001 to 01111	9 to 15	Reserved Mode Commands 9 - 15 with $T/\overline{R}$ bit = 1	No	Yes	(2)
0	10000	16	Undefined Mode Command	Yes	No	(1)
1	10000	16	Transmit Vector Word	Yes	No	
0	10001	17	Synchronize With Data	Yes	Yes	
1	10001	17	Undefined Mode Command	Yes	No	(1)
0	10010	18	Undefined Mode Command	Yes	No	(1)

Command T/$\overline{R}$ bit	Mode Code		MIL-STD-1553 Defined Function	Associated Data Word	Broadcast Allowed	See Note
	Binary	Dec.				
1	10010	18	Transmit Last Command	Yes	No	
0	10011	19	Undefined Mode Command	Yes	No	(1)
1	10011	19	Transmit Built-In Test Word	Yes	No	
0	10100	20	Selected Transmitter Shutdown	Yes	Yes	
1	10100	20	Undefined Mode Command	Yes	No	(1)
0	10101	21	Override Selected Transmitter Shutdown	Yes	Yes	
1	10101	21	Undefined Mode Command	Yes	No	(1)
0	01001 to 01111	22 to 31	Reserved Mode Commands 22 - 31 with T/$\overline{R}$ bit = 0	Yes	Yes	(2)
1	01001 to 01111	22 to 31	Reserved Mode Commands 22 - 31 with T/$\overline{R}$ bit = 1	Yes	No	(2)

NOTES:

1. The 22 undefined mode commands can be rendered invalid by setting the UMCINV (undefined mode codes invalid) option bit in “Remote Terminal Configuration Register (0x0017)”. If UMCINV is asserted, there is no recognition of the undefined command by the terminal. If UMCINV is zero, the commands are considered valid. Terminal response when UMCINV equals 0 is wholly determined by the Illegalization Table:

   a. If a command’s bit in the Illegalization Table equals zero, the terminal responds “in form” with Clear Status. Mode commands 17, 20 and 21 are undefined when T/$\overline{R}$ bit equals one, but will transmit a contiguous data word. Mode commands 16, 18 or 19 are undefined when T/$\overline{R}$ bit equals 0, but will receive a contiguous data word.

   b. If a command’s bit in the Illegalization Table equals one, the command is considered illegal. The Message Error (ME) status bit is asserted and the terminal transmits status without data word. Illegal mode commands 16-31 will not transmit or receive a mode data word.

2. Response to the reserved mode commands is fully defined by Illegalization Table settings. As described in (a) and (b) above, the terminal illegalizes any reserved mode command having Illegalization Table bit equal to 1, and responds “in form” when the Table bit equals zero. The “in form” response for reserved mode commands 16 through 31 transacts a received or transmitted data word.

20.4. Standard Mode Command Processing

Data buffer options for mode commands differ from buffer options for subaddress commands. Mode commands can use ping-pong buffering or indexed buffering. When mode commands use indexed buffers, "single message mode" (INDX = 0) is recommended. When using indexed or ping-pong buffers for mode commands:

- For mode commands without associated data word (mode codes 0-15 decimal), only the Message Information and Time-Tag words are updated in the mode command's assigned data buffer in RAM.
- For mode commands 16-31 (decimal) that receive a data word, indexed and ping-pong buffer methods copy the received mode data word to the mode command's assigned data buffer in shared RAM, after the message is transacted. The Message Information and Time-Tag words are also updated.
- For most mode commands 16-31 (decimal) that transmit a data word, the IP core reads the data word for transmit from the buffer location assigned in the Descriptor Table. Exceptions occur for MC18 "transmit last command" and for MC19 "transmit BIT word." The MC18 data word is automatically provided by the IP core, based on recent command transactions. The MC19 data word comes from register 0x1E or 0x1F, selected by the ALTBITW option in the "Remote Terminal Configuration Register (0x0017)". For both MC18 and MC19, the transmitted data word is automatically recorded in the mode command's assigned data buffer in RAM, after message completion. The Message Information and Time-Tag words are also updated.

20.5. Simplified Mode Command Processing

Mode commands have a buffer alternative that is unavailable for subaddress commands. The SMCP bit in the "Remote Terminal Configuration Register (0x0017)" selects Simplified Mode Command Processing, a global option applying to all mode commands. When the SMCP bit is high, mode command descriptor blocks (in the Descriptor Table) do not contain data pointers to reserved buffers elsewhere in the shared RAM. Instead, each 4-word descriptor block itself contains the message information word, the time-tag word and the data from the most recent occurrence of each mode command:

Descriptor Word 1	Mode Command Control Word
Descriptor Word 2	Message Information Word
Descriptor Word 3	Time-Tag Word
Descriptor Word 4	Mode Data Word

Descriptor Word 1 contains the receive or transmit mode command Control Word. When SMCP is used, just two Control Word bits are used: DBAC (descriptor block accessed) and BCAST (broadcast).

When SMCP is enabled, the host need not initialize the mode code command segments in the Descriptor Table. When Simplified Mode Command Processing is selected, the host does not write Descriptor Words 2-3 in the Descriptor Table entries for mode commands. For mode code values 0 to 15 decimal, the Descriptor Word 4 serves no function because these mode codes do not have an associated data word. For transmit mode code values 16 to 31, the host may initialize Descriptor Word 4. The default transmit value is 0x0000. Mode command MC16 "transmit vector word" is one of the three defined mode commands that transmit a data word: MC16, MC18 and MC19. Its Descriptor Word 4 should be initialized if a value other than 0x0000 is needed. MC18 and MC19 are discussed below.

- For mode commands without associated data word (mode codes 0-15 decimal), Simplified Mode Command Processing updates the Message Information and Time-Tag words in Descriptor Words 2 and 3, and Descriptor Word 1 (bits 9,11). For these commands, SMCP does not update Descriptor Word 4, which may be non-zero if written earlier by the host.
- For receive mode commands 16-31 (decimal) that receive a data word, Simplified Mode Command Processing copies the received mode data word to Descriptor Word 4. The Message Information and Time-Tag words in Descriptor Words 2 and 3, and Descriptor Word 1 (bits 9, 11) are also updated.
- For most transmit mode codes 16-31 (decimal), the IP core reads the data word for transmission from each command's Descriptor Word 4. Exceptions occur for MC18 "transmit last command" and for MC19 "transmit built-in test word". The MC18 data word is automatically provided, based on the last command transacted. The

MC19 data word comes from register 0x1E or 0x1F, selected by the ALTBITW option in the “Remote Terminal Configuration Register (0x0017)”. For MC18 and MC19, the transmitted data value is automatically copied to the mode command’s Descriptor Word 4 after message completion. The Message Information and Time-Tag words in Descriptor Words 2 and 3, and Descriptor Word 1 (bits 9, 11) are also updated.

“Appendix: RT Messages Responses, Options & Exceptions” on page 203 shows terminal response to all possible subaddress and mode code command combinations. The table summarizes terminal response for the full range of message conditions, including errors, incomplete messages, etc. The table explicitly describes terminal response and impact on terminal Status Word, Descriptor Control Words and data buffer Message Information Words. The table includes effects for all pertinent setup options and identifies all interrupt options available. Bold text blocks indicate error-free messages or “in form” Clear Status responses when the terminal is not using “illegal command detection”.

21. RESET AND INITIALIZATION

This section describes the hardware and software reset mechanisms. Hardware Master Reset returns the IP core to the uninitialized state, requiring IP dongle verification and register and RAM initialization before terminal execution can begin. Hardware reset is initiated by assertion of the $\overline{\text{RESET}}$ Master Reset input signal (200ns minimum assertion time). Software reset does not result in IP dongle verification and is individually or simultaneously asserted for the Bus Monitor or RT by setting the corresponding bit(s) in the “Master Status and Reset Register (0x0001)”. The Bus Controller does not have software reset.

21.1. Hardware Master Reset

Hardware master reset is initiated by a low to high transition on the $\overline{\text{RESET}}$ input signal; it should be applied after power-up, but may be used any time afterward. When asserted, the $\overline{\text{RESET}}$ input signal causes immediate unconditional hardware reset for all terminal cores. Command processing is terminated and reset, the bus decoders and encoders are cleared, and all Time Tag counters are reset. All internal logic is cleared. Registers are restored to the power up reset states shown in Table 9. The READY, ACTIVE and $\overline{\text{INT}}$ output signals are negated if previously asserted. The READY signal remains low until the entire reset process and IP Dongle verification is complete. During this interval, a host read cycle to any address returns the value of the “IP Core Security Register (0x0005)”.

21.2. Software Reset

21.2.1. Remote Terminal

This is initiated when RTRESET bit 10 is set in the “Master Status and Reset Register (0x0001)”. The following actions are performed.

Table 21. RT Soft Reset Summary

Action	Registers Affected
Clears these individual register bits	0x0000 Master Configuration Register, RTSTEX bit 4
	0x0006 Hardware Pending Int Register, RTIP bit 2
	0x0006 Hardware Pending Int Register, RTAPF bit 3
	0x0009 RT Pending Int Register, RT int bits 8 – 3
Clears these entire register addresses	0x0018 RT Operational Status Register (see Note 1)
	0x001A RT MIL-STD-1553 Status Word Bits Register
	0x001E RT Built-In Test Word Register
	0x0049 RT Time Tag Counter

To start RT after soft reset completion (indicated by READY signal assertion), the host must set RTSTEX bit 4 in register 0x0000.

Note 1: A software reset will NOT clear LOCK bit 9 in “Remote Terminal Operational Status Register (0x0018)”. If the LOCK bit is set, it may be cleared by asserting the $\overline{\text{RESET}}$ master reset input signal.

21.2.2. Bus Monitor SMT

This is initiated when MTRESET bit 12 is set in the "Master Status and Reset Register (0x0001)". The following actions are performed.

Table 22. SMT Soft Reset Summary

Action	Registers Affected
Clears these individual register bits	0x0000 Master Configuration Register, MTENA bit 8
	0x0006 Hardware Pending Int Register, MTIP bit 1
Clears these entire register addresses	0x0008 SMT Pending Interrupt Register
	0x0030 SMT Next Message Buffer Address Pointer
	0x0031 SMT Last Message Buffer Address Pointer
	0x003A SMT Time Tag Counter, Low
	0x003B SMT Time Tag Counter, Mid
	0x003C SMT Time Tag Counter, High

Note: Soft reset for the Bus Monitor re-initializes the buffer address pointers, but does not clear the allocated buffer space in the buffer(s).

To start the Bus Monitor after soft reset completion (indicated by READY signal assertion), the host must set MTENA bit 8 in register 0x0000.

21.3. MIL-STD-1760: Busy Status Assertion After Power-Up

A MIL-STD-1760 RT must be able to respond on the bus within 150ms following power turn-on. Between power-on and 150ms, it is acceptable for the RT to respond with the "Busy" bit set in the RT Status Word (see Section 17.6 on page 128). This indicates the RT is awake but not ready to transfer data. Alternatively, the RT may respond Clear Status with valid data.

In order to engage 1760 mode, the MODE1760 input signal is asserted during a hardware reset. Following IP Dongle verification, the MODE1760 input status will be latched 200ns after the rising edge of $\overline{\text{RESET}}$ (the same time as the RT address). During 1760 mode, the IP core will respond to any valid command (with matching RT address) with the BUSY bit set in the status word. No data words will be transmitted and no interrupts or logging of data will occur. Mode 1760 operation may be confirmed by the host by reading Mode 1760 Status bit 7 in "Master Configuration Register 2 (0x004E)".

Within 500ms following power turn-on, the MIL-STD-1760 RT must respond with data as defined by the MIL-STD-1760 standard, with "Busy" status bit reset. The RT host processor must be fully operational at this time. After system initialization is complete, the host can deactivate 1760 mode by writing "1" to bit 7 in "Master Configuration Register 2 (0x004E)". Alternatively, bit 4 RTSTEX and bit 6 RTENA in "Master Configuration Register 1 (0x0000)" may be written "1" by the host.

22. SELF-TEST

The HI-6300 provides several host-directed RAM self-tests, as well as an automatic (but optional) RAM self-test performed after Master Reset. In addition, on-line analog and off-line digital transmit/receive loopback tests are provided, with different options for BC and RT terminal modes.

22.1. Host-Directed Self-Test

The IP core supports host-directed RAM self-test (sometimes called RAM built-in self-test, or RAM BIST) and single-word transmit/receive loopback, which may be off-line digital or on-line analog. Host-directed self-test is configured and operated using register read/write operations. The host selects the size of the RAM (8K or 64K) to undergo self-test by writing bit 8 RAM_SZE in "IP Core Configuration Register (0x0004)".

The host initiates self-test mode by asserting the TEST input signal to logic 1. When the TEST signal is high, four registers are active for performing RAM self-test or RT mode loopback self-tests:

22.1.1. Self-Test Control Register (0x0028)

FRAMA	RBFFAIL	RBSEL2	RBSEL1	RBSEL0	RBSTRT	RBPASS	RBFAIL	Not Used	Not Used	LBALOG	LBSYNC	LBBUSEL	LBSTART	LBPASS	LBFAIL	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW						R				RW				R		Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The function of this register is multiplexed by the IP core TEST input signal. When the TEST signal is logic 0 (normal operating mode), register address 0x0028 has no function.

When the TEST input signal is logic 1, register address 0x0028 functions as the Self-Test Control Register, a Read-Write register used for RAM memory testing, or analog or digital loopback tests. Bits 0, 1, 8, 9 are Read-Only. The remaining bits in this register are Read-Write.

After test completion, the TEST input signal should be reset to logic 0, restoring all register bits to Read-Write.

Descriptions below apply when the TEST input signal is logic 1; the register is operating as the Self-Test Control Register.

This register supports two types of test: Register bits 15 - 8 are used for RAM built-in self test (RAM BIST). Register bits 5 − 0 are used for transceiver loopback testing (either digital loopback or analog loopback).

Self-Test Control Register bits 15 - 8 provide a means for the host to perform RAM self-test at other times. Register bits 13:11 select RAM test type. Then bit 10 assertion starts the selected RAM test, and bits 9-8 report a pass/fail result after test completion. All tests are destructive, overwriting data present before test commencement.

NOTE: '*Reset*' refers to bit value following <u>either</u> Master Reset ($\overline{\text{RESET}}$) <u>or</u> software reset.

<table>
<tr><th><i>Bit No.</i></th><th><i>Mnemonic</i></th><th><i>R/W</i></th><th><i>Reset</i></th><th colspan="4"><i>Function</i></th></tr>
<tr><td>15</td><td>FRAMA</td><td>R/W</td><td>0</td><td colspan="4">Full RAM Access Enable.
During normal operation, some bits in certain RAM locations (e.g., Descriptor Table Control Words) cannot be written by the host. When the FRAMA bit is asserted, host writes to RAM are unrestricted to permit full testing. During normal completion, this bit must be reset to logic 0.</td></tr>
<tr><td>14</td><td>RBFFAIL</td><td>R/W</td><td>0</td><td colspan="4">RAM BIST Force Failure.
When this bit is asserted, RAM test failure is forced to verify that RAM BIST logic is functional.</td></tr>
<tr><td rowspan="11">13,12,11</td><td rowspan="11">RBSEL2:0</td><td rowspan="11">R/W</td><td rowspan="11">0</td><td colspan="4">RAM BIST Select Bits 2-0.
This 3-bit field selects the RAM BIST test mode applied when the RBSTART bit is set:</td></tr>
<tr><td rowspan="2"><b>RBSEL2:0</b></td><td rowspan="2"><b>Selected RAM Test</b></td><td colspan="2"><b>Test Time
(# of Clocks)</b></td></tr>
<tr><td><b>64K RAM</b></td><td><b>8K RAM</b></td></tr>
<tr><td>000</td><td>Idle</td><td>-</td><td>-</td></tr>
<tr><td>001</td><td>Pattern Test, described below</td><td>5,898,244</td><td>737,284</td></tr>
<tr><td>010</td><td>Write 0x0000 to RAM address range:
0x0000 − 0xFFFF (64K RAM)
0x0000 − 0x1FFF (8K RAM)</td><td>65,541</td><td>8,197</td></tr>
<tr><td>011</td><td>Read and verify 0x0000 over RAM address range:
0x0000 − 0xFFFF (64K RAM)
0x0000 − 0x1FFF (8K RAM)</td><td>196,613</td><td>24,581</td></tr>
<tr><td>100</td><td>Write 0xFFFF to RAM address range:
0x0000 − 0xFFFF (64K RAM)
0x0000 − 0x1FFF (8K RAM)</td><td>65,541</td><td>8,197</td></tr>
<tr><td>101</td><td>Read and verify 0xFFFF over RAM address range:
0x0000 − 0xFFFF (64K RAM)
0x0000 − 0x1FFF (8K RAM)</td><td>196,613</td><td>24,581</td></tr>
<tr><td>110</td><td>Inc / Dec Test performs only steps 5 − 8 of the Pattern Test below</td><td>524,293</td><td>65,541</td></tr>
<tr><td>111</td><td>Idle</td><td>-</td><td>-</td></tr>
</table>

Bit No.	Mnemonic	R/W	Reset	Function
13,12,11	RBSEL2:0 (continued)	R/W	0	**Description of the 64K RAM BIST "PATTERN" test, selected when register bits RBSEL2:0 = 001 (For 8K RAM, address range is 0x0000 - 0x1FFF):** *Note: Test read /write accesses to addresses 0x0000 - 0x0051 involve 82 RAM locations not accessible to the host. These accesses do not affect the host-accessible registers, overlaying the same address range.* 1. Write 0x0000 to all RAM locations, 0x0000 through 0xFFFF. 2. Repeat the following sequence for each RAM location from 0x00000 through 0xFFFF: a. Read and verify 0x0000 b. Write then read and verify 0x5555 c. Write then read and verify 0xAAAA d. Write then read and verify 0x3333 e. Write then read and verify 0xCCCC f. Write then read and verify 0x0F0F g. Write then read and verify 0xF0F0 h. Write then read and verify 0x00FF i. Write then read and verify 0xFF00 j. Write 0x0000 then increment RAM address and go to step (a) 3. Write 0xFFFF to all RAM locations, 0x0000 through 0xFFFF 4. Repeat the following sequence for each RAM location from 0x00000 through 0xFFFF: a. Read and verify 0xFFFF b. Write then read and verify 0x5555 c. Write then read and verify 0xAAAA d. Write then read and verify 0x3333 e. Write then read and verify 0xCCCC f. Write then read and verify 0x0F0F g. Write then read and verify 0xF0F0 h. Write then read and verify 0x00FF i. Write then read and verify 0xFF00 j. Write 0xFFFFthen increment RAM address and go to step (a) 5. Write each cell's memory address into each RAM location from 0x00020 to 0xFFFF. 6. Read each memory location from 0x00000 to 0xFFFF and verify it contains its address. 7. Write 1s complement of each cell's memory address, into each RAM location (same addr range). 8. Read each memory location and verify it contains the 1s complement of its address. **NOTE: RAM ECC bits are overwritten with incrementing test data during the RAM BIST test.** Therefore, reading the RAM immediately following BIST, before new data is stored, may result in erroneous ECC corrections. If it is a requirement to read RAM immediately following BIST, it is recommended to write the entire address range with 0x0000 (write bits RBSEL2:0 = 010) prior to reading the RAM.

Bit No.	Mnemonic	R/W	Reset	Function
10	RBSTRT	R/W	0	RAM BIST Start. Writing logic 1 to this bit initiates the RAM BIST test selected by register bits RBSEL2:0. The RBSTRT bit can only be set if the TEST input signal is high and if register bit 15 is already asserted. This bit is automatically cleared upon test completion. Register bits 9:8 indicate pass / fail test result.
9	RBPASS	R	0	RAM BIST Pass. IP Core logic asserts this bit when the selected RAM test completes without error. This bit is automatically cleared when RBSTRT bit 10 is set.
8	RBFAIL	R	0	RAM BIST Fail. IP Core logic asserts this bit when failure occurs while performing the selected RAM test. This bit is automatically cleared when RBSTRT bit 10 is set. When BIST failure occurs, the first failed RAM address can be read in “RAM Self-Test Fail Address Register (0x001B)”. When the TEST input signal is logic 0, register address 0x001B function reverts to the read-only “Remote Terminal Current Message Information Word Register (0x001B)”.
7,6	-----	R	0	Not Used. These bits cannot be set. A READ will return 0-0.
5	LBALOG	R/W	0	Loopback Test Analog. The IP core supports either digital or analog loopback testing for either bus transceiver. When the LBALOG bit is low, digital loopback is selected and no data is transmitted onto the selected external MIL-STD-1553 bus. When the LBALOG bit is high, analog loopback is selected and a test word is transmitted onto and received from the selected external MIL-STD-1553 bus.
4	LBSYNC	R/W	0	Loopback Test Word Sync Select. When the LBSYNC bit is high, the loopback test word is transmitted with command sync. When the LBSYNC bit is low, the loopback test word is transmitted with data sync.
3	LBBUSEL	R/W	0	Loopback Test Bus Select. When this bit is low, loopback testing occurs on Bus A. When this bit is high, loopback testing occurs on Bus B.
2	LBSTART	R/W	0	Loopback Test Start. Writing logic 1 to this bit initiates the loopback test selected by register bits 3, 4 and 5. The LBSTRT bit can only be set if the external TEST signal is already asserted, and is automatically cleared upon test completion. Register bits 1,0 indicate pass / fail test result.
1	LBPASS	R	0	Loopback Test Pass. IP Core logic asserts this bit when the selected RAM test completes without error. This bit is automatically cleared when LBSTART bit 2 is set.
0	LBFAIL	R	0	Loopback Test Fail. IP Core logic asserts this bit when failure occurs while performing the selected loopback test. Failure is comprised of Manchester encoding error, parity error, wrong sync type or data mismatch. This bit is automatically cleared when LBSTART bit 2 is set.

22.1.2. Loopback Test Transmit Data Register (0x001F)

MSB				Loopback Test Transmit Data Register											LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is not affected by SRST software reset. The function of this register is multiplexed by the IP core TEST input signal. When TEST is logic 0 (normal operating mode), register address 0x001F is the “Remote Terminal Alternate Built-In Test (BIT) Word Register (0x001F)”.

When the TEST input signal is logic 1, register address 0x001F becomes the Loopback Test Transmit Data Register, a Read-Write register used for analog or digital loopback tests. When a loopback test is performed, the value in this register is transmitted, and should appear in the Loopback Test Receive Data Register 0x0002. See Section “22.1.1. Self-Test Control Register (0x0028)” on page 195 (bits 0-5) for additional information.

After test completion, the TEST input signal should be reset to logic 0. The host should restore the desired alternate BIT Word value for the RT at register address 0x001F.

22.1.3. Loopback Test Receive Data Register (0x0002)

MSB				Loopback Test Receive Data Register											LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

This register is cleared after $\overline{\text{RESET}}$ signal master reset, but is not affected by SRST software reset. The function of this register is multiplexed by the IP core TEST input signal. When the TEST input signal is logic 0 (normal operating mode), register address 0x0002 is the “Remote Terminal Current Command Register (0x0002)”.

When the TEST input signal is logic 1, register address 0x0002 becomes the Loopback Test Receive Data Register, a read-only register used for analog or digital loopback tests. When loop back is performed, the value in the Loopback Test Transmit Data Register 0x001F is transmitted and should appear in this register. See Section “22.1.1. Self-Test Control Register (0x0028)” on page 195 (bits 0-5) for additional information.

After test completion, the TEST input signal should be reset to logic 0, reverting this register address 0x0002 to the read-only “Remote Terminal Current Command Register (0x0002)”. The contained register value will not have meaning until the RT receives its next valid command.

22.1.4. RAM Self-Test Fail Address Register (0x001B)

MSB				RAM BIST Fail Address Register											LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The function of this register is multiplexed by the IP core TEST input signal. When the TEST input signal is logic 0 (normal operating mode), this register address is the “Remote Terminal Current Message Information Word Register (0x001B)”. When the TEST input signal is logic 1, register address 0x001B becomes the RAM Self-Test Fail Address Register.

Upon test completion, Self-Test Control Register bit 9 (see Section “22.1.1. Self-Test Control Register (0x0028)” on page 195) is set if the test passed, otherwise bit 8 is set if the test failed. If failure occurs, the first failed RAM address is written to the “RAM Self-Test Fail Address Register (0x001B)”. Memory test fail also asserts the BTMF (BIST

Memory Test Fail) bit 3 in “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.

At test completion, the host should clear the Self-Test Control Register 0x0028, and then reset the TEST input signal to logic 0.

22.1.5. Host-Directed RAM Self-Test

Host-directed RAM self-test overwrites preexisting RAM contents and should only be performed when complete re-configuration of the RAM address space will occur after RAM test completion. If the IP core is operational, terminal execution should be stopped. If asserted, reset bits 4-8 and bit 12 in “11.1. Master Configuration Register 1 (0x0000)”, stopping BC, MT, and RT. Assert the TEST signal to activate register 0x0028 as the Self-Test Control Register.

After asserting the TEST input signal, RAM self-test is configured and started by writing bits 15:10 in the Self-Test Control Register, described on page 195. Register bits 13:11 select one of the five test protocols. Register bit 15 is usually set to provide unrestricted RAM read/write access. Register bit 10 is then asserted to start the RAM test selected by bits 13:11. All of these bits may be written simultaneously, and bits 7:0 should be written as zeros. Test time varies based on complexity; test times are shown in the Self-Test Control Register 0x0028 description.

Upon RAM test completion, “Self-Test Control Register (0x0028)” bit 9 is set if the RAM test was successful, otherwise bit 8 is set if the test failed. If failure occurs, the first failed RAM address is written to the “RAM Self-Test Fail Address Register (0x001B)”. Memory test fail also asserts the BTMF (BIST Memory Test Fail) bit 3 in “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.

At RAM test completion, the host should clear the Self-Test Control Register 0x0028, reset the TEST input signal to logic 0, then re-initialize registers and RAM, and finally restart terminal execution.

22.1.6. Host-Directed RT-Mode Loopback Testing (On-Line Analog or Off-Line Digital)

RT mode loopback testing involves transmission and reception of a single Manchester-encoded word with correct parity. On-line Analog Loopback transmits the specified test word onto the external MIL-STD-1553 bus. The internal receiver for the same bus is totally independent from the encoder logic used for bus transmission. The bus receiver detects and decodes the received replica of the transceiver’s own transmission. Off-line Digital Loopback does not disturb the selected MIL-STD-1553 bus; the digital signal paths used for encoding and transmission (as well as reception and decoding) are fully tested without involving the external MIL-STD-1553 bus; only the analog bus driver and analog receiver are bypassed for digital loopback tests. The HI-6300 cannot be configured for loopback transmitting on one bus and receiving on the other bus.

RT mode loopback testing requires one or both RTs to be enabled. If the RT is not already running, set the RTENA bit in “11.1. Master Configuration Register 1 (0x0000)”. Then set the RTSTEX bit in “Master Configuration Register 1 (0x0000)”. Then initiate test mode by asserting the TEST input signal to logic 1. **Note, for digital loopback testing, bit 0 BCLBK in “Master Configuration Register 2 (0x004E)” must be asserted to prevent contention with external bus activity.**

Write a 16-bit transmit value to the “Loopback Test Transmit Data Register (0x001F)”. With the TEST input set to logic 1, the host can write bits 5:3 in the Self-Test Control Register 0x0028 to select analog or digital loopback, command sync or data sync, and select test Bus A or Bus B. Then, without modifying bits 5:3, write the Self-Test Control Register again to set bit 2, starting loopback test. Note: Self-Test Control Register bits 5:2 can be written simultaneously; the remaining register bits 15:6 and 1:0 should all be written as zeros.

After 20µs or so, RT loopback transmission is complete. Self-Test Control Register bit 1 is set for successful loopback test, otherwise bit 0 is set if loopback failed. The received word has been written into the “Loopback Test Receive Data Register (0x0002)”. It should match the value in the “Loopback Test Transmit Data Register (0x001F)”. Test failure also asserts either the LBFA (Loopback Fail A) bit 5 or LBFB (Loopback Fail B) bit 4 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.

At RT loopback test completion, the host should clear Self-Test Control Register 0x0028, and then reset the TEST input signal to logic 0.

22.1.7. Programmed BC-Mode Digital Loopback Testing (Off-Line)

For any BC message block, off-line digital loopback self-test can be programmed. The TEST input signal should be logic 0, normal operational state. BC transmission onto the 1553 bus is inhibited for the message, but the digital transmit encoding and receive decoding signal paths can be checked.

The BC Instruction List in RAM comprises a series of 2-word entries, an instruction Op Code Word followed by a Parameter Word. While sequencing through the Instruction List, the BC control logic fetches and executes conditional and unconditional instruction op codes referenced by the “Bus Controller (BC) Instruction List Pointer (0x0034)”. For executable messages, the Parameter Word following the Op Code Word contains the starting address of a Message Control/Status Block.

As described in Section “12.4. Bus Controller Message Control / Status Blocks” on page 67, each Message Control/Status Block begins with a BC Control Word. When Control Word SELFTST bit 6 is set, off-line self-test is enabled, inhibiting transmission onto the 1553 bus. Instead the output of the bus Manchester II serial encoder is routed directly to the decoder input for the bus selected by Control Word bit 7 (USEBUSA). A validity check is performed on the received replica of each transmitted word (sync, encoding, bit count and parity). As received, each word replica is stored in the Loopback Word location in the Message Control/Status Block. The data value for the final word received is also checked with the transmitted final word. If any word fails validity check (or if the final word has data mismatch) test logic sets the LBE (loopback error) bit 8 in the Block Status Word.

After message processing, off-line self-test success or failure can be determined by reading the received Loopback Word (stored in the Message Control/Status Block if BC is using 16-bit time base) or by reading the LBE (loopback error) bit 8 in the Block Status Word. (Note: If the BC is using 32-bit time base, the final received loopback word in the Loopback Word location is overwritten at the end of message post-processing when time tag bits 31:16 are written there.)

The “BADMSG” BC condition code 0xC is updated based on the outcome of the off-line SELFTST loopback message. BADMSG is set to logic 1 for Loopback Test error. This permits conditional execution, including jumps or subroutine calls, based on the outcome of the message having SELFTST asserted in its Control Word.

The BADMSG condition code is also set for Format Error or No Response error, but is not affected by a Status Set condition. For non-broadcast commands using off-line SELFTST loopback, No Response error always occurs since the BC message processor expects an RT response. Since BC bus transmission is inhibited, off-line SELFTST loopback should use broadcast commands. This avoids BADMSG condition codes caused by No Response error.

22.1.8. Continuous BC-Mode Analog Loopback Testing (On-Line)

The BC performs continuous analog loopback on all Bus Controller transmissions when executing normal Message Control/Status Blocks having off-line SELFTST bit = 0 in each Control Word. The TEST input signal is logic 0, normal operational state. For each Manchester II word transmitted by the BC, a validity check is performed on the received replica, checking sync, encoding, bit count and parity. In the Message Control/Status Block, each received word replica is stored in the Loopback Word location when decoded, overwriting the previous word stored there. The data value for the final BC word received is also checked for data value. If any word fails validity check (or if the final word has data mismatch) test logic sets the LBE (loopback error) bit 8 in the Block Status Word. (Note: If the BC is using 32-bit time base, the final word replica in the Loopback Word location is overwritten at the end of message post-processing when time tag bits 31:16 are written there.)

The “BADMSG” BC condition code 0xC is updated based on the outcome of continuous BC-mode analog loopback checking. The BADMSG condition code is set for Loopback Error, Format Error or No Response error, but is not affected by a Status Set condition. BC analog loopback failure also sets LBE loopback error bit 8 in the Block Status Word.

22.1.9. MIL-STD-1553 Terminal Fail-Safe Timer

BUS A and BUS B transmitters both have continuously running watchdog timers that prevent continuous transmission beyond 663µs. See description of bit 2, WDTXINH (Watchdog Timer Transmitter Timeout Transmit Inhibit) described in “Master Configuration Register 2 (0x004E)”. The host may test the transmitter timeout protection feature by setting TXTEST bit 15 in “Master Configuration Register 2 (0x004E)”.

23. APPENDIX: RT MESSAGES RESPONSES, OPTIONS & EXCEPTIONS

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Invalid Command Word (Manchester, parity or bit count error)	No terminal response, the message is ignored. No Status Word change.	No change	No Message Info Word is written	None
Any valid command to RT31 (broadcast). when the BCSTINV bit in the RT Configuration Register equals 1.	No terminal response, the message is ignored. No Status Word change. (Broadcast commands are rendered invalid.)	No change	No Message Info Word is written	None
RT Address Parity Error based on RTADR and RTADP bits in the Operational Status Register	For commands to the RT's own address or to broadcast address RT31: No terminal response, message is ignored. No Status Word change.	No change	No Message Info Word is written	RTAPF (not optional)
Any valid non-mode (subaddress 1-30) transmit command to RT31 (undefined broadcast transmit).	No terminal response, Set Message Error (ME) and BCR status bits.	DBAC bit set. DPB bit toggles. BCAST bit set.	MERR bit set. BUSID bit updated.	IWA IBR (IXEQZ)
Any valid non-mode (subaddress 1-30) transmit command except for RT31. The corresponding bit in the Illegalization Table equals 0.*	Normal Status Word response (Clear Status). Data words for transmit are read from the RAM data buffer assigned by the Descriptor Table entry for the transmit subaddress.	DBAC bit set. DPB bit toggles. BCAST bit reset.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit updated. (Other error bits reset).	IWA IBR (IXEQZ)
Any valid non-mode (subaddress 1-30) transmit command except for RT31. The corresponding bit in the Illegalization Table equals 1. **	Assert Message Error (ME) status, then transmit ME Status Word without following data words.	DBAC bit set. DPB bit toggles. BCAST bit reset.	ILCMD bit set. BUSID bit updated. MERR bit set. RTRT bit updated. (Other error bits reset).	ILCMD IWA
Any valid non-mode (subaddress 1-30) receive command. The corresponding bit in the Illegalization Table equals 0. *	Normal Status Word response (Clear Status). After message completion, the data words received are stored in the data buffer RAM location assigned by the Descriptor Table entry for the receive subaddress.	DBAC bit set. DPB bit toggles. BCAST bit reset.	Normal update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit updated. (Other error bits reset).	IWA IBR (IXEQZ)

* Terminal is using "illegal command detection" and command is legal
OR terminal is not using "illegal command detection" and command may be legal or illegal (in form response).

** Terminal is using "illegal command detection" and command is illegal.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Any valid non-mode (subaddress 1-30) receive command. The corresponding bit in the Illegalization Table equals 1. **	Assert Message Error (ME) status and set BCR if broadcast. Any received data words are ignored and are not saved. When data reception stops, transmit Status Word.	DBAC bit set. DPB bit toggles. BCAST bit updated.	ILCMD bit set. BUSID bit updated. MERR bit set. RTRT bit updated. (Other error bits reset)	ILCMD IWA IBR (IXEQZ)
Valid receive command followed by invalid data word (Manchester, parity or bit count error).	No terminal response. Set Message Error (ME) status. If broadcast (RT31), also set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. BUSID bit updated. IWDERR bit set. ILCMD bit reset. RTRT bit updated (Other error bits reset).	MERR IWA IBR
Valid receive command followed by one or more good data words, then a data word having Command Sync.	No terminal response. Set Message Error (ME) status. If broadcast (RT31), also set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. BUSID bit updated. SYNERR bit set. ILCMD bit reset. (Other error bits reset).	MERR IWA IBR
Any valid command followed by wrong number of data words (too few or too many words)	No terminal response. Set Message Error (ME) status. If broadcast (RT31), also set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. BUSID bit updated. Set WCTERR (too few) or GAPERR (too many). ILCMD bit reset. RTRT bit updated. (Other error bits reset).	MERR IWA IBR
RT-RT where CW1 is a valid non-mode receive command. CW2 is a non-mode transmit command valid for different RT. (Normal RT-RT receive message)	Normal Status Word response (Clear Status). If RT-RT Command Word 1 is broadcast (RT31) set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. RTRT bit set. RTCWERR bit reset. ILCMD bit reset. (All error bits reset).	IWA IBR (IXEQZ)
RT-RT where CW1 is a valid non-mode receive command. Transmit command CW2 has an error: $T/\overline{R}$ bit = 0, or CW2 subaddress equals 0 or 31 (mode code), or CW2 has same RT address as CW1.	No terminal response. Set Message Error (ME) status. If RT-RT Command Word 1 is broadcast (RT31) also set the BCR status bit,	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. BUSID bit updated. RTRT bit set. RTRTCWERR bit set. ILCMD bit reset. (Other error bits reset).	MERR IWA IBR

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
RT-RT where CW1 is a valid non-mode receive command. CW2 is valid for different RT but transmitting RT does not respond in time.	No terminal response. Set Message Error (ME) status. If RT-RT Command Word 1 is broadcast (RT31), also set the BCR status bit.	DBAC bit set. BCAST bit updated DPB bit toggles.	MERR bit set. BUSID bit updated. RTRT bit set. TMOERR bit set. ILCMD bit reset. (Other error bits reset).	MERR IWA IBR
RT-RT receive command (CW1 is valid). The transmitting RT response has one of these errors: invalid word (Manchester, (sync, bit count, parity or word count error). Also includes transmitting RT response with Message Error or Busy status followed by no data words.	No terminal response. Set Message Error (ME) status. If RT-RT Command Word 1 is broadcast (RT31) also set the BCR status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. BUISID bit reset. RTRT bit set. IWDERR bit set, or WCTERR bit set for Tx RT Busy case. ILCMD bit reset. (Other error bits reset).	MERR IWA IBR
RT-RT command where CW2 is a valid non-mode (subaddress 1-30) transmit command. CW1 is a non-mode receive command for RT31. (Normal broadcast RT-RT transmit)	Normal Status Word response. Clear status is transmitted with the commanded number of data words. Data words for transmit are read from the RAM data buffer assigned in the Descriptor Table entry for the transmit subaddress.	DBAC bit set. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit set. (All error bits reset).	IWA (IXEQZ)
Valid mode code command to RT31 (broadcast). The BCSTINV bit in the RT Configuration Register equals 1.	No terminal response, the message is ignored. No Status Word change.	No change	No Message Info Word is written	None
Valid undefined mode code command. The UMCINV bit in the RT Configuration Register equals 1.	No terminal response, the message is ignored. No Status Word change. **NOTE:** This only applies for the undefined mode codes: MC0 to MC15 with T/$\overline{R}$ = 0 MC16,18 & 19 with T/$\overline{R}$ = 0 MC17,20 & 21 with T/$\overline{R}$ = 1	No change	No Message Info Word is written	None

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Valid defined mode code command (including reserved mode code) not “illegalized” by Illegalization Table (table bit equals 0 *)	If MC2 (transmit status) or MC18 (transmit last command) status word from last command is transmitted. If MC18, data word transmitted is read from an internal register. **OR** If not MC2 or MC18, normal Status Word response. If broadcast, assert Status Word BCR bit. For mode codes 16-31 with T/$\overline{R}$ bit = 1 which transmit a data word, the word for transmit is read from the Mode Command Data Table. **AND** For all mode commands with mode data word (mode codes 16-31), the transmitted or received data word is written to command’s Descriptor Word 4.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR (IXEQZ)
Valid defined mode code command that is “illegalized” by the IllegalizationTable (table bit equals 1 **)	Set Message Error (ME) status. If not broadcast (RT31), transmit. Status Word without a following mode data word. If broadcast (RT31), also assert the BCR status bit. **AND** For mode commands with a mode data word (mode codes 16-31), no updates are made to the Mode Command Data Table or to the command’s Word 4 in Descriptor Table.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. BUSID bit updated. MERR bit reset. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR

* Terminal is using “illegal command detection” and command is legal
OR terminal is not using “illegal command detection” and command may be legal or illegal (in form response).

** Terminal is using “illegal command detection” and command is illegal.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Valid undefined mode code command. The UMCINV bit in the RT Configuration Register equals 0.	If bit in Illegalization Table that corresponds to the undefined mode code command equals 1 ** Set Message Error (ME) status, If not broadcast (RT31), transmit Status Word without a following mode data word. If broadcast (RT31), also assert the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. BUSID bit updated. MERR bit reset. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
	OR If bit in Illegalization Table that corresponds to the undefined mode code command equals 0 * Normal Status Word (Clear Status) response. If command was broadcast (RT31), assert the BCR status bit. **AND** For mode codes 16-31 with $T/\overline{R}$ bit = 1 which transmit a data word, the word for transmit is read from the Mode Command Data Table. **AND** For all mode commands with mode data word (mode codes 16-31), the transmitted or received data word is written to command's Descriptor Word 4.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR (IXEQZ)

* Terminal is using "illegal command detection" and command is legal OR terminal is not using "illegal command detection" and command may be legal or illegal (in form response).

** Terminal is using "illegal command detection" and command is illegal.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Valid receive command followed by invalid data word (Manchester, parity or bit count error).	No terminal response. Set Status Word ME bit, If broadcast, also set Status Word BCR bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. BUSID bit updated. IWDERR bit set. ILCMD bit reset. RTRT bit updated. (Other error bits reset.)	MERR IWA IBR
Superseded Message: Terminal receives an incomplete message interrupted by a gap of at least 3.5 us, followed by a new valid command on the same bus or on the other bus **OR** Terminal is transacting a transmit message on one bus and receives the start of a valid command on the other bus.	Terminal aborts processing for first message and responds in full to the second (superseding) message. The Status Word BCR bit reflects broadcast status for: the second command, unless second command is MC2 (transmit status) or MC18 (transmit last command).	No change to superseded command's Control Word. For superseding command's Control Word: DBAC bit set. BCAST bit updated DPB bit toggles.	No Msg Info Word written for the superseded command. For superseding command's data buffer, a normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit updated. (All error bits reset.)	None for super-seded command IWA IBR (IXEQZ)
Terminal is Busy for a valid receive command either globally (BUSY bit set in Status Word Bits register) or in response to a particular valid receive command (MKBUSY bit set in the command's Descriptor Table control word.)	Busy bit is set in the 1553 Status Bits register. Status Word is transmitted, unless broadcast. If broadcast, the BCR bit in Status Word is also set. After message completion, data words received are stored in the data buffer assigned by the receive subaddress Descriptor Table entry.	DBAC bit set. BCAST bit updated. DPB bit toggles.	WASBSY bit set. BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit updated. (All error bits reset.)	IWA IBR
Terminal is Busy for a valid transmit command either globally (BUSY bit set in Status Word Bits register) or in response to a particular valid receive command (MKBUSY bit set in the command's Descriptor Table control word.)	Busy bit is set in the 1553 Status Bits register. If not broadcast, Status Word is transmitted without data. If broadcast, the BCR bit in Status Word is also set.	DBAC bit set. BCAST bit updated, (mode commands with $T/\overline{R} = 1$) DPB bit toggles	WASBSY bit set. BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit updated. (All error bits reset.)	IWA IBR

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
DYNAMIC BUS CONTROL (MC0): Mode code command with mode code 00000 and T/$\overline{R}$ bit equals 1	See Dynamic Bus Control Enable, DBCENA bit 10 in “Remote Terminal Configuration Register (0x0017)” on page 119.			
The mode code’s bit in Illegalization Table equals 0 *	RT is not using “illegal command detection.” Respond “in form”: Reset Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
OR The mode code’s bit in Illegalization Table equals 1 **	**OR** RT is using “illegal command detection”and mode code is illegalized. Set Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit reset. DPB bit toggles.	ILCMD bit set. BUSID bit updated. MERR bit reset. RTRT bit reset. (Other error bits reset.)	ILCMD IWA
MC0 EXCEPTIONS:				
Broadcast address RT31 (broadcast not allowed)	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB toggles.	MERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD, RTRT bits reset. (Other error bits reset.)	MERR IWA
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (MC0 is not ndefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
$T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 0. Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 0. Illegalization Table bit equals 1 **	Set Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit reset. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
SYNCHRONIZE WITHOUT DATA (MC1): **Mode code command with mode code 00001 and $T/\overline{R}$ bit equals 1**	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set the Status Word BCR bit and suppress Status Word transmit. Reset the Time Tag counter to 0x0000.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
MC1 EXCEPTIONS:				
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response. The Time Tag counter is not reset.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response. The Time Tag counter is not reset.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
TRANSMIT STATUS (MC2): **Mode code command with mode code 00010** **and $T/\overline{R}$ bit equals 1**	No Status Word updates, Transmit Status from last valid command (assuming last command was not a "Transmit Status" or a "Transmit last Command" mode command.	DBAC bit set. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
MC2 EXCEPTIONS:				
Broadcast address RT31 (broadcast not allowed)	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB bit toggles.	MERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0 The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response. Time Tag counter is not reset.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0 The Illegalization Table. bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
INITIATE SELF TEST (MC3): Mode code command with mode code 00011 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set the Status Word BCR bit and suppress status transmit. Host should initiate self-test then update Built-In Test word at shared RAM address 0x0093. Resume terminal execution.	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
MC3 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
TRANSMITTER SHUTDOWN (MC4): Mode code command with mode code 00100 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set Status Word BCR bit and suppress status. transmit. After Status transmission, inhibit the inactive bus:	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
The IP core automatically shuts down either **transmit and receive** or **transmit only** for the inactive bus, depending on the state of the BSDTXO bit in the Master Configuration Register. Refer to the description of the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” for further information. When MC4 results in transmitter shutdown, the condition is reflected by assertion of a TXASD or TXBSD bit in the corresponding “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. If BSDTXO equals logic 0, an RXASD or RXBSD bit will also be asserted, indicating full bus shutdown (transmit and receive). Once shutdown, the inactive bus transmitter (or transmitter and receiver) can be reactivated by an “Override Transmitter Shutdown” MC5 or MC21 or “Reset Remote Terminal” MC8 mode code command, or by software reset (initiated by setting the RTRESET bit in the “Master Status and Reset Register (0x0001)”) or by hardware reset initiated by asserting the $\overline{\text{RESET}}$ Master Reset input signal.				
MC4 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
OVERRIDE TRANSMITTER SHUTDOWN (MC5): Mode code command with mode code 00101 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set the Status Word BCR bit and suppress status transmit. This command is only used with dual redundant buses. After Status transmission, reactivate inactive bus:	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
The IP core automatically re-enables **transmit and receive** for the inactive bus, without considering BSDTXO bit status in the Master Configuration Register. The IP core affirms re-enabled bus status by resetting all four TXASD, TXBSD, RXASD and RXBSD "shutdown status" bits in the "Remote Terminal Built-In Test (BIT) Word Register (0x001E)". ***Note:*** *If the TXINHA or TXINHB input signals are asserted, the IP core cannot override the resulting hardware transmit inhibit for the affected bus. In this case, the corresponding TXASD and/or TXBSD bits remain high. See Built-In Test Register description for further information.*				
MC5 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
INHIBIT TERMINAL FLAG BIT (MC6): Mode code command with mode code 00110 and T/$\overline{\text{R}}$ bit equals 1	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set the Status Word BCR bit and suppress status transmit.	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
The IP core automatically sets the TF Inhibit bit in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. While the TF inhibit bit is set, the IP core disregards assertion of the Terminal Flag (TF) bit in the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” and only transmits status with the Terminal Flag status bit reset. Once the Terminal Flag has been inhibited, it can be reactivated by an “Override Inhibit Terminal Flag” MC7 or “Reset Remote Terminal” MC8 mode command, by software reset (asserting the SRST bit in the “Remote Terminal Configuration Register (0x0017)”) or by asserting the $\overline{\text{RESET}}$ master reset input signal.				
MC6 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{\text{R}}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{\text{R}}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{\text{R}}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{\text{R}}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
OVERRIDE INHIBIT TERMINAL FLAG BIT (MC7): **Mode code command with mode code 00111 and $T/\overline{R}$ bit equals 1**	Default response: Reset Message Error (ME) status then transmit Status Word. If broadcast, set the Status Word BCR bit and suppress status transmit.	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
The IP core automatically resets the TF Inhibit bit in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. While the TF inhibit bit is reset, the IP core transmits status with the Terminal Flag status bit set if the Terminal Flag (TF) bit is asserted in the “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”.				
MC7 EXCEPTIONS:				
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
RESET REMOTE TERMINAL (MC8): Mode code command with mode code 01000 and $T/\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status. If not broadcast, transmit Status Word.	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA IBR
After Status transmission, the IP core automatically resets the status Message Error (ME) Busy and Broadcast Command received (BCR) bits in its internal status register. The BIT Word at shared RAM address is reset to 0x0000. If either transmitter was shutdown, the shutdown condition is overridden. If the Terminal Flag (TF) status bit was inhibited, the inhibit is reset. This command does not reset any of the host-programmed registers that configure the terminal for operation. To complete the terminal reset process, the host must assert either RESET hardware master reset or assert the SRST bit in the “Remote Terminal Configuration Register (0x0017)” to execute software reset. See following section entitled Reset and Initialization for additional details. Because MC8 requires host interaction, most applications will probably utilize the IWA interrupt to alert the host when received.				
MC8 EXCEPTIONS:				
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. GAPERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
RESERVED MODE CODES MC9 - MC15: Mode code command with mode codes 01001 through 01111 and T/$\overline{R}$ bit equals 1	The reserved mode code commands do not have defined terminal actions. Host must initialize IP core to respond using either of the two following methods:			
The mode code's bit in Illegalization Table equals 0 *	RT is not using "illegal command detection." Respond "in form": Reset Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit reset. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
OR The mode code's bit in Illegalization Table equals 1 **	**OR** RT is using "illegal command detection"and mode code is illegalized. Set Message Error (ME) status and transmit Status Word.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. BUSID bit updated. MERR bit set. RTRT bit reset. (Other error bits reset.)	ILCMD IWA
MC9 - MC15 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Config. Reg. equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Config. Reg. equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Config. Reg. equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
TRANSMIT VECTOR WORD (MC16): **Mode code command with mode code 10000** **and $T/\overline{R}$ bit equals 1**	Default CS response: Reset Message Error (ME) and BCR status bits. then transmit Status Word followed by the data word stored in the assigned index or ping-pong data buffer (or in Descriptor Word 4 for SMCP Simplified Mode Command Processing).	DBAC bit reset. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
MC16 EXCEPTIONS:				
Broadcast address RT31 (broadcast not allowed)	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB toggles.	MERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
SYNCHRONIZE WITH DATA WORD (MC17): Mode code command with mode code 10001 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status. and transmit Status Word. If broadcast, set BCR status bit and suppress Status response.	DBAC bit reset. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
IP Core stores received data word in the assigned ping-pong or index data buffer (or in Descriptor Word 4 for SMCP Simplified Mode Command Processing). “Remote Terminal Configuration Register (0x0017)” MCOPT2 and MCOPT3 bits allow automatic Time-Tag count loading using the data word received. If MCOPT2 equals 1, the received data word is automatically loaded to the Time-Tag counter if the low order bit of the received data word (bit 0 equals 0. If MCOPT3 equals 1, the received data word is automatically loaded to the Time-Tag counter if the low order bit of the received data word (bit 0) equals 1. If both bits are set, the received data word is unconditionally loaded into the Time-Tag counter. For non-broadcast commands, counter load occurs before status word transmission.				
MC17 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Config. Reg. equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Config. Reg. equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Config. Reg. equals 0. The Illegalization Table bit equals 1 **	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word not followed by a contiguous data word (missing data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
Mode code command word followed by data word with Manchester encoding or parity error (bad data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. IWDERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
TRANSMIT LAST COMMAND (MC18): Mode code command with mode code 10010 and T/$\overline{R}$ bit equals 1	Default response: Status is not updated. Transmit Status Word from the previous command, with data word containing the last valid command word (assuming it was not a "Transmit Status" or a "Transmit Last Command" mode command.	DBAC bit reset. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
Transmitted data word is automatically provided from an internal register, and is copied to assigned index or ping-pong buffer (or to Descriptor Word 4 for SMCP Simplified Mode Command Processing)				
MC18 EXCEPTIONS:				
Broadcast address RT31 (broadcast not allowed)	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB toggles.	MERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
TRANSMIT BIT WORD (MC19): **Mode code command with mode code 10011** **and $T/\overline{R}$ bit equals 1**	Default response: Reset Message Error (ME) and BCR status bits. then transmit Status Word followed by data word from either BIT Word Register or Alternate BIT Word Register, depending on Configuration Reg. 2 option bit ALTBITW.	DBAC bit reset. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit reset. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
Transmitted data word is automatically copied to the assigned index or ping-pong buffer (or to Descriptor Word 4 for SMCP Simplified Mode Command Processing)				
MC19 EXCEPTIONS:				
Broadcast address RT31 (broadcast not allowed)	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB toggles.	MERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Invalid command word. **OR** $T/\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when $T/\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond "In form": Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using "illegal command detection" (in form response).

** Command is illegal and terminal is using "illegal command detection"

*** Undefined mode command rendered invalid by UMCINV option bit. Command's bit in Illegalization Table is "don't care".

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
SELECTED TRANSMITTER SHUTDOWN (MC20): Mode code command with mode code 10100 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status. and transmit Status Word. If broadcast, set BCR status bit and suppress Status response. This command is intended for use in 1553 systems with more than one dual redundant bus.	DBAC bit reset. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA

After Status Word transmission, the IP core stores received data word in the assigned index or ping-pong buffer (or in Descriptor Word 4 if SMCP Simplified Mode Command Processing applies).

If the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 0, the received data word is compared to the value in the Bus Select Register corresponding to the inactive bus. For example, if the command is received on Bus A, the comparison uses the Bus B Select Register value. If the compared values match, the IP core automatically shuts down either **transmit and receive** or **transmit only** for the inactive bus, depending on the state of the BSDTXO bit in the “Master Configuration Register 1 (0x0000)”. See descriptions for the BSDTXO bit in Master Configuration Register and the AUTOBSD bit in the RT Configuration Register for further information. When a bus transmitter (or transmitter and receiver) is shut down by this mode command, bus status is reflected by assertion of a TXASD or TXBSD bit in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. If BSDTXO equals logic 0, an RXASD or RXBSD bit will also be asserted.

If the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 1, the IWA interrupt is typically used to alert the host when an MC20 command is received. **The host** must evaluate whether the received mode data word matches the bus selection criteria. If bus selection match occurs, **the host** fulfills bus shutdown command using one of two options:

1. set the bus shutdown bit RTINHA or RTINHB for the inactive bus in the “Remote Terminal Configuration Register (0x0017)” **to inhibit both transmit and receive,**

OR

2. assert the transmit shutdown input signal TXINHA or TXINHB for the inactive **bus to inhibit only transmit**. The inactive bus receiver remains active; all valid commands are heeded without transmit. **This option is rarely applied.**

Once shutdown, the inactive bus transmitter (or transmitter and receiver) can be reactivated five ways: an “Override Transmitter Shutdown” MC5, a MC21 command with data word that matches “bus select” criteria, a “Reset Remote Terminal” MC8 mode code command, a software reset initiated by setting the RT$\overline{\text{RESET}}$ bit in the “Master Status and Reset Register (0x0001)”, or by hardware reset initiated by asserting the $\overline{\text{RESET}}$ Master Reset input signal.

MC20 EXCEPTIONS:				
Invalid command word. **OR** T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***	No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)	No Change	No Message Info Word is written	None

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
T/$\overline{\text{R}}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *	Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	IWA IBR
T/$\overline{\text{R}}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word not followed by a contiguous data word (missing data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. WCTERR bit updated. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Mode code command word followed by data word with Manchester encoding or parity error (bad data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. IWDERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

<table>
<tr><th>Circumstances for Received Message</th><th>Terminal Response to Received Command</th><th>Bits Updated in Descriptor Control Word</th><th>Bits Updated in Data Buffer Msg Info Word</th><th>Interrupt Options</th></tr>
<tr><td><b>OVERRIDE SELECTED TRANSMITTER SHUTDOWN (MC21): Mode code command with mode code 10101
and T/$\overline{R}$ bit equals 1</b></td><td>Default response: Reset Message Error (ME) status. and transmit Status Word. If broadcast, set the BCR status bit and suppress Status response.</td><td>DBAC bit reset.
BCAST bit reset.
DPB bit toggles.</td><td>Normal CS update:
BUSID bit updated.
MERR bit reset.
ILCMD bit reset.
RTRT bit reset.
(All error bits reset.)</td><td>IWA</td></tr>
<tr><td colspan="5">After Status Word transmission, the IP core stores the received data word in the assigned index or ping-pong buffer (or in Descriptor Word 4 if SMCP Simplified Mode Command Processing applies).

If the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 0, the received data word is compared to the value in the Bus Select Register corresponding to the inactive bus. For example, if the command is received on Bus A, the comparison uses the Bus B Select Register value. If the compared values match, the IP core automatically re-enables <b>transmit and receive</b> for the inactive bus, regardless of the state of the BSDTXO bit in the Master Configuration Register. The IP core affirms fully re-enabled bus status by resetting all four TXASD, TXBSD, RXASD and/or RXBSD bits in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. <b><i>Note:</i></b> <i>If the TXINHA or TXINHB input signals are asserted, the IP core cannot override the resulting hardware transmit inhibit for the affected bus. In this case, the corresponding TXASD and/or TXBSD bits remain high. See RT Built-In Test Register description for further information.</i>

If the AUTOBSD bit in the “Remote Terminal Configuration Register (0x0017)” equals 1, the IWA interrupt is typically used to alert the host when an MC21 command is received. <b>The host</b> must evaluate whether the received mode data word matches the bus selection criteria. If bus selection match occurs, <b>the host</b> fulfills the “override shutdown” command using one of two options:

1. reset the RTINHA or RTINHB bus shutdown bit corresponding to the inactive bus in the “Remote Terminal Configuration Register (0x0017)” to re-enable both transmit and receive, if the host used this bit to shut down transmit and receive for an earlier MC4 or MC20 command. <b><i>Note:</i></b> <i>Resetting the RTINHA or RTINHB shutdown bit cannot restore bus transmit capability if the TXINHA or TXINHB input signal is asserted,</i>
<b>OR</b>
2. reset the transmit shutdown input signal TXINHA or TXINHB for the inactive bus to re-enable transmit if the host used this signal to shut down transmit only for an earlier MC4 or MC20 command.</td></tr>
<tr><td><b>MC21 EXCEPTIONS:</b></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Invalid command word.
<b>OR</b>
T/$\overline{R}$ bit equals 0 and UMCINV bit in the RT Configuration Register equals 1 ***</td><td>No terminal response, the message is ignored. No Status Word change. (mode code is undefined when T/$\overline{R}$ bit equals 0)</td><td>No Change</td><td>No Message Info Word is written</td><td>None</td></tr>
<tr><td>T/$\overline{R}$ bit equals 0
<b>AND</b>
UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 0 *</td><td>Respond “In form”: Reset Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, set the BCR status bit and suppress status response.</td><td>DBAC bit set.
BCAST bit updated.
DPB bit toggles.</td><td>Normal CS update:
BUSID bit updated.
MERR bit reset.
ILCMD bit reset.
RTRT bit reset.
(Other error bits reset.)</td><td>IWA
IBR</td></tr>
</table>

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
$T/\overline{R}$ bit equals 0 **AND** UMCINV bit in the RT Configuration Register equals 0. The Illegalization Table bit equals 1 **	Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.	DBAC bit set. BCAST bit updated. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA IBR
Mode code command word not followed by a contiguous data word (missing data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR
Mode code command word followed by data word with Manchester encoding or parity error (bad data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. IWDERR bit set. BUSID bit updated. ILCMD bit reset. RTRT bit reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

*** Undefined mode command rendered invalid by UMCINV option bit. Command’s bit in Illegalization Table is “don’t care”.

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
RESERVED MODE CODES MC22 - MC31: Mode code commands having mode codes 10110 through 11111	The reserved mode code commands do not have defined actions. Host must initialize IP core to respond using either of the two following methods:			
The mode code's bit in Illegalization Table equals 1 ** (RT is using "illegal command detection")	Mode code is illegalized. Set Message Error (ME) status and transmit Status Word. If T/$\overline{R}$ bit equals 1, suppress data word transmission.	DBAC bit set. BCAST bit reset. DPB bit toggles.	ILCMD bit set. MERR bit set. BUSID bit updated. RTRT bit reset. (Other error bits reset.)	ILCMD IWA
OR	**OR**			
The mode code's bit in Illegalization Table equals 0 * (RT not using "illegal command detection," respond "in form")	**If T/$\overline{R}$ bit equals 1,** Reset Message Error (ME) status. Transmit Status Word with contiguous data word read from assigned index or ping-pong buffer (or from Descriptor Word 4 if the SMCP option applies.)	DBAC bit set. BCAST bit reset. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset. (All error bits reset.)	IWA
	If T/$\overline{R}$ bit equals 0, Reset Message Error (ME) status and transmit Status. If broadcast, also set BCR status and suppress Status transmit. IP Core stores received data word in assigned index or ping-pong buffer (or in Descriptor Word 4 if SMCP Simplified Mode Command Processing applies).	DBAC bit reset. BCAST bit updated. DPB bit toggles.	Normal CS update: BUSID bit updated. MERR bit reset. ILCMD bit reset. RTRT bit reset (All error bits reset.)	IWA IBR
MC22 - MC31 EXCEPTIONS:				
Invalid command word.	No terminal response, the message is ignored. No Status Word change.	No Change	No Message Info Word is written	None
T/$\overline{R}$ bit equals 0 and mode code command word is not followed by a contiguous data word (missing data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. WCTERR bit set. BUSID bit updated. ILCMD, RTRT bits reset. (Other error bits reset.)	MERR IWA IBR

Circumstances for Received Message	Terminal Response to Received Command	Bits Updated in Descriptor Control Word	Bits Updated in Data Buffer Msg Info Word	Interrupt Options
T/$\overline{R}$ bit equals 0 and command word is followed by data word with Manchester or parity error (bad data word)	No Status Word transmit. Set the Message Error (ME) status bit. If broadcast, set the BCR status bit.	DBAC bit set. BCAST bit updated. DPB bit toggles.	MERR bit set. IWDERR bit set. BUSID bit updated. ILCMD, RTRT bits reset. (Other error bits reset.)	MERR IWA IBR
T/$\overline{R}$ bit equals 1 and mode code command word is followed by a contiguous data word	No Status Word transmit. Set the Message Error (ME) status bit.	DBAC bit set. BCAST bit reset. DPB bit toggles.	MERR, WCTERR bits set. BUSID bit updated. ILCMD, RTRT bits reset. (Other error bits reset.)	MERR IWA
T/$\overline{R}$ bit equals 1 and mode code command is addressed to RT31	No Status Word transmit. Set the Message Error (ME) and BCR status bits.	DBAC bit set. BCAST bit set. DPB bit toggles.	MERR bit set. BUSID bit updated. ILCMD, RTRT bits reset. (Other error bits reset.)	MERR IWA IBR

* Command is illegal but terminal is not using “illegal command detection” (in form response).

** Command is illegal and terminal is using “illegal command detection”

24. MIL-STD-1553 BUS INTERFACE

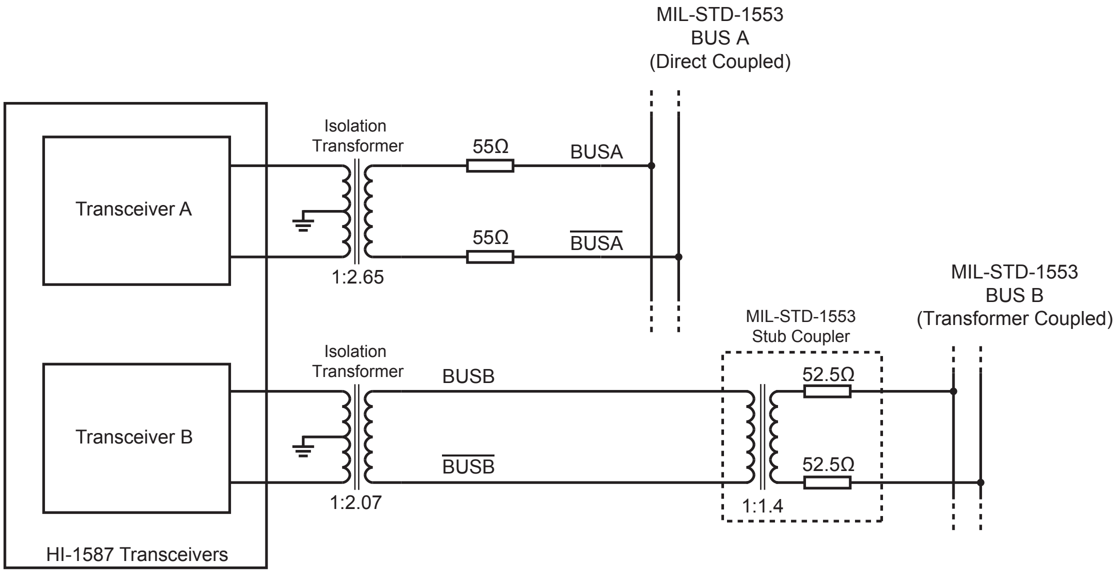


Figure 27. Bus Connection Example using HI-1587 IP Dongle Transceiver.

25. ORDERING INFORMATION

HI - 6300 - xxx x

PART NUMBER	IP CORE FUNCTIONALITY
B	Remote Terminal and Monitor Terminal (RT-MT)
C	Bus Controller and Remote Terminal (BC-RT)
D	Bus Controller, Remote Terminal and Monitor Terminal (BC-RT-MT)
F	RT-MT, DO-254 Design Assurance Level A Compliant
G	BC-RT, DO-254 Design Assurance Level A Compliant
H	BC-RT-MT, DO-254 Design Assurance Level A Compliant

CUSTOMER ID #
Unique 3-digit customer project code, e.g. 001, 002, 003, etc.

26. REVISION HISTORY

Revision	Date	Description of Change
DS6300, Rev. New	04/18/18	Initial Release
Rev. A	05/18/18	Remove registers 0x0050 and 0x0051 from “Table 9. Register Summary”. They are RAM locations, not registers. Clarify in “Table 9. Register Summary” that reset value of RAM location 0x004F (BC Last Message Block Address) is undefined. Add footnote. Clarify format of Interrupt Information Word (IIW) and Interrupt Address Word (IAW) in “Interrupt Log Buffer”. See Table 10. Clarify function of AUTOBSD, bit 4 in “Remote Terminal Configuration Register (0x0017)”. Bit 1, MC16OPT, in “Remote Terminal Configuration Register (0x0017)” is erroneously called MCOPT1 in describing SSYSF bit 2 in “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)”. Fix minor typographical errors throughout the document.
Rev. B	04/10/19	Correct typo in “Figure 14. Deriving the Illegalization Table Address From the Received Command Word”. Correct description for RD_DATA[15:0] in “Table 1. IP Core Host Interface Signals”.
Rev. C	02/07/20	Correct errors in Hard Reset Default values in “Table 9. Register Summary”. Bits [13,12] in “Master Configuration Register 2 (0x004E)” are read-only. Bits [4,3] in “Hardware Interrupt Output Enable Register (0x0013)” are read-only. Clarify how to clear bits [12,11], LBFA and LBFB respectively in “Hardware Pending Interrupt Register (0x0006)”. Correct text typo in “BC Block Status Word”, bit 7, MSTATSET. Clarify RW status of bits in “Bus Controller (BC) General Purpose Queue Pointer Register (0x0038)”. Bit [11] in “SMT Configuration Register (0x0029)” is read-only. Update “Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)” to reflect correct function of MC16OPT bit (should refer to resetting SVCREQ status bit and not SSYSF bit). Correct typographical errors in “Figure 19. Illustration of Ping-Pong Buffer Mode”. Add clarification for digital loopback testing in “22.1.6. Host-Directed RT-Mode Loopback Testing (On-Line Analog or Off-Line Digital)”. Remove parts HI-6300-xxx A and HI-6300-xxx E from “Ordering Information”.