HOLT INC.
INTEGRATED CIRCUITS

MAMBA™: HI-6136

3.3V RT / MT MIL-STD-1553 / MIL-STD-1760

Compact Multi-Terminal Device with SPI Host Interface

October, 2015

GENERAL DESCPIPTION

The 3.3V CMOS HI-6136 device is a member of Holt's MIL-STD-1553 MAMBA™ family and provides a complete single- or multi-function interface between a host processor and MIL-STD-1553B bus. Each IC contains a Remote Terminal (RT) and a Bus Monitor Terminal (MT). Any combination of the contained 1553 functions can be enabled for concurrent operation. The enabled terminals communicate with the MIL-STD-1553 buses through a shared on-chip dual bus transceiver and external transformers. The user allocates 16K bytes of on-chip static RAM between devices to suit application requirements.

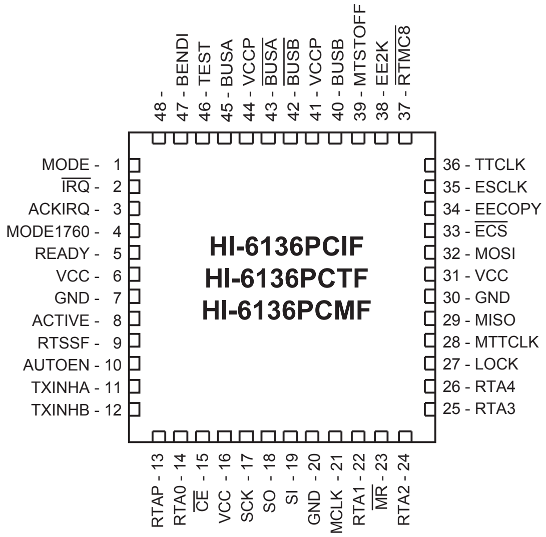
The HI-6136 communicates with the host via a 40 MHz 4-wire serial peripheral interface (SPI). Programmable interrupts provide terminal status to the host processor. Circular data buffers in RAM have interrupts for rollover and programmable “level attained”.

The HI-6136 can be configured for automatic self-initialization after reset. A dedicated SPI port reads data from an external serial EEPROM to fully configure registers and RAM and optionally start execution for any of the terminal devices.

FEATURES

- Concurrent multi-terminal operation for RT and/or MT MIL-STD-1553B functions.
- 8K x 17-bit words internal static RAM with parity
- Autonomous terminal operation requires minimal host intervention.
- 40 MHz SPI Host Interface.
- MIL-STD-1760 option sets Busy bit in Status Word response during initialization.
- World's smallest MIL-STD-1553 terminal, QFN package measures just 6mm x 6mm.
- Simple Monitor Terminal (SMT) Mode records commands and data separately, with 16-bit or 48-bit time tagging.
- Independent 16-bit time tag counters and clock sources for RT and MT. The MT also has a 48-bit time count option.
- 64-Word Interrupt Log Buffer queues the most recent 32 interrupts. Hardware-assisted interrupt decoding quickly identifies interrupt sources.
- Built-in self-test for protocol logic, digital signal paths and internal RAM.
- Optional self-initialization at reset uses external serial EEPROM.
- ±8kV ESD Protection (HBM, all pins).
- Two temperature ranges: -40°C to +85°C, or -55°C to +125°C with optional burn-in.
- RoHS compliant and Tin / Lead options available.

PIN CONFIGURATION (TOP)



48 - Pin Plastic 6mm x 6mm Chip-Scale Package (QFN)

See Section 24.1 on page 202 for 48-Pin PQFP Configuration

NOTES:

Table of Contents

List of Figures

List of Tables

1. BLOCK DIAGRAM

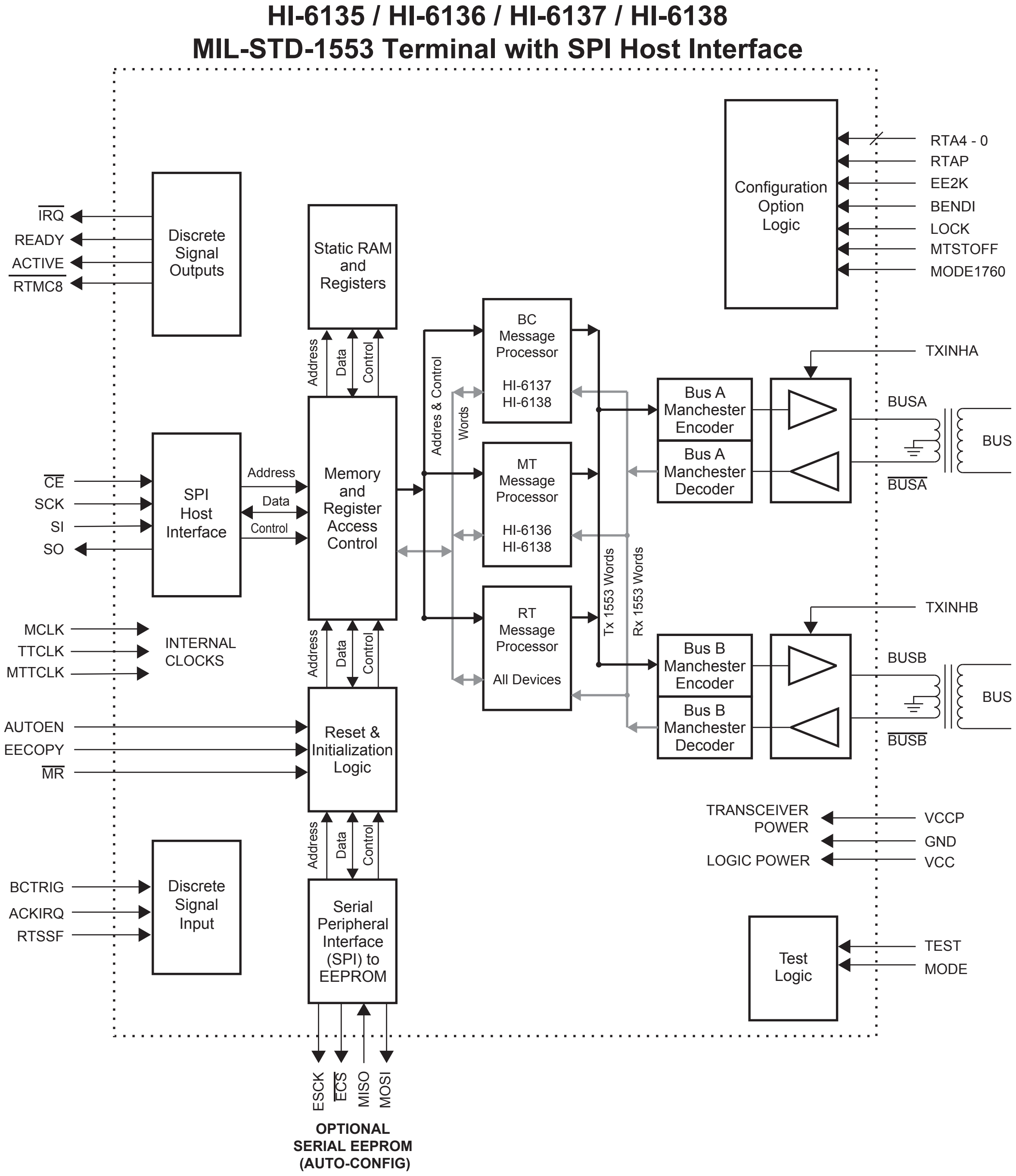


Figure 1. Block Diagram

2. FEATURE OVERVIEW

2.1. Remote Terminal Operation

The HI-6136 is configurable to operate one a Remote Terminal. The RT is software compatible with Holt's popular HI-6130/31 Remote Terminal.

2.2. Monitor Terminal Operation

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

2.3. Interrupts

Host interrupts can originate from device hardware or any of the enabled terminal devices (up to 3 devices). 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 device: MT, RT or hardware. 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 MT or RT, the three low-order bits in the same register identify the specific interrupt register (or registers) with pending interrupts: that is, the MT or RT Pending Interrupt registers.

2.4. Reset and Initialization

After hardware Master Reset, there are two initialization methods: host initialization or self-initialization from external serial EEPROM. For host initialization, the host processor uses the 4-wire SPI to load registers and initialize tables, data buffers, etc. in internal static RAM. For self-initialization, the device uses setup information contained within an external serial EEPROM. A dedicated 4-wire SPI port reads data from the serial EEPROM and writes it to registers and RAM. Error checking is performed, looking for data mismatch or an EEPROM checksum error.

Individual 1553 terminal devices (MT or RT) can be re-initialized from the serial EEPROM. Refer to “Reset and Initialization” on page 147.

3. PIN DESCRIPTIONS

Table 1. Pin Descriptions

Pin	Function	Description
MCLK	Input 50kΩ pull-down	Master clock input, 50.0MHz +/-100 ppm.
TTCLK MTTCLK	Inputs 50kΩ pull-down	Optional clock input for RT time tag counter. Optional clock input for the MT time tag counter. Each function (MT or RT) has an independent time tag counter. The RT counter clock is selectable from internally generated frequencies, or an external clock input. The MT time tag counter has its own external or internal clock source.
$\overline{\text{MR}}$	Input 50kΩ pull-up	Master reset, active low. The host can also assert software reset by setting bits in the “Master Status and Reset Register (0x0001)”.
TXINHA TXINHB	Inputs 50kΩ pull-up	Transmit inhibit inputs for Bus A and Bus B, active high. These two inputs are logically ORed with the pair of corresponding bits in the “Master Configuration Register 1 (0x0000)” to enable or inhibit transmit on Bus A or Bus B, affecting behavior for all enabled 1553 devices.
MTSTOFF	Input 50kΩ pull-down	Memory test disable, active high. When this pin is low, the device performs a memory test on the entire RAM after rising edge on the $\overline{\text{MR}}$ reset pin. When this pin is high, RAM testing is skipped, resulting in a faster reset process. For further information, refer to “Reset and Initialization” on page 147.
AUTOEN	Input 50kΩ pull-down	Auto-Initialize Enable, active high. If this pin is high at rising edge on $\overline{\text{MR}}$ reset pin, self-initialization proceeds, copying configuration data to registers and RAM from an external serial EEPROM via a dedicated EEPROM SPI port. Refer to “Reset and Initialization” on page 147.
EE2K	Input 50kΩ pull-down	When the AUTOEN pin is high, the EE2K input sets the range of the auto-initialization process. When EE2K is low, registers and RAM occupying the 8K address range from 0x0 to 0x1FFF are initialized. For applications needing faster initialization, when EE2K is high, only registers and RAM occupying the 2K address range from 0x0 to 0x07FF are initialized. If the AUTOEN pin is low, this pin is not used.
READY	Output	This pin is low when auto-initialization or built-in test is in process. The host cannot read or write device RAM or registers when pin state is low; reads to any address return the value in the “Master Status and Reset Register (0x0001)”. When the AUTOEN pin is low at Master Reset, the host can configure device RAM and registers after READY goes high.
ACTIVE	Output	This pin is high while an enabled RT in the device is processing a 1553 message. The IMTA bit 1 in “Master Configuration Register 1 (0x0000)” logically-ORs bus monitor (MT) activity as well.
$\overline{\text{ECS}}$ ESCK MOSI MISO	Output Output Output Input 50kΩ pull-down	Dedicated 4-wire Serial Peripheral Interface (SPI) for connection to an optional external EEPROM used for automatic self-initialization when AUTOEN is high at Master Reset.

Pin	Function	Description
EECOPY	Input 50kΩ pull-down	EEPROM Copy, active high. Asserting this input initiates RAM and register copy into serial EEPROM used for auto-initialization. Refer to “Reset and Initialization” on page 147.
MODE1760	Input 50kΩ pull-down	Mode 1760 enable, active high. Assert this pin during a hardware reset to enable 1760 mode. During 1760 mode, the device 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.
LOCK	Input 50kΩ pull-down	Pin states are latched to the Lock bit in the RT Operational Status Register (see page 75) when rising edge occurs on the $\overline{\text{MR}}$ pin. If status register Lock bit is high, the host cannot overwrite the terminal address in the same register. If status register Lock bit is low, the host can overwrite the terminal address and parity (and the Lock bit) in the RT Operational Status register.
$\overline{\text{RTMC8}}$	Input Open-Drain	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” mode code command. The minimum output pulse width is 100ns, unaffected by $\overline{\text{MR}}$ assertion.
RTSSF	Input 50kΩ pull-down	RT Subsystem Fail input, active high. When this input is high, the RT sets the Subsystem Fail flag in its transmit status word. This input is logically-ORed with the SSYSF bit in the RTs 1553 Status Word Bits Register.
$\overline{\text{IRQ}}$	Output Open-Drain	Interrupt request, active low. This pin is asserted each time an enabled interrupt event occurs. This signal is programmed as a brief low-going pulse output or as a level output by the INTSEL bit in the “Master Configuration Register 1 (0x0000)”. If level output is selected, $\overline{\text{IRQ}}$ stays low until the host acknowledges $\overline{\text{IRQ}}$ by pulsing a rising edge at the ACKIRQ pin.
ACKIRQ	Input 50kΩ pull-down	Interrupt Acknowledge, active high. This input is only used when the INTSEL bit in the RT Configuration Register is high, enabling level interrupt assertion for the $\overline{\text{IRQ}}$ pin. When interrupt assertion causes the $\overline{\text{IRQ}}$ pin to go low, a high-going pulse on ACKIRQ (250ns minimum duration) clears the $\overline{\text{IRQ}}$ output to logic 1.
RTA4:0 RTAP	Inputs 50kΩ pull-ups	Remote terminal address bits 4 - 0, and parity bit for the RT. The RTAP pin provides odd parity for the address on pins RTA4:0. The terminal address and parity pin levels are latched into the RT Operational Status Register (see page 75) when rising edge occurs on the $\overline{\text{MR}}$ pin. The RT Operational Status Register value (not these pins) reflects the active terminal address. The host can overwrite the RT Operational Status register address value only when the register Lock bit is reset.
BENDI	Input 50kΩ pull-down	Big Endian configuration pin for selecting “endianness” or byte order, when using byte transfers. Endianness is the system attribute that indicates whether integers are represented with the most significant byte stored at the lowest address (big endian) or at the highest address (little endian). Internal register / RAM storage is “big endian.” This pin controls the byte order of transferred 16-bit data following the SPI command. When BENDI is low, “little endian” is chosen; the low order byte (bits 7:0) is transacted on the SPI before the high order byte (bits 15:8). When BENDI is high, “big endian” is chosen and the high order byte is transacted on the SPI before the low order byte.

Pin	Function	Description
TEST	Input 50kΩ pull-down	Test enable input. The host asserts this pin to perform RAM self-test and loop-back tests.
TESTA	Input 50kΩ pull-down	Pin used for factory test. Do not connect.
MODE	Input 50kΩ pull-up	Pin used for factory test. Do not connect.
VCC VCCP GND	Power Supply	3.3VDC power supply for logic and bus transceiver.
BUSA $\overline{\text{BUSA}}$	Analog	Bi-directional Bus A interface to external MIL-STD-1553 isolation transformer. Observe positive / negative polarity.
BUSB $\overline{\text{BUSB}}$	Analog	Bi-directional Bus B interface to external MIL-STD-1553 isolation transformer. Observe positive / negative polarity.

Table 2. Host Interface Pins

Pin	Function	Description
$\overline{\text{CE}}$	Input 50kΩ pull-up	Chip Enable, active low. When asserted, this pin enables host read or write accesses to device RAM or registers via host SPI port. The SPI port operates in Slave mode. This pin is connected to the Slave Select output on the host SPI port.
SO	Output	Serial Peripheral Interface (SPI) Serial Output pin. This pin is connected to MISO (Master In - Slave Out) pin on host SPI port. The SO pin is tri-stated when not transmitting serial data to the host.
SI	Input 50kΩ pull-down	Serial Peripheral Interface (SPI) Serial Input pin. This pin is connected to MOSI (Master Out - Slave In) pin on host SPI port.
SCK	Input 50kΩ pull-down	Serial Peripheral Interface (SPI) Serial Clock pin. This pin is connected to SCK output pin on host SPI port.

4. MEMORY MAP

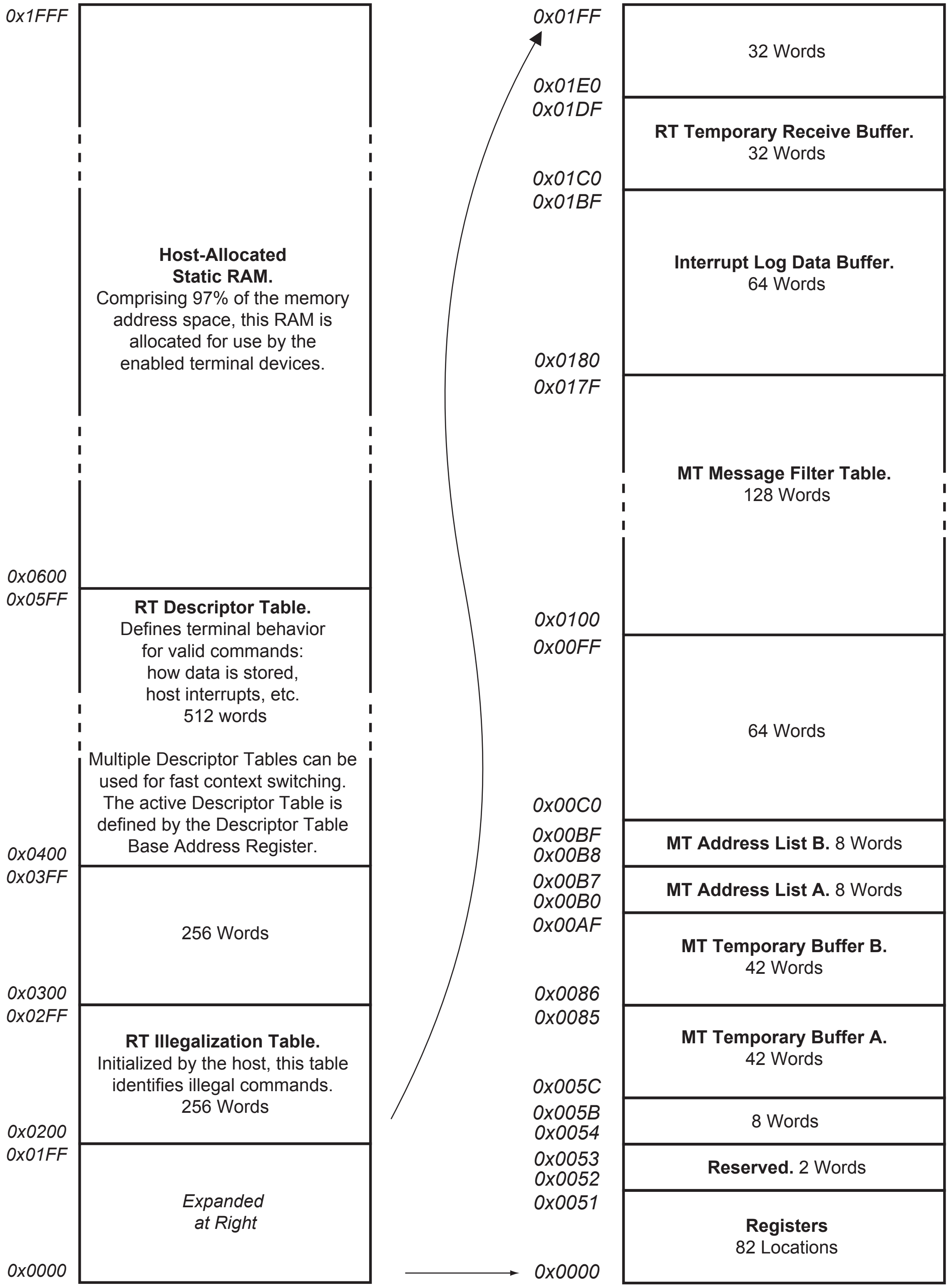


Figure 2. Address Mapping for Registers and RAM

5. RAM STRUCTURES

Figure 2 shows a map for memory and registers. Application requirements dictate the specific RAM structures needed. These structures listed here are explained later.

5.1. Interrupt Log Data Buffer

The device 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 device updates the register to point to the log address used for the next occurring interrupt.

5.2. 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).

5.3. 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 device-maintained data buffer pointer(s).

5.4. 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.

5.5. 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.

5.6. 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.

5.7. 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.

5.8. 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.

5.9. 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.

5.10. 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.

6. HARDWARE FEATURES

6.1. Remote Terminal Address Inputs

The 5-bit Remote Terminal address for the RT is set using pins RTA4:0. The RTAP input pin should be set or reset to present matching odd parity. The state of the RT address and parity pins is latched into the RT Operational Status register (see page 75) upon rising edge on the $\overline{\text{MR}}$ master reset input. The state of the LOCK input is latched into the RT Operational Status register at the same time, and controls whether or not the terminal address and parity in the RT Operational Status register can be overwritten by host writes into the register. When the LOCK input pin is high, auto-initialization cannot overwrite the RT address value latched from the input pins. Between Master Reset assertions, the state of the RTA and RTAP input pins is “don’t care”. If the value of RT address and parity in the RT Operational Status register has parity error, terminal operation is disabled.

6.2. Dual Transceivers for MIL-STD-1553 Bus

Built-in bus transceivers provide direct interface between the device and MIL-STD-1553 bus isolation transformers. The transceivers convert digital data to and from differential Manchester II encoded bus signals. A pair of “transmit inhibit” input pins exercises direct control over transmission for both buses.

6.3. Encoder and Decoders

The device 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.

6.4. Auto-Initialization Serial EEPROM Interface

The device has an automatic self-initialization feature. If self-initialization is enabled after $\overline{\text{MR}}$ master reset, the device reads configuration settings from an external serial EEPROM during initialization to initialize registers and RAM structures. A method is provided to initially program or later overwrite the external serial EEPROM by copying host-loaded tables and register values to the serial EEPROM.

6.5. 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.

6.6. MIL-STD-1760 Mode

A dedicated hardware pin is available to activate MIL-STD-1760 Mode. When operating in MIL-STD-1760 mode, the device 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.

7. REGISTER & MEMORY ADDRESSING

The HI-6136 has an internal address space of 8,192 (8K) 17-bit words. 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 2 shows address mapping for registers and RAM. All registers and some RAM structures have fixed addresses. Other RAM structures shown are relocatable; Each relocatable structure has a base address register. Figure 2 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 RT RAM for monitor needs. Device RAM and register address mapping is word oriented, rather than byte oriented.

8. 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 3 lists all device registers.

Table 3. Register Summary

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
0	0x0000	All	"Master Configuration Register 1 (0x0000)" on page 26	0x0000
1	0x0001	All	"Master Status and Reset Register (0x0001)" on page 31	0x0000
2	0x0002	RT	**When TEST pin is logic 0**, this address is "Remote Terminal Current Command Register (0x0002)" on page 77 **When TEST pin is logic 1**, this address is "Loopback Test Receive Data Register (0x0002)" on page 158	0x0000
3	0x0003	RT	"Remote Terminal Current Control Word Address Register (0x0003)" on page 77	0x0000
4	0x0004	----	Reserved	0x0000
5	0x0005	----	Reserved	0x0000
6	0x0006	All	"Hardware Pending Interrupt Register (0x0006)" on page 38	0x0000
7	0x0007	----	Reserved	0x0000
8	0x0008	MT	"SMT Bus Monitor Pending Interrupt Register (0x0008)" on page 68	0x0000
9	0x0009	RT	"Remote Terminal (RT) Pending Interrupt Register (0x0009)" on page 85	0x0000
10	0x000A	All	"Interrupt Count & Log Address Register (0x000A)" on page 35	0x0180
11	0x000B	All	"Memory Address Pointer Registers" on page 48	0x0000
12	0x000C	All	"Memory Address Pointer Registers" on page 48	0x0000
13	0x000D	All	"Memory Address Pointer Registers" on page 48	0x0000
14	0x000E	All	"Memory Address Pointer Registers" on page 48	0x0000
15	0x000F	All	"Hardware Interrupt Enable Register (0x000F)" on page 38	0x6000
16	0x0010	----	Reserved	0x6000
17	0x0011	MT	"SMT Bus Monitor Interrupt Enable Register (0x0011)" on page 68	0x6000
18	0x0012	RT	"Remote Terminal (RT) Interrupt Enable Register (0x0012)" on page 85	0x6000
19	0x0013	All	"Hardware Interrupt Output Enable Register (0x0013)" on page 38	0x6000
20	0x0014	----	Reserved	0x0000
21	0x0015	MT	"SMT Bus Monitor Interrupt Output Enable Register (0x0015)" on page 68	0x0000
22	0x0016	RT	"Remote Terminal (RT) Interrupt Output Enable Register (0x0016)" on page 85	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
23	0x0017	RT	"Remote Terminal Configuration Register (0x0017)" on page 71	0x0000
24	0x0018	RT	"Remote Terminal Operational Status Register (0x0018)" on page 75	0x0000
25	0x0019	RT	"Remote Terminal Descriptor Table Base Address Register (0x0019)" on page 77	0x0400
26	0x001A	RT	"Remote Terminal MIL-STD-1553 Status Word Bits Register (0x001A)" on page 78	0x0000
27	0x001B	RT	**When TEST pin is logic 0**, this address is "Remote Terminal Current Message Information Word Register (0x001B)" on page 79. **When TEST pin is logic 1**, this address is "RAM Self-Test Fail Address Register (0x001B)" on page 158.	0x0000
28	0x001C	RT	"Remote Terminal Bus A Select Register (0x001C)" on page 80	0x0000
29	0x001D	RT	"Remote Terminal Bus B Select Register (0x001D)" on page 80	0x0000
30	0x001E	RT	"Remote Terminal Built-In Test (BIT) Word Register (0x001E)" on page 81	0x0000
31	0x001F	RT	**When TEST pin is logic 0**, this address is "Remote Terminal Alternate Built-In Test (BIT) Word Register (0x001F)" on page 82. **When TEST pin is logic 1**, this address is "Loopback Test Transmit Data Register (0x001F)" on page 158.	0x0000
32	0x0020	----	Reserved	0x0000
33	0x0021	----	Reserved	0x0000
34	0x0022	----	Reserved	0x0600
35	0x0023	----	Reserved	0x0000
36	0x0024	----	**When the AUTOEN input pin is logic 1** at rising edge of $\overline{\text{MR}}$ Master Reset and RAM or register initialization failure (RAMIF) occurs, this register is the "Memory Test Fail Address Register (0x0024)" on page 150. This register holds the first encountered RAM / register address with data mismatch. There may be others. Once execution starts (or when not using auto-initialization), this register has no function.	0x0000
37	0x0025	----	Reserved	0x0000
38	0x0026	----	Reserved	0x0000
39	0x0027	----	Reserved	0x0000
40	0x0028	----	**When TEST pin is logic 0**, this register has no function. **When TEST pin is logic 1**, this address is "Self-Test Control Register (0x0028)" on page 154	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
41	0x0029	MT	“SMT Configuration Register (0x0029)” on page 60	0x0800
42	0x002A	----	Reserved	0x0000
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 63	0x0060
48	0x0030	MT	“SMT Next Message Command Buffer Address (0x0030)” on page 63	0x0000
49	0x0031	MT	“SMT Last Message Command Buffer Address (0x0031)” on page 64	0x0000
50	0x0032	----	Reserved	0x0000
51	0x0033	----	Reserved	0x0000
52	0x0034	----	Reserved	0x0000
53	0x0035	----	Reserved	0x0000
54	0x0036	----	Reserved	0x0000
55	0x0037	----	Reserved	0x0000
56	0x0038	----	Reserved	0x00C0
57	0x0039	All	“Time Tag Counter Configuration Register (0x0039)” on page 45	0x0000
58	0x003A	MT	“SMT Bus Monitor Time Tag Count Register (0x003A)” on page 64	0x0000
59	0x003B	MT	“SMT Bus Monitor Time Tag Count Mid Register (0x003B)” on page 64	0x0000
60	0x003C	MT	“SMT Bus Monitor Time Tag Count High Register (0x003C)” on page 64	0x0000
61	0x003D	MT	“SMT Bus Monitor Time Tag Utility Register (0x003D)” on page 65	0x0000
62	0x003E	MT	“SMT Bus Monitor Time Tag Utility Mid Register (0x003E)” on page 65	0x0000
63	0x003F	MT	“SMT Bus Monitor Time Tag Utility High Register (0x003F)” on page 65	0x0000
64	0x0040	MT	“SMT Bus Monitor Time Tag Match Register (0x0040)” on page 66	0x0000

Register Number	*Hex Address*	*Used By*	*Register Name*	*Hard Reset Default*
65	0x0041	MT	"SMT Bus Monitor Time Tag Match Mid Register (0x0041)" on page 66	0x0000
66	0x0042	MT	"SMT Bus Monitor Time Tag Match High Register (0x0042)" on page 66	0x0000
67	0x0043	----	Reserved	0x0000
68	0x0044	----	Reserved	0x0000
69	0x0045	----	Reserved	0x0000
70	0x0046	----	Reserved	0x0000
71	0x0047	----	Reserved	0x0000
72	0x0048	----	Reserved	0x0000
73	0x0049	RT	"Remote Terminal Time Tag Counter Register (0x0049)" on page 83	0x0000
74	0x004A	RT	"Remote Terminal Time Tag Utility Register (0x004A)" on page 83	0x0000
75	0x004B	----	Reserved	0x0000
76	0x004C	----	Reserved	0x0000
77	0x004D	RT	"Extended Configuration Register (0x004D)" on page 41	0x0000
78	0x004E	All	"Master Configuration Register 2 (0x004E)" on page 29	0x1100
79	0x004F	----	Reserved	0x0000
80	0x0050	----	Reserved	0x0000
81	0x0051	----	Checksum Fail Address & EEPROM Lock/Unlock	0x0000

9. REGISTERS USED BY ALL DEVICE FUNCTIONS

9.1. Master Configuration Register 1 (0x0000)

TXINHA	TXINHB	Reserved	Reserved	MAPSEL1	MAPSEL0	Reserved	MTENA	Reserved	RTENA	Reserved	RTSTEX	BSDTXO	INTSEL	IMTA	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin master reset, and is unaffected by assertion of the MTRESET, RTRESET bits in the “Master Status and Reset Register (0x0001)”.

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 pin. This register bit and the corresponding TXINHA pin globally affect all enabled 1553 devices (MT or 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 pin. This register bit and the corresponding TXINHB pin globally affect all enabled 1553 devices (MT or RT). This inhibit disables all transmission on Bus B.
13	Reserved	-	-	This bit is not used and reads logic 0.
12	Reserved	-	-	This bit is not used and reads logic 0.

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>11 − 10</td><td>MAPSEL1:0</td><td>R/W</td><td>0</td><td>
<p>MAP (Memory Address Pointer) Select.</p>
<p>The host SPI relies on a hardware memory address pointer for many SPI register or RAM accesses. This 2-bit field specifies which MAP is active for SPI transactions:</p>
<table>
<tr><th>Register Bit 11-10</th><th>Active Map</th><th>MAP Register Address</th><th>Enabling SPI Op Code</th></tr>
<tr><td>0-0</td><td>MAP1</td><td>0x000B</td><td>0xD8</td></tr>
<tr><td>0-1</td><td>MAP2</td><td>0x000C</td><td>0xD9</td></tr>
<tr><td>1-0</td><td>MAP3</td><td>0x000D</td><td>0xDA</td></tr>
<tr><td>1-1</td><td>MAP4</td><td>0x000E</td><td>0xDB</td></tr>
</table>
<p>The full 16-bit register can be directly written by the host using SPI op code 0x10, followed by 16-bit data word. An alternative method uses SPI op codes 0xD8 – 0xDA that write just the 2-bit MAPSEL field, without affecting other register data. These four SPI op codes only require transmission of an 8-bit instruction, without accompanying data.</p>
<p>Note: “Fast access” SPI op codes contain embedded register addresses and use a separate memory address pointer. This preserves values contained in MAP1 through MAP4. The “fast access MAP” cannot be read by the host but is written each time a “fast access” op code is processed. Fast Access op codes are provided for these SPI operations:</p>

SPI reads to register addresses 0 through 0x000F (decimal 15)
SPI writes to register addresses 0 through 0x003F (decimal 63)

</td></tr>
<tr><td>9</td><td>Reserved</td><td>-</td><td>-</td><td>This bit is not used and reads logic 0.</td></tr>
<tr><td>8</td><td>MTENA</td><td>R/W</td><td>0</td><td>
<p>Bus Monitor Enable.</p>
<p>If this bit equals logic 0, Bus Monitor operation is disabled.</p>
<p>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.</p>
</td></tr>
<tr><td>7</td><td>Reserved</td><td>-</td><td>-</td><td>This bit is not used and reads logic 0.</td></tr>
<tr><td>6</td><td>RTENA</td><td>R/W</td><td>0</td><td>
<p>Remote Terminal Enable.</p>
<p>If this bit equals logic 0, RT operation is disabled.</p>
<p>When this bit is logic 1, Remote Terminal is enabled, but operation is controlled by the state of the RTSTEX register bit.</p>
</td></tr>
<tr><td>5</td><td>Reserved</td><td>-</td><td>-</td><td>This bit is not used and reads logic 0.</td></tr>
<tr><td>4</td><td>RTSTEX</td><td>R/W</td><td>0</td><td>
<p>Remote Terminal Start Execution.</p>
<p>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.</p>
</td></tr>
</table>

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 MC21: • 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 fulfill 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 71), **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	INTSEL	R/W	0	IRQ Output Type Select. When this bit is 0, the $\overline{\text{IRQ}}$ (interrupt request) output generates a 1µs negative pulse when enabled interrupt events occur. When this bit is logic 1, the $\overline{\text{IRQ}}$ output consists of a continuous low level output requiring host action to negate $\overline{\text{IRQ}}$ to the high state. When level interrupts are enabled, the host negates $\overline{\text{IRQ}}$ by asserting the ACKIRQ input pin for at least 250ns.

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-chip terminal). When this bit equals 1, enabled Bus Monitor activity is logically-ORed with the activity of the other on-chip devices to determine ACTIVE status; the ACTIVE output is asserted during such Bus Monitor activity, whether or not the monitored message involves another on-chip terminal.
0	Reserved	-	-	This bit is not used and reads logic 0.

9.2. Master Configuration Register 2 (0x004E)

	TXTEST	TXTSTPND	DEVID1	DEVID0	REVID3	REVID2	REVID1	REVID0	MODE1760	Reserved	Reserved	LBFTF	WDTF	WDTXINH	Reserved	Reserved
$\overline{MR}$ Reset	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
Host Access	RW															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

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 pin. • Write logic “1” to TXTEST bit 15 (the test pin 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 − 12	DEVID[1:0]	R/W	01	Device ID These bits read 01 for HI-6136 following reset.
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 pin was high at $\overline{\text{MR}}$ rising edge, so the device is operating in 1760 mode (see "MIL-STD-1760: Busy Status Assertion After Power-Up"). The host may read this bit to confirm the device is operating in 1760 mode. Once in 1760 mode, the RT can/will immediately respond 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, autoinitialization from EEPROM 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 pin 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 pin 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.
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.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
1	Reserved	R/W	0	These bits are not used and read logic 0.
0	Reserved	R/W	0	These bits are not used and read logic 0.

9.3. Master Status and Reset Register (0x0001)

READY	Reserved	AUTOEN	MTRESET	Reserved	RTRESET	Reserved	Reserved	MTMIP	MTPIP	Reserved	RTMIP	Reserved	RTINH	EECKE	RAMIF	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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 pin reflects the state of this register bit. READY is low when auto-initialization, a soft reset caused by bit 12~10 assertion, or built-in test is underway. Host access to device 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	Reserved	R	0	This bit is not used.
13	AUTOEN	R	0	Auto-Initialization Enable Status. This bit reflects the state of the AUTOEN input pin that applied at the rising edge on the $\overline{\text{MR}}$ Master Reset input pin. If the register bit is high, auto-initialization was performed following $\overline{\text{MR}}$ reset.
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 pin 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 pin goes high and host read/write access is restored.
11	Reserved	R	0	This bit is not used.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
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 147. This bit remains high until reset is complete. While this bit remains high, the READY output pin 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 pin goes high and host read/write access is restored.
9	Reserved	R	0	This bit is not used.
8	Reserved	R	0	This bit is not used.
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	MTPIP	R	0	Bus Monitor Packet in Process. This bit is set at start of a data packet and is reset when the packet is deemed complete.
5	Reserved	R	0	This bit is 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.
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 81). Shut-down can be ended by “bus shutdown override” mode code command, $\overline{\text{MR}}$ reset or setting the RTRESET bit in this register.
1	EECKE	R	0	EEPROM Checksum Error. This function only applies when the AUTOEN input pin is logic 1 at rising edge of $\overline{\text{MR}}$ Master Reset. This enables auto-initialization from serial EEPROM, as well as RT or MT soft reset with auto-initialization. The EECKE bit is set (as well as bit 14 in the “Hardware Pending Interrupt Register (0x0006)” on page 38) when a serial EEPROM checksum failure occurs. Such failure may occur during full auto-initialization after $\overline{\text{MR}}$ master reset, or during execution of a partial, terminal-specific reset after assertion of the RTRESET or MTRESET bits in this register.

Bit No.	Mnemonic	R/W	Reset	Function
0	RAMIF	R	0	RAM Initialization Fail Interrupt. This function only applies when the AUTOEN input pin is logic 1 at rising edge of $\overline{\text{MR}}$ Master Reset. This enables auto-initialization from serial EEPROM, as well as terminal-specific partial auto-initialization during RT or MT soft reset. The RAMIF bit is set (as well as bit 13 in the “Hardware Pending Interrupt Register (0x0006)” on page 38) when one or more initialized RAM locations do not match their two corresponding serial EEPROM byte locations. Such failure may occur during full auto-initialization after $\overline{\text{MR}}$ master reset, or during execution of a partial, terminal-specific reset after assertion of the RTRESET or MTRESET bits in this register.

9.4. Overview of Interrupts

For interrupt management, the host accesses up to seven 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 three 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 Monitor and Remote Terminal interrupts.

Each interrupt register triplet for 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 $\overline{\text{IRQ}}$ 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-6136; 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.

9.5. Hardware Interrupt Behavior

Behavior described here for Hardware Interrupts closely resembles the behavior for the 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 $\overline{\text{IRQ}}$ 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 $\overline{\text{IRQ}}$ signal output, while low priority interrupts do not. The host detects low priority interrupts by polling the “Hardware Pending Interrupt Register (0x0006)”.

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

When the host receives an $\overline{\text{IRQ}}$ signal from the device, 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 “9.6. Interrupt Count & Log Address Register (0x000A)” and “9.7. 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 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 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 RT and MT Pending Interrupt Registers.
- 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 $\overline{\text{IRQ}}$ 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 $\overline{\text{IRQ}}$ 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)”.

9.6. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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 $\overline{\text{MR}}$ master reset fully enabled (see 9.8.1). 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 $\overline{\text{MR}}$ master reset, the HI-6136 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.

9.7. Interrupt Log Buffer

Shown in Figure 3, the Interrupt Log Buffer is a circular 64-word buffer in RAM, residing at address range 0x0180 to 0x01BF. Device 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 35).

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 using a word format identical 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 pin 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. For 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 Command Word for the message in which the interrupt occurred:

Interrupt Type	Interrupt Identification Word (IIW)	Interrupt Address Word (IAW)
Hardware	Matches format of Hardware Pending Interrupt Register 0x0006 on page 38	Always 0x0000
Simple Bus Monitor (SMT)	Matches the format of the SMT Pending Interrupt Register 0x0008 on page 68	An SMT Command Buffer address, points to the Block Status Word of the message in which interrupt occurred
Remote Terminal RT	Matches format of “Remote Terminal (RT) Pending Interrupt Register (0x0009)” on page 85, **EXCEPT** 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 41. **Note:** Busy status is not an interrupt causing event.	RT Descriptor Table address pointing to the Command Word of the message in which interrupt occurred

For a given terminal (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 $\overline{\text{IRQ}}$ output to the host.

When operating with two enabled terminal devices (SMT and RT), simultaneous interrupts can occur in the same message for multiple terminals. Each terminal device 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 $\overline{\text{IRQ}}$ output to the host.

In later data sheet sections, definitions are provided for interrupt register triplets used by the 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, device 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 ($\overline{MR}$) or software reset

Figure 3. Fixed Address Mapping for Interrupt Log Buffer

9.8. Hardware Interrupt Registers

9.8.1. Hardware Interrupt Enable Register (0x000F)

HSPIINT	EECKE / RAMPF	RAMIF / WDTX	LBFA	LBFB	MTTTRO	Reserved	Reserved	RTTTM	MTTTM	Reserved	Reserved	RTAPF	Reserved	Reserved	Reserved	
0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	$\overline{MR}$ 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.*

9.8.2. Hardware Pending Interrupt Register (0x0006)

HSPIINT	EECKE / RAMPF	RAMIF / WDTX	LBFA	LBFB	MTTTRO	Reserved	Reserved	RTTTM	MTTTM	Reserved	Reserved	RTAPF	RTIP	MTIP	Reserved	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{MR}$ Reset
R													R *			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

** Bits 2 and 1 are set for pending interrupts from RT and MT respectively*

9.8.3. Hardware Interrupt Output Enable Register (0x0013)

HSPIINT	EECKE / RAMPF	RAMIF / WDTX	LBFA	LBFB	MTTTRO	Reserved	Reserved	RTTTM	MTTTM	Reserved	Reserved	RTAPF	Reserved	Reserved	Reserved	
0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	$\overline{MR}$ Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

** Bits 2 - 0 are read-only and cannot be set by host.*

These three registers govern hardware interrupt behavior. As explained on the preceding page, bits 2 and 1 in the Hardware Pending Interrupt Register are set whenever interrupt bits are set in the other two pending interrupt registers (RT and MT). 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	HSPIINT	R/W	0	Host SPI Interrupt. An unexpected number of SCK clock pulses occured during a SPI transaction.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
14	EECKE	R/W	1	EEPROM Checksum Error Interrupt. This function only applies when the AUTOEN input pin is logic 1 at rising edge of $\overline{\text{MR}}$ Master Reset. This enables auto-initialization from serial EEPROM, as well as RT soft reset with auto-initialization. This bit is logic 1 in the “Hardware Interrupt Enable Register (0x000F)” and in the “Hardware Interrupt Output Enable Register (0x0013)” after $\overline{\text{MR}}$ master reset. The EECKE bit is set in the “Hardware Pending Interrupt Register (0x0006)” (as well as bit 1 in the “Master Status and Reset Register (0x0001)” when a serial EEPROM checksum failure occurs during auto initialization, or execution of a partial reset caused by assertion of the RTRESET bit in the “Master Status and Reset Register (0x0001)”.
	RAMPF			RAM Parity Fail Interrupt. This bit will be set to logic “1” when the device fails RAM parity check. RAM data errors are uncorrectable. When a RAM parity error is detected, register bit 13 is set to logic 1. The RAMIF bit 1 in register “Master Status and Reset Register (0x0001)” is also cleared. When bit 13 is asserted in the “Hardware Pending Interrupt Register (0x0006)”, the host can distinguish a RAMIF interrupt from a RAMPF interrupt by reading the “Master Status and Reset Register (0x0001)”. RAMIF bit 1 in that register will be logic 1 for RAMIF interrupt, logic 0 for RAMPF interrupt.
13	RAMIF	R/W	1	RAM Initialization Fail Interrupt. This function only applies when the AUTOEN input pin is logic 1 at rising edge of $\overline{\text{MR}}$ Master Reset. This enables auto-initialization from serial EEPROM, as well as RT soft reset with auto-initialization. This bit is logic 1 in the “Hardware Interrupt Enable Register (0x000F)” and in the “Hardware Interrupt Output Enable Register (0x0013)” after $\overline{\text{MR}}$ master reset. The RAMIF bit is set in the “Hardware Pending Interrupt Register (0x0006)” (as well as bit 0 in the “Master Status and Reset Register (0x0001)”) when one or more initialized RAM locations do not match their two corresponding serial EEPROM byte locations. Such failure occurs during auto initialization, or execution of a partial reset caused by assertion of the RTRESET bit in the “Master Status and Reset Register (0x0001)”.
	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 $\overline{\text{IRQ}}$ pin.

Bit No.	Mnemonic	R/W	Reset	Function
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 device 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.
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	Reserved	R	0	This bit is not used.
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	Reserved	R	0	This bit is not used.
4	Reserved	R	0	This bit is not used.
3	RTAPF	R/W	1	RT Terminal Address Parity Fail Interrupt. The Remote Terminal address and parity bits (latched into the RT Operational Status Register (see page 75) at rising edge of $\overline{\text{MR}}$) do not exhibit odd parity (do not have an odd number of bits having logic 1 state). Note: RT address parity is only checked if the RTENA pin is logic 1 at rising edge of $\overline{\text{MR}}$.
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).

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
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	Reserved	R	0	This bit is not used.

9.9. Extended Configuration Register (0x004D)

												Reserved	CIRBCST	INTBUSY	DPBTOFF	IRQOFF
$\overline{MR}$ Reset	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0
Host Access	R/W															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

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

Bit No.	*Mnemonic*	*Function*
15 − 5	-----	Not Used.
4	Reserved	Do not use. Setting this bit may cause unpredictable behavior.
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 112 and Receive and Transmit “Mode Command Message Information Words” on page 116 is a **Gap Error flag**. When CIRBCST = 1, bit 13 GAP/BCAST in Receive and Transmit “Subaddress Message Information Words” on page 112 and Receive and Transmit “Mode Command Message Information Words” on page 116 is a **Broadcast flag**.

Bit No.	*Mnemonic*	*Function*
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 35. 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 85; 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 120), the device 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 “14.4. Descriptor Table” on page 95). 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	IRQOFF	Suppress $\overline{\text{IRQ}}$ interrupt pin 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 85. 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 35 is updated. In addition, the $\overline{\text{IRQ}}$ output pin is normally asserted (low) if the corresponding bit is also set in the “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)”. This IRQOFF option bit modifies interrupt output behavior. The IRQOFF option bit prevents nuisance $\overline{\text{IRQ}}$ pin 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.

9.10. Time Tag Counter Configuration

Each device (RT 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 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 83) 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 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 RT time tag clock source.

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 device reach preset values contained in Time Tag Match registers. Refer to the Section 9.4.

9.11. Time Tag Counter Configuration Register (0x0039)

MTTTA1	MTTTA0	Reserved	Reserved	Reserved	Reserved	RTTTA1	RTTTA0	MTTCK3	MTTCK2	MTTCK1	MTTCK0	Reserved	TTCK2	TTCK1	TTCK0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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 device. 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>MT Configuration Register, bits 1-0</th><th>MT Operation</th><th>Time Tag Mode</th></tr>
<tr><td>01</td><td>SMT (simple) monitor mode</td><td>16-bit time tag</td></tr>
<tr><td>11</td><td>SMT (simple) monitor mode</td><td>48-bit time tag</td></tr>
</table>
When the MT is operating in Simple mode 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>
<tr><td>13 − 10</td><td>Reserved</td><td>R/W</td><td>0</td><td>These bits are not used.</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="2"><i>Function</i></th></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><i>Bits 9-8</i></td><td><i>Action</i></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="11">7
6
5
4</td><td rowspan="11">MTTCK3
MTTCK2
MTTCK1
MTTCK0</td><td rowspan="11">R/W</td><td rowspan="11">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><i>Bits 7-6-5-4</i></td><td><i>MT Time Tag Counter Clock Source</i></td></tr>
<tr><td>0-0-0-0</td><td>Time Tag Counter Disabled.</td></tr>
<tr><td>0-0-0-1</td><td>External clock provided at the MTTCLK input pin.</td></tr>
<tr><td>0-0-1-0</td><td>Internally generated 2µs clock.</td></tr>
<tr><td>0-0-1-1</td><td>Internally generated 4µs clock.</td></tr>
<tr><td>0-1-0-0</td><td>Internally generated 8µs clock.</td></tr>
<tr><td>0-1-0-1</td><td>Internally generated 16µs clock.</td></tr>
<tr><td>0-1-1-0</td><td>Internally generated 32µs clock.</td></tr>
<tr><td>0-1-1-1</td><td>Internally generated 64µs clock.</td></tr>
<tr><td>1-x-x-x</td><td>Internally generated 100ns clock.</td></tr>
<tr><td>3</td><td>Reserved</td><td>R/W</td><td>0</td><td colspan="2">These bits are not used.</td></tr>
<tr><td rowspan="10">2
1
0</td><td rowspan="10">TTCK2
TTCK1
TTCK0</td><td rowspan="10">R/W</td><td rowspan="10">0</td><td colspan="2">RT Time Tag Clock Selection Bits 2-0.
These bits select the clock source for the RT Time Tag Counters from the following options:</td></tr>
<tr><td><i>Bits 2-1-0</i></td><td><i>Time Tag Counter Clock Source</i></td></tr>
<tr><td>0-0-0</td><td>Time Tag Counter Disabled.</td></tr>
<tr><td>0-0-1</td><td>External clock provided at TTCLK input pin.</td></tr>
<tr><td>0-1-0</td><td>Internally generated 2µs clock.</td></tr>
<tr><td>0-1-1</td><td>Internally generated 4µs clock.</td></tr>
<tr><td>1-0-0</td><td>Internally generated 8µs clock.</td></tr>
<tr><td>1-0-1</td><td>Internally generated 16µs clock.</td></tr>
<tr><td>1-1-0</td><td>Internally generated 32µs clock.</td></tr>
<tr><td>1-1-1</td><td>Internally generated 64µs clock.</td></tr>
</table>

9.12. Memory Address Pointer Registers

The Serial Peripheral Interface (SPI) uses predefined 8-bit instruction op codes to perform a variety of predetermined actions. Some op codes must be followed by two or more operand bytes, while other SPI codes perform their desired action without additional operands. Examples of self-contained SPI op codes include the “Fast access” op codes used for reading or writing registers at the low end of the address space.

“Fast access” op codes used for direct addressing contain embedded register addresses, but only work over a limited address range.

- SPI reads to register addresses 0 through 0x000F (decimal 15) use an 8-bit op code of the form **0x00 + (Reg_Addr << 2)** where Reg_Addr = 0 to 0xF before left-shifting two bits.
- SPI writes to register addresses 0 through 0x003F (decimal 63) use an 8-bit op code of the form **0x80 + Reg_Addr** where Reg_Addr equals 0 to 0x3F.

The two “fast access” op codes use a dedicated memory address pointer to perform their duties without affecting values contained in other Memory Address Pointer registers. The “fast access” Memory Address Pointer cannot be read by the host, but is written each time a “fast access” op code is processed.

The HI-6136 uses a Memory Address Pointer for SPI reads to register addresses over 0x000F, or for SPI writes to register addresses over 0x003F. For most SPI read and write operations, the starting memory address for the requested operation is written to the Memory Address Pointer (or MAP) before the op code (using the MAP) is invoked.

In the case of a multiword data transfer involving a range of sequential addresses, the memory address pointer is initialized with the starting (lowest) address. After the SPI transfers data from the first address, the memory address pointer automatically increments to the next address. When read access occurs, the device prefetches the data stored at the next address to support the fastest possible data rates. As long as the chip select stays low (asserted) and the SPI master continues to provide serial clocks, data read/write transfers for sequential addresses continue until the chip select is negated. Please refer to section “Host Serial Peripheral Interface (SPI)” on page 160, describing SPI host access and the SPI op codes used for data transfer.

For flexibility in configuring the device, four independent Memory Address Pointers are available. These can be assigned in any manner that supports application requirements. For example:

- To simplify data access while supporting concurrent terminal devices (MT and RT), some devices may need a dedicated Memory Address Pointer (MAP) while other devices may be able to share a MAP.
- Consider using a dedicated Memory Address Pointer for interrupt service routines. Many SPI operations are multiword transfers that utilize the Memory Address Pointer auto-increment feature. If interrupts are enabled during multiword transfers, a dedicated Memory Address Pointer for the interrupt service routine avoids corruption of the MAP used by the interrupted routine.

Residing in the lower register address space, the four Memory Address Pointers can be read or written with a single 8-bit “fast access” op code (plus the desired 16-bit data value, when writing).

Just one of the four MAP registers is enabled at any time. Each of the four Memory Address Pointers has a dedicated 8-bit “MAP Select” op code that enables it by writing the “Master Configuration Register 1 (0x0000)”. Or the host can directly write the MAPSEL (Memory Address Pointer select) bits 11-10 in the “Master Configuration Register 1 (0x0000)” to enable the desired MAP register. Full descriptions of SPI data transfer methods are provided later in this document.

The four memory address pointer registers are:

MAP1 Memory Address Pointer Register	**0x000B**
MAP2 Memory Address Pointer Register	**0x000C**
MAP3 Memory Address Pointer Register	**0x000D**
MAP4 Memory Address Pointer Register	**0x000E**

MSB							Register Value								LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These 16-bit registers are read-write and are fully maintained by the host. These registers are cleared after $\overline{\text{MR}}$ pin master reset, but are unaffected by assertion of RTRESET or MTRESET bits in the “Master Status and Reset Register (0x0001)”.

Each of these registers has a unique SPI op code that reads the MAP value in the register, and another op code that writes a new MAP value into the register. See SPI op code table. The host selects the active MAP register by writing the MAPSEL (memory address pointer select) bits 11-10 in the “Master Configuration Register 1 (0x0000)”, or by using the four defined “MAP Select” SPI op codes, described in the section “Host Serial Peripheral Interface (SPI)” on page 160. The active MAP register contains the memory address used for SPI read write access to registers and RAM.

Please refer to Section 20 for a full description of the interface and the SPI instruction op codes.

10. SIMPLE MONITOR TERMINAL (SMT)

The HI-6136 can operate as an autonomous MIL-STD-1553 Bus Monitor, requiring minimal host support.

10.1. Overview

The 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).

In SMT mode, both circular buffers are fully utilized for recording message data. SMT monitor mode 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-6136 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 4.

Table 4. 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 10.2. 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 5. 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 5. 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 number of bytes stored in the Message Data Block. The range is 0 to 70 bytes, corresponding to 0 to 35 16-bit words stored.
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 10.2.
Word 2	Message Time Stamp Bits 47 ~ 32	Upper 16-bit word of message 48-bit time stamp.

Message Word Block	*Word Name*	*Word Function when using 16-bit time tag*
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 device 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 mode, the 8-word Monitor Address List is defined in Table 6.

Table 6. Monitor Address List for SMT Mode

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 device** 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.
Word 2	Command Buffer End Address	Host initialized, defines SMT Circular Command Buffer upper (rollover) address.

Address List Word	*Word Name*	*Description*
Word 1	Command Buffer Next Address	**Must be host initialized**, usually to match Command Buffer Start Address. **Updated by device** 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 4.

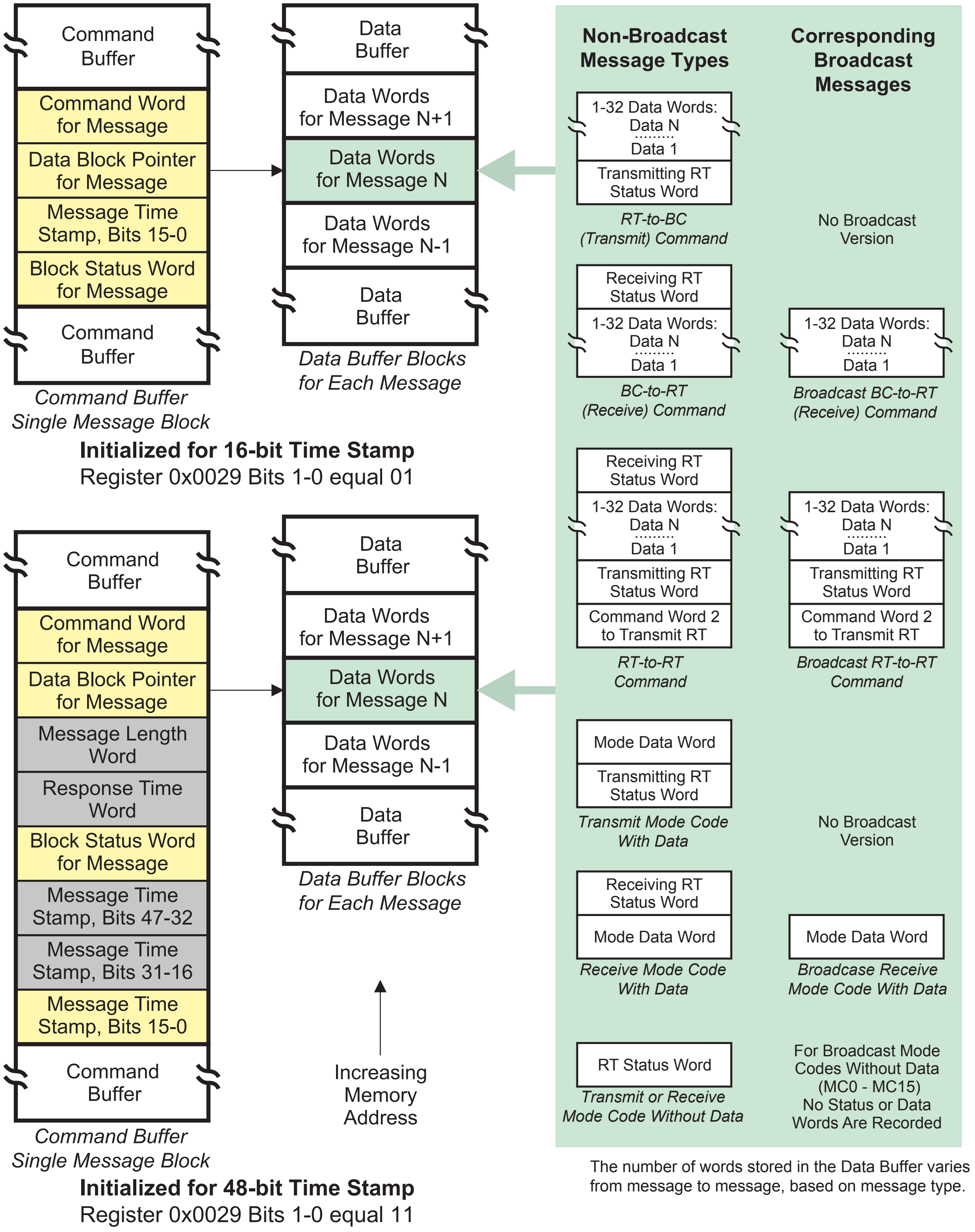


Figure 4. Simple Monitor Terminal (SMT) Data Storage

10.2. 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 mode 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	$\overline{\text{MR}}$ Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

The device 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	$\overline{\text{MR}}$ 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 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 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)

10.3. 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 $\overline{\text{MR}}$ 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 5.

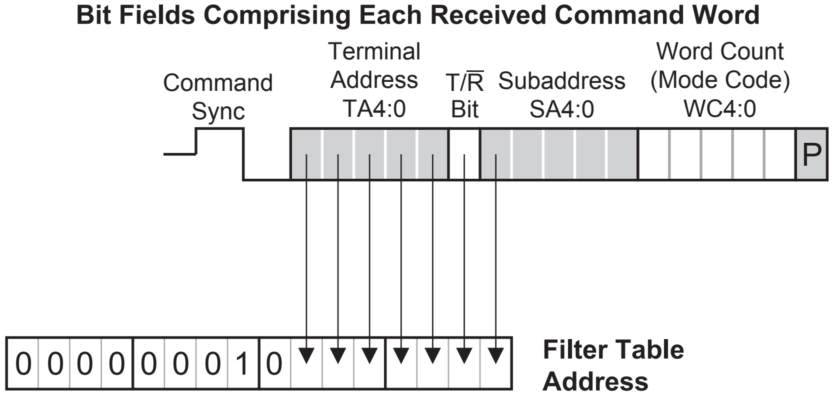


Figure 5. Deriving the Monitor Filter Table Address from the Received Command Word

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 7. This 4-word pattern repeats for all 32 Subaddresses, 0-31 decimal.

Table 7. 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.

11. SIMPLE MONITOR TERMINAL (SMT) REGISTER DESCRIPTION

In addition to the registers described here, the SMT Bus Monitor also utilizes one or more Memory Address Pointer registers (described in Section 9.12) for managing SPI read/write operations.

11.1. SMT Configuration Register (0x0029)

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

<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="4">15 − 14</td><td rowspan="4">MTTO1:0</td><td rowspan="4">R/W</td><td rowspan="4">0</td><td colspan="4">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.</td></tr>
<tr><th>Bit 15:14</th><th>Bus Dead Time</th><th>Time Out (excludes RT-RT)</th><th>RT-RT Time Out</th></tr>
<tr><td>00
01
10
11</td><td>16µs
21µs
80µs
138µs</td><td>18µs
23µs
82µs
140µs</td><td>61µs
66µs
122µs
180µs</td></tr>
<tr><td colspan="4">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 colspan="4">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 colspan="4">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 − 10</td><td>Reserved</td><td>R/W</td><td>0</td><td colspan="4">Bits 11-10 are not used by the bus monitor operating in SMT mode.
Initialize these bits to logic 0.</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><i>Bit 9:8</i></td><td><i>Time Tag Event</i></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 operating in SMT mode.
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 operating in SMT mode. 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 11.5.
0	Reserved	R	1	This bit is not used and always reads logic “1”.

11.2. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin master reset, this register is initialized with 0x00B0, the default base address of the MT Address Table in device RAM. The host can overwrite the default base address. 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 6 on page 52.

11.3. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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 6 on page 52).

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.

11.4. 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	$\overline{MR}$ 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 $\overline{MR}$ pin 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.

11.5. 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	$\overline{MR}$ 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:

11.6. 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	$\overline{MR}$ Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

11.7. 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	$\overline{MR}$ 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)”.

11.8. 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	$\overline{\text{MR}}$ 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:

11.9. 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	$\overline{\text{MR}}$ Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

11.10. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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.

11.11. 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	$\overline{\text{MR}}$ 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 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 time tag matching operations require the above register plus two additional registers:

11.12. 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	$\overline{\text{MR}}$ Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

11.13. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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 38.

11.14. SMT Bus Monitor Interrupt Registers and Their Use

Section 9.5 on page 34 through Section 9.7 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 $\overline{\text{IRQ}}$ 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 device. 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 RT interrupt pending bit (RTIP).
- If the matching bit is already set in the “SMT Bus Monitor Interrupt Output Enable Register (0x0015)”, an $\overline{\text{IRQ}}$ 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 $\overline{\text{IRQ}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the three Interrupt Output Enable registers (hardware, RT and SMT). Multiple interrupt-causing events can occur simultaneously, so single or simultaneous interrupt events can assert the $\overline{\text{IRQ}}$ host interrupt output.

When the host receives an $\overline{\text{IRQ}}$ signal from the device, it identifies the event(s) that triggered the interrupt. Section 9.5 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 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 “9.5. Hardware Interrupt Behavior” has a detailed description.

11.14.1. 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	$\overline{\text{MR}}$ Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

11.14.2. 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	$\overline{\text{MR}}$ Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

11.14.3. 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	$\overline{\text{MR}}$ 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 $\overline{\text{IRQ}}$ output pin is asserted at Pending Interrupt Register assertion. The SMT Interrupt Output Enable Register establishes two priority levels: high priority interrupts generate an $\overline{\text{IRQ}}$ 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.

12. REMOTE TERMINAL − OVERVIEW

The HI-6136 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-6121 integrated circuit. The Remote Terminal has its own Descriptor Table, command Illegalization Table and host interrupt configuration.

The following signal pins are provided for the Remote Terminal:

- RT Terminal Address 4 - 0 input pins
- RT Terminal Address Parity input pin
- RT (Address) Lock input pin
- RT Subsystem Fail input pins
- RT Mode Code 8 (Reset Remote Terminal) output pins

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 configuration, the serial auto-initialization EEPROM should already be programmed with the desired attributes for the terminal.

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

13. REMOTE TERMINAL REGISTERS

In addition to the registers described here, the RT can also utilize one or more Memory Address Pointer registers (described in Section 9.12) for managing SPI read/write operations.

13.1. Remote Terminal Configuration Register (0x0017)

RTTO1	RTTO0	RTINHA	RTINHB	BCSTINV	Reserved	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	$\overline{MR}$ 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 colspan="3"><i>Function</i></th></tr>
<tr><td rowspan="4">15 − 14</td><td rowspan="4">RTTO1:0</td><td rowspan="4">R/W</td><td rowspan="4">0</td><td colspan="3">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:</td></tr>
<tr><td><i>Bit 15:14</i></td><td><i>Bus Dead Time</i></td><td><i>RT-RT Time Out</i></td></tr>
<tr><td>00
01
10
11</td><td>15µs
20µs
58µs
138µs</td><td>57µs
62µs
100µs
180µs</td></tr>
<tr><td colspan="3">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 colspan="3">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 pin, 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 colspan="3">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 pin, or logic 1 at the TXINHB bit in the “Master Status and Reset Register (0x0001)”.</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
11	BCSTINV	R/W	0	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.
10	Reserved	R/W	0	This bit is not used and reads logic 0.
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 $\overline{\text{MR}}$ pin 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 16.5 on page 141.

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 14.3 on page 95. 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 9.1 on page 26). 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, device-controlled status bits.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
4	AUTOBSD	R/W	0	Automatic Bus Shutdown Enable. 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 device inhibits further transmission affected 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 decimal) mode command, (c) a “reset remote terminal” (MC8) mode command, (d) asserting hardware $\overline{\text{MR}}$ Master Reset input pin or (e) software reset initiated by setting the RTRESET bit in the “Master Status and Reset Register (0x0001)”. With the AUTOBSD bit set, the device only transmits RT status upon receiving MC4, MC5, MC20 or MC21. 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 pins with the same function. With AUTOBSD bit reset, upon receiving MC4 (or MC20 with data word matching “bus select” criteria), the device automatically fulfills transmit shutdown for the inactive bus. When the BSDTXO bit in the “Master Configuration Register 1 (0x0000)” equals 0, the inactive bus receiver is also shutdown (full bus shutdown vs. only transmit shutdown). The device affirms shutdown status by setting the corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. With AUTOBSD bit reset, upon receiving MC5 (or MC21 with data word matching “bus select” criteria), the device automatically re-enables both transmit and receive for the inactive bus (without regard to BSDTXO bit in the “Master Configuration Register 1 (0x0000)”). The device affirms shutdown override status by resetting corresponding “shutdown status” bits 15-12 in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
3 − 2	MC17OP1:0	R/W	0	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.
1	MC16OPT	R/W	0	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 the MCOPT1 bit is logic 1, the external host assumes responsibility for resetting the Service Request bit in the Status Word Bits register.
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.

13.2. Remote Terminal Operational Status Register (0x0018)

RTA4	RTA3	RTA2	RTA1	RTA0	RTAP	LOCK	Reserved	Reserved	Reserved	MCND	MCRD	MCTD	RTAPF	Reserved	Reserved	
These bits latch pins							0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ Master Reset input pin, register bits 15-9 capture the logic states (0 or 1) of the corresponding input pins 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 $\overline{\text{MR}}$ pin master reset, but are unaffected by assertion of RTRESET remote terminal software reset in the "Master Status and Reset Register (0x0001)".

<table>
<tr><th>Bit No.</th><th>Mnemonic</th><th>R/W</th><th>Reset</th><th>Function</th></tr>
<tr><td>15 − 11
10</td><td>RTA4:0
RTAP</td><td>R/W</td><td>0</td><td>Remote Terminal Address bits 4-0.
Remote Terminal Address Parity.
These bits reflect the state of the input pins RTA4 through RTA0 that applied at the rising edge of the $\overline{MR}$ master reset input signal (i.e. the active remote terminal address). The RTAP 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.</td></tr>
<tr><td>9</td><td>LOCK</td><td>R/W</td><td>0</td><td>Remote Terminal Address Lock.
This bit reflects the state of the LOCK input pin that applied at the rising edge of the $\overline{MR}$ master reset input signal. After reset, the host can overwrite bits 15-9 only if register LOCK bit 9 is logic 0. When the LOCK bit is high, the host cannot overwrite register bits 15-9. To restore host write capability for these bits, the $\overline{MR}$ master reset input signal must first be asserted with the LOCK input pin held low to restore register LOCK bit 9 to logic 0.</td></tr>
<tr><td>8 − 6</td><td>Reserved</td><td>R</td><td>0</td><td>These bits are not used.</td></tr>
<tr><td>5
4
3</td><td>MCND
MCRD
MCTD</td><td>R</td><td>0</td><td>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:
<table>
<tr><th>Current Command Type</th><th>Bits 5-4-3</th><th>Current Command Word</th></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>
</table></td></tr>
<tr><td>2</td><td>RTAPF</td><td>R</td><td>0</td><td>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>These bits are not used.</td></tr>
</table>

13.3. 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	$\overline{\text{MR}}$ Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These 16-bit registers are read-only and are fully maintained by the device. These registers are cleared after $\overline{\text{MR}}$ pin master reset, but are 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.

13.4. 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	$\overline{\text{MR}}$ Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

These 16-bit registers are read-only and are fully maintained by the device. These registers are cleared after $\overline{\text{MR}}$ pin master reset, but are unaffected by assertion of RTRESET remote terminal software reset in the "Master Status and Reset Register (0x0001)".

These registers contain the address for the descriptor table Control Word corresponding to the current command stored in the Current Command Register, above. These registers are 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.

13.5. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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.

13.6. 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	$\overline{MR}$ 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 $\overline{MR}$ pin 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 device. 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 device 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.
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 device 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 41.
2	SSYSF	R/W	0	Subsystem Fail status bit. The host maintains this read-write bit. This register bit is logically ORed with the RTSSF input pin. If either SSYSF register bit or RTSSF pin is asserted, the SSYSF Subsystem Flag status bit is set. If the RT’s Configuration Register MCOPT1 bit equals 0, reception of a “transmit vector word” mode command (MC16) causes automatic reset of the SSYSF status bit in this register; when this occurs, the register bit is reset before status word transmission begins.
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.

13.7. 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	$\overline{\text{MR}}$ 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 device. This register is cleared after $\overline{\text{MR}}$ pin 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.

13.8. 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	$\overline{\text{MR}}$ 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 device. This register is cleared after $\overline{\text{MR}}$ pin 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 device 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 device 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 device 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.

13.9. 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	$\overline{\text{MR}}$ 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 device. This register is cleared after $\overline{\text{MR}}$ pin 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 device 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 device 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 device 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.

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

TXASD	TXBSD	RXASD	RXBSD	User Assigned Bits						LBFA	LBFB	BMTF	RT	EELF	TFBINH	
P	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{\text{MR}}$ 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 device when predetermined events occur. The host may overwrite the device-written bits 5 and 4. After $\overline{\text{MR}}$ pin master reset, bits 13-12, 5-4 and 0 are reset. Bits 15-14 will be set if the corresponding TXINHA or TXINHB input pins are high. Bits 3-1 will be set if RT address parity error, or post-MR memory test failure or auto-initialization 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 pin, 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 "19.2.1. Self-Test Control Register (0x0028)" on page 154).

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 device Random Access Memory (RAM) (see Section “19.2.1. Self-Test Control Register (0x0028)” on page 154).
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 $\overline{\text{MR}}$ master reset, bits 15:10 in the RT’s Operational Status Register reflect input pin states, but will be overwritten if subsequent auto-initialization is performed (if AUTOEN pin is high) and the initialization EEPROM contains different data for RT Operational Status Register bits 15:10.
1	EELF	R	0	Auto-Initialization EEPROM Load Fail. This bit only applies when auto-initialization is enabled (AUTOEN input pin state equals 1). This bit is set if, after $\overline{\text{MR}}$ master reset, failure occurs when copying serial EEPROM to registers and RAM. When this occurs, bit 0 or bit 1 will be set in the “Master Status and Reset Register (0x0001)” to indicate type of failure.
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).

13.11. 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	$\overline{\text{MR}}$ 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 $\overline{\text{MR}}$ pin 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.

13.12. 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	$\overline{MR}$ 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 $\overline{MR}$ pin 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 device 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.

13.13. 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	$\overline{MR}$ 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 $\overline{MR}$ pin 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:

13.13.1. 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)”.

13.13.2. 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)”.

13.14. Remote Terminal Interrupt Registers and Their Use

Section 9.5 on page 34 through Section 9.7 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 $\overline{\text{IRQ}}$ 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 device. 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 MT interrupt pending bit (MTIP).
- If the matching bit is already set in the “Remote Terminal (RT) Interrupt Output Enable Register (0x0016)”, an $\overline{\text{IRQ}}$ 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 $\overline{\text{IRQ}}$ host interrupt output signal is shared by all enabled interrupt conditions having bits set in the three Interrupt Output Enable registers (hardware, RT and SMT). Multiple interrupt-causing events can occur simultaneously, so single or simultaneous interrupt events can assert the $\overline{\text{IRQ}}$ host interrupt output.

When the host receives an $\overline{\text{IRQ}}$ signal from the device, it identifies the event(s) that triggered the interrupt. Section 9.5 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 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 9.5 has a detailed description.

13.14.1. 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	$\overline{\text{MR}}$ Reset
RW													R			Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.14.2. 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	$\overline{\text{MR}}$ Reset
R																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

13.14.3. 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	$\overline{\text{MR}}$ 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 $\overline{\text{IRQ}}$ output is asserted at message completion. The RT Interrupt Output Enable Register establishes two priority levels: high priority interrupts generate an $\overline{\text{IRQ}}$ 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 IRQOFF bit 0 is set in “Extended Configuration Register (0x004D)” on page 41, $\overline{\text{IRQ}}$ pin 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 35 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 41 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:

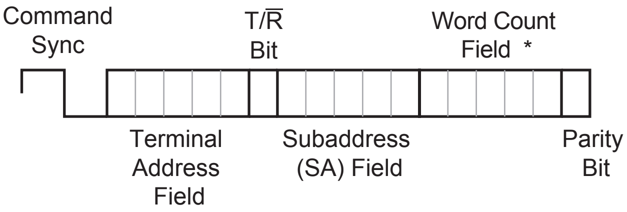
Bit No.	*Mnemonic*	*Function*
15 − 10	Reserved	These bits are not used.
9	Reserved	Bit 9 cannot be written and reads back logic 0.
8	RTMC8	RT Mode Code 8 Command Interrupt. Remote terminal processed a valid MIL-STD-1553 “reset remote terminal” mode code command. An RTMC8 interrupt notifies the host when 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 $\overline{\text{IRQ}}$ output, but the $\overline{\text{RTMC8}}$ output pin is asserted to indicate that the RT needs reset. In the case of a broadcast command to RT31, $\overline{\text{RTMC8}}$ output pin 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 $\overline{\text{IRQ}}$ 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.

14. REMOTE TERMINAL CONFIGURATION AND OPERATION

14.1. 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 6, 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.



* Word Count field is replaced by Mode Code field when the SA field equals 0x00 or 0x1F

Figure 6. 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 device. If the UMCINV configuration bit equals 1, the undefined mode commands are treated as invalid. They are not recognized by the device. 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 14.2 on page 90. After $\overline{\text{MR}}$ 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. For example, command MC0 (with T/$\overline{\text{R}}$ = 1) is probably unimplemented because the device does not have provisions for accepting “dynamic bus control”.

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 $\overline{\text{MR}}$ 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 device 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.

14.1.1. 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 device 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 device 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 device 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.

14.2. 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 $\overline{\text{MR}}$ master reset

The device uses a 256-word “Illegalization Table” in shared RAM to distinguish between legal and illegal commands. After the ($\overline{\text{MR}}$) master reset input is negated, the device 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 (or auto-initialization) 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. Host initialization of the table can be replaced by auto-initialization.

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 7. 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 7.

Bit Fields Comprise Each Received Command Word

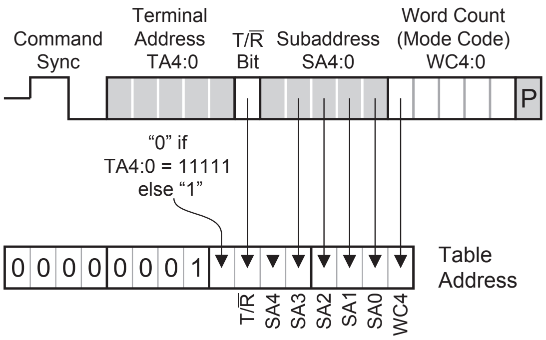


Figure 7. Deriving the Illegalization Table Address From the Received Command Word

Figure 9 on page 94 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-$\overline{\text{MR}}$ 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 8, 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).

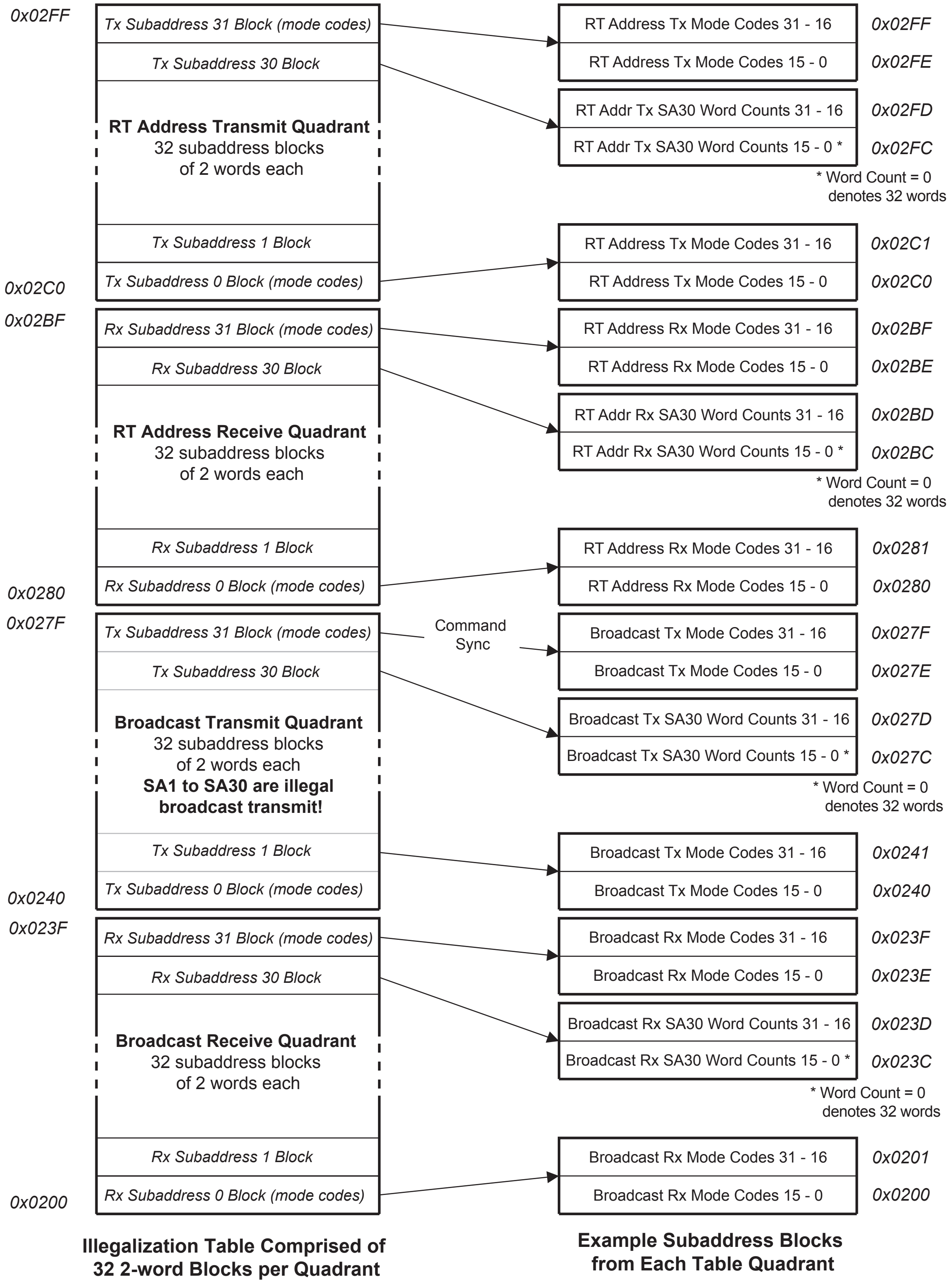


Figure 8. Address Mapping for Illegalization Table

Note: Default start address is 0x0200.

Figure 9 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 9. Summary of RT Illegalization Table Addresses for Mode Code Commands

14.3. Temporary Receive Data Buffer

The 32-word temporary receive data buffer resides in shared RAM in address space 0x01C0 to 0x01DF. The device 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.

14.4. Descriptor Table

The Descriptor Table, resides in shared RAM, at default address ranges 0x0400 to 0x05FF. This table is initialized by the host (or auto-initialization) 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 10, 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/$\overline{\text{R}}$ bit equals 0, the receive command quadrant applies. When the T/$\overline{\text{R}}$ 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/$\overline{\text{R}}$ bit equals 0, while the Transmit Mode Command quadrants applies when T/$\overline{\text{R}}$ equals 1.

The term "Transmit Mode Command" is misleading. All defined mode commands with mode code less than 0x0F haveT/$\overline{\text{R}}$ 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 11 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 (or auto-initialization) 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.

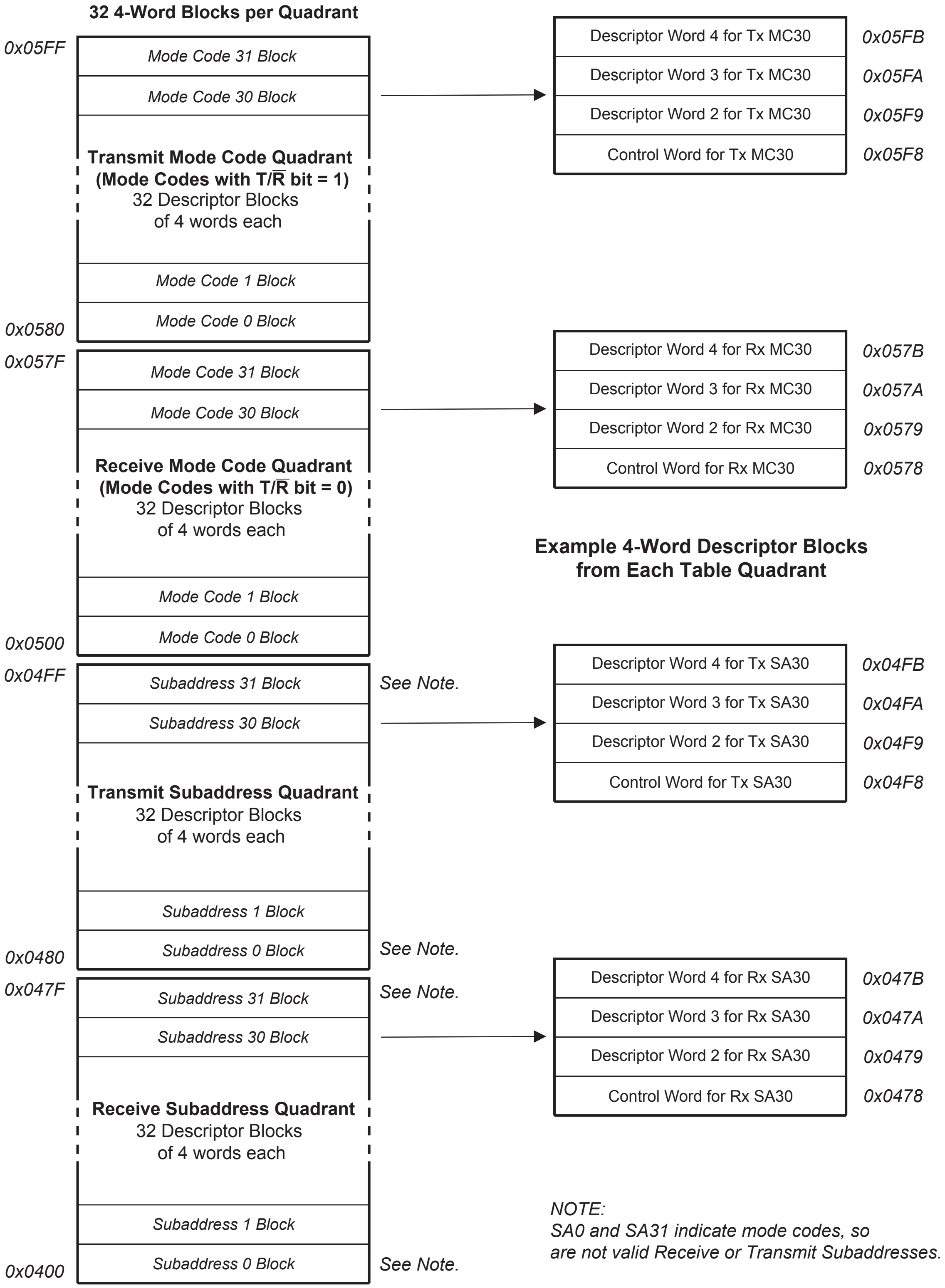
Circular Buffer Mode 1 where buffer boundaries determine when the bulk transfer is complete and message information and time-tag words are stored with message data in a common buffer. Several host interrupt options are offered, including an interrupt generated when the allocated data buffer is full.

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 device updates the command’s descriptor Control Word. Update will differ based on the chosen data buffer method. Reading the descriptor table can differ from other RAM accesses. See Sections 20.6 and 20.8.



NOTE:
SA0 and SA31 indicate mode codes, so
are not valid Receive or Transmit Subaddresses.

Figure 10. 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.

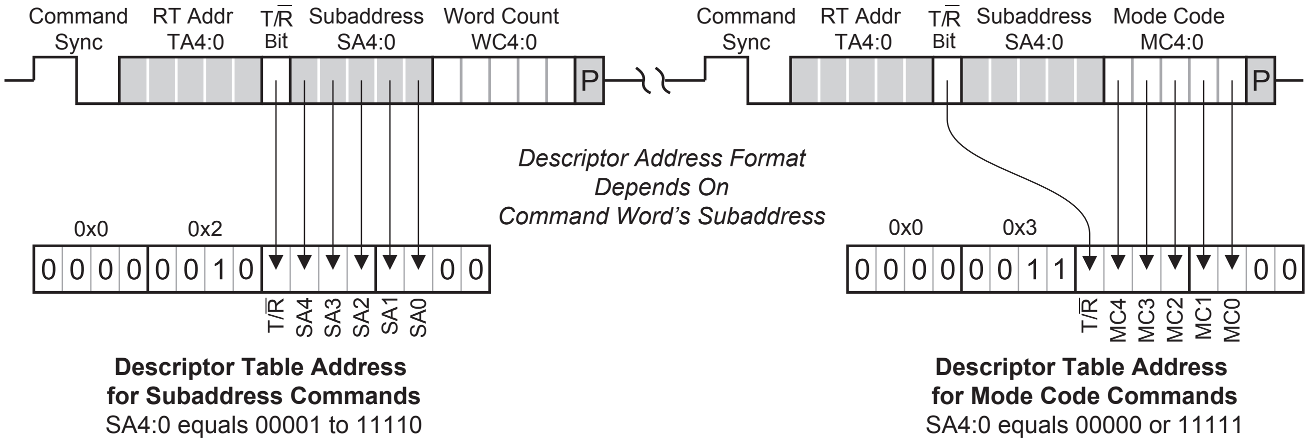


Figure 11. Deriving a Descriptor Table Control Word Address From Command Word
(assumes table base address = 0x0200)

14.4.1. Receive Subaddress Control Word

Receive Subaddress Control Words apply when a valid command word T/R̄ 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 device 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 M̄R̄ 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 device
D1	Bit set by device, reset by host read cycle

NOTE: '*Reset*' refers to bit value following Master Reset (M̄R̄). 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 pin $\overline{\text{INT}}$ 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 pin $\overline{\text{INT}}$ 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 pin $\overline{\text{INT}}$ 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 device.
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 device 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 $\overline{\text{MR}}$ 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 device 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 41.** 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 121). The DBP bit is reset to logic 0 by $\overline{\text{MR}}$ 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. Device 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 pin $\overline{\text{INT}}$ 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 device. This bit is reset to logic 0 by $\overline{\text{MR}}$ master reset or SRST software reset.
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is controlled by the device 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. Device 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 device to temporarily disable ping-pong by asserting STOPP bit 3. The device 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 device to re-enable ping-pong by negating STOPP bit 3. The device 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 15.6, 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 device to re-enable ping-pong. The device confirms recognition of ping-pong enable or disable status by writing PPON bit 8. Refer to Section 15.3, 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 device to temporarily disable ping-pong. Negating the STOPP bit asks the device to re-enable ping-pong. The device 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><th>PPEN</th><th>CIR2EN</th><th>CIR1EN</th><th>Buffer Mode</th></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>

14.4.2. 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 device 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 $\overline{MR}$ 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.**

	IXEQZ	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 device
D1	Bit set by device, 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 ($\overline{MR}$). 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 pin $\overline{\text{INT}}$ 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 pin $\overline{\text{INT}}$ 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 device 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 $\overline{\text{MR}}$ 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 device 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 41.** 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 121). The DBP bit is reset to logic 0 by $\overline{\text{MR}}$ 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 device 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 device 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 device. This bit is reset to logic 0 by $\overline{\text{MR}}$ master reset or SRST software reset.
8	PPON	R/W	0	Ping-Pong Enable Acknowledge. This bit is controlled by the device 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 15.6, 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 15.3, which describes ping-pong mode in detail.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
2	PPEN	R/W	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 device to temporarily disable ping-pong. Negating the STOPP bit asks the device to re-enable ping-pong. The device 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)”.
1	CIR2EN	R/W	0	
0	CIR1EN	R/W	0	

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

14.4.3. 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 15.4.1). An alternative called **Simplified Mode Command Processing** (SMCP) may be globally applied for all mode code commands (see Section 16.5).

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 16.5.

14.4.4. 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 device 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 $\overline{MR}$ 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 device
D1	Bit set by device, 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 device
D1	Bit set by device, 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 ($\overline{MR}$). 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 $\overline{INT}$ output pin 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 pin 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 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 pin 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 device.
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 device 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 $\overline{\text{MR}}$ 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 device 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 41.** 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 121). The DBP bit is reset to logic 0 by $\overline{\text{MR}}$ 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. Device 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 device. This bit is reset to logic 0 by $\overline{\text{MR}}$ 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 device 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 device to temporarily disable ping-pong by asserting STOPP bit 3. The device 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 device to re-enable ping-pong by negating STOPP bit 3. The device 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 device to re-enable ping-pong. The device 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 device to temporarily disable ping-pong. Negating the STOPP bit asks the device to re-enable ping-pong. The device 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.

14.4.5. 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 device 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 $\overline{\text{MR}}$ 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 device
D1	Bit set by device, 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 device
D1	Bit set by device, 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 ($\overline{\text{MR}}$). 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 pin 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 pin 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 pin 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 device.
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 device 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 $\overline{\text{MR}}$ 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 device 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 41.** 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 121). The DBP bit is reset to logic 0 by $\overline{\text{MR}}$ 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. Device 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 device. This bit is reset to logic 0 by $\overline{MR}$ 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 device 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 device to temporarily disable ping-pong by asserting STOPP bit 3. The device 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 device to re-enable ping-pong by negating STOPP bit 3. The device 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 device 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 device to re-enable ping-pong. The device 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 device to temporarily disable ping-pong. Negating the STOPP bit asks the device to re-enable ping-pong. The device 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.

15. 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 device 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 device 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 device 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 device 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 15.4).**

   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 15.3).**

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

3. **Circular Buffer Mode 1 (see 15.5).**

   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 15.6).**

   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 8.

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 16.5.

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 8. 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.

15.1. Subaddress Message Information Words

15.1.1. Receive Subaddress Command

For receive subaddress commands, the device stores the received data words plus two additional words. The device adds a receive subaddress Message Information Word and a Time-Tag Word to the received data words. The device 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 device 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	Device 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 TOSEL0-1 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 41 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 41 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 14.2 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/$\overline{R}$ 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.

15.1.2. 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 device 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	Device, 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 41 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 41 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 14.2 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.

15.2. 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.

15.2.1. 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 device 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 10.

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	Device, 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 device 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 device 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	Device, 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 41 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 41 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 14.2 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.

15.2.2. 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 device automatically transmits the data word then copies the transmitted data value to the stored data structure.

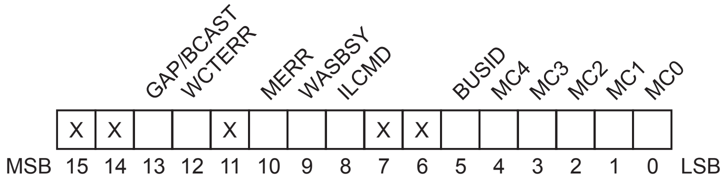
Here is an example data structure for a transmit mode command with data (mode code values 0x10 through 0x1F). This applies to MC16 “Transmit Vector Word”. Notice that the data pointer points to the data structure starting address, not the mode data word. The data pointer is located in the transmit mode 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	Device, 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 device 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 device 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	Device, 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 41 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 41 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 14.2 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.

15.3. Ping-Pong Data Buffering

15.3.1. 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 device retrieves buffer data for transmit commands, or stores buffer data for receive commands. During ping-pong operation, the device 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 device 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 41. 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 12 is a general illustration of ping-pong buffer mode. Figure 13 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).

15.3.2. Ping-Pong Enable / Disable Handshake

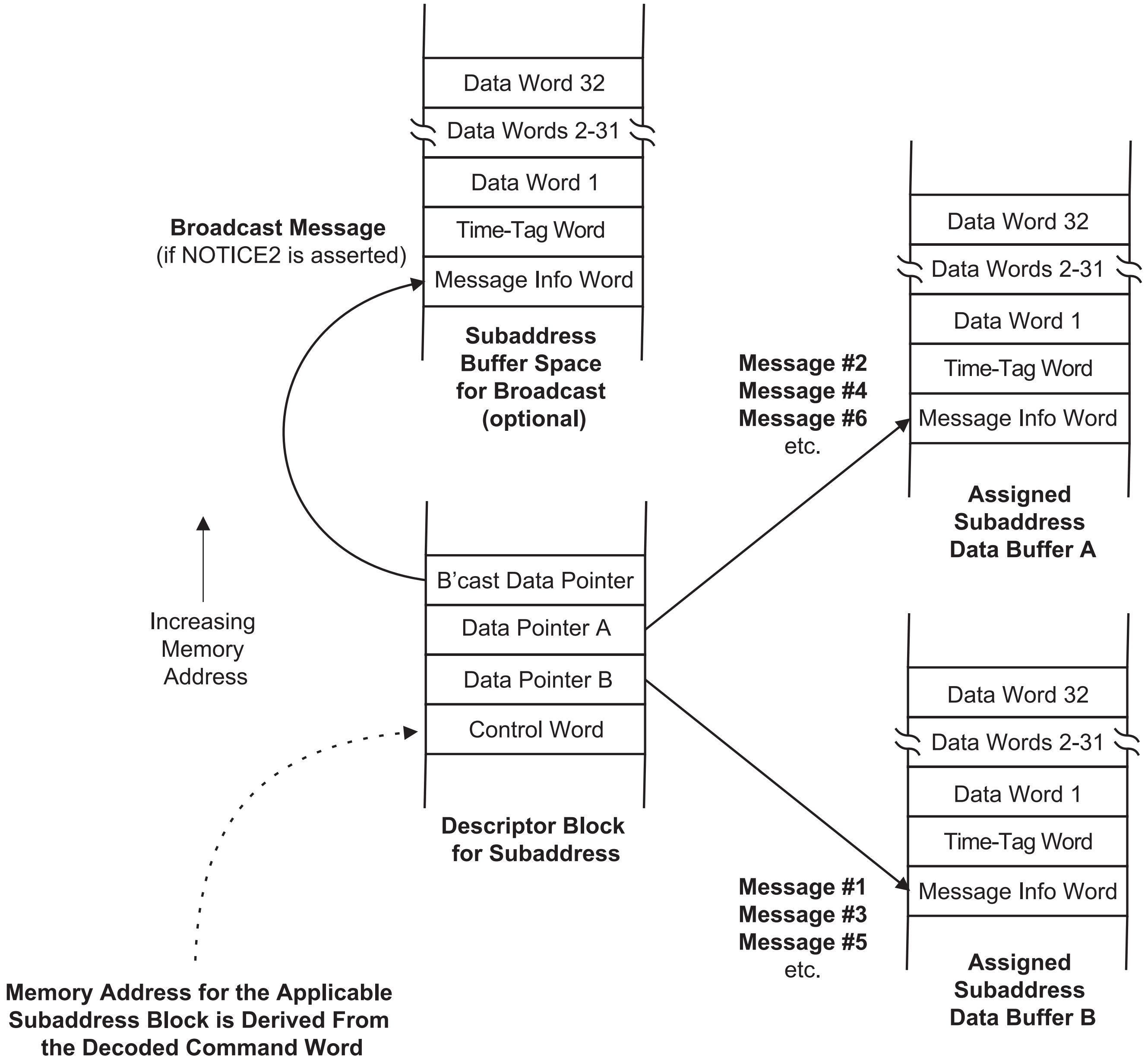
Because ping-pong messages and host buffer servicing are asynchronous, there is potential for “data collision”. Here is a data collision example: The host reads data from an earlier message while the device simultaneously writes new message data to the same buffer. The host reads a mix of new and old 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 device 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 device 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 device 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 12. Illustration of Ping-Pong Buffer Mode

15.3.3. 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 device 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)”.

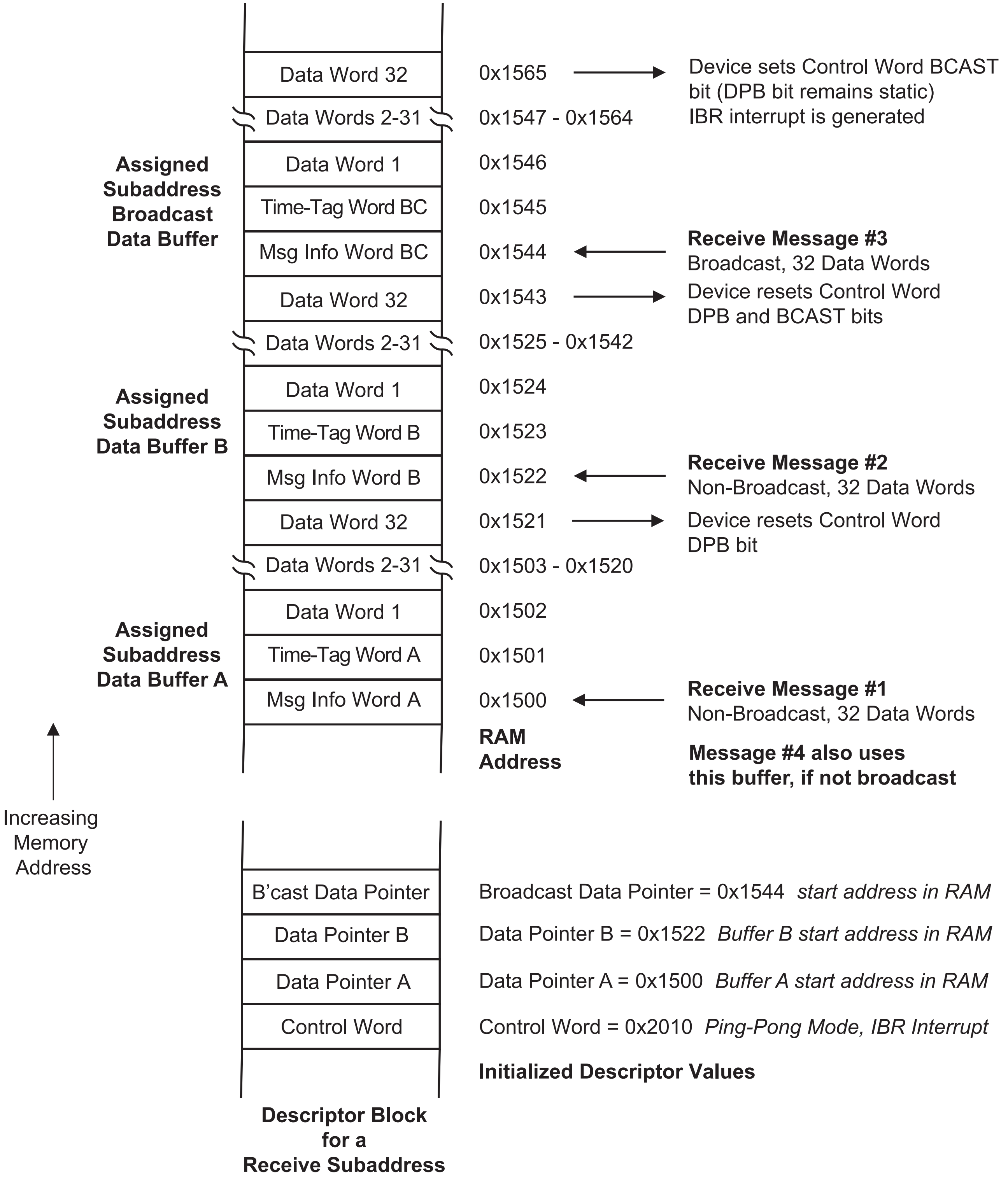
When a broadcast command is received, message information and data is stored in the broadcast data buffer and an $\overline{\text{INT}}$ interrupt is generated. The host must read the Interrupt Log to determine the originating subaddress (or mode code), then service the broadcast data buffer for that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



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 device segregates data from broadcast and non-broadcast messages. Message #2 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 13. Ping-Pong Buffer Mode Example for a Receive Subaddress

15.4. 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 device 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 14 is a general illustration of indexed single buffer mode. Figure 15 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 device 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.

15.4.1. 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.

15.4.2. 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 device 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)".

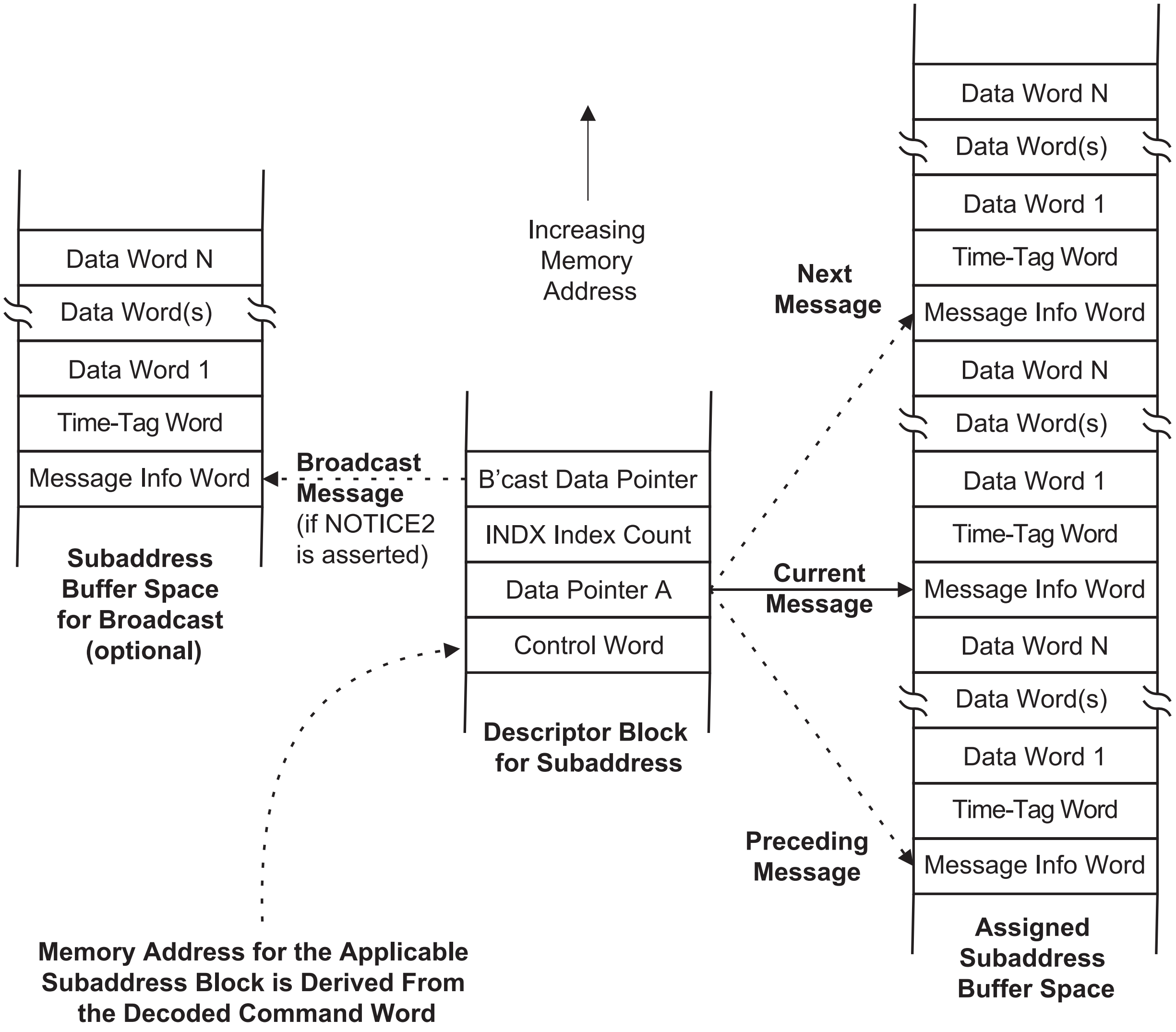
When a broadcast command is received, message information and data are stored in the broadcast data buffer. If descriptor Control Word IBR bit is set, an $\overline{\text{INT}}$ interrupt is generated. The host must read the Interrupt Log to determine the originating subaddress (or mode code) then service 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 14. Illustration of Indexed Buffer Mode

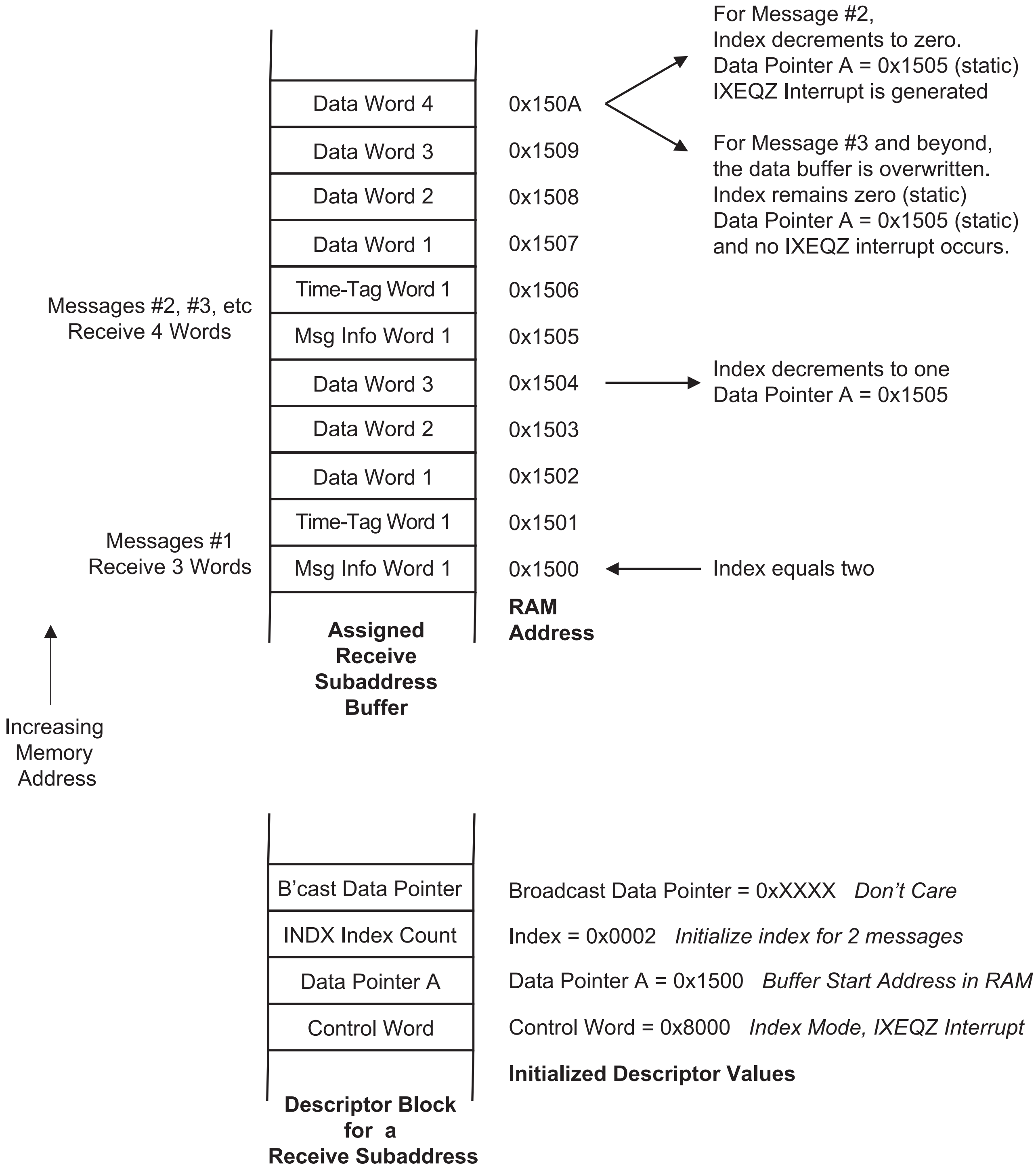


Figure 15. Indexed Buffer Mode Example for a Receive Subaddress (broadcast disabled)

15.5. Circular Buffer Mode 1

The device 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 16 provides a generalized illustration of Circular Buffer Mode 1, while Figure 17 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 device enters a Message Information word, Time-Tag word and data word(s) into the circular receive buffer. For each valid transmit message, the device enters a Message Information word and a Time-Tag word into reserved memory locations within the circular transmit buffer. The device 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 bottom 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 device 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/$\overline{\text{R}}$ 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 device 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 device 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 device 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 device reading the unique descriptor block for the subaddress or mode code specified by the T/$\overline{\text{R}}$ bit, subaddress and word count fields in the received command word.

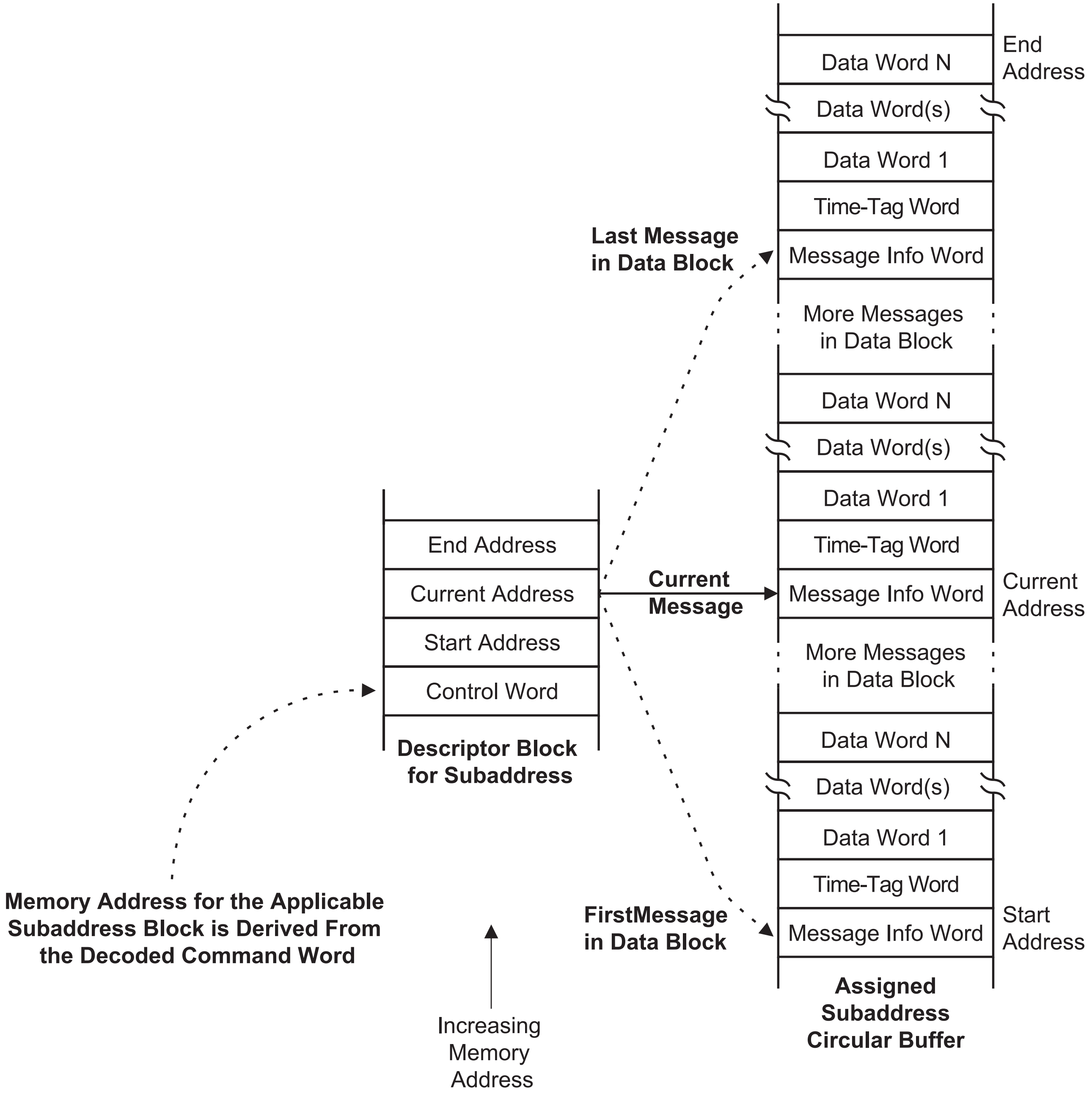
For receive messages, the device stores “N” received data words in the circular data buffer. The first data word received is stored at the location specified by the CA pointer +2. After message completion, the device stores the Message Information word and Time-Tag words to addresses CA and CA+1 respectively. If no errors were detected, the device 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 device 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 device 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 device writes the Message Information and Time-Tag words into the buffer. If no errors were detected, the device 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 device indicates "transmit buffer empty" by asserting the $\overline{\text{INT}}$ interrupt output.

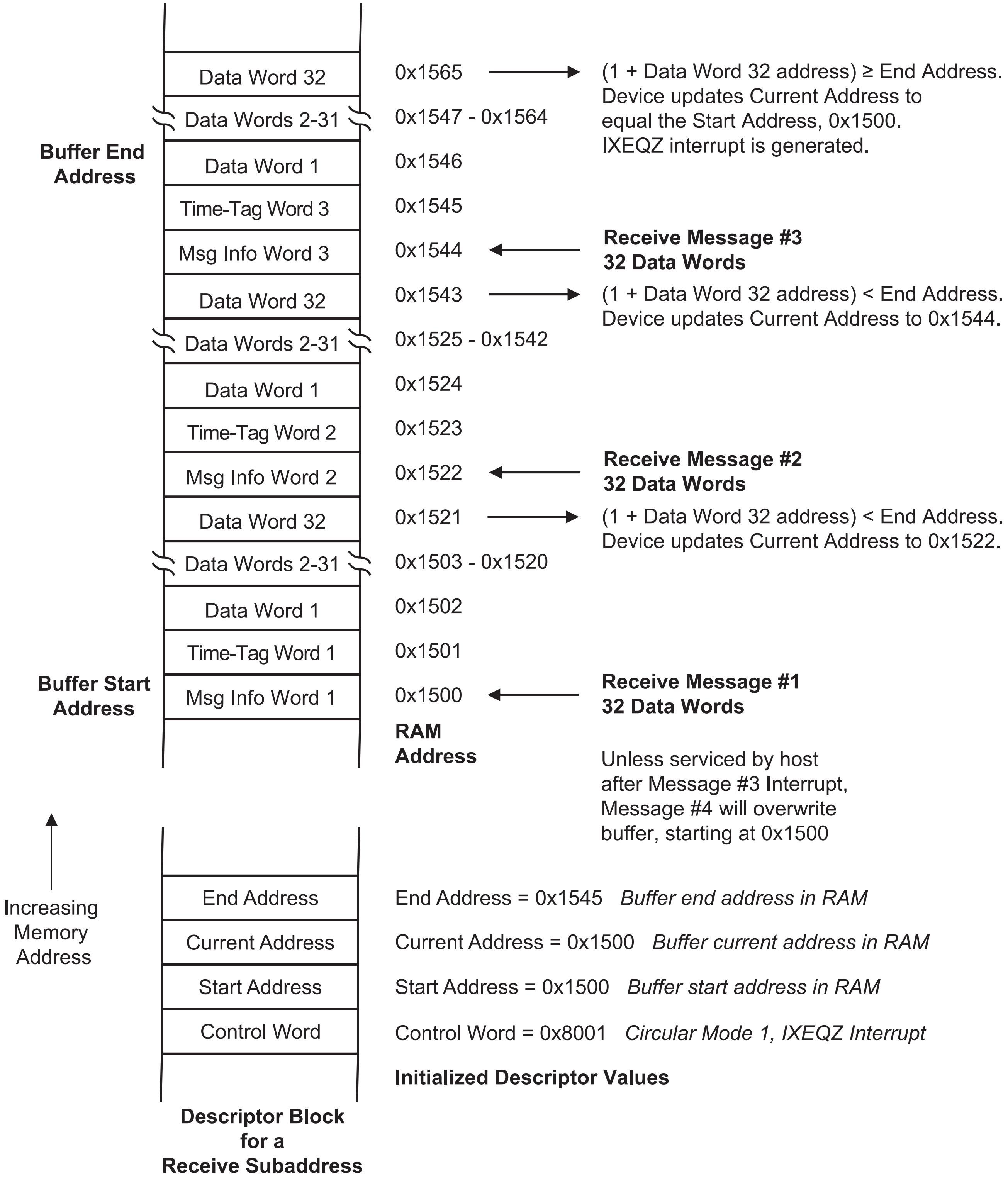
Device hardware does not segregate broadcast and non-broadcast data for this circular buffer mode, even when the NOTICE2 bit is set in the "Remote Terminal Configuration Register (0x0017)". Data words from broadcast receive commands are stored in the same buffer with data from non-broadcast receive commands. 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 16. Illustration of Circular Buffer Mode 1



(1 + Data Word 32 address) ≥ End Address. Device updates Current Address to equal the Start Address, 0x1500. IXEQZ interrupt is generated.

(1 + Data Word 32 address) < End Address. Device updates Current Address to 0x1544.

(1 + Data Word 32 address) < End Address. Device updates Current Address to 0x1522.

Unless serviced by host after Message #3 Interrupt, Message #4 will overwrite buffer, starting at 0x1500

End Address = 0x1545 *Buffer end address in RAM*

Current Address = 0x1500 *Buffer current address in RAM*

Start Address = 0x1500 *Buffer start address in RAM*

Control Word = 0x8001 *Circular Mode 1, IXEQZ Interrupt*

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 device behavior.
The circular buffer should have a 33-word pad beyond its End Address to deal with buffer overrun without data loss.

Figure 17. Circular Buffer Mode 1 Example for a Receive Subaddress

15.6. 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 18 is a generalized illustration of Circular Buffer Mode 2, while Figure 19 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 device 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 device 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 device.

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 9. 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 device can restore the MIBA pointer to the initial MIB base address after the predetermined message count is transacted. As illustrated in Table 9, 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 9) 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 9 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 9, 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 device 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 device 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 device 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 9. Circular Buffer Mode 2 (Initialization factors based on message block size)

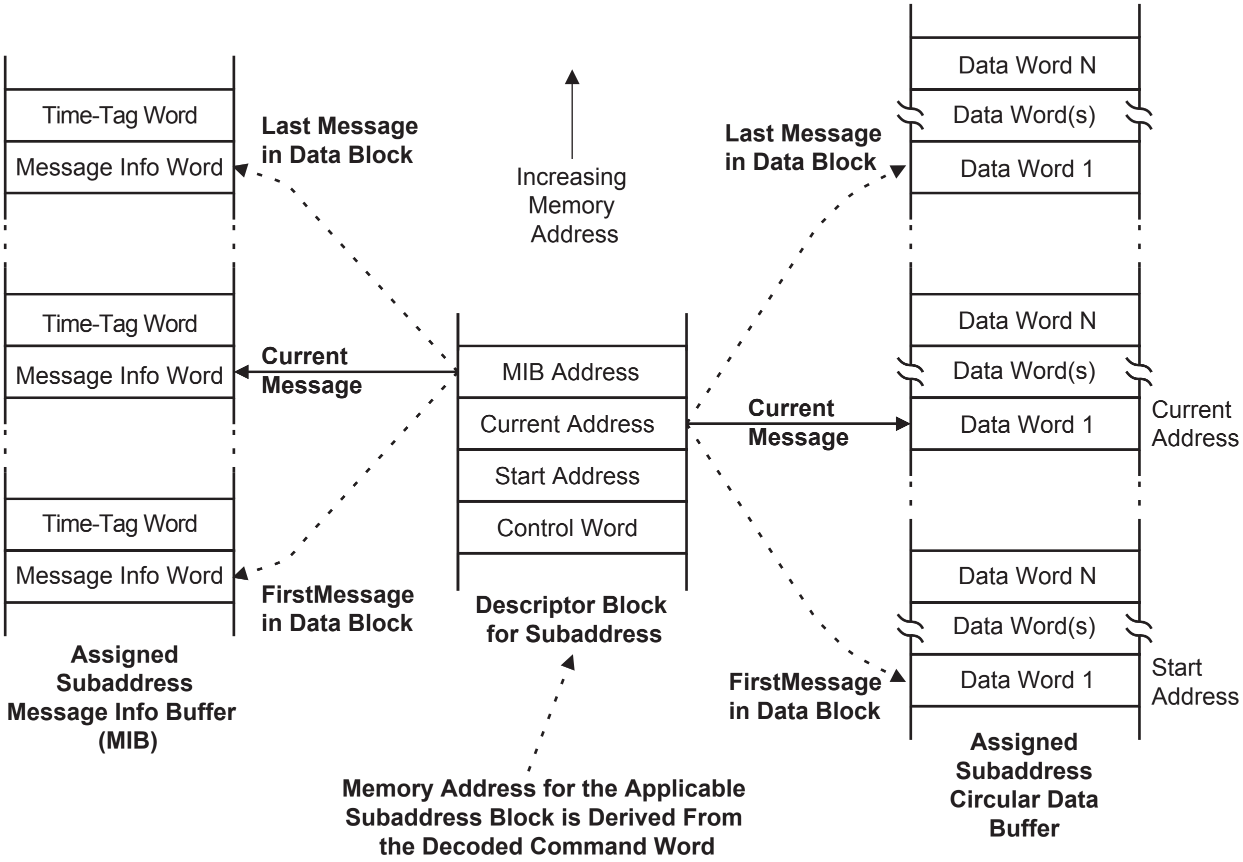
Number of Messages	Control Word Bits 7:4 CIR2ZN Field	Required Data Space if 32 Words / Msg	Required MIB Space, 2 Words / Msg	Initial MIBA Value, Showing the Required Leading and Trailing Zeros
2	0010 (2)	64	4	0xxxxxxxxxxxxx00
4	0011 (3)	128	8	0xxxxxxxxxxxx000
8	0100 (4)	256	16	0xxxxxxxxxxx0000
16	0101 (5)	512	32	0xxxxxxxxxx00000
32	0110 (6)	1,024	64	0xxxxxxxxx000000
64	0111 (7)	2,048	128	0xxxxxxxx0000000
128	1000 (8)	4,096	256	0xxxxxxx00000000
256	1001 (9)	8,192	512	0xxxxxx000000000
512	1010 (A)	16,384	1,024	0xxxxx0000000000

For receive subaddresses using Circular Buffer Mode 2, the device 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 device 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 device 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 device generates a Buffer Full / Empty interrupt, asserting the $\overline{\text{INT}}$ interrupt output.

For transmit subaddresses using Circular Buffer Mode 2, the device 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 device 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 device 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 device generates a Buffer Full / Empty interrupt, asserting the $\overline{\text{INT}}$ interrupt output.

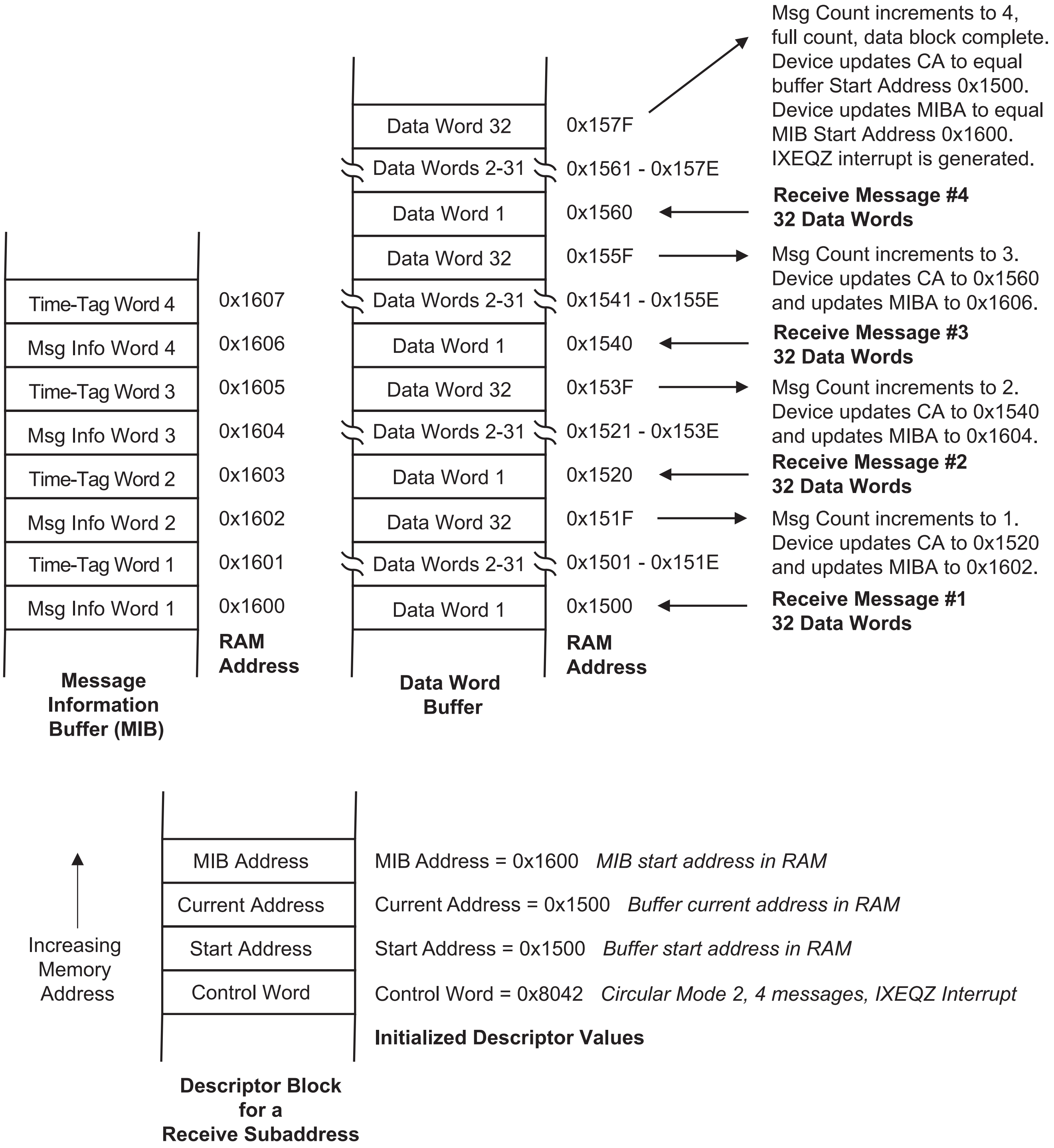
Device hardware does not segregate broadcast and non-broadcast data for this circular buffer mode, even when the NOTICE2 bit is set in the “Remote Terminal Configuration Register (0x0017)”. Data words from broadcast receive commands are stored in the same buffer with data from non-broadcast receive commands. 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 18. 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 19. Circular Buffer Mode 2 Example for a Receive Subaddress

16. REMOTE TERMINAL MODE COMMAND PROCESSING

16.1. General Considerations

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

If the UMCINV bit is low (default after $\overline{MR}$ 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 table bit equals 0, the mode command is legal; the RT responds “in form” and updates status. If the command’s 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 10 describes explicit terminal response for each mode code value and command T/$\overline{R}$ 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 device 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/$\overline{R}$ 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/$\overline{R}$ = 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.

16.2. 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 $\overline{INT}$ interrupt output is asserted.

Before $\overline{INT}$ interrupt assertion, the device 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/$\overline{R}$ bit state. The host reads the IAW to determine the command that caused the interrupt.

16.3. 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 10. 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.

16.4. 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 device 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 device, 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.

16.5. 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 device 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.

The Appendix 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”.

17. SERIAL EEPROM PROGRAMMING UTILITY

The HI-6136 can program a serial EEPROM via the dedicated EEPROM SPI port for subsequent auto-initialization events. The device copies host-configured registers and RAM (configuration tables and possibly data buffers) to serial EEPROM.

Compatible SPI serial EEPROMs are 3.3V, operate in SPI modes 0 or 3 and have 128-byte pages. The serial SPI data is clocked at 8.3 MHz SCK frequency. A 2K x 8 EEPROM can restore the lower 1K x 16 device address space. A 16K x 8 EEPROM retains the entire 8K x 16 register/RAM address space.

17.1. Writing the Auto-Initialization EEPROM

A deliberate series of events initiates copy of data from registers and RAM to serial EEPROM. This reduces the likelihood of accidental EEPROM overwrites. **Note: The RT address must have correct (odd) parity before EEPROM read or write can occur.** The following series of events must occur to initiate programming:

1. **Using a fresh host initialization immediately following $\overline{\text{MR}}$ master reset as the basis for EEPROM copy**

   Until EEPROM reprogramming is complete, disconnect the terminal from MIL-STD-1553 buses, or take other measures to prevent bus activity detection by the device. **With the AUTOEN, TXINHA and TXINHB pins in logic 0 state,** apply $\overline{\text{MR}}$ master reset and wait for READY output assertion. Verify that the $\overline{\text{IRQ}}$ interrupt output does not pulse low at READY assertion, indicating likely RT address parity error at the RTA4:0 and RTAP pins. Using known good parameters, the host initializes device registers, the RAM descriptor table and transmit data buffers (if necessary).

   - If auto-initialization will be used to configure the Bus Monitor, the MTENA bit 8 in “Master Configuration Register 1 (0x0000)” should be logic 0. Until EEPROM programming is complete, the terminal should be disconnected from MIL-STD-1553 buses (or other measures taken) to prevent bus activity detection by the monitor.
   - If auto-initialization will be used to configure the Remote Terminal, the RTENA bit 6 in “Master Configuration Register 1 (0x0000)” should be logic 1, but RTSTEX register bit 4 must remain in the post-reset logic 0 state.

2. **Using the existing EEPROM configuration as the baseline for a new EEPROM configuration**

   Until EEPROM reprogramming is complete, disconnect the terminal from MIL-STD-1553 buses, or take other measures to prevent bus activity detection by the device. With the AUTOEN pin in logic 1 state and the TXINHA and TXINHB pins in logic 0 state, apply and release $\overline{\text{MR}}$ master reset and wait for READY output assertion. Verify that the $\overline{\text{IRQ}}$ output does not pulse low (or go and remain low) at READY assertion. Confirm that the EECKE and RAMIF bits are logic 0 in the “Master Status and Reset Register (0x0001)”. If register bit 4 (RTSTEX) in “Master Configuration Register 1 (0x0000)” was set by auto-initialization, reset it now. Modify register and RAM values to reflect the new changes.

EEPROM programming is locked out at step 3 for the following conditions:

- ACTIVE output pin assertion occurs after $\overline{\text{MR}}$ master reset.
- RTSTEX bit 4 or MTENA bit 8 is set in “Master Configuration Register 1 (0x0000)”.

3. The host writes a 2-part “unlock code” to RAM address 0x0051. The unlock code value selectively enables any combination of terminal devices (MT or RT) to automatically start execution, after subsequent auto-initialization sequences are performed. Programmed here, the same combination of terminal devices is simultaneously enabled after every initialization. Unlock words are encoded as shown in Table 11.

Table 11. Terminal Unlock Word Encoding

Word 1	Word 2	Initialize RT [2]	Initialize MT [3]
0xA00A	0x5FF5	No auto init. [1]	No auto init.
0xA03A	0x5FC5	X	
0xA0CA	0x5F35		
0xA0FA	0x5F05	X	
0xA30A	0x5CF5		X
0xA33A	0x5CC5	X	X
0xA3CA	0x5C35		X
0xA3FA	0x5C05	X	X
0xAC0A	0x53F5		
0xAC3A	0x53C5	X	
0xACCA	0x5335		
0xACFA	0x5305	X	
0xAF0A	0x50F5		X
0xAF3A	0x50C5	X	X
0xAFCA	0x5035		X
0xAFFA	0x5005	X	X

Note 1: Default. No terminal devices (MT or RT) are started. The host must write “Master Configuration Register 1 (0x0000)” to start terminals.

Note 2: The RTENA register bit 6 in register 0x0000 must be set before step 4. During auto-initialization events, the AUTOEN input pin must be logic 1 before rising edge of $\overline{\text{MR}}$ master reset. After auto-initialization, RTSTEX bit 4 is automatically set in “Master Configuration Register 1 (0x0000)”, starting Remote Terminal execution.

Note 3: The MTENA register bit 8 in register 0x0000 must be set before step 4. During auto-initialization events, the AUTOEN input pin must be logic 1 before rising edge of $\overline{\text{MR}}$ master reset. After auto-initialization, the SMT MTENA bit is automatically set in “Master Configuration Register 1 (0x0000)”, starting Bus Monitor execution.

The terminals can be automatically started (or not started) in any combination. For example, exclusive-ORing both default unlock Words 1 and 2 with 0x0FF0 results in unlock Word 1 = 0xAFFA and Word 2 = 0x5005. This combination automatically and simultaneously enables execution for both terminal devices: MT and RT, at every subsequent auto-initialization from EEPROM. Individual soft resets for a single terminal device will automatically enable that device, if enabled here.

4. The EECOPY input pin is driven high for at least 1 ms, then driven low. In response, the READY output goes low while EEPROM memory is written. The unlock code at address 0x0051 is cleared. Device register and RAM contents are written to the serial EEPROM, one byte at a time.

   During programming, terminal checksums are tallied for the RT and SMT terminal devices, if used. An overall 8K checksum is also tallied. These checksums, stored in the EEPROM, are used for error detection later, during auto-initialization and soft reset events.

Checksum Type	EEPROM location corresponds to RAM Address
Overall Checksum	0x0051
RT Checksum	0x01C0
SMT Checksum	0x005C

On the following pages, see the list of registers included in the stored overall and terminal checksums. When the READY output goes high, EEPROM copy is complete.

5. For terminal devices selected for auto-enable by step 3 unlock word selection, the RTSTEX and/or MTENA bits are set in the 2-byte EEPROM image corresponding to "Master Configuration Register 1 (0x0000)". During subsequent auto-initialization events, these are the last bits written, just before READY assertion. Terminal devices having enable bits set to logic 1 in the EEPROM image are automatically and simultaneously enabled just before READY assertion. Terminal devices not automatically enabled (by step 4 unlock word selection) have logic 0 enable bits RTSTEX and/or MTENA in the 2-byte EEPROM image corresponding to "Master Configuration Register 1 (0x0000)". After auto-initialization, these terminal devices remain in standby until enabled by host write to the "Master Configuration Register 1 (0x0000)".

17.2. Overall 8K Word Checksum Used by Auto-Initialization

When auto-initialization is performed, the overall checksum (stored in EEPROM by the EECOPY process) is used for error checking. EECOPY stored this checksum value in the two 8-bit locations corresponding to RAM address 0x0051. This checksum covers the entire 8K word register and RAM address range from 0 to 0x1FFF, excluding the following register addresses. At auto-initialization, the following registers are not written using EEPROM data:

Table 12. Registers are not written using EEPROM data

Address	Excluded Register Name
0x0002	Master Status & Reset Register
0x0008	MT Pending Interrupt Register
0x0009	RT Pending Interrupt Register
0x000A	Interrupt Log Address Pointer
0x000B	RT Memory Address Pointer
0x000D	MT Memory Address Pointer
0x0018	RT Operational Status Register
0x001E	RT Built-In Test Word Register
0x0030	MT Current Address Pointer
0x0031	MT Last Address Pointer
0x003A	MT Time Tag Counter Low
0x003B	MT Time Tag Counter Mid
0x003C	MT Time Tag Counter High
0x0049	RT Time Tag Counter

The overall checksum includes individual terminal checksums for RT and Bus Monitor, which the EECOPY process stored at EEPROM locations corresponding to RAM addresses 0x01C0 and 0x005C respectively. All checksums stored by the EECOPY process use two's complement format. Each checksum is calculated by summing the individual 16-bit data values (ignoring carry) over the full set of included register and RAM addresses. The summation is then complemented, then incremented (ignoring carry) to yield the stored two's complement checksum value.

When the device performs checksum-based error checking, a new summation is tallied (ignoring carry) for the individual 16-bit data values over the range of included register and RAM addresses. When this summation is added to the previously stored two's complement checksum, the result is zero when the new data summation is the same as that tallied by EECOPY when the checksum was stored.

The registers tallied in each of the RT and SMT terminal checksums are summarized on the next pages. These individual checksums are used for error checking during soft reset events. Soft reset occurs for RT or SMT when RTRESET bit 10 and/or MTRESET bit 12 is set in the "Master Status and Reset Register (0x0001)". One or two terminal reset bits can be individually or simultaneously set.

17.3. Fast Initialization Option using EE2K Pin

The address range copied during EEPROM programming depends on the state of the EE2K input pin when rising edge occurs on the EECOPY input:

If EE2K is high when EECOPY is asserted, the lower 2K x 16 address range from 0x0 to 0x07FF is copied from device registers and RAM to EEPROM. This includes all registers, all configuration tables in RAM and the primary Descriptor Table in RAM at address 0x0400 to 0x05FF.

If EE2K is low when EECOPY is asserted, the entire 8K x 16 address range from 0x0 to 0x1FFF is copied from device registers and RAM to EEPROM. This range covers all registers, all configuration tables in RAM and the primary Descriptor Table in RAM at address 0x0400 to 0x05FF. As long as EE2K remains low when auto-initialization occurs, the 8K x 16 programming option can initialize secondary Descriptor Tables above address 0x0600, if used.

The 8K x 16 programming option (EE2K equals zero) can also initialize fixed data for any subset of the 32 possible transmit subaddress buffers, using any of the defined data buffer schemes. To enable EEPROM copy for transmit subaddress data buffers, the buffer space must be pre-loaded with the desired data. Be sure to reserve space for Message Information and Time-Tag Word locations, as required for the transmit subaddress buffer method.

18. RESET AND INITIALIZATION

This section describes the software reset mechanisms. Hardware Master Reset returns the device to the uninitialized state, requiring register and RAM initialization before terminal execution can begin. Hardware reset is initiated by assertion of the $\overline{\text{MR}}$ Master Reset pin (200ns minimum assertion time). Software reset 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)”. If AUTOEN is enabled, both hardware and software reset copy initialization values from EEPROM into registers and RAM.

18.1. Hardware Master Reset and Optional Auto-Initialization

Hardware master reset is initiated by a low to high transition on the $\overline{\text{MR}}$ pin; it should be applied after power-up, but may be used any time afterward. When asserted, the $\overline{\text{MR}}$ input pin causes immediate unconditional hardware reset for all device terminals. 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 3. The READY, ACTIVE and $\overline{\text{IRQ}}$ output pins are negated if previously asserted. The READY pin remains low until the entire reset process is complete. During this interval, a host read cycle to any address returns the value of the “Master Status and Reset Register (0x0001)”.

After a low to high transition on the $\overline{\text{MR}}$ pin, these events occur:

1. After 200ns, input states for the RTA4-RTA0, RTAP and LOCK pins are latched into the “Remote Terminal Operational Status Register (0x0018)”. The input state for the AUTOEN pin is latched into the “Master Status and Reset Register (0x0001)”.
2. At master reset, there are four pin-selected combinations for RAM self-test on/off and auto-initialize on/off. For the 4 combinations, hardware $\overline{\text{MR}}$ rising edge to READY assertion times are summarized in the following table.

Table 13. READY delay times: from $\overline{\text{MR}}$ input pin rising edge to READY output pin rising edge

Combinations	MTSTOFF Pin State	EE2K Pin State	AUTOEN Pin State	READY Delay (µs)
No RAM test, No auto-initialize	1	0	0	164
Perform RAM test, No auto-initialize	0	0	0	737
No RAM test, auto-initialize (2K)	1	1	0	3,945
Perform RAM test, auto-initialize (2K)	0	1	0	4,025
No RAM test, auto-initialize (8K)	1	0	1	15,775
Perform RAM test, auto-initialize (8K)	0	0	1	16,100

   If memory error occurs, the BMTF bits are set in the “Remote Terminal Built-In Test (BIT) Word Register (0x001E)”. If the MTSTOFF pin is logic 1, the RAM test is bypassed. This option might be chosen if a faster reset process is needed. Regardless of the MTSTOFF pin state, all RAM locations above address 0x004F are reset to 0x0000.

3. After internal processes are initialized, the device checks the state of the AUTOEN bit latched into the “Master Status and Reset Register (0x0001)” at step 1:

If the AUTOEN bit in the “Master Status and Reset Register (0x0001)” is logic 0, auto initialization from EEPROM is bypassed. After the RAM memory test is complete, the device asserts the READY output pin to indicate that the device is ready for host initialization of registers and RAM:

- The “Master Configuration Register 1 (0x0000)” is initialized to indicate which terminal device is enabled (RTENA). The corresponding input pin must already be logic 1, or this register bit cannot be set. If using Bus Monitor, the MTRUN input pin should be logic 1 but the corresponding MTENA register bit should remain logic 0 until initialization is completed. Likewise, the RTSTEX bit should remain low at this time.
- Other configuration registers are initialized by the host to define interrupt behavior and time tag counter behavior for enabled terminal devices.

- The following registers are initialized: the RT Configuration Register, the RT Descriptor Table Base Address Register, the RT Bus A/B Select Register and the RT Interrupt Enable Register. In RAM, the RT Illegalization Table and RT Descriptor Table are initialized. Initial data for assigned Transmit Subaddress Data Buffers may be initialized at this time.
- If using a Simple Bus Monitor, the MT Configuration, SMT Address List Start Address and SMT Interrupt Enable registers are initialized. In RAM, the SMT Address List and SMT Filter Table are initialized.
- Upon completing all initialization for registers and RAM, the host writes the "Master Configuration Register 1 (0x0000)" to start operation for the enabled terminal devices. A combination of RTSTEX and/or MTENA register bits are asserted to match the subset of initialized terminal devices (RT and SMT). Device operation begins.

If the AUTOEN bit in the "Master Status and Reset Register (0x0001)" is logic 1, auto initialization from EEPROM is performed after completion of the RAM memory test. The READY output pin remains at logic 0 during the self-initialization process. Initialization data is read from the previously-written external EEPROM and copied to the entire range of registers and RAM, from address 0x0000 to address 0x1FFF. This process typically requires 63 ms (see Table 13).

During auto initialization, the written value for each register or RAM location is read back for confirmation. If the value read fails to match the corresponding value in EEPROM, an initialization error is saved. This error results in action taken later when the initialization process is finished.

While performing initialization a running checksum is tallied. A properly-configured serial EEPROM contains a 16-bit checksum value stored at the EEPROM byte pair locations corresponding to RAM address 0x0051. The stored checksum is tallied as if RAM address 0x0051 equals zero, and twenty-five register locations listed in Table 12 are also excluded from the stored checksum value. The stored value is actually the twos complement of the 16-bit memory checksum, ($\overline{\text{CHECKSUM}}$ + 1). As each individual register and RAM location is initialized, its written value is added to a copy of the stored checksum value from EEPROM. If all locations match at the end, the running checksum tally added to the twos complemented EEPROM checksum should equal zero.

After initialization, when READY is asserted, the 16-bit twos complement checksum value is copied from EEPROM to device RAM address 0x0051.

If an initialization error occurred, the following events take place immediately after READY assertion:

- the $\overline{\text{IRQ}}$ interrupt output pin is asserted.
- The "Master Status and Reset Register (0x0001)" is written to indicate type of error. If checksum failure, the EECKF register bit is asserted. If data mismatch between EEPROM and read back RAM value, the RAMIF register bit is asserted.
- The EELF bit is asserted in the "Remote Terminal Built-In Test (BIT) Word Register (0x001E)".
- If RAMIF read back error occurred, the address of the first occurring instance is written to register address 0x0024. See Section "18.3. Memory Test Fail Address Register (0x0024)" on page 150 for further information. Additional locations beyond the saved address may have mismatch, but only the first instance is logged.

After copying the full range of register and RAM addresses, the RTSTEX and MTENA bits in the "Master Configuration Register 1 (0x0000)" are still zero.

The EEPROM is written using methods described in Section 17.1. Each of the terminal devices can independently be configured to self-start when error-free auto-initialization is complete, or not self-start, requiring a host write to the "Master Configuration Register 1 (0x0000)" after READY assertion. In the EEPROM byte pair corresponding to "Master Configuration Register 1 (0x0000)":

- **If the EEPROM RTSTEX and RTENA bits are both logic 1,** and if the RTAPF bit is logic 0 in the "Remote Terminal Operational Status Register (0x0018)", the Remote Terminal will automatically start just before READY assertion. Once RT is started, the "Master Configuration Register 1 (0x0000)" RTSTEX bit will read logic 1. **If the EEPROM RTSTEX bit is logic 0,** the host must write the "Master Configuration Register 1 (0x0000)" RTSTEX bit high after READY assertion to start the RT. The RTENA register bit must be logic 1 before RTSTEX assertion.

- **If the EEPROM MTENA bit is logic 1,** the Bus Monitor will automatically start just before READY assertion. Once the Bus Monitor is started, the “Master Configuration Register 1 (0x0000)” MTENA will read logic 1. **If the EEPROM MTENA bit is logic 0,** the host must write the “Master Configuration Register 1 (0x0000)” MTENA bit high after READY assertion to start the Bus Monitor.

Note that automatic-self-start for RT requires the RTAPF status bit to be logic 0 in the “Remote Terminal Operational Status Register (0x0018)”. **This indicates valid odd parity for the terminal address and parity bits latched in the RT Operational Status Register, not necessarily the state of the RT address and parity pins.** Because auto-initialization follows master reset, the mirrored pin states latched at reset is overwritten by EEPROM values if LOCK input pin is logic 0.

- **When the LOCK input pin is logic 1 at Master Reset rising edge,** the “Remote Terminal Operational Status Register (0x0018)” terminal address and parity bits reflect address input pin states 200ns after reset rising edge.
- **When the LOCK input pin is logic 0 at Master Reset rising edge,** the latched RT address, parity and LOCK bit values are overwritten by values from the initialization EEPROM.

If automatic-self-start for RT was blocked due to invalid odd parity for the terminal address and parity bits latched in the “Remote Terminal Operational Status Register (0x0018)”, the RTSTEX bit cannot be asserted in the “Master Configuration Register 1 (0x0000)” until the parity error is corrected. The host may overwrite the “Remote Terminal Operational Status Register (0x0018)” to correct the parity error, then assert RTSTEX in the “Master Configuration Register 1 (0x0000)”.

If automatic-self-start for terminal devices was blocked due to RAMIF or EECKF auto-initialization errors, the host can override the error condition after READY assertion by setting the RTSTEX and/or MTENA bits in the “Master Configuration Register 1 (0x0000)” , providing the other operational conditions (in the 4-bullet list above) are met.

A method for programming the initialization EEPROM from a fully configured terminal is explained in Section 17. If a different method is used for writing the serial EEPROM, for successful self-initialization after Master Reset, the twos-complemented checksum (described earlier) must be saved in EEPROM locations corresponding to device RAM address 0x0051.

If a different method is used for writing the serial EEPROM, in order to perform soft resets, twos-complement checksum must be written for the Remote Terminal at 0x01C0 and the Monitor Terminal at 0x005C.

A compatible serial EEPROM uses a SPI interface for byte-access read and write operations. Sixteen-bit register and RAM values are stored as upper and lower bytes in the EEPROM, in “big endian” fashion. For example, the upper byte for register address 0x0000 is stored at EEPROM address 0x0000 while the lower byte is stored at EEPROM address 0x0001. A 16K x 8 EEPROM is required to store the entire 8K x 16 address range.

Serial EEPROM data mapping follows the device memory map shown in Figure 2. The three exceptions:

1. the two EEPROM bytes corresponding to device RAM address 0x0051 must contain the expected overall checksum value and if software resets are expected
2. the two EEPROM bytes corresponding to device RAM address 0x01C0 must contain the expected terminal checksum value for Remote Terminal 1
3. the two EEPROM bytes corresponding to device RAM address 0x005C must contain the expected terminal checksum value for the Bus Monitor.

The serial EEPROM used for auto-initialization should be fully written to cover the upper address limit of 0x1FFF. Ideally the EEPROM image reflects a post-MR reset followed by fresh initialization by the host, with nothing written to reset-cleared registers or RAM as a result of command processing.

18.2. Initialization using EE2K Pin

The address range copied during EEPROM programming depends on the state of the EE2K input pin when rising edge occurs on the EECOPY input: A smaller memory space may be initialized faster by asserting the EE2K pin. See “Fast Initialization Option using EE2K Pin”.

18.3. Memory Test Fail Address Register (0x0024)

MSB	Memory Test Fail Address Register														LSB	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$\overline{MR}$ Reset
RW																Host Access
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit

If the AUTOEN input pin is logic 1, auto-initialization from EEPROM is enabled. When one or more initialized RAM locations do not match their two corresponding serial EEPROM byte locations, RAMIF bit 13 is set in the “Hardware Pending Interrupt Register (0x0006)”, as well as RAMIF bit 0 in the “Master Status and Reset Register (0x0001)”. Such failure may occur during auto initialization, or execution of a partial reset caused by assertion of the RTRESET, or MTRESET bits in the “Master Status and Reset Register (0x0001)”. The address of the first occurring RAM/EEPROM mismatch is written to the Memory Test Fail Address Register, 0x0024. Additional locations beyond the saved address may have mismatch, but just the first instance is logged and the test stops.

Once terminal operation starts, register 0x0024 has no function.

18.4. Software Reset

18.4.1. 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 14. 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
	0x001A RT MIL-STD-1553 Status Word Bits Register
	0x001E RT Built-In Test Word Register
	0x0049 RT Time Tag Counter
Loads these registers from EEPROM *These 522 locations comprise the RT Terminal Checksum stored at 0x01C0 by EECOPY process*	0x000F Hardware Interrupt Enable Register
	0x0012 RT Interrupt Enable Register **(see note)**
	0x0013 Hardware Interrupt Output Enable Register
	0x0016 RT Interrupt Output Enable Register **(see note)**
	0x0017 RT Configuration Register
	0x0019 RT Descriptor Table Base Address Register
	0x001C RT Bus A Select Register
	0x001D RT Bus B Select Register
	0x001F RT Alternate BIT Word Register
	0x004A RT Time Tag Utility Register
	0x0200 through 0x02FF: RT Illegalization Table
	0x0400 through 0x05FF: RT Descriptor Table

RT automatically starts (RTSTEX is set in register 0x0000) after soft reset completion, if these requirements are met:

- The RTENA bit 6 is logic 1 in the "Master Configuration Register 1 (0x0000)" when EECOPY created the EEPROM image.
- The RTSTEX bit 4 is logic 1 in the EEPROM "Master Configuration Register 1 (0x0000)" image because the EECOPY unlock codes during programming were Unlock Word 1 = 1010-XXXX-XX11-1010 and Unlock Word 2 = 0101-XXXX-XX00-0101 where X denotes "don't care".

To manually start RT after soft reset completion (indicated by READY signal assertion), the host must set RTSTEX bit 4 in register 0x0000, if RT auto-start was disabled or otherwise failed for one or more of these reasons:

- The RTENA bit 6 was logic 0 in the "Master Configuration Register 1 (0x0000)" when EECOPY created the EEPROM image. A new EEPROM image is needed to allow auto-start after RT soft reset.
- The RTSTEX bit 4 is logic 0 in the EEPROM "Master Configuration Register 1 (0x0000)" image because the unlock codes used by EECOPY were wrong. A new EEPROM image is needed to allow auto-start after RT soft reset.

18.4.2. 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 15. 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
Loads these registers from EEPROM *These 561 locations comprise the MT Terminal Checksum stored at 0x005C by EECOPY process*	0x000F Hardware Interrupt Enable Register
	0x0011 SMT Interrupt Enable Register
	0x0013 Hardware Interrupt Output Enable Register
	0x0015 SMT Interrupt Output Enable Register
	0x0029 SMT Configuration Register
	0x002F SMT Buffer Address Table Start Address Register
	0x003D SMT Time Tag Utility Register, Low
	0x003E SMT Time Tag Utility Register, Mid
	0x003F SMT Time Tag Utility Register, High
	0x0040 SMT Time Tag Match Register, Low
	0x0041 SMT Time Tag Match Register, Mid
	0x0042 SMT Time Tag Match Register, High
	0x00B0 through 0x00BF: SMT Buffer Address Table(s)
	0x0100 through 0x017F: SMT Message Filter Table

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).

The Bus Monitor automatically starts (MTENA is set in register 0x0000) after soft reset completion, if the following requirements are met. Message recording commences when a new valid command is received:

- The MTENA bit 8 was logic 1 in the “Master Configuration Register 1 (0x0000)” when EECOPY created the EEPROM image.
- The MTENA bit 8 is logic 1 in the EEPROM “Master Configuration Register 1 (0x0000)” image because EECOPY used these unlock codes during EEPROM programming: Unlock Word 1 = 1010-XX11-XXXX-1010 and Unlock Word 2 = 0101-XX00-XXXX-0101 where X denotes “don’t care”.

To manually start the Bus Monitor after soft reset completion (indicated by READY signal assertion), the host must set MTENA bit 8 in register 0x0000, if auto-start was disabled or otherwise failed for one or more of these reasons:

- The MTENA bit 8 was logic 0 in the “Master Configuration Register 1 (0x0000)” when EECOPY created the EEPROM image. A new EEPROM image is needed to allow auto-start after MT soft reset.
- The MTENA bit 8 is logic 0 in the EEPROM “Master Configuration Register 1 (0x0000)” image because the unlock codes used by EECOPY were wrong. A new EEPROM image is needed to allow auto-start after Bus Monitor soft reset.

18.5. 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 13.6 on page 78). 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 pin MODE1760 is asserted during a hardware reset. The pin status will be latched 200ns after the rising edge of $\overline{\text{MR}}$ Master Reset (the same time as the RT address). During 1760 mode, the device 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” either by the host or by EEPROM auto initialization (see “Hardware Master Reset and Optional Auto-Initialization”).

19. SELF-TEST

The HI-6136 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 for RT terminal mode.

19.1. Optional RAM Self-Test after Hardware Master Reset

When the MTSTOFF input pin is logic 0, the device automatically performs RAM self-test after each hardware master reset, following the rising edge of $\overline{\text{MR}}$ input signal. See Section “18.1. Hardware Master Reset and Optional Auto-Initialization”. The READY output pin goes low at $\overline{\text{MR}}$ assertion. READY remains low after $\overline{\text{MR}}$ rising edge and during RAM self-test. The RAM self-test performed is the increment/decrement (Inc/Dec) method described on page 159. When successful RAM self-test is complete, the READY output pin goes high, indicating that device registers and RAM can be configured for operation. The entire RAM address space from 0x0052 to 0x1FFF is cleared to 0x0000. RAM self-test after hardware master reset is optional. If the MTSTOFF input pin is logic 1, RAM testing is skipped, speeding up READY assertion. Table 13 on page 147 shows the reset timing options.

19.2. Host-Directed Self-Test

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

19.2.1. 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	$\overline{\text{MR}}$ 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 device TEST input pin. When the TEST pin is logic 0 (normal operating mode), register address 0x0028 has no function.

When the TEST input pin 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 pin should be reset to logic 0, restoring all register bits to Read-Write.

Descriptions below apply when the TEST input pin 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 7 - 2 are used for transceiver loopback testing (either digital loopback or analog loopback).

Under internal logic control, this device uses one RAM self test (Inc / Dec Test described below) to check internal RAM memory after every $\overline{\text{MR}}$ pin master reset, unless the MTSTOFF input pin is logic 1. This option may be used to speed up reset completion. 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 ($\overline{MR}$) <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="3"><i>Function</i></th></tr>
<tr><td>15</td><td>FRAMA</td><td>R/W</td><td>0</td><td colspan="3">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="3">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="10">13,12,11</td><td rowspan="10">RBSEL2:0</td><td rowspan="10">R/W</td><td rowspan="10">0</td><td colspan="3">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>RBSEL2:0</td><td>Selected RAM Test</td><td>Test Time</td></tr>
<tr><td>000</td><td>Idle</td><td>-</td></tr>
<tr><td>001</td><td>Pattern Test, described below</td><td>14.42ms</td></tr>
<tr><td>010</td><td>Write 0x0000 to RAM address range 0x0000 - 0x1FFF</td><td>170µs</td></tr>
<tr><td>011</td><td>Read and verify 0x0000 over RAM address range 0x0000 - 0x1FFF</td><td>500µs</td></tr>
<tr><td>100</td><td>Write 0xFFFF to RAM address range 0x0000 - 0x1FFF</td><td>170µs</td></tr>
<tr><td>101</td><td>Read and verify 0xFFFF over RAM address range 0x0000 - 0x1FFF</td><td>500µs</td></tr>
<tr><td>110</td><td>Inc / Dec Test performs only steps 5 - 8 of the Pattern Test below</td><td>1.32ms</td></tr>
<tr><td>111</td><td>Idle</td><td>-</td></tr>
</table>

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
13,12,11	RBSEL2:0 (continued)	R/W	0	**Description of the RAM BIST "PATTERN" test selected when register bits RBSEL2:0 = 001:** *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 0x1FFF. 2. Repeat the following sequence for each RAM location from 0x00000 through 0x1FFF: 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 0x1FFF 4. Repeat the following sequence for each RAM location from 0x00000 through 0x1FFF: 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 0x1FFF. 6. Read each memory location from 0x00000 to 0x1FFF 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.
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 pin 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. The user must reload MAP register with 0x28 before each register read that polls result bits 9:8.

Bit No.	*Mnemonic*	*R/W*	*Reset*	*Function*
9	RBPASS	R	0	RAM BIST Pass. Device 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. Device 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, a clue to the failing RAM address can be read at register address 0x001B. For speed, the RAM BIST concurrently tests 4 quadrants of the RAM address range, in parallel. If test failure occurs, register address 0x001B contains the RAM address being tested in the lowest RAM quadrant. Actual failure has occurred in any of these four locations: at RAM address “ADDR” stored in register 0x001B, or ADDR+0x2000, or ADDR+0x4000 or ADDR+0x6000. When the TEST input pin 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 device 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 pin 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. Device 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. Device 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.

19.2.2. 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	$\overline{MR}$ 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 $\overline{MR}$ pin master reset, but is not affected by SRST software reset. The function of this register is multiplexed by the device TEST input pin. 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 pin 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 "19.2.1. Self-Test Control Register (0x0028)" on page 154 (bits 0-5) for additional information.

After test completion, the TEST input pin should be reset to logic 0. The host should restore the desired alternate BIT Word value for the RT at register address 0x001F.

19.2.3. 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	$\overline{MR}$ 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 $\overline{MR}$ pin master reset, but is not affected by SRST software reset. The function of this register is multiplexed by the device TEST input pin. When the TEST input pin is logic 0 (normal operating mode), register address 0x0002 is the "Remote Terminal Current Command Register (0x0002)".

When the TEST input pin 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 "19.2.1. Self-Test Control Register (0x0028)" on page 154 (bits 0-5) for additional information.

After test completion, the TEST input pin 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.

19.2.4. 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	$\overline{MR}$ 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 device TEST input pin. When the TEST input pin is logic 0 (normal operating mode), this register address is the "Remote Terminal Current Message Information Word Register (0x001B)". When the TEST input pin 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 "19.2.1. Self-Test Control Register (0x0028)" on page 154) 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 BIST Fail Address Register. (This is actually an offset value within a memory quadrant; see the Self-Test Control Register bit 8 description.) 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 pin to logic 0.

19.2.5. 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 device is operational, terminal execution should be stopped. If asserted, reset bits 4-8 and bit 12 in "9.1. Master Configuration Register 1 (0x0000)", stopping MT, and RT. Assert the TEST pin to activate register 0x0028 as the Self-Test Control Register.

After asserting the TEST input pin, RAM self-test is configured and started by writing bits 15:10 in the Self-Test Control Register, described on page 154. 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 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)". The written value is actually an offset value within a memory quadrant; see "Self-Test Control Register (0x0028)", bit 8. 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 pin to logic 0, then re-initialize registers and RAM, and finally restart terminal execution.

19.2.6. 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-6136 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 "9.1. 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 pin to logic 1.

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 pin to logic 0.

20. HOST SERIAL PERIPHERAL INTERFACE (SPI)

Internal RAM and registers occupy an 8K x 17 address space. The lowest 80 addresses access registers and the remaining addresses access RAM locations. Timing is identical for register operations and RAM operations via the serial interface, and read and write operations have likewise identical timing.

20.1. Serial Peripheral Interface (SPI) Basics

The device uses an SPI synchronous serial interface for host access to registers and RAM. Host serial communication is enabled through the Chip Enable (CE) pin, and is accessed via a three-wire interface consisting of Serial Data Input (SI) from the host, Serial Data Output (SO) to the host and Serial Clock (SCK). All programming cycles are completely self-timed, and no erase cycle is required before write.

The SPI (Serial Peripheral Interface) protocol specifies master and slave operation; the HI-6136 operates as an SPI slave.

The SPI protocol defines two parameters, CPOL (clock polarity) and CPHA (clock phase). The possible CPOL-CPHA combinations define four possible “SPI Modes.” Without describing details of the SPI modes, the device operates in the two modes where input data for each device (master and slave) is clocked on the rising edge of SCK, and output data for each device changes on the falling edge. These are known as SPI Mode 0 (CPHA = 0, CPOL = 0) and SPI Mode 3 (CPHA = 1, CPOL = 1). Be sure to set the host SPI logic for one of these modes.

The difference between SPI Modes 0 and 3 is the idle state for the SCK signal, which is logic 0 for Mode 0 state and logic 1 for Mode 3 state (see Figure 20). There is no configuration setting to select SPI Mode 0 or Mode 3 because compatibility is automatic. Beyond this point, the data sheet only shows the SPI Mode 0 SCK signal in timing diagrams.

The SPI protocol transfers serial data as 8-bit bytes. Once CE chip enable is asserted, the next 8 rising edges on SCK latch input data into the master and slave devices, starting with each byte’s most-significant bit. The SPI can be clocked at 40 MHz.

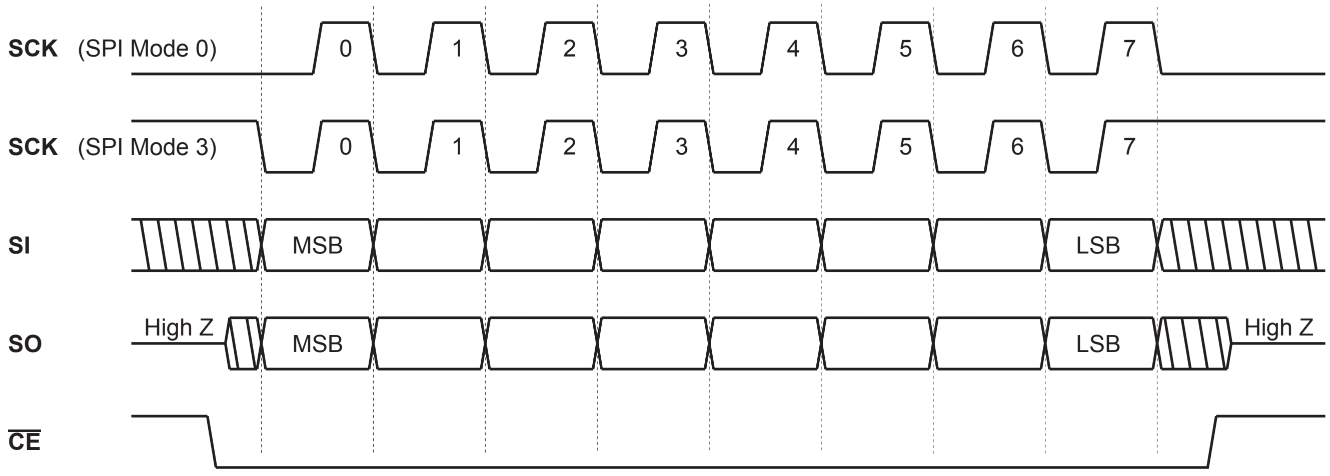


Figure 20. Generalized Single-Byte Transfer Using SPI Protocol. SCK is Shown for SPI Modes 0 and 3

Multiple bytes may be transferred when the host holds CE low after the first byte transferred, and continues to clock SCK in multiples of 8 clocks. A rising edge on CE chip enable terminates the serial transfer and reinitializes the SPI for the next transfer. If CE goes high before a full byte is clocked by SCK, the incomplete byte clocked into the device SI pin is discarded.

Two byte transfers are needed for SPI exchange of 16- bit register values or RAM data. “Big endian” byte order is used for SPI data transfers. The high order byte (bits 15:8) is transferred before the low order byte (bits 7:0).

In the general case, both master and slave simultaneously send and receive serial data (full duplex) per Figure 20. However the device operates half duplex, maintaining high impedance on the SO output, except when actually transmitting serial data. When sending data on SO during read operations, activity on its SI input is ignored. Figure 21 and Figure 22 show actual behavior for the SO output.

20.2. SPI Commands

Each SPI read or write operation begins with an 8-bit command byte transferred from the host to the device after assertion of CE. Since command byte reception is half-duplex, the host discards the dummy byte it receives while serially transmitting the command byte.

The SPI command set uses the most significant command bit to specify whether the command is Read or Write. The command byte MSB is zero for read commands, and one for write commands.

20.3. Fast-Access Commands for Registers 0-15

The SPI command set includes directly-addressed read commands for registers 0 through 15. The 8-bit pattern for these read commands has the general form

0-0-R-R-R-R-0-0

where RRRR is the 4-bit register address. These fast-access read commands appear in Table 16.

20.4. Fast-Access Write Commands for Registers 0-63

The SPI command set includes directly-addressed write commands for registers 0 through 63. The 8-bit pattern for these read commands has the general form

1-0-R-R-R-R-R-R

where RRRRRR is the 6-bit register address. The fast-access write commands appear in Table 16.

Figure 21 and Figure 22 show read and write timing as it appears for fast-access register operations. The command byte is immediately followed by two data bytes comprising the 16-bit data word read or written. For a register read or write, $\overline{\text{CE}}$ is negated after the 2-byte data word is transferred.

20.5. Indirect Addressing of RAM and Registers

Refer to the SPI command set shown in Table 17. All SPI commands other than fast-access use a Memory Address Pointer register to indicate the starting address for read or write transactions. Four “Memory Address Pointers” reside at register addresses 0x000B through 0x000E. Just one Memory Address Pointer (MAP) is active (enabled) at any time.

The active Memory Address Pointer is selected by writing the MAPSEL bits 11-10 in the “Master Configuration Register 1 (0x0000)”. Or use the SPI instruction op codes 0xD8, 0xD9, 0xDA or 0xDB which enable MAP registers 0x000B, 0x000C, 0x000D or 0x000E respectively, by automatically writing MAPSEL bits 11-10 in the “Master Configuration Register 1 (0x0000)”.

The active Memory Address Pointer must be initialized before any read or write operation, other than fast-access.

To write the active MAP register, use a fast-access write op code, followed by the desired 16-bit memory address:

- Writing MAP register 0x000B uses SPI op code 0x8B followed by 16-bit address.
- Writing MAP register 0x000C uses SPI op code 0x8C followed by 16-bit address
- Writing MAP register 0x000D uses SPI op code 0x8D followed by 16-bit address.
- Writing MAP register 0x000E uses SPI op code 0x8E followed by 16-bit address.

To read the active MAP register, use a fast-access write op code. The current MAP 16-bit value is clocked out in the next 16 sequential SCK clock cycles:

- Reading MAP register 0x000B uses SPI op code 0x2C
- Reading MAP register 0x000C uses SPI op code 0x30
- Reading MAP register 0x000D uses SPI op code 0x34
- Reading MAP register 0x000E uses SPI op code 0x38

While SPI command op codes are always 8 bits, transacted addresses and register or memory data are always 16-bit words, transferred by the SPI as two sequential bytes. After a 2-byte read/write completion, the active Memory Address Pointer automatically increments to the following register address. The host can extend the read or write operation to the next register address by continuing to hold CE low while clocking SCK 16 additional times. This auto-increment feature can be used to access one or more sequential register addresses above the command address. Auto-increment applies (ranging to the top of the address space) as long as SCK continues to be clocked under continuous CE assertion. Caution: When the primary address pointer is used for auto-incrementing multi-word read/write and reaches the top of the address range (0x1FFF) the next increment rolls over the MAP value to 0x0000. The host should avoid this situation.

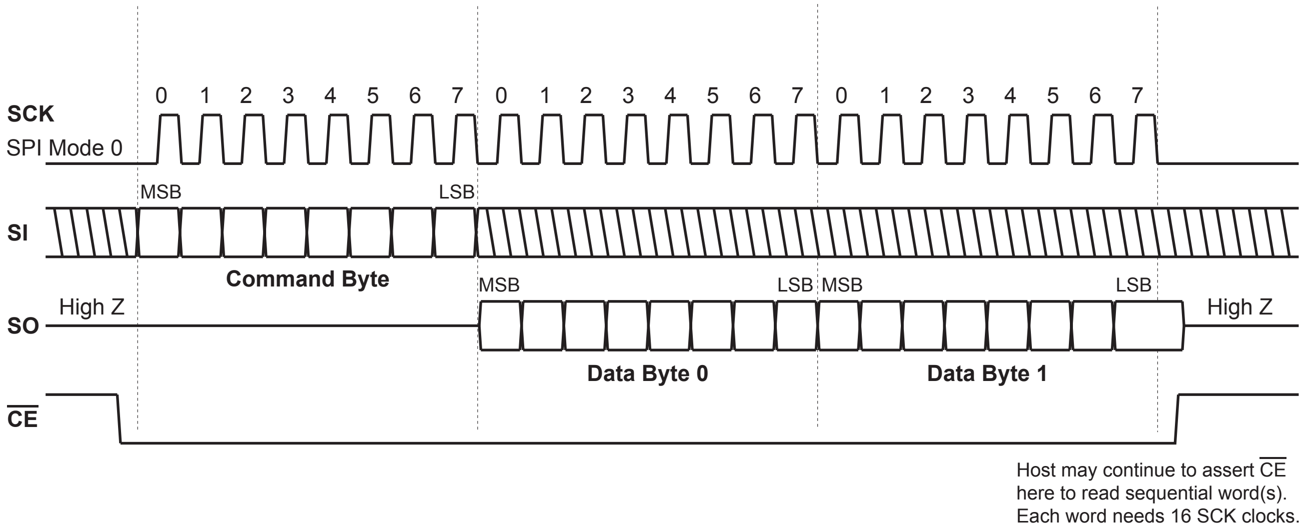


Figure 21. Single-Word (2-Byte) Read From RAM or a Register

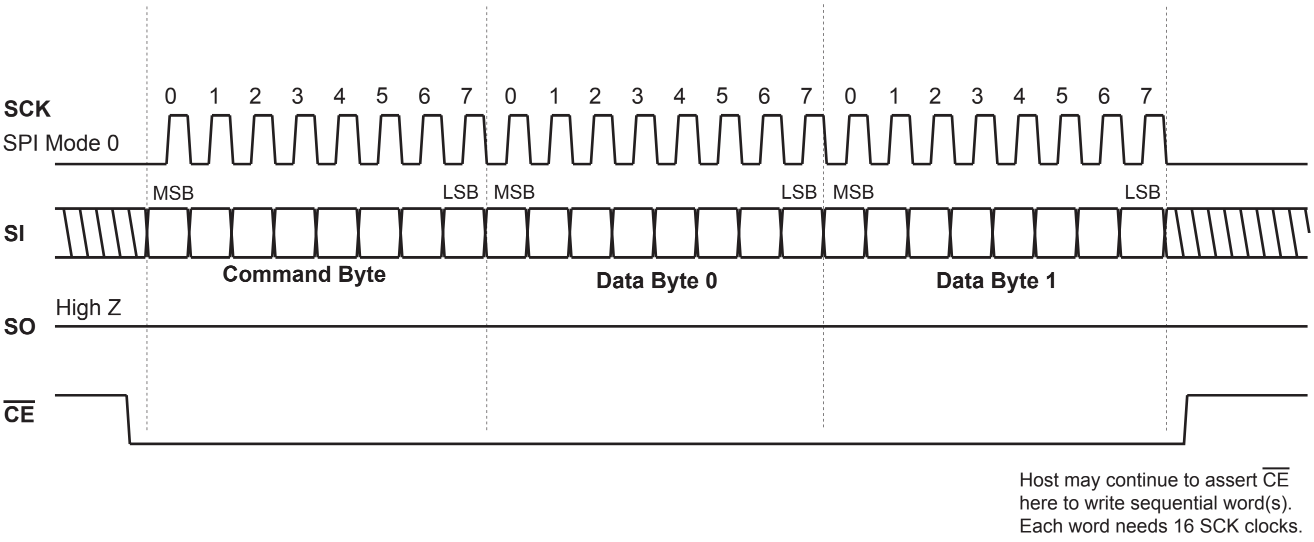


Figure 22. Single-Word (2-Byte) Write To RAM or a Register

Three single-byte SPI commands modify the value in the active Memory Address Pointer, selected by MAPSEL bits 11-10 in the “Master Configuration Register 1 (0x0000)”:

Command	Address Pointer Operation
0xD0	increment enabled Memory Address Pointer value
0xD2	add 2 to enabled Memory Address Pointer value
0xD4	add 4 to enabled Memory Address Pointer value

The “Add 4” command may be useful when sequentially accessing the same word (for example, the Control Word) in a series of 4-word Descriptor Table entries. The “Add 2” command might be useful for reading the Interrupt Log Buffer, comprised of 2-word log entries. In both cases, the Add command would be probably followed by Read command 0x40 to read the location addressed by the current pointer value. Similarly, Write command 0xC0 writes the location addressed by the current pointer value. Two command bytes cannot be “chained”; the host SPI Slave Select CE must be negated after the Add command, then reasserted for the following Read or Write command.

The active Memory Address Pointer is not affected by fast-access read/writes to the low register addresses because fast-access SPI commands use a separate, internal pointer not directly accessible to the host.

Two single-byte SPI commands use the current value of the enabled Memory Address Pointer without first loading or otherwise modifying it:

Command	Read Operation
0x40	read location addressed by enabled Memory Address Pointer

Command	Write Operation
0xC0	write location addressed by enabled Memory Address Pointer

Either of these commands can be used to read or write a single location, or may be used to start a multi-word read or write that uses the MAP pointer’s auto-increment feature.

One single-byte SPI command increment the current value of the enabled Memory Address Pointer, then performs a write:

Command	Write Operation
0xC8	add 1 to enabled Memory Address Pointer then write addressed location

20.6. Data Prefetch for SPI Read Cycles

Data prefetch is a technique used to speed up host multi-word read access to registers or RAM. Prefetching occurs when logic accesses data before it is actually needed. Because register or RAM locations are often read sequentially, performance improves when data is prefetched in address sequence order. For any SPI read cycle, the device first fetches the addressed location, then increments the memory address pointer and prefetches the following address, to speed up access in the likely event that the following word will be read next. The read cycle prefetch allows the SPI host to read sequential locations back-to-back, continuing as long as the host asserts chip select and provides SPI clock. This is described as the Memory Address Pointer “auto-increment” feature.

There is an exception: read cycle prefetch is blocked when the next RAM address is a Control Word in the RT Descriptor Table. If allowed, pre-fetch (like any other read) resets the Control Word DBAC status bit. To preserve DBAC status bit function, prefetch is disabled when reading Control Words within Descriptor Table address range. The table base address (set by the value in “Remote Terminal Descriptor Table Base Address Register (0x0019)”) and every fourth word thereafter is a Control Word. This consists of table addresses having these offsets from the table start address: 0, 4, 8, 0xC through and including 0x1F8 and 0x1FC. See further information in Section 20.8.

These two commands can be used to read or write a single location, or may be used to start a multi-word read or write that uses the pointer’s auto-increment feature.

20.7. Special Purpose Commands

Several other SPI commands load or otherwise modify the active Memory Address Pointer before initiating a read or write process. These commands were tailored to the specific needs of Remote Terminal host software.

Using a single-byte SPI command, the active Memory Address Pointer can be directly loaded with the memory address for the RT descriptor table Control Word corresponding to the last completed MIL-STD-1553 command. The Control Word is then read.

Command	Read Operation
0x48	Copy RT Current Control Word Address register 3 into the enabled Memory Address Pointer. Read the location addressed by the new pointer value.

This command can be used to read just the current RT Control Word, or may be used to start a multi-word read because memory pointer auto-increment occurs after the Control Word is read.

Six single-byte SPI commands add an offset to the current address pointer value, then read the addressed memory location; the read value is then written to the address pointer register 15. The new pointer value is used to start a read or write operation:

Command	Read Operation
0x68	Read the location addressed by the enabled Memory Address Pointer. Write the value just read into the Memory Address Pointer. Then read.
0x70	Add 1 to the enabled Memory Address Pointer. Read value at newly addressed location and write it into the Memory Address Pointer. Then read.
0x78	Add 2 to the enabled Memory Address Pointer. Read value at newly addressed location and write it into the Memory Address Pointer. Then read.

Command	Write Operation
0xE8	Read the location addressed by the enabled Memory Address Pointer. Write the value just read into the Memory Address Pointer. Then write.
0xF0	Add 1 to the enabled Memory Address Pointer. Read value at newly addressed location and write it into the Memory Address Pointer. Then write.
0xF8	Add 2 to the enabled Memory Address Pointer. Read value at newly addressed location and write it into the Memory Address Pointer. Then write.

Primary use occurs when the RT Descriptor Table Control Word was just read. For example, the last op code performed was 0x48, reading the RT Control Word for the last command. After reading the Control Word, the enabled Memory Address Pointer automatically incremented. The host can examine flag bits contained in the just-read Control Word to determine the applicable data buffer (e.g., Data Buffer A, Data Buffer B or the Broadcast Data Buffer) then directly service that buffer using these op codes; the three data buffer pointers occur in the three words following the initially read Control Word.

These six commands can be used to read or write a single location, or may be used to start a multi-word read or write that uses the pointer's auto-increment feature.

When some or all subaddress or mode commands are not programmed to trigger host interrupts, a different single-byte SPI command may be useful if polling the RT Descriptor Table for message activity. In this situation, the host may poll a series of Descriptor Table Control Words looking for instances where the DBAC activity bit is set. The DBAC (Descriptor Block Accessed) flag is set in the Control Word each time the corresponding command is completed. The process of reading the Control Word automatically resets the register's DBAC bit so the host can detect activity the next time the DBAC flag is set by the device.

Since RT Descriptor Table Control Words are spaced four words apart, this command is useful when polling a series of descriptor table Control Words:

Command	Read Operation
0x60	read addressed location then add 4 to pointer

Primary use occurs when the address pointer initially points to the first Descriptor Table Control Word in a series of Control Words to be polled (every fourth word).

After 8 SCK clocks for the SPI command, each instance of this command reads a single location using 16 SCK clocks. If CS remains low after 24 clocks and SCK continues, a multi-word read begins, using the address pointer's auto-increment feature. The second word read is at (Control Word address + 4), the next Control Word in the table.

Another single-byte SPI command is useful when servicing interrupts. When enabled interrupts occur, two words are written to the circular 64-word Interrupt Log Buffer, and the Interrupt Log Address register 0x000A is updated to show the storage address where interrupt information words will be stored for the next occurring interrupt. Buffer starting address is 0x0180 and ending address is 0x01BF. Because two words are written to the buffer for each interrupt, the Interrupt Log Address register always contains an even value in the range of 0x0180 to 0x01BE.

When servicing an interrupt that just occurred, the host wants timely information on that interrupt. An SPI command is provided to simplify interrupt handling:

Command	Read Operation
0x58	Write enabled Memory Address Pointer with the current bit 8:0 address value in Interrupt Log Address register minus 1. (See note.) If address bits 8:0 equal 0x0180, then 0x1BF is written into Memory Address Pointer. Then read the newly-addressed RAM location, containing the last-written Interrupt Address Word. Then decrement the Memory Address Pointer, which then addresses the corresponding Interrupt Information Word

Note: Bits 15:9 in the Interrupt Log Address register contain the interrupt count since the Log Address register was last read. These bits are not written to the Memory Address Pointer.

This command can be used to read a single location (the last-written Interrupt Address Word), or may be used to start a multi-word read in which the Memory Address Pointer automatically decrements after each word is read, reading words stored in the Interrupt Log Buffer, in Last In First Out order. This is the only SPI op code that decrements the Memory Address Pointer for multi-word operations. Repeated memory pointer decrements will wrap around the 0x0180 to 0x01BF Interrupt Log Buffer boundary.

20.8. RT Descriptor Table Prefetch Exceptions

For the SPI-interface, the enabled Memory Address Pointer (register 0x000F) contains the address for each new word read by the host. When starting a read access, the host usually writes the enabled Memory Address Pointer with the address for the first word to read. The host then uses an SPI op code to initiate the read process. After the addressed word is transferred by SPI to the host, the device continues to read and transmit words from sequential RAM memory addresses, as long as the host continuously asserts chip select while providing SCK serial clock pulses. After fetching each new word, the device increments the Memory Address Pointer and prefetches the data contained in the newly addressed location. The next word is prefetched even when the host does not ultimately read the following address. Sequential reads from the RT Descriptor Table that rely on MAP auto-increment will stop advancing when the next address contains a RT Descriptor Table Control Word. Properly designed SPI transfers overcome this behavior.

Using SPI command op codes, the host must consider prefetch and pointer behavior when reading data from the Descriptor Table. Applied outside the RT Descriptor Table, the following SPI sequence would read data from six successive memory addresses. But below, applied within the table, the sequence gets stuck at the fourth word read.

Below we assume the RT Descriptor Table starts at default base address, 0x0400. The host first uses SPI op codes 0xD8 and 0x8B to enable Memory Address Pointer 0x000B then write the table start address 0x0400 into it. The sequence then uses op code 0x40 (and MAP auto-increment) to read the first MAP-addressed location and successive locations.

Notice: There is no MAP auto-increment or data prefetch when MAP equals 0x0403, so the final two read cycles repeat the previous read value and address.

```
FROM   FROM
HOST   HI-6136   COMMENT
===== ======= ==========================================
0xD8   ---- SPI op enables memory address pointer (MAP) 0x000B.
0x8B   ---- SPI op code writes memory address pointer (MAP).
0x0400 ---- RT Descriptor Table start address written to MAP.
0x40   ---- SPI op code to “read location addressed by MAP”
---- <data> data from 0x0400 (SCK continues afterward)
---- <data> data from 0x0401 (SCK continues afterward)
---- <data> data from 0x0402 (SCK continues afterward)
---- <data> data from 0x0403 (SCK continues afterward, Control Word next)
---- <data> data from 0x0403 (SCK continues afterward, Control Word next)
---- <data> data from 0x0403 (SCK stops and /CS is then negated)
```

Using a different SPI op code sequence, the host can read the entire RT Descriptor Table without getting stuck at the first Control Word read by the multi-word transfer using op code 0x40:

```
FROM   FROM
HOST   HI-6136   COMMENT
===== ======= ==========================================
0xD8   ---- SPI op enables memory address pointer (MAP) 0x000B.
0x8B   ---- op code writes memory address pointer (MAP)
0x03FF ---- decremented table start addr 0x0400 - 1 written to MAP

0xD0   ---- op code increments enabled MAP
0x40   ---- op code reads MAP-addressed location
---- <data> data from 0x0400 (SCK continues afterward)
---- <data> data from 0x0401 (SCK continues afterward)
---- <data> data from 0x0402 (SCK continues afterward)
---- <data> data from 0x0403 (SCK stops and /CS is negated)
```

```
0xD0   ---- op code increments enabled MAP
0x40   ---- op code reads MAP-addressed location
---- <data> data from 0x0404 (SCK continues afterward)
---- <data> data from 0x0405 (SCK continues afterward)
---- <data> data from 0x0406 (SCK continues afterward)
---- <data> data from 0x0407 (SCK stops and /CS is negated)

0xD0   ---- op code increments enabled MAP
0x40   ---- op code reads MAP-addressed location
---- <data> data from 0x0408 (SCK continues afterward)
---- <data> data from 0x0409 (SCK continues afterward)
---- <data> data from 0x040A (SCK continues afterward)
---- <data> data from 0x040B (SCK stops and /CS is negated)
```

The host may repeat this sequence until the entire RT Descriptor Table is read. The repeating read process is not shown, but the sequence could end like this, stopping at the upper table boundary…

```
0xD0   ---- op code increments enabled MAP
0x40   ---- op code reads MAP-addressed location
---- <data> data from 0x05FC (SCK continues afterward)
---- <data> data from 0x05FD (SCK continues afterward)
---- <data> data from 0x05FE (SCK continues afterward)
---- <data> data from 0x05FF (SCK stops and /CS is negated) TABLE ENDS
```

In most situations, the repeating op code block (0xD0 with 8 SCK clocks and op code 0x40 with 72 SCK clocks) would be implemented as a loop, rather than straight-line code. A total of 128 loop repetitions would be required to read the RT Descriptor Table from start to finish.

Table 16. Fast-Access SPI Commands for Lower Registers

Op Code Bits 5:2 Convey the 4-Bit Register Read Address.
Command Bits 5:0 Convey the 6-Bit Register Write Address.

Command Bits 7 6 5 4 3 2 1 0	HEX Byte	Fast-Access Read	Command Bits 7 6 5 4 3 2 1 0	HEX Byte	Fast-Access Write
0 0 0 0 0 0 0 0	0x00	Read Register 0	1 0 0 0 0 0 0 0	0x80	Write Register 0
0 0 0 0 0 1 0 0	0x04	Read Register 1	1 0 0 0 0 0 0 1	0x81	Write Register 1
0 0 0 0 1 0 0 0	0x08	Read Register 2	1 0 0 0 0 0 1 0	0x82	Write Register 2
0 0 0 0 1 1 0 0	0x0C	Read Register 3	1 0 0 0 0 0 1 1	0x83	Write Register 3
0 0 0 1 0 0 0 0	0x10	Read Register 4	1 0 0 0 0 1 0 0	0x84	Write Register 4
0 0 0 1 0 1 0 0	0x14	Read Register 5	1 0 0 0 0 1 0 1	0x85	Write Register 5
0 0 0 1 1 0 0 0	0x18	Read Register 6	1 0 0 0 0 1 1 0	0x86	Write Register 6
0 0 0 1 1 1 0 0	0x1C	Read Register 7	1 0 0 0 0 1 1 1	0x87	Write Register 7
0 0 1 0 0 0 0 0	0x20	Read Register 8	1 0 0 0 1 0 0 0	0x88	Write Register 8
0 0 1 0 0 1 0 0	0x24	Read Register 9	1 0 0 0 1 0 0 1	0x89	Write Register 9
0 0 1 0 1 0 0 0	0x28	Read Register 10	1 0 0 0 1 0 1 0	0x8A	Write Register 10
0 0 1 0 1 1 0 0	0x2C	Read Register 11	1 0 0 0 1 0 1 1	0x8B	Write Register 11
0 0 1 1 0 0 0 0	0x30	Read Register 12	1 0 0 0 1 1 0 0	0x8C	Write Register 12
0 0 1 1 0 1 0 0	0x34	Read Register 13	1 0 0 0 1 1 0 1	0x8D	Write Register 13
0 0 1 1 1 0 0 0	0x38	Read Register 14	1 0 0 0 1 1 1 0	0x8E	Write Register 14
0 0 1 1 1 1 0 0	0x3C	Read Register 15	1 0 0 0 1 1 1 1	0x8F	Write Register 15
decimal 15 is end of read address range			1 0 0 1 0 0 0 0	0x90	Write Register 16
			1 0 0 1 0 0 0 1	0x91	Write Register 17
			1 0 0 1 0 0 1 0	0x92	Write Register 18
			1 0 0 1 0 0 1 1	0x93	Write Register 19
			1 0 0 1 0 1 0 0	0x94	Write Register 20
			1 0 0 1 0 1 0 1	0x95	Write Register 21
			and so on, to 63 decimal	. .	
			1 0 1 1 1 1 0 1	0xBD	Write Register 61
			1 0 1 1 1 1 1 0	0xBE	Write Register 62
			1 0 1 1 1 1 1 1	0xBF	Write Register 63
			decimal 63 is end of write address range		

Table 17. SPI Commands using Memory Address Pointer

Hex Byte	Read or Write	Read
		Select / Enable a Memory Address Pointer
0xD8	-------	Enable Memory Address Pointer at register 0x000B
0xD9	-------	Enable Memory Address Pointer at register 0x000C
0xDA	-------	Enable Memory Address Pointer at register 0x000D
0xDB	-------	Enable Memory Address Pointer at register 0x000E
		Memory Address Pointer Operations (no data is written or read)
0xD0	-------	Add 1 to the current value of the enabled Memory Address Pointer
0xD2	-------	Add 2 to the current value of the enabled Memory Address Pointer
0xD4	-------	Add 4 to the current value of the enabled Memory Address Pointer
		Read / Write RAM or Register Location Using Current Address Pointer Value
0x40	R	Read location addressed by the current value of the enabled Memory Address Pointer
0xC0	W	Write location addressed by the current value of the enabled Memory Address Pointer
		Increment Address Pointer Then Write Addressed RAM or Register Location
0xC8	W	Write addressed location after incrementing the enabled Memory Address Pointer
		Special Purpose Commands
0x48	R	Copy RT current Control Word address (register 3) to enabled Memory Address Pointer, then read the location addressed by the new pointer value (read the current Control Word)
0x68	R	Add 0 to the current value of the enabled Memory Address Pointer. Then . . .
0x70	R	Add 1 to the current value of the enabled Memory Address Pointer. Then . . .
0x78	R	Add 2 to the current value of the enabled Memory Address Pointer. Then copy value from newly addressed location to the enabled Memory Address Pointer then read newly addressed location.
0xE8	W	Add 0 to the current value of the enabled Memory Address Pointer. Then . . .
0xF0	W	Add 1 to the current value of the enabled Memory Address Pointer. Then . . .
0xF8	W	Add 2 to the current value of the enabled Memory Address Pointer. Then copy value from newly addressed location to the enabled Memory Address Pointer then write newly addressed location.
0x60	R	Read then add 4 to the current value of the enabled Memory Address Pointer.
0x58	R	Write storage address of last-written Interrupt Address Word to the enabled Memory Address Pointer, then read the Interrupt Address Word from the Interrupt Log buffer. Decrement Memory Address Pointer after read operation

21. 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 RTA and RTAP 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	HI-6136 is not equipped to accept bus control duties. The host must initialize device 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 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 device 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 $\overline{MR}$ Master Reset input pin.				
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 device automatically re-enables **transmit and receive** for the inactive bus, without considering BSDTXO bit status in the Master Configuration Register. The device 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 pins are asserted, the device 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/$\overline{R}$ 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 device 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 device 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 $\overline{MR}$ master reset input pin.				
MC6 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 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 device 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 device 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”.

<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>RESET REMOTE TERMINAL (MC8):
Mode code command with mode code 01000
and T/$\overline{R}$ bit equals 1</td><td>Default response: Reset Message Error (ME) status. If not broadcast, transmit Status Word.</td><td>DBAC bit reset.
BCAST bit updated.
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
IBR</td></tr>
<tr><td colspan="5">After Status transmission, the device 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 MR hardware master reset (with or without auto-initialization) 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.</td></tr>
<tr><td>MC8 EXCEPTIONS:</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Invalid command word.
OR
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
AND
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>
<tr><td>T/$\overline{R}$ bit equals 0
AND
UMCINV bit in the RT Configuration Register equals 0.
The Illegalization Table bit equals 1 **</td><td>Set Message Error (ME) status. If not broadcast, transmit Status Word. If broadcast, also set Status Word BCR bit and suppress Status response.</td><td>DBAC bit set.
BCAST bit updated.
DPB bit toggles.</td><td>ILCMD bit set.
MERR bit set.
BUSID bit updated.
RTRT bit reset.
(Other error bits reset.)</td><td>ILCMD
IWA
IBR</td></tr>
<tr><td>Mode code command word is followed by a contiguous data word</td><td>No Status Word transmit. Set the Message Error (ME) status bit.</td><td>DBAC bit set.
BCAST bit reset.
DPB bit toggles.</td><td>MERR bit set.
GAPERR bit set.
BUSID bit updated.
ILCMD bit reset.
RTRT bit reset.
(Other error bits reset.)</td><td>MERR
IWA</td></tr>
</table>

* 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 device 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
Device 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
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 device 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 device 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 pin 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 RTRESET bit in the “Master Status and Reset Register (0x0001)”, or by hardware reset initiated by asserting the $\overline{MR}$ Master Reset input pin.

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
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
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 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".

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 SELECTED TRANSMITTER SHUTDOWN (MC21): Mode code command with mode code 10101 and T/$\overline{R}$ bit equals 1	Default response: Reset Message Error (ME) status. and transmit Status Word. If broadcast, set the 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

After Status Word transmission, the device 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 device automatically re-enables **transmit and receive** for the inactive bus, regardless of the state of the BSDTXO bit in the Master Configuration Register. The device 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)”. ***Note:*** *If the TXINHA or TXINHB input pins are asserted, the device 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.*

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. **The host** must evaluate whether the received mode data word matches the bus selection criteria. If bus selection match occurs, **the host** 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. ***Note:*** *Resetting the RTINHA or RTINHB shutdown bit cannot restore bus transmit capability if the TXINHA or TXINHB input pin is asserted,*

OR

2. reset the transmit shutdown input pin TXINHA or TXINHB for the inactive bus to re-enable transmit if the host used this pin to shut down transmit only for an earlier MC4 or MC20 command.

MC21 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/$\overline{R}$ 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 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 device 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. Device 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”

22. ELECTRICAL CHARACTERISTICS

22.1. Absolute Maximum Ratings

Supply voltage (V_{DD})	-0.3 V to +5.0 V
Logic input voltage range	-0.3 V to +3.6 V
Receiver differential voltage	10 Vp-p
Power dissipation at 25°C	1.0 W
Solder Temperature (reflow)	260°C
Junction Temperature	175°C
Storage Temperature	-65°C to +150°C

22.2. Recommended Operating Conditions

Operating Supply voltage (V_{DD})	3.3 VDC ± 5%
Operating Temperature Range	
Industrial Extended	-40°C to +85°C -55°C to +125°C

NOTE: Stresses above absolute maximum ratings or outside recommended operating conditions may cause permanent damage to the device. These are stress ratings only. Operation at the limits is not recommended.

22.3. DC Electrical Characteristics

V_{DD} = 3.3V, GND = 0V, T_A = Operating Temperature Range (unless otherwise stated)

Parameters	Symbol	Test Conditions	Limits			Unit
			Min	Typ	Max	
Operating Voltage	V_{DD}		3.15	3.3	3.45	V
Power Supply Current See Note 1 on next page	I_{CC1}	Not Transmitting	-	10	15	mA
	I_{CC2}	Continuous supply current while one bus transmits @ 100% duty cycle, 70Ω resistive load	-	875	975	mA
Power Dissipation See Note 2 on next page	PD_1	Not Transmitting	-	-	60	mW
	PD_2	Transmit one bus @ 100% duty cycle, 70Ω resistive load	-	492	650	mW
Input Voltage (High)	V_{IH}	Digital Inputs	70%	-	-	V_{DD}
Input Voltage (Low)	V_{IL}	Digital Inputs	-	-	30%	V_{DD}
Input Current (High): Inputs with pull-down.	I_{IH}	Digital Inputs (each digital input pulled high)	-	-	100	µA

Parameters		Symbol	Test Conditions	Limits			Unit
				Min	Typ	Max	
Input Current (High): Inputs with pull-up.		I_{IH}	Digital Inputs (each digital input pulled high)	+1	-	-	µA
Input Current (Low) Inputs with pull-up.		I_{IL}	Digital Inputs (each digital input pulled low)	-	-	-100	µA
Input Current (Low) Inputs with pull-down.		I_{IL}	Digital Inputs (each digital input pulled low)	-1	-	-	µA
Output Voltage (High)		V_{OH}	I_{OUT} = -1.0mA, Digital outputs	90%	-	-	V_{DD}
Output Voltage (Low)		V_{IH}	I_{OUT} = 1.0mA, Digital outputs	-	-	10%	V_{DD}
RECEIVER (Measured at Point "AD" in Figure 25 unless otherwise specified)							
Input Resistance		R_{IN}	Differential	20	-	-	kΩ
Input Capacitance		C_{IN}	Differential	-	-	5	pF
Common Mode Rejection Ratio		CMRR		40	-	-	dB
Input Level		V_{IN}	Differential	-	-	9	Vp-p
Input Common Mode Voltage		V_{ICM}		-5	-	+5	V-pk
Threshold Voltage (Direct-Coupled)	Detect	V_{THD}	1 MHz Sine Wave (Measured at Point "AD" in Figure 25)	1.15	-	20.0	Vp-p
	No Detect	V_{THND}		-	-	0.28	Vp-p
Threshold Voltage (Transformer-Coupled)	Detect	V_{THD}	1 MHz Sine Wave (Measured at Point "AT" in Figure 26)	0.86	-	14.0	Vp-p
	No Detect	V_{THND}		-	-	0.2	Vp-p
TRANSMITTER (Measured at Point "AD" in Figure 25 unless otherwise specified)							
Output Voltage	Direct Coupled	V_{OUT}	35Ω Load	6.6	-	9.0	Vp-p
	Transformer Coupled	V_{OUT}	70Ω Load (Measured at Point "AT" in Figure 26)	20.0	-	27.0	Vp-p
Output Noise		V_{ON}	Differential, inhibited	-	-	10.0	mVp-p
Output Dynamic Offset Voltage	Direct Coupled	V_{DYN}	35Ω Load	-90	-	90	mV
	Transformer Coupled	V_{DYN}	70Ω Load (Measured at Point "AT" in Figure 26)	-250	-	250	mV
Output Resistance		R_{OUT}	Differential, not transmitting	10	-	-	kΩ
Output Capacitance		C_{OUT}	1 MHz sine wave	-	-	15	pF

Note 1: In actual use, the highest practical transmit duty cycle is 96%, occurring when a Remote Terminal responds to a series of 32 data word transmit commands (RT to BC) repeating with minimum intermessage gap of 4µs (2µs dead time) and typical RT response delay of 5µs.

Note 2: While one bus continuously transmits, the power delivered by the 3.3V power supply is 3.3V × 875mA typical = 2.9W. Of this, 492mW is dissipated in the device, the remainder in the load.

22.4. AC Electrical Characteristics — Host Bus Interface (SPI) Timing

V_{DD} = 3.3V, GND = 0V, T_A = Operating Temperature Range (unless otherwise stated)

Parameters	Symbol	Limits			Units
		Min	Typ	Max	
INTERFACE TIMING (SPI Host Bus Interface)					
SCK clock Period	t_{CYC}	25	-	-	ns
$\overline{CE}$ set-up time to first SCK rising edge	t_{CES}	15	-	-	ns
$\overline{CE}$ hold time after last SCK rising edge	t_{CEH}	15	-	-	ns
$\overline{CE}$ inactive between SPI instructions	t_{CPH}	100	-	-	ns
SPI SI Data set-up time to SCK rising edge	t_{DS}	10	-	-	ns
SPI SI Data hold time after SCK rising edge	t_{DH}	10	-	-	ns
SO valid after SCK falling edge	t_{DV}	-	-	10	ns
SO high-impedance after $\overline{CE}$ inactive	t_{CHZ}	-	-	90	ns

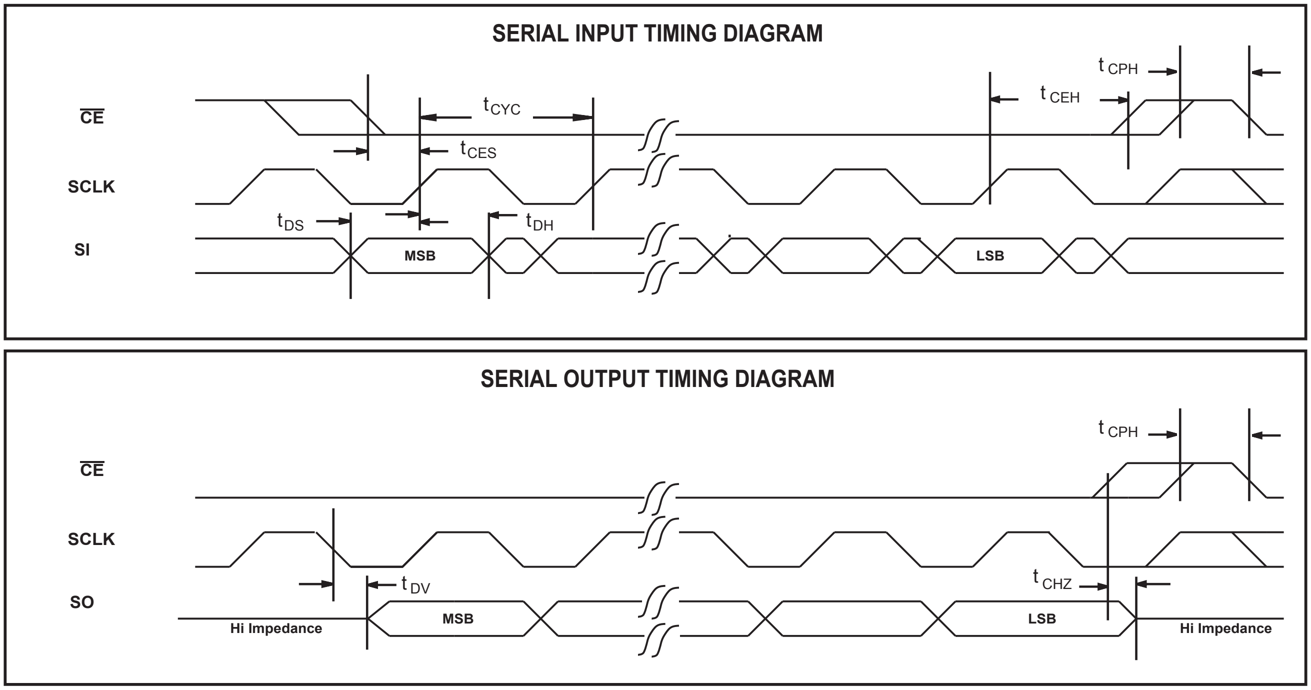


Figure 23. Host Bus Interface Timing Diagram

23. MIL-STD-1553 BUS INTERFACE

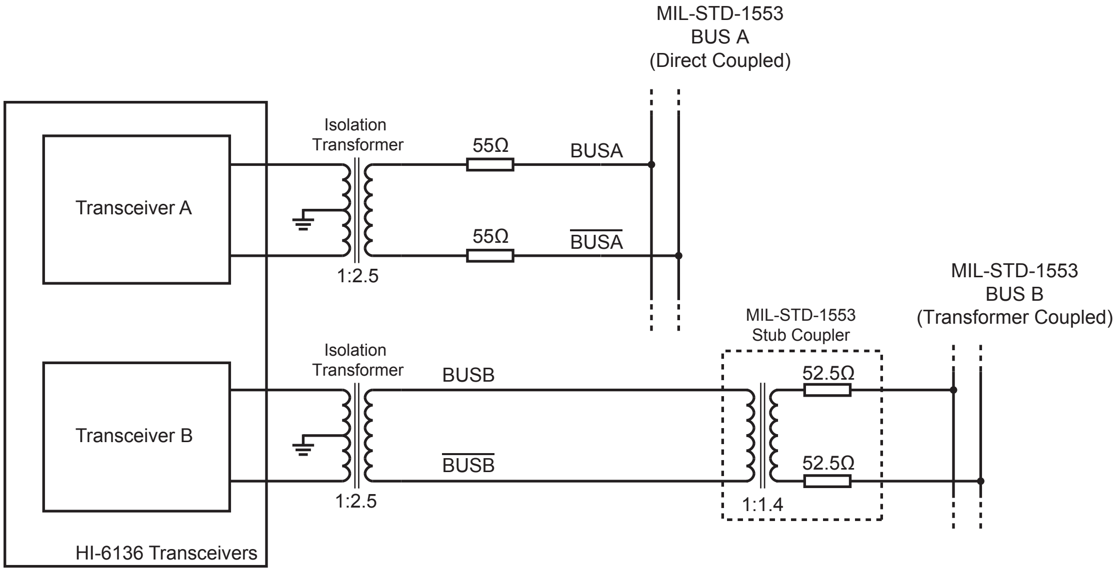


Figure 24. Bus Connection Example using HI-6136

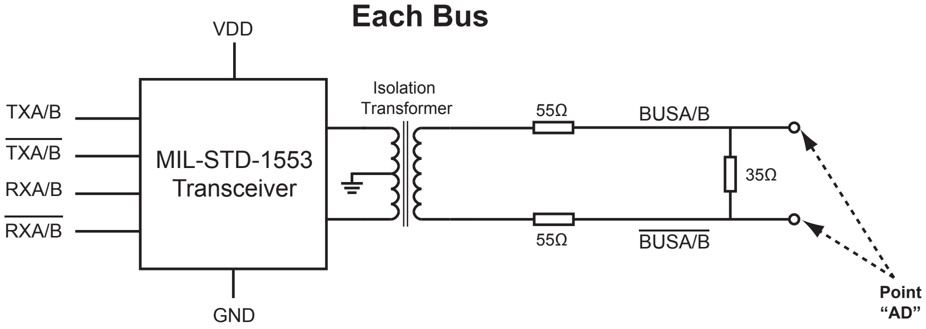


Figure 25. MIL-STD-1553 Direct Coupled Test Circuits

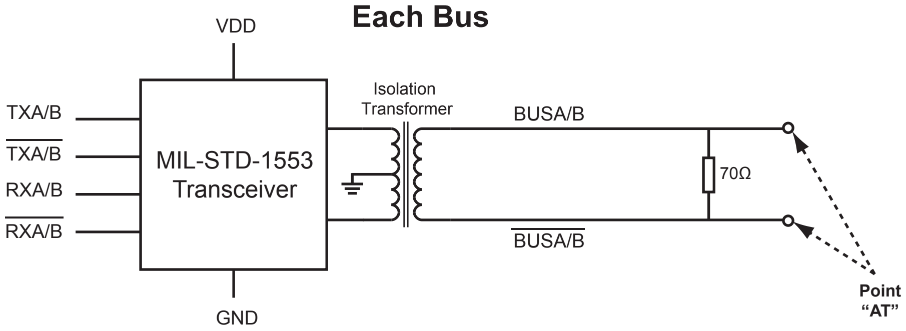


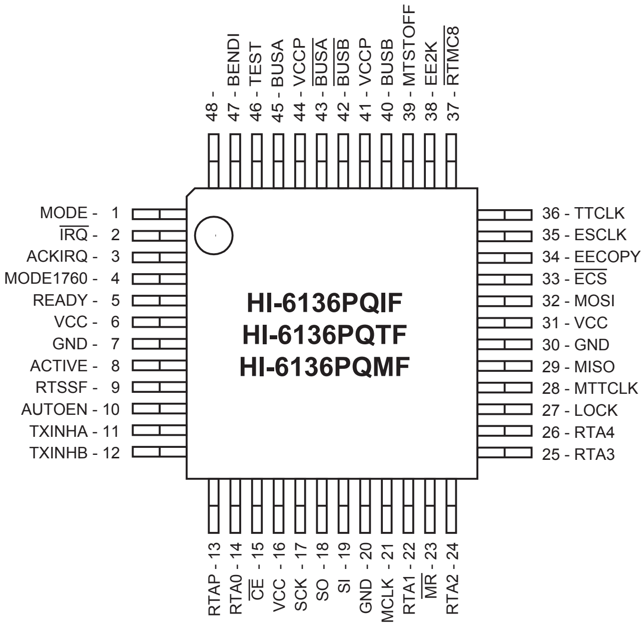
Figure 26. MIL-STD-1553 Transformer Coupled Test Circuits

24. ADDITIONAL PIN / PACKAGE CONFIGURATIONS

Notes:

1. All VCC, VCCP and GND pins must be connected.
2. See page 1 for HI-6136, 48-Pin QFN Package Configuration.

24.1. HI-6136PQx (48-pin PQFP)



48 - Pin Plastic Quad Flat Pack (PQFP)

25. ORDERING INFORMATION

HI - 6136 Px x x

PART NUMBER	LEAD FINISH
Blank	Tin / Lead (Sn / Pb) Solder
F	100% Matte Tin (Pb-free, RoHS compliant)

PART NUMBER	TEMPERATURE RANGE	FLOW	BURN IN
I	-40°C to +85°C	I	No
T	-55°C to +125°C	T	No
M	-55°C to +125°C	M	Yes

PART NUMBER	PACKAGE DESCRIPTION
PC	48 PIN PLASTIC CHIP-SCALE PACKAGE, 6 x 6mm QFN (48PCS6)
PQ	48 PIN PLASTIC QUAD FLAT PACK, LQFP (48PQS)

26. REVISION HISTORY

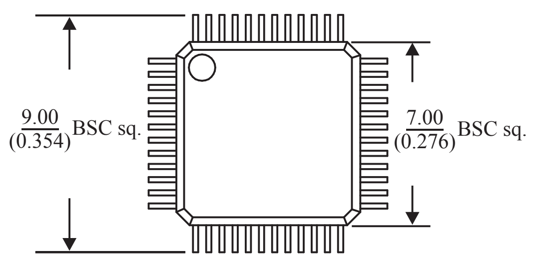
Revision	Date	Description of Change
DS6136, Rev. New	7/14/15	Initial Release
Rev. A	10/5/15	In “DC Electrical Characteristics” on page 198, update Power Supply Current (Not Transmitting) from 4 mA typ. and 10 mA max. to 10 mA typ. and 15 mA max. respectively. Update Power Supply Current (Transmitting) from 720 mA typ. and 760 mA max. to 875 mA typ. and 975 mA max. respectively. Update Power Dissipation. In “AC Electrical Characteristics — Host Bus Interface (SPI) Timing” on page 200, change “SO high-impedance after $\overline{CE}$ inactive” parameter from 75ns to 90ns. Indicate t_{CYL} on SPI Input Timing Diagram. Correct SPI operation modes in “Serial EEPROM Programming Utility” on page 143 Functional polarity of AUTOBSD bit in “Remote Terminal Configuration Register (0x0017)” on page 71 is reversed. Correct throughout datasheet. Update READY pin delay time to include EE2K auto-initialization option (see “Hardware Master Reset and Optional Auto-Initialization” on page 147). Update MODE1760 bit description in “Master Configuration Register 2 (0x004E)” on page 29.

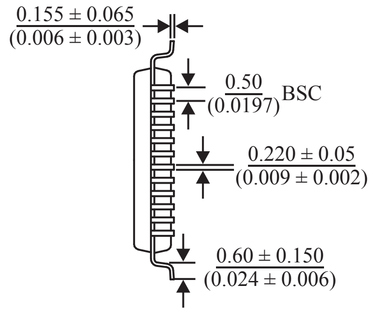
27. PACKAGE DIMENSIONS

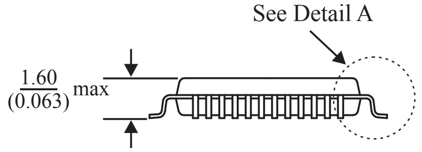
48-PIN PLASTIC QUAD FLAT PACK (PQFP)

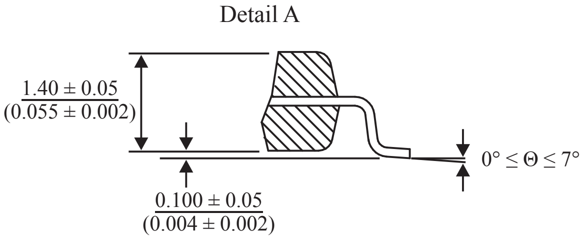
millimeters (inches)

Package Type: 48PQS







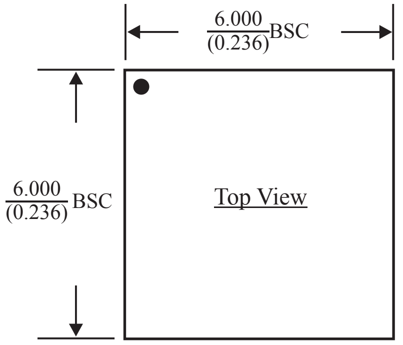


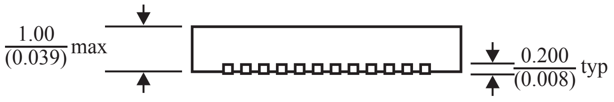
BSC = “Basic Spacing between Centers” is theoretical true position dimension and has no tolerance. (JEDEC Standard 95)

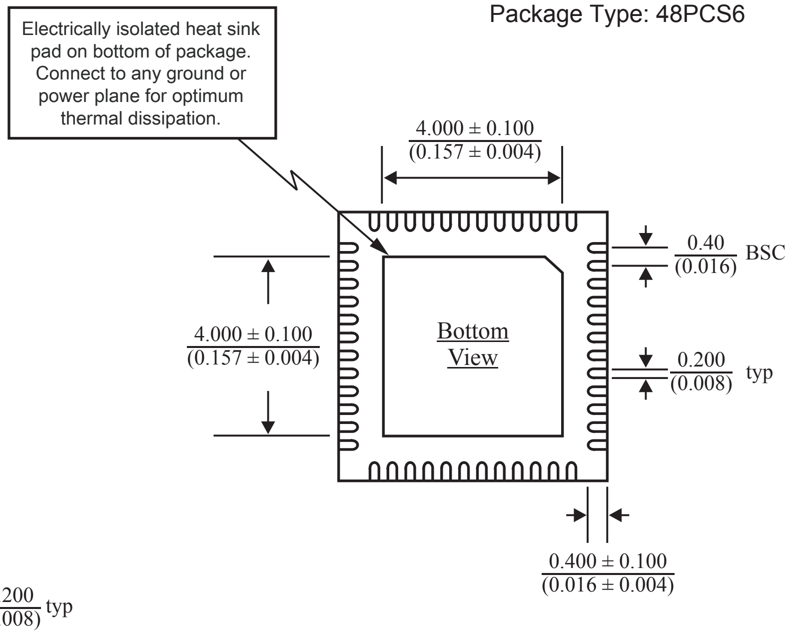
48-PIN PLASTIC CHIP-SCALE PACKAGE (QFN)

millimeters (inches)

Package Type: 48PCS6






Electrically isolated heat sink pad on bottom of package. Connect to any ground or power plane for optimum thermal dissipation.

BSC = “Basic Spacing between Centers” is theoretical true position dimension and has no tolerance. (JEDEC Standard 95)