DDC
°Data Device Corporation

MIL-STD-1553
DESIGNER'S GUIDE

SIXTH EDITION

12/8/2003

First Edition . . . .(13,500) 1982
   Second Printing . . . .(10,000) 1983
   Third Printing . . . .(10,000) 1984
   Fourth Printing . . . .(10,000) 1986
Second Edition . . . .(10,000) 1987
   Second Printing . . . .(10,000) 1988
Third Edition . . . .(10,000) 1990
Fourth Edition . . . .(10,000) 1993
Fifth Edition . . . .(6,000) 1995
   Second Printing . . . .(3,500) 1997
Sixth Edition . . . .(8,000) 1998

PREFACE

Since its inception in 1973 and in subsequent revisions during the ensuing years, MIL-STD-1553 has evolved into the predominant, internationally accepted networking standard for the integration of military platforms. Supporting documents such as the *MIL-STD-1553 Multiplex Applications Handbook*, the RT *Validation and Production Test Plans*, and DDC's *MIL-STD-1553 Designer's Guide*, have positively influenced the standard's worldwide familiarity and acceptance. Today, 1553 has expanded beyond its traditional domain of US Air Force and Navy aircraft to encompass applications for combat tanks, ships, satellites, missiles and the International Space Station Program. Despite the recent advent of newer and higher-speed technologies, it is clearly evident that 1553 will continue to be used extensively in evolving upgrade programs as well as for new applications and integration platforms for years to come.

Focused on meeting the needs of hardware and software design engineers confronted with 1553/1760/1773 interface and/or test requirements, DDC is pleased to publish and provide you with this sixth edition of the *Designer's Guide* for the MIL-STD-1553 community.

As in its five previous editions, the *Designer's Guide* is divided into sections to facilitate your ease of reference and review. Section I provides you with an overview of the philosophy and implementation of MIL-STD-1553B and several related standards. Using DDC 1553 products to illustrate key points of interest within the framework of these technical discussions, Section I also presents a comparison of MIL-STDs-1553A and B as well as an overview of MIL-STD-1760B. Taken directly from the *MIL-STD-1553 Multiplex Applications Handbook*, Section II provides you with a complete copy of MIL-STD-1553 for your reference, and, a section-by-section interpretation of the rationale behind the various paragraphs in the standard. Plus, for your ongoing reference requirements, copies of Notices 1 and 2 to MIL-STD-1553B, as well as the Production Test Plan and Validation Test Plan are presented in Sections III through VI.

As the leading full-range developer and manufacturer of 1553 data bus components, boards and software, DDC offers a comprehensive collection of technical data sheets delineating our wide array of 1553 data conversion and communications products in Section VII. These include fully integrated BC/RT/MT components, transceivers, RT components, transformers, software and a host PCMCIA, VME/VXI and PC-compatible communications and tester/simulator board products.

The most recent information regarding DDC's 1553 product uses, including space-level applications and our series of Advance Communication Engine (ACE) terminals, is presented in Section VIII. A series of application notes, highlighting practical uses and design concerns when implementing 1553 technology, is provided in Section IX. A listing of corresponding DDC and DESC part numbers is provided in Section X. For your reference, Subject and Product Number Indices are provided in Section XI.

DDC would like to thank Chris deLong and Crossgrove & Associates, for their contributions to the *Designer's Guide's* development, and the U.S. Air Force for allowing us to present portions of their *MIL-STD-1553 Multiplex Applications Handbook*. We would also like to extend our gratitude to our customers and friends who have attended our seminars and shared their 1553 experiences with us.

We sincerely hope this sixth edition of DDC's *MIL-STD-1553 Designer's Guide* will prove useful as a working document and reference source, enabling you to resolve your requirements in a practical, efficient, cost-effective, reliable manner. While we have made every effort to ensure the accuracy and correctness of the information presented within these pages, we ask that you do not hesitate to call our attention to any omissions, oversights, errors, miscalculations or any suggestions you might have for enhancing and improving this publication.

Thank you for your attention, ongoing consideration and patronage.

ILC Data Device Corporation

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

I.
DESIGNER'S NOTES

INTRODUCTION

MIL-STD-1553, Digital Internal Time Division Command/Response Multiplex Data Bus, is a military standard (presently in revision B), which has become one of the basic tools being used today by the DoD for integration of weapon systems. The standard describes the method of communication and the electrical interface requirements for subsystems connected to the data bus. The 1 Mbps serial communication bus is used to achieve aircraft avionic (MIL-STD-1553B) and stores management (MIL-STD-1760B) integration. In the future it will be used to extend the systems integration to flight controls, propulsion controls, and vehicle management system (electrical, hydraulic, environmental control, etc.).

Several other documents exist, which are related to MIL-STD-1553. MIL-HDBK-1553 describes the implementation practices for this standard including: design considerations, examples of applications, and guidelines for implementations. Portions of MIL-STD-1553 are also the foundation of the MIL-STD-1773 (Fiber-optics), and MIL STD-1760B (Stores Management). In addition, MIL-STD-1553 is embodied in or referenced by the following international documents: NATO STANAG 3838, ASCC Air Standard 50/2, and UK DEF STAN 00-18 (Part 2)/Issue 1.

This Designer's Guide will provide; 1) a synopsis of the standard and some frequently asked questions and their answers, 2) a specification comparison of the detailed uses of the standard, 3) a review of system design considerations, and 4) a discussion of remote terminal and bus controller designs.

1.0 MIL-STD-1553 OVERVIEW

The 1553 standard is organized similar to most military standards with a foreword, scope, referenced document section, definitions, general requirements, the appendix, and a tri-service Notice 2. Notice 2, which supersedes Notice 1, was developed to define which options of the standard are required to enhance tri-service interoperability and to further define some of the open-ended timing variables implied within the standard.

1.1 Key Elements

Some of the key MIL-STD-1553B elements are the bus controller, the embedded remote terminal (a sensor or subsystem that provides its own internal 1553 interface), the stand-alone remote terminal, bus monitor, and two other devices that are part of

the 1553 integration; the twisted shielded pair wire data bus and the isolation couplers that are optional.

The bus controller's main function is to provide data flow control for all transmissions on the bus. In this role, the bus controller is the sole source of communication. The system uses a command /response method.

The embedded remote terminal consists of interface circuitry located inside a sensor or subsystem directly connected to the data bus. Its primary job is to perform the transfer of data in and out of the subsystem as controlled by the bus controller. This type of terminal usually does not have bus controller capability. However, if the sensor itself is fairly intelligent, it can become a candidate for the backup bus controller function. Generally, **an intelligent subsystem (i.e., computer based) can become a backup bus controller if a second computer, equal in function to the primary, is unavailable.**

The stand-alone remote terminal is the only device solely dedicated to the multiplex system. It is used to interface various subsystem(s), which are not 1553 compatible with the 1553 data bus system. Its primary function is to interface and monitor transmission in and out of these non-1553 subsystem(s).

The bus monitor listens to all messages, and subsequently collects data, from the data bus. Primary applications of this mode of operation include: collection of data for on board bulk storage or remote telemetry; or use within a "hot" or off-line back-up controller to observe the state and operational mode of the system and subsystems.

The fourth item is the data bus itself. The standard defines specific characteristics for the twisted pair shielded cable. Notice 2 tightens these requirements and adds a definition for connector polarity.

The last item to be discussed is the data bus coupler unit that isolates the main bus from the terminals. MIL-STD-1553B allows two types of data bus interface techniques; direct coupling and transformer coupling. Subsystems and 1553 bus elements are interfaced to the main data bus by interconnection buses (called "stubs"). These stubs are either connected directly to the main bus or interfaced via data bus couplers. The data bus couplers contain two isolation resistors (one per wire) and an isolation transformer (with a ratio of 1 to the square root of 2). The purpose of the data bus couplers is to

I.
DESIGNER'S NOTES

INTRODUCTION

MIL-STD-1553, Digital Internal Time Division Command/Response Multiplex Data Bus, is a military standard (presently in revision B), which has become one of the basic tools being used today by the DoD for integration of weapon systems. The standard describes the method of communication and the electrical interface requirements for subsystems connected to the data bus. The 1 Mbps serial communication bus is used to achieve aircraft avionic (MIL-STD-1553B) and stores management (MIL-STD-1760B) integration. In the future it will be used to extend the systems integration to flight controls, propulsion controls, and vehicle management system (electrical, hydraulic, environmental control, etc.).

Several other documents exist, which are related to MIL-STD-1553. MIL-HDBK-1553 describes the implementation practices for this standard including: design considerations, examples of applications, and guidelines for implementations. Portions of MIL-STD-1553 are also the foundation of the MIL-STD-1773 (Fiber-optics), and MIL STD-1760B (Stores Management). In addition, MIL-STD-1553 is embodied in or referenced by the following international documents: NATO STANAG 3838, ASCC Air Standard 50/2, and UK DEF STAN 00-18 (Part 2)/Issue 1.

This Designer's Guide will provide; 1) a synopsis of the standard and some frequently asked questions and their answers, 2) a specification comparison of the detailed uses of the standard, 3) a review of system design considerations, and 4) a discussion of remote terminal and bus controller designs.

1.0 MIL-STD-1553 OVERVIEW

The 1553 standard is organized similar to most military standards with a foreword, scope, referenced document section, definitions, general requirements, the appendix, and a tri-service Notice 2. Notice 2, which supersedes Notice 1, was developed to define which options of the standard are required to enhance tri-service interoperability and to further define some of the open-ended timing variables implied within the standard.

1.1 Key Elements

Some of the key MIL-STD-1553B elements are the bus controller, the embedded remote terminal (a sensor or subsystem that provides its own internal 1553 interface), the stand-alone remote terminal, bus monitor, and two other devices that are part of

the 1553 integration; the twisted shielded pair wire data bus and the isolation couplers that are optional.

The bus controller's main function is to provide data flow control for all transmissions on the bus. In this role, the bus controller is the sole source of communication. The system uses a command /response method.

The embedded remote terminal consists of interface circuitry located inside a sensor or subsystem directly connected to the data bus. Its primary job is to perform the transfer of data in and out of the subsystem as controlled by the bus controller. This type of terminal usually does not have bus controller capability. However, if the sensor itself is fairly intelligent, it can become a candidate for the backup bus controller function. Generally, **an intelligent subsystem (i.e., computer based) can become a backup bus controller if a second computer, equal in function to the primary, is unavailable.**

The stand-alone remote terminal is the only device solely dedicated to the multiplex system. It is used to interface various subsystem(s), which are not 1553 compatible with the 1553 data bus system. Its primary function is to interface and monitor transmission in and out of these non-1553 subsystem(s).

The bus monitor listens to all messages, and subsequently collects data, from the data bus. Primary applications of this mode of operation include: collection of data for on board bulk storage or remote telemetry; or use within a "hot" or off-line back-up controller to observe the state and operational mode of the system and subsystems.

The fourth item is the data bus itself. The standard defines specific characteristics for the twisted pair shielded cable. Notice 2 tightens these requirements and adds a definition for connector polarity.

The last item to be discussed is the data bus coupler unit that isolates the main bus from the terminals. MIL-STD-1553B allows two types of data bus interface techniques; direct coupling and transformer coupling. Subsystems and 1553 bus elements are interfaced to the main data bus by interconnection buses (called "stubs"). These stubs are either connected directly to the main bus or interfaced via data bus couplers. The data bus couplers contain two isolation resistors (one per wire) and an isolation transformer (with a ratio of 1 to the square root of 2). The purpose of the data bus couplers is to

prevent a short on a single stub from shorting the main data bus. The selection of the value of the resistors, the transformer's turn ratio, and the receiver impedance are such that the stub appears to the main bus as a "clean interface" (i.e., high impedance). This technique reduces the distortion caused on the main data bus by the termination. The characteristics of the data bus couplers are discussed in paragraph 4.2.4. Main buses utilizing direct coupled stubs must be designed to withstand the impedance mismatch of the stubs. This can be reduced by minimizing stub length (less than one foot) and "tuning" the bus by terminal spacing. **Designs not using data bus couplers should be carefully analyzed and tested to determine if waveform distortion is significant enough to cause receiver problems.** The other risk associated with direct coupled stubs is a short on a stub will cause the main bus to fail. The obvious advantage to direct coupled stubs is the elimination of the logistical problems associated with another device and the installation problem of locating these small devices (approximately 1 inch cube) in the aircraft. Today, data bus couplers and line terminating resistors are available in molded packages, which can become part of the wiring harness, thus eliminating some of the installation problems. Also, multiple data bus couplers and data bus line terminating resistors are available in single packages, which reduces the number of unique units installed per aircraft.

1.2 Message Types

MIL-STD-1553 is a serial data bus based on message transmission. Therefore, considerable emphasis is placed on the term "information transfer formats," which describe each of the 10 message types. Within these 10 message types are the formats used to achieve communication, the primary function of the data bus system. **Each message format is made up of control words called command and status. Data words are used to encode communication between system elements.** Both control words and data words are used in system communication as well as data bus system control. These message formats have been subdivided into two groups by 1553B and are shown in figure I-1.1; the "information transfer formats" and the "broadcast information transfer formats." These two groups can be easily segregated because the broadcast group does not conform directly to the command/response philosophy of the other (non-broadcast) message formats. **This command/response philosophy requires that all error free messages received by a remote terminal be followed by the transmission of a remote terminal status word.** This handshaking process validates error free message completion. Since broadcast message formats are transmitted to multiple receivers, a more detailed scheme is required to validate error free message reception (this method is described in paragraph 1.4.2). Also, since address 31 is used by all terminals receiving a broadcast mes-
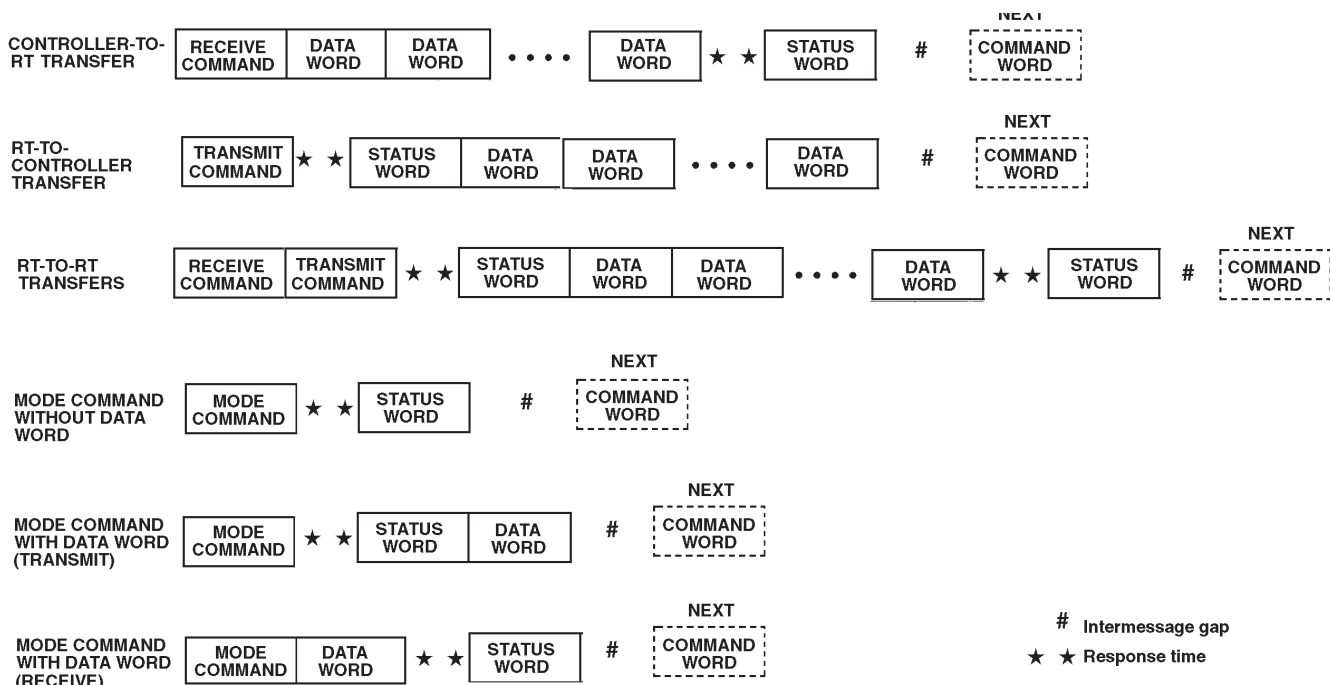
Figure I-1.1 Information Transfer Formats
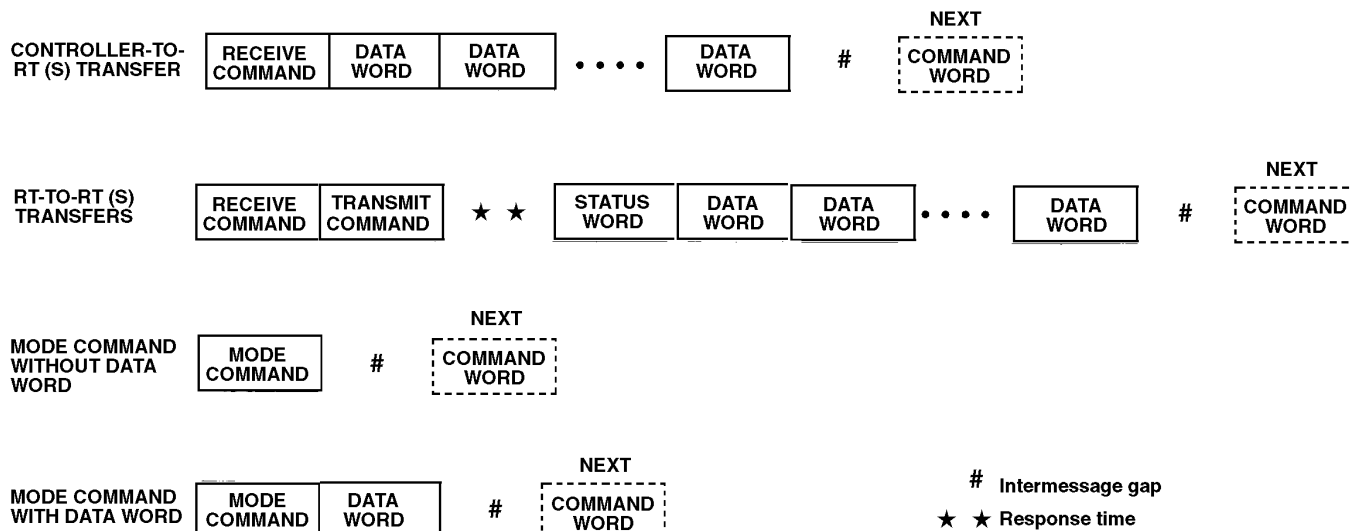
Figure I-1.1 (con't) Information Transfer Formats

sage, subaddressing needs to be managed on a data bus system basis rather than on a remote terminal basis. The ability to define and use more than 30 subaddresses is discussed in paragraph 3.7.1.

The information transfer formats allow communications between two elements in the data bus system the bus controller and the remote terminal (RT). In 1553, the bus controller is in control of all communication and it is the sole device allowed to transmit command words. **Notice that all messages are initiated by the bus controller using command word(s).**

Messages to a device (remote terminal) from the bus controller are issued using a command word (see figure II1.2) containing the remote terminal's address, direction of message transmission (transmit/receive bit), subaddress (destination within the specific remote terminal or subsystem), and the word count. The command word is immediately followed by the appropriate number of data words specified in the command word. **The receiving terminal validates error free message reception by transmitting a status word** (see figure I-1.3), which contains information about its health. Using this technique, the bus controller can transmit data to any terminal attached to the data bus. In a similar manner, the bus controller can initiate a command to a remote terminal, which requires the remote terminal to transmit a specific message to the bus controller. This is accomplished using the RT to bus controller message format. Similarly **communication**

can be established between two unique remote terminals, when the bus controller commands one terminal to receive data and the other terminal to transmit data. Neither the receiving nor the transmitting terminal knows where the message originated or destined. Both will transmit status words in the proper formats. Each status word is evaluated by the bus controller to verify error free message completion. In addition to these three message formats, three control message formats are provided to support data bus system management. These formats are mode code formats allowing the transmission of a command word and up to one data word from the bus controller to a unique remote terminal. The remote terminal's response involves the transmission of a status word and up to one data word upon receipt of the mode command. The use of each mode code will be discussed in paragraph 1.4.2.
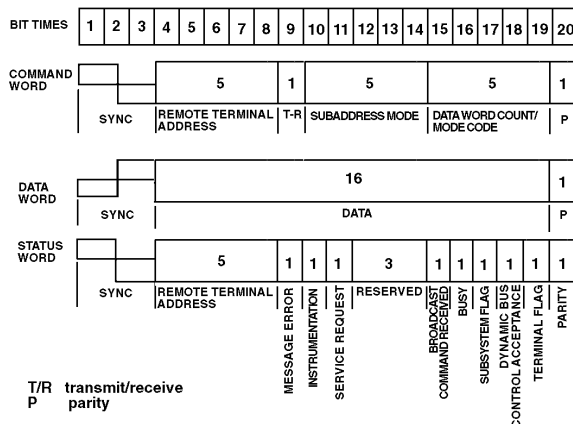
Figure I-1.2 Word Formats
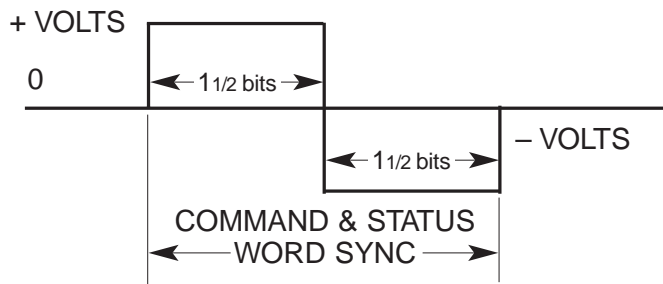
Figure I-1.3 Command and Status Word Sync

Broadcast formats are identical to the nonbroadcast formats discussed above, except the bus controller transmits commands using a remote terminal address of "31" (11111), which has been reserved for this function. All remote terminals with the broadcast reception capability will receive this command, validate error free message reception, and establish a status word response, but the transmission of the status word will be suppressed. Obviously, if multiple terminals simultaneously transmitted their status responses a bus "crash" (undetectable message transmissions) would occur. As can be seen by the RT to RT(s) transfer formats (see figure I-1.1), the bus controller can set up a broadcast data reception for all terminals and then command a single terminal (unique address) to transmit. Since the data bus system is required to follow the last valid command, the unique terminal will transmit its status word (which will be ignored by the other terminals) followed by the specified number of data words. **This format allow a a subsystem to broadcast its data directly to multiple users.** Broadcast messages can be validated using a transmit status or transmit last command mode code (see paragraph 1.4). Therefore, even broadcast message formats have a command/response approach to message validation, if required by the system design. Notice 2 of the standard allows for the transmission by the bus controller of broadcast mode codes only (whereas Notice 1 forbid all broadcast messages). This will ease the bus controller's overhead in performing such tasks as minor/major frame synchronization.

1.3 Word Types

The standard allows for only three types of words as discussed in the previous message format section; command word, status word, and data word. 1553 requires each word to consist of 16 bit of data plus, a sync pattern (3 bit times long), and a one bit parity providing a 20 bit word format. The contents of each word type is shown in figure II1.2.

The command word provides the definition of the message format to be transmitted and can only be transmitted by the bus controller. As seen in the message format section, the command word may be followed by data, another command word, or a response time gap prior to status word transmission by the remote terminal. The command word sync pattern is a unique invalid Manchester waveform, which cannot be duplicated by data (see figure I-1.3). The command word sync and the status word sync are identical and the inverse of the data word sync pattern. Therefore, command and status words, which initiate a sequence, can always be distinguished from data word sync patterns. **The command word address is always the address of the remote terminal being com- mended; a bus controller does not have an address while in the active bus controller mode** (backup bus controllers can function as an RT with a unique address or as a bus monitor with no address until they become an active bus controller). The transmit/receive (T/R) bit indicates the direction of flow of data words (i.e., receive means data to be received by the remote terminal). The subaddress/mode code field has two purposes. When a unique terminal is to receive or transmit data, the subaddress acts as an internal address to point to the type of data desired, the location of a data pointer in memory, subsystem interface, etc. (see paragraphs 3.7.1 and 5.3 for further discussion on subaddressing).

When the subaddress field is 00000 or 11111, it indicates that the next field contains the number of the mode code. The next field (word count/mode number) contains the number of data word(s) in the message or the number of the mode code. Odd parity is established for all words based on the 16 bits of data plus parity bit.

The data word contains a unique sync (three bit times long), 16 data bits, and a one bit parity. No restrictions are placed on the encoding of the data field, except that the "most significant bit shall be transmitted first." Once again, parity is odd and established on the 16 bits of data plus the parity bit.

Recently, there have been several efforts to standardize on data word formats for the more commonly used functions. The Army and Navy have recently developed data bases for some avionics equipment. The designer should check the status of these efforts before establishing a unique set of data formats for the system. Section 80 (formerly

Chapter 11) of the "MiL-HDBK-1553 Multiplex Applications Handbook" has established guidelines for the development of data word and message formats for data bus applications. (See paragraph 3.9 for further discussion on system documentation and Interface Control Documents.) The status word utilizes the same sync format as the command word. The remote terminal address is placed in the transmitted status word for two reasons; so that the status word can be validated by the bus controller (a remote terminal may also validate the status word address field against that of the command word's during an RT-RT message transfer), and so that the status word will not be misinterpreted by the other terminals as a command word. Any status word transmitted, must contain valid information at all times (i.e., following RT power-up, during initialization, and during normal operation). The message error bit is the only required status bit and it is used to identify messages which do not pass the word or message validation tests of 1553 (1553B paragraph 4.4.1.1 and 4.4.1.2). MIL-STD-1553 requires this bit to be set if a message fails to pass the tests and the status word to be suppressed (NOT to be transmitted). This means that **all messages that are NOT error tree will NOT have a responding status word.** This allows the bus controller to timeout on the no status response, thus alerting the bus controller of a failure condition. **To obtain the status word with the error bit set requires a transmit status mode code or transmit last command word mode code to the terminal to retrieve the untransmitted status word (see paragraph 1.4.2).** The only exception is the illegal command option (see paragraph 1.6.3). The instrumentation, service request, broadcast command received, busy, subsystem flag, dynamic bus controller acceptance, and terminal flag bits are all optional and are discussed in paragraph 1.4.1. Notice 2 tightens the requirements for remote terminals employing broadcast recognition, capability of dynamic bus control, RT built-in-test, or subsystem built-in-test, by requiring the use of the bits in the status word associated with these functions. Bit positions 12 through 14 are reserved for future use and must be transmitted as zeros. To obtain usage of these bits requires DoD approval (no approval has been given as of 7/87). The last bit of the status word is the odd parity bit, which is calculated in a similar manner to all other parity bits.

1.4 Options Within the Standard

Since the standard covers a wide variety of designs, flexibility has been achieved without loss of interface compatibility, by allowing options to be selected by the user. **If an option is selected, it MUST operate per the standard.** If an option is not used in the design, it must follow the standard's selected method (e.g., set a bit to zero if unused). **No alternative or options other than those specified by the standard are allowed.** Notice 2 was developed to define which options of the standard are required to enhance tri-service interoperability and to further define some of the open-ended timing variables implied within the standard. As stated previously, options are available in the following areas; status word bits, mode codes, data bus redundancy, and coupling techniques.

1.4.1 Status bits

The optional status bits are; instrumentation, service request, broadcast command received, busy, subsystem flag, dynamic bus control acceptance and terminal flag.

The instrumentation bit in the status field Is set to distinguish the status word from the command word. Since the sync field is used to distinguish the command and status words from a data word, a mechanism to distinguish command and status word is provided by the instrumentation bit. By setting this bit to logic zero in the status word for all conditions and setting the same bit position in the command word to a logic one, the command and status words are identifiable. If used, this approach reduces the possible subaddresses in the command word to 15 and requires subaddress 31 (11111) to be used to identify mode commands (both 11111 and 00000 are allowed). If the instrumentation bit is not used, the bit will remain set to logic zero in the status word for all conditions.

The service request bit is provided to indicate to the active bus controller that the remote terminal is requesting service. When this bit in the status word is set to logic one, the active bus controller may take a predetermined action (if only one action can occur) or use the transmit vector word mode command to identify the specified request. The message format for acquiring this is discussed under transmit vector word mode command (below).

For terminals implementing the broadcast option, the broadcast command received bit is set to logic one when the proceeding valid command word was a broadcast command (remote terminals address 31). Since the broadcast message formats require the receiving remote terminals to suppress their status responses, the broadcast com-

mand receive bit is set to identify that the message was received error free. To allow the bus controller to examine the status word requires
the use of the transmit status word mode code or the transmit last command word mode code. The broadcast command received bit will be reset, when the next valid command occurs if it is NOT one of two mode codes; transmit status word or transmit last command word. Therefore, **to analyze the status word after a broadcast message has occurred requires a mode code message to a unique terminal.** The mode code message must be transmitted before any other message transmission to that unique terminal in order to retrieve to proper status word.

The busy bit in the status word is set to logic one to Indicate to the active bus controller that the remote terminal is unable to move data to or from the subsystem in compliance with the bus controller's command. A busy condition can exist within a remote terminal at any time causing it to be non-responsive to a command to send data or to receive data. This condition can exist for all message formats. In each case, except the broadcast message formats, the active bus controller will determine the busy condition immediately upon status response. In the case of the broadcast message formats, this information will not be known unless the receiving terminals are analyzed using transmit status mode code after the broadcast message. If the status word has the broadcast received bit set, the message was received and the terminal was not busy. Notice 2 to the standard discourages the use and existence of busy conditions, as they affect the overall communications flow on the bus, add overhead to the bus controller, and may have adverse effects upon data latency requirements within time critical systems (i.e., flight controls). A busy condition must only occur as the results of a particular command or message received by the terminal and NOT due to an internal periodic function. Thus for a non-failed terminal, the bus controller, with prior knowledge of the RT's characteristics, can determine what actions (commands/messages) will cause an RT to become busy, thus preventing any unnecessary busy conditions.

The subsystem flag bit is provided to Indicate to the active bus controller that a subsystem fault condition exists and that data being requested from the subsystem may be invalid. The subsystem flag may be set in any transmitted status word. If more than one subsystem is interfaced to the remote terminal the subsystem flag is the ORed results of all subsystems. The only method available

in 1553B to determine subsystem(s) health is via a normal message. The use of the subsystem flag bit requires considerable system control philosophy. If upon receiving the bit set, the bus controller is to do anything other than stop all communication with the terminal, a detailed protocol is required.

The system protocol must define the message (NOT a mode code) that the bus controller uses to poll the subsystem to determine the reason the bit was set. This polling will provide system application software the knowledge to determine the availability and status of the unit(s). If the unit(s) are usable, the subsystem flag bit must be cleared so that troubleshooting analysis does not occur, until another failure occurs. This can be accomplished using additional messages or upon transmission of the first message. This process usually requires several transmissions during which time the remote terminal is NOT a part of the normal periodic data bus traffic. Obviously user subsystems must deal with this temporary situation; permanent loss of this data. Mode commands can not be used to acquire subsystem built-in-test results.

The next bit in the status word is provided to Indicate the acceptance of the bus controller's offer to become the next bus controller. The offer of bus control occurs when the presently active bus controller has completed its established message list and issues a dynamic bus control mode command to the remote terminal that is to be the next potential controller. To accept the offer the potential bus controller sets its dynamic bus control acceptance bit in the status word and transmits the status word. The establishment of which controller should be the next potential controller will be discussed in the system design section (see paragraph 3.2). For Air Force applications, Notice 2 prohibits the bus controller from issuing a dynamic bus mode code, therefore this bit would always be set to zero.

The terminal flag bit is set to a logic one to Indicate a fault within the remote terminal. This bit is used in conjunction with the three mode code commands, inhibit T/F flag, override inhibit T/F flag, and transmit BIT word. The first two mode code commands deactivate and activate the functional operation of the bit. The transmit BIT word mode code command is used to acquire more detailed information about the terminal's failure. Most MIL-STD-1553B RT chip sets provide BIT word responses indicating the health of the chip. Notice 2 requires implementation of this bit within the status word if the remote terminal is capable of any form of self test (including what may be performed internal to the various chip sets such as

1.4.2 Mode Codes

The basic philosophy of the information transfer system is that it operates as a transparent communication link. "Transparent" means that an application's function does not need to be involved with the management of communication. Obviously, the information transfer system requires management that introduces overhead in the data bus system. The command words, status words, status word response gaps, and intermessage gaps are the overhead. Within the command word the **mode codes provide data bus management capability.** The mode codes (see table I-1.1) have been divided into two groups; mode codes without a data word (00000-01111) and mode codes with a data word (10000-11111). The use of bit 15 in the command word to identify the two types was provided to aid in the decoding process. Also, the use of a single data word instead of multiple data words was adopted to

data wraparound and data comparisons).

simplify the mode circuitry.

Generally, with these two types of mode commands, all data bus system management requirements can be met. Additional overhead is required by the system to maintain RT health, system time control (synchronization), subaddress message mapping, aperiodic message control, initialization/shutdown messages, etc. The determination of whether the command word contains a mode code is accomplished by decoding the subaddress/mode field (bit times 10-14). This field being either all zero's [00000] or all one's [11111] indicates that the command is a mode code and that the word count/mode code field (bit times 1519) contain the mode code type. Notice 2 requires that terminals must decode both indicators and that they must not convey different information. (Some earlier designs had used the [00000] indicator for the terminal hardware and the [11111] indicator for subsystem hardware.

Table I-1.1 Assigned Mode Codes

Transmit-receive bit	Mode code	Function	Associated data word	Broadcast command allowed
1	00000	Dynamic bus control	No	No
1	00001	Synchronize	No	Yes
1	00010	Transmit status word	No	No
1	00011	Initiate self-test	No	Yes
1	00100	Transmitter shutdown	No	Yes
1	00101	Override transmitter shutdown	No	Yes
1	00110	Inhibit terminal flag bit	No	Yes
1	00111	Override inhibit terminal flag bit	No	Yes
1	01000	Reset remote terminal	No	Yes
1	01001 ↓ 01111	Reserved ↓ Reserved	No ↓ No	TBD TBD
1	10000	Transmit vector word	Yes	No
0	10001	Synchronize	Yes	Yes
1	10010	Transmit last command	Yes	No
1	10011	Transmit bit word	Yes	No
0	10100	Selected transmitter shutdown	Yes	Yes
0	10101	Override selected transmitter shutdown	Yes	Yes
1 or 0	10110 ↓ 11111	Reserved ↓ Reserved	Yes ↓ Yes	TBD TBD

Note: TBD — to be determined.

Notice 2 also permits the broadcasting of mode codes. While this can reduce bus controller overhead in the areas of frame synchronization, it can have adverse effects upon the system if inadvertently issued (e.g., broadcast of a reset mode code). The system's designer is cautioned to research the implications of the use of the broadcast mode code commands before implementation.

There is no particular reason for the numerical assignment of the mode codes, except for dynamic bus control [00000], which was previously defined in 1553A. The separation of mode commands into two categories (with and without data words) is important to allow for controlled expansion of the standard. By controlling the mode code command number and its definition, commonalty between various terminals can be maintained. **All undefined 1unused] mode codes are considered Illegal commands** (see paragraph 1.6.3 for discussion of illegal command protocol). Notice 2 requires that all remote terminals must implement the following mode codes as a minimum: a) transmit status word, b) transmitter shutdown, c) override transmitter shutdown, and d) reset remote terminal. The bus controller must have the capability of implementing all mode code commands. The message formats associated with mode commands are shown in figure I-1.1.

The dynamic bus control mode code [00000] is provided to allow the active bus controller a mechanism (using the information transfer system message formats) to offer a potential bus controller (operating as a remote terminal) control of the data bus. Only the single receiver command (unique address) is allowed to be issued by the active bus controller. The response to this offering of the bus controller is provided by the receiving remote terminal, using the dynamic bus control acceptance bit in the status word. Rejection of this request by the remote terminal requires that the presently active bus controller must continue offering control to other potential controllers, the same controller, or remain in control. When a remote terminal accepts control of the data bus system by setting the dynamic bus control acceptance bit in the status word, control is relinquished by the presently active bus controller and the potential bus controller can then begin bus control. Note that Notice 2 prohibits the bus controller from ever issuing this mode command for Air Force applications.

Two mode codes are used to **synchronize** the data bus system; synchronize without a data word and synchronize with a data word. Synchronization informs the terminal(s) of an event time, to allow coordination between the active bus controller and terminals. Synchronization information may be implicit in the command word [mode code 00001] or in the data word [mode code 10001] following the command word. If a data word is used, the definition of the bits are the responsibility of the system designer and may contain system information such as timing data, data mapping pointers, etc. Paragraph 3.6 provides a system discussion of the use of the synchronize with data word mode code for remote terminal synchronization and subaddress mapping within a remote terminal for both periodic, aperiodic, and time critical messages.

The **status word** associated with the transmit status word mode code [00010] is identical in format to the status word transmitted with every error free message. However, this is one of two mode codes, which do not change the state of the status word. Therefore, the status word returned with this mode code represents the status word of the previous message NOT the status of the mode code message. Paragraph 5.2 provides a discussion on the affects of back to back transmission of this mode code. This mode code gives the bus controller the ability to analyze problems associated with the previous message. Using this approach, a bus controller can organize the communication with a terminal to obtain error analysis data.

The initiate self-test mode command [00011] is provided to initiate built-in-test (BIT) circuitry within a remote terminal. The mode code is usually followed, after sufficient time for test completion, by a **transmit BIT word mode command [10011]** yielding the results of the test. Notice 2 specifies that the RT receiving this mode code must complete its test and have the results ready within 100 milliseconds. The purpose of establishing an upper limit to the amount of time required to perform a reset is that the bus controller, with this prior knowledge, is aware of the maximum amount of time that the controller will be off-line" or busy, and may cease further communications with this terminal until such time has elapsed. This prevents the bus controller's software from continuously being vectored to error handling routines. The message formats provided for the initiate self-test mode command allow for both individual requests and multiple requests (broadcast). Notice that **the initiate self-test mode command is associated with the 1553 multiplex hardware only** and NOT with the interfacing subsystem. Paragraph 5.2 discusses the problem of

separating subsystem self test and health status messages from RT tests in terminals with embedded RTs.

The transmit BIT word mode command [10011] provides the BIT results available from a terminal, as well as the status word. Typical BIT word information for both embedded and stand-alone remote terminals include encoder-decoder failures, analog transmitter/receiver failures, terminal control circuitry failures, power failures, subsystem interface failures, and protocol errors (e.g., parity, Manchester, word count, status word errors, and status word exceptions). The internal contents of the BIT data word are provided to supplement the appropriate bits already available via the status word. Notice that the transmit BIT word within the remote terminal "**. . . shall not be altered by the reception of a transmit last command or transmit status word mode code**" received by the terminal. This allows error handling and recovery procedures to be used without changing the error data recorded in this word. **However, the RT will only save the last command and the status code field [of the status word] if the mode code transmit last command or transmit status word are transmitted.** Broadcasting of this command by the bus controller is not allowed. Another point, which needs to be mentioned again, is that **the function of transmitting RT BIT data ". . . shall not be used to convey BIT data from the associated subsystem[s]."** See discussion on subsystem flag bit in paragraph 1.4.1.

Four mode code commands are provided to control the transmitters associated with terminals in a system. These commands can be sent to a single receiver or broadcasted to multiple users. **The transmitter shutdown mode code [00100]** is used in a dual-redundant bus structure, where the command causes the transmitter associated with the other bus to terminate transmissions. No data word is required for this mode command. **The overrIde transmitter shutdown mode code [00101]** is used in a dual-redundant bus structure where a previously disabled transmitter is enabled. No data word is provided for this mode code.

The selected transmitter shutdown mode code [10100] is used in a multiple (greater than two) redundant bus structure where the command causes the selected transmitter to terminate transmissions on its bus. A data word is used to identify the selected data bus. **The override selected transmitter shutdown mode code [10101]** is used in a multiple (greater than two) redundant bus structure where the command allows the selected transmitter to transmit on its bus when commanded. The format of the data word associated with these two mode commands is NOT controlled by the standard and must be defined by the systems designer. Another method of overriding the transmitter shutdown mode codes [00100 and 10100] is to issue a reset mode code to the terminal.

Note that the standard or Notice 2 does not imply that issuance of either the transmitter shutdown or the selected transmitter shutdown mode codes will cause the associated receiver to also be shutdown. If the receiver does not "shutdown," then the system's designer should be aware that the terminal may still be receiving and processing data (on the shutdown bus). Therefore, the bus controller should remove the terminal from the active periodic communications list.

The inhibit terminal flag mode code [00110] **is used to set the terminal flag bit to zero in the status word.** When issued, the status word indicates an UNFAILED condition regardless of the actual failure state of the terminal. **This mode code is primarily used to prevent continued Interrupts and error handling analysis when the failure has been noted and the system reconfigured as required. Commanding this mode code prevents further failures from being reported, which normally would be reported using the terminal flag in each subsequent status word response. The message format associated with this mode code allows for both single and multiple receivers to be commanded. No data word is required with this mode code.** Note that the terminal flag, which is used to indicate an RT fault condition is limited to the remote terminal NOT any subsystem faults. (See the previous discussion on initiate self test and transmit BIT word mode codes.)

The override inhibit terminal flag mode command [00110] **negates the inhibit function thus allowing the T/F flag bit In the status response to report the present condition of the terminal. This mode code can be transmitted by the active bus controller to both single and multiple receivers. There Is no data word associated with this mode code.** This mode code could be used to analyze all the faults that exist in a remote terminal, since the inhibit terminal flag mode code will mask faults that have occurred since its implementation.

The reset remote terminal mode code [01000]

causes the addressed terminal(s) to reset themselves to a power-on initialized state. This mode code may be transmitted to an individual remote terminal or to multiple remote terminals (broadcast). Notice 2 requires that all terminals receiving this command shall complete their reset function within 5 milliseconds following the transmission of their status word (non-broadcast message). This mode code provides a means to "start over" at a known point. It may be the last thing the error handling software tries before giving up on a terminal.

The transmit vector word mode code [10000] is associated with the service request bit in the status word and is used to determine specific service being required by the terminal. The service request bit and the transmit vector word mode command provide the only means available for the terminal to request the scheduling of an asynchronous message, if more than one service request exists per terminal. If a remote terminal contains only a single service request, the bus controller may be "smart enough" to perform the request without the transmit vector mode code data word to point to the location of the commands within the bus controller to perform the service. The message format for this single receiver operation contains a data word associated with the terminal's response. Figure I-1.4 illustrates the message formats associated with this mode command. **Since a service request Is handled at the convenience of the bus controller and NOT the remote terminal's, a remote terminal design feature Is required to queue multiple service requests until they can be serviced by the bus controller.** The transmission of the transmit vector word mode code by the bus controller does not necessarily release the remote terminal to set the service request bit for a new request. There are several methods that can be used to resolve this handshake problem. Whatever the solution chosen by the systems designer, it should be applied to all terminals in the system. Thus, it's a system design issue, which should be addressed in the multiplex system specification. Solutions that are acceptable to most designers are; after transmission of the transmit vector word code allow new service request codes, and use of the synchronize without data word mode code command to release the RT to set a new service request. However, the first method has the potential of losing a critical request if the transmit vector word is not received properly by the bus controller. To counter this problem, previous designs have used the second method to release the service

request queue. The transmission of the synchronize without data word mode code after proper receipt of the transmit vector mode code frees the terminal's service request bit at the expense of an extra transmission.

The transmit last command [1001 0] is used in the error handling and recovery process to determine the last valid command received by the terminal, except for this mode code. The message format contains the previous status word and a data word. **The data word contains the previous 16 bits of the last valid command word received.** Notice that this mode command will not alter the state of the receiving terminal's status word. This fact allows this mode command to be used in error handling and recovery operations without affecting the status word, which could contain added error information. Some hardware is designed such that both the last command word and the status word are made available to the error handling software when the transmit last command mode code is received by the bus controller.

Each of the mode code types (with and without data words) have several **unused mode codes that are reserved for future use and cannot be used without the permission of the Office of Prime Responsibility (OPR), which Is the USAF.** See paragraph 1.6.3 fore discussion of illegal command protocol, that could be used if a remote terminal received a reserved mode code.

1.5 Data Bus Network Considerations

One of the most significant considerations facing the data bus system designer and integrator is the definition of the data bus network. The bus network must be designed for signal integrity to achieve the bit error and word error rate performance required by the standard. In addition, fault isolation and redundancy must be considered to allow the design to meet the required system reliability. It is important to note that one of the reasons for the requirements of dual redundant standby systems is system survival in case of battle damage to the communications media. The physical layout of the media, including the separation of cabling, location of couplers' length of stubs, and the connection of the media to the terminals are of importance not only from a maintainability point of view, but also regarding system operation, survivability, fault isolation, and systems security. The following discussion provides a summary of MIL-STD-1553B requirements that have significant effects on the design.

Single Receiver Only — Bus Controller to Remote Terminal*

| COMMAND WORD | DATA WORD | DATA WORD | ★ | STATUS RESPONSE | ☐ CONTINUED BELOW |

Source: bus controller
Address: unique ≠ 31
Subaddress: unique ≠ 31 and 32
Data word count: 1-32
T/R receiver

Source: single receiver
Service request bit set to 1

STATUS WORD

| SYNC | REMOTE TERMINAL ACCESS (5) | MESSAGE ERROR (1) | INSTRUMENTATION (1) | SERVICE REQUEST (1) | RESERVED (3) | BROADCAST COMMAND RECEIVED (1) | BUSY (1) | SUBSYSTEM FLAG (1) | DYNAMIC BUS CONTROL ACCEPTANCE (1) | TERMINAL FLAG (1) | PARITY (1) |

(bit positions: 3, 8, 11, 14, 20)

| MODE CODE COMMAND | ★ | STATUS RESPONSE | DATA WORD | ☐ CONTINUED BELOW |

Source: bus controller
Address: unique address of S/R status response
Subaddress: 31 or 32
Mode code: 1000

Source: single receiver

Service vector information

Service request bit set to 0
(if no new service requests are present)

STATUS WORD

| SYNC | REMOTE TERMINAL ACCESS (5) | MESSAGE ERROR (1) | INSTRUMENTATION (1) | SERVICE REQUEST (1) | RESERVED (3) | BROADCAST COMMAND RECEIVED (1) | BUSY (1) | SUBSYSTEM FLAG (1) | DYNAMIC BUS CONTROL ACCEPTANCE (1) | TERMINAL FLAG (1) | PARITY (1) |

(bit positions: 3, 8, 11, 14, 20)

| COMMAND WORD | ★ | STATUS RESPONSE | DATA WORD | DATA WORD | •••☐ MASTER CONTROLLER RETURN TO SYNCHRONOUS BUS LIST |

Source: bus controller
Address: unique address of S/R status response
Subaddress: unique address of service vector information
T/R: transmit

Source: single receiver

Asynchronous information

* Service request bit can be set for transmittal in any status response (all control and data message formats)
★ Response time delay gap
☐ End of message delay or gap

Figure I-1.4 Transmit Vector Word Transfer Formats

1.5.1 MIL-STD-1553B Bus Network Requirements

Table I-1.2 is a summary listing of the data bus coupling requirements contained in 1553B. The characteristics of the twisted-shielded pair cable have been relaxed from 1553A to allow selection of cable types from a variety of manufactures. It has been shown that minor variations from the specified fed cable characteristics do not significantly affect the system performance. Today it is possible to obtain high quality cabling from various manufacturers which meet all the requirements of the standard. Notice 2 tightens the shielding requirements to 90.0 percent coverage for

Parameter	MIL-STD-1553B
Transmission line	
Cable type	Twisted-shielded pair
Capacitance (wire to wire)	30 pF/ft, maximum
Twist	Four per foot (0.33/in), minimum
Characteristic impedance, (Z_O)	70 to 85 ohms at 1.0 MHz
Attenuation	1.5 dB/100 ft at 1.0 MHz, maximum
Length of main bus	Not specified
Termination	Two ends terminated in resistors equal to $Z_O \pm 2\%$
Shielding	75% coverage minimum
Cable coupling	
Stub definition	Short stub < 1 ft Long stub > 1 to 20 feet (may be exceeded)
Coupler requirement	Direct coupled—short stub; transformer coupled—long stub (ref. fig. I-1.7)
Coupler transformer	
Turns ratio	1:1.41
Input impedance	3,000 ohms, minimum (75 kHz to 1.0 MHz)
Droop	20% maximum (250 kHz)
Overshoot and ringing	± 1.0V peak (250-kHz square wave with 100-ns maximum rise and fall time)
Common mode rejection	45.0 dB at 1.0 MHz
Fault protection	Resistor in series with each connection equal to $(0.75\ Z_O) + 2.0\%$ ohms
Stub voltage	1.0V to 14.0V p-p, I-1, minimum signal voltage (transformer coupled); 1.4V to 20.0V, p-p, I-1, minimum signal voltage (direct coupled)

the cable and 360 degree shielding with 75.0 percent coverage for connector junctions, cable terminations, and bus-stub junctions.

A great deal of concern is attributed to the cable network requirements including; bus length, coupling and stubbing. MIL-STD-1553B does not specify a maximum main bus length, because the cable length, number of terminals and length of stubs are all subject to trade-offs and must be considered in the design for reliable system operation. To help understand these problems a generalized multiplex bus network configuration is shown in figure I-1.5. The main bus is terminated at each end of the cable with the characteristic impedance to minimize reflections caused by transmission line mismatch. With no stubs attached, the main bus looks like an infinite length transmission line with no disturbing reflections. When the stubs are added to connect terminals the bus is loaded locally and a mismatch occurs, which can result in reflections. The degree of mismatch and resulting signal distortion is a function of the absolute impedance Z presented by the stub and terminal impedance. **To minimize signal distortion it is desirable to maintain a high stub reflected impedance into to the main bus.** At the same time the impedance needs to be low so that adequate signal power will be delivered to the receiver. A trade-off and compromise between these conflicting requirements is necessary to achieve the specified signal-to-noise ratio and system error rate performance. In addition to these trade-offs, careful consideration must be made in the determination of the type of connector used to connect the terminal to the bus. This consideration should include the required integrity, isolation, and shielding (Notice 2 requires 360 degrees of shielding with a minimum of 75%

coverage from the cable to the connector). Other issues, which need to be addressed, include the location and type of connector (concentric, twinax, etc.), and if the data bus connectors will be stand-alone or included with other signals in a multi-contact connector. As a minimum, the two buses (A and B) should be brought into the terminal via separate connectors for isolation purposes.

Two methods for coupling' a terminal to the main bus are defined in 1553B, transformer coupling and direct coupling (see figure I-1.6). Transformer coupling is usually used with long stubs (1 to 20 ft.) and requires a coupler box or in-line molded coupler, separate from the terminal, located at the junction of the main bus and stub. Direct coupling is usually limited to stubs of less than 1 ft. In transformer isolated stubs, fault isolation resistors are included to provide protection for the main bus in case of a short circuit in the stub or terminal. The transformer characteristics defined in 1553B and listed in table I-1.2 provides a compromise for the signal level and distortion characteristics delivered to the terminals. The transformer turns ratio (1 :1.41) provides beneficial impedance transformations for both terminal reception and transmission.

The advantages of transformer coupling are as follows:

(1) The 1:1.4 turns ratio increases the impedance of the stub and terminal by a factor of two, as seen by the main bus. This reduces the loading effects of the distributed capacitance of the stub cable. This is advantageous by reducing reflections on the bus and increasing the overall bus impedance.
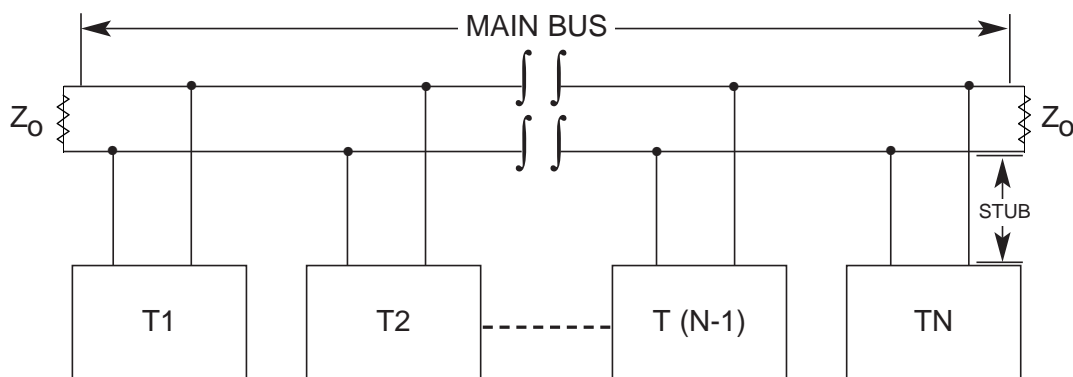
Figure I-1.5 Data Bus Network
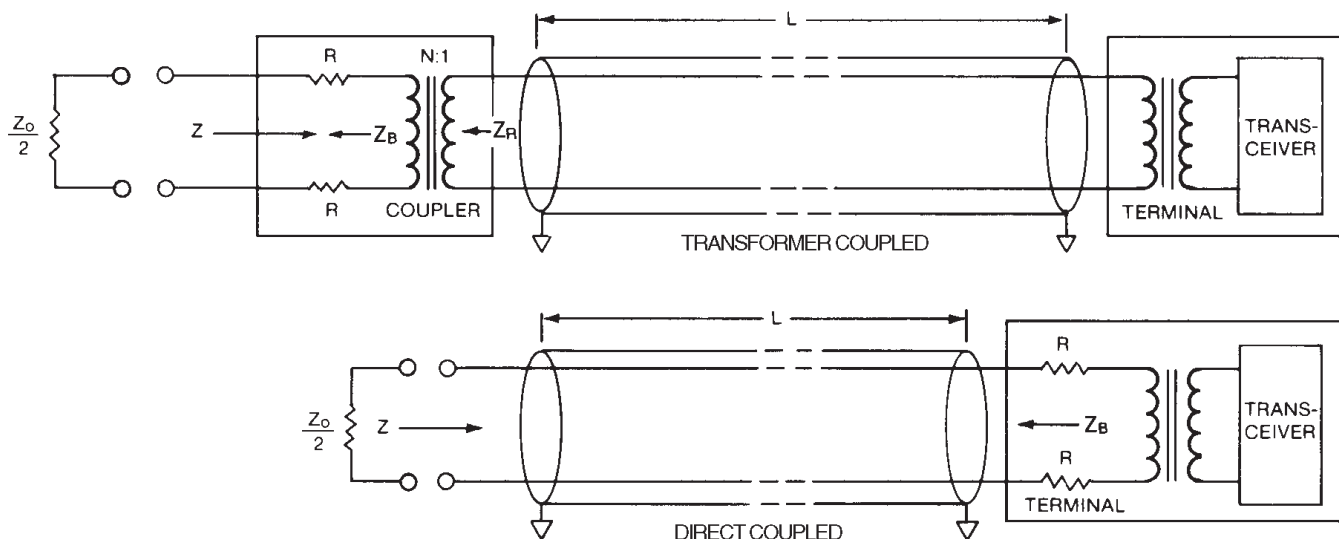
Figure I-1.6 Transformer Coupled and Direct Coupled Stubs

(2) For a direct coupled terminal, the impedance of the stub, as viewed from the secondary side of the terminal isolation transformer, is $2Z_O$. For a transformer coupled terminal, this impedance is Z_O. Allowing the terminal to drive into a matched impedance improves the terminal's transmitter waveform by minimizing reflections back to the terminal.

(3) Improved DC isolation.

(4) Improved common mode rejection:

Notice 2 states that for Navy applications, terminals shall have both transformers and direct coupled stubs available, whereas for Army and Air Force applications, only transformer coupled stubs are required.

1.5.2 Data Bus Network Analysis

A plot of the calculated first-order-magnitude stub absolute impedance Z versus stub length is presented in figure I-1.7. As indicated, the improvement of stub load impedance is a result of impedance transformation that is proportional to the square of the turns ratio, assuming an ideal transformer. The band of curves for the transformer-coupled case indicated by the darkened area between the curves results from assuming various values of transformer shunt impedance. The lower bound is the curve using a transformer with the minimum impedance specified in 1553B. The upper bound is for an ideal transformer with very high impedance. All values of stub impedance are magnitude values for a 70-ohm

cable with 30 pF/ft capacitance and are calculated for 1,000 ohms terminal input impedance, with the exception of the upper direct-coupled curve. This curve is based on the 1553B specified terminal input impedance of 2,000 ohms. It can be seen from these curves that stub impedance values are increased generally by use of the transformer, which provides at least a 2 to 1 improvement for the longer (greater than 10 ft) stubs. The curves also show the importance of the transformer characteristics for maintaining the expected improvement.

As indicated above, the 1:1.41 transformer also provides ideal termination of the stub for transmission of signals from the terminal to the main bus. Impedance at the main bus is:

$$Z_B = Z_O + 2R$$

where,

$$R = 0.75\ Z_O$$

$$Z_B = 0.5\ Z_O + 1.5\ Z_O = 2\ Z_O\ \text{ohms}$$

Z_O is the characteristic impedance of the data bus and Z_R is the reflected impedance from the bus to the stub. Therefore, the transformer impedance transformation is:

$$Z_R = \frac{Z_B}{(1.41)^2} = \frac{2Z_O}{2} = Z_O$$

Therefore, **the coupling transformer specified in 1553B provides the characteristics desired for reducing reflections and maintaining signal levels for systems where long stubs are required.**
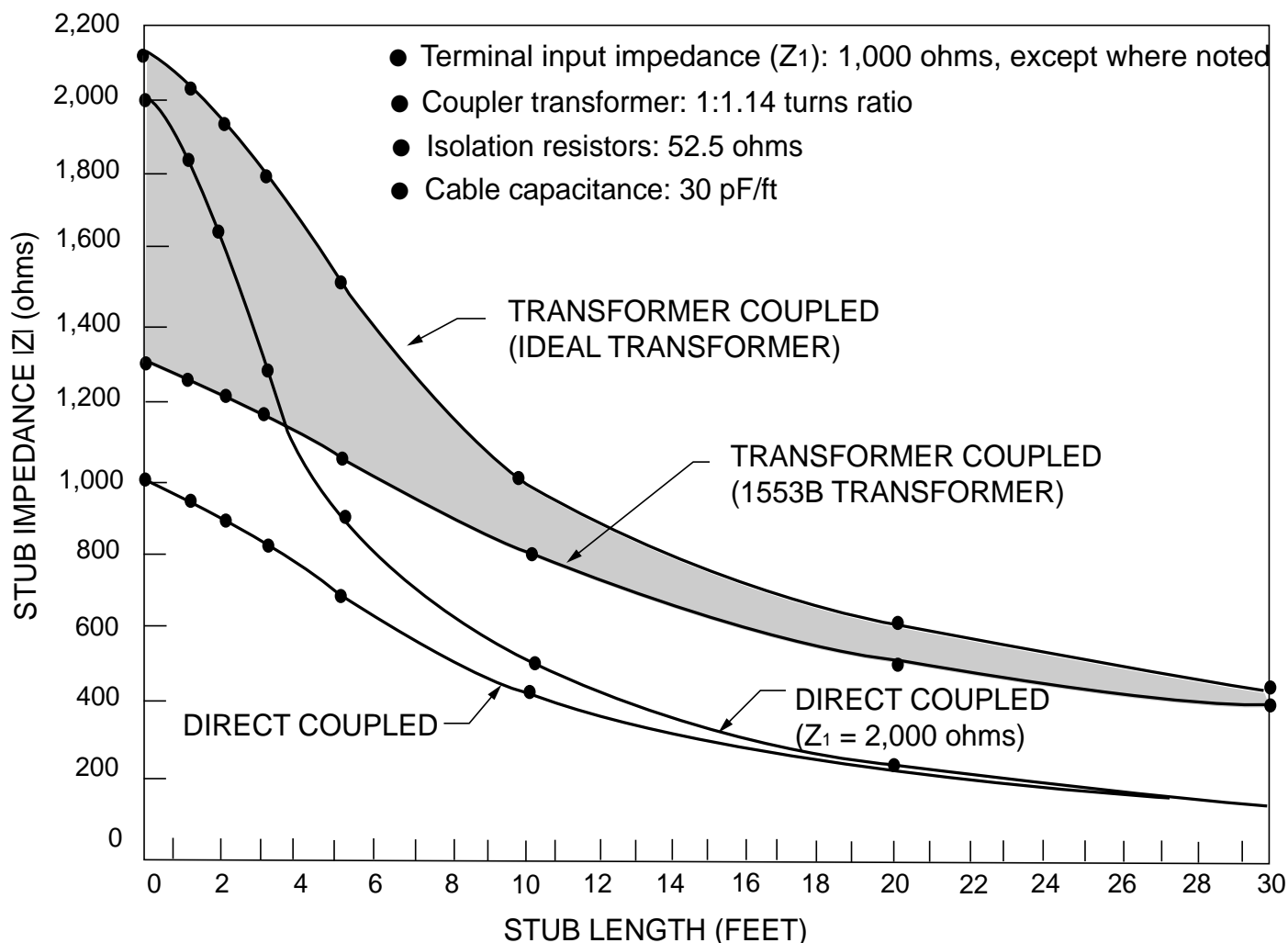
Figure I-1.7 Stub Impedance vs. Length

Direct coupling can be used for stub connections of 3 ft or less if terminal input impedance is maintained at the specified value:

Many configurations can be built reliably if careful attention is paid to the number, length and location of the stubs on the main bus. **It is highly desirable to test a proposed network using a computer simulation and a laboratory test setup.** The computer-generated data bus simulation provides more flexibility during the early design stages. The laboratory mockup with proper lengths of main bus and stubs is the final test of a good design.

1.6 The Most Frequently Asked Questions Concerning MIL-STD-1553

In an effort to help designers understand MIL-STD-1553, this section discusses some of the more frequently misunderstood portions of 1553 systems. Since 1553 is a standard and not a specification, it establishes requirements, which are often hard to

understand, and since the standard does not provide application guidance, questions arise. In order to solve this problem, the military has developed the "MIL-HDBK-1553 Multiplex Applications Handbook" to help convey the meaning of the standard and some lessons learned. In addition, industry and the government have jointly prepared a series of test plans, which help clear up interpretations of the standard by defining how a test will be performed and the expected results. This designer's guide has been prepared to highlight some of the significant aspects for the designer.

1.6.1 When to Transmit a Status Word [1553B paragraph 4.4.3.3 and 4.4.3.4]

The MIL-STD-1553 status word is a vital part of achieving the command/response protocol. **The transmission of a status word by a remote terminal indicates to the bus controller that the previous command or command and data was**

received without error. The suppression of the status word (status word not transmitted after reception of a command word) is used to indicate a message error exists if the message was NOT a broadcast message. Broadcast message formats require the status word to be suppressed to prevent a bus crash (multiple terminals transmitting simultaneously). Therefore, status words will always be transmitted for all error free, non-broadcast messages. Several types of errors can exist:

(1) **If a command word contains an error, the terminal rejects the message and resets,** and starts "looking" for a new valid command word with its unique terminal address or the broadcast address. No status word transmission should occur.

(2) **It a command word requests data from a subaddress, which is not operational for this terminal, the standard calls this failure an "Illegal command."** Two options are available to the terminal designers; a) respond with the normal message protocol required by the standard with data from a fixed location containing "don't care" data words, or b) design into the terminal the capability to recognize the failure and set the message error bit in the status word and transmit the status word only. **This is the only time when the statue word will ever be transmitted with the message error bit set as a response to a non-mode code (normal) message.** If the remote terminal design selects method a) above, usually a common area of 32 data words is set aside and all unused subaddress transmit messages are retrieved from there and all unused subaddress receive messages are deposited there. This technique is called "responding in form." Thus, a garbage-in/garbage-out buffer is provided for all unused subaddresses, which are NOT to be transmitted by or requested by the bus controller in the first place. **The standard also Identifies illegal commands as incorrect T/R bit setting and a word count that Is not allowed in the terminal.** Since the standard allows the designer the choice of implementing this analysis or not, most designers ignore monitoring for illegal commands (use method a) above rather than implement the analysis. These designers depend on the bus controller NOT transmitting illegal commands see paragraph 1.6.3 also). SEAFAC testing is normally done to the above b) option.

(3) If the command word is a broadcast address 31, the terminal analyzes the message, sets the proper bits in the status word and then suppresses the transmission of the status word.

(4) If the command word is correct and the message contains an error then, the data is rejected, the message error bit is set in the status word and status word is suppressed.

Therefore, **the suppression of the status word is used to indicate a broadcast message or an error condition.**

1.6.2 Superseding Valid Commands [1553B paragraph 4.4.3.2] and Reset Data Bus Transmitter [1553B paragraph 4.6.3.2]

The superseding valid command requirement in MIL-STD-1553B allows communication with a remote terminal that is presently working on a command. To better understand this requirement the data bus system redundancy needs to be considered. Both a single bus system and a dual standby redundant bus system (1553B paragraph 4.6.3) will be discussed.

MIL-STD-1553B states that a superseding command can not occur before time "T" (the response time between command and status words-maximum 12 microseconds). In any bus system, an error condition exists if there is no response in time "T." Communications can, therefore, be re-established with the remote terminal, if operational, by transmitting another command on the data bus that the previous command occurred on or any other data bus the terminal is connected to. **Using the superseding valid command requirement, the remote terminal is required to respond to the new command,** hopefully resolving the nonresponse to the previous command. This new command can be a normal message format or a mode code command.

In dual standby redundant systems, the "Reset Data Bus Transmitter" is a specific paragraph of 1553B (4.6.3.2), which provides specific requirements. . .**"If while operating on a command, a terminal receives another valid command, from either data bus, it shall reset and respond to the new command...."** Since another bus is available to the bus controller (i.e., the standby bus), the bus controller can transmit a new command to the remote terminal using the other bus. This allows the bus controller to identify messages containing a data word error, for example, to be retried on the alternate bus before the message is completed on the previous bus. It also allows the approach discussed in the paragraph above for any bus system to be used on either the active or standby bus to hopefully resolve the

condition of a non-responding (silent) terminal. Therefore, superseding commends and reset data bus transmitters provide an effective means of recovery. This error management (e.g., retrying or resolving a no response) can be initiated rapidly, thus reducing bus down time (bus inefficiency).

1.6.3 Illegal Command [1553B paragraph 4.4.3.41

As discussed in paragraph 1.6.1, concerning status word transmissions, illegal commands occur when; 1) an error exists in the T/R bit; 2) the subaddress/mode code field is an unused subaddress; or 3) an unimplemented mode code is transmitted. **The command word that contains an illegal condition must first meet all 1553B word validation requirements (1553B paragraph 4.4.1.1) before Its Illegal nature can be established.** If the command word does not meet these requirements, the remote terminal ignores the command. If the command word is correct, but has an illegal condition, the standard allows the designers two options: provide NO unique capability in the remote terminal to recognize the condition and respond normally as if there was NOT an illegal command, and respond with the message error bit set in the status word. The first approach is often referred to as "responding in form. This means that the remote terminal uses the normal message protocol response (figure 6 and 7 of 1553B) and the receiving device (bus controller or remote terminal for an RT-RT message) will obtain garbage" data. The second approach requires some analysis on the part of the remote terminal, but the response is straightforward. A message error has occurred and data should NOT be used. The message error alerts the bus controller of the problem. However, this is a system designer's problem, which can not be solved by the bus controller. So reporting it helps no one. **Not using the data or having the data cause a remote terminal problem is the key.** This must be solved regardless of the selection of the option to respond with a status word or respond in form as if no illegal command occurred. To understand how this command can cause remote terminal data problems each illegal command will be examined.

(1) The impact of receiving a transmit/receive (T/R) bit set to the inverse of data flow can cause internal confusion and errors within the remote terminal or the system. If the T/R bit indicates the terminal is to transmit data, but the bus controller sends data to the terminal a problem exists. **The remote terminal will recognize that data words are coming and the T/R is set to transmit and will NOT transmit.** However, the remote terminal must now

deal with the incoming data words. Most terminal designs without illegal command detection hardware will still map the data to a designated memory location. This must be considered when mapping data into and out of memory. If detection hardware is available, the status word message error bit will be set and the status word transmitted. When the inverted T/R bit says receive data and no data follows the command word, a real message error exists— to few words and the message error bit will be set in the status word and its transmission will depend on if the detection circuitry is present. If not, the status word will be suppressed.

(2) The reception of a message to an unused subaddress can cause internal remote terminal problems. Most remote terminals are designed to use mapping schemes, which map messages to memory using the T/R bit and the subaddress. Therefore, an illegal command could cause data to be written into areas of memory used for other functions. In the same way, improper data can be transmitted on the data bus by a remote terminal using an illegal command from an unused subaddress. This information could be anything and if used by a source could cause unknown responses. Therefore, as a minimum the designer should map all unused transmit and receive subaddresses to a common 32 word buffer that could be set to all zeros at initialization. Therefore, the first transmission of an unused subaddress should not affect anyone. After that if both receive and transmit commands arrive for unused subaddresses, the system could have problems. Obviously, additional memory space can be provided to separate receive and transmit unused subaddress memory areas to prevent bad data escaping the terminal. **Remember the bus controller is not allowed to transmit commands requesting or transmitting data from unused subaddresses of remote terminals.**

(3) Illegal commands, which request action associated with unused or unimplemented mode codes, must also be faced without causing remote terminal errors. These are more easily handled, because most remote terminals use chip sets which have selected when and how they will respond to each illegal mode condition. If you are interested in designing your own remote terminal, the "RT Validation Test Plan" (see Section VI) should be used as a reference guide in determining acceptable performance. **The general philosophy Is to "respond inform," in other words, according to the**

protocol If It was legal. This might mean bit bucketing data words or generating data words for transmission. If a remote terminal uses no mode codes (remember that Notice 2 requires at least four mode codes), subaddress codes 00000 and 11111 (the mode codes designator) is also not needed and it can be treated as an unused subaddress. Therefore, **even it a remote terminal designer chooses not to implement the option to analyze illegal commands, the design must be able to withstand the existence of these commands without causing the remote terminal internal errors or data bus system problems.**

1.6.4 Invalid Command [1553B paragraph 4.4.33]

An invalid command in 1553B occurs when the command word fails to meet the criteria established for the word validation by the standard (1553B paragraph 4.4.1.1). These tests for proper sync, valid Manchester II biphase data, and information format of 16 bits plus odd parity are basic to all 1553B words, but in the case of the command word they are essential. Since the protocol depends on a command/response action, remote terminals must only respond when "spoken to" (commanded to) by the bus controller. This is accomplished by the address validation (between the particular remote terminal's internal address and the address in the command word). Notice 2 requires that no single point failure in the address selection circuitry may cause a terminal to validate a false address. **To prevent multiple terminal reception or multiple terminals transmitting data, the command word must be correct.** Command word verification involves more than just using the remote terminal's address, it includes all of the 1553 required (4.4.3.5 in 1553B) word checks. Many remote terminal designers, in an effort to get a head start on handling a command word, examine only the address and fail to validate the remainder of the word before accepting it as a command.

1.6.5 Impact of Notice 2 to MIL-STD-1553B

Notice 2 to MIL-STD-1553B, issued 8 September 1986, was developed to define which options within the standard were required to enhance tri-service interoperability of systems. The Notice goes on to further define some of the open-ended timing constraints, which were undefined within the standard.

A notice to a standard is applied whenever the standard is referenced or specified. The notice does not apply to previous versions of the standard or applications to which the standard was applied prior to the release date of the notice. Waivers to the notice can be sought using the same technique allowed for the standard.

Notice 2 to MIL-STD-1553B supersedes Notice 1. The primary Notice 2 restrictions to the standard are as follows:

(1) *Broadcast Message Formats.*
Broadcast message formats are not restricted from being implemented in hardware. However, use of the broadcast command is limited to mode code commands only. This is a major departure from the Notice 1 restrictions, which prohibited broadcast messages altogether. The broadcast option for non-mode code messages may be designed into the remote terminal. If implemented, then the terminal MUST be capable of distinguishing between broadcast and nonbroadcast messages to the same subaddress.

(2) *Mode Codes and Mode Code Indicators.*
The terminal hardware may be designed with any or all mode codes, but the following mode codes must be implemented: transmit status word, transmitter shutdown, override transmitter shutdown, and reset remote terminal. Remote terminal's must be designed to recognize both 00000 and 11111 as a designator in the subaddress/mode code field of the command word. The two indicators must not convey different information. Also bus controllers must be designed to support all modes regardless of system usage.

For Air Force applications, the bus controller is prohibited from issuing a dynamic bus control mode code (one of the few exceptions to a total tri-service agreement).

Also timing constraints have been established for two of the mode codes: reset remote terminal, and initiate FT self test. For the reset remote terminal, the terminal must complete its reset function within 5 milliseconds after transmitting its status word to this command. For the initiate self test mode code, the terminal must complete its self test function within 100 milliseconds after transmitting its status word to this command. In both cases, while the terminal is performing the commanded function (reset or self test) the terminal may respond to a valid command by: a) no response on either bus, b) status word transmitted with the busy bit set; or c) normal response. However, any data transmitted as the result of this command MUST be valid data.

(3) *Status Word Bits.*

The only required status word bit is the message error bit. However, if the terminal employs broadcast recognition, capability of dynamic bus control, RT built-in-test, or subsystem built-in-test then the bits in the status word associated with these functions are also required.

The busy bits' use (due to existence of busy conditions within a terminal) is **strongly discouraged.** However if these conditions affect the terminal's ability to properly communicate via the bus, then the busy bit is to be used. Setting of the busy bit must be caused by the result of a command to the terminal. The busy bit may also be set as the result of a failure within the terminal or subsystem.

(4) *Message Formats and Subaddresses.*

Remote terminal must be capable of the following non-broadcast message formats: RT-BC, BC-RT RT-RT (receive and transmit), and mode codes commands without data words. Bus controllers must have the capability to transmit all message formats.

The remote terminal must provide a data wrap-around capability equal to the maximum number of data words it's capable of processing for any subaddress. The desired subaddress for this function is 30 [11110]. The purpose of this function is to provide the bus controller the capability to test the data flow through a terminal's front end (1553 hardware), initial subsystem interface (memory buffers), and the data bus media (cabling and bus couplers).

(5) *RT-RT Timeout.*

In addition to normal message validation criteria, the remote terminal must timeout (invalidate the message) if the receive command word is not followed by the first data word within 57 ± 3 microseconds. Although this poses few problems for current hardware designs and chip sets, some of the earlier designs would wait "forever" then take the data intended for some other terminal (the transmitting terminal's status word was ignored since it contained a command/status sync).

(6) *Electrical.*

The data bus cable characteristics have been tightened in the area of cable shielding (90%), connector/junction shielding (360 degrees at 75%), and impedance (actual to within the range of 70-85 ohms at 1 megahertz). In addition, the polarity for concentric connectors has been defined with the center pin being the high (positive) bus signal and the inter-ring being the low (negative) bus signal. The outer ring is obviously the bus shield.

(7) *Coupling Stubs.*

For Navy applications, terminals must have both transformer coupled stub and direct coupled stub connections externally available (either may be used). For Army and Air Force applications, only transformer coupled stubs are required.

The remote terminal must limit spurious output signals during power up or down sequences to ± 250 mV for transformer coupled stubs and ± 90 mV for direct coupled stubs.

1.6.6 Responses to Nonimplemented Mode Code Commands and Undefined Mode Codes.

Table 1 of 1553B and paragraph 4.3.3.5.1.7 define mode codes while paragraph 4.3.3.6.4 identify the two types of mode codes; without data word and with data word. Table 1 also identifies a group of mode codes as RESERVED. However, table 1 fails to identify all of the binary combinations that can exist in the T/R bit and five bit mode code field. Table I-1.3 expands the 1553B table to include these possibilities. As can be seen 22 of the 64 entries are undefined and are considered INVALID. Notice from the table that these DO NOT include the RESERVED mode codes.

Several terms must be clearly defined in order to proceed with an explanation of the handling of RESERVED mode codes or UNDEFINED mode codes. For convenience, the following definitions are recapitulated with reference to 1553B.

Valid Words. (paragraph 4.4.1.1) Every word must meet four tests of validity to be declared a valid word. These tests are:

a. Valid sync field
b. All bits are encoded as valid Manchester II code
c. The information field has 16 bits + 1 parity bit
d. The parity over the word is odd parity.

A word failing any of these tests would be considered to be invalid.

Invalid Command. (paragraph 4.4.3.3j A command that does not have an RT address field that matches the address of the receiving RT or that does not have an address of 11111 if the broadcast option is implemented in the RT, or that fails to satisfy the test for a valid word shall be considered to be an invalid command.

Invalid data reception. If any data word that is part of a BC to RT message fails the tests for being a valid word, or if the message transmission if not contigu-

Table I-1.3 MIL-STD-1553B Mode Command Organization

T/R	MODE CODE	DATA WORD ASSOCIATED	BROADCAST ALLOWED	RESERVED	INVALID
0	00000	UND	UND	UND	YES
0	00001	UND	UND	UND	YES
0	00010	UND	UND	UND	YES
0	00011	UND	UND	UND	YES
0	00100	UND	UND	UND	YES
0	00101	UND	UND	UND	YES
0	00110	UND	UND	UND	YES
0	00111	UND	UND	UND	YES
0	01000	UND	UND	UND	YES
0	01001	UND	UND	UND	YES
0	01010	UND	UND	UND	YES
0	01011	UND	UND	UND	YES
0	01100	UND	UND	UND	YES
0	01101	UND	UND	UND	YES
0	01110	UND	UND	UND	YES
0	01111	UND	UND	UND	YES
0	10000	UND	UND	UND	YES
0	10001	YES	YES	NO	NO
0	10010	UND	UND	UND	YES
0	10011	UND	UND	UND	YES
0	10100	YES	YES	NO	NO
0	10101	YES	YES	NO	NO
0	10110	YES	TBD	YES	NO
0	10111	YES	TBD	YES	NO
0	11000	YES	TBD	YES	NO
0	11001	YES	TBD	YES	NO
0	11010	YES	TBD	YES	NO
0	11011	YES	TBD	YES	NO
0	11100	YES	TBD	YES	NO
0	11101	YES	TBD	YES	NO
0	11110	YES	TBD	YES	NO
0	11111	YES	TBD	YES	NO
1	00000	NO	NO	NO	NO
1	00001	NO	YES	NO	NO
1	00010	NO	NO	NO	NO
1	00011	NO	YES	NO	NO
1	00100	NO	YES	NO	NO
1	00101	NO	YES	NO	NO
1	00110	NO	YES	NO	NO
1	00111	NO	YES	NO	NO
1	01000	NO	YES	NO	NO
1	01001	NO	TBD	YES	NO
1	01010	NO	TBD	YES	NO
1	01011	NO	TBD	YES	NO
1	01100	NO	TBD	YES	NO
1	01101	NO	TBD	YES	NO

Table I-1.3 MIL-STD-1553B Mode Command Organization (continued)

T/R	MODE CODE	DATA WORD ASSOCIATED	BROADCAST ALLOWED	RESERVED	INVALID
1	01110	NO	TBD	YES	NO
1	01111	NO	TBD	YES	NO
1	10000	YES	NO	NO	NO
1	10001	UND	TBD	UND	YES
1	10010	YES	NO	NO	NO
1	10011	YES	NO	NO	NO
1	10100	UND	TBD	UND	YES
1	10101	UND	TBD	UND	YES
1	10110	YES	TBD	YES	NO
1	10111	YES	TBD	YES	NO
1	11000	YES	TBD	YES	NO
1	11001	YES	TBD	YES	NO
1	11010	YES	TBD	YES	NO
1	11011	YES	TBD	YES	NO
1	11100	YES	TBD	YES	NO
1	11101	YES	TBD	YES	NO
1	11110	YES	TBD	YES	NO
1	11111	YES	TBD	YES	NO

TBD = "To be Determined" as indicated in 1553B
UND = Undefined, not specifically assigned in 1553B

ous, or if the number of data words received is not correct (does not match the command word count field), an invalid data reception has occurred.

Illegal commands. (paragraph 4.4.3.4) An illegal command is a valid command word, but which contains a combination of T/R bit, subaddress/mode field, and data word count/mode code field that has not been implemented in the receiving RT. Notwithstanding a side discussion as to which commands may be considered to be optional, the detection of illegal commands is itself optional.

Now, the rules for setting of the message error bit in the status word and when to transmit status or to suppress transmission of a status word will be reviewed.

Paragraph 4.3.3.5.3.3 states that the message error (ME) bit of a status word shall be set to indicate that:

1. One or more of the data words associated with the proceeding BC-to-RT command (receive command) was not a VALID WORD, or
2. The message transmission was not contiguous per paragraph 4.4.1.2, or
3. The command received was an ILLEGAL COMMAND and the RT has implemented the option for illegal command detection, or

4. There was an INVALID DATA RECEPTION.
Based on paragraphs 4.4.3.1 and 4.4.3.5, a status word response is required if the RT receives a command that is a valid command and if any associated data reception is a valid data reception. Paragraph 4.4.3.3 specifically states that a RT shall not respond to an INVALID COMMAND and paragraph 4.4.3.6 requires the status word response be suppressed for the case of INVALID DATA RECEPTION.

By definition, RESERVED mode codes are represented by command word formats that have been defined in table 1 of 1553B, but which are not to be implemented at present. Consequently, an RT design will not implement these T/R, subaddress/mode field, and data word/mode code combinations and if the RT receives such a combination, it must regard the command as an ILLEGAL COMMAND per the definition given in the standard. The proper action required, per the standard, is to set the ME bit of the status word and transmit the status word to the bus controller, if the option for illegal command monitoring is being observed.

Notice that there is no restriction on a bus controller not to transmit a RESERVED mode code command. RESERVED mode commands may be defined in the future, in which case receipt of that command would be

an absolutely normal occurrence. Simply treating the RESERVED mode commands as an ILLEGAL COMMAND is consistent with the potential for future expansion of mode commands since a pre-existing RT design should not implement the new command created from a RESERVED mode command. The "RT Validation Test Plan" (Notice 1 to MIL-HDBK-1553) supports this discussion.

The UNDEFINED mode code commands present a more difficult situation. If an RT receives an undefined mode command, by default it could treat this command as an ILLEGAL COMMAND from the standpoint of it's being a combination of T/R, subaddress/mode, and data word/mode code fields that definitely will not have been implemented by the RT design. But the undefined mode command can be regarded as an ILLEGAL COMMAND only if it is also a valid command. This point raises questions as to how such a command came to be received by the RT. **Since the undefined mode command is not a valid combination of T/R, subaddress/mode field, and data word/mode code field as defined in table 1 of 1553B, it must be presumed that a bus controller would never purposely send such a combination.** The case can be made that some failure mode has to have occurred in order for the command received by the RT to be interpreted as an undefined mode command. Invalid encoding, undetected multiple bit errors, or bus controller failures could be possible sources for the erroneous mode command. It may be surmised that the receipt of an undefined mode command implies that the command could be an INVALID COMMAND. It is also possible that all of the checks for validity of the command word genuinely pass, but that a failure mode in the bus controller has resulted in the transmission of an undefined mode command. As such, it would be proper to regard the command as an ILLEGAL COMMAND and to set the ME bit of the status word register and either transmit or suppress the status word depending on the implementation of the ILLEGAL COMMAND option.

The validity of the command determines whether or not a status response is to be sent. Due to the likelihood that the undefined command is in reality an INVALID COMMAND generated by undetected errors in encoding, **the safest approach to the decision concerning whether or not to transmit the status word would be to suppress transmission of the status word.**

So, depending upon the failure mode that produces an undefined mode command, attributes of either an ILLEGAL COMMAND or an INVALID COMMAND or both may be present. The required response is not clear since the prescribed responses are opposing when comparing an ILLEGAL COMMAND with an INVALID COMMAND. For this reason, this category of command is sometimes referred to an illogical command. By implementing the dominant responses for both an ILLEGAL COMMAND and for an INVALID COMMAND, the safest overall response results. **The ME bit of the status word is set to indicate an error and the transmission of the status word is suppressed to prevent violation of the command/response protocol for a potentially INVALID COMMAND.** Suppression of the transmission of the status word in response to the receipt of an undefined/illogical command has the added benefit of distinguishing the associated failure mode from the receipt of a reserved mode command. In any event, diagnosability of errors resulting from the receipt of an improper mode command can be improved by providing bits in the BIT word to expand the encoding of errors.

The "RT Validation Plan" allows two responses; a) treat the condition as an ILLEGAL COMMAND, or b) treat the condition as an INVALID COMMAND. In the first case, the terminal would respond with a status word with no errors set if it did not have ILLEGAL COMMAND detection capability. If the terminal did have ILLEGAL COMMAND detection capability, a status word with the ME bit set would be transmitted only. In the second case, if the INVALID COMMAND was detected, the status response would be suppressed and the ME bit set in the status word. This message error would NOT be recorded by the bus controller unless the next message was a transmit status word mode code or transmit last command word mode code.

1.6.7 Two Single-Channel Remote Terminal Operating in a Dual Standby Redundant System A few remote terminal designers have designed subsystems using the channel concept (see figure I-3.7 right side) only to discover that the terminal failed to meet all the 1553B requirements when used in a dual standby redundant data bus network (see 1553B paragraph 4.6.3). 1553B provides certain general capabilities, which are influenced when a system is operating in a dual standby redundant network, as most are, compared to a single channel network. Mode codes that convey the operational capability of the entire remote terminal as compared to that of the channel are the most affected; a)transmit status word, b)reset remote terminal, c) transmit last command word, and d) transmit BIT word. Transmit status word mode code provides the message error results of the last message processed by the terminal regardless of which bus (channel—active or standby)

the message was received on. The other bits in the status word represent the health of the entire terminal, not just the channel receiving the request. The reset remote terminal mode code also applies to the entire terminal and NOT just the channel the request was received on. Transmit last command word must come from a register, which contains the 16 bit contents of the last command received regardless of the bus received on. The same is true for the terminal's BIT word and vector word, it must reflect the terminal's state and not that of channel. Figure I-3.7 illustrates that too little redundancy is unacceptable just as well as to much isolation is unacceptable.

1.6.8 Testing of 1553 Terminals

MIL-STD-1553B terminal testing has been conducted by designers and the US Air Force SEAFAC Laboratories at Wright Patterson AFB. In the past, the proof that a terminal's design was acceptable, required it to pass the SEAFAC Validation Tests. Several years ago, the government recognized that a single laboratory could not test everyone's designs. With the increasing numbers and suppliers of 1553 embedded terminals and the extensive number of chip sets being developed, something had to be done to distribute the testing load while maintaining the highly successful design verification effort.

To this end, the SAE Avionics Systems Division and the government began developing a series of verification and production test plans and procedures, which could be conducted by independent laboratories to validate the design approach and provide a basis for production level testing of 1553 terminals. This approach would allow multiple industry sources to be used to verify compliance of a design and provide industry with ground rules for production testing. It was clear from the beginning that remote terminal test plans were required first, followed by bus controller test plans and then system data bus test plans. The initial remote terminal test plans grew out of the test procedures developed and used at SEAFAC. It was obvious from the start that the test plans would set a lower level of definition for "acceptable terminal performance" than existed in the standard. This caused extensive work and coordination between industry and the government. Today (7/87) these test plans are available in preliminary form from the SAE (Society of Automotive Engineering) under the following Aerospace Standard numbers:

AS4111 Rev. D 3/84	Validation Test Plan for Aircraft Internal Time Division Command/Response Multiplex Data Bus Remote Terminals. (See Section VI).
AS4112 Rev. 4/85	Production Test Plan for Aircraft Internal Time Division Command/Response Multiplex Data Bus Remote Terminals. (See Section V).
AS4113 Rev. F 3/86	Validation Test Plan for Aircraft Internal Time Division Command/Response Multiplex Data Bus Bus Controllers.
AS4114 Final Draft 2/87	Production Test Plan for Aircraft Internal Time Division Command/Response Multiplex Data Bus Bus Controllers.
AS4115 Final Draft 2/87	Test Plan for the Digital Internal Time Division Command/Response Multiplex Data Bus System.

Note that the Validation Test Plan for Remote Terminals (AS4111) has been adapted by the military and released as Notice 1 to MIL-HDBK-1553 Multiplex Applications Handbook.

In the near future, these documents will be approved by the SAE and published as Aerospace Standards. The military is considering placing some of these in future updates of MIL-HDBK-1553 Handbook. Some differences still exist between the SEAFAC test plans and procedures and the SAE documents, which must be resolved. In addition, with the release of Notice 2 to MIL-STD-1553B these documents will be updated by the SAE.

2.0 A COMPARISON OF DATA BUS SPECIFICATIONS

Today several MIL-STD-1553 data bus systems are in full use. To better understand the slight differences between these systems, this section consists of several tables comparing performance characteristics of MIL-STD-1553B and several aircraft specifications (e.g., A-10, F-16, F-18, and B-52). Most 1553 applications will require an aircraft specification in addition to MIL-STD-1553B to allow the aircraft designer a method of identifying the applicable options of the standard. Prior to MIL-STD-1553B these aircraft specifications provided the appropriate level of detail, since earlier versions of the standard were unclear in several critical design areas. Each of these parameters can be examined on a comparative basis using these tables.

A thorough paragraph-by-paragraph analysis of MIL-STD-1553B is given in Section II, where differences between A and B and the reasons for various requirements are discussed.

2.1 Summary of Data Bus Requirements

Table I-2.1 Summary of Data Bus Requirements

Applications	DoD Avionics
Data Rate	1 MHz
Word Length	20 bits
Number of data bits/word	16
Transmission technique	half-duplex
Operation	Asynchronous
Encoding	Manchester II biphase
Bus Coupling	Transformer
Bus Control	Single or multiple
Transmission media	Twisted pair shielded

2.2 Summary of Status Word Protocols Requirements

Table I-2.2 Summary of Status Word Protocol for Various Message Errors

Application

Error Condition	F-16*A/C	B-52OAS	A-10	F-18	OH-58D	1553B Notice 2
Data Parity Error	Transmit Status	Suppress Status	Transmit Status	Transmit Status	Suppress Status	Suppress Status
Invalid Data (invalid sync. code bit count)	Transmit Status	Suppress Status	Transmit Status	Transmit Status	Suppress Status	Suppress Status
No Data Received	Suppress Status	Suppress Status	Suppress Status	Suppress Status	Suppress Status	Suppress Status
Word Count Error	Suppress Status	Suppress Status	Suppress Status	Suppress Status	Suppress Status	Suppress Status

Note for a broadcast command, the status word is always suppressed. The broadcast command received bit shall not be set if a message error occurs for a broadcast message.

*F-16A/C message error bit set for data parity error only

2.3 Summary of Status Word Bit Assignments

Table I-2.3 Summary of Status Word Bit Assignments

Status Bits	B-52 OAS	A-10	F-18	F-16 A/C	OH-58D	1553B Notice 2	1553A
Sync	1-3	1-3	1-3	1-3	1-3	1-3	1-3
Terminal Address	4-8	4-8	4-8	4-8	4-8	4-8	4-8
Message Error	9	9	9	9	9	9	9
Instrumentation	NU	10	NU	10	NU	10	u
Data Quality*	NU	NU	NU	11†	NU	10	u
Service Request	15	NU	10-18‡	NU	11	11	u
Broadcast Received	NU	NU	NU	15	NU	15	u
Busy	16	NU	NU	NU	16	16	u
Subsystem Flag	10-14	NU	NU	NU	17	17	u
Dynamic Bus Control Acceptance	NU	NU	NU	NU	NU	18§	u
Dedicated Function Received*	NU	NU	NU	16	NU	NU	u
Bus Shutdown**	NU	17-18	NU	17-18	NU	NU	u
Reserved	17-18	11-16	NONE	12-14	12-14	12-14	u
Terminal Flag	19	19	19	19	19	19	19
Parity	20	20	20	20	20	20	20

Note: Bits which are not used shall be considered reserved per the application and sent to zero.

*   F-16 unique

‡   F-18 unique

**   F-18 and A-10 unique

u   Undefined

†   F-16 uses data parity same as message error

§   Not permitted in Air Force applications

NU Not Used

2.4 Summary of Mode Code Usage

Table I-2.4 Summary of Mode Code Usage

MODE COMMANDS	MODE CODES						
	B-52	A-10	F-18	F-16 A/C	OH-58D	1553B Notice 2	1553A
WITHOUT DATA							
Dynamic Bus Control	NU	NU	0	NU	NU	0	ND
Synchronize	NU	NU	NU	1	1	1	ND
Transmit Status	1	NU	NU	2-31	2	2	ND
Initiate Self Test	2	NU	NU	NU	3	3	ND
Transmitter Shutdown	3	NU	NU	NU	NU	4	ND
Override Shutdown	4	NU	NU	NU	NU	5	ND
Inhibit T/F	NU	NU	NU	NU	NU	6	ND
Override Inhibit T/F	NU	NU	NU	NU	NU	7	ND
Reset RT	NU	NU	NU	NU	NU	8	ND
DATA FROM TERMINAL							
Transmit Vector	NU	NU	NU	NU	NU	16	ND
Transmit Last Command	NU	NU	NU	NU	NU	18	ND
Transmit Bit Word	NU	NU	NU	NU	19	19	ND
DATA TO TERMINAL							
Synchronize	NU	NU	NU	NU	NU	17	ND
Selected Shutdown	NU	NU	NU	NU	NU	20	ND
Selected Override	NU	NU	NU	NU	NU	21	ND
ILLEGAL MODE CODES	0,5-31	ALL	1-31	NONE	9-15, 22-31	9-15, 22-31	ND

NU = Not Used

ND = Not Defined

2.5 Comparison of Data Bus Characteristics

Table I-2.5 Comparison of Data Bus Characteristics

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
1. Twisted, shielded, jacketed	Yes 4.2.4.1	Yes 4.5.1.1
2. Minimum cable shield coverage	80% 4.2.4.1	75% 4.5.1.1
3. Minimum cable twist	1 twist/in (12 twists/ft) 4.2.4.1	4 twists/ft 4.5.1.1
4. Wire-to-wire distributed capacitance (maximum)	30 pF/ft 4.2.4.1	30 pF/ft 4.5.1.1
5. Characteristic impedance of cable	70 ± 10% at 1 MHz 4.2.4.2	Nominal 70 to 80 at 1MHz 4.5.1.2
6. Cable attenuation	1 dB/100 ft at 1 MHz 4.2.4.3	1.5 dB/100 ft at 1 MHz 4.5.1.3
7. Cable length	300 ft maximum 4.2.4.4	Unspecified —
8. Cable termination using a resistance at both ends	Characteristic impedance 4.2.4.6	Nominal characteristic impedance ± 2% 4.5.1.4
9. Cable stubbing	Transformer coupling for stubs longer than 1 ft but less than 20 ft; direct coupling if stub is less than 1 ft; maximum stub length of 20 ft 4.2.4.5 Figure II-14 page II-41	Transformer coupling or direct coupling allowed; maximum stub length suggested 20 ft 4.5.1.5.1 or 4.5.1.5.2
10. Cable coupling (connector)	Compatible with Amphenol type 31-235 or Trompeter type TEI-14949-E137 receptacles and Amphenol type 31-224 or Trompeter type TEI-14949-PL36 plugs 4.2.4.6	Figures 9 or 10 of 1553B Unspecified —

Table I-2.5 Comparison of Data Bus Characteristics (Continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
11. Cable Coupling shielded box	Shielded coupler box 4.2.4.6	75% coverage, minimum 4.5.1.5.1.3 and 4.5.1.5.2.2
12. Coupling transformer turns ratio	Unspecified —	1:141 ± 3% higher turns on isolation resistor side of stub 4.5.1.5.1.1
13. Transformer open circuit impedance	Unspecified —	3,000 ohms over frequency of 75 kHz -1MHz with 1V RMS sine wave 4.5.1.5.1.1
14. Transformer waveform integrity	Unspecified —	Droop not to exceed 20% overshoot and ringing less than ± 1V peak under test of figure 11 of 1553B 4.5.1.5.1.1.2
15. Transformer common mode rejection	Unspecified —	45 dB at 1MHz 4.5.1.5.1.1.3
16. Fault isolation — Isolation resistor in series with data bus cable (coupler)	$R = 0.75Z_O$* ± 5% 4.2.5.2	$R = 0.75 Z_O$* ± 2% 4.5.1.5.1.2
Direct coupled case with the isolation resistor in the RT	Figure II-14 page II-41	R = 55 ohms ± 2% 4.5.1.5.2.1 Figure 10 of 1553B
17. Impedance across the data bus for any failure of coupling transformer, cable stub, or terminal receiver and transmitter transformer coupling	No less than 1.5 Z_O * 4.2.5.2	No less than 1.5 Z_O * 4.5.1.5.1.2
Direct coupling		No less than 110 ohms 4.5.1.5.2.3
18. Stub voltage requirements and input level transformer coupling	**Range of the 0.5V to 10V peak; 1.0V to 20V p-p, I-I 4.2.5.4.1 Figure II-14 page II-41	**Range of the 1.0V to 14.0V p-p**, I-I with one fault as stated in 17 above 4.5.1.5.1.4 Figure 9 of 1553B

a*Z_O = cable normal characteristic
** Assumes one fault of a coupling transformer, cable stub, or terminal receiver or transmitter

DESIGNER'S NOTES

Table I-2.5 Comparison of Data Bus Characteristics (Continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
Direct Coupling	**Range of the 0.5V to 10V peak; 1.0V to 20 V p-p, I-I 4.2.5.4.1 Figure II-14 page II-41	**Range of 1.4V to 20V p-p, I-I with one fault as stated in 17 above 4.5.1.5.2.3 Figure 10 of 1553B
19. Wiring and cabling for electromagnetic capability	MIL-E-6051 4.2.4.7	MIL-E-6051 4.5.1.5.3

** Assumes one fault of a coupling transformer, cable stub, or terminal receiver or transmitter

2.6 Comparison of Terminal Characteristics

Table I-2.6 Comparison of Terminal Characteristics

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
1. Output level — transformer coupling	± 3.0V to ± 10V peak (6.0V to 20.0V p-p) I-I with no faults; with one fault of a coupling transformer, cable stub, or terminal receiver-transmitter, ± 2.25V to ± 11.25V peak (4.5V to 15V p-p) I-I	With RL =70 ± 2%, 18.0V to 27.0V p-p, I-I
	4.2.5.3.1	4.5.2.1.1.1 Figure 12 of 1553B
Direct coupling		With RL = 35 ± 2%, 6.0V to 9.0V p-p, I-I 4.5.2.2.1.1 Figure 12 of 1553B
2. Output waveform — Zero crossing deviation	± 25 ns 4.2.5.3.2 Point C, figure II-14 page II-41	± 25 ns 4.5.2.1.1.2 Figure 12 of 1553B
Rise and fall time (10% to 90%)	100 ns 4.2.5.3.2 Figure 13 of 1553B	100 to 300 ns 4.5.2.1.1.2 Figure 13 of 1553B
Transformer coupling distortion (including overshoot and ringing)	Unspecified	± 900-mV peak, I-I 4.5.2.2.1.2 Point A, figure 12 of 1553B
Direct coupling distortion (including overshoot and ringing)	Unspecified	± 300-mV peak, 1-1 4.5.2.1.1.2 Point A, figure 12 of 1553B
3. Output noise — Transformer	10-mV p-p, I-I 4.2.5.3.3 Point A, figure II-14 page II-41	14-mV, RMS, I-I 4.5.2.1.1.3 Point A, figure 12 of 1553B

Table I-2.6 Comparison of Terminal Characteristics (Continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
Direct coupling		5-mV, RMS, I-I 4.5.2.2.1.3 Point A, figure 12 of 1553B
4. Output symmetry (after 2.5 us of midbit crossing the last parity bit — Transformer coupling	Unspecified	± 250-mV peak, I-I 4.5.2.1.1.4 Point A, figure 12 of 1553B
Directed Coupling	Unspecified	± 90-mV peak, I-I 4.5.2.2.1.4 Point A, figure 12 of 1553B
5. Input waveform — Maximum zero crossing deviation	Unspecified	± 150 ns 4.5.2.1.2.1 Point A, figures 9 or 10 of 1553B
6. Input signal response range Transformer coupling	± 0.5V to ±10V peak (1.0V to 20V p-p), I-I 4.2.5.4.1 Point C, figure II-14 page II-41	0.86V to 14.0V p-p, I-I 4.5.2.1.2.1 Point A, figure 9 of 1553B
Direct coupling		1.2V to 20V p-p, I-I 4.5.2.2.2.1 Point A, figure 10 of 1553B
7. Input signal no response range Transformer coupling	Unspecified	0.0V to 0.2V p-p, I-I 4.5.2.1.2.1 Point A, figure 9 of 1553B
Direct Coupling	Unspecified	0.0V to 0.28V p-p, I-I 4.5.2.2.2.1 Point A, figure 10 of 1553B Point A, figure 10 of 1553B

Table I-2.6 Comparison of Terminal Characteristics (Continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
8. Common mode rejection	± 10.0V peak, line-to-ground, dc to 2MHz 4.2.5.4.2 Point A, figure II-14 page II-41	± 10.0V peak, line-to-ground, dc to 2MHz 4.5.2.1.2.2 or 4.5.2.2.2.2 Point A, figures 9 or 10 of 1553B
9. Input impedance — Transformer coupling	Minimum of 2,000 ohms over a frequency range of 100 kHz to 1MHz, I-I 4.2.5.4.3 Point C, figure II-14 page II-41	Minimum of 1,000 ohms over a frequency range of 75 kHz to 1 MHz, I-I 4.5.2.1.2.3 Point A, figure 9 of 1553B
Direct coupling		Minimum of 2,000 ohms over a frequency range of 75 kHz to 1 MHz, I-I 4.5.2.2.2.3 Point A, figure 10 of 1553B
10. Noise rejection or error rate — Transformer coupling	Maximum bit error rate of 10^{-12} and a maximum incomplete message rate of 10^{-6} in a configuration of one bus controller on a 20-ft stub with a minimum of 100 ft of main bus cable between coupling boxes; test is conducted in presence of magnetic field per MIL-STD-462 method RS02 (spike test) with the limits of MIL-STD-461 RS02 4.3.3	Maximum of one part in 10^7 word error in the presence of additive white gaussian noise of 140-mV RMS over a bandwidth of 1.0 kHz to 4 MHz; input voltage 2.1V p-p, I-I Point A, figure 9 of 1553B, accept/reject Table II-6 page II-46 4.5.2.1.2.4
Direct coupling		Maximum of one part in 10^7 word error rate in the presence of additive white gaussian noise of 200-mV RMS over a bandwidth of 1.0 kHz to 4MHz; input voltage 3.0V p-p, I-I Point A, figure 10 of 1553B, accept/reject Table II-6 page II-47 4.5.2.2.2.4

DESIGNER'S NOTES

2.7 Remote Terminal Characteristics

Table I-2.7 Remote Terminal Characteristics

	1553A	1553B Notice 2	A-10	B-52 OAS	F-16 A/C	F-18	OH-58D
1. Terminal Address							
unique	■	■	■	■	■	■	■
broadcast		■					
2. Subaddress/Mode Code							
Subaddress 0 (mode data)	■	■	■	■		■	■
Subaddress 31 (mode data)		■					■
3. Response Time							
4.0-7.0μs	■		■	■	■	■	
4.o-12.0μs		■					
4. Transmission Bit Rate							
Long term	±0.01%	±0.1%					
Short term	±0.001%	±0.01%					
5. Terminal Fail-Safe Time Out							
660μs	■		■	■		■	
660-1000μs					■		
800μs		■					■
6. No-Response Time Out							
6.5μs						■	
14μs		■					■
Not Specified	■		■	■			
7. Coupling Transformer							
Direct coupled only							
Transformer only			■	■			■
Both	■	■			■	■	
8. Transformer Ratio							
1:1 ± 3%			■	■	■	■	
1:1.41 ± 3%		■					■
Unspecified	■						

2.8 Transmitter/Receiver Response Voltage Range

Table I-2.8 Transmitter/Receiver Response Voltage Range

Output Level Output Voltage Range (V)	RL	Application
18.0–27.0 p-p, 1-1	70 ohm	1553B, Notice 2
6.0–20.0 p-p, 1-1	Unspecified	1553A
24.0–26.0 p-p, 1-1	140 ohm	A-10, B-52 OAS, F-16A/C
28.0–36.0 p-p, 1-1	140 ohm	F-18
Input Level Input Voltage Range (V) Transformer Coupled	No Response Range	Application
0.86–14.0 p-p, 1-1	0.0–0.2v, p-p, 1-1	1553B, Notice 2
1.0–20.0 p-p, 1-1	0.0–0.4v, p-p, 1-1	A-10, B-52 OAS, OH-58D

Table I-2.8 Transmitter/Receiver Response Voltage Range (Continued)

1.0–20.0 p-p, 1-1	0.0–0.7v, p-p, 1-1*	F-18, 1553A
1.2–8.0 p-p, 1-1	0.0–0.9v, p-p, 1-1	F-16A/C
*Not specified for 1553A		
Direct Coupled		
1.0–20.0 p-p, 1-1	0.0–0.7v, p-p, 1-1	F-18
1.2–8.0 p-p, 1-1	0.0–0.9v, p-p, 1-1	F-16A/C
1.2–20.0 p-p, 1-1	0.0–0.28v, p-p, 1-1*	1553B, Notice 2
Not Allowed		A-10, B-52 OAS, OH-58D

3.0 SYSTEM DESIGN

The interconnection of subsystems using MIL-STD-1553 can be subdivided into two categories; data bus topology and data bus control. Other areas of concern to the system designer are functional partitioning, redundancy, and data bus analysis. Each of these topics will be discussed in this sector.

3.1 Data Bus Topology

The topology of a data bus system is the map of physical connections of each unit to the data bus. **Two types of data bus topologies exist; single level and multiple level (hierarchical). The single level bus topology ia the simplest interconnect scheme** and is the most commonly used architecture. In this approach **all terminals are connected to a single level bus** (see figure 13.1a) or multiple single level buses (see figure I-1.3.1b) each with an equal topology relationship. Notice that this can occur regardless of the data bus redundancy requirements. The use of multiple buses for redundancy in a single level system does not change to type of topology. A more detailed discussion of redundancy follows later in this section.

Multiple level or hierarchical bus topology ia an extension of the single level concept. If single level buses are interconnected in a certain manner data on one bus system will pass to another bus system. These buses differ from the multiple bus systems shown in figure I-3.1b, because they are not equal. Thus, one bus system is subservient to another bus system. This approach can be achieved using several different architectures as seen in figure I-3.2. **Several systems today are using this approach to achieve functional partitioning** (e.g., navigation to/from weapon delivery, avionics to/from stores management, etc.).

All 1553B data bus systems relate or communicate with each other via two control schemes; equivalent levels of control and hierarchical levels of control. The most common approach is equivalent levels of control. In this approach, coordination is required only when data is transferred. The autonomous operation and individual error handling and recovery schemes remain for each bus. Since this mechanization is the simplest and achieves the greatest separation between data buses, it is widely used.

Hierarchical control schemes are viable and have been developed and analyzed in research applications, but until recently had few applications. With the development and application of MIL-STD-1760B (see paragraph 7.2) hierarchical control will be applied to control a weapons bus from avionic buses or store management system buses. The weapons bus may further control a 1553 protocol type bus within a store (missile or pod) to communicate directly with warheads, navigation, communication, or radar sensors. When these approaches are discussed in the literature, the **bus level inequality ia usually expressed as local buses (subordinate - under submission of another bus) and global buses (superior - controller of local buses).** Regardless of the control scheme selected, each of these methods fit into a single category; multiple levels or hierarchical.

3.2 Data Bus Control

3.2.1 Bus Control Mechanization

The second part of the bus topology description is the control philosophy. Two terms have been used in the literature to describe these control schemes; stationary master and non-stationary master. The stationary master approach to bus control occurs when a single bus controller manages the bus communication for all devices on the data bus. If this bus controller fails, depending on redundancy requirements, a backup bus controller can take over operation of the bus. This takeover procedure usually involves several actions involving the data bus, external wiring between the two controllers, and internal self test in both controllers (see figure I-3.3). The use of data bus messages to keep both controllers aware of the others status and health are usually required. Data bus synchronization
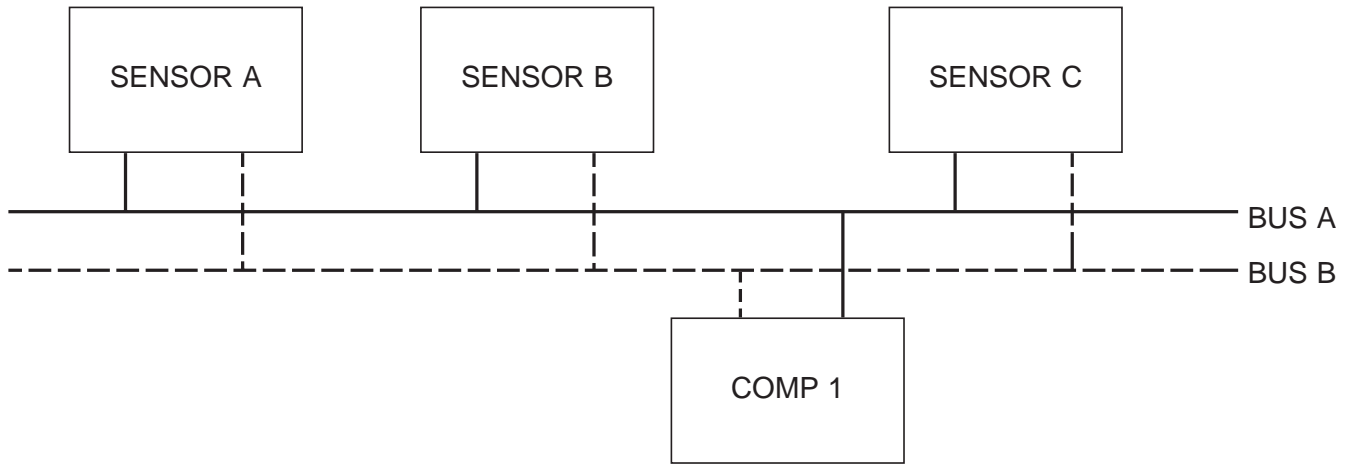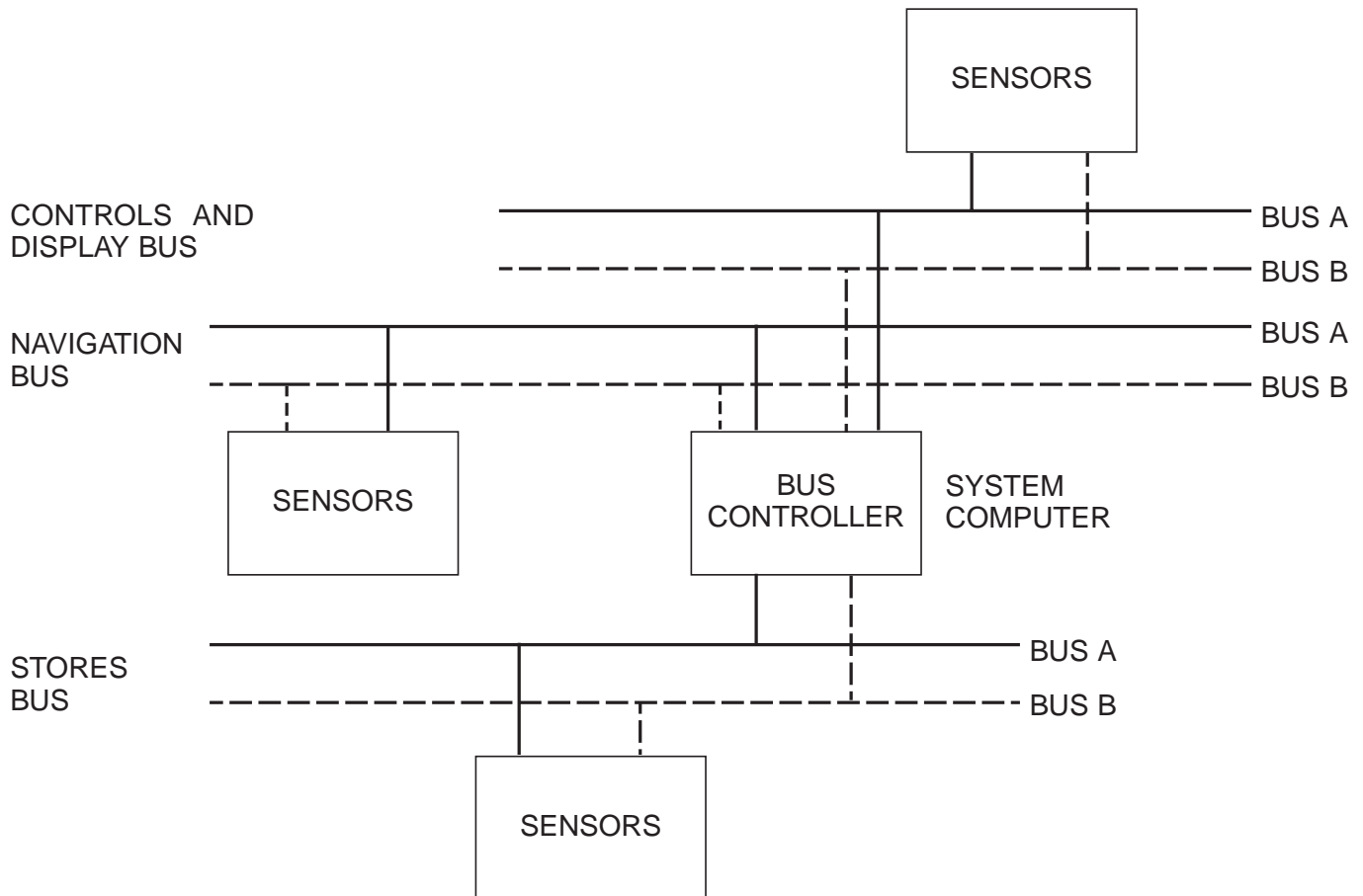
Figure I-3-1a Single Level Bus Topology

Figure I-3-1b Multiple Single Level Bus Topology

Figure I-3.2 (A)
Controls & Display Bus
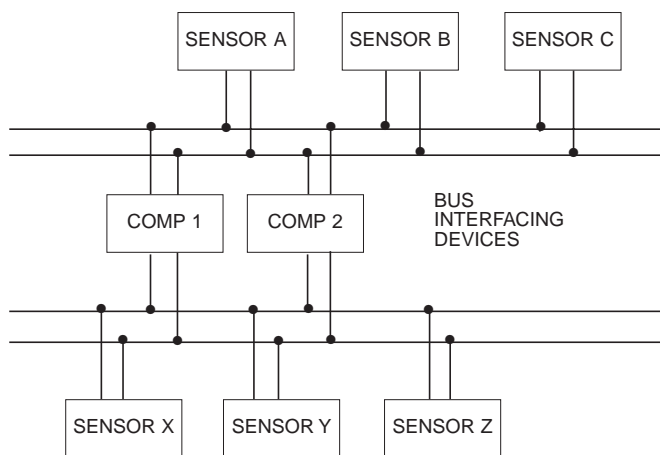
Figure I-3.2 (B)
Navigation & Weapons Delivery
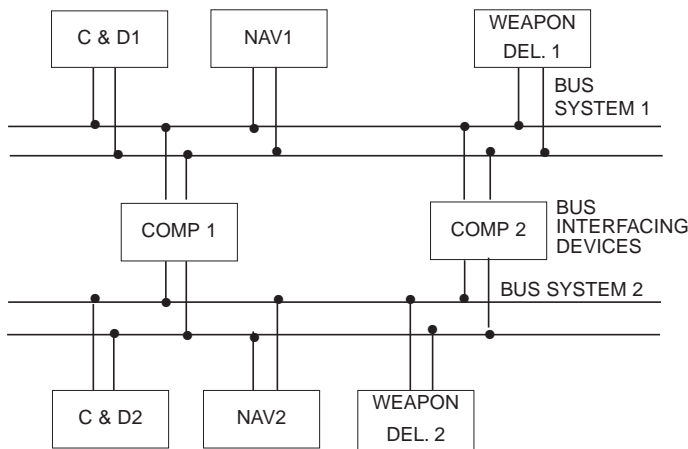
Figure I-3.2 Multiple Level Bus Topology

using clocks or synchronize mode codes also provide the two controllers with usable switchover data. Internal self test software is responsible for identifying faults and removing the faulty unit from operation. Giving power control to the other processor (BC) with proper time delays also resolves conflicts in which should be in control upon initialization. Dedicated discretes with hardware watchdog timers provide yet another measure of switchover monitoring. All or most of these approaches are used today to achieve switchover. The development of a safe and secure method to switch to the backup bus controller is an essential part of any stationary master control philosophy. When this bus control approach is applied to a single level bus system topology, a single bus controller will be used with a backup bus controller, as necessary. However, in a multiple level topology, stationary master bus controllers are located on each data bus system with a backup controller for each bus as necessary.

An alternative to a stationary master bus control system is a non-stationary master bus control philosophy. In this scheme, multiple bus controllers can control the single data bus system. Therefore, even in a single level topology several bus controllers can exist. Obviously, to allow all of the controllers the capability to control a single system, a method of passing control from one controller to another is essential, because 1553B allows only one controller to be in control at a time. MIL-STD-1553B uses the dynamic bus control mode code to accomplish this task. As discussed in section 1.4.2, the MIL-STD-1553B protocol provides a method for issuing a bus controller offer, thus allowing a potential bus controller to accept or reject control via a bit in the returning status word. The key to non-stationary master operation is to establish the number of controllers required by the application and when to pass control. As in the stationary master system, once control is transferred the operation is identical.

Two methods have been used; time based (producing a system similar to TDMA) and round robin (producing a system with a preordered list of bus controller). More complex schemes have been discussed where dynamically the existing controller polls (using conventional messages) to establish a priority of control and then passes the control to the controller with the greatest need. However, most of these approaches are very complex and require extensive error monitoring to provide confidence that the system as a whole is operational (only a single controller is in operation—no more or no less).

The round robin control mechanization utilizes a fixed order of bus control. This is usually accomplished by assignment of incrementing or decrementing addresses to all potential controllers. Since a potential controller is a remote terminal, when it is not in the controller mode, it responds to its remote terminal address, when the dynamic bus control mode code offer is provided. In these types of designs, the potential bus controller must accept the offer to become the next bus controller. If the new controller has no messages to transmit, it offers control to the next potential controller. Failures of potential controllers are managed by the active controller in one of two ways; automatically going to the next potential controller (requires extra knowledge on how to accomplish this task) or by appealing to a backup controller whose job it is to resolve system problems and relate the results back to all controllers. This provides each controller with the knowledge needed to pass control. Obviously, there are always design trades between increasing intelligence in each

PROCESSOR/SYSTEM
STATUS DATA

SYNCHRONIZE
MODE CODE

BUS A

BUS B

HEALTH DISCRETES

BC/BBC/DISCRETES

ON-LINE DISCRETES

SELF-TEST

SELF-TEST

PRIMARY BUS
CPNTROLLER

BACKUP BUS
CONTROLLER
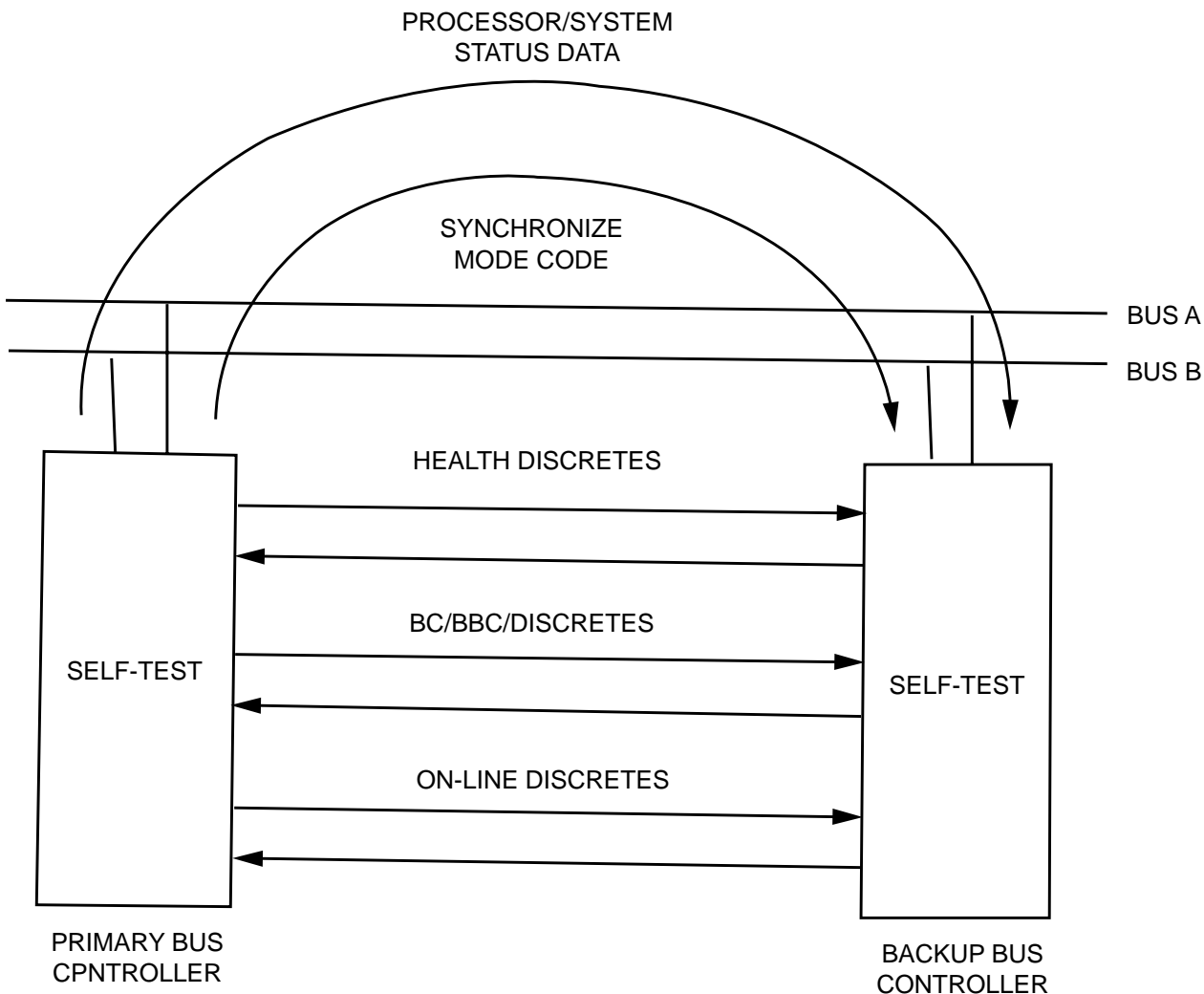
Figure I-3.3 Bus Controller Handshaking

potential controller or using a backup (single point failure) source to resolve the system configuration. This same problem also occurs at system startup. It is because of these complexities, that this type of system has had limited acceptance. However, with the growing use of multiple buses and the need to achieve data exchange between bus systems, this approach is being considered more and more often. Once again MIL-STD-1760B and other applications of multiple bus layers will yield a high degree of intelligence at the global bus level and thus a desire to maintain independence. All these affects will increase the interest of the system designer toward non-stationary master systems at global levels. This approach also has some unique system problems that must be resolved. Since each potential controller controls the bus each update period (maximum update rate of data transmitted in the system)

and each controller has different message transmission requirements, bus usage times will differ. This difference will occur between controllers and between update for a single controller (see figure I-3.4). This shifting of transmission times makes synchronous updates difficult. Any system using this method to achieve integration must be analyzed to determine if it can accept these asynchronous data transfers without time tagging data or paying a heavy penalty in bus usage (efficiency).

The time based mechanization allocates a fixed time for each potential controller to control the data bus. Once the maximum update is known then the period of time between updates can be allocated to individual controllers. The simplest approach is to divide the time equally. However, this may not have any bearing on the controller's need to transmit data during the

update period. This is where the system designer must trade controller complexity (different time bases for each potential controller) versus message update needs and message partitioning between controllers. This method allows the system to achieve synchronous operation at the cost of bus system efficiency if equal or fixed times are maintained. Bus efficiency is thus lowered by allocating fixed times to each potential controller. Obviously, during certain periods the controller will be in control and will have very few requirements (e.g., transmit or receive). Therefore, unused, unrecoverable bus time produces poor efficiency.

3.2.2 Error Management

Another aspect of data bus control mechanization is error management. Regardless of approach, an error management approach is required. Two types of failure condition exist; data bus system problems;

and subsystem or sensor problems. The method used to identify, determine the cause, and achieve corrective actions are all part of the multiplex system control requirements.

The data bus system can provide a communication medium to support error analysis associated with failure of sensor(s). Since sensors are quite different, the only capability the data bus system has is to convey the results of sensor built-in-test results or identify a non-communicating sensor. However, the data bus system is totally responsible for managing its own errors, including message failures and core element failures (i.e., bus controller and standalone remote terminals).

The 1553B standard requires correct word and message completion (1553B paragraph 4.4.1.2). These tests, plus system level checking, allow 1553 core element failure analysis. Some examples of test
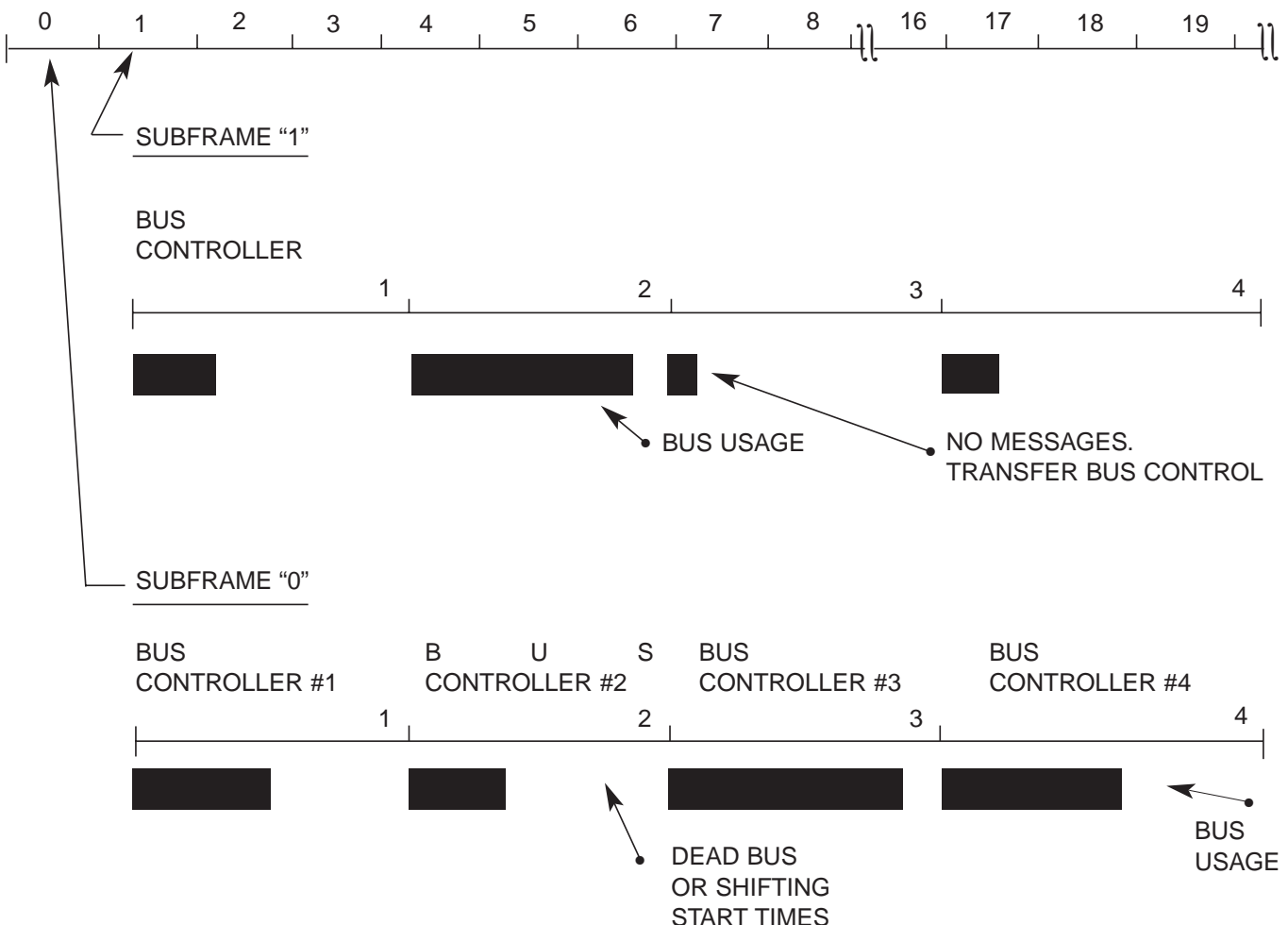
Figure I-3.3 Bus Controller Handshaking

results and correction techniques are provided in tables I-3.1 and I-3.2. No requirement as to how error analysis is to occur is provided in the standard but the standard provides a mandatory set of tests for word and message validation (improper sync waveform, invalid Manchester data bits analysis, number of data bits per word, odd parity, measure of discontinuity between data words within messages, and no response by remote terminal to a message error) and an optional set of remote terminal status bits (service request, broadcast command receive, busy, subsystem flag, dynamic bus control acceptance, and terminal flag) to manage the data bus system. The action to be taken depends on the error identification and the error correction desired. All these decisions remain with the system designer. Error identification is the responsibility of all terminals connected to the data bus, while error determinations and correction are the responsibility of the bus controller based on the systems designer approach.

It is the systems designer's responsibility to establish the systems response to each and every error identified by remote terminals or the bus controller. The error handling and recovery approach selected should be general in order to minimize hardware and software complexity. Usually the prime design

issue facing the designer is: "How far should error analysis go before making a decision as to the corrective action required?" The ability to minimize analysis without discarding operational resources is the challenge. Therefore, the system designer should establish these approaches prior to hardware and software definition. Most current BC and RT components contain the necessary logic to detect bus errors and inform the subsystem or host processor of the error conditions. For remote terminals, this entails including discrete signals and/or interrupt request/status logic to inform the subsystem of invalid messages. Bus controllers need to be able to autonomously detect if RT Status Word bit(s) are set as well as to detect no response and format errors from responding RTs. In addition, most current bus controllers, such as the ACE series, incorporate capability for performing automatic message retries. One method to convey these requirements to the hardware and software designer is the control procedure. Control procedures (see figure I-3.5) define the response to all normal and abnormal messages. In addition, control procedures establish how the system is to initialize, reconfigure, synchronize, shutdown, etc. To prepare this level of detail the system must establish the hardware and software partitioning and communicate the specific hardware design

Table I-3.1 Error Determination Approach

Error Identification	Failure Classes	
	Bus System	Sensor
a) Message error	Transmission from bus controller to terminal was decoded with error condition by receiving remote terminal.	—
b) Busy	Remote terminal unable to transmit or receive data at this time.	Remote terminal and sensor unable to transmit or receive data at this time.
c) Subsystem flag	—	Sensor failure preventing proper sensor actions.
d) Terminal flag	Remote terminal failure preventing complete action by terminal.	Remote terminal portion of sensor interface has failure preventing complete action by terminal.
e) Parity error (incorrect odd parity)	Error set in status word; data not usable by system.	Error set in status word; data not usable by sensor.
f) Improper sync	Unknown problem – ignore; continue to look for valid sync.	—
g) Invalid Manchester	Error in message – ignore data in message.	Ignore all data stored from message.
h) Improper number of data bits and parity	Error in message – ignore data in message.	Ignore all data stored from message.
i) Discontinuity of data words	Error in message – ignore data in message.	Ignore all data stored from message.
j) No status word response	Unknown problem – requires further investigation.	

Figure I-3.2 Typical Error Correction Techniques

Error Identification Types	Error Correction Technique
1. Bus system failures a} No status word response b) Message error c) Parity error d) Invalid manchester e) Improper number of data bits and parity f) Discontinuity of data words	Retry message on same alternate bus n times. Transmit reset remote terminal mode code if retry fails.
g) Busy	Retry message on same bus after a fixed delay time.

h) Terminal flag
 	
i) Improper sync
	If necessary, transmit initiate self-test mode code.
	Transmit BIT mode code.
	Analyze failure and determine corrective action, which may involve the following mode code commands:
	    Shut down transmitter (00100 or 10100)
	    Inhibit terminal flag bit (00110)
	Transmit reset remote terminal mode code of retry fails.
	Ignore and reset for valid sync.

j) Subsystem flag	Normal data communication messages (address/subaddress) to examine sensor BIT discretes or words.
2. Sensor failure a) Discretes b) BIT data word(s)	Analyze failure and determine system-oriented corrective action.

required to the hardware manufacturer and the software requirements to the software designer. This is usually accomplished with a detailed hardware specification and a software requirements document. The combination of control procedures and the actual hardware implementation yields sufficient data to develop software requirements. Then the software design accomplishes the system design approach.

The failure response to problems within the core elements of the data bus system include the data bus, the remote terminals, and the primary bus controller must be established prior to implementation. For the stationary master scheme, the failure of the bus controller causes a fail-safe transfer to the backup controller. This usually begins when the primary bus controller recognizes internal problems and ceases operation. The failure to notify the backup controller of operational capability on a regular basis and the lack of bus traffic alerts the backup controller to the fact that it should attempt to gain control.

In addition, hardwired discretes between the primary and backup controller can be used to indicate "who's in control," the terminal's health status (a discrete which toggles each frame), and if the terminal is "on-line" (capable of 1553 communication). The first signal is used during system startup to prevent collisions by the two controllers. Later, if the signal is removed by the primary controller, the backup then assumes control. The health indicator is used to verify that the two controllers are executing their software correctly. The command that toggles this line is usually performed after completing a portion of the background built-in-test. Failure to toggle this signal indicates a software problem, and that control of the bus should be changed. The last signal ("on-line") is again used during system initialization and later during monitoring by the backup controller if all communication on either bus ceases.

A positive method of access control is essential. Figure I-3.3 shows a simple method. See paragraph 3.2.1 for a complete discussion of bus switchover for a stationary master system. Obviously, it is much

more complicated with the nonstationary master control schemes. The problem is who will transfer control to the next potential controller" when the controller presently in charge has failed. As discussed earlier a system monitor is usually required to resolve these and other abnormalities. The design of the monitor then becomes a trade between system autonomy and error analysis and correction.

The failure of a remote terminal is usually detected by blts set in the status word or the lack of transmission from the terminal. The terminal flag bit provides an indicator of a problem, which can be investigated using the mode code transmit built-in-test word (BIT). There are no restrictions or definitions concerning the contents of the BIT word. However, the data should convey information, which allows the bus controller to take corrective action. If internal redundancy exists within the remote terminal, the ability to use the operational portion of the terminal must be conveyed to the bus controller.

Otherwise, the system must reconfigure to replace the failed device if such a system reconfiguration is possible. Since most

remote terminals use 1553 chip sets, the designer of the chip set has already established the BIT word format. Some chips allow host (non-chip hardware) to download some health data, others do not. The hardware and system designer's should establish what is acceptable for their particular system and select chips appropriately. The STIC and ACE component products include capability for formulating an internal BIT word. The internal BIT word, which is transmitted in response to a Transmit BIT word mode command, includes indications of transmitter timeout, internal loop test failure, DMA handshake failure, transmitter shutdown, terminal flag inhibited, bus channel for the last received message, and various message error conditions. The latter includes high and low word count, incorrect sync type, parity/Manchester error, various RT-to-RT transfer errors, and undefined
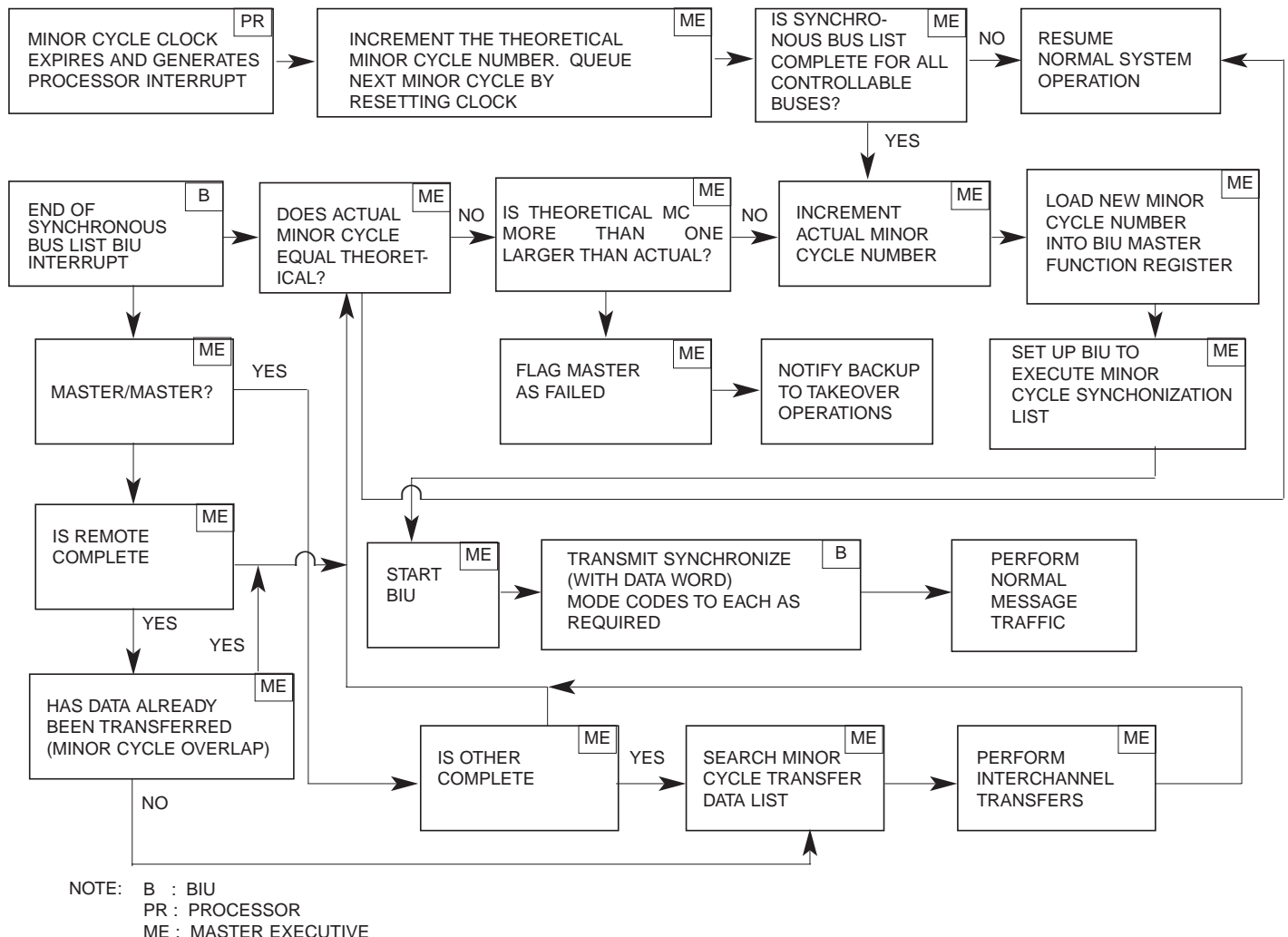

Figure I-3.5 Control Procedures

Command Word error. The ACE also includes capability for the host processor to re-format the BIT word via software.

The other core element is the data bus system. **A data bus system failure is usually recognized by the lack of communication with terminal(s).** In dual or greater redundant data bus systems alternate data buses can be used to re-establish communication.

3.3 Functional Partitioning and Data Bus Element Redundancy

To achieve the proper functional partitioning of a data bus system, requires the system integration knowledge and data bus system architecture experience. This type of integration must be viewed from an overall integrated approach rather than from a conglomeration of individual sensors, subsystems, controls and displays. The data bus system requires detail interface definitions and information flow to begin the orderly integration process. Since most of the integration is achieved using bus controller messages, identification of device to device communication begins the integration process. Before a topology or a control scheme is established a functional flow should be developed (see figure I-3.6). With functional flows, message definitions can begin.

It is at this time that the functional partitioning philosophy must be established if multiple bus systems are to be used. If a single level system is selected, the following discussion concerning partitioning is obviously not relevant. **For multiple bus topologies two philosophies are popular today; partitioning by function (control and display bus, navigation bus, weapon delivery bus) or partitioning by redundancy (separation of similar function—AHRS from INS). These philosophies should not be mixed in a single design.** It is the system designer's choice based on the application since neither philosophy has proven to be better than the other. System partitioning by function seems to be more prevalent today, because of the engineering organizational structure, laboratories, and the need for parallel development. Also the system engineers can best be used if they provide the guidelines and general requirements rather than be responsible for a closely integrated system. **This approach also allows loosely coupled subsystems.** The functional partitioning approach allows direct messages between similar functions (RT to RT) and an easy transfer method to redundant elements, since they exist on the same bus system. However, since all the redundant elements are on one bus system, total bus system failure must be considered. In a dual redundant system this would require a dual failure, which is a remote possibility for mission oriented applications.

Using the redundancy philosophy, similar devices are separated from each other and placed on multiple bus systems. This provides greater autonomy and failure protection. It also creates more traffic between bus systems, if the redundant elements share data or check each other. This method then requires greater time for messages to pass from one bus system to the other and an efficient data passing mechanism is essential for multiple bus systems. Regardless of the method chosen, partitioning begins with functional interconnections leading to message definition. At this point, the selection of the bus controller scheme can be made. **The type of control (stationary or non-stationary) and the level of redundancy of the data bus and the bus controller should be established. Generally, redundant bus controllers (active and backup) are used in stationary master systems.** Most non-stationary master systems do not provide redundant controllers because of the extensive use of controllers already. In the event of loss of an important or critical function, the monitor bus controller in a non-stationary master system can assume a limited set of these functions providing a degree of redundancy. **Data bus redundancy is usually dual. MIL-STD-1553B paragraph 4.6.3 states that if dual redundancy is used, the system must operate in an active/standby mode.** However, the system is not restricted to dual redundancy and it can operate at any level of redundancy required to meet system requirements. If a dual redundant system is selected, the active/standby mode can be implemented in at least two ways; indi-
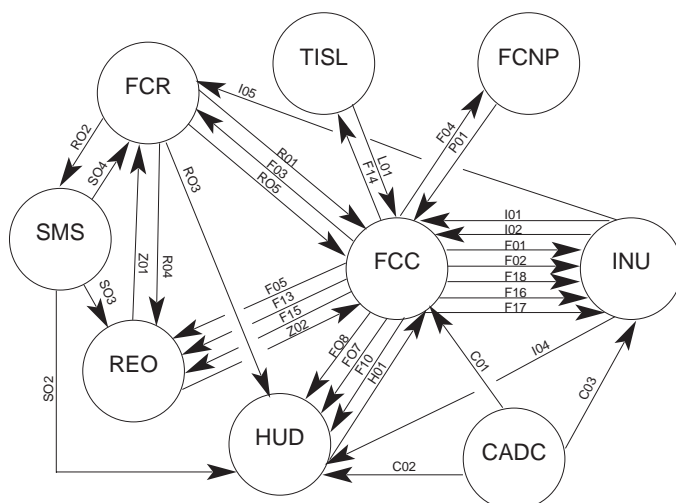
Figure I-3.6 Multiplex Data Bus Functional Block Diagram (Primary Mode)

vidual devices use either bus for communication (selection established by communication of the bus controller to the remote terminal) or block switch (all devices are communicated with over the same bus until a failure and then all are switched to the redundant bus). **The most common approach is the first one, which allows the bus controller to communicate using either bus with any terminal.** Since this method allows the greatest flexibility it is used almost exclusively. This approach does require the bus controller hardware and software to have the capability to distinguish, on a message basis, which bus it is to use.

The block switch mode is a slightly simpler bus controller mechanization, since it is not message or terminal selective, but system selective. However, this method limits flexibility and has not found wide usage.

Remote terminal redundancy is obviously equivalent to data bus system redundancy in the 1553 interface area. The design issue associated with the remote terminal is the extent of redundancy occurring in the circuitry approaching the subsystem side of the interface. Figure I-3.7 describes the functional elements within a remote terminal and shows three different design approaches to redundancy. In the first approach, the analog section is the only dual redundant section. **This approach is NOT compatible with MIL-STD-1553B dual redundant systems, because the command word validation must be established for each bus to meet the requirements of the standard.** This is accomplished in the second and third approach. The third approach uses completely independent interfaces to the subsystem, while the second approach duplicates only the minimum circuitry necessary to meet the standard. Another design is being used for remote terminals only, which includes two decoders and a single encoder. This method allows a terminal to receive on two buses while using only a single transmitter. This is an acceptable approach, but it has not received wide usage. This will change as more monolithic solutions evolve.

The advantages of the second approach are minimum hardware and good input/output flexibility, while the advantage of the third approach is isolation between channels. **Most interfaces for both remote terminals and bus controllers are built using the second approach.** This approach allows a bus controller to switch buses for retries without additional 1/0 support, a capability not available in the third approach. The third approach is used in systems requiring greater hardware isolation between the buses (i.e., flight controls). **Care should be**
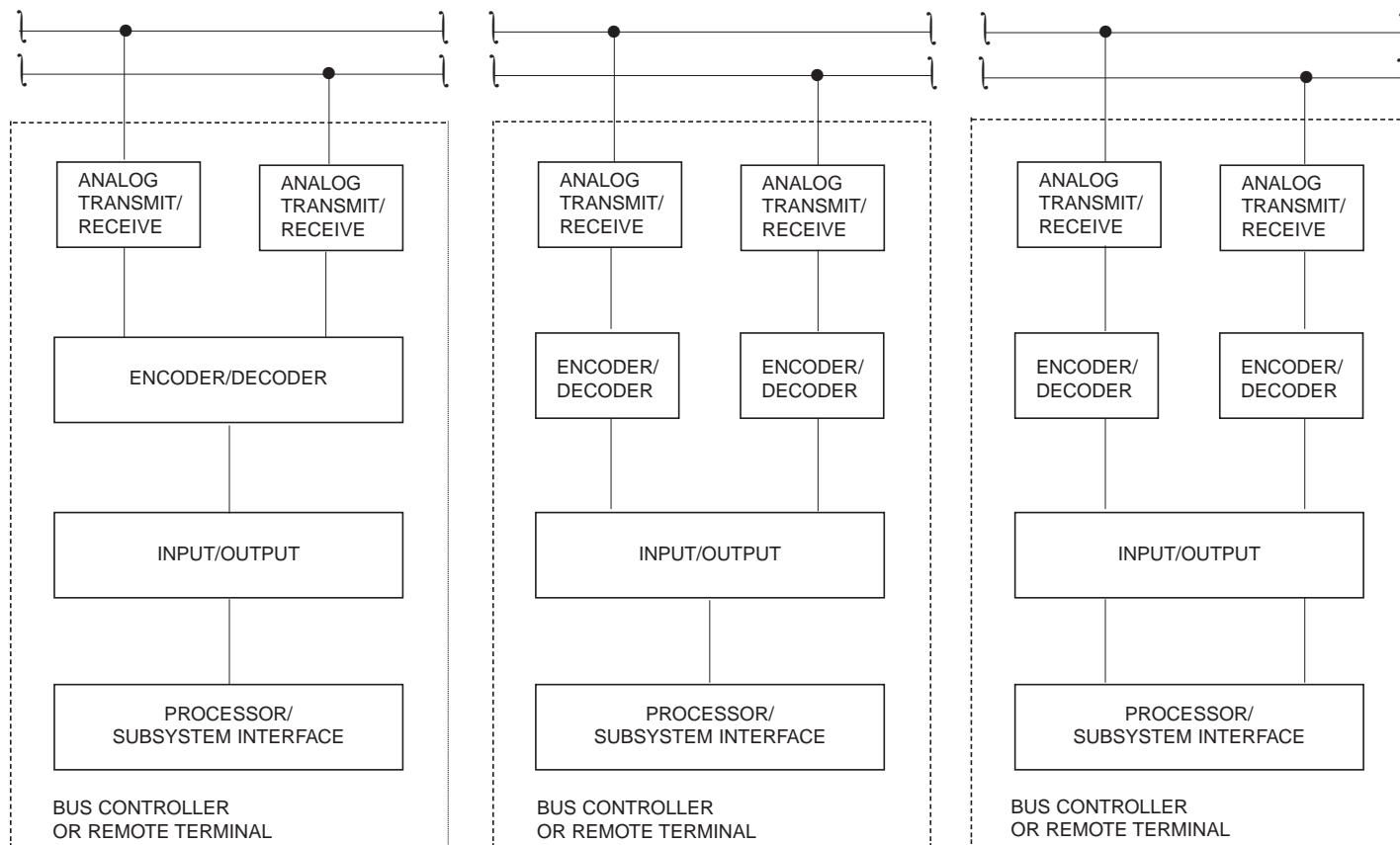
Figure I-3.7 Terminal Bus Interface Redundancy

taken in using the third approach for active/standby system. The channelization could prevent proper operation of mode codes like transmit last command. If this mode code is received in an active/standby bus system, it must report the last command regardless of which bus it was received on (Single channel chipsets employed in a dual redundant design will not meet this requirement without the addition of external circuitry).

3.4 Data and Control Passing in Hierarchical Networks

With the advent of multiple hierarchical networks, it became necessary to pass data from one network to another. At first this was accomplished using the bus controller processor. The B52 Offensive Avionics System shown in figure I-3.8 is an example of early architectures, which used the mission computer as the gate between networks. Note that for definition purposes, a GATE (also referred to as a GATEWAY) is used to pass data between similar buses (i.e., a 1553B bus to a 1760B weapons bus) both based on 1553B, whereas a BRIDGE is used to pass data between dissimilar buses (i.e., a 1553B data bus and a high-speed type of bus). In the B-52 example, the control and display bus was connected to the navigation and weapons delivery bus network using two mission computers.

Notice from the figure that one bus network is controlled by the mission computer acting as the primary bus controller while the other mission computer acts as a remote terminal and the backup bus controller. For the other bus network, the roles of the two mission computers are reversed. Data collected in the navigation and weapons delivery system is made available to the control and display network via the remote terminal side of the mission computer. Therefore, the most general example of 1553 gates are single units acting as bus controllers on one network and remote terminals on the other network. However, two other approaches are possible: remote terminals on both networks and bus controllers on both networks. Figure I-3.9 shows each configuration. The least likely architecture is both remote terminals (configuration 11). This configuration will be discussed briefly because of its drawbacks. The remote terminals in this configuration receive data asynchronously on each bus network, thus requiring extensive data buffering or inter-bus network synchronization to prevent data contamination. This configuration requires data protection to prevent data reception on one bus network while at the same time the same data is being transmitted on the other bus network. There are hardware and software methods of dealing with this problem, but it requires

extra effort to operate smoothly.

The architecture of configuration iii is most often seen in modern systems when a bus controller/processor controls multiple data bus networks in a synchronous fashion. Since control resides within a common computer, synchronization and data mapping control from one network to another is quite easy to accomplish. The most popular method is to operate all bus networks synchronously and globally map data in and out of large memory areas using message pointer tables (see paragraph 3.7.1 on sub-addressing) that directs the data arriving from one bus network into a common area, which is accessible by the other network for transmission. This reduces or eliminates the internal data movement required to pass messages from one network to the other. In this configuration, navigation data can be passed to stores management networks without the computer moving the data internally. Another primary reason to synchronize the control of the two networks from a single unit is to reduce the data latency time between networks. Since control of both networks resides in one computer, the knowledge of when time critical data has arrived and is available for transmission to other users is always present.

The category I configuration is the most commonly used for a gate in today's architectures because it supports a global network feeding the local networks or visa versa. Today's hierarchical bus network structures are composed of two types of local networks: data source networks and data sink networks. This can best be explained by examining a few examples. If a navigation local bus existed in a hierarchical architecture, its primary function would be to provide source data to other networks. In other words, most of the data generated would have destinations outside the local network. Contrast this with a control and display network. Its function is to sink data from many other networks. Only man-machine control data is likely to be transmitted outside of this local network. As you can tell from the example, no network is completely unidirectional, but data bus traffic will predominate in one direction or the other. Local networks, like these, tend to be remote terminals on global buses in large architectural systems and bus controllers on their local networks.

As a rule of thumb, it is always better to be a bus controller if the unit is a source of data and a remote terminal if it is a sink of data. Obviously, the benefit of bus control is it allows the subsystems timing control thus providing source data freshness. The counter is also true, if the network is sink oriented, the role of a

CONTROL AND DISPLAY (CAD) BUS

10100	00100	01010	10000	10001	10010	10011	00001		00011

| TEST INSTRU-MENTATION (TI) | DISPLAY ELECTRON-ICS UNIT (DEU) | CONTROL AND DIS-PLAY INTERFACE UNIT (CDIU) | DATA TRANSFER UNIT (DTU) A | DATA TRANSFER UNIT (DTU) B | DATA TRANSFER UNIT (DTU) C | DATA TRANSFER UNIT (DTU) D | IU / PROCESSOR A / BCIU | BCIU / PROCESSOR B / IU | NAVIGATION REFERENCE SYSTEM (CIRIS) |

| 10100 | | | | | | | | 00011 | 00011 |

NAVIGATION AND WEAPON DELIVERY (NAWD) BUS

01000	01001	01011	00000	01110	01111	00111	00110	00101	01100

| RADAR INTERFACE UNIT (RIU) | EVS INTERFACE UNIT (EIU) | ARMAMENT INTERFACE UNIT (AIU) | DOPPLER VELOCITY SENSOR | INTERFACE ELECTRONICS UNIT 1 | INTERFACE ELECTRONICS UNIT 2 | LAUNCHER MISSILE INTERFACE UNIT (MIU) | LEFT PYLON MISSILE INTERFACE UNIT (MIU) | RIGHT PYLON MISSILE INTERFACE UNIT (MIU) | ELEC STR ANT SYS (ESAS) |

Figure I-3.8 B-52 OAS

DESIGNER'S NOTES

CONFIGURATION I
BUS CONTROLLER/REMOTE TERMINAL

BUS 1

BUS CONTROLLER

PROCESSOR

REMOTE TERMINAL

BUS 2

CONFIGURATION II
REMOTE TERMINAL/REMOTE TERMINAL

BUS 1

REMOTE TERMINAL

PROCESSOR

REMOTE TERMINAL

BUS 2

CONFIGURATION III
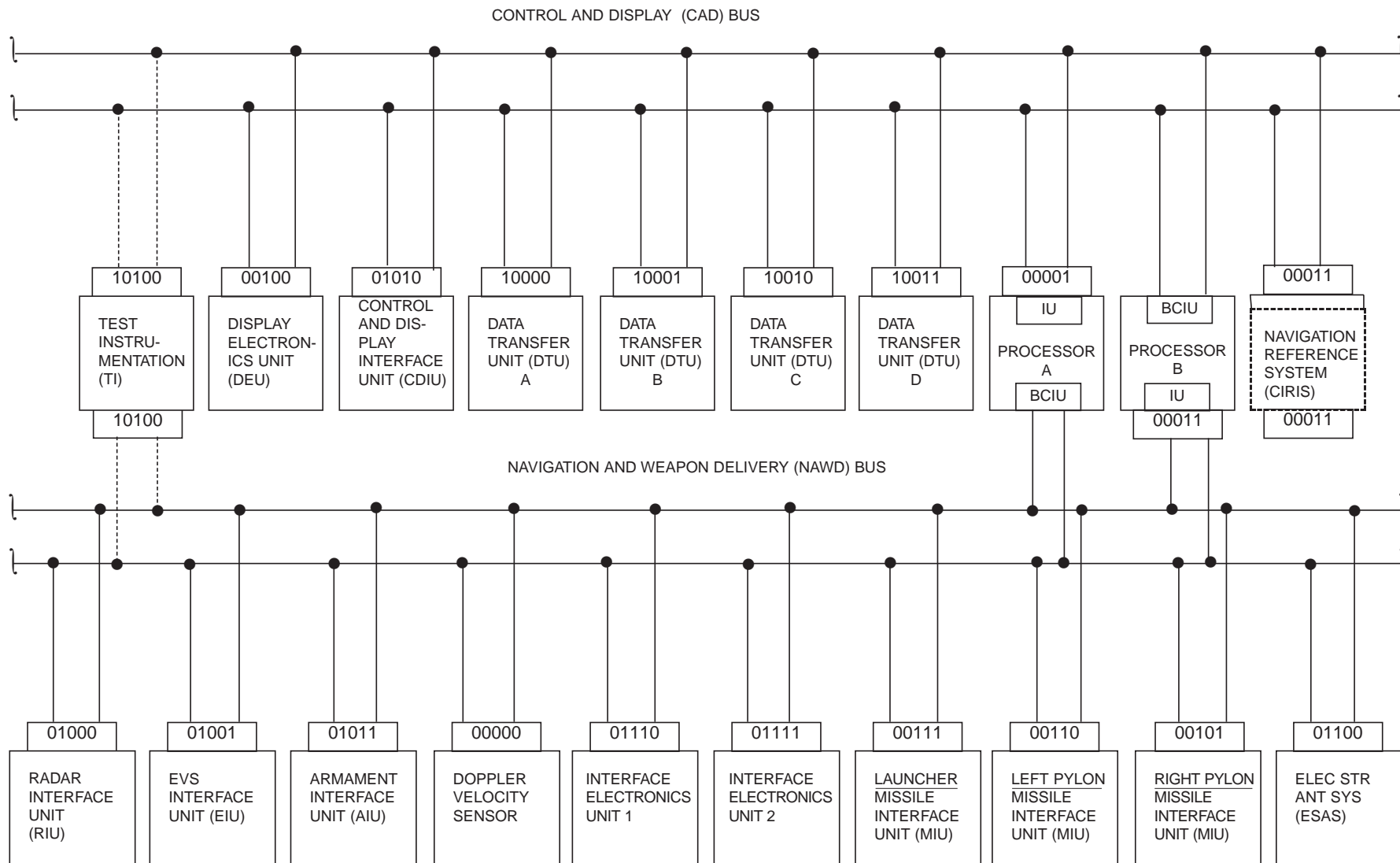**BUS CONTROLLER/BUS CONTROLLER**

BUS 1

BUS CONTROLLER

PROCESSOR

BUS CONTROLLER

BUS 2

Figure I-3.9 MIL-STD-1553 Gate Configurations

remote terminal is more than satisfactory. However, often these simple rules are not always practical in the architecture selected. Notice in the example of figure I-3.10. In this example, a mission computer is used to perform control of multiple global buses acting as a configuration III gate, while local buses (both sources and sinks) are configured as configuration 11 gates. Thus the design of a configuration 11 gate must be capable of data transfers in either direction. Generally, the data will have a definite direction, primarily in or out of the network, not both.

In the examples discussed so far, a large mission computer was used with its powerful, often processor based input/output section, to perform the functions of a configuration II or III gate. These units built by traditional airborne computer manufacturers often have very capable input/output circuits, which can perform either the role of bus controller or remote terminal. However, with the application of MIL-STD-1760B (see paragraph 7.2 for a discussion of the standard) in airborne 1553B networks, the need has developed for simple configuration II gates as shown in the weapons bus of figure I-3.10. The generic weapon interface unit is one of several units connected to the stores bus network as a remote terminal and to the weapons bus network as a bus controller. Many other simple hierarchical architectures may choose the same approach as opposed to complex mission computers. Therefore, a need has developed for a simple, almost hardware intensive, configuration II gate. The function of this device, (see figure I-3.9) is to receive and transmit 1553 data as a remote terminal on a higher level bus and transmit and receive 1553 data on a lower level bus as a bus controller. The features of this gate can vary from a unit as powerful as any gate built within a mission computer to a gate that can only carry out one message at a time from the higher level bus controller. With these extremes in mind and with the type of data flow that will be seen on the lower level bus (periodic or aperiodic), two basic design concepts have evolved in industry: transaction table controllers and peeling controllers. Each of these units will be described along with some of the reasons for determination of which is best for a given application.

The transaction table gate takes its name from the method it uses to perform the controller function. Transaction tables are lists of instructions, which are downloaded into the gate to direct the hardware to perform typical bus controller functions: transmit messages, receive messages, use

Figure I-3.10 Hierarchical Bus Network

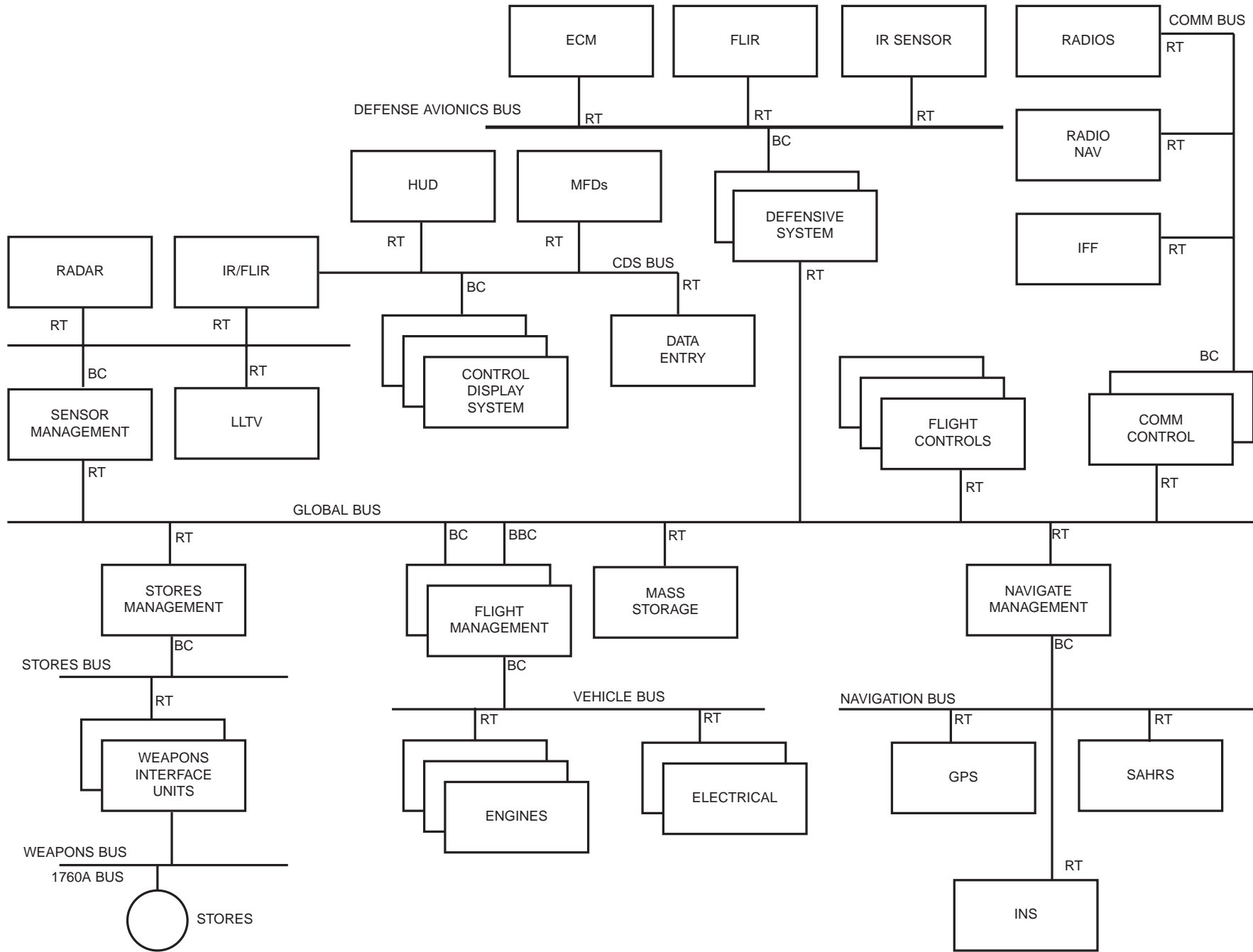mode code commands, skip messages (no-ops), jump from one list of messages to another, start, stop, use bus A or bus B, retry, interrupt on message transmission or reception, etc. All the features of a normal bus controller are built as instructions to the hardware. These called Channel Control Words (CCW) (see paragraph 5.1.1 for basic discussion on bus control and table I-5.1). Since each set of CCW's describes a single message, multiple sets of CCW's linked together form a minor cycle (see figure I-5.1). Since the flexibility exists to string messages together, periodic message strings can be built. Also error handling and recovery techniques are available using a fixed error handling message string. Generally, if a status word is not received or if received with errors, the limited computing capacity in this level of a gate requires error analysis and recovery to be accomplished in the higher level bus controller and passed back to the gate as a new set of CCW's or a change of the existing CCW string (e.g., no-ops messages to a failed unit). The ability to dynamically download changes to the CCW or new CCW strings is also needed to support aperiodic and time critical messages through the gate to the subsystems at the lower level. Because of its construction of tables this type of gate performs very effectively when periodic traffic predominates the lower level bus network. To support time critical messages, they must be prepositioned or the bus controllers internal message position (instruction address pointer) must be available to the upper level bus controller in order to link in the time critical message. This is an overhead intensive function and should occur on as few occasions as possible. However, it is a very usable controller method and can be performed in a timely manner. All system timing should be met with this approach because it fundamentally is no slower, and often times faster, than the upper level bus controller, which initiated the traffic. Time critical messages, such as clock updates and weapons release, can also be performed using transaction table gates supported by a 1553 bus controller on the higher bus network. As can be seen with the use of transaction tables for multiple CCW downloads, aperiodic communications links, normal data traffic, and error recovery procedures considerable subaddress expansion will be required by the remote terminal (see paragraph 3.7.1.1 for a discussion on expanded subaddressing). Also the local bus performance and data exchange between networks will work smoother if the higher level and lower level buses are synchronized. Synchronization of the two buses is achievable using 1553B synchronize mode code with the data word command described in paragraph 3.6.

The second method used to meet the configuration II gate is known as peeling. The name is derived from the procedure used to generate instructions to the gate. Each message passed, from the bus controller on the higher level bus, to the gate contains up to two words at the start of the data message, which are peeled off the message (not part of the message data content) as instructions to be used by the gate in the processing of the message. Usually, the first word contains mode information (e.g., message type (BC-RT, RT-BC, usually not capable of an RT-RT transfer), bus to be used (A or B), retry options, etc.). The second word is the complete command word for the lower level bus. The remaining words of the message constitute the data words of the message (maximum of 30 data words). Note that MIL-STD-1760B (see paragraph 7.2) limits weapons bus messages to 30 words in order to support this gate technique. This approach works well for aperiodic and time critical messages because the message itself contains the data that is to be passed to the lower level bus. The peeling technique is much less effective when trying to get the lower bus to source (transmit) data to the higher bus levels. In order to get data from the lower level to the higher level network, a two word message is transmitted to the gate, which becomes the transmit command on the lower bus. At some later time, the higher level bus controller will transmit a new message to the remote terminal (gate) to collect this data. This two step process is somewhat difficult for aperiodic messages, but extremely time consuming for periodic traffic. Figure I-3.11 shows the protocol sequence required to support this technique.

3.5 Network Startup and Shutdown

The system designer develops the operational capability of the 1553 network by establishing the options within 1553, which the particular system requires. Then the designer establishes the message mapping requirements and their update rates. Since 1553 was designed to perform primarily with periodic message traffic, each periodic message will have a source, destination, update rate and periodic position in selected minor frames. However, the reality of 1553 systems is that they must be started and stopped. Since the system usually operates in the periodic too little attention is directed at starting and stopping of the system. Why should this be a problem? In most applications power is applied to a few elements, which initialize and then startup others or all units are powered on when their power bus receives power. Asynchronously electrical bus power initialization yields subsystems in various stages of initialization. The problem is identical in shutdown. The error monitoring and control logic of
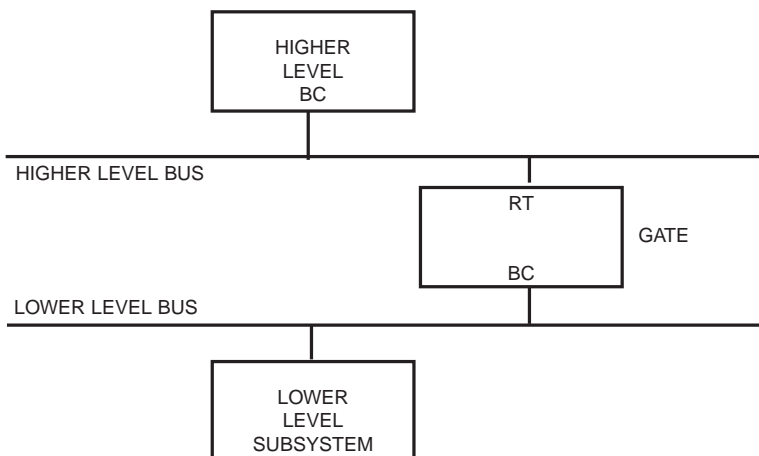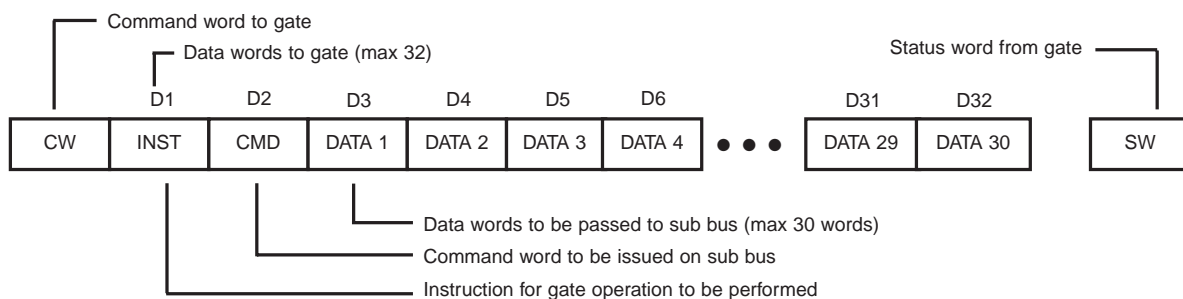
Figure I-3.11 Protocol to Support Peeling Gates

the bus controller utilizes the fact that a terminal is not communicating as a sure sign of trouble. Previous designs have been known to attempt communications with a terminal, which was either not powered or had not completed its internal self test and placed its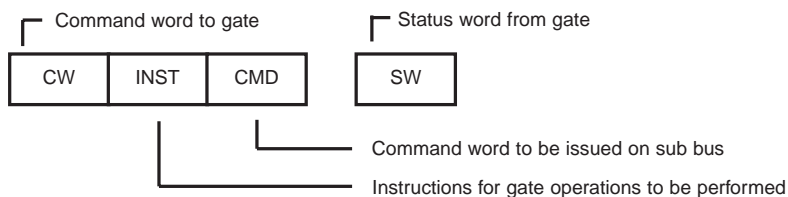elf *on-line." Having failed communication attempts, the bus controller terminal "dead," removed its messages from the periodic lists, and reported the failure to the maintenance program. Therefore, this logic must recognize startups and shutdowns as conditions where periodic messages may not occur due to the subsystem being powered off. Obviously at some point in any 1553 network, periodic communication can not exist successfully because too many messages in the system are missing. The system designer needs to establish the minimum message list necessary to continue to operate a particular periodic network. When a 1553 network is not operating in a periodic fashion, only minimal activity is possible. Aperiodic operations can be used by the bus controller to establish the primary bus controller and to initialize various 1553 subsystems. Initialization usually consists of the following:

a) initial communication with subsystems to determine their powered on state and health. Usually a unique subaddress, which provides subsystem health data, is chosen.
b) some subsystems require extensive initialization including program loads, system parameters, unique subsystem constants, time, etc. These messages are handled aperiodically with the subsystem prior to initiating periodic traffic.
c) system synchronization (see paragraph 3.6 on synchronization of terminals).

Once a sufficient number of subsystems exist (have been initialized and are "on-line") within the network to begin minimal operation, the bus controller can begin periodic message (minimal list) communications. After periodic communication begins on a network basis, all non-communicating terminals will be dealt with by error handling and recovery logic, which usually reports to application software that these subsystems are non-operational and presumed failed. Therefore, entering periodic processing too soon can cause false error reporting. If the bus controller later conducts a poll of all possible subsystems, late powered devices can be initialized and brought into the network, often upgrading the system to a more capable mode. Prior to periodic communication with these new devices, extensive aperiodic communication may be necessary. This could even be larger than at normal startup time, if the subsystem requires knowledge of the present state of the network.

Another important case, which needs to be considered, is the initial checking and monitoring between the primary and backup controllers. If one of the methods in which the backup controller determines it should takeover control of the bus is detection of bus dead time (a period of inactivity on either bus usually on the order of several minor frames), then the startup sequence has all the ingredients for a bus collision between the two controllers. To prevent this from occurring, one of two things must be done: a) always power up the primary controller first such that it will have completed its internal tests and will communicate on the bus first, or b) program a software time delay into the backup controller's bus control program such that it will wait a "reasonable" amount of time after it completes its internal self test before attempting to take control of the bus.

Generally, error handling and recovery software deals with startup and shutdown in a different manner that normal operation, due to the apparent error conditions that exist because of asynchronous power control and subsystem warmup. If power control of the network is available within the processor acting as the bus controller, system startup can be a much more coordinated operation. The same is true for shutdown. If this is the case, usually the bus controller is the first system on-line and the last system off-line. This is often very important in military systems, which can contain classified data. Often the bus controller executive/application software is responsible for commanding classified data erasure in other subsystems and finally within itself prior to power removal. If the bus controller is not responsible for power control, the startup length will depend on how long it takes the system to reach a minimum capability.

3.6 System Synchronization and Protocol

1553 systems are synchronized to allow an orderly processing approach to data arrival and departure. Most systems process data at fixed update rates (i.e., mostly binary rates, however some decimal based systems do exist), where on one cycle input data messages are received and on the next one or more cycles the processing of that data occurs, and on a future cycle data transmitted. Thus, synchronous operation must be timed to achieve the input, process, and output procedures. If the software in the terminal is to process the data during a certain interval, a flag is required to identify when all the required data has arrived and processing can begin. Most 1553 systems use a synchronization signal to

announce (flag) this occurrence. The synchronize signal can be the synchronize with or without data word mode code [00001 or 10001]. It the synchronize without data word mode code is used, only a time "tick" is announced not a specific frame number. Therefore, generally the synchronize with data word mode code is used to convey the event. The smallest increments of time (highest periodic data rate) are usually called a minor cycle or minor frame. Minor cycles are most often initiated by a system timer within the bus controller that interrupts the bus control software and declares "it's time to start a new frame." When this occurs, the bus controller usually completes the message in process and then decides if all of the planned periodic transmissions have been communicated for the present cycle. If not, the minor cycle must be completed or terminated. Systems which fail to complete a particular minor cycle cause terminals to miss their periodic data update. Also if minor cycles continually exceed their allocated time, the system will "jitter" in its periodicity. Both of these problems can cause serious systems problems and solutions need to be built into the system to deal with them. A "rule of thumb," which has been used in several systems is to allow the minor frame to overflow (extend the minor cycle until completed) up to one full minor cycle. After this time, the bus controller should either go on to the next minor frame or relinquish control to the backup bus controller.

A given minor cycle can be described (see figure I-3.12) as the time the system requires to: a) synchronize appropriate remote terminals, b) transmit and receive all periodic communications scheduled for this time interval, c) transmit and receive any aperiodic requests generated by the bus controller or remote terminals (via the service request bit in the status word and the transmit vector word mode code), d) transmit or receive background messages (multiple messages exceeding 32 data words occurring over a period of time, i.e., mass transfers and bulk data such as data bases or program loads), which do not require completion this frame, and e) bus controller polling of subsystem status and health messages. Items d) and e) will occur only on a "time available basis." Notice that the 1553 synchronize with data word mode code example provides a method of synchronizing remote terminals using minor cycle numbers. It also provides a method of updating an internal real-time clock with a resolution of 15 bits. Using the data bus to transfer system time can be done if exact time is not required. Resolutions under 100 microseconds are achievable if special hardware is provided in the terminal to

store the arrival of the data word and read and save the internal reading of the terminal's clock. With both the received clock word and the internal clock value upon reception, software can reset the clock by the off-set or ignore the difference due to the small error that exist. As can be seen in the example data word, (figure I-3.12) a method to achieve subaddress mapping commands is provided (see paragraph 3.7.1 for subaddress mapping discussion). By setting or resetting the time, update minor cycle, and update subaddress mapping bits, the data word can convey up to several encoded actions for a terminal with a single mode code transmission. An example of the events that would occur during a typical minor cycle using the features described above for the synchronize with data word mode code are as follows:

a) bus controller's real-time internal clock interrupts its software indicating that it's "time to start a new minor cycle"
b) the bus controller software determines that all mandatory messages are completed from the previous minor cycle
c) the bus controller transmits a synchronize with data word mode code to N terminals requiring real-time clock updates this frame (usually many frames can elapse between clock updates to a given terminal). This can be broadcast, thus updating all terminals using only a single mode code message.
d) the bus controller transmits synchronize with data word mode code to all terminals (discrete or broadcast) requiring synchronization. The data word is encoded to indicate minor cycle update minor cycle number, subaddress mapping data, and the subaddress mapping number. Since the next traffic is periodic messages, the terminal can be designed to accept the same subaddress mapping number for a given minor frame (usually both encoded values are identical)
e) the bus controller utilizes command words to perform all periodic communications required for this particular minor frame
f) if time critical messages must be introduced during the periodic traffic, a synchronize with data word mode code MIGHT be used to a specific terminal to switch its subaddress mapping number to an aperiodic (interrupt upon message arrival) table. If this occurs, the terminal will need to be returned to its periodic subaddress mapping number with another synchronize with data word mode code. Notice that changes can be made with subaddress mapping numbers without affecting the minor cycle number.

MINOR CYCLE
BOUNDARY

MINOR CYCLE
SYNCHONIZATION/
REAL TIME CLOCK

MINOR CYCLE
BOUNDARY
(RUBBER FRAME)

SYNCHRONOUS
MESSAGES

POLLING
FOR HEALTH

PERIODIC

• TIME CRITICAL

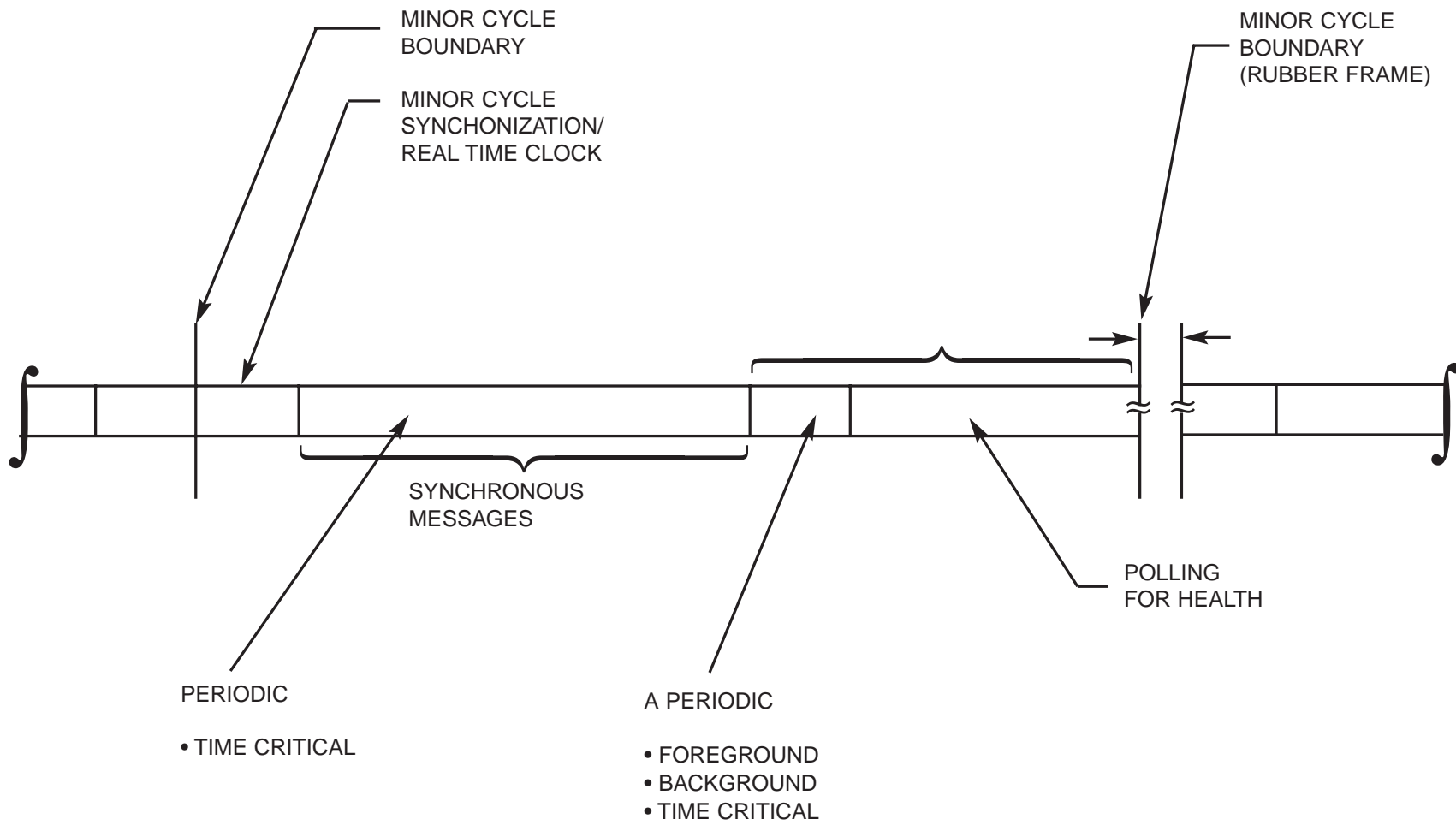A PERIODIC

• FOREGROUND
• BACKGROUND
• TIME CRITICAL

Figure I-3.12 Message Transmission During a Minor Cycle

DESIGNER'S NOTES

9) at the completion of periodic traffic, some terminals may be involved in aperiodic communications (foreground, background, time critical). This is accomplished by switching to the appropriate subaddress mapping table for each terminal. Each process may require use of the synchronize with data word mode code.

h) the final activity in a given cycle may be to switch to a set of communications, which is really a background message list. This list contains a set of messages, which request and collect 1553 status and subsystem health and can be stopped when the time runs out for the minor cycle.

i) return to a) or dead bus time until a) occurs.

The BC architecture for the ACE series includes several capabilities to off-load the host processor for implementing minor and major frame times. These include the capability to program intermessage gap times on a message-by-message basis. The resolution for the intermessage gap time is 1µs, with a maximum value of 65ms. In addition the ACE supports major frame times by allowing a frame of up to 512 messages to be automatically repeated without host intervention. The major frame time for the ACE's auto-frame repeat mode is programmable in 100µs increments, up to 6.55 seconds.

3.7 Data Control

3.7.1 Subaddress Selection/Operation and Data Storage

1553B remote terminal's utilizes a 5 bit address (bit times 4-8) within the command word to identify data reception requests or data transmission requests. The bus controller also specifies which message set within the terminal is to communicate based on the 5 bit subaddress field (bit times 10-14). Two subaddress codes (00000 and 11111) are used to designate a mode code, therefore 30 subaddresses are available (30 transmit and 30 receive) for data use.

1553B does not assign any subaddresses. However, Notice 2 says "a data wrap-around receive and transmit subaddress of 30 (11110) is desired." The maximum number of data words required to be received/transmitted is equal to the maximum word count the terminal is capable of for any subaddress. This additional requirement was added to provide the bus controller with a method of performing data pattern continuity (Manchester encoding/decoding) through a terminal's front end (1553 hardware) and to the beginning of the subsystem interface (i.e., memory buffer). Also MIL-STD-1760B defines subaddresses (see paragraph 7.2 for a discussion of MIL-STD-1760B). System 2, the nuclear weapons specification, also specifies certain subaddresses for buses having nuclear weapons as remote terminals (see paragraph 7.3 for a discussion on System 2). Prior to release of these notices, the industry had consistently been using receive subaddress 30 as an interrupt and transmit subaddress 30 as a subsystem health message. Also industry has typically selected lower number subaddresses for data messages when only a few were required. Table I-3.3 shows the subaddresses identified in the standards and specifications from (7/87). As can be seen from this list, a growing number of selected subaddresses are being assigned, which may conflict with existing equipment. Also Notice 2 requires remote terminals implementing the broadcast option to be capable of distinguishing between broadcast and non-broadcast messages to the same subaddress.

The system designer has always been concerned about address and subaddress assignments. The proper use and selection of subaddressing can save considerable embedded protocol in data messages.

Table I-3.3 Subaddress Usage

Standard Specification	Usage	Length
1. 1553B Notice 2 SA30RT	Data word wrap-around	Equal to longest data message of subsystem

2. 1760A Notice 2
    SA   1T
    SA 8R/T
    SA 19R/T
    SA 27R/T
	Store identification
	Test use only
	Nuclear weapons only
	Nuclear weapons only
		All 1760A Notice 2 and 3 messages are 30 words

3. 1760A Notice 3
    SA 11R/T
    SA 14R/T
	Storage control/monitor
	Store mass data transfers
		All 1760A Notice 2 and 3 messages are 30 words

Often mass transfers of data (greater than one 32 word message) occurs in 1553 systems. With proper use of subaddresses and memory mapping, multiple 32 word messages can be mapped into a contiguous memory area. This can be accomplished by assigning N (usually 16 or less) sequential subaddresses and then transmitting from or receiving these subaddresses and mapping to a contiguous block of memory. With the normal features of 1553B, an error can be detected and messages retired. Also this method does not require any particular order to the string of messages. In contrast, using an embedded protocol in 1553B data messages would require adherence to message order, sequence, and extensive error monitoring and correcting, which is not required with a set of sequential subaddresses. Another area where proper subaddress selection is necessary is for messages that require the terminal to interrupt and begin working on the data immediately upon arrival. These are usually aperiodic messages, which require immediate attention by the subsystem. To meet this demand, one or more subaddress may be dedicated to interrupt on reception or transmission.

A third category of messages are safety related. These might include flight safety or weapon safety (conventional or nuclear). In these cases, subaddresses are spaced digitally (i.e., maximum hamming distance) to prevent single or multiple subaddress bit encoding/decoding errors from causing the received data from arriving at an incorrect internal memory location or a transmitted message coming from an incorrect memory area. Each of these three categories of message along with the needs of complex subsystems has increased the pressure on finding sufficient subaddresses when only 30 transmit and 30 receive are available. Early in the 1553 development process this problem became apparent, when two computers communicating with each other via the 1553 data bus, exceeded 30 data messages. Therefore, the need for expanded subaddressing was realized. The basic idea has been used for many years. Previous designs have employed the first data word as a flag or control word to provide a new subaddress for the remaining data words. This obviously carries and overhead of word count reduction. However, in recent years a more generic usage of these techniques has developed, which allows application to any remote terminal needing more subaddresses. This method is discussed next.

3.7.1.1 Extended Subaddressing

The ability to request a remote terminal to remap its 30 subaddresses as a function of time, minor frame

processing, or message type (periodic, aperiodic, background, time critical) has been accomplished by tying message mapping and synchronization together in the system. To synchronize a 1553B system two mode codes are available: synchronize with data word (mode code 17 - 10001) and synchronize without data word (mode code 1 - 00001). Synchronize with data word is used to allow the system designer to identify which minor cycle or frame the remote terminal is to enter. Most systems operate with less than 100 frames per major cycle (time required for all periodic traffic to be transmitted once).

The data word in the synchronize with data word mode code allows the controller to switch subaddress mapping tables at anytime. The combining of synchronize and message mapping is shown in figure I-3.13, where the data word field is subdivided into a minor cycle number and an independent subaddress mapping field. This allows the mapping and minor cycle change to occur together or separately. If the subaddress field has one or more additional bits, the higher binary numbers (numbers greater that the number of minor cycles) can be used for aperiodic message mapping (foreground and time critical), background message mapping (multiple messages which do not complete within a minor frame), and large data transfer maps. Included in the large data transfer maps can even be program downloads.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	CYCLE NUMBER						1	SUBADR MAP NUMBER						

BIT	DESCRIPTION
0	Time info bit—if bit = 1 then bits 1–15 should be decoded as clock value
1	Minor cycle change bit—if bit = 1, set new minor cycle to value contained in bits 2–7
2–7	Minor cycle number
8	Subaddress change bit—if bit = 1, set new subaddress map to value contained in bits 9–15
9–15	Subaddress mapping number

Figure I-3.13 Synchronize Data Word
Format Example

3.7.2 Data Buffering and Validity

Message validity requirements within 1553B necessitate the buffering of the entire receive message until validity of the last data word can be determined. The matter is further complicated by the desire to ensure that transmitted data from a remote terminal is from the same sample set. Mechanizations which have been implemented to meet these needs include: first-in/first-out (FIFO) buffers contained within the Bus Interface Units (BIU) circuitry and standard memory devices, which employ a buffer switching scheme. The use of the first concept, a FIFO memory, is fairly straightforward. However, to ensure data continuity, a direct memory access (DMA) cycle long enough to read or write the total word count of the message is required. In today's processor systems, this timing constraint poses no problems. For the reception of data, the 1553 protocol control logic must be capable of clearing the FIFO (resetting the pointers to the starting address) in the event of an invalid message condition.

The second scheme, which is more commonly employed, allows the BIU to DMA directly into a memory buffer, usually on a word by word basis. A minimum of two buffers is established for each receive and transmit subaddress, each buffer being 32 words long (see figure I-3.14). The starting location of the buffers, a tag word for each buffer (see paragraph 5.1.4), and a series of control flags is usually contained in the Buffer Descriptor Block (figure I-3.15). These flags specify which buffer is to be used by the BIU circuitry and which is used by the host processor. In its simplest case, when the flag is set to the receive state, the BIU writes its data into one buffer as each word is received, while the host processor reads data from another buffer. When a complete message has been received, validated, and the final word stored, the BIU circuitry toggles the flag so that the buffers are now swapped. Obviously, if an error occurred within the message, the BIU would not toggle the flag. **For receive messages, the BIU controls the setting and changing of the buffer flag.** For transmit messages, when the flag is set for transmit, the BIU reads from one buffer while the host updates the data in another buffer. When completed with its update, the host will toggle the flag. **For transmit messages, the host updating the data controls, the setting and changing of the buffer flag.**

Now there are obvious conditions during which switching of the flag (and hence buffers) is not a good idea. For receive data, with the BIU in control
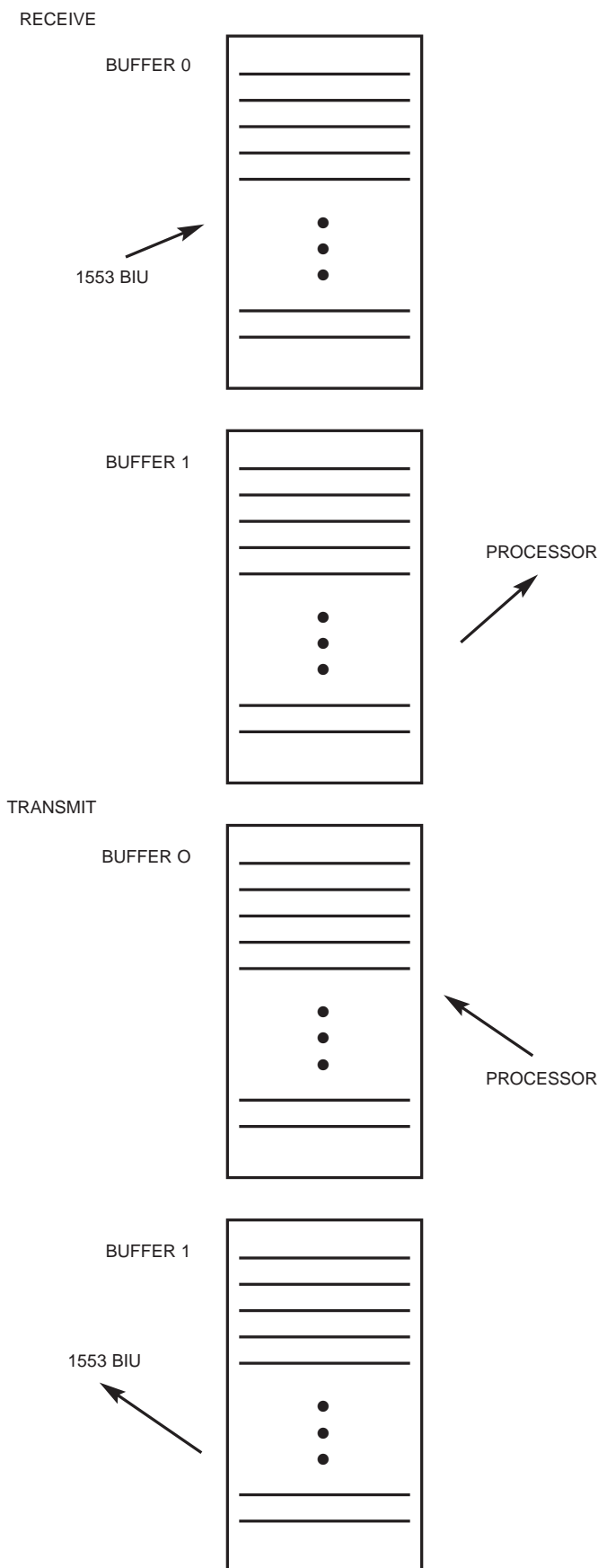
Figure I-3.14 Double Buffering Mechanization

this might occur after reception of a message, but NOT while the host is reading data from the opposite buffer, hence causing a mixture of old and new data. For transmit date, the host does not want to switch the buffers while the BIU is reading data from the other buffer. Therefore, some sort of handshaking between the host and the BIU is required.

However, before getting into the handshake protocol, there is a much more basic question regarding the transfer of data, which needs to be addressed. Does the terminal want to always receive the NEWEST DATA available (e.g., in the case of flight control parameters), OR is it more important to receive ALL DATA in sequence (e.g., the case of data base transfers or program loads)? In the case where newest data is required, the buffers are continuously swapped for each new message (assuming no errors). But in the case of wanting to receive all data, when both buffers are full (e.g., 2 messages received) AND the host has not read the contents of the first buffer, a problem exists because the next message would overwrite the first buffers contents, hence losing the data.

There are two simple solutions to the second case: first the use of more than two buffers; and second, the implementation of the busy bit in the status word. The first solution is simple, requiring only additional memory and the "housekeeping" logic to keep track of the sequence in which the buffers have been used (moving address pointers within a list). Some of the 1553 chip sets available support this approach. The second solution requires only additional logic within the BIU circuitry. The handshaking between the BIU, host, and busy bit is detailed as follows:

A BIU status flag is used to determine the status of the BIU with respect to the data buffer it is using. Similarly, a host status flag is used to determine the status of the host with respect to the data buffer it is using.

When newest data is desired, if the BIU and host status flags are cleared (inactive state), then the BIU/host is currently not using the buffer, while if these flags are set to the active state, the BIU/host is accessing the buffer. Note that these flags indicate the status regardless of whether the buffer is a transmit or receive data buffer.

Buffer swapping (the changing of the buffer flag) is performed as follows: Initially the BIU and host status flags are cleared (inactive state) and no valid data is contained in either buffer. For transmit messages, the host shall control the buffer swapping. To fill the

current (selected) buffer, the host first sets its status flag to the active state, indicating its buffer is busy. When the host has completed updating the data, it changes its status back to the inactive state. When this occurs, the host looks at the BIU status flag and toggles the buffer flag if the BIU's status flag indicates it is inactive (not using its buffer). If the BIU's flag indicated it was accessing its buffer, then the host must wait until the BIU is finished prior to toggling the buffer flag. It is important to note here that Notice 2 requires that all data being transmitted on the bus must be valid. Therefore in the initialization process, the BIU must either be kept "off-line" until the host has updated one of its buffers and toggled the buffer flag, or the BIU must respond with the busy bit set in the status word until the host has updated one of the buffers.

For receive buffers, the BIU is in control of the buffer swapping. To fill the current buffer, the BIU first sets its status flag to the active state to indicate the buffer is busy (done at validation of the command word from the bus). When the BIU finishes writing data, it changes its status back to the inactive state. When this occurs, the BIU will look at the host status flag and toggle the buffer flag as soon as the host's status indicates it is inactive. If an error occurred, the buffers are not swapped, and the erroneous data is overwritten by the next message to that subaddress.

When the collection of all data is desired, a BIU and host status bit cleared (set to inactive state) would indicate that the BIU/host has accessed and is finished with the buffer, while a status bit set to an active state would indicate that access to the buffer is currently in process or not yet started.

Swapping of the buffers and setting of the busy bit is performed as follows: For transmit data, the BIU and host status flags are initialized to indicate inactivity and activity respectively. This is done so that the BIU knows that there is erroneous data in its current transmit buffer. Here again, Notice 2 would require that the BIU respond with the busy bit set in the status word or be kept "off-line" until the host updates the first buffer. When the BIU or host is done accessing (updating) its buffer, it sets its status flag to the inactive state and checks the other flag. If the other flag is also inactive, then the data has not as yet been transferred out onto the bus by the BIU and the host must store the new data in the other buffer or somewhere else. For receive buffers, the BIU and host status flags are initialized to indicate activity and inactivity respectively. This is done so that the host knows there is erroneous data in its buffer. When a

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
DATA BUFFER POINTER 0	16 BIT ADDRESS POINTER TO BUFFER 0															
DATA BUFFER POINTER 1	16 BIT ADDRESS POINTER TO BUFFER 1															
TAGWORD 0	14 BIT TIME TAG WORD														SF	TF
TAGWORD 1	14 BIT TIME TAG WORD														SF	TF
BUFFER DATA	DB	BE	BS	HS	UNUSED											ET

Figure I-3.15 Data Descriptor Block Format

message comes in and the BIU has completed accessing its buffer, it shall set its status flag to indicate inactivity and checks the host's status flag. If it is also inactive, the BIU shall toggle the buffer flags (swap data buffers) and set both the BIU and host status flags to the active state. BUT if the BIU's status flag is inactive and the host's status flag is active (both buffers full), and a new command is received over the bus, then the BIU shall respond with the busy bit set in the status word and not store the data. If an error is detected in a message, the buffers are not swapped and the BIU's and host's status flags are not changed.

Due to the diversity of terminal designs, and the levels and complexity of memory buffering techniques, the design and implementation of the BIU/DMA/host interfacing logic is left to the subsystem designer and is usually not considered as part of the BIU protocol logic.

3.7.3 Block Transfers

Block transfers, moving of large amounts of data via the 1553B data bus, are common with today's data bus architectures. These transfers are used to exchange data bases between processor (e.g., primary and backup controllers), update navigational systems (e.g., GPS almanac data, digital map data bases, etc.), and perform operational program downloads.

One of the principal concerns in executing a block transfer is the handling of communications errors. This has to be accomplished such that there are not redundant or missed messages in memory. Auto retry capabilities are usually implemented in the execution of these transfers. Subsequently, when an auto retry occurs because of an invalid message, the designer must implement the control protocol such that the retired messages will not appear twice in memory with one being invalid and the other valid.

Various schemes have been employed to accomplish these bulk transfers. Commonly used methodologies are: time spaced messages to a single subaddress; multiple messages with subaddresses; and the use of expanded subaddresses (see paragraph 3.7.1). The designer must make trade-offs between the amount of memory available, levels of buffering desired or possible, and the amount of subaddressing available for use to insure that the bulk transfers succeed in moving the proper amounts of data, in the correct sequence, with no errors.

3.7.4 Data Protection

Additional protection of the data, other than the message and protocol checks contained with the 1553 standard, can be accomplished. Since data words on the bus are no different than the data contents of a 16 bit computer (with the exception of being transferred serially, and containing a sync and parity field), the traditional methods of computer data protection can be applied. These include checksums and cycle redundancy checks (CRC). Each have their advantages and have been successfully used in 1553 applications. The major disadvantage is that since their purpose is to protect data being "communicated," they need to be included with the data (meaning the codes must be sent with each block of data versus a separate message). Therefore, the use of these codes reduces the amount of data which can be transferred in any given message.

During early applications of the standard, some terminals did not protect against the mixing of old and new data within a message. While checksums and CRC codes were capable of detecting this type of error, their overhead was considered too great to be used in all messages. Therefore an effective, low overhead solution was developed — the validity bit. A validity bit (single bit) was assigned to each data

word. Implementations varied as to the inclusion of this bit into each word (often the MSB) or placing all bits in the last two words of the message. The placing of the bit within the word reduced bus loading and allowed a single word read by the applications software, but it reduced the resolution of the signal it was attempting to protect. When set, the bit indicated that the associated data was good. When multiple words were involved, the processor updating the data would set the bits to the same pattern (0 or 1), alternating patterns at each update. Hence it was possible to distinguish data not of the same sample set. Today, most terminal designs, by the method of double or multi-level buffering, assure all data transferred is of the same sample set.

For data that must have extended protection, multi-bit detection and correction, a hamming code protection scheme is recommended. Section 80 (formerly Chapter 11) of MIL-HDBK-1553 Multiplex Applications Handbook provides an error protection word based upon a BCH (31,16, 3) code, which will provide error correction up to 3 bits. The overhead is great. For each protected word, an additional word is required, but if this level of protection is required—use it.

3.8 Data Bus Loading Analysis

Most newcomers to MIL-STD-1553B are concerned with bus loading. The serial bus standard is like many others, it works fine until you try to overload it. On the positive side of efficiency, the protocol established by the standard is a very efficient protocol and provides extensive data transfer capability.

Analysis of bus loading is a relatively simple matter requiring only a hand-held calculator, some data about the system, and some basic system decisions concerning use of the standard. There have been several computer programs developed to calculate bus loading. These programs are generally used on very large and extensive systems with many messages. Bus loading programs are used more to manage the size of the message data base and its complexity than to calculate bus loading or build message sequencing. The basic data necessary for a simplified calculation of average bus loading is:

(1) Message/type
(2) Words/message
(3) Overhead associated with each message type
(4) Overhead associated with mode codes
(5) Intermessage gap (6) Average response time

(7) Overhead associated with nonstationary master bus controller passing

The overhead constants to consider (in microseconds) are:

(1) Command word 20
(2) Status word 20
(3) Response time .....2-10 (average 8)
(4) Intermessage gap ...2-100 (average 50)
(5) Mode codes without data words ..20
(6) Mode codes with data words 40
(7) Data words 20
(8) Nonstationary master bus controller passing ... (48 minimum)

When calculating average bus loading each message type has a value where "N" is the number of words in the message.

(1) Bus controller to remote terminal and remote terminal to bus controller. 20N + 68 = Value for BC-RT

(2) Remote terminal to remote terminal 20N + 116 = Value for RT-RT

(3) Bus controller to remote terminals (broadcast) 20N + 40 = Value for BC-RT (broadcast)

(4) Remote terminal to remote terminals (broadcast) 20N + 88 = Value for RT-RT (broadcast)

(5) Mode code without data word 68 = Value for MAC without data word

(6) Mode code with data word 88 = Value for MC with data word

(7) Mode code without data word (broadcast) 40 = Value for MC without data word (broadcast)

(8) Mode code with data word (broadcast) 60 = Value for MC with data word (broadcast)

Therefore, the average bus loading is the sum of the message type values divided by 1,000,000 (maximum no. of bits/see) times 100%. A system should not exceed 40% bus loading at initial design and 60% at fielding, in order to provide time for error recovery/automatic retry and to allow growth during the system's life.

3.9 Interface Control Documents

Section 80 of MIL-HDBK-1553 Multiplex Applications Handbook (formerly Chapter 11) provides the guidelines for the development of data words and message structure formats needed in the generation of Interface Control Documents (ICDs) for data bus compatible equipment. In addition, it provides suggestions for ICD format presentations and recommended coding techniques for various data words. Both the Navy and Army have developed data bases for terminals and systems, which are available. Systems and terminal designers are encouraged to adapt these guidelines and formats in the generation of their own documentation and to make use of the data bases where possible.

Inasmuch as Section 80 has been successfully adapted by much of industry for standardization of message and data word formats, it has some shortcomings in identifying all the required data needed by the systems level designer. Some of this data is available at the initial design while other parts may not be available until verification or final buy-off tests of the terminal have been performed. The purpose of this section is solely to identify the types to terminal data which the systems level and bus control software engineers need.

a) Condition which sets the optional status word bits (busy, service request, subsystem flag, terminal flag) and any required responses by the bus controller)

b) Specific conditions for generation of mode codes and required responses to these

c) Timing limits associated the mode codes (i.e., how long to reset or perform self test)

d) Discretes to be monitored and a detailed operation of each (e.g.,'on-line', BC/BBC, etc.)

e) Modes of operation and specific procedures associated with each (e.g., built-in-test, initialization, normal operation, maintenance, priority operation, etc.)

f) Specific power-up initialization sequence including required messages or programming (e.g., data loads, parameter initialization, operational controls, etc.)

g) Maximum power-up time till 'on-line' including sequence (e.g., no response, followed by busy, followed by normal)

h) Message timing constraints and interaction (if any) with the busy bit

i) Self test procedures (internal, data wraparound, etc.)

j) Data coherence and sample consistency procedures

k) Service request procedures (command sequence or vector word definitions)

l) Bus electrical characteristics (output voltage, impedance, etc.)

m) Electrical interface requirements (connector type, pin assignments, voltages, currents, etc.)

n) Programmable terminal parameters (terminal address, operational modes such as monitor or backup controller)

o) Specific hardware self test procedures, BIT word definitions, maintenance code definitions

p) Specific polling procedures for monitoring bus and terminal health

q) All subaddress and mode code message and data word formats in accordance with Section 8-0 of MIL-HDBK-1553.

r) First minor frame present and offset in each future periodic communications frame

s) Unique shutdown requirements

t) If message requires interrupt upon reception or transmission

u) Unique or subsystem specific message strings (loading, mass transfer, sequence of messages, etc.)

v) Mode switching constraints

w) Validity bits (usage, rules for setting, interaction with data)

x) operational characteristics during "shutdown" transmitter mode code (i.e., is the terminal still capable of receiving and processing data from the bus)

4.0 HARDWARE

4.1 Types of Terminals

1553B defines a terminal as "the electronic module necessary to interface the data bus with the subsystem and the subsystem with the data bus. Terminals may exist as separate line replaceable units (LRUs) or be contained within the elements of the subsystem." **A terminal is further categorized by 1553B as either a bus controller, bus monitor, or remote terminal.**

(1) Bus Controller
MIL-STD-1553B defines a bus controller as "the terminal assigned the task of initiating information transfers on the data bus." Notice that the definition does not necessarily depend on the physical design of the terminal but is determined by the assigned task of bus control. This implies that a terminal may have the capability of performing other functions, but during the time when it is assigned the task of bus control it is by definition a bus controller.

Figure I-4.1 shows the generalized terminal functional elements that apply to a bus controller.

(2) Bus Monitor

A bus monitor is defined as "the terminal assigned the task of receiving bus traffic and extracting selected information to be used at a later time." A bus monitor, therefore, is unique in that it performs no transactions on the bus (see figure I-4.2). It not only does not initiate information transfers as a bus controller, it is incapable of any response on the bus, including status response in the monitor mode. In fact, the bus monitor does not require a transmitter so it may be a "receive-only" terminal if it never performs another function (bus controller or remote terminal). However, **monitors usually have remote terminal capability to allow the bus controller to check their status health and command new operational modes.** Primary applications include collection of data for analysis and back-up bus controller monitoring of system status. The bus monitor, for instrumentation applications, listens to all messages or a subset of messages and stores the data internally or formats and outputs the data to a mass storage device or telemetry equipment. Many monitors are configuration programmable as to their function of message monitoring capability via 1553 messages. In association with a back-up bus controller (whether the back-up controller is either a monitor or a remote terminal), the monitor observes the bus transmissions and collects data, often performing the same operations as the bus controller with the exception of issuing commands on the bus. 8y operating as a monitor, the back-up controller is continuously aware of the state and operational mode of the system and subsystems. Therefore in the necessity of assuming control of the bus, the back-up controller can reduce its start-up time. This type of operation is often referred to as a "hot back-up."

(3) Remote Terminal

A remote terminal is defined as "all terminals not operating as the bus controller or a bus monitor." This means that an RT cannot initiate information transfers on the bus as a bus controller and cannot perform the monitor function. It must respond to commands issued by the bus controller in a normal command/response manner. **An RT Is identified by a unique address that allows the bus controller to direct specific information to it.** Figure I-4.3 shows the generalized terminal functional elements that apply to a remote terminal. Once more it must be emphasized that the definition of an RT is one of function, not form. Any active terminal that is not performing bus control or monitor functions on a given bus at that time by definition is a remote terminal.

WORD PROCESSOR — MESSAGE PROCESSOR

DATA BUS INTERFACE → ANALOG TRANSMIT/ RECEIVE → BIT/WORD PROCESSOR (ENCODER/ DECODER) → MESSAGE PROCESSOR → SUBSYSTEM INTERFACE CIRCUITS → SUBSYSTEM OR HOST PROCESSOR INTERFACE

- Receive
  - Signal limiting
  - Filter
  - Threshold detection
- Transmit
  - Driver
  - Transmitter control
  - Timer
- Common
  - Bus coupling

- Receive
  - Receiver
  - Sync detection
  - Data detection
  - Manchester error detection
  - Parity check
  - Bits/word count
- Transmit
  - Transmit control
  - Sync/data encode
  - Parity generation
- Clock generation

- Receive
  - Command word decode
  - Status word decode
  - Address recognition
  - Mode execution
  - Word count recognition
  - Message error detection
- Transmit
  - Word count (BC)
  - Status register
- Built-in test
- Subsystem interface control
  - Data address
  - Control registers
- Memory buffer

- Channel selection
- Data sampling
- Conversion
- Subsystem timing
- Buffer registers
- Calibration
- Self-test
- DMA to host memory
- Interrupt lines
- Control lines
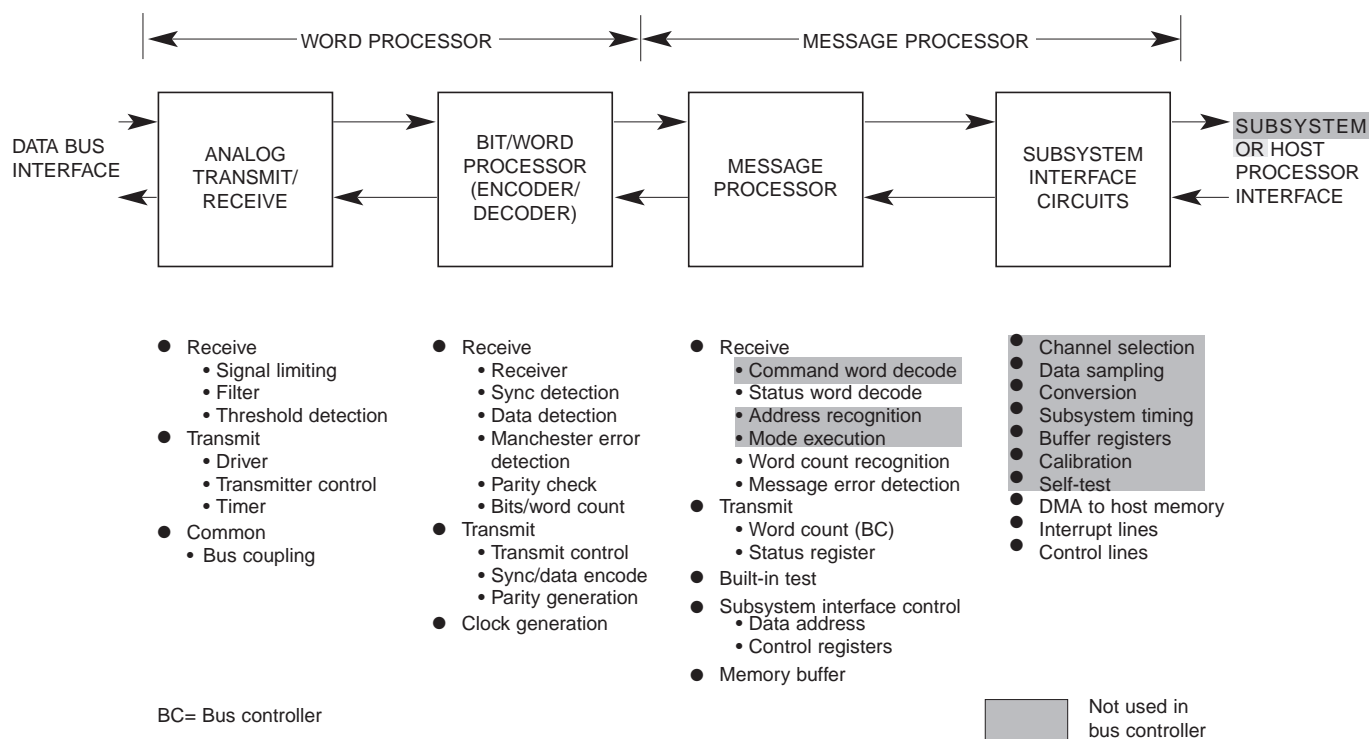
BC= Bus controller

Not used in bus controller

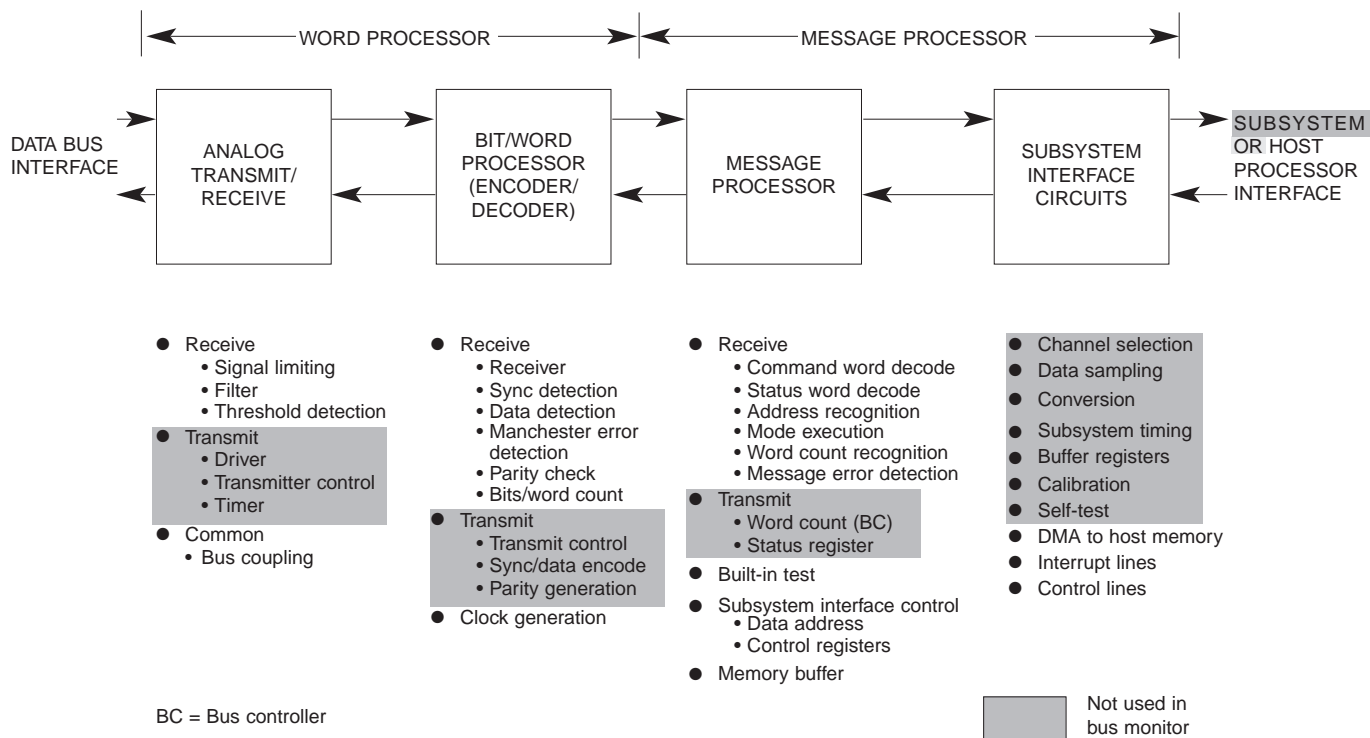Figure I-4.1 Bus Controller Functional Elements

Figure I-4.2 Bus Monitor Functional Elements

It is a common practice that intelligent interfaces provide both bus control and monitor capability as well as RT functions within a terminal. This allows an RT to become a bus controller in the event of failure of the active bus controller.

A clear understanding of the functional nature of these 1553 terminals is essential to the understanding of the terminal partitioning. Also, the difference in requirements for each type of terminal will impact their design. Table I-4.1 summarizes a typical set of requirements for several terminal types.

4.2 Functional Elements

A MIL-STD-1553 terminal may be specifically designed as a standalone remote terminal, embedded remote terminal, bus controller or bus monitor. **Flexible terminal designs often perform both bus controller and remote terminal functions. State of the art 1553 terminals, such as the STIC and ACE, are implemented as fully Integrated components.** This is achievable because of their intelligence (processing power) and a common front end design compatible with both remote terminal and bus controller functions. This section will describe the functional elements of a generalized terminal design. Figure I-4.4 identifies the major functions incorporated into a generalized
terminal design. The four major functional elements are: (1) the analog receiver/transmitter, (2) the digital

encoder/decoder, (3) the digital message processor, and (4) the subsystem interface. Transfers of data in and out of each of these sections may be either
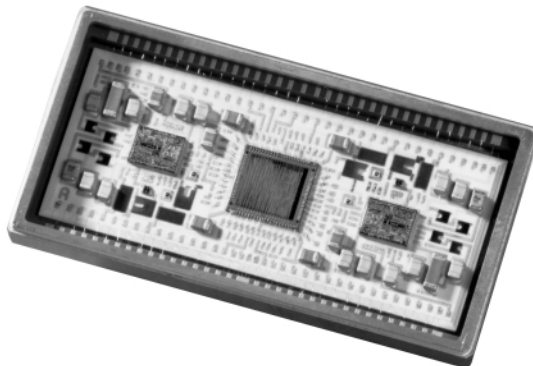
BUS-65153. The BUS-65153 Small Terminal Interface Circuit (STIC) integrates dual transceiver and RT protocol logic in a 1 " x 1.9" hybrid. The STIC complies with MIL-STD-1553B Notice 2 and meets the response time requirements of MIL-STD-1553A. The BUS-65153 includes a DMA interface that is pin programmable for 8-bit or 16-bit data buses. The STIC is ideal for interfacing a dual 1553 bus to simple systems that do not have a microprocessor. The STIC's small size and low cost make it well suited for MIL-STD-1760 stores management applications
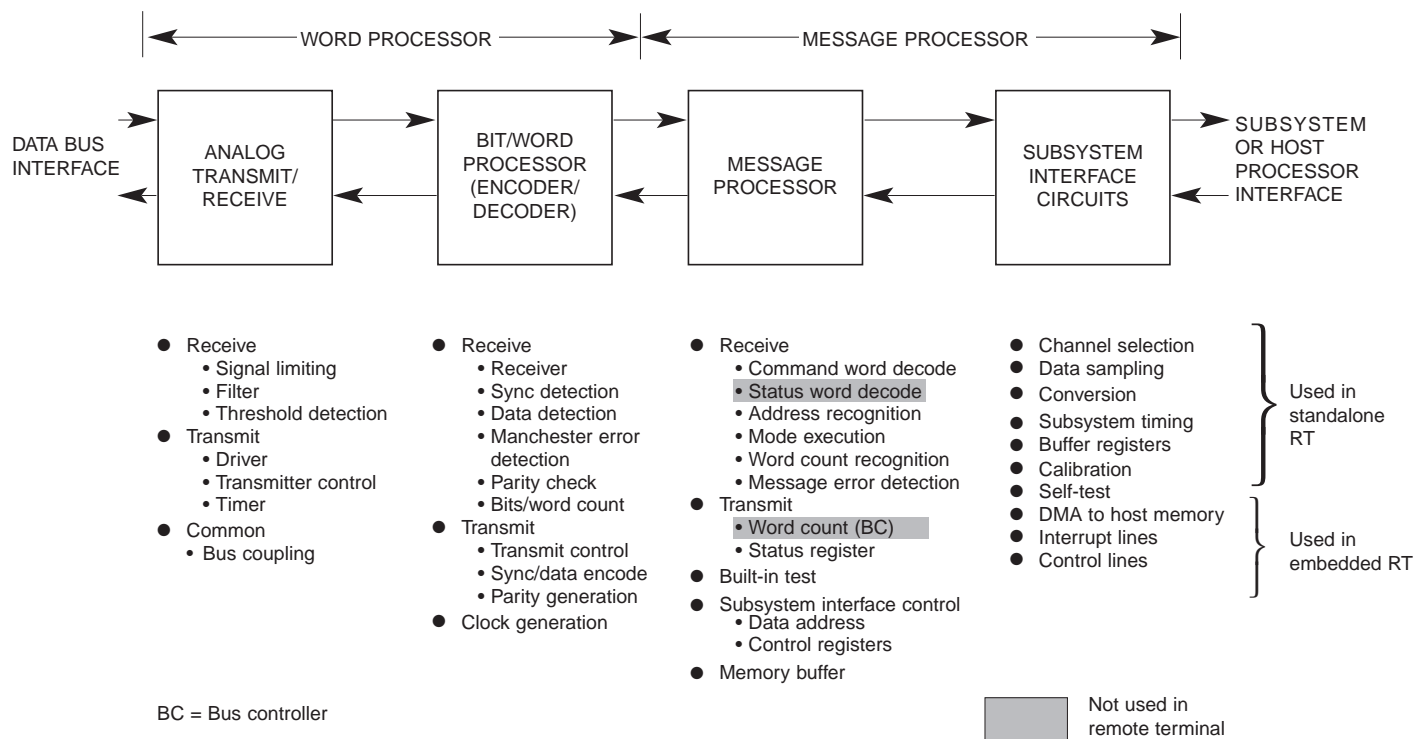
Figure I-4.3 Remote Terminal Functional Elements

serial or parallel or combinations of both depending on the design and the circuit implementation. As stated earlier, a 1553 terminal can be designed to perform both the bus controller and remote terminal functions. However, only terminals with a bus controller requirement should be designed to meet the more complex bus control functions. This section will discuss the common functions associated with all terminals and the unique functions associated with remote terminals and bus controllers. Two types of remote terminals will be discussed: (1) a remote terminal with a direct parallel interface to a simple system with no processor, and (2) an RT with integrated RAM, plus memory management, interrupt, and processor interface logic. The unique features of terminals performing as bus controllers will also be discussed.

4.2.1 Analog Transmitter/Receiver

The analog transmitter/receiver functional element is primarily the analog front end required to interface the terminal's digital logic with the data bus. This section includes the coupling transformer and fault isolation resistors required for a direct coupled connection to the data bus. The 1553 receiver provides low level noise rejection and a digital output compatible with the digital logic that follows in the decoder. The encoder and transmitter drive the biphase modulated signal to form data word formats defined in MIL-STD-1553B. A timer (analog or digital) is provided to cease transmission after a predetermined time (maximum 800 microseconds). This prevents uncontrolled activity on the data bus. The transmitter/receiver portion is generally implemented using a combination of monolithic chips and/or thick-film hybrids. **Today these circuits are widely available in small low cost packages, thus eliminating the need to design your own device.**
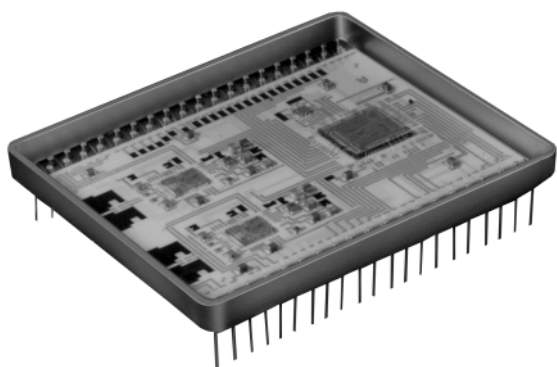
BUS-65142. The BUS-65142 provides an RT interface that is suitable for space and other applications requiring radiation hardness and high-rel screening. Integrating bipolar transceivers and SOS/CMOS protocol logic, the BUS-65142 is suitable for simple system (no processor) applications. It is being used on the Centaur launch vehicle and the Space Station Freedom Program (SSFP).

Figure I-4.1 Bus Elements Capabilities

Capability	Bus controller	Minimum standalone RT (dual bus)	Standalone RT (dual bus)	Standalone RT (multiple bus)	Minimum embedded terminal (dual bus)	Embedded terminal (dual bus)	Embedded terminal (multiple bus)	Intelligent device—RT or embedded (dual bus)	Intelligent device—RT or embedded (multiple bus)
1. Information transfer formats									
a. Controller to remote terminal	X	X	X	X	X	X	X	X	X
b. Remote terminal to controller	X	X	X	X	X	X	X	X	X
c. Remote terminal to remote terminal	X	X	X	X	X	X	X	X	X
d. Mode command without data word	X		X	X		X	X	X	X
e. Mode command with data word (transmit)	X		X	X		X	X	X	X
f. Mode command with data word (receive)	X			X		X	X	X	X
2. Broadcast information transfer formats									
a. Controller to RT(s) transfer	X					X	X	X	X
b. RT to RT(s) transfer	X					X	X	X	X
c. Mode command without data word	X		X	X		X	X	X	X
d. Mode command with data word	X		X	X		X	X	X	
3. Mode codes									
a. Dynamic bus control	X								
b. Synchronize	X		X	X					
c. Transmit status word	X	X	X	X	X	X	X	X	X
d. Initiate self-test	X		X	X					
e. Transmitter shutdown	X		X			X		X	
f. Override transmitter shutdown	X		X			X		X	
g. Inhibit terminal flag bit	X		X	X				X	X
h. Override inhibit terminal flag bit	X		X	X				X	X
i. Reset remote terminal	X		X	X		X	X		X
j. Transmit vector word	X		X	X			X	X	X
k. Synchronize	X							X	X
l. Transmit last command	X		X	X		X	X	X	X
m. Transmit bit word	X		X	X			X		X
n. Selected transmitter shutdown	X			X					X
o. Override selected transmitter shutdown	X			X					X
4. Status bit field									
a. Message error		X	X	X	X	X	X	X	X
b. Instrumentation (set to zero)									
c. Service Request			X	X			X	X	X
d. Broadcast received command			X	X	X	X	X	X	X
e. Busy			X	X		X	X	X	X
f. Subsystem flag					X	X			
g. Dynamic bus control acceptance								X	X
h. Terminal flag			X	X				X	X

ACE YOUR 1553 DESIGN with DDC's Advanced Communication Engine (ACE), Mini-ACE, and Mini-ACE-Plus hybrids, the BU-65179, 65178, BU-65170 series RT only or the BU-61688/9, 61588, BU-61580 series BC/RT/MT. The ACE terminals integrate dual transceiver, encoder/decoder, protocol, memory management, interrupt logic, processor interface, and 64K or 4K x 16 RAM into a 1" x 1" package. The ACE's flexible processor interface supports a myriad of configurations for connecting to a host processor and external RAM. The ACE's flexible architecture serves to minimize processor software overhead for a wide variety of applications.

4.2.2 Encoder/Decoder

The encoder/decoder (see figure I-4.4) is used to analyze the data bits and words required for data transfer on the receiver side of the terminal. The squared up signal from the receiver is input to the decoder, which senses bit timing to decode the sync pattern, data bits, and parity to identify command/status or data words. Bit patterns other than the sync pattern are checked to verofu proper Manchester encoding. The number of bits per word and parity are verified to complete the bit and word analysis. Error conditions are flagged if any portion of a bit or word is unsatisfactory. The encoder or data transmitter section contains the digital logic to establish both command, status, and data words for formatting, including sync, data, and parity generation. Control signals are provided to the analog transmitter/receiver section to indicate appropriate actions. In state of the art terminals, the **encoder/decoder is implemented in a single monolithic, along with the protocol state machine and subsystem interface logic.** Usually

these integrated chips offer the best way to meet this design function.

4.2.3 Protocol Sequencer and Subsystem Interface

The encoder/decoder is transparent to data word type and contents. Therefore, it passes the words with the error indications to the message processor for further analysis. The message processor continues the analysis by performing command word decoding, address validation, data flow direction (transmission or reception), data reception, message length validation, and data storage when receiving data. and data transmission for transmitting data.

A major task of the protocol sequencer is the subsystem interface control. The subsystem interface complexity may vary from a single parallel or serial handshake for communication to a DMA or shared RAM interface to buffer RAM and/or mail memory.

WORD PROCESSOR ← | → MESSAGE PROCESSOR

DATA BUS INTERFACE → ANALOG TRANSMIT/ RECEIVE → BIT/WORD PROCESSOR (ENCODER/ DECODER) → MESSAGE PROCESSOR → SUBSYSTEM INTERFACE CIRCUITS → SUBSYSTEM OR HOST PROCESSOR INTERFACE

- Receive
  - Signal limiting
  - Filter
  - Threshold detection
- Transmit
  - Driver
  - Transmitter control
  - Timer
- Common
  - Bus coupling

- Receive
  - Receiver
  - Sync detection
  - Data detection
  - Manchester error detection
  - Parity check
  - Bits/word count
- Transmit
  - Transmit control
  - Sync/data encode
  - Parity generation
- Clock generation

- Receive
  - Command word decode
  - Status word decode
  - Address recognition
  - Mode execution
  - Word count recognition
  - Message error detection
- Transmit
  - Word count (BC)
  - Status register
- Built-in test
- Subsystem interface control
  - Data address
  - Control registers
- Memory buffer

- Channel selection
- Data sampling
- Conversion
- Subsystem timing
- Buffer registers
- Calibration
- Self-test
- DMA to host memory
- Interrupt lines
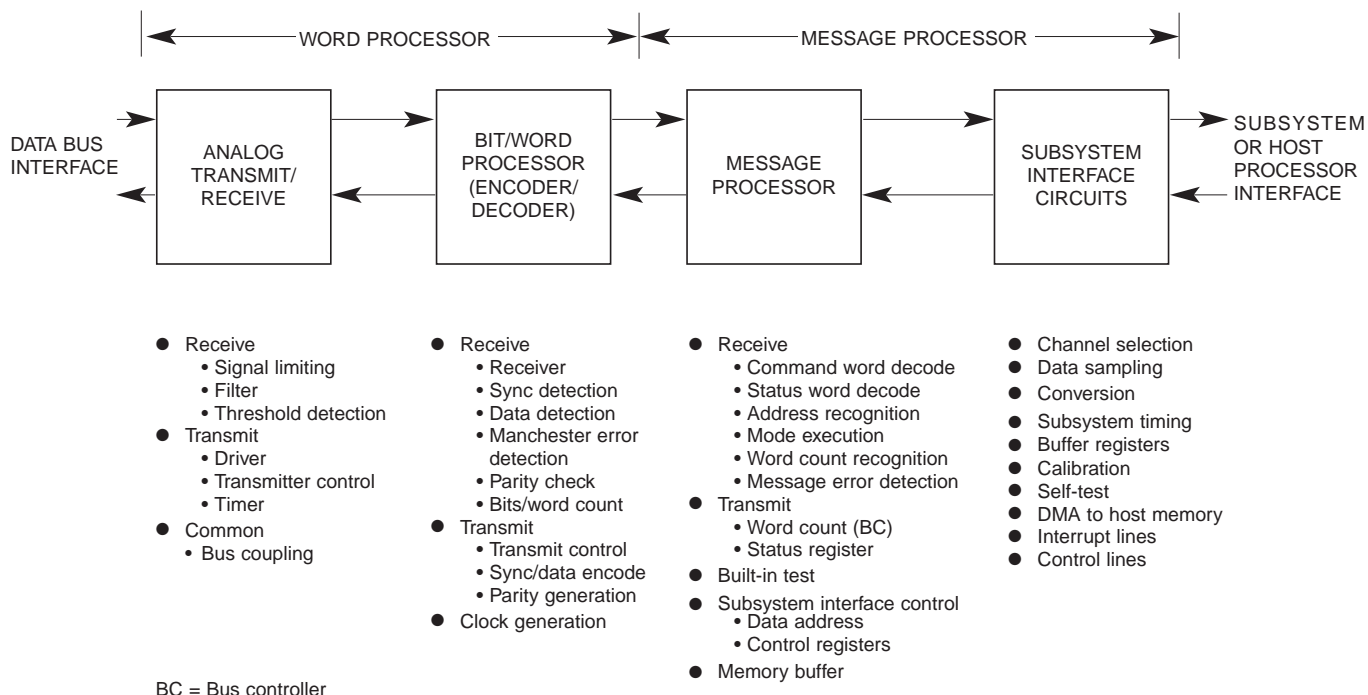- Control lines

BC = Bus controller

Figure I-4.4 Generalized Terminal Functional Elements

Also, for standalone remote terminals, which must interface to simple subsystems, the subsystem interface may control and interface with a wide variety of signal conditioners (A/D converter, D/A converters, input and output discretes, serial and parallel channels) either directly or through a controller. For example, the STIC (BUS-65153) provides an 8/16-bit parallel bus for interfacing to a wide variety of simple subsystems. Most of these systems do not contain a

microprocessor. Moreover, the ACE series RT-only (BU-65179, BU-65178, BU-65170) and BC/RT/MT (BU-61688, BU-61689, BU-61588, BU-61580) terminals may be interfaced to a host processor and RAM through a variety of configurations: 8-bit and 16-bit internal shared RAM, 16-bit shared RAM (transparent configuration), DMA, and dual port RAM. In addition, the ACE series terminals have built-in architectural features to off-load the operation of the host CPU, control and monitor message timing, ensure data consistency, and facilitate bulk data transfers. The ACE terminals include a number of features to support both polling-driven and interrupt-driven software methodologies
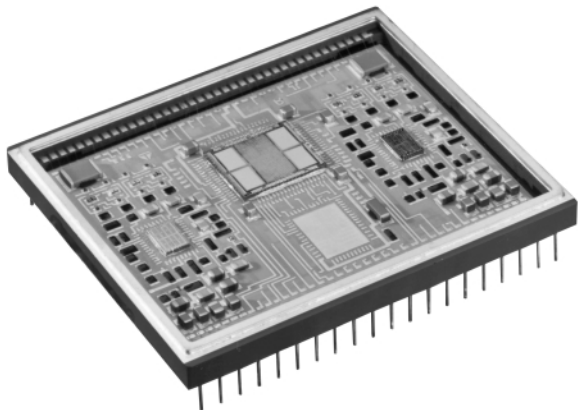
BU-61590 Universal Terminal. The BU-61590 integrates a dual McAir/universal (sinusoidal) transceiver, encoder/decoder, BC/RT/MT protocol, processor interface, and 4K x 16 RAM in a 78-pin 1.87 x 2.1 inch hybrid. The BU-61590 supports all 1553 protocols: MIL-STD-1553A/B, McAir A5232, A3818, and A5690; G.D. 16PP303 (F16), Grumman SPG151A, and STANAG 3838.
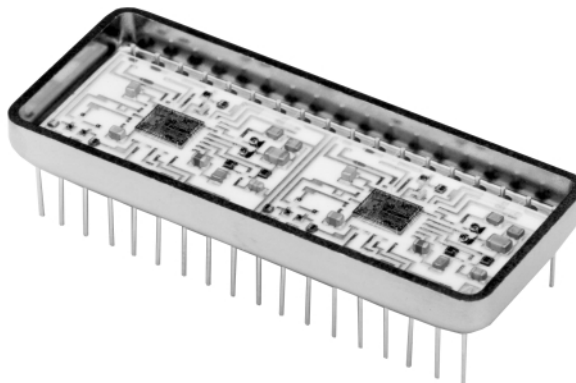
BUS-63125. This industry standard dual transceiver is now lower power and available as DESC No. 5962-87579-01XX, which eliminates the need for non-standard part approval.

### 4.2.3.1 Protocol Sequencer with Remote Terminal Capability

To initialize a remote terminal requires a mechanism for establishing its unique address. Two ways are available; establishing the address using an external strapping of pins to "open circuits" or "ground circuits" to establish a five bit address or host (subsystem or processor) software loaded. Each method has its place. For intelligent subsystems with a capability to change the software and for processors with software loads, software initialization of the terminal address is the most appropriate. **External pin addressing in software systems is required to identify acceptable software load and the remote terminal address used for loading the device if the load is supplied via the 1553 bus. For sensors or subsystems with fixed (unchanging) programs and for standalone RTs the aircraft wiring can supply the terminal's unique address.** Notice 2 requires that remote terminals must be capable of being assigned a unique address (address 11111 is still reserved for broadcast for those systems implementing this function) and that all addresses be established via an external connector to the remote terminal. The notice also specifies that changing the address must not require a physical modification or manipulation of any part of the remote terminal (i.e., changing of a memory device containing the terminal's address). No single point failure in the address strapping should cause a terminal to validate a false address. This means that, as a minimum, the remote terminal should include a parity bit along with the five address lines and should perform a parity check on the programmed address. Notice 2 requires that an RT detect all single failures of the RT address inputs. As a minimum, that address should be validated during power-on initialization.

Figure I-4.5 illustrates an interface between a Remote Terminal (RT), (BUS-65153) and a minimum complexity system. That is, a system that does not include a microprocessor. Such systems typically include A/D and D/A converters, switch closures, and simple displays. The STIC incorporates an 8/16-bit DMA parallel data interface, plus a 12-bit address bus. As shown, the STIC may be interfaced to these types of systems with a bare minimum of "glue" logic. Implementing an interface between a 1553 bus and a simple system without the use of a CPU eliminates not only the cost of the processor, but of ROM, RAM, other logic, and software development. For military programs, the latter can be considerable.

Most current subsystems incorporate one or more microprocessors. These range from low-end microcontrollers, used for self-test and simple control functions, to 32-bit processors such as 68040s and 1960s. In either case, an RT with RAM and other support functions, such as the ACE (BU-65170), is generally the preferred solution. The support functions typically include software programmable command illegalization, memory management, message logging, interrupt logic, RT BIT Word implementation, and processor/memory interface. The ACE RT provides all of these functions. In addition, MIL-STD-1553B Notice 2 requires that an RT provide a mechanism for distinguishing between non-broadcast and broadcast messages. With the BU-65170, this includes the capability for storing data words received from broadcast messages in a separate area from non-broadcast data words.

Figure I-4.6 illustrates three methods of RT memory management provided by the ACE. That is, it shows different methods for storing data words for individual RT subaddresses. In each case, there
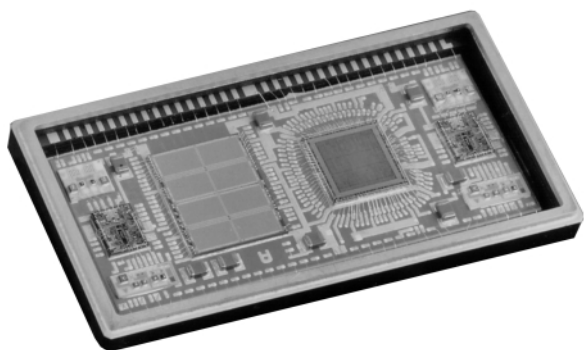
BU-61582. The BU-61582 "SP'ACE" provides a complete integrated BC/RT/MT terminal that is suited for space and other applications requiring RAD hardness and high-rel screening. Integrating bipolar transceivers, RICMOS protocol and interface logic, and 16K x 16 of RAD-hard RAM, the BU-61582 is suitable for interfacing between a host microprocessor and a 1553 bus.
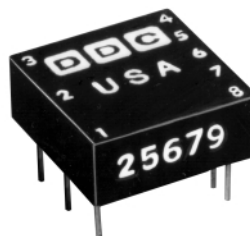
BUS-25679. DDC's BUS-25679 is typical of the many standard transformers available to couple the transceiver or integrated terminals to the 1553 databus.
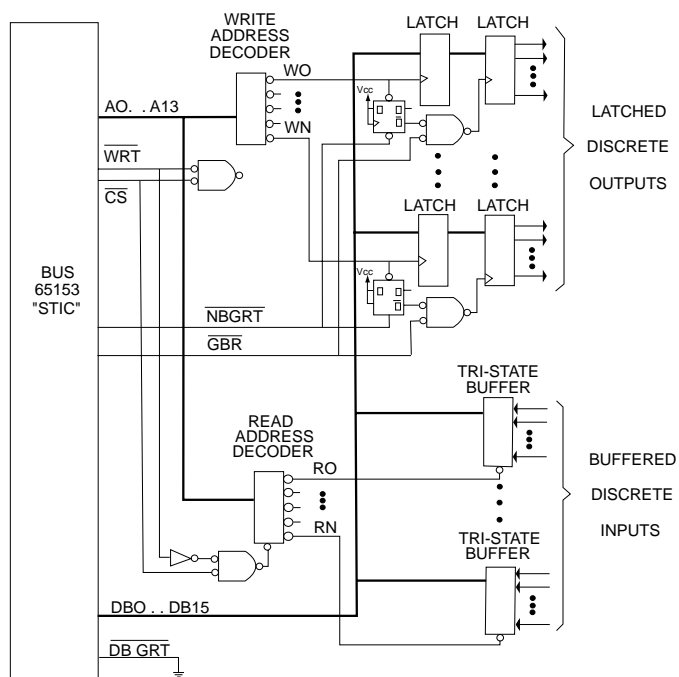
Figure I-4.5 RT-to-Simple-System Interface

is a lookup table pointer used to locate the start of a message data block in RAM. The lookup table pointer is initialized by the host CPU. For each mode, note that a 4-word descriptor stack stores information regarding bus channel, message completion and validity status, error conditions, time tagging information, the actual Command Word, and a copy of the pointer that was read from the lookup table. In the single message mode (Fig. 1-4.6 [a]), the same block of data words is overwritten for receive messages, or overread for transmit messages for a given subaddress. For received messages, the use of subaddress double buffering (Fig. I-4.6 [b]) serves to ensure data consistency by allowing easy CPU access to the last, valid, received data block for a given subaddress. For bulk data transfers such as program downloading the use of the circular buffered mode (Fig. I-4.6[c]) off-loads the host processor by having the RT, rather than the CPU, update the value of the lookup table pointer after valid messages. The purpose of the double buffer and circular buffer features is to off-load the operation of the host processor. For example, the use of circular buffers eliminates the need for the host CPU to be concerned with individual messages, message errors, or retried messages. To receive a bulk data transfer, it is only necessary for the host to wait for an interrupt request (for circular buffer rollover) and then read the multimessage block of received data words.
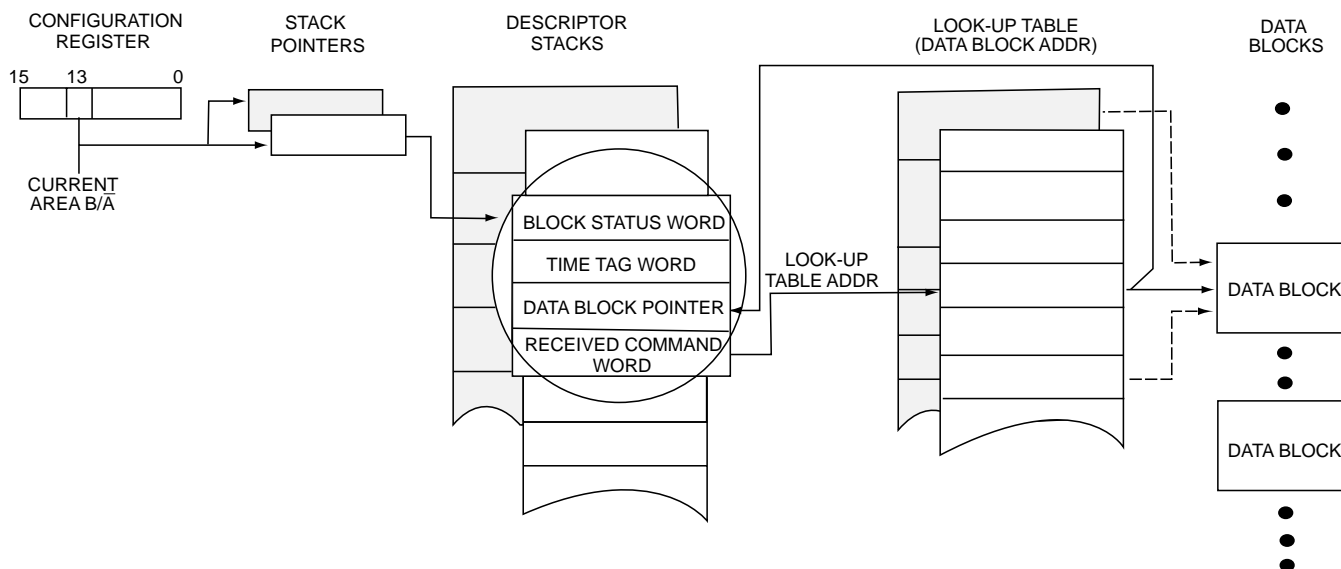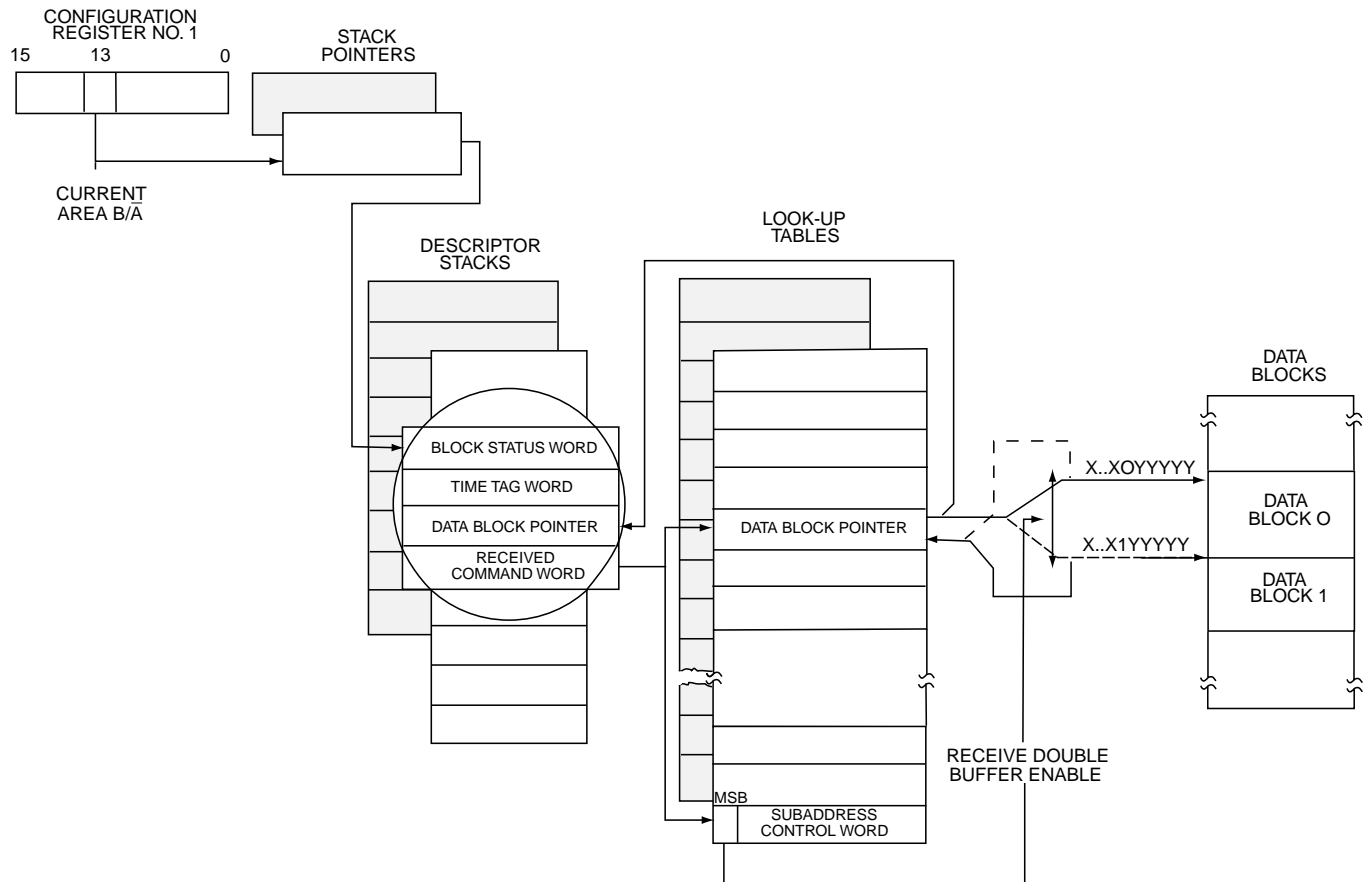

Figure I-4.6a RT Stack and Memory Management Operation: Single Message Mode

Figure I-4.6b RT Stack and Memory Management Operation: Double Buffer Mode
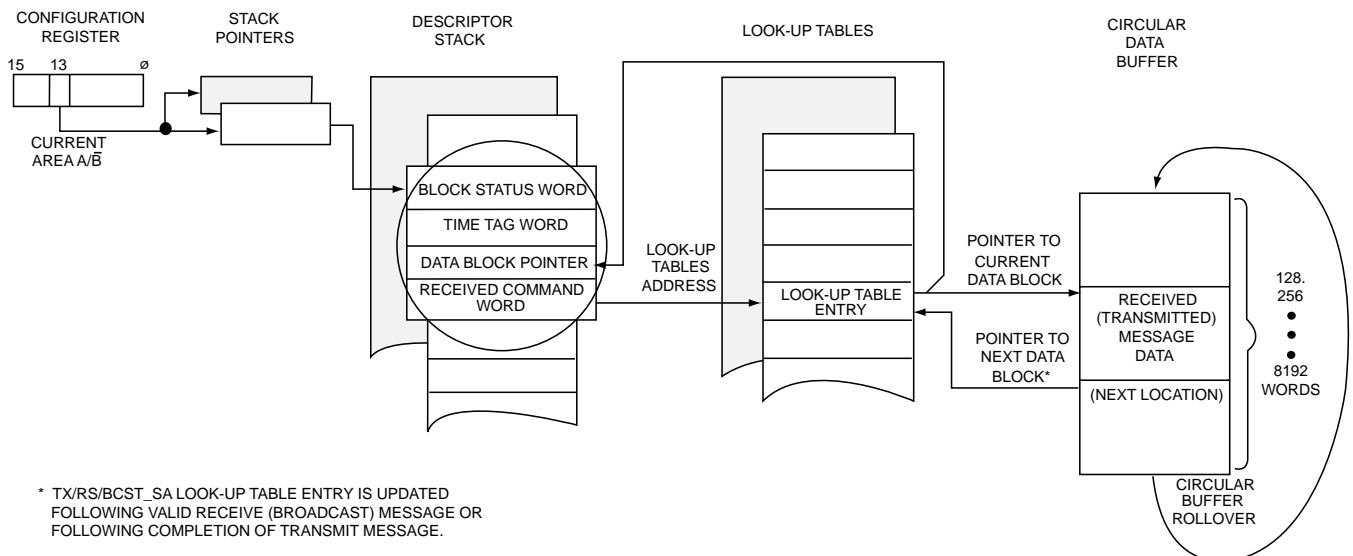

* TX/RS/BCST_SA LOOK-UP TABLE ENTRY IS UPDATED
  FOLLOWING VALID RECEIVE (BROADCAST) MESSAGE OR
  FOLLOWING COMPLETION OF TRANSMIT MESSAGE.

Figure I-4.6c RT Stack and Memory Management Operation: Circular Buffer Mode

4.2.3.2 Protocol Sequencer with BUS Controller Capability

In addition to the capabilities identified above for the remote terminals, bus controller functioning word processors must contain the capability to develop and transmit command words and analyze status words. The same interface control and handshaking is necessary with the bus controller that was needed for the RT. However, since the bus controller can take on many different levels of complexity and the partitioning between hardware and software is not as clear, a general discussion is provided in section 5 on this subject.

4.2.4 Bus Coupler Design

Bus coupler networks, separate from the terminals, are required by 1553B when connected to the data bus via "long stubs." A long stub is defined to be greater than 1 ft. Direct coupling, which can be implemented without a separate coupler box, is defined for short-stub connections of 1 ft or less (Note: Notice 2 requires that for Navy applications, terminals must have both transformer and direct coupled stub connections externally available, such that either may be used; and that for Air Force and Army applications, ONLY transformer coupled stub connections will be used.) The long-stub coupler network incorporates isolation resistors and a coupling transformer. The isolation resistors are located in the terminal for the direct-coupled case, thus eliminating the need for a separate coupler box if a reliable shielded splice can be made. In most cases, the bus connections can be spliced in the terminal connector.

The coupler-transformer characteristics are very important to the signal integrity and noise performance of the data bus system. **The purposes of the coupler are to; (1) provide isolation of the main bus for fault conditions on the stub or in the terminal, (2) provide reduced bus signal distortion effect by increasing effective stub Impedance, and (3) provide termination of the stub when transmitting from the terminal.** The isolation resistors and the transformer turns ratio provide the benefits listed above. The terminal input and output specifications for the transformer coupled and direct-coupled connections are separated in 1553B, because of the effects on signal levels and impedance caused by the transformer turns/ratio.

The use of transformers as a means of coupling baseband signals in a balanced transmission line system has proved to be an extremely effective approach. When designed and used properly, trans-

formers can perform effectively to maintain isolation and achieve signal integrity at a relatively low cost and with high reliability. They are not extremely lossy and can readily achieve high common mode rejection.

Coupling transformers are presently commercially available from several sources thus freeing the designer from this design task. Usually the manufacturer of the transceiver will recommend a transformer for use with the device. One point not to be overlooked is temperature rating. Most transformers have a lower maximum operating temperature than the other associated circuit components.

A review of transformer design considerations is given in Chapter 4 of the "MIL-HDBK-1553 Multiplex Applications Handbook."

5.0 BUS CONTROLLER IMPLEMENTATION

At the lowest level, the function of a bus controller is to cause data to be transmitted over the 1553 bus. For the purposes of this discussion, the collection of hardware and/or software that implements the transfer of information over the 1553 bus will be referred to as the Bus Interface Unit (BIU). The bus interface may be implemented via several devices or by one of the sophisticated 1553 control units currently available on the market. These devices automatically wait a specified time. (Per MIL-STD-1553B, this time is 14µs minimum. In current integrated bus controllers, there is usually a programmable response timeout to accommodate long buses; for example, the ACE terminals provide a choice between 18, 22, 50, and 130 µs for a response from a Remote Terminal. A dedicated processor or state machine may be utilized to off load the host CPU. In integrated terminals such as the ACE, the BC state machine is a built-in feature. This protocol state machine handles the tasks of message formatting, intermessage gap times, Status Word bit masking, error detection, retries, memory management, and interrupts. The host CPU has the responsibility of downloading 1553 messages to the BIU and transferring data between the BIU and other system elements.

As a result, today's designers have a choice of two basic types of 1553 Bus Controller Units:

(1) Single Message Processors – This type of 1553 interface outputs individual messages without requiring overhead from the host CPU. The Bus Interface Unit determines the validity of the individual message and/or looks for anomalies in the status word. The host CPU is responsible for determining the next message to be transmitted,

or the error recovery scheme to be implemented.

(2) Multiple Message Processors – This type of 1553 interface processes a series of messages over the 1553 bus without host CPU intervention. Each message is validated by the Bus Interface Unit. If no errors were detected, the Bus Interface Unit will automatically issue the next message in the frame. If an error condition is detected, the Bus Interface Unit may halt 1553 transmissions or may continue normal operation until the end of the frame. Alternatively, the BIU may perform automatic retries on failed messages. Various error handling techniques may be initiated by the hardware, or may be left to the subsystem to resolve.

5.1 Single Message BIU's

Single Message BIU's essentially require that the subsystem initiate and evaluate each message that is transferred across the 1553 bus. These devices are usually less expensive than the more sophisticated Multiple Message BIU's. However, they require more host CPU overhead. With the low duty cycles typically encountered in 1553, it may seem that the CPU can easily handle both the 1553 activity in addition to other applications. The problem is that application software and bus traffic frequently grow in proportion with each other. The growth in bus traffic decreases the subsystem's processing capacity at the same time that more applications software is required to process the new data. This "double growth requirement" must be considered when performing the sizing and timing analysis of a Single Message type of BIU.

5.2 Multiple Message BIU's

In a Multiple Message BIU such as the ACE, once a START command is issued to the BIU, the subsystem is free to handle other activities. It does not need to be involved in the transfer of each individual message over the 1553 bus.

To accomplish this, the subsystem typically stores the 1553 messages in some portion of RAM that is also accessible to the BIU. The subsystem indicates which messages are to be transmitted by loading a stack with the addresses of the data blocks containing the messages, in the order in which they are to be transmitted. Alternately, the subsystem may chain the messages together, so that each message points to the next message to be transmitted. See Figures I-5.1 and I-5.3.
In either case, the subsystem must indicate how

many messages are to be transmitted. This may be accomplished by loading a reserved word of the RAM with a message count, or by loading an "end of list marker" in the last message's forward pointer.

Most current BC architectures allow programming of the intermessage gap times between individual messages. Some BCs, such as the ACE, include the added capabilities to start a BC frame from an external signal and/or automatically repeat a programmed frame of messages with a programmable frame (repetition) time.

5.3 Message Headers

Regardless of the type of chaining architecture implemented, the BIU requires some "overhead information" for each message being processed. Typically, this consists of a control word indicating the type of 1553 message (Broadcast, Mode Code or RT to RT), a location to be used for storing the message's Time Tag, the channel on which the message is to be transmitted, the area of RAM to be used for storing or retrieving the associated data and command words, and the time to the start of the next message. Additional information regarding error recovery options or interrupt control may also be included. (Error recovery techniques will be discussed in more detail in section 5.6).

In Multiple Message BIU's, the format of this header information is specified by the vendor. In Single Message BIU's, this header structure is defined by the user, and is implemented in software via the host processor.

The actual 1553 command words may be stored with the overhead information, or it may be stored with the data words associated with the message, depending on the architecture of the particular system. One or two words may be reserved for the command word(s) depending on the type of message (RT to RT for instance).

Most 1553 messages are time tagged. This value may be the actual contents of the system clock at the time the message transaction began or completed, or it may be an a system defined 16 bit number. In any event, it is linked to the particular message either by a pointer or by storing the actual Time Tag with the rest of the message's overhead information.

5.4 Stacks vs. LInked Lists

As previously mentioned, the address of the message(s) may be stored in either a stack configura-
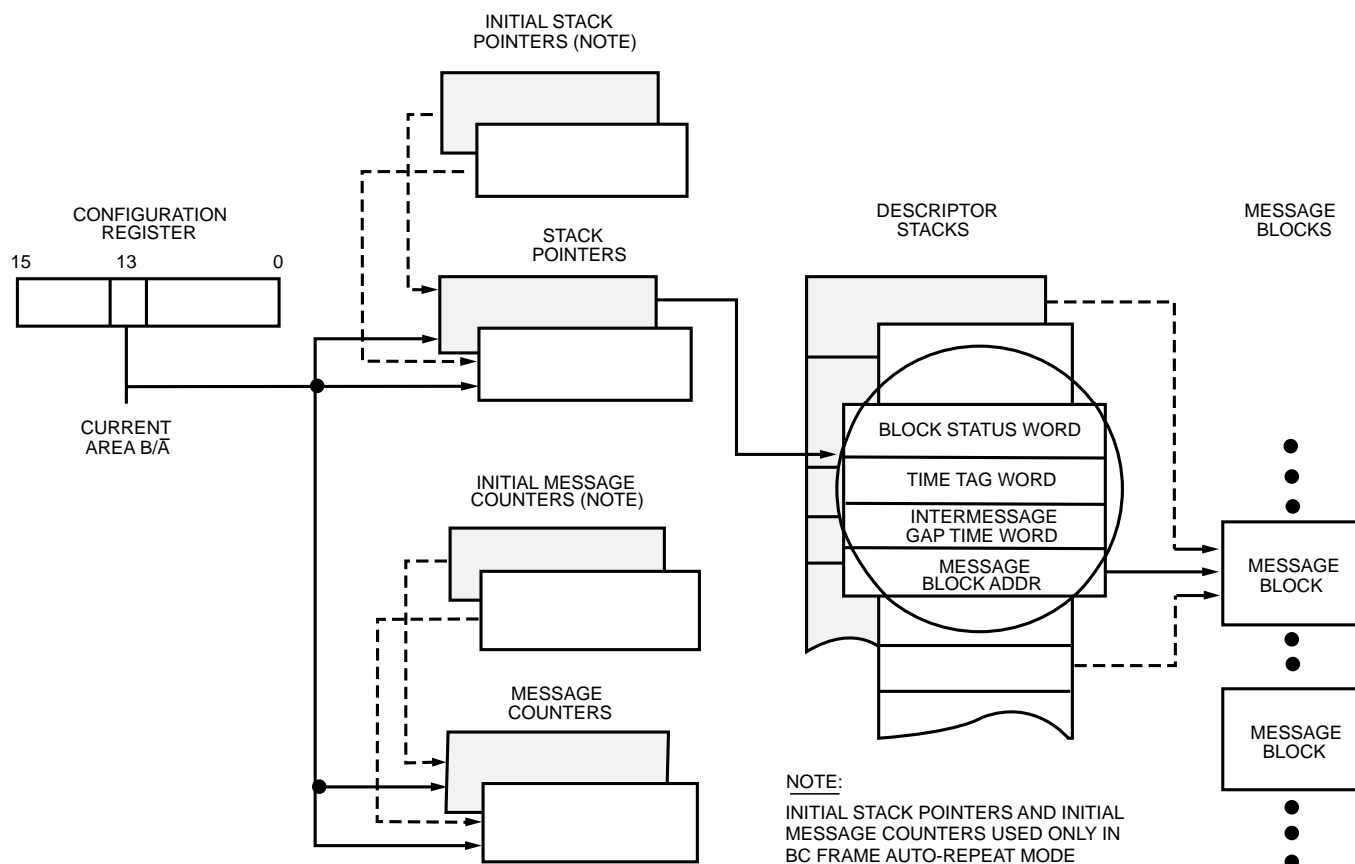
Figure I-5.1 BC Mode Memory Management

tion or in a linked list. Each has advantages and disadvantages. A stack is simple to implement, and facilitates the implementation of minor frames. Separate stacks can be used for each minor frame. The host simply initializes the stack pointer to the appropriate stack each time the particular minor frame is to begin. See Figure I-5.4. Alternatively, the user can imbed the minor frames in the contents of the stack itself (see Figures I-5.1, I-5.2, and I-5.5). As shown in Figure I-5.1, the INTERMESSAGE GAP TIME word is stored in the third entry of the message block descriptor in the stack. In this way the CPU is able to schedule the time sequence of all messages to be processed. As illustrated in Figure I-5.2, this includes the structuring of minor frames. In a typical system, the minor frame time might assume a value like 5 ms. This supports synchronous message periodicities of 5 ms (up to 200 Hz); in this scenario, other message rates could be 10, 20, 40, 50, and 100 Hz. The use of a major frame parameter (Figure I-5.2[a]) allows the overall frame of messages to be automatically repeated by the bus controller without CPU intervention. For example, the ACE BC's major frame time is programmable with 100 μs resolution, to a maximum value of 6.55 seconds.

For instance (reference Figure I-5.5), assume that the frame consists of Messages 1, 2,1, 3, 1, 4,1, 4 and then the sequence repeats. The host can simply load the messages into the stack in that order, and issue a START command at the beginning of each major cycle. A disadvantage of a stack architecture is that it is not easy to insert messages into the middle of an existing stack. A typical application would be to insert an error recovery procedure into the message stream upon detection of a no response or other error condition. The host would have to physically change the value of the stack pointer to point to the new message stack. When the error recovery messages completed, the host would have to reinitialize the stack pointer to point to the next message in the original stack.

In a linked list architecture, each message points to the next message to be transmitted. This makes it very easy to insert messages into the middle of the message stream. The bus controller's software simply replaces the forward pointer of the particular message with the address of the message to be inserted. The last message to be inserted would then simply point to the next message in the original list. See Figure I-5.6.
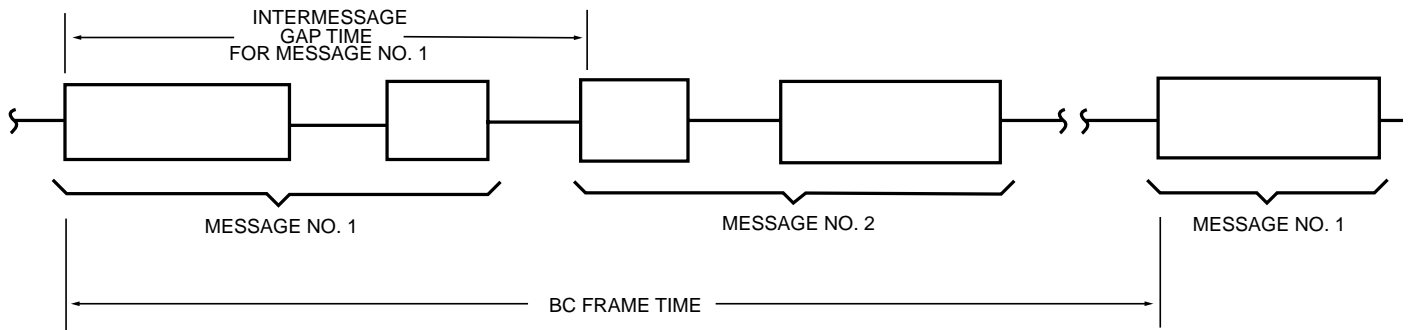
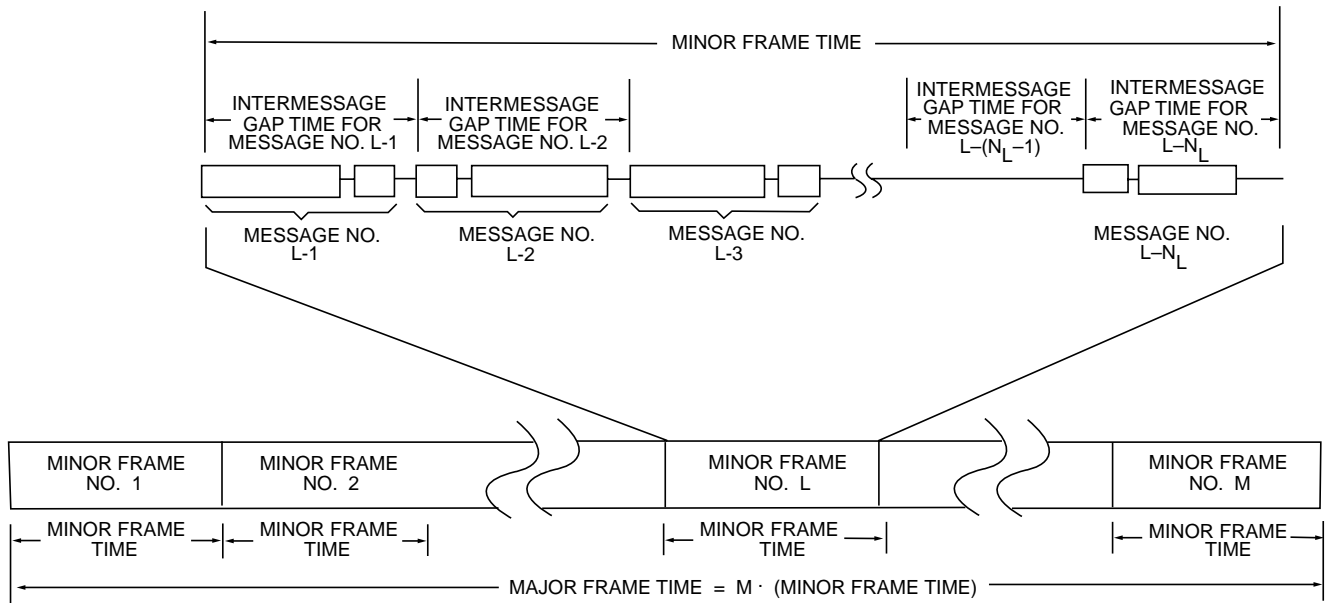Figure I-5.2a BC Intermessage Gap and Frame Timing

Figure I-5.2b Intermessage Gap Times for Minor Frame Formulation

However, it is not as easy to implement the repetitive minor frames with a linked list architecture. For instance, let's assume the that the frame consists of Messages 1, 2, 1, 3, 1, 4, 1, 4 and then the sequence repeats. The host can set the forward pointer of message number 1 to point to message number 2. The problem arises with the forward pointer for message number 2. If it points back to message number 1, the host will have set up an infinite loop since message number 1 points back to message number 2. One solution is to make another copy of message number 1 and store it in a different portion of RAM. Message 2 can then point to this copy of message number 1. See Figure I-5.7. That copy would then point to message number 3. The problem arises again with the decision of what to do with the forward pointer for message number 3. Additional copies can, of course, be made. However, this architecture requires much more memory than does the stack approach, as well as requiring much more system overhead to maintain the individual messages.
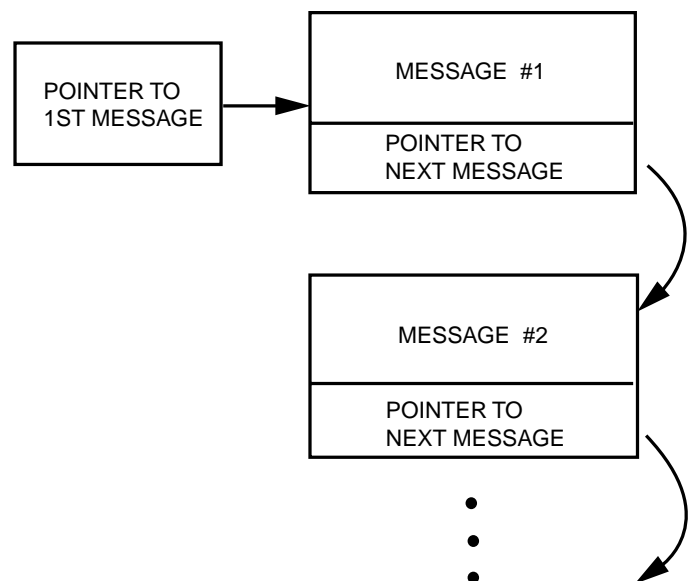

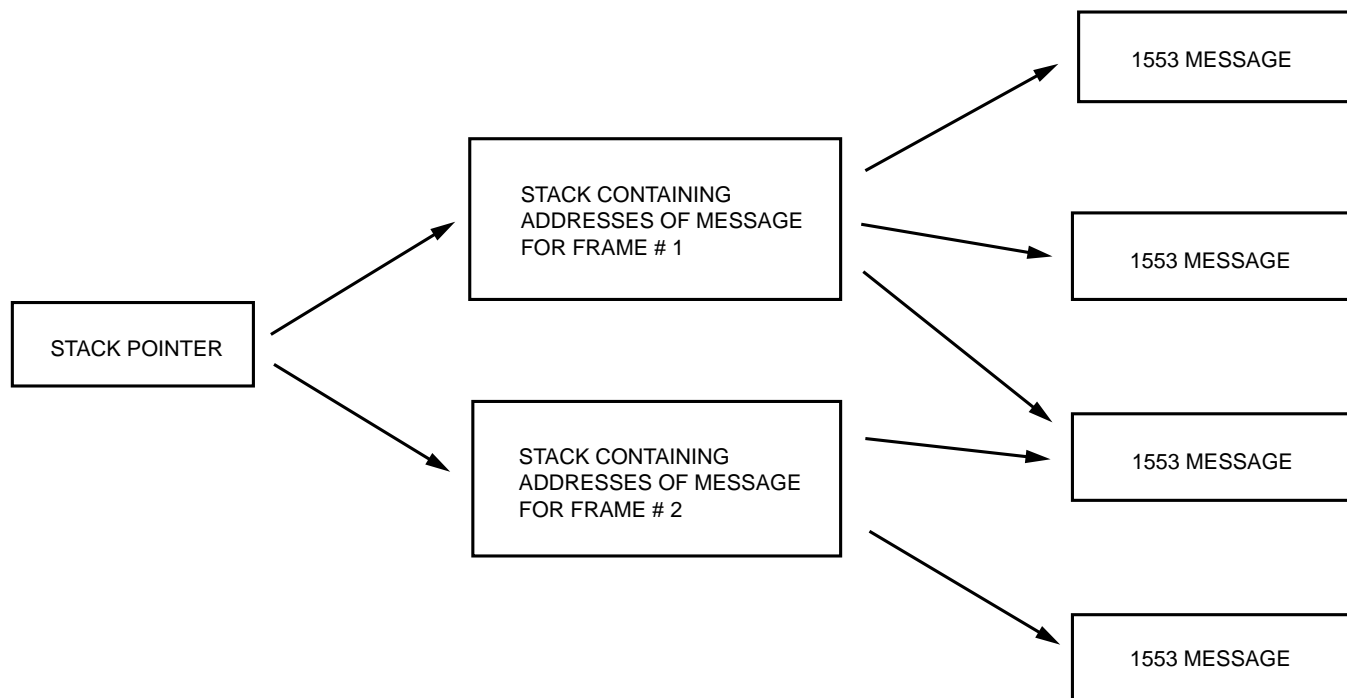Figure I-5.3 Linked List Architecture

Figure I-5.4 Use of Stacks to Implement Minor Frames

A linked list approach is somewhat more complex to set up. The bus controller software must embed the addresses of the next message in the body of each individual message block. As these messages may be scattered throughout memory, the possibility for error is somewhat greater than it is in the case of a stack, where the host simply loads a fixed portion of RAM with the starting addresses of the individual messages. With a stack, it is also possible to see at a glance what messages will be transmitted, and in what order. A linked list typically requires a user to perform a memory dump of a much larger area of memory to trace the 1553 setup.

5.5 Status Word Analysis

The primitive bus controller must analyze each bit in the status word to determine if any extraordinary action must be taken. If several different 1553 protocols are used, the controller must differentiate the significance of the bits and act appropriately. Because 1553A (F-16) allowed the message error bit to set when the status word was returned following message transmissions, that bit must be checked for each 1553A type transmission. The type of protocol (1553A, 1553B, F-16, etc.) will be indicated by either a field in the message header control word if each message can be a different protocol or by a control field accessible to the bus control software for block changes. **For example, with the ACE bus controller, the mode of operation (-1553A or -1553B) is programmable on a message-by-mes-**

sage basis. If a message has been designated for the "-1553A" mode, a Status Word with the Message Error bit set is considered a failed message, that can be retried. Because of the large inventory of 1553A equipment, the mixed mode of operation will exist for many years. If a message error is indicated or if the status word is withheld the message is to be retried according to the established procedures.

If the service request bit is set, then the bus controller must either perform an interrogation to determine the specific service desired (transmit vector word mode code) and present the data to the main bus controller for a decision, or act upon the request based upon a table of actions known to the bus controller. See paragraph 3.6 for a detailed discussion of the protocol. The terminal flag and subsystem flag indicate faulty equipment. These problems are referred onto the main bus control program, unless an option exists to mask the problem (using an interrupt or to provide a branch to a set of instructions, which collects data or resolves the problem). It may also be a requirement to pass these two flags onto applications software such as a fault detection, maintenance, or caution/warning, or other general applications programs. In addition to these two flags, the terminal address might also be required so as to be able to identify the faulty terminal. Since the data was received with no errors (for the fact a status word was received), the applications software may use the subsystem flag as one of its validity checks in determining

whether or not to use the data. For these reasons, some applications store the status word along with the data words in the memory buffer. In other applications, if the status word and/or tag words are stored, they should be separated from the data in order to allow contiguous data blocks of more than 32 words. The designer should analyze the systems requirements and determine the method of storage to meet his own requirements.

The remainder of the flags (dynamic bus control acceptance or broadcast command received) are normally passed to the main bus control software as a special condition, but the bus control software should be used in performing dynamic bus allocation as an operational function. The busy bit is normally handled the same as an error, since a retry could be quicker than the analysis to determine that the device was busy rather than an error had occurred. By the time the retry is initiated, the device should have removed the busy bit.

Current BC designs off-load the host processor by performing Status Word analysis. For example, the ACE BC can be programmed such that the individual Status Word bits for an RT response may be masked ("care" or "don't care") on a message basis. If a "Status Set" condition occurs (one or more unmasked bits are set), the BC can be programmed to retry the message and/or interrupt the host CPU.

5.6 Error Handling

The extent to which specific error handling is included in the bus control software is a matter of design and the level of protocol, which is embedded within the bus controller. It is difficult to embed specific error handling and recovery procedures because each application has different needs.

The bus controller is responsible for the execution of the parameters indicated in the channel control words. Certain error handling activities are implicit for these parameters. **The bus controller Is responsible for detecting that a message has not been completed; either by hardware indicating a bit error, word count error, Manchester waveform error, or lack o1 a status word.** Then the bus controller will time out and request the status word (transmit status word mode code) or simply retry the message according to the rules given in the message header blocks. The most commonly used procedure is to retry once on opposite bus before trying next message in the list. Many variations are available once it has been determined that the message has failed to pass the retry procedure. These issues are discussed in Error Recovery (see paragraph 5.8.4).
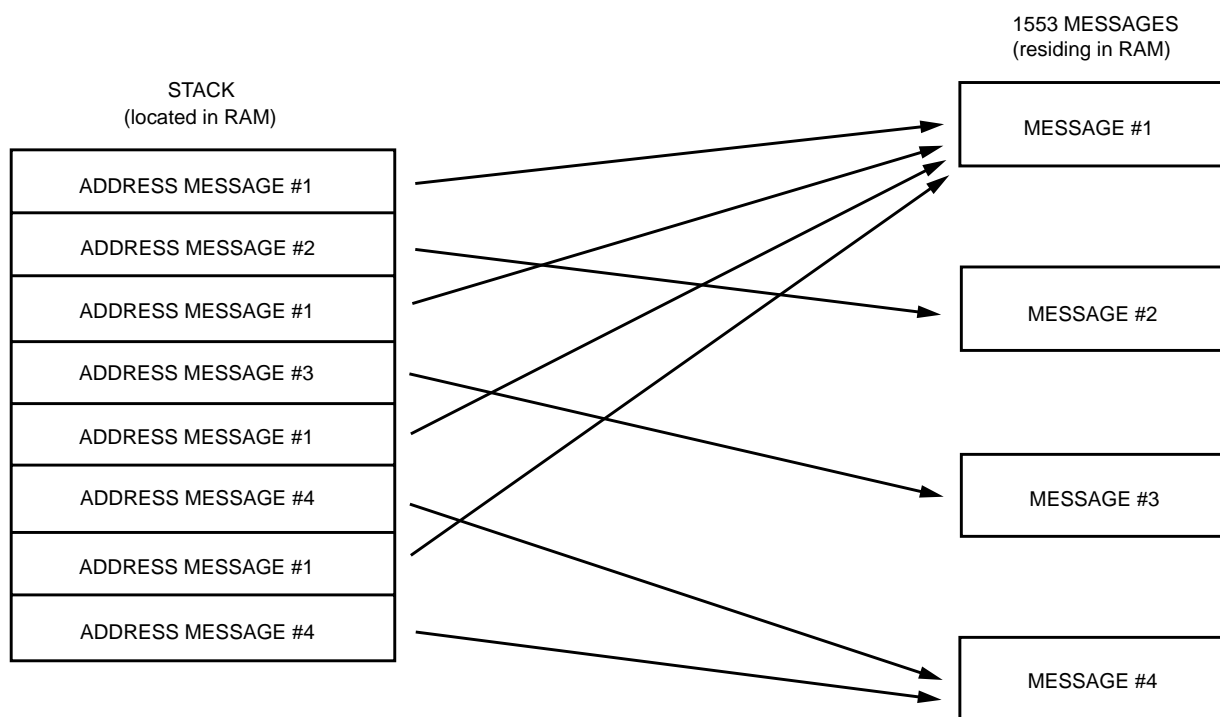
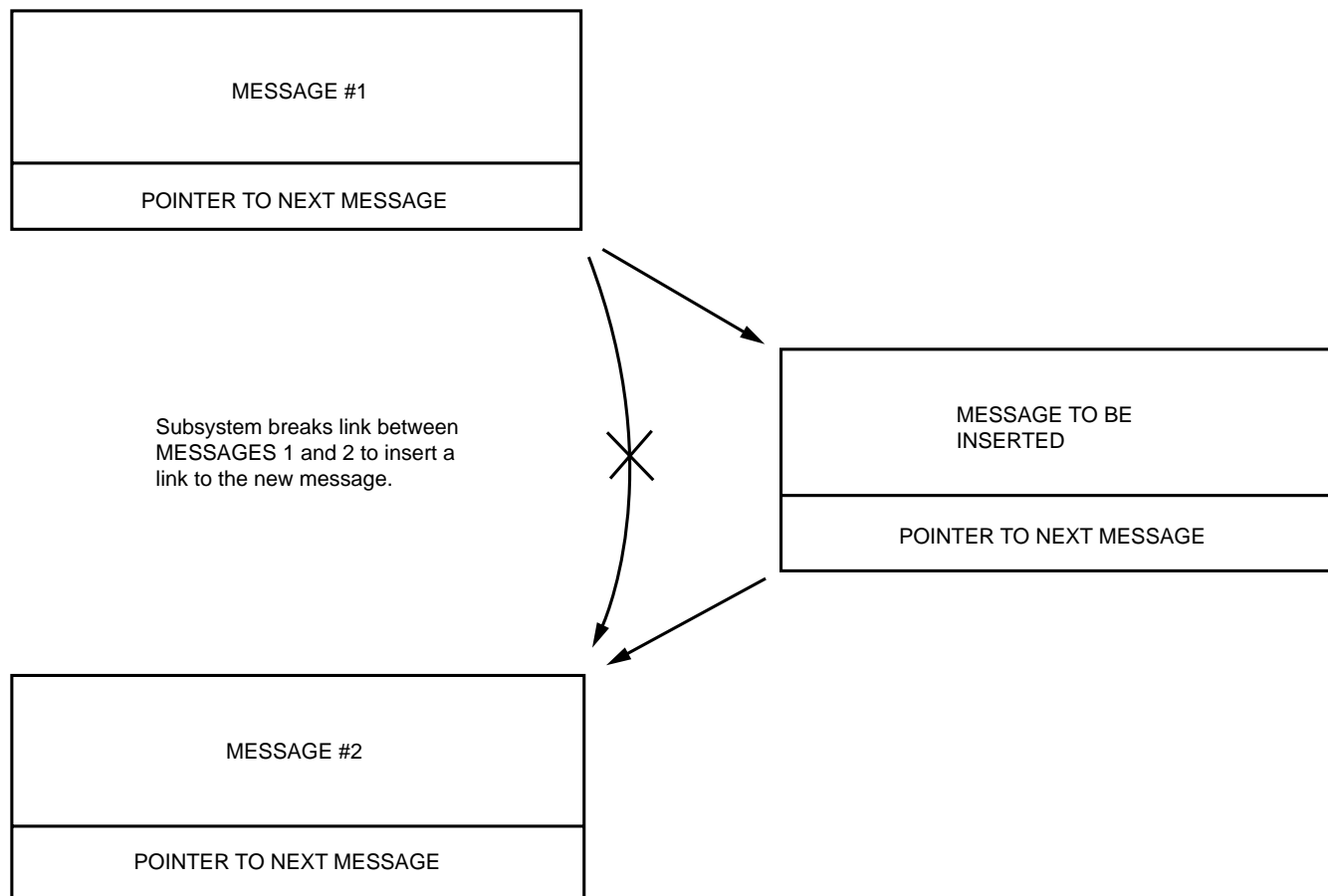Figure I-5.5 Alternate Use of Stack to Implement Minor Frame

Figure I-5.6 Inserting a Message Using a Linked List Architecture

5.7 Tag Words

In time critical functions (e.g., flight controls), it is often necessary to "time tag" the data such that the applications program has an indication as to the data latency or sample period of the data it is processing. In some applications, the "tagging" of the data occurs at the source with the tag word being inserted (usually in the first data word) to the message containing the data. During bus transmissions, this would be processed as normal data and passed to the applications program for processing.

In other applications, the terminal receiving the data will time stamp the data itself, usually using its systems clock as the tag. This tag word is then stored not in the data buffer with the data, but in an accessible location to the applications program, usually the Data Descriptor Block (see Figure I-3.14). If multiple buffering schemes are implemented (see paragraph 3.7.2), then a tag word for each buffer would be required. One other tag word approach is used, where the receiving remote terminal uses the minor cycle number, mes-

sage validity indication, and an active buffer indication. These tag words are collected in a common buffer area for the applications software to examine when it begins processing of this minor frame data. In hierarchical architectures, where data must be passed between bus networks, data latencies can expect to be increased.

Terminals "tagging" the data, which are functioning as the gateway, present a problem. It may be necessary for this tag word to be appended to the data being passed through. In order to accomplish this, the tag word must now be stored with the data in the data buffer. By the terminal appending this tag word, the maximum word count of the original message is limited to 31 words.

The ACE BC provides a time tag word for every message processed. The 16-bit time value is stored at the beginning, and again at the end of every message processed. The resolution of the time tag is programmable from among 2, 4, 8, 16, 32, or 64 µs/LSB.
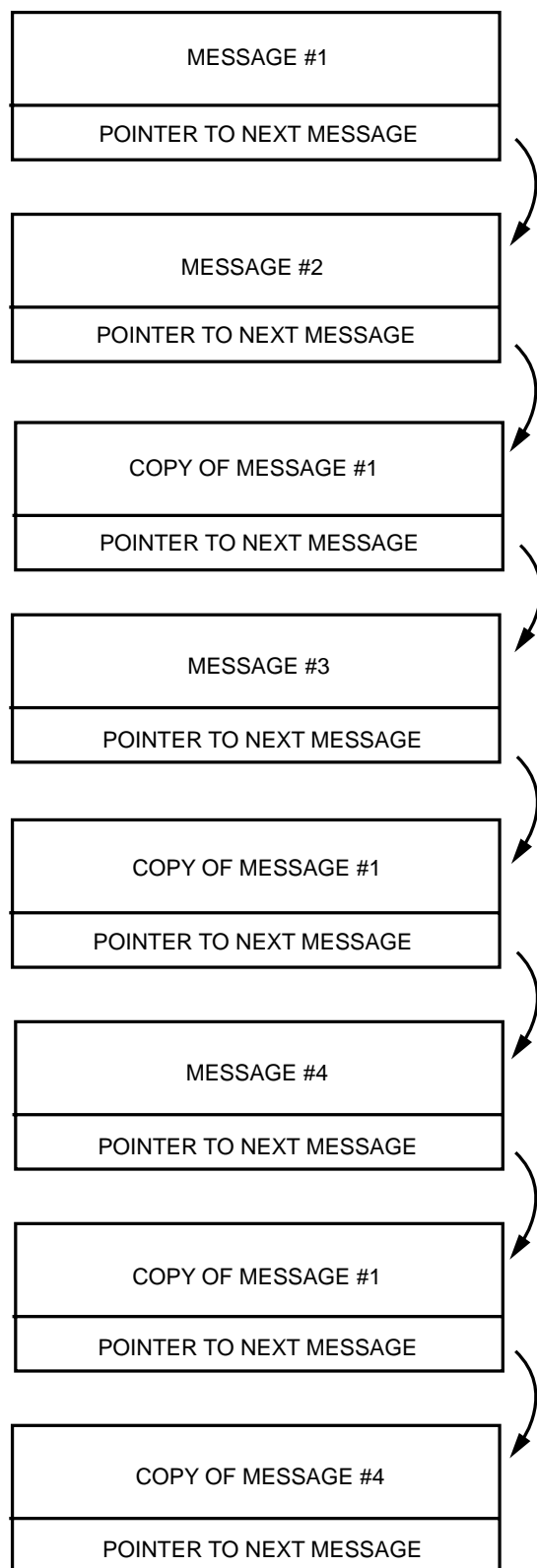
Figure I-5.7 Use of Multiple Copies of Individual Messages to Implement Major Frames

5.8 General Software Considerations

Many of the functions previously discussed could be implemented as separate BIU software or BIU hardware/firmware. **Functions which almost always fall into the software domain outside of the BIU are: BIU initialization, synchronization, asynchronous message insertion, scheduling tasks waiting for bus-oriented events, and message/device error handling, recovery and reconfiguration.**

5.8.1 BIU Initialization

The BIU must be set up for communications and the flow of information controlled at a high level. The interface between the bus controller software and the BIU hardware/firmware is: (1) control words, which direct the BIU to perform certain functions during the normal course of message transmissions and reception. This type of information is conveyed in the message header blocks, and (2) an I/O interfaces, which provides the mechanism for the initialization to be performed.

The following types of data are typically communicated through this interface:

1) Characterization Data
   a) Whether the BIU is to act as a bus controller or remote terminal.
   b) Terminal number (unique MIL-STD-1553 address).

2) Access Data
   a) Pointer to the next message header block that should be accessed for normal operation.
   b) Pointer to next message header block to be accessed for high priority message operation.
   c) Pointer (base address register) to the list of pointers containing the message areas.
   d) Pointer to the tag word area.

3) Operational Data
   a) Start operation.
   b) Perform self test.
   c) Receive minor cycle number or the current time for tag word entry. If time tagging is used the BIU must maintain or read a clock to update the time.
   d) Start high priority message.
   e) Halt at completion of current message including retries or additional error management.
   f) Halt immediately.
   g) Resume operation from wherever the suspension occurred.

4) Status Data
   a) Current message header block address that is being executed.
   b) Self-test results.
   c) Reason for interrupting.
   d) Characterization data.
   e) Access data.
   f) Operational data.

As can be seen by the list, all of the functions that are necessary to get the BIU going, determine what the BIU is doing, and to redirect it are available. In some BIUs the only way to read status data is to wait until it is in a quiescent state after commanding it to halt its operations. Other BIUs allow the reading of data at any time. Clearly, for the operation of the bus, **it is desirable to be able to (at least) determine the current and next message header block addresses, so that alterations in the chain can be made for asynchronous messages.**

5.8.2 Controller Synchronization

Two or more processors (as well as the primary and back-up controller) may require knowledge as to what "address state" the others are in, as well as "what time it is." In large systems more than 30 different input messages are required to be transmitted on the same bus to the same device. This condition leads either to the addition of more buses or a change of message subaddress meaning between each minor cycle (see paragraph 3.1.1 for protocol discussion). Synchronization is a method of signaling each processor that it is time to change the message memory map to the indicated minor cycle. Minor cycle synchronization also indicates that the previous set of transmissions have been completed. This signal allows tasks waiting for completion to begin processing on the data that arrived during the last minor cycle. This type of coordination can assure that the task will operate on data that is integral and not subject to change. Consequently, all synchronous tasks can wait for the start of the next minor cycle, unless a severe timing constraint exists that requires a task to process its data immediately upon arrival. In this case an interrupt must be set up to accomplish the scheduling of the affected task.

5.8.3 Scheduling Via Bus Event

A scheduler can potentially activate tasks based upon a number of events, which can occur over the bus such as minor cycle synchronization, arrival of a message, and completion of a message transmission. **The most common occurring event is the minor cycle. All normal synchronous tasks can wait for this event, either In a wait queue or by hav-**ing the cyclic tasks grouped together. Because this is so dominant, a special routine may speed the process.** The remaining tasks will be waiting on event queues, one per event. The message arrival/transmission event can be signaled in some BIU's by interrupting if a particular bit is set in the control word for transmission or reception on the bus controller, or as part of the message pointer in a remote terminal. The interrupt handler must determine the cause of the interrupt, then it will pass control to the bus controller to determine the reason for the interrupt. After this it will pass control to the event handler to set the event, which will schedule tasks that have been waiting for the event.

5.8.4 Message Error Handling and Recovery

The bus controller is responsible for the management of errors on the bus. If programmed, the BIU will perform retries according to the commands in the control word. If the retry procedures fail to achieve a valid transmission, then one of three courses are available; (1) the message header block may point to another set of message lists which will direct the BIU to perform further diagnostic tests, messages, or other recovery functions, (2) the message header block may point to a specific error recovery routine, which will be executed once the executive has determined the reason for the BIU error-interrupt, and (3) the executive can perform general error management by determining the type of error, type of message, source/destination, and by performing various types of diagnostic analyses according to time availability and need.

Many different error management schemes exist based upon the system requirements and individual designer's choice. Three examples are given to show the variety, which can exist for error management routines:

(1) Some error handling mechanisms simply delete communication with that remote terminal until the conclusion of the major cycle, at which point a test message (e.g., transmit status word mode code) is sent to the RT to re-establish communications. The results of the test message weighted by some recovery algorithm (e.g., 5 correct responses in a row) will allow the readmission of communication with the RT.

(2) Ignore the failure and proceed. This moves error recovery from the channel controller to the applications software for determination that a problem exists, which requires additional action. The tag word is a valuable tool in this mode, since it pro-

vides a time stamp (in terms of either minor cycle number or BIU/processor clock time of arrival). The application software must determine when the non-arrival of data becomes important and must invoke higher level application software and bus control functions to acquire alternate sources of data if such exist. The problem encountered in ignoring faults in communication are twofold: a) data is not necessarily gathered relative to the device failures and b) multiple failures can cause excessive transmission delays, which can in turn cause minor cycle overruns. A variation on this is to perform error retries until a maximum number of retries have been used. At that point discontinue retries and finish the transmissions necessary for completion of the minor cycle.

(3) Respond immediately to the failure and initiate simple retry procedures based upon the channel control word parameters. The message header block may point to individual applications oriented executive software routines that involve specific error management routines embedded in the bus controller. These functions may be necessary to perform a self-test or to cause a message sequence to be retried from the beginning. This approach provides more flexibility in the management of errors than the first approach, since the first approach could be implemented in terms of the capabilities described in the second. **The bus control interface must provide utility functions that could be Invoked by user-defined routines for specific message sequences.** The more preprogrammed version of this type of error management the more a common recovery procedure is established for a specific class messages (there may be two or three general classes). These procedures can invoke self-test in other RTs, decide when a device is faulty and invoke reconfiguration functions. **Generally, the quicker the problem can be Identified, the guilty established and recorded for later analysis (off-line) and the recovery performed, the simpler the system design will be.** The availability of redundancy (similar or dissimilar) plays an important role in arriving at a quick solution. Remember, while the system is resolving the problem, it is not doing its primary job.

Current BCs, such as the ACE, provide capability for performing automatic retries. Having retries performed by the BC serves to off-load the operation of the host processor by eliminating the need for the CPU to react to "soft error" conditions. Retries are generally software programmable, usually on a message basis. Retries are attempted following no response and format error conditions. In addition, retries may be attempted if specific Status Word bits are set. For instance, a response by a MIL-STD-1553A RT with the Message Error bit set in the Status Word may result in a retry. Other programmable parameters relating to retries are the number of retries to be attempted (one, two, etc.), bus selection (same or alternate), and interrupts and other mechanisms for reporting the occurrence and result of retries to the host CPU.

5.9 Message Reception and Use

When a bus controller commands a message to be transmitted to itself, the memory address into which the message is to be sent is computed in two ways; directly and indirectly. The direct address method uses an address in the channel control word to explicitly state the destination address. The advantage of this method is that it is very efficient for the bus controller and retains very explicit control at all times as to where the message will go. The indirect method makes use of a message subaddress and a base register. **The subaddress and T/R bit of the 1553 command word is added onto the base address to determine the address of the memory location containing the destination address of the message.** This extra level of indirection is used to make it very easy to change the location to which a message will go by simply changing the base register to point to another block of message addresses. The advantage of this method is that it is very easy to double buffer (or multi-level buffer) messages coming into a device at the highest update rate. Note that all message addresses can be changed when changing the base address register. Even if none are to change the base address concept could be employed. Each method requires that the number of words in the message be contained within the message header block. The disadvantage is for each message access, a separate memory fetch is required to determine the actual message location. A combination of the two methods results in a base address pointing to the beginning of multiple fixed 32-word message blocks. This method was used on some of the simpler bus controllers. The cost of simplicity is the 32-word message buffer regardless of message length.

If a bus controller must also act as a remote terminal, the message reception mechanism must be different from a direct address scheme. **The subaddress and pointer mechanism is the normal method for the reception and transmission of messages.** In the most general case, each message pointer can also contain a descriptor as to the action, which should be taken when a particular message has either arrived or been transmitted.

An example of such a descriptor format is shown in Figure I-3.14. This method allows double buffering, which the remote terminal BIU must perform by exchanging the two pointer variables. Additional actions which may occur are setting an event flag or causing an interrupt on either the arrival or the departure of a message. These constructs allow two messages to come to the same subaddress without immediate software processing of the message. Such constructs also allow a "loose" coupling between the timing of the message arrival and the processing of the message. If only a single buffer were used the first message must either be processed or it would be written over with the arrival of the next message to the same subaddress. The interrupt, on the other hand, allows immediate attention upon arrival of a particular message (e.g., deliver the weapon).

In conjunction with the arrival of each message can be a tag word identifying some characteristics of the transmission. For example the validity of the message and time of arrived are two transmission parameters of use to applications software. The 1553B standard states that invalid messages are not to be used. In reality most designs require a message to be deposited word by word as they arrive. If a word in the middle of the transmission is in error, the remainder of the message may not be deposited. This situation would result in part of the new message and part of the old message being consolidated together. At best the data would be independent and not matter or at worst the data might be word-coupled (e.g., two words in a floating point number) and changing one part and not the other would completely change the meaning of each. A tag word allows the application software to determine validity and decide whether to use the old data, the new data, (valid to the point of error discovery) or neither in its computation. The tag word could also be the key into the error management/reconfiguration routines as discussed in paragraph 5.2.

Once the data has been received, the final step is to make the data available to the application software. In most 1553 systems, the data is available at all times to the applications software, and it is incumbent upon the applications to determine the validity of the data (e.g., was the subsystem flag bit set or is a validity bit associated with the data words) and which of multiple buffers to use. In these systems data integrity is achieved by waiting until all inputs are completed before commencing processing or by using buffers which have been filled and isolated by the 1553 BIU. If the system is very concerned about data integrity then completed buffers can be moved to the applications software for use. Then when complete they

can be output from the applications software to the 1553 transmission buffers. However, the rather high penalty of copying each message discourages most users from applying this technique.

6.0 TESTING, INTEGRATION, AND INSTRUMENTATION

6.1 Testing

The purpose of any form of testing is to determine the quality or functional capability of an item. In the testing of 1553 devices, the purpose of the test procedure is to verify that the design does meet the 1553 specifications and that the options implemented have been done so correctly.

In the early days of 1553 implementation, there were few, if any, standard components (e.g. transceivers, encoders/decoders, protocol sequencers, etc.) available for the designers to use. Therefore, designers were developing their own based upon their interpretation of the standard. In addition, there was no standard "off-the-shelf" 1553 test equipment available to test the design, so the designers had to develop limited capability testers themselves to prove that their own designs worked and to provide their production facilities with a method of testing. As time and the popularity of the standard progressed, standardized components (sometimes available from multiple sources) started to become available. More important though, the experience in testing and integrating 1553 systems has led to the generation of a series of standardized test plans to support these requirements.

6.1.1 Levels of Testing

The levels of testing can be identified as follows: developmental, design verification, production, systems integration, and field/operational testing. These levels may be applied to the remote terminal, bus controller, monitor, or actual bus system (cable, couplers, stubs, etc.). These levels are discussed as follows:

(1) Developmental Testing. Developmental testing begins with the breadboard of the design. It is used to determine the circuits operation and to eliminate any design flaws. This level of testing also includes the testing of the circuits operating margins and tolerance limits usually over the required operational requirements (e.g., temperature range, etc.).

(2) Design Verification. For design verification, usually a preproduction unit is subjected to a series of tests designed to verify that the unit satisfies the requirements of the 1553 standard and the options specified in the system specification. This level is generally the most extensive phase of testing.

The BUS-65518 and BUS-69124 (formerly BUS-65520) validation tester plan (VTP) provides a turnkey tool for performing this test.

(3) Production Testing. Production testing is generally referred to as "end item" or "acceptance" testing, but can also be applied to in progress or sub-assembly items. It is assumed that the design has been previously verified and that this level of testing is performed to verify that all of the circuitry is functioning properly including mode code operation, error message validation, and any other special sequences that can normally be performed. Production testing procedures usually consist of a subset of the design verification tests. The BUS-65518 and BUS-69123 (formerly BUS-65519) production tester plan (PTP) provides a turnkey tool for performing this test (see photo below).

(4) Systems Integration Testing. The purpose of systems level testing is to insure that all elements of the bus network *play" together. This level of testing is usually centered around the operation of the bus controller's software and its ability to manage the data flow on the bus. This level of testing is generally performed in a Systems Integration Laboratory (see paragraph 6.2 on integration).

(5) Field/Operational Testing. Regardless of the amount of integration testing performed, the final design verification is the actual field/operational testing of the system. Often this is the first time the actual subsystems (as opposed to simulated systems) are interfaced to the bus network. For military applications this level of testing is usually referred to as Development Test/Operational Test (DT/OT). This level of testing is systems oriented and encompasses a total examination of all systems functions from the man/machine interface to the accuracy to the systems performance.
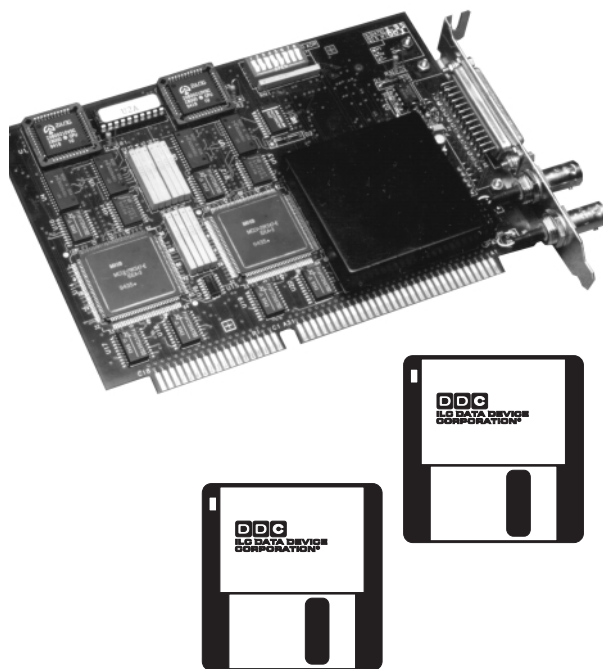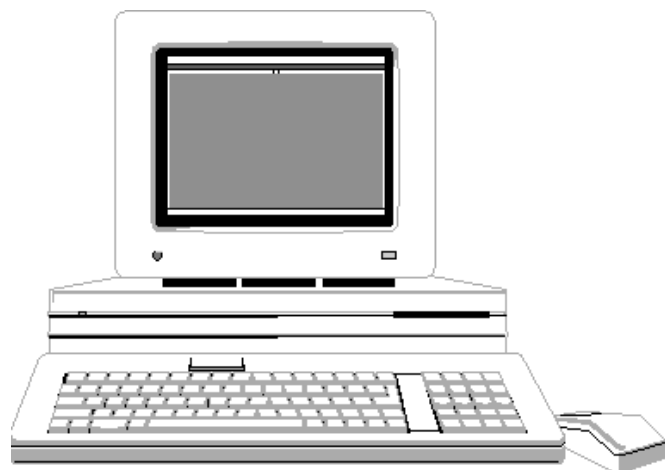
BUS-65518 and BUS-69123 Production Test Plan Board and Software. The BUS-65518 and BUS-69123 provides a turnkey means of configuring and running all protocol tests of the SAE RT production test plan. The BUS-65518 tester/simulator board is installed in an IBM PC/XT/AT® or compatible computer. The BUS-69123 software provides pop-up menus for tailoring the parameters of the test plan for a given RT. Once configured, the BUS-65518 is capable of performing a complete protocol test on a dual redundant RT in about 18 seconds.

® IBM PC/XT/AT is a registered trademark of International Business Machines Corporation.

Paragraph 1.6.8 describes the efforts of the Air Force testing program and the development of a series of test plans to cover both the remote terminal and bus controller for validation and production testing. These test plans have been jointly developed by industry and government 1553 experts and should be used as a source of guidance in the areas of circuit design, specification interpretations, and testing procedures.

6.1.2 Test Requirements

Test requirements for terminals can be divided into two main topics: electrical interface tests (including noise tests), and protocol tests. The electrical interface tests apply to all terminal types (remote terminals, bus controllers, and monitors), whereas the protocol tests are a function of the type of terminal being tested. All tests should be applied to each of the buses when the terminal contains redundant buses. Some of the tests are used to "characterize" the terminal rather that to verify compliance to the standard and the results should be included in the Interface Control Document (see paragraph 3.9). Each of the tests are summarized as follows:

6.1.2.1 Electrical Interface Tests

The specifications called out in the standard define the requirements at the connector pins of the terminal. These points are defined by point A in figures 9 and 10 of MIL-STD-1553B. It is important to note that all of the specified requirements are for the terminal itself and are not to be measured with the terminal connected to a system where they would be dependent on other system elements. The electrical interface tests can be subdivided into four parts: input, output, isolation, and noise tests.

The input tests include:
   a) input polarity
   b) input impedance
   c) input signal amplitude
   d) zero crossing distortion
   e) rise/fall times
   f) common mode rejection
   g) input bit rate stability

The output tests include:
   a) output polarity
   b) output amplitude
   c) zero crossing stability
   d) rise/fall times
   e) distortion, overshoot, and ringing
   f) output symmetry/tailoff
   g) output noise

   h) power on/off noise
   i) output bit rate stability

Since terminals functioning as monitors need not be designed with a transmitter, the output tests do not always apply.

The isolation tests are used to verify the input and output isolation between buses in a redundant bus design. The requirement is given as the ratio in dB between the voltage on the active bus and the voltage on the inactive bus.

The noise rejection tests are required to verify that the terminal exhibits a maximum word error rate of one part in 10 when operating in the presence of additive white Gaussian noise. The noise test is run continuously until the number of words received by the terminal exceeds the required number for acceptance or is less that the required number for rejection for a particular number of errors. The acceptance/rejection criteria is specified in Table II in the standard. In this test, it is the common practice to gate the noise source off while the terminal responds to the bus tester. This reduces the probability of the status word being *garbled." A typical set up for the noise rejection test is shown in figure 6.1.

6.1.2.2 Protocol Tests

The protocol tests are performed as a function of the terminal type. Bus controllers which are also capable of performing as a remote terminal (e.g. while acting as the backup bus controller) need to be tested for both functions. The protocol tests for the remote terminal and bus controller are summarized as follows:

(1) *Remote Terminal Protocol Tests.* The first step in testing the remote terminal is to verify that the terminal responds properly to all the legal (valid) information transfer formats including BC-RT, RT-BC, RT-RT, mode commands with and without data words (receive and transmit). These formats should be tested with all subaddress and word counts implemented in the terminal. All implemented mode code operations also need to be tested. If the terminal is designed to recognize illegal commends, then all of these commands, and their specified response, need to be tested.

The unique terminal address of the terminal needs to be checked by sequencing the address through all combinations while issuing commands to all the other addresses.

Notes:
(1) External coupler boxes (not included) require 1:14 turns ratio: DDC's BUS-25679 recommended.
(2) For noise tests the data output is always stub coupled.
(3) For noise tests, the noise output can be selected (via the front panel or GPIB)
    for either stub or direct coupling.
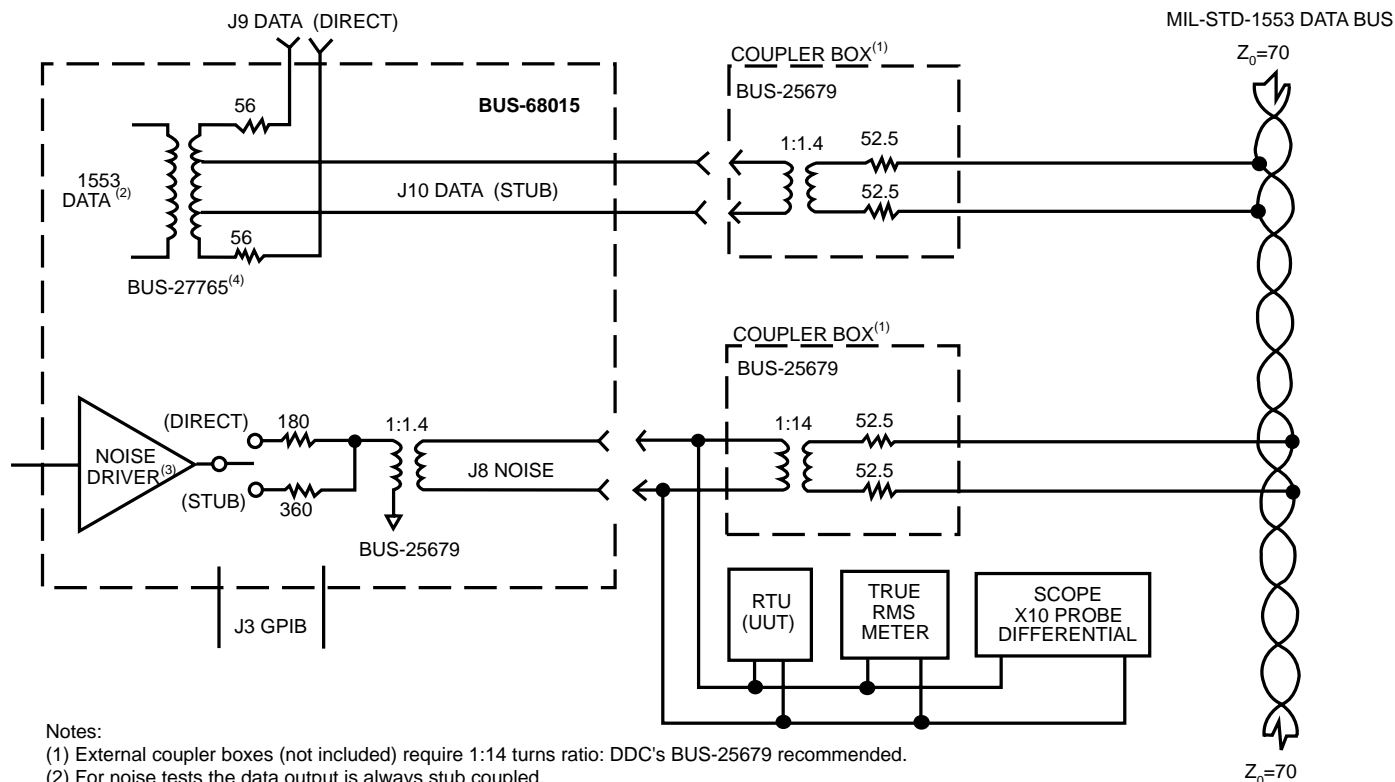(4) Turns ratio 1:1 for direct coupling and 1.4:1 for stub.

Figure I-6.1 Typical BUS-68015 Based Noise Test

In addition, if the broadcast option has been implemented, all of these tests need to be repeated for the broadcast address.

The terminals timing characteristics need to be verified and characterized. This includes the measurement of the status word response time. The terminal should also be tested for its ability to respond to superseding commands.

The message validation of the terminal needs to be examined by injecting messages with various error conditions. The validation criteria which needs to be tested includes:

a) sync errors
b) encoding errors
c) bit count errors (word length)
d) parity errors
e) word count errors (message length)
f) gaps (discontinuous data)

(2) *Bus Controller Protocol Tests.* Testing of the bus controller function requires prior knowledge of the bus controllers software. The first step is to verify that the bus controller is capable of issuing the desired command list and data. This is often difficult to do, especially in systems where events occurring on the bus or data word patterns cause the insertion of various aperiodic messages into the command list. Also an important part of this test is to monitor that the controller never issues invalid commands (1553) OR commands prohibited by the systems specification (i.e., dynamic bus control mode codes). The bus controller must also be tested to insure that it transmits on only one bus at a time.

If possible, the proper processing of the normal valid terminal responses must be tested. The bus controller must also be tested for its processing of abnormal or invalid responses. These may include: no response; improper status bits; word errors (sync, encoding, parity, etc.); discontinuous data words; and word count errors.

The bus controllers timing characteristics also needs to be verified. This includes the minimum intermessage gap and the minimum no response time out. As can be seen, knowledge of the software operation is required in order to perform most of the bus controller protocol tests.
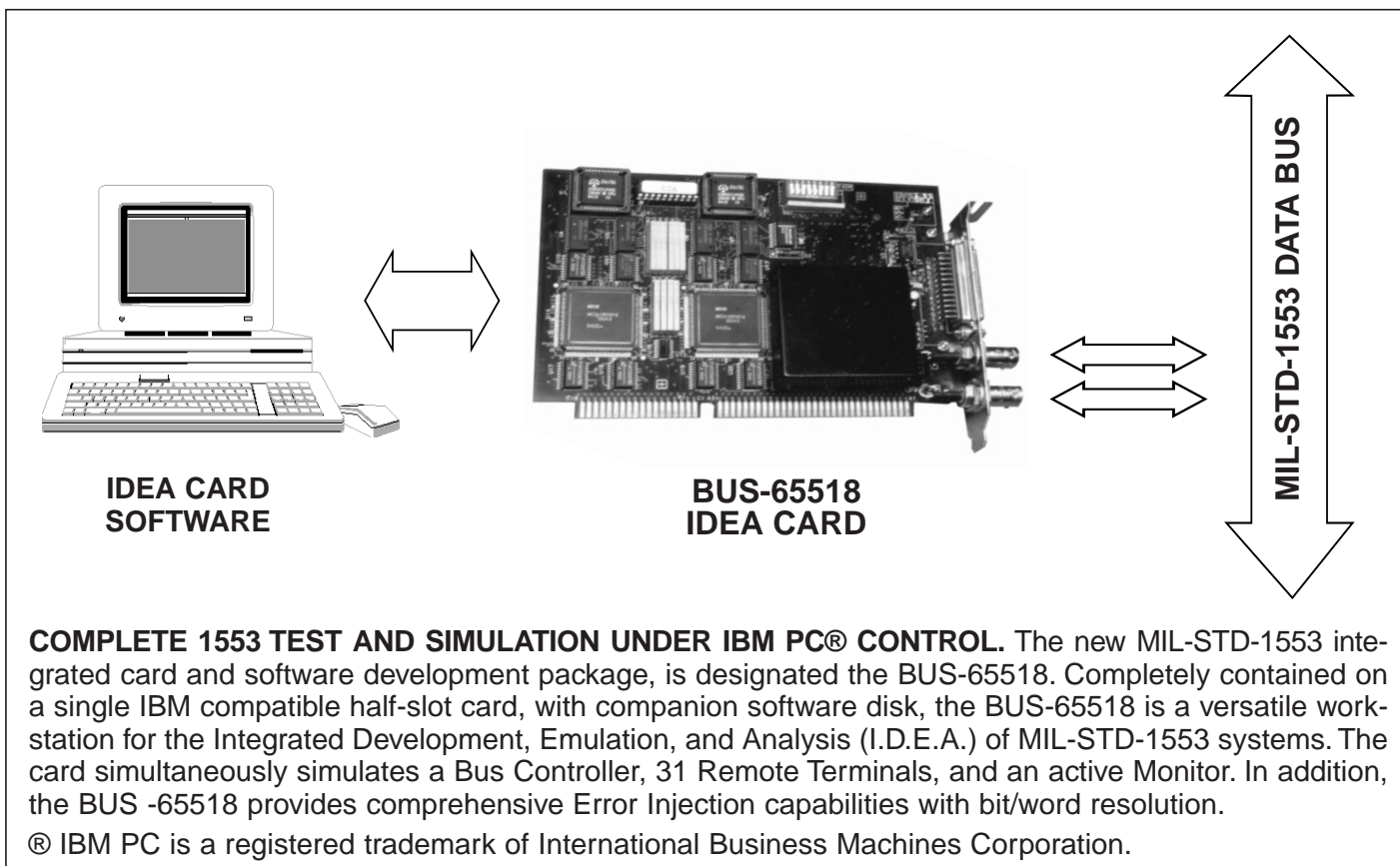
6.2 Systems Integration

Systems integration is the process where all of the components (bus controllers, remote terminals, and subsystems) are brought together and "married" into the common bus network. For multilevel or hierarchical networks, this may be first done at the lower bus level and then later combined into the total system. The process almost always includes the *marriage" of the system hardware (bus network) and software (bus controller and backup bus controller). In past applications, the systems integration task has been the responsibility of the aircraft manufacturer or the developer of the bus controller.

To assist in the integration effort, most companies have developed Systems Integration Laboratories (SIL), usually dedicated to the particular program. The SIL facility can vary greatly from a "hot bench" to full computer simulation. Today most SIL's provide for a mixture of capabilities allowing for the emulation/simulation of a component until a breadboard or preproduction unit is delivered. The advantage of this method is that the development (and checkout) of the bus controller software and the system can proceed without having to have the actual equipment.

While it is the desire of every systems designer to *marry" all the components, apply power, and watch everything work (correctly), it is seldom, if ever, accomplished. To this end, the method of "incremental integration" is most often applied. With this method, a portion of the bus control software is developed and initially integrated with one or two terminals. Then as more software is developed or more terminals are delivered, they are added to the system until at last the entire system has been successfully "married" together.

6.2.1 Simulation

Due to the schedule compression of most programs today, the systems integrator and system/bus controller software developers are turning to the emulation/simulation of the systems components. For definition purposes: emulation is used to replace a hardware function (e.g., use of a bus tester to communicate as a remote terminal); and simulation is used to perform the operational function of a terminal (e.g., manipulation of the data within the terminal based upon data word contents or a pre-programmed algorithm). In short, emulation applies to the hardware, whereas simulation applies to the software.

IDEA CARD SOFTWARE

BUS-65518 IDEA CARD

MIL-STD-1553 DATA BUS

COMPLETE 1553 TEST AND SIMULATION UNDER IBM PC® CONTROL. The new MIL-STD-1553 integrated card and software development package, is designated the BUS-65518. Completely contained on a single IBM compatible half-slot card, with companion software disk, the BUS-65518 is a versatile workstation for the Integrated Development, Emulation, and Analysis (I.D.E.A.) of MIL-STD-1553 systems. The card simultaneously simulates a Bus Controller, 31 Remote Terminals, and an active Monitor. In addition, the BUS -65518 provides comprehensive Error Injection capabilities with bit/word resolution.

® IBM PC is a registered trademark of International Business Machines Corporation.

By using the incremental integration method, it is possible to get an early start on the integration of the bus controllers and the development of the bus control and applications software development and checkout. Some systems designers have used bus testers (functioning as remote terminals) to emulate the various terminals on the bus. Some testers are even capable of emulating multiple terminals based upon the total number of messages which can be handled by the tester. Initially, the actual content of the data words is of little or no importance and therefore they can be set to random or constant data patterns or be set to incrementally change for each message transmission. The use of the bus testers for this emulation function will allow for the injection of errors in the messages and allow for the verification of the bus controllers error handling and recovery procedures.
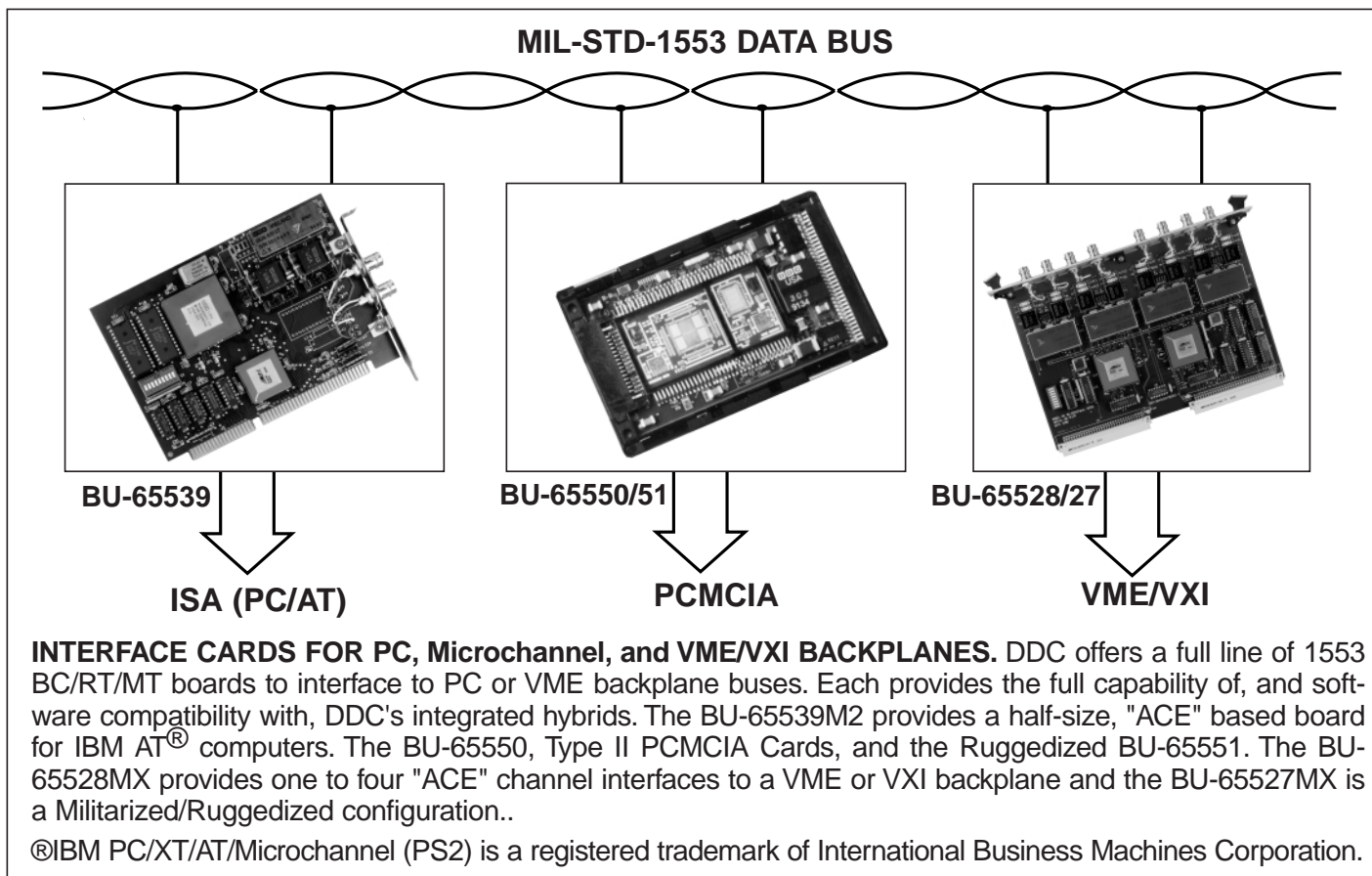
As the integration and software development progress, the content of the data words will start to become of some importance, especially to the applications programs. Some of the bus tester have limited processing capability which would allow for the movement of data between buffers or simple operations upon the data. For some terminals, this level of simulation may be sufficient. However, for the more complex subsystems and terminals, it may be necessary to go to a bigger computer with more computational power. To this end, several manufacturers have developed 1553 cards which interface to mini/micro computer backplanes (e.g. LS1-11, PDP-11, VAX, NOVA, Eclipse, S.E.L., Multibus) and interface via DMA techniques to the computer. This allows for some large applications simulation programs (i.e. aircraft flight characteristics, navigational models, etc.) to be developed on the host computer and the results downloaded to the 1553 interface card to be transmitted on the bus when commanded.

This simulation method can be used until actual hardware is available or until it is necessary to install the components on the aircraft or platform. Most systems integrators retain the SIL facility for use in solving system problems encountered during DT/OT and for the integration of future additions to the system.

6.3 Instrumentation

The ultimate systems test is DT/OT testing. Whether the application is airborne or ground based, this is often the first time some of the platform or mission equipment (e.g., engines, sensors,

MIL-STD-1553 DATA BUS

BU-65539 → ISA (PC/AT)

BU-65550/51 → PCMCIA

BU-65528/27 → VME/VXI

INTERFACE CARDS FOR PC, Microchannel, and VME/VXI BACKPLANES. DDC offers a full line of 1553 BC/RT/MT boards to interface to PC or VME backplane buses. Each provides the full capability of, and software compatibility with, DDC's integrated hybrids. The BU-65539M2 provides a half-size, "ACE" based board for IBM AT® computers. The BU-65550, Type II PCMCIA Cards, and the Ruggedized BU-65551. The BU-65528MX provides one to four "ACE" channel interfaces to a VME or VXI backplane and the BU-65527MX is a Militarized/Ruggedized configuration..

®IBM PC/XT/AT/Microchannel (PS2) is a registered trademark of International Business Machines Corporation.

weapons systems, etc.) are connected to the bus network. It is also during the DT/OT phase that the customer evaluates the performance of the product. Therefore, it is necessary in the effort of continued systems integration or performance evaluation to monitor and collect data from the bus network. In addition to the testing phase, some applications continuously monitor certain bus parameters as part of the vehicle/platform maintenance activity.

For validation and testing purposes, it is often necessary to collect one hundred percent of the data bus traffic. Today, bus monitors exist which are capable of extracting all data or being programmed to collect only certain messages to selective terminal addresses and subaddresses or error conditions.

Since 1553 is a one megabit asynchronous bus, the high speed (compared to other platform functions) and the non-synchronous nature of the data traffic present particular problems in the acquisition and analysis of the data. There are two primary methods for acquiring and storing this data; on board bulk storage, and telemetry.

On board bulk storage is performed by means of magnetic tape or semiconductor memory. High density, multitrack tape has advantages of allowing the data bus to be recorded and synchronized to a clock. Its obvious disadvantage is the amount of data which can be acquired due to the time limitations of the tape (speed, length, etc.). The amount of data storable in semiconductor memory can easily be increased by the addition of more memory, but there are limitations. Additionally, data compression techniques and algorithms can be applied to condense the data and increase storage capacity without loss of data or its resolution.

Telemetry systems collect and encode the data for transmission, via a radio communications link, to a ground station for recording and analysis. Systems available today monitor for the desired data and then synchronize the data with a clock. The synchronized data is then encoded via pulse code modulation (PCM) and time multiplexed along with other signals into the telemetry system. The data is usually formatted to the IRIG (Inter-range Instrumentation Group) standards and hence existing ground receiving equipment can handle the decommutation of the data.

The instrumentation of the data bus needs to be addressed as part of the overall testing and systems integration activities. The systems designer needs to analyze and identify the signals with need to be monitored such that an adequate instrumentation package can be designed.

The BUS-65518 tester/simulator card (see photo on page I-84) provides a tool that can serve many uses in 1553 development, testing, and maintenance. The board, which plugs into an IBM PC or compatible, is capable of emulating bus controller, up to 32 remote terminals, and bus monitor simultaneously. The board has capabilities for BC message scheduling (including minor and major frames), extensive error injection for BC and RTs, and "trigger" and *selection" (filtering) capabilities for the monitor. The BUS-65518 comes with interactive menu software allowing easy setup of all modes with no programming. The board can be used in a lab environment for software development and systems integration and debug.

In addition to the interactive menu software, real time libraries are available for C, Pascal, Windows 3.1X, 95 and NT.. The real-time libraries, which provide an interface between the user's program and the board, are ideal for test, simulation, and demonstration applications. Other software options for the BUS-65518 allow emulation of MIL-STD-1553A RTs, implement the validation and production RT test plans, support monitor-to-hard-disk storage, provide parameter monitor capabilities, and provide capabilities to reconstruct message traffic from monitored data.

7.0 OTHER ISSUES

7.1 Network System Security

The application of multiplex data bus networks to military aircraft has provided for a significant level of subsystems and data integration, expanding the capabilities of the avionics systems. In doing so, the *old" techniques of maintaining data security by using only discrete electronics systems and wiring is no longer possible without sacrificing many of the benefits of multiplex systems integration. Therefore, military data bus network designers must address the maintenance of data security within this integrated avionics system for both flight and ground operations.

7.1.1 Definitions

In order to provide a common lexicon from which to work with, the following definitions are provided.
RED – A designation applied to: a) all wirelines within the terminals or data routing equipment carrying classified plain language, b) all wirelines between the

unencrypted side of the on-line crypto equipment used and individual subscriber terminal equipment, c) equipment originating or terminating classified plain text language, and d) areas containing these wirelines, equipment, and the interconnecting lines.

BLACK – The designation applied to all wirelines, components, equipment, and systems which handle ONLY encrypted or unclassified signals, and areas to which no unencrypted or classified signals occur.

RED/BLACK – The concept that electrical circuits components, equipment, systems, etc. which handle classified plain language data in electrical form (RED) be separated from those which handle encrypted or unclassified information (BLACK). Under this concept the terms RED and BLACK are used to clarify specific criteria related to, and to differentiate between such circuits, components equipment, and systems, etc., and the area in which they are contained.

TEMPEST – An unclassified short name referring to investigations and studies of compromising emanations. It is sometimes used synonymously for the term "compromising emanations" (e.g., TEMPEST tests, TEMPEST inspections TEMPEST control plan).

7.1.2 System Security Policy

The design (system, hardware, and software) must fulfill the operational, maintenance, and logistic security policy of the aircraft, ship, or system, on which it is used. This policy is established by the government program manager in conjunction with the operational user. The details of this policy, which impact system designers, are provided in the program security classification guide and its references.

Some major areas capable of impacting the design are:
  a) The maximum classification of the data processed by each of the systems using the bus.
  b) The maximum classification and compartmentalization of the data transferred via the bus.
  c) The maximum authorized classification and compartment level of the systems connected to the bus.

  d) The access authorization of the various crew and maintenance personnel who will be using systems connected via the bus.

7.1.3 System Security Architecture

The data bus network operates as a time shared system between various processors and users of the data. If any of these subsystems process RED information, then the data bus network design must address three complementary security areas:
1) compromising emanations (i.e., TEMPEST), 2) encryption, and 3) trusted message routing and control. Each of these areas interacts with the other, and the system designer must determine the optimum combination to satisfy the security requirements.

System security architectures, each with different advantages and disadvantages. are:

  1) All BLACK bus – No RED data or RED data processor is directly connected to the bus.
  2) ALL RED bus – All systems connected to the bus are authorized access to the highest data classification and all compartments using the bus.
  3) RED/BLACK Gateway – Separate all RED and all BLACK buses, as defined above, are used with a gateway to allow necessary data to securely pass between them.
  4) RED/BLACK Composite – On a time-share basis, the bus processes RED and BLACK data between RED and BLACK subsystems securely.

7.1.4 TEMPEST Design

TEMPEST requirements must be fulfilled by the bus interface circuitry and the transmission media (bus network) for data bus systems processing RED information. The data bus network shall fulfill the TEMPEST provisions of the following documents:

  1) NACSIM-5100 and NACSIM-5112 for U.S. Military Systems
  2) BID/01/202 (4) and BID/01/200 (Series) for United Kingdom Military Systems
  3) AMSG-719 and AMSG-720 for NATO Military Systems
  4) Equivalent national requirements for other national military systems.

7.1.5 Encryption Designs

Encryption techniques may be used by the data bus network and its associated terminal's and processor to convert RED data into BLACK data and to isolate multiple classification levels and compartments of RED data. The specific encryption technique and system design must be approved by the government

agency responsible for encryption certification. While a discussion of specific acceptable encryption techniques is beyond the scope of this Designer's Guide and is classified, any encryption design for military systems must include the following:

1) TEMPEST design to preclude compromising emanations from by passing the encryption circuitry.
2) Trusted design of the encryption control, input/output and data processing circuitry, and software to preclude information compromise through failure or manipulation.
3) Cryptographic key management, coordination, distribution, and zeroize techniques, circuitry and software.
4) Synchronization and timing protocols.
5) Encryption alarm and alarm check techniques.

7.1.6 Trusted Message Routing and Control Design

The data bus network design must fulfill the requirements of DoD 5200.28-STD, "Department of Defense Trusted Computer System Evaluation Criteria," or national equivalent, with less than one undetected control error per 10,000 hours of operation at the maximum bus message transfer rate. Corrective control action to maintain and restore system trust must be incorporated for detected system security failures.

The data bus network can be designed to meet these requirements by using the following:

1) Low bit error rate circuitry and cabling techniques.
2) Parity coding of control words involved in message routing.
3) Bus controller monitor to detect and correct message routing errors.

Each of the System Security Architectures, as defined in the previous sections, have individual TEMPEST, encryption, isolation, and Trusted Computer System requirements for the data bus network's transmissions media, terminal and controller designs and gateway designs. Designers are directed to their Program's Security Classification Guide, DoD 5200.28-STD, and the TEMPEST provisions in the aforementioned documents for assistance in determination of the design requirements.

7.2 MIL-STD-1760B Interconnect Standard for Aircraft Stores

MIL-STD-1553B, due to its maturity and adaptability,

has found itself being used, either in total or parts thereof, in other standards and specifications within various military architectures. One such adaptation has been to MIL-STD-1760B. In addition to the adoption of 1553B, new hierarchical multi-level bus network architectures are being developed and imposed with this new standard. For this reason, an overview of 1760B, and supporting specifications is presented here. MIL-STD-1760B, released APRIL 15,1991, supercedes 1760A SEPTEMBER 3, 1985.

7.2.1 Definitions

MIL-STD-1760B standard defines implementation requirements for the Aircraft/Store Electrical Interconnection System (AEIS) in aircraft and stores. The electrical signal characteristics, connectors, and interface software (messages) for US/NATO aircraft and stores have been included. For definition purposes, a store is defined as any external device attached to the aircraft and is composed of devices which separate (leave the aircraft) such as missiles and bombs, and as permanent modules such as electronics pods (e.g., ECM, LANTIRN, etc.) or fuel tanks. Retrofit systems employing 1760B stores include existing aircraft such as the F-5E, F-16A/B, B-52G/H, and the B-1B. 1760B stores will also be designed into the new ATA, ATF, EFA, FSX and LHX aircraft.

The control path for these stores is a MIL-STD1553B data bus. However, there are restrictions, selected options, and exceptions to 1553B taken within 1760B to meet the aircraft/stores interoperability that requires attention by system designers.

MIL-STD-1760B defines a Mission Stores Interface (MSI) as a single electrical receptacle external to the store structure. The aircraft connection is defined as the Aircraft Station Interface (ASI). Connection point examples are pylons, fuselage hard points, internal weapons bays, and wing tips. Umbilical cables connect the aircraft ASI to the store MSI. Figure I-7.1 illustrates a weapon attached to an ejector (suspension and release equipment), which is attached to a wing pylon. The primary 1760B signal set is available at the pylon ASI and connects to the store via the umbilical to the MSI. Each store uses selected signals from the available set as required. A carriage store is a device that attaches a Carriage Store Interface (CSI) to a single weapon station ASI and can interface with multiple mission stores.

In 1760B, no spare pins are available to aircraft or store designers. All interface pins in the connector are fully defined. Instead of unique signals over dedicated

PYLON

DATA BUS

AC POWER

DC POWER

RELEASE CONSENT

INTERLOCK

HB (1–4)

LB

ADR SELECT

TO OTHER
ASIs

ASI

SUSPENSION AND RELEASE
EQUIPMENT

UMBILICAL
CABLE

MSI

REMOTE TERMINAL
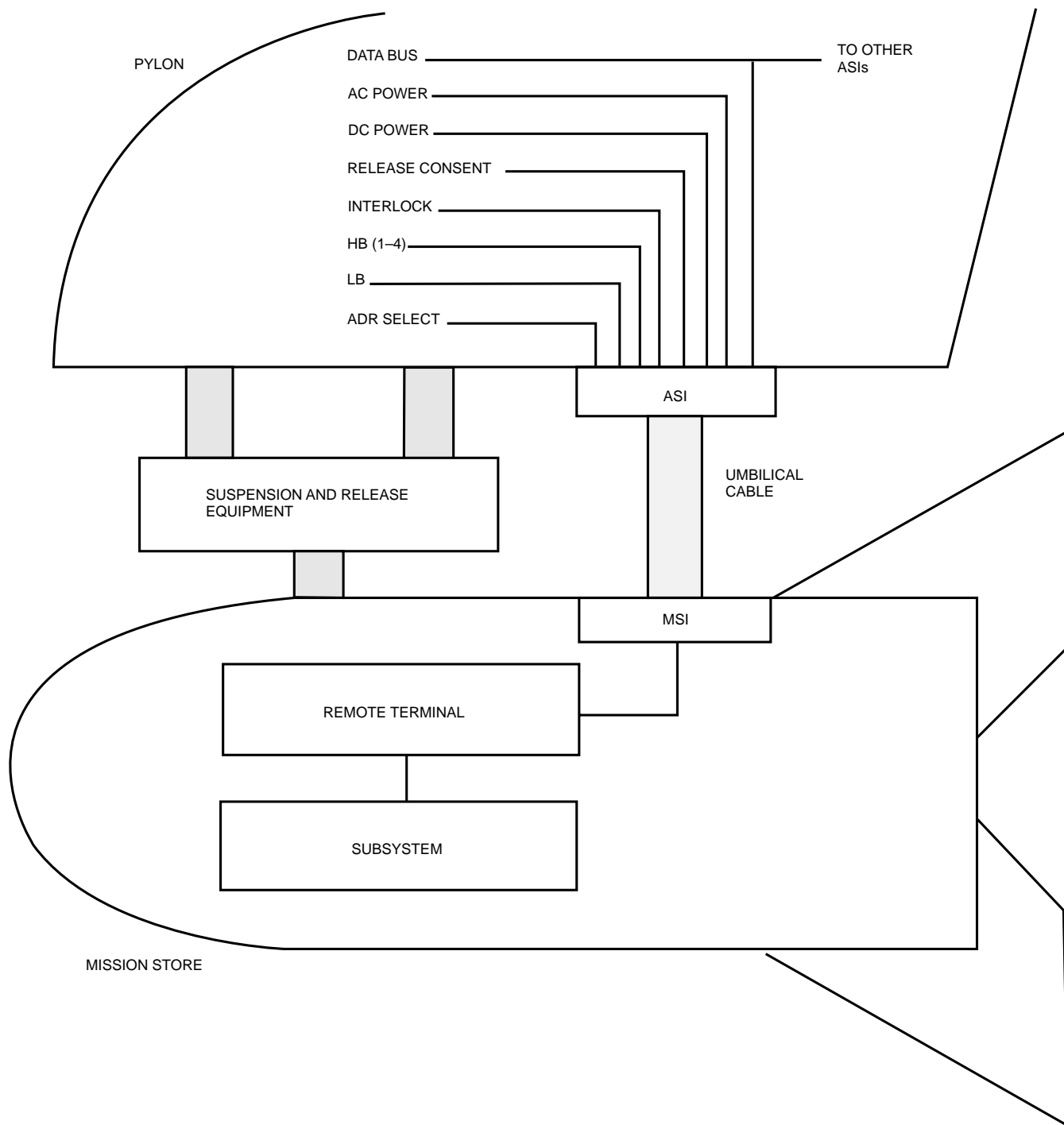
SUBSYSTEM

MISSION STORE

Figure I-7.1 Store Station Utilizing Primary Signal Set

wires, data in 1553 word format is transmitted over a 1553 dual redundant standby bus network.

Mission stores contain embedded 1553 remote terminals. Per the standard, mission stores must be capable of BC-RT, RT-BC, and RT-RT message transfers. 1760B allows for both normal 1553 bus communication protocol (command/response) and broadcast message protocol. **The aircraft has the responsibility to function as the bus controller.** Due to the safety critical nature of stores, dynamic bus control is prohibited from use on a 1760B bus network.

7.2.2 Restrictions

Subaddress assignments are defined in TABLE B-I in APPENDIX B of MIL-STD-1760B. Defined subaddresses include: a) store description, b) test operations, c) nuclear store operations, d) control and monitoring, and e) mass data transfer, and f) Data wraparound. As in 1553B both mode code indicator (00000 and 11111) are required. Mission store terminals are required to implement the transmit status word, reset remote terminal, transmitter shutdown, override transmitter shutdown, transmit vector word, synchronize with data word, and transmit last command word mode codes. If the terminal flag bit is implemented, then the inhibit terminal flag mode code is required. In addition, if the service request bit is used, the data word associated with the transmit vector word mode code must be available when the bit is set. The setting of the subsystem flag bit is interpreted as a total loss of the store. MIL-STD-1760B imposes a 30 word limit on both transmit and receive messages to/from a store (see paragraph B.40.2.1 of APPENDIX B). The use of checksums in the data message is optional except for: a) Mission store control, b) Mission store monitoring messages, and c) Store description messages. All checksummed messages shall satisfy the modulo 2 arithmetic to each bit (no carries), of all data words including the checksum, the sum shall be zero.

Store initialization sequences. First communication, is defined in Paragraph 5.1.1.12.3. on page 33. The initialization procedure imposes on the network bus controller a message timing sequence to first communication no sooner then 150 milliseconds for first status response (with potential for the busy bit to be set in the status word) and up to 500 milliseconds for response of a valid store identification message. The timing reference point is 150 milliseconds after the application of power to the store (115/120V ac and 28V dc power #1). The 30 word identification message consists of: a) a one word header with a fixed hexadecimal pattern of 0421, b) a one word

country code, c) nine words of store identity (e.g., AGM-131), d) Maximum interruptive BIT time with eighteen words of null characters, and e) one word of the checksum value for the message.

7.2.3 MIL-STD-1760B Signals Related to 1553B

Figure I-7.2 shows the interface signals related to the data bus system. Mux A and mux B are the 1553 transformer coupled stubs from the aircraft. The address and odd parity bits are for wiring the unique remote terminal address, which is established at the ASI connector. It is permissible for stores to latch the address upon initialization rather than continuously reading it.

As shown in figure I-7.2, the aircraft determines the stores presence by monitoring the interlock circuitry. The store is designed to jumper the interlock signal to the interlock return signal. The circuit is broken when the store separates. The store is not permitted to use the interlock signals to determine the aircraft's presence. The store may use the terminal address lines to determine if the aircraft is present (attached). However, since a single fault failure is possible at the address interface, **the store designer should not allow safety critical functions to occur based solely upon detection of the no aircraft.**

Enabling of release consent is used to signal to the store that safety critical messages arriving via the data bus are to be acted upon as valid consent from the aircraft (crew). Safety critical events include irreversible functions such as rocket motor ignition, suspension hook release, and deployment of chutes and fins. 1760B allows 10 milliseconds for the consent signal stabilization before the messages start arriving. The return reference for release consent is the return of dc power #2.

7.2.4 Data Bus Networks and Components

The network configuration depends upon the mission. The initial store load determines the number of remote terminals to be connected to the weapons network. During flight weapons are expended and removed from the bus. Since the number of remote terminals is changing, the bus loading and network reflections is also changing. Improper store separation may physically damage the umbilical. Hot rocket exhaust plume potentially could cause multiple shorts on the bus, hampering communications with the remaining stores. Since ASI and MSI connectors and umbilicals are exposed to extreme physical environments, the stubs are more likely to experience shorts than the network contained within the aircraft fuselage. Therefore, the systems
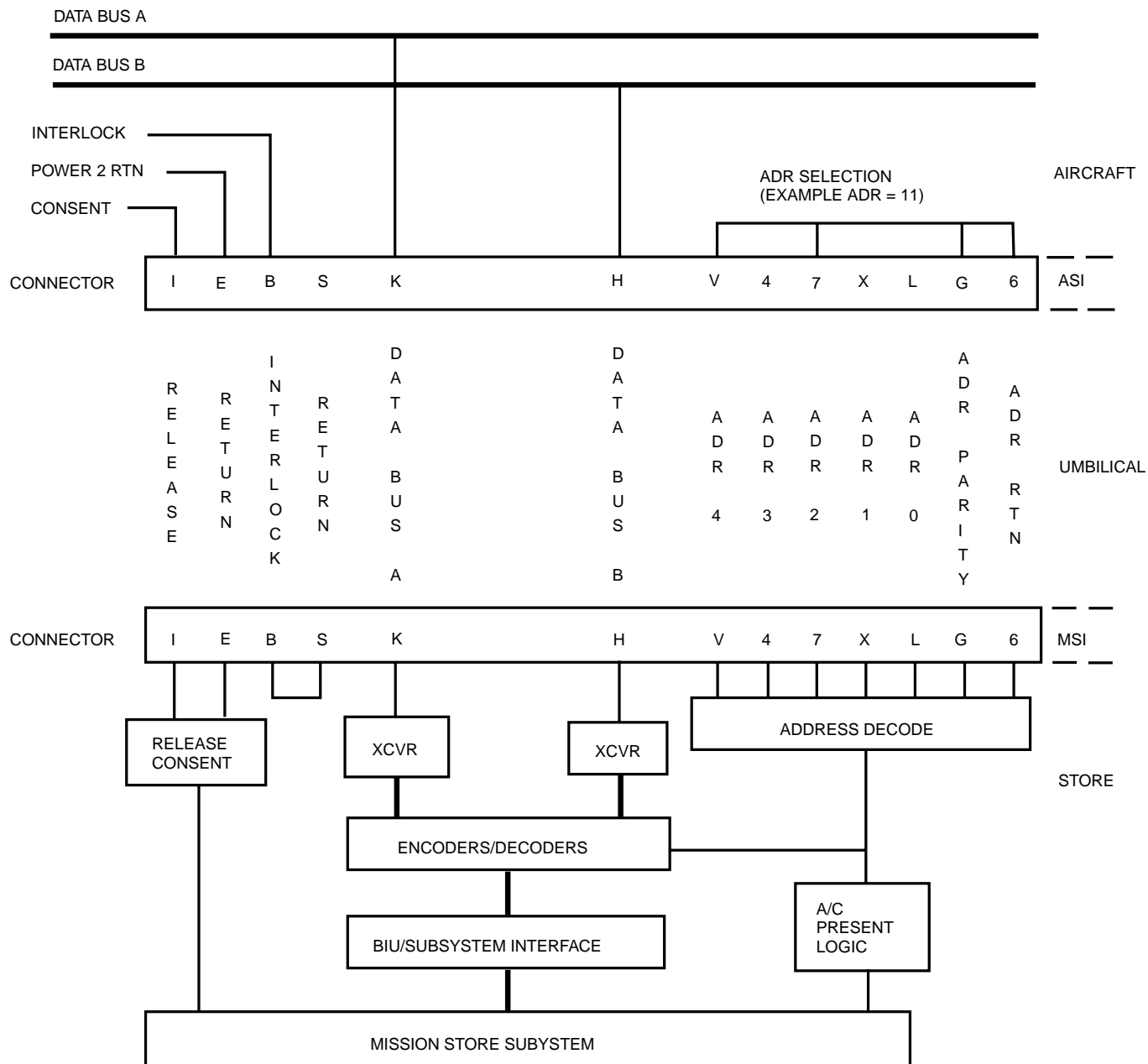
DATA BUS A

DATA BUS B

INTERLOCK

POWER 2 RTN

CONSENT

ADR SELECTION
(EXAMPLE ADR = 11)

AIRCRAFT

CONNECTOR

| I | E | B | S | K | H | V | 4 | 7 | X | L | G | 6 | ASI |

RELEASE | RETURN | INTERLOCK | RETURN | DATA BUS A | DATA BUS B | ADR 4 | ADR 3 | ADR 2 | ADR 1 | ADR 0 | ADR PARITY | ADR RTN | UMBILICAL

CONNECTOR

| I | E | B | S | K | H | V | 4 | 7 | X | L | G | 6 | MSI |

RELEASE CONSENT

XCVR

XCVR

ADDRESS DECODE

STORE

ENCODERS/DECODERS

A/C PRESENT LOGIC

BIU/SUBSYSTEM INTERFACE

MISSION STORE SUBSYSTEM

Figure I-7.2 1553B Signals in 1760A

designer must take special precautions when designing 1760B bus networks.

Transformer coupled stubs are required for all connections in 1760B. **To be interoperable to all stores, the aircraft coupler transformer should not be canter tapped on the stub side.** In order to support multiple aircraft store stations without routing the bus, coupler boxes with five to eight transformer coupled stubs are now available. This allows the bus to remain in a more protected environment, such as inside the fuselage or wing, and branch out with stubs to a cluster of external connectors at the stations.

Transceiver output voltage for the stores on the stub, is set at a minimum of 20 volts peak to peak when measured at the MSI; two volts higher than the 1553B minimum. This additional requirement on transceivers is necessary since the network paths vary from aircraft to aircraft and station to station. The ASI and MSI also cause network breakpoints that are not usually made for stubs. Constant voltage transceivers will provide a tighter control of output

voltage (as opposed to constant current transceivers) as the network load changes. Current limiting of the transceiver output stage provides protection against multiple high loads on the network.

Twinax connector design is required in 1760B for both data bus signals (A and B). The coaxial data high, data low, and shield provides a protection from the electromagnetic environment (EMI, EMP, and lighting).

7.3 System 2

System 2 is a specification conforming to MIL-STD-1760B covering the interface between an aircraft and a nuclear weapon. Aircraft Monitor and Control (AMAC) Project Officers Group (POG) specification system 2001-01 System 2, is managed from WL/NTSAC, AMAC/POG, Kirkland AFB, NM 87117. The classified System 2 specification has been included since there is a need for aircraft, fire control, and weapons manufacturers and designers to understand the requirements and goals of this specification. The development of System 2 goes back as far as the development of MIL-STD-1760. Early in the development stage of both documents, the decision was made to keep System 2 as a specification that complied with the 1760 standard. The evolution of System 2 has been slow, principally because of the rapid evolution of 1760, and the need to avoid differences between System 2 and 1760.

The System 2 specification will fully specify the electrical and logical interface for a nuclear bomb and will specify the electrical interface and part of the logical interface concerning warhead functions for missiles with nuclear warheads.

7.3.1 Significant Elements of System 2

There is one provision of paramount importance in the System 2 specification that impacts upon hardware and system designers. Remote terminal subaddresses 19 and 27 are reserved by System 2 for exclusive use by nuclear weapons. This means that **NO stores bus network can have any remote terminals except nuclear weapons that use subad-dress 19 or 27 for any purpose if that stores bus network is ever to be used in conjunction with nuclear weapons.** This also means that non-nuclear (non-System 2) weapons must not use sub-address 19 or 27 if they are ever to be carried in a *mixed load" on the same stores bus network with nuclear weapons. This restriction is significant enough to impact most aircraft bus architectures and, unless it is absolutely unavoidable, a stores bus network should contain only stores with embedded remote terminals, remote terminals for controlling the weapon (its suspension and release equipment), and the stores fire control (equipment controlling 1760 and non-1760 signals). No other avionics should be connected to the stores bus network. The reason for this restriction is to improve nuclear safety by simplifying the control of critical functions and minimizing the possibility of interaction with other functions. For this reason, the use of an RT-RT transfer, involving either of the two restricted subaddresses, is prohibited in System 2. The use of dynamic bus control is also prohibited.

Nuclear certification becomes more difficult and costly with complex systems. Highly integrated architectures with tightly coupled subsystems necessitate nuclear certification of subsystems having no intrinsic functional involvement with the nuclear weapon, other than perhaps being connected to the same data bus network. Therefore, it is the avowed intention of the System 2 specification and the nuclear certification community to influence aircraft and systems designers to build simply connected, highly isolated, stores bus networks for nuclear stores. This purpose should not be overlooked when designing these systems.

System 2 defines its interface requirements between an aircraft system and a single nuclear weapon. The logical interface, that is the bus messages definitions, sequences, and protocols have been developed. Care should be taken before designing any stores network system (nuclear or non-nuclear) to insure that all the proper safeguards have been accounted for. Probably the first implementation will become the de facto standard for all following implementations.

II.
REVIEW AND RATIONALE OF
MIL-STD-1553 A AND B

II. REVIEW AND RATIONALE OF MIL-STD-1553A AND MIL-STD-1553B.

This section is an explanation of each part of MIL-STD-1553B on a paragraph-by-paragraph basis. The descriptions include (1) rationale for the requirements specified; (2) the requirements; and (3) identification of differences between MIL-STD-1553A and MIL-STD-1553B. The 1553B part that is discussed is presented first (indented as below), followed by the rationale, explanations, and differences from 1553A.

1.0 SCOPE

1.1 Scope. This standard establishes requirements for digital, command/response, time division multiplexing (Data Bus) techniques on aircraft. It encompasses the data bus line and its interface electronics illustrated on figure 1, and also defines the concept of operation and information flow on the multiplex data bus and the electrical and functional formats to be employed.

1.2 Application. When invoked in a specification or statement of work, these requirements shall apply to the multiplex data bus and associated equipment which is developed either alone or as a portion of an aircraft weapon system or subsystem development. The contractor is responsible for invoking all the applicable requirements of this Military Standard on any and all subcontractors he may employ.

Additional sentences were added to 1553B to clarify designer selected options. The basic difference between 1553A and 1553B is that in 1553B the options are defined rather than being left for the user to define as required. It was found that when the standard did not define an item, there was no coordination in its use. Hardware and software had to be redesigned for each new application. Therefore, the one primary goal of 1553B was to provide flexibility without creating new hardware and software designs for each new user. This was accomplished by specifying the "use of" rather than the requirement "to use" in the functional areas and by specifying the electrical interfaces explicitly so that compatibility between designs by different manufacturers could be electrically interchangeable.

2.0 REFERENCED DOCUMENTS

2.1 Issue ot document. The following document of the issue in effect on date of invitation for bid or request for proposal, forms a part of the standard to the extent specified herein.

SPECIFICATION
MILITARY

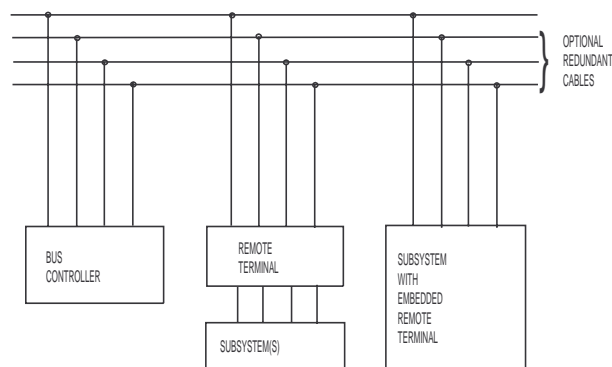**MIL-E-6051** Electromagnetic Compatibility Requirements, Systems

Figure 1 of 1553B. Sample Multiplex Data Bus Architecture

(Copies of specifications, standards, drawings, and publications required by contractors in connection with specific procurement functions should be obtained from the procuring activity or as directed by the contracting officer.)

The only difference between the two revisions was that in 1553B the references to MIL-STD-461 and MIL-STD-462 were removed (i.e., both electromagnetic interference requirements and measurement standards). MIL-E-6051 "System Electromagnetic Compatibility Requirements" is still applicable in 1553B to define the wiring and cabling provisions of the specification (see MIL-STD-1553B, paragraph 4.5.1.1.5.3).

The definition section of the standard has been expanded and reordered in 1553B. The purpose for the change was to address definitions in order of complexity and to describe new functions, modes, and devices. A comparison of the definition included in 1553A and 1553B is presented in table II-1.

3.0 DEFINITIONS

3.1 Bit. Contraction of binary digit: may be either zero or one. In information theory a binary digit is equal to one binary decision or the designation of one or two possible values of states of anything used to store or convey information.

3.2 Bit rate. The number of bits transmitted per second.

3.3 Pulse code modulation (PCM). The form of modulation in which the modulation signal is sampled, quantized, and coded so that each element of information consists of different types or numbers of pulses and spaces.

3.4 Time division multiplexing (TDM). The transmission of information from several signal sources through one communication system with different signal samples staggered in time to form a composite pulse train.

3.5 Halt duplex. Operation of a data transfer system in either direction over a single line, but not in both directions on that line simultaneously.

3.6 Word. In this document a word is a sequence of 16 bits plus sync and parity. There are three types of words: command, status and data.

3.7 Message. A single message is the transmission of a command word, status word, and data words if they are specified. For the case of a remote terminal to remote terminal (RT to RT) transmission, the message shall include the two command words, the two status words, and data words.

3.8 Subsystem. The device or functional unit receiving data transfer service from the data bus.

3.9 Data bus. Whenever a data bus or bus is referred to in this document it shall imply all the hardware including twisted shielded pair cables, isolation resistors, transformers, etc., required to provide a single data path between the bus controller and all the associated remote terminals.

3.10 Terminal. The electronic module necessary to interface the data bus with the subsystem and the subsystem with the data bus. Terminals may

Table II-1. Comparison of MIL-STD-1553A and MIL-STD-1553B Definitions

MIL-STD-1553A definitions (paragraph number)		MIL-STD-1553B definitions (paragraph number)	
Bit	(3.2)	Bit	(3.1)
Bit Rate	(3.3)	Bit rate	(3.2)
Pulse code modulation	(3.4)	Pulse code modulation	(3.3)
Time division multiplexing	(3.5)	Time division multiplexing	(3.4)
Half duplex	(3.7)	Half duplex	(3.5)
Word	(3.10)	Word	(3.6)
Message	(3.11)	* Message	(3.7)
—		** Subsystem	(3.8)
Data bus	(3.12)	* Data bus	(3.9)
—		** Terminal	(3.10)
Bus controller	(3.13)	* Bus controller	(3.11)
—		** Bus monitor	(3.12)
Remote terminal	(3.1)	* Remote terminal	(3.13)
Asynchronous operation	(3.8)	Asynchronous operation	(3.14)
Dynamic bus allocation	(3.9)	Dynamic bus control	(3.15)
Command or response mode	(3.6)	Command or response mode	(3.16)
—		** Redundant data bus	(3.17)
—		** Broadcast	(3.18)
—		** Mode code	(3.19)

*Definition changed significantly.
*Not previously defined.

exist as separate replaceable units (LRU's) or be contained within the elements of the subsystem.

This definition of terminal is intentionally broad. Terminals in 1553 have common operational characteristics, as well as assigned roles in data bus operation. The three allowable roles are defined in 3.11, 3.12, and 3.13. Common operational requirements of terminals are given in 1553B, paragraph 4.4.1. Note that the definition gives designers complete freedom of functional partitioning of the operating parts of a terminal, and that there is also no restriction of physical partitioning.

3.11 Bus controller. The terminal assigned the task of initiating information transfers on the data bus.

3.12 Bus monitor. The terminal assigned the task of receiving bus traffic and extracting selected information to be used at a later time.

3.13 Remote terminal (RT). All terminals not operating as the bus controller or as a bus monitor.

3.14 Asynchronous operation. For the purpose of this standard, asynchronous operation is the use of an independent clock source in each terminal for message transmission. Decoding is achieved in receiving terminals using clock information derived from the message.

This definition refers to the electrical characteristic by which the timing of message bits in a word are decoded. This use of "asynchronous operation" should not be confused with an asynchronous message that may interrupt or suspend the transmission of synchronous (i.e., periodic) messages in an avionic system.

3.15 Dynamic bus control. The operation of a data bus system in which designated terminals are offered control of the data bus.

3.16 Command/Response. Operation of a data bus system such that remote terminals receive and transmit data only when commanded to do so by the bus controller.

In the case of the definitions for message, bus controller, remote terminal, asynchronous operation, dynamic bus control, and command/ response, the change from 1553A to 1553B was developed to produce a more general definition. However, in the definition of data bus, 1553B encompasses more

equipment. Instead of including only the wire, the data bus couplers are also included. Two definitions were added for clarity: subsystem and terminal. The others (bus monitor, redundant data bus, broadcast, and mode codes) were added to define the additional requirements stated in 1553B. The function of a bus monitor is to monitor the data bus and record specified bus activity. The objective of defining a bus monitor function is new to 1553B. Two basic capabilities have been identified for the monitor in paragraph 4.4.4 of 1553B: (1) an off-line application including a flight test recording, maintenance recording, or mission analysis, and (2) a unique data bus terminal, which provides an internal backup bus controller function, with sufficient information to take over as the active bus controller in the even of a switchover or a failure of the active bus controller. In these two roles, the bus monitor hardware may have the performance capability of a terminal (unique address) or may be attached to the data bus without the knowledge of the other bus users (including the bus controller). In this second approach, no bus communication from or to the bus monitor by the bus controller is possible. The bus monitor acts as a passive listener to the specified traffic it is assigned to record. Obviously, the performance of a bus monitor requires the monitoring of the data bus for command words, status words, and data words. From this monitoring, the specific message collection process can occur during normal and abnormal (bus error and recovery) bus traffic. To aid in accomplishing the detection of these words (command and status), the optional instrumentation bits (bit 10 in the status word) and the associated bit in the command word (bit 10) can be set to a logic 1 and a logic 0, respectively.

3.17 Redundant data bus. The use of more than one data bus to provide more than one data path between the subsystem, i.e., dual redundant data bus, tri-redundant data bus, etc.

The redundant data bus definition was added to 1553B to identify a particular approach for obtaining multiple data paths to improve message arrival probability. Paragraph 4.6 of 1553B discusses the use of a dual-redundant data bus where the operation is identified as dual standby. In this mode, only one bus is active at any given time, except when superseding commands are sent on the standby bus. Under this condition, the terminal responds to the most recent command.

3.18 Broadcast. Operation of a data bus system such that information transmitted by the bus controller or a remote terminal is addressed to more than one of the remote terminals connected to the data bus.

The broadcast definition has been added to 1553B to describe a new protocol option. The use of this protocol allows a bus controller or a remote terminal to address more than one terminal connected to the system. This is accomplished by transmitting a dedicated terminal address (11111) and each recover withholding the normal status word response.

3.19 Mode code. *A means by which the bus controller can communicate with the multiplex bus related hardware, in order to assist in the management of information flow.*

The mode code definition was added to 1553B because of the definition of several mode code operations in paragraph 4.3.3.5.1.7. These optional mode codes are used to manage the information transfer system. The basic philosophy of the data bus system is that it is a "transparent data communication link." This means that its operation and management does not involve the use of the sensor data that it is transmitting or receiving. However, overhead is required to manage such a data link. Therefore, command words, status words, and message gaps are required to provide this capability. The combination of command word, mode codes, and responses to these mode codes provide the basis for managing the multiplex system.

4.0 GENERAL REQUIREMENTS

Several paragraphs have been added, changed, and renumbered in the requirements section of 1553B compared to 1553A.

4.1 Test and operating requirements. All requirements as specified herein shall be valid over the environmental conditions which the multiplex data bus system shall be required to operate.

This new paragraph of the 1553B was added to indicate that the performance requirements specified in this standard shall apply over the environmental conditions in which the multiplex data bus system shall be required to operate. It is anticipated that for most military applications this will be described by MIL-E-5400 Class II and 'some nuclear-hardening specifications. Because of the diversity of environmental conditions, the standard does not specify these requirements. Therefore, the system designer must determine, from appropriate vehicles or system specifications, the environmental conditions imposed on the multiplex data bus system.

4.2 Data bus operation. The multiplex data bus system in its most elemental configuration shall be as shown on figure 1. The multiplex data bus system shall function asynchronously in a command/response mode, and transmission shall occur in a half-duplex manner. Sole control of information transmission on the bus shall reside with the bus controller, which shall initiate all transmissions. The information flow on the data bus shall be comprised of messages which are, in turn, formed by three types of words (command, data, and status) as defined in 4.3.3.5.

This paragraph is identical to paragraph 4.1 in 1553A with the exclusion of the reference to electromagnetic compatibility, which appears in paragraph 4.5.1.1.5.3 of 1553B.

4.3 Characteristics

4.3.1 Data form. Digital data may be transmitted in any desired form, provided that the chosen form shall be compatible with the message and word formats defined in this standard. Any unused bit positions in a word shall be transmitted as logic zeroes.

4.3.2 Bit priority. The most significant bit shall be transmitted first with the less significant bits following in descending order of value in the data word. The number of bits required to define a quantity shall be consistent with the resolution or accuracy required. In the event that multiple precision quantities (information accuracy or resolution requiring more than 16 bits) are transmitted, the most significant bits shall be transmitted first, followed by the word(s) containing the lesser significant bits in numerical descending order. Bit packing of multiple quantities in a single data word is permitted.

This paragraph is identical to paragraph 4.2 in 1553A with the additional capability in paragraph 4.3.2 of 1553B concerning bit-packing of multiple quantities.

Bit-packing is a method used to improve transmission efficiency when subsystem data, which contain less than 16 bits of information per parameter (word), are collected and distributed in one word or message. Single-bit data and other parameters that are characterized by bit patterns of fewer than 16 bits will not fill the 16 bits of data allowed in 1553 data word format. Two approaches are used to utilize all bits in a word: (1) packing multiple parameters and words and (2) filling in zeros for all unused bits. In the first approach, the encoding and decoding cost must be considered, while in the second approach the inefficiency of sending as little as one bit per word must be considered.

4.3.3 Transmission Method

4.3.3.1 Modulation. The signal shall be transferred over the data bus in serial digital pulse code modulation form.

This paragraph remained unchanged in both revisions of the standard.

4.3.3.2 Data code. The data code shall be Manchester II biphase level. A logic one shall be transmitted as a bipolar coded signal 1/0 (i.e., a positive pulse followed by a negative pulse). A logic zero shall be a bipolar coded signal 0/1 (i.e., a negative pulse followed by a positive pulse). A transition through zero occurs at the midpoint of each bit time (see figure 2).

This paragraph remained unchanged in both revisions of the standard.
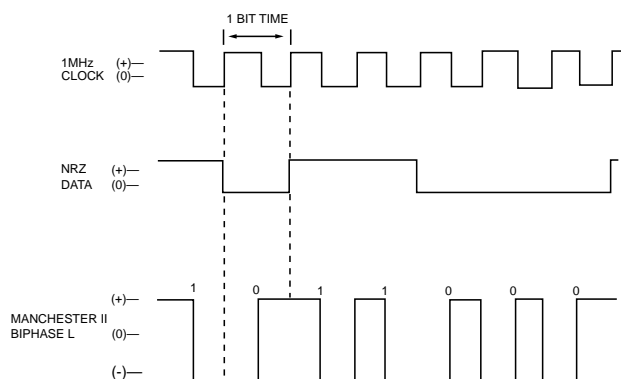
Figure 2 of 1553B. Data Encoding

4.3.3.3 Transmission bit rate. The transmission bit rate on the bus shall be 1.0 megabit per second with a combined accuracy and long-term stability of ± 0.1 percent(i.e., ± 1000 Hz).The short term stability (i.e., stability over 1.0 second interval) shall be at least 0.01 percent (i.e., ± 100 Hz).

The long- and short-term stability of the individual internal clocks used to transmit encoded data have been relaxed in 1553B. The order of magnitude reduction in transmission bit rate stability allows for the selection of multiplex bus interface clocks that can meet long-shelf-life requirements of some weapons.

4.3.3.4 Word size. The word size shall be 16 bits plus the sync waveform and the parity bit for a total of 20 bits times as shown on figure 3.

The 20-bit word size was selected because it represented the minimum number of bits in a word, when
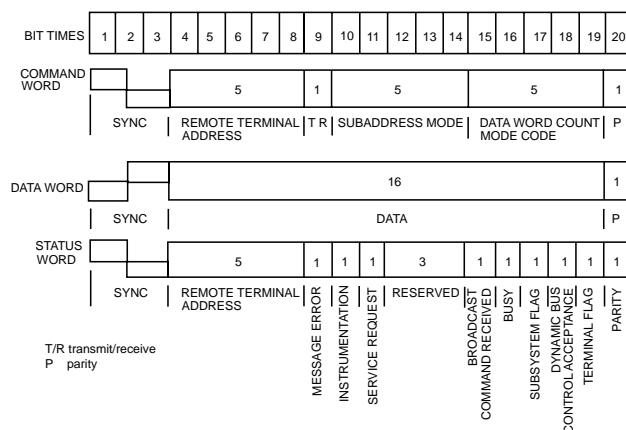
Figure 3 of 1553B. Word Formats

16 bits of data, a three-bit invalid Manchester sync pattern, and a single parity bit are used. Except for paragraph changes of 4.2.3.4 (1553A) to 4.3.3.4 (1553B) this paragraph remained unchanged. Three-bit invalid Manchester sync pattern is described in 1553B, paragraph 4.3.3.5.1.1 Figure 3 (referenced in this paragraph) is modified in 1553B to reflect the identification of status codes.

4.3.3.5 Word formats. The word formats shall be as shown on figure 3 for the command, data, and status words.

Three types of word formats were selected to operate the information transfer system. Each format and the changes reflected in 1553B will be discussed in the following paragraphs.

4.3.3.5.1 Command word. A command word shall be comprised of a sync waveform, remote terminal address field, transmit/receive (T/R) bit, subaddress/mode field, word count/mode code field, and a parity (P) bit (see figure 3).

The command word format is used to control and manage the information transfer system. Two basic additions were made to the command word by 1553B. These include the broadcast mode and the identification of the optional mode codes.

4.3.3.5.1.1 Sync. The command sync waveform shall be an invalid Manchester waveform as shown on figure 4. The width shall be three bit times, with the sync waveform being positive for the first one and one-half bit times, and then negative for the following one and one-half bit times. If the next bit following the sync waveform is a logic zero, then the last half of the sync controller waveform will have an apparent width of two clock periods due to the Manchester encoding.

The sync pattern used in the standard remained unchanged.

4.3.3.5.1.2 Remote terminal address. The next five bits following the sync shall be the RT address. Each RT shall be assigned a unique address. Decimal address 31 (11111) shall not be assigned as a unique address. In addition to its unique address, a RT shall be assigned decimal address 31 (11111) as the common address, if the broadcast option is used.

Each remote terminal will be assigned a unique address for which it is responsible to respond when the address is transmitted as part of a command word on the data
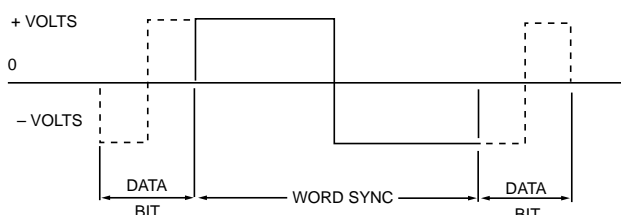
Figure 4 of 1553B. Command and Status Sync

bus by the active bus controller. Only one remote terminal address cannot be assigned as a unique address (decimal address 31). This address has been assigned to all remote terminals as the common address for which they will receive broadcasted data if the system uses the broadcast option.

The broadcast mode provides a mechanism for transmitting information to multiple users with a single message. The mechanism for accomplishing this was to dedicate address 31 (11111) to be reserved for broadcast messages. Anytime a broadcast message is transmitted, the transmitting terminal will use address 31 rather than a unique terminal address. All other addresses can be assigned as in 1553A. Since multiple sets of broadcast messages can be responding status word must be suppressed. By choosing the address method to accomplish the broadcast mode, all the other formats of the command word are available for use. Broadcast messages can be used with subaddresses and mode codes. The subaddress in a broadcast message can allow multiple users with broadcast reception capability to sort our specific broadcast messages transmitted, if given this capability in hardware or software. Therefore, multiple sets of broadcast messages can be defined. In addition, the broadcast format can be used with mode codes. This allows simultaneous transmission of mode commands to users.

Indiscriminate use of the broadcast technique is not advisable. Designers must question the benefit of dis-carding the command/response format, in which all message completion failures are know to the bus controller, to the benefits described below. Broadcast use may increase system operation complexity since subaddresses of broadcast address and addressed terminal will not likely be the same. This requires additional subaddresses. Finally, the broadcast technique, if used, adds a failure mode to the system if a terminal in a failure mode used address 31 for a message. Proper use of the broadcast mode may yield several benefits:

a. *Multiple terminals can be communicated with simultaneously, thereby permitting time synchronization of data or commands.*

b. *Bus duty cycle can be reduced by transmitting data required by multiple users simultaneously instead of sequentially.*

c. *Some error management can be enhanced by providing a single address by which all terminals can receive commands simultaneously. This permits the bus controller to immediately command a state for the system rather than polling each unit individually with the same command in a serial fashion.*

The broadcast message capability can produce considerable reduction in bus usage. This is particularly true for systems using multiple units for redundancy or systems dependent on parallel processing, thus requiring simultaneous data arrival at the processing units. As noted in 1553B, paragraph 10.6 (appendix to 1553B), improper use of the broadcast format can result in undesirable system operation. Since no status word response is allowed from the receiving terminal, discretion must be exercised when applying the capability. To provide message arrival verification, a bit in the status word is set when a valid broadcast message is received. This allows reporting of the reception if requested by the active bus controller using the mode code "transmit status word." In error situations, it may be advisable for the bus controller to request the last command word to verify that the broadcast command was received. There may be situations for which rebroadcast cannot be permitted. Asking for last command first preserves the last status word (i.e., the terminal does not reset or update status). In addition to data transfers, the ability to transmit a broadcast command message provides an effective method for managing the data bus system. This capability is performed using the broadcast address in combination with mode commands.

4.3.3.5.1.3 Transmit/receive. The next bit following the remote terminal address shall be the T/R bit, which shall indicate the action required of the RT. A logic zero shall indicate the RT is to receive, and a logic one shall indicate the RT is to transmit.

The transmit/receive bit in the command word indicates the source of data flow in the information transfer system. Basically, the paragraph remained unchanged for both revisions of the standard except for wording changes and paragraph numbering differences.

4.3.3.5.1.4 Subaddress/mode. The next five bits following the T/R bit shall be utilized to indicate an RT subaddress or use of mode control, as is dictated by the individual terminal requirements. The subaddress/mode values of 00000 and 11111 are reserved for special purposes, as specified in 4.3.3.5.1.7, and shall not be utilized for any other function.

This field has two functions: (1) the subaddress identification of specific messages to a remote terminal and (2) reserved subaddresses that serve as the identification that a mode command to the information transfer system is being transmitted. Both of these capabilities were present in 1553A. However, an additional mode code designator has been established in 1553B (decimal 31). The use of either 00000 or 11111 in the subaddress/mode field will be decoded to indicate that a mode code command is present in the next five-bit field. This limits the subaddress range to a maximum of 30 unique addresses. If the instrumentation bit (par. 4.3.3.5.3) in the status word is implemented, the subaddresses will be limited to 15 unique subaddresses. The requirements for use of the instrumentation bit are in 1553B, paragraph 4.3.3.5.3.4 In complex remote terminals (i.e., terminals interfacing with several sensors or multiple interface types), the subaddress capacity of a terminal can be exceeded. In addition, messages to a given remote terminal-subaddress may contain "packed" data requiring additional decoding prior to distribution within the terminal. Both of these conditions can cause the remote terminal's design to incorporate a map (i.e., look-up table) approach for subaddress message distribution.

4.3.3.5.1.5 Data word count/mode code. The next five bits following the subaddress/mode control shall be the quantity of data words to be either sent out or received by the RT or the optional mode code as specified in 4.3.3.5.1.7. A maximum of 32 data words may be transmitted or received in any one message block. All 1's shall indicate a decimal count

of 31, and all 0's shall indicate a decimal count of 32.

The dual function of this field provides for the identification of message lengths for data message or mode codes for managing the information transfer system. The identification of both of these capabilities was provided in 1553B but only the word count was specified in 1553A (even though the mode code function has remained the same in both revisions). The five-bit field allows 32 data words to be transmitted in a message or 32 specific mode codes. This is accomplished by one data word being represented as 00001, and all zeros being arbitrarily defined as decimal 32

4.3.3.5.1.6 Parity. The last bit in the word shall be used for parity over the preceding 16 bits. Odd parity shall be utilized.

The use of a single parity bit per word was provided to identify bit errors occurring during the transmission and detection of a word. According to the statement in the appendix to 1553B, "Theoretical and empirical evidence indicates than an undetected bit error rate of 10^{-12} can be expected from a practical multiplex system built to this standard." See 1553B, paragraph 10.4. Also see noise test in 1553B, paragraph 4.5.2.1.2.4. This paragraph remained unchanged during both revisions.

4.3.3.5.1.7 Optional mode control. For RT's exercising this option a subaddress/mode code of 00000 or 11111 shall imply that the contents of the word count field are to be decoded as a five bit mode command. The mode code shall only be used to communicate with the multiplex bus related hardware, and to assist in the management of information flow, and not to extract data from or feed data to a functional subsystem. Codes 00000 through 01111 shall only be used for mode codes which do not require transfer of a data word. For these codes, the T/R bit shall be set to 1. Codes 10000 through 11111 shall only be used for mode codes which require transfer of a single data word. For these mode codes, the T/R bit shall indicate the direction of data word flow as specified in 4.3.3.5.1.3. No multiple data word transfer shall be implemented with any mode code. The mode codes are reserved for the specific functions as specified in table I and shall not be used for any other purposes. If the designer chooses to implement any of these functions, the specific codes, T/R bit assignments, and use of a data word, shall be used as indicated. The use

Table II-1 Assigned Mode Codes

Transmit-receive bit	Mode code	Function	Associated data word	Broadcast command allowed
1	00000	Dynamic bus control	No	No
1	00001	Synchronize	No	Yes
1	00010	Transmit status word	No	No
1	00011	Initiate self-test	No	Yes
1	00100	Transmitter shutdown	No	Yes
1	00101	Override transmitter shutdown	No	Yes
1	00110	Inhibit terminal flag bit	No	Yes
1	00111	Override inhibit terminal flag bit	No	Yes
1	01000	Reset remote terminal	No	Yes
1	01001 ↓	Reserved	No ↓	TBD ↓
1	01111	Reserved	No	TBD
1	10000	Transmit vector word	Yes	No
0	10001	Synchronize	Yes	Yes
1	10010	Transmit last command	Yes	No
1	10011	Transmit bit word	Yes	No
0	10100	Selected transmitter shutdown	Yes	Yes
0	10101	Override selected transmitter shutdown	Yes	Yes
1 or 0	10110 ↓	Reserved	Yes ↓	TBD ↓
1 or 0	11111	Reserved	Yes	TBD

Note: TBD — to be determined.

of the broadcast command option shall only be applied to particular mode codes as specified in table II-1.

The basic philosophy of the information transfer system is that it operates as a transparent communication link. "Transparent" means that an application's function does not need to be involved with the management of communication control. Obviously, the information transfer system requires management that introduces overhead into the transmission of data. The command words, status words, status word gaps, and message gaps are the overhead. Within the command word the mode codes provide data bus management capability. The mode codes have been divided into two groups: mode codes without a data word (00000-01111) and mode codes with a data word (10000-11111). The use of bit 15 in the command word to identify the two types was provided to aid in the decoding process. Also, the use of a single data word instead of multiple data words was adopted to simplify the mode circuitry. Generally, with these two types of mode commands, all management requirements of an information transfer system can be met.

Control messages are identified by the subaddress/mode field in the command word being set to 32(00000)or 31 (11111). (In this case, 1553B defines decimal subaddress 32 to be equal to binary 00000 so that decimal 1 through decimal 31 correspond to binary 00001 through 11111.) All control messages originate with the active bus controller and are received by a single receiver or by multiple receivers (broadcast). A terminal address value of 31 (11111) in the command word indicates a broadcast message, while any other terminal addresses are to identify unique messages to a terminal on the bus. The mode command information is contained completely in the mode code/word field of the command word.

Table I of 1553B should be examined carefully to see the symmetry of the mode codes. The first 16 codes are not transmitted with a data word; the last 16 are. It is not appropriate to broadcast some of the mode codes because of the possibility of bus crashes -- simultaneous transmission by two or more terminals. Examples are requests for transmissions from RTs. Also, broadcast of dynamic bus control makes no sense. The T/R bit is important for mode codes 17 to 31 because it defines whether bus controller or RT is to transmit the associated data word.

The use of mode commands option is defined in both versions of the standard; however, 1553B defines each mode command while 1553A only defines dynamic bus control. There is no particular reason for the assignment of the mode codes, except for dynamic bus control (00000), which was previously defined in 1553A and this separation of mode command by the use of a data word. The purpose of reserved mode commands in each category (with and without data words) is important to allow for controlled expansion of the standard. By controlling the mode code command number and its definition, commonality between various terminal can be maintained. Each mode code command identification is listed in 1553B table 1. All other mode codes are considered illegal commands. The message formats associated with mode commands are shown in figure II-1.

4.3.3.5.1.7.1 Dynamic bus control. The controller shall issue a transmit command to an RT capable of performing the bus control function. This RT shall respond with a status word as specified in 4.3.3.5.3. Control of the data bus passes from the offering bus controller to the accepting RT upon completion of the transmission of the status word by the RT. If the RT rejects control of the data bus, the offering bus controller retains control of the data bus.

The dynamic bus control mode command (00000) is provided to allow the active bus controller a mechanism (using the information transfer system message formats) to offer a potential bus controller (operating as a remote terminal) control of the data bus. Only the single receiver command request (unique address) is allowed to be issued by the active bus controller. The response to this offering of bus controller is provided by the receiving remote terminal using the dynamic bus control acceptance bit in the status word (par. 4.3.3.5.3). Rejection of this request by the remote terminal requires the presently active bus controller to continue offering control to other potential controllers
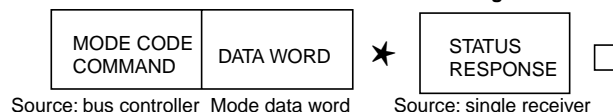
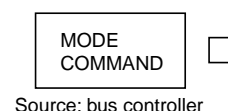**Mode Command Without Data Word to a Single Receiver**
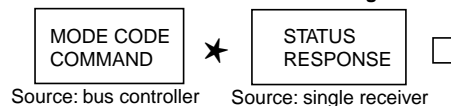
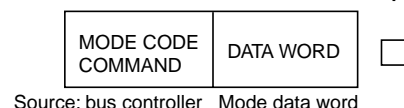**Transmit Mode Command With Data Word to a Single Receiver**

Receive Mode Command With Data Word to a Single Receiver

Transmit Mode Command Without Data Word to Multiple Receivers

Transmit Mode Command With Data Word to Multiple Receivers

★ Response time delay or gap
☐ End of message delay or gap

Figure II-1. Mode Command Message Transfer Formats

or remain in control. When a remote terminal accepts control of the data bus system by setting the dynamic bus control acceptance bit in the status word, control is relinquished by the presently active bus controller, and the potential bus controller begins bus control.

Note that the sequence above requires software (or firmware) implementation in all bus controllers.

4.3.3.5.1.7.2 Synchronize (without data word). This command shall cause the RT to synchronize (e.g., to reset the internal timer, to start a sequence, etc.) The RT shall transmit the status word as specified in 4.3.3.5.3.

4.3.3.5.1.7.3 Transmit status word. This command shall cause the RT to transmit the status word associated with the last valid command word preceding this command. This mode command shall not alter the state of the status word.

The status word associated with mode code (00010) is shown in figure II-2 and contains the following information:

a. Transmitting terminal address
b. Message error bit
c. Instrumentation bit
d. Service request bit
e. Broadcast command receive bit
f. Busy bit
9. Subsystem flag bit .
h. Terminal flag bit

Details concerning the usage of the status bits are discussed in 1553B, paragraph 4.3.3.5.3. The only message format for acquiring the status word using this mode code is for the bus controller to request the status word from a single receiver. Note that use of this mode code by the bus controller causes the last status word to be transmitted. Some subtle conditions need to be examined by the designer who uses this mode command. For example, if the transmit built-in test mode command is needed to verify the terminal is operational, that request (see 4.3.3.5.1.7.14) must be issued after the transmit status word mode code to prevent loss of the previous message's status.
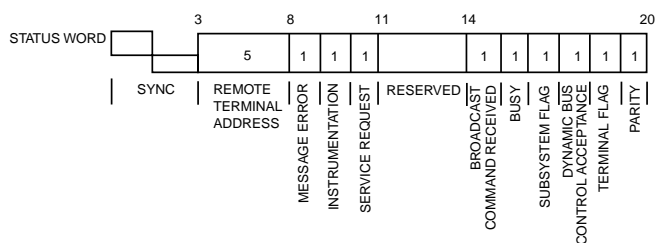
Figure II-2. Status Word

4.3.3.5.1.7.4 Initiate self test. This command shall be used to initiate self test within the RT. The RT shall transmit the status as specified in 4.3.3.5.3.

The initiate self-test mode command (00011) is provided to initiate built-in-test (BIT) circuitry within remote terminals. The mode code is usually followed, after sufficient time for test completion, by a transmit BIT word mode command yielding the results of the test. The message formats provided for this mode command allow for both individual

requests and multiple requests. Notice that the initiate self-test mode command is associated with the multiplex system terminal hardware only.

4.3.3.5.1.7.5 Transmitter shutdown. This command (to only be used with dual redundant bus systems) shall cause the RT to disable the transmitter associated with the redundant bus. The RT shall not comply with a command to shut down a transmitter on the bus from which this command is received. In all cases, the RT shall respond with a status word as specified in 4.3.3.5.3. after this command.

4.3.3.5.1.7.6 Override transmitter shutdown. This command (to be used only with dual redundant bus system) shall cause the RT to enable a transmitter which was previously disabled. The RT shall not comply with a command to enable a transmitter on the bus from which this command is received. In all cases, the RT shall respond with a status word as specified in 4.3.3.5.3 after this command.

4.3.3.5.1.7.7 Inhibit terminal flag (T/F) bit. This command shall cause the RT to set the T/F bit in the status word specified in 4.3.3.5.3 to logic zero until otherwise commanded. The RT shall transmit the status word as specified in 4.3.3.5.3.

The inhibit terminal flag mode code (00110) is used to set the terminal flag bit in the status word to an unfailed condition regardless of the actual state of the terminal being addressed. This mode code is primarily used to prevent continued interrupts to the error handling and recovery system when the failure has been noted and the system reconfigured as required. Commanding this mode code prevents future failures from being reported, which normally would be reported using the terminal flag in each subsequent status word response. The message format associated with the mode code allows for both single receivers and multiple receivers to respond. No data word is required with this mode code. Note that the terminal flag, which is used to indicate an RT fault condition is implicitly limited to terminal faults.

4.3.3.5.1.7.8 Override Inhibit T/F bit. This command shall cause the RT to override the inhibit T/F bit specified in 4.3 3.51.7.7. The RT shall transmit the status word as specified in 4.3.3.5.3.

The override inhibit T/F flag mode command (00111) negates the inhibit function thus allowing the T/F flag bit in the status response to report present condition of the terminal. This mode code can be transmitted by the active bus controller to both

single and multiple receivers. There is no data word associated with this mode code.

4.3.3.5.1.7.9 Reset remote terminal. This command shall be used to reset the RT to a power up initialized state. The RT shall first transmit its status word, and then reset.

The reset remote terminal mode code (01000) causes the addressed terminal to reset itself to a power-up initialized state. This mode code may be transmitted to an individual or to multiple terminals.

4.3.3.5.1.7.10 Reserved mode codes (01001 to 01111). These mode codes are reserved for future use and shall not be used.

4.3.3.5.1.7.11 Transmit vector word. This command shall cause the RT to transmit a status word as specified in 4.3.3 5.3 and a data word containing service request information

The transmit vector word mode code (10000) is associated with the service request bit in the status word and is used to determine specific service being required by the terminal. The service request bit and the transmit vector word provide the only means available for the terminal to request the scheduling of an asynchronous message if more than one service request exists per terminal. The message format for this single receiver operation contains a data word associated with the terminal's response. Figure II-3 illustrates the message formats associated with this mode command.

4.3.3.5.1.7.12 Synchronize (with data word). The RT shall receive a command word followed by data word as specified in 4.3.3.5.2. The data word shall contain synchronization information for the RT. After receiving the command and data word, the RT shall transmit the status word as specified in 4.3.3.5.3.

Synchronization informs the terminal(s) of an event time to allow coordination between the active bus controller and receiving terminals. Synchronization information may be implicit in the command word (mode code 00001) or a data word (mode code 10001) may be used to follow the command word to provide the synchronization information. /f a data word is used, the definition of the bit meanings is the responsibility of the system designer.

4.3.3.5.1.7.13 Transmit last command word. This command shall cause the RT to transmit its status word as specified in 4 3.3 5.3 followed by a single data word

which contains bits 4-19 of the last command word, excluding a transmit last command word mode code received by the RT This mode command shall not alter the state of the RT's status word.

The transmit last command mode code (10010) is used in the error handling and recovery process to determine the last valid command received by the terminal, except for this mode code. Also this mode code will not change the state of the status word. The message format associated with the single receiver last command word contains a data word from the responding terminal. The data word contains the previous 16 bits of the last valid command word received. Notice that this mode command will not alter the state of the receiving terminals status word. This fact allows this mode command to be used in error handling and recovery operation without affecting the status word, which can have added error data.

4.3.3.5.1.7.14 Transmit built-in-test (BIT) word. This command shall cause the RT to transmit its status word as specified in 4.3.3.5.3 followed by a single data word containing the RT BIT data. This function is intended to supplement the available bits in the status words when the RT hardware is sufficiently complex to warrant its use. The data word, containing the RT BIT data, shall not be altered by the reception of a transmit last command or a transmit status word mode code. This function shall not be used to convey BIT data from the associated subsystem(s).

The transmit BIT word mode command (10011) provides the BIT results available from a terminal, as well as the status word. Typical BIT word information for both embedded and standalone remote terminals includes encoder-decoder failure analog T/R failures, terminal control circuitry failures, power failures, subsystem interface failures, and protocol errors (e.g., parity, Manchester, word count, status word errors, and status word exceptions). The internal contents of the BIT data word are provided to supplement the appropriate bits already available via the status word for complex terminals. Notice that the transmit BIT word within the remote terminal "…shall not be altered by the reception of a transmit last command or transmit status word mode code" received by the terminal. This allows error handling and recovery procedures to be used without changing the error data recorded in this word. However, the RT will only save the last command, and the status code field (of the status word) will not be changed if transmit last command or transmit status word mode commands are transmitted. If, however, any other transmissions are

made to the RT, the status code field may change (e.g., if a message error occurred during the transmission). Broadcast of this command by the bus controller is not allowed. See paragraphs 4.3.3.5.1.7.3 and 4.3.3.5.1.7.13.

Another point worth noting is that the function of transmitting RT BIT data ". . . shall not be used to convey BIT data from the associated subsystem(s)." Subsystem fault investigation, when indicated by the subsystem flag, is not specified or otherwise restricted by 1553. Therefore, system designers must make the necessary provisions.

4.3.3.5.1.7.15 Selected transmitter shutdown. This command shall cause the RT to disable the transmitter associated with a specified redundant data bus. The command is designed for use with systems employing more than two redundant buses. The transmitter that is to be disabled shall be identified in the data word following the command word in the format as specified in 4.3.3.5.2. The RT shall not comply with a command to shut down a transmitter on the bus from which this command is received. In all cases, the RT shall respond with a status word as specified in 4.3.3.5.3.

4.3.3.5.1.7.16 Override selected transmitter shutdown. This command shall cause the RT to enable a transmitter which was previously disabled. The command is designed for use with systems employing more than two redundant buses. The transmitter that is to be enabled shall be identified in the data word following the command word in the format as specified in 4.3.3.5.2. The RT shall not comply with a command to enable a transmitter on the bus from which this command is received. In all cases, the RT shall respond with a status word as specified in 4.3.3.5.3.
Four mode code commands are provided to control transmitters associated with terminals in a system. These commands can be sent to a single receiver or broadcast to multiple users.

The transmitter shutdown mode code (00100) is used in a dual-redundant bus structure where the command causes the transmitter associated with the other redundant bus to terminate transmissions. No data word is provided for this mode.

The override transmitter shutdown mode code (00101) is used in a dual-redundant bus structure where the command allows the transmitter previously disabled associated with the redundant bus to transmit when commanded by a normal bus command initiated by the active bus controller. No

data word is provided for this mode code.

The selected transmitter shutdown mode code (10100) is used in a multiple (greater than two) redundant bus structure where the command causes the selected transmitter to terminate transmissions on its bus. A data word is used to identify the selected transmitter.

The override selected transmitter shutdown mode code (10101) is used in a multiple (greater than two) redundant bus structure where the command allows the selected transmitter to transmit on its bus when commanded by a normal bus command initiated by the active bus controller. A data word is used to identify the selected transmitter.

4.3.3.5.1.7.17 Reserved mode codes (10110 to 11111). These mode codes are reserved for future use and shall not be used.

Each of the mode code types (with and without data words) have several unused mode codes that are reserved for future use and cannot be used without the permission of the Military Standard's Controlling Agency.

4.3.3.5.2 Data word. A data word shall be comprised of a sync waveform, data bits, and a parity bit (see figure 3).
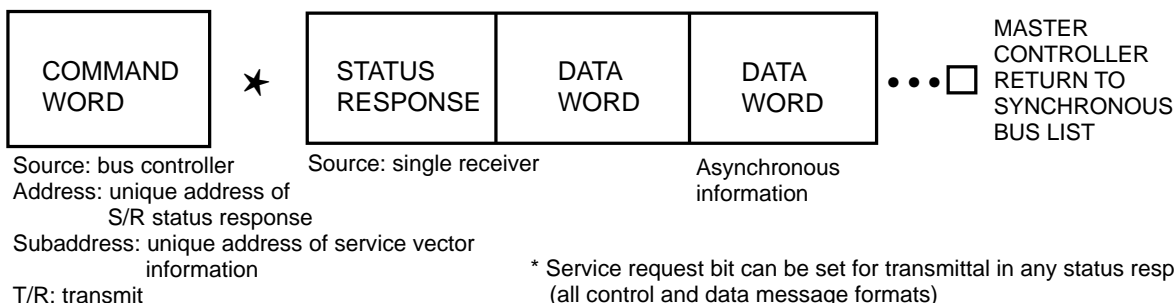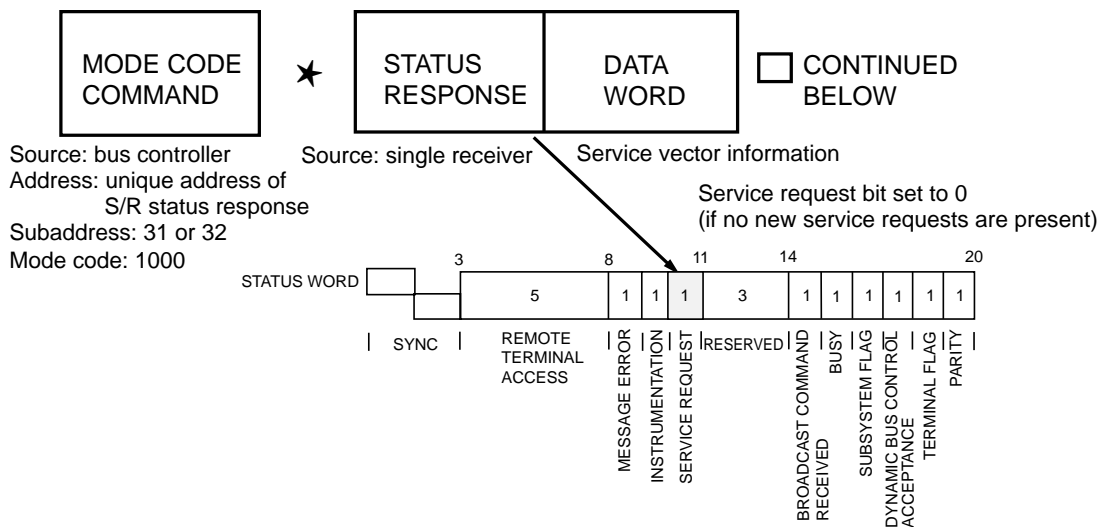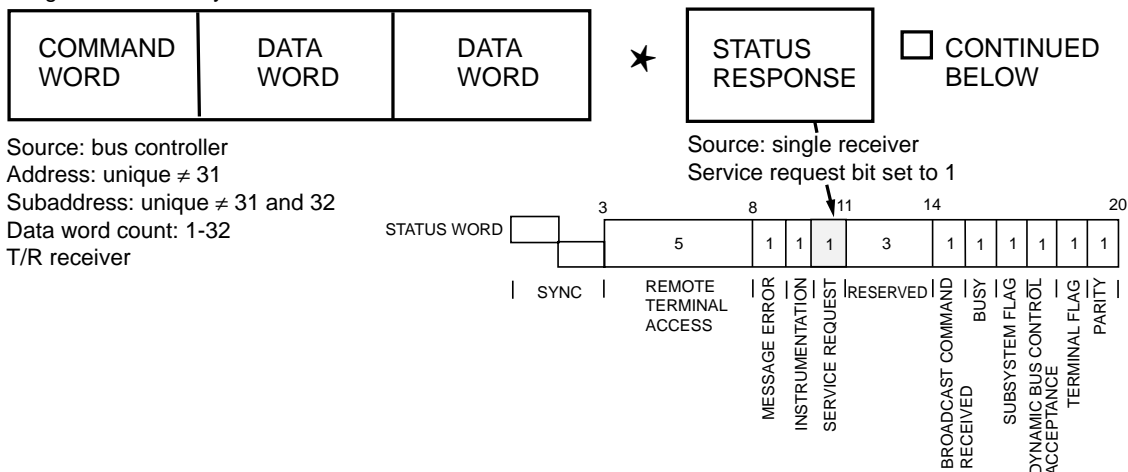
Figure II-4 illustrates the 1553 data word.

4.3.3.5.2.1 Sync. The data sync waveform shall be an invalid Manchester waveform as shown on figure 5. The width shall be three bit times, with the waveform being negative for the first one and one-half bit times, and then positive for the following one and one-half bit times: Note that if the bits preceding and following the sync are logic ones, then the apparent width of the sync waveform will be increased to four bit times.

4.3.3.5.2.2 Data. The sixteen bits following the sync shall be utilized for data transmission as specified in 4 3 2

4.3.3.5.2.3 Parity. The last bit shall be utilized for parity as specified in 4.3.3.5.1.6

Data words are used to transmit parameter data, which is the goal of the information transfer system. Data words are distinguished from command and status words by the inverted three bit sync pattern. Both packed and unpacked data may be transmitted in the 16-bit data field.

Single Receiver Only — Bus Controller to Remote Terminal*

| COMMAND WORD | DATA WORD | DATA WORD | ★ | STATUS RESPONSE | ☐ CONTINUED BELOW |

Source: bus controller
Address: unique ≠ 31
Subaddress: unique ≠ 31 and 32
Data word count: 1-32
T/R receiver

Source: single receiver
Service request bit set to 1

STATUS WORD

| 5 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 |

SYNC — REMOTE TERMINAL ACCESS — MESSAGE ERROR — INSTRUMENTATION — SERVICE REQUEST — RESERVED — BROADCAST COMMAND RECEIVED — BUSY — SUBSYSTEM FLAG — DYNAMIC BUS CONTROL ACCEPTANCE — TERMINAL FLAG — PARITY

| MODE CODE COMMAND | ★ | STATUS RESPONSE | DATA WORD | ☐ CONTINUED BELOW |

Source: bus controller
Address: unique address of S/R status response
Subaddress: 31 or 32
Mode code: 1000

Source: single receiver

Service vector information

Service request bit set to 0
(if no new service requests are present)

STATUS WORD

| 5 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 |

SYNC — REMOTE TERMINAL ACCESS — MESSAGE ERROR — INSTRUMENTATION — SERVICE REQUEST — RESERVED — BROADCAST COMMAND RECEIVED — BUSY — SUBSYSTEM FLAG — DYNAMIC BUS CONTROL ACCEPTANCE — TERMINAL FLAG — PARITY

| COMMAND WORD | ★ | STATUS RESPONSE | DATA WORD | DATA WORD | • • • ☐ | MASTER CONTROLLER RETURN TO SYNCHRONOUS BUS LIST |

Source: bus controller
Address: unique address of S/R status response
Subaddress: unique address of service vector information
T/R: transmit

Source: single receiver

Asynchronous information

* Service request bit can be set for transmittal in any status response (all control and data message formats)
★ Response time delay gap
☐ End of message delay or gap

Figure II-3. Transmit Vector Word Transfer Format
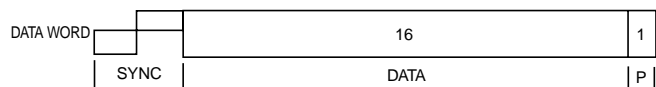
Figure II-4. 1553 Data Word

Odd parity on the data field provides data integrity identical to the command and status word formats. No changes in the 1553A or 1553B have occurred in these paragraphs except for paragraph numbering (e.g, 4.2.3.5.2 for 1553A and 4.3.3.5.2 for 1553B).

4.3.3.5.3. Status word. A status word shall be comprised of a sync waveform, RT address, message error bit, instrumentation bit, service request bit, three reserved bits, broadcast command received bit, busy bit, subsystem flag bit, dynamic bus control bit, terminal flag bit, and a parity bit. For optional broadcast operation, transmission of the status word shall be suppressed as specified in 4 3 3 6 7.

4.3.3.5.3.1 Sync. The status sync waveform shall be as specified in 4.3.3.5.1.1.

4.3.3.5.3.2 RT address. The next five bits following the sync shall contain the address of the RT which is transmitting the status word as defined in 4.3.3.5.1.2.

The status word is part of the basic overhead requirements of the data bus system. The status word is shown in figure II-5 is divided into the following fields:

a. *Sync (same as command sync)*
b. *Terminal address*
c. *Status field*
d. *Parity (P)*

The five-bit address field identifies the transmitting terminal's address, while the remote terminal's status is based on bits set in the status field. The status field consists of the following information:

a. *Message error bit*
b. *Instrumentation bit*

Figure 5 of 1553B. Data Sync

c. *Service request bit*
d. *Reserved field*
e. *Broadcast command received bit*
f. *Busy bit*
9. *Subsystem flag*
h. *Dynamic bus control acceptance bit*
i. *Terminal flag*

4.3.3.5.3.3 Message error bit. The status word bit at bit time nine (see figure 3) shall be utilized to indicate that one or more of the data words associated with the preceding receive command word from the bus controller has failed to pass the RT's validity tests as specified in 4.4.1.1 This bit shall also be set under the conditions specified in 4.4.1.2, 4.4.3.4 and 4.4.3.6. A logic one shall indicate the presence of a message error, and a logic zero shall show its absence All RT's shall implement the message error bit.

The message error bit is set to logic one to indicate that one or more of the data words associated with the preceding received message failed to pass the message validity test. Message validity requirements are:

a. *Word validation — word begins with valid sync, Manchester II code correctly transmitted, 16 data bits plus parity, and word parity odd*
b. *Contiguous words within a message*
c. *Address validation — matches address unique terminal or broadcast address*
d. *Illegal command — a terminal with the illegal command detection circuitry detects an illegal command*

The status word will be transmitted if the message validity requirements are met (see para. 4.4.3.5 and 4.4.3.6). When a message error occurs in a broadcast message format, the message error bit will be set in the status word and the status response withheld as required by broadcast message format.

4.3.3.5.3.4 Instrumentation bit. The status word bit time of 10 (see figure 3) shall be reserved for the instrumentation bit and shall always be a logic zero.
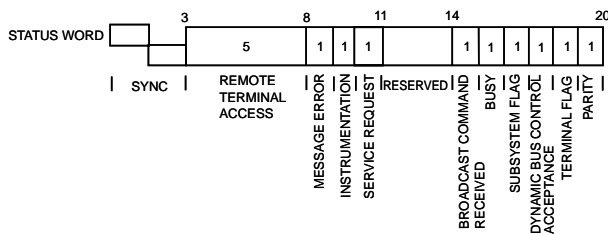

Figure II-5. Status Word

This bit is intended to be used in conjunction with a logic one in bit time 10 of the command word to distinguish between a command word and a status word. The use of the instrumentation bit is optional.

The instrumentation bit in the status field is set to distinguish the status word from the command word. Since the sync field (three bits) is used to distinguish the command and status words from a data word, a mechanism to distinguish command and status is provided by the instrumentation bit. By setting this bit to logic zero for all conditions and setting the same bit position in the command word to a logic one, the command and status words are identifiable. If used, this approach reduces the possible subaddress in the command word to 15 and requires subaddress 31 (11111) to be used to identify mode commands (both 31 and 32 are allowed). If not used, the bit will remain set to logic zero in the status word for all conditions.

4.3.3.5.3.5 Service request bit. The status word bit at bit time eleven (see figure 3) shall be reserved for the service request bit. The use of this bit is optional. This bit, when used, shall indicate the need for the bus controller to take specific predefined actions relative to either the RT or associated subsystem. Multiple subsystems, interfaced to a single RT, which individually require a service request signal shall logically OR their individual signals into the single status word bit. In the event this logical OR is performed, then the designer must make provisions in a separate data word to identify the specific requesting subsystem. The service request bit is intended to be used only to trigger data transfer operations which take place on an exception rather than periodic basis. A logic one shall indicate the presence of a service request, and a logic zero its absence. If this function is not implemented, the bit shall be set to zero.

The service request bit is provided to indicate to the active bus controller that a remote terminal requests service. When this bit in the status word is set to logic one, the active bus controller may take a predetermined action or use mode command (transmit vector word) to identify the specific request. The message format for acquiring this is discussed under transmit vector word mode command (see fig. II-3).

4.3.3.5.3.6 Reserved status bits. The status word bits at bit times 12 through 14 are reserved for future use and shall not be used. These bits shall be set to a logic zero.

The three bit-field (12-14) is reserved for future

requirements and is set to logic zero. Any bit in this field not set to logic zero will be disregarded.

4.3.3.5.3.7 Broadcast command received bit. The status word at bit time 15 shall be set to a logic one to indicate that the preceding valid command word was a broadcast command and a logic zero shall show it was not a broadcast command. If the broadcast command option is not used, this bit shall be set to a logic zero.

The broadcast command received bit is set to logic one when the preceding valid command word was a broadcast command (address 31). Since broadcast message formats require the receiving remote terminals to suppress their status words, the broadcast command received bit is set to identify that the command was received properly. If the broadcast message validity is desired, the message format shown in figure II-6 is used to determine this information. The broadcast command received bit will be reset when the next valid command is received by the remote terminal, unless the next valid command is transmit status word or transmit last command.

4.3.3.5.3.8 Busy bit. The status word bit at bit time 16 (see figure 3) shall be reserved for the busy bit. The use of this bit is optional. This bit, when used, shall indicate that the RT or subsystem is unable to move data to or from the subsystem in compliance with the bus controller's command. A logic one shall indicate the presence of a busy condition, and a logic zero its absence. In the event the busy bit is set in response to a transmit command, then the RT shall transmit its status word only. If this function is not implemented, the bit shall be set to logic zero.

The busy bit in the status word is set to logic one to indicate to the active bus controller that the remote terminal is unable to move data to or from the subsystem in compliance with the bus controller's command. The message format associated with a busy condition is shown in figure II-7. A busy condition can exist within a remote terminal at any time causing it to be nonresponsive to a command to send data or to be unable to receive data. This condition can exist for all message formats. In each case except the broadcast message formats, the active bus controller will determine the busy condition immediately upon status response. In the case of the broadcast message formats, this information will not be known unless the receiving terminals are polled after the broadcast message requesting their status. If the status word has the broadcast received bit set, the message was received and the terminal was not busy.

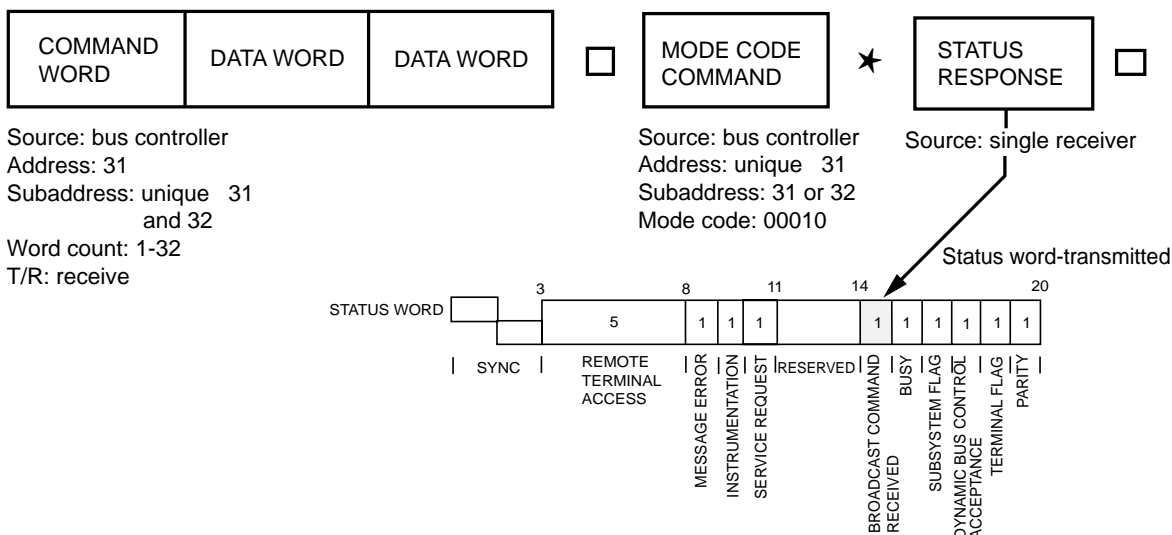**Multiple Receivers — Bus Controller to Remote Terminals**

Figure II-6. Broadcast Command Receive Bit

4.3.3.5.3.9 Subsystem flag bit. The status word bit at bit time 17 (see figure 3) shall be reserved for the subsystem flag bit. The use of this bit is optional. This bit, when used, shall flag a subsystem fault condition, and alert the bus controller to potentially invalid data. Multiple subsystems, interfaced to a single RT, which individually require a subsystem flag bit signal shall logically OR their individual signals into the single status word bit. In the event this logical OR is performed, then the designer must make provisions in a separate data word to identify the specific reporting subsystem. A logic one shall indicate the presence of the flag, and a logic zero its absence. If not used, this bit shall be set to logic zero.

The subsystem flag bit is provided to indicate to the active bus controller that a subsystem fault condition exists and that data being requested from the subsystem may be invalid. The subsystem flag may be set in any transmitted status word.

4.3.3.5.3.10 Dynamic bus control acceptance bit. The status word bit at bit time 18 (see figure 3) shall be reserved for the acceptance of dynamic bus control. This bit shall be used if the RT implements the optional dynamic bus control function. This bit, when used, shall indicate acceptance or rejection of a dynamic bus control offer as specified in 4.3.3.5.1.7.1. A logic one shall indicate acceptance of control, and a logic zero shall indicate rejection of control. If this function is not used, this bit shall be set to logic zero.

This bit is provided to indicate the acceptance of the bus controller offer by the active bus controller to become the next bus controller. The offer of bus control occurs when the presently active bus controller has completed its established message list and issues a dynamic bus control mode command to the remote terminal that is to be the next potential controller. To

Single Receiver — Bus Controller to Remote Terminal

★ Response time delay or gap

☐ End of message delay or gap
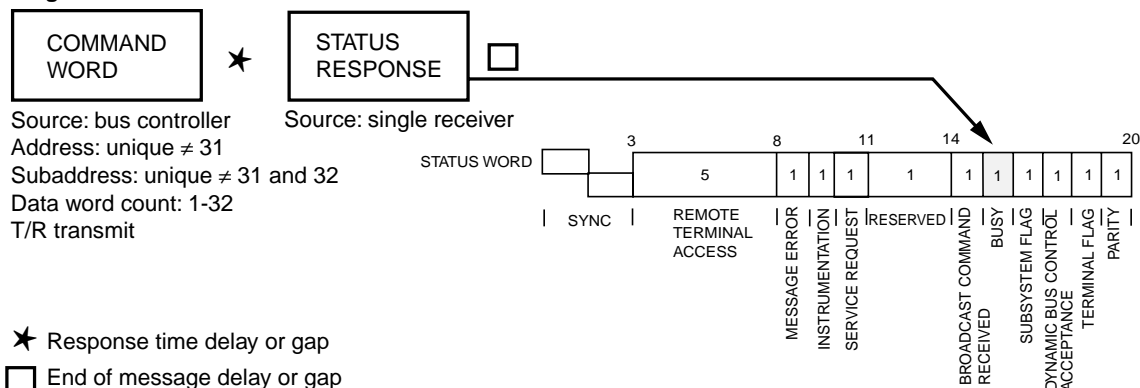
Figure II-7. Busy Bit

accept the offer the potential bus controller sets its dynamic bus control acceptance bit in the status word and transmits the status word. The establishment of who the next potential controller will be is a system issue.

4.3.3.5.3.11 Terminal flag bit. The status word bit at bit time 19 (see figure 3) shall be reserved for the terminal flag function. The use of this bit is optional. This bit, when used, shall flag a RT fault condition. A logic one shall indicate the presence of the flag, and a logic zero, its absence If not used, this bit shall be set to logic zero.

The terminal flag bit is set to a logic one to indicate a fault within the remote terminal. This bit is used in connection with three mode code commands:

a. Inhibit T/F flag
b. Override inhibit T/F flag
c. Transmit BIT word

The first two mode code commands deactivate and activate the functional operation of the bit. The transmit BIT word mode code command is used to acquire more detailed information about the terminal's failure.

4.3.3.5.3.12 Parity bit. The least significant bit in the status word shall be utilized for parity as specified in 4 3.3 5.1.6.

The use of a single parity bit per word was provided to identify any bit errors occurring during the transmission and detection of a word. This odd parity check will detect an odd number of bit errors occurring in a word. This requirement produces an undetected bit error rate of 10-'2, which was considered satisfactory for a general-purpose information transfer system. This paragraph remained unchanged during both revisions. See also 1553B, paragraph 4.3.3.5.1.6.

4.3.3.5.4. Status word reset. The status word bit with the exception of the address, shall be set to logic zero after a valid command word is received by the RT with the exception as specified in 4.3.3.5.1.7. If the conditions which caused bits in the status word to be set (e.g., terminal flag) continue after the bits are reset to logic zero, then the affected status word bit shall be again set, and then transmitted on the bus as required.

This paragraph was added to 1553B to clarify the hardware requirements associated with resetting the status code field of the status word. Figure II-5

shows the status word and the information available in this field.

One reason for the reset definition is to provide:

a. The ability to obtain the latest status information of the remote terminal: this prevents conditions from being reported for longer than they actually exist.

b. The ability to obtain the status code analysis of the previous results of a valid command: this allows an orderly error handling and recovery approach to be accomplished by the bus controller with the information associated with error analysis data contained within this field or other data associated within the RT (e.g., last command word and BIT).

The second reason for obtaining the status code field not reset was to allow error recovery using two mode codes of paragraph 4.3.3.5.1.7. Even though all mode codes are referenced in the status word reset paragraph, only two are required to retain the last status word in the terminal:

a. Transmit status word
b. Transmit last command word

In other words, all other valid messages received, including mode commands, will allow the RT to reset the status word, except these two.

Both of these mode codes can be transmitted to the RT without changing the bits in the status code field of the last valid command word in question. Therefore, it is essential that an error recovery procedure be established for the bus controller that takes into account (1) the ability of the RT hardware to collect error data, (2) the format of the data that must be requested by the bus controller to prevent data lost, and (3) the ability of the bus controller hardware and software to receive and react to these data. As many as three mode codes may be involved in this process:

a. Transmit last command
b. Transmit status word
c. Transmit BIT word

4.3.3.6 Message formats. The messages transmitted on the data bus shall be in accordance with the formats on figure 6 and figure 7. The maximum and minimum response times shall be as stated in 4.3.3.7 and 4.3.3.8. No message formats, other than those defined herein, shall be used on the bus.

The 1553B section of the standard contains two additional message format description that are not contained in revision A. One of these is an explanation of the optional mode code message format that
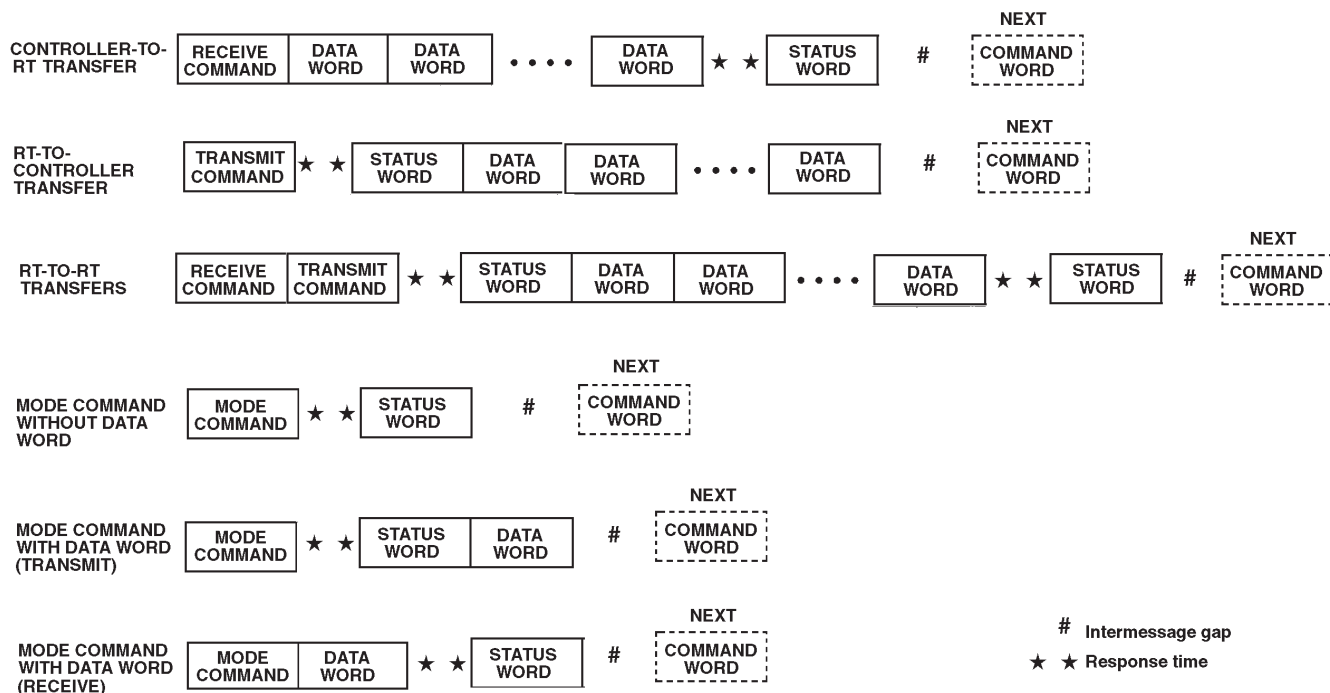
Figure 6 of 1553B. Information Transfer Formats

is allowed in revision A but not described with message format diagrams. The other description is the message formats associated with the optional broadcast protocol. The command/response protocol provides two types of message formats:

a. Data massages
b. Control messages

4.3.3.6.1 Bus controller to remote terminal transfers. The bus controller shall issue a receive command followed by the specified number of data words. The RT shall, after message validation, transmit a status word back to the controller. The command and data words shall be transmitted in a contiguous fashion with no interword gaps.

4.3.3.6.2 Remote terminal to bus controller transfers. The bus controller shall issue a transmit command to the RT. The RT shall after command word validation, transmit a status word back to the bus controller, followed by the specified number of data words. The status and data words shall be transmitted in a contiguous fashion with no interword gaps.
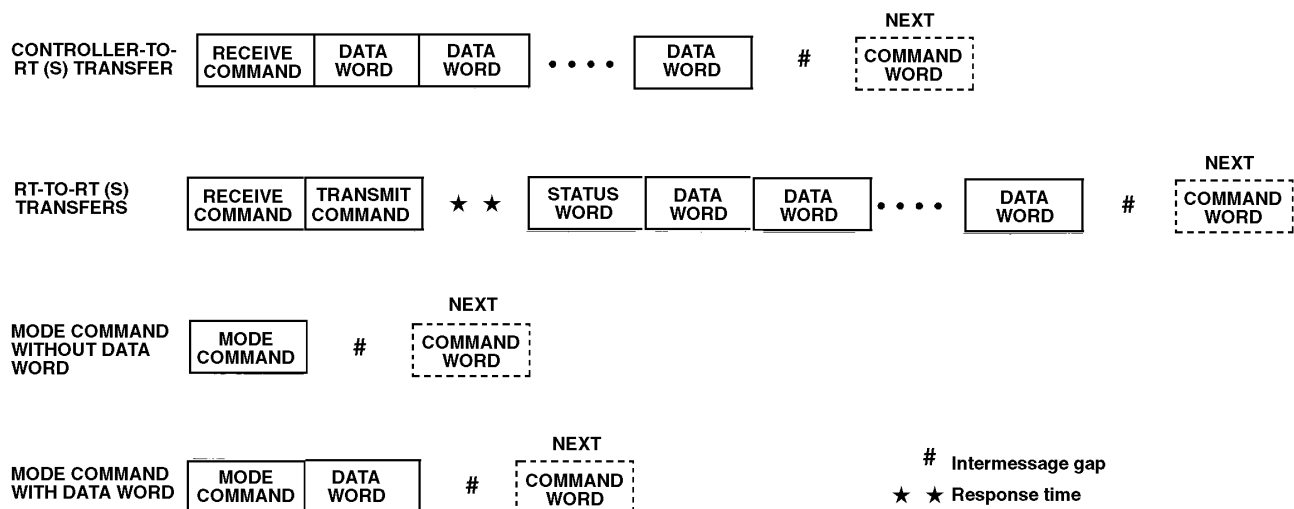
Figure 7 of 1553B. Broadcast Information Transfer Formats

4.3.3.6.3 Remote terminal to remote terminal transfers. The bus controller shall issue a receive command to RT A followed contiguously by a transmit command to RT B. RT B shall, after command verification, transmit a status word followed by the specified number of data words. The status and data words shall be transmitted in a contiguous fashion with no gap. At the conclusion of the data transmission by RT B, RT A shall transmit a status word within the specified time period.

4.3.3.6.4. Mode command without data word. The bus controller shall issue a transmit command to the RT using a mode code specified in table I. The RT shall, after command word validation transmit a status word.

4.3.3.6.5 Mode command with data word (transmit). The bus controller shall issue a transmit command to the RT using a mode code specified in table I. The RT shall, after command word validation, transmit a status word followed by one data word. The status word and data word shall be transmitted in a contiguous fashion with no gap.

4.3.3.6.6 Mode command with data word (receive). The bus controller shall issue a receive command to the RT using a mode code specified in table I, followed by one data word. The command word and data word shall be transmitted in a contiguous fashion with no gap. The RT shall, after command and data word validation, transmit a status word back to the controller.

4.3.3.6.7 Optional broadcast command. See 10.6 for additional information on the use of the broadcast command.

4.3.3.6.7.1 Bus controller to remote terminal(s) transfer (broadcast). The bus controller shall issue a receive command word with 11111 in the RT address field followed by the specified number of data words. The command word and data words shall be transmitted in a contiguous fashion with no gap. The RT(s) with the broadcast option shall after message validation, set the broadcast command received bit in the status word as specified in 4.3.3.5.3.7 and shall not transmit the status word.

4.3.3.6.7.2 Remote terminal to remote terminal(s) transfers (broadcast). The bus controller shall issue a receive command word with 11111 in the RT address field followed by a transmit command to RT A using the RT's address. RT A shall, after command word validation, transmit a status word followed by the specified

number of data words. The status and data words shall be transmitted in a contiguous fashion with no gap. The RT(s) with the broadcast option, excluding RT A, shall after message validation, set the broadcast received bit in the status word as specified in 4.3.3.5.3.7 and shall not transmit the status word.

4.3.3.6.7.3 Mode command without data word (broadcast). The bus controller shall issue a transmit command word with 11111 in the RT address field, and a mode code specified in table I. The RT(s) with the broadcast option shall after command word validation, set the broadcast received bit in the status word as specified in 4.3.3.5.3.7 and shall not transmit the status word.

4.3.3.6.7.4 Mode command with data word (broadcast). The bus controller shall issue a receive command word with 11111 in the RT address field and a mode code specified in table I, followed by one data word. The command word and data word shall be transmitted in a contiguous fashion with no gap. The RT(s) with the broadcast option shall, after message validation set the broadcast received bit in the status word as specified in 4.3.3.5.3.7 and shall not transmit the status word.

Data messages are used to communicate subsystem data to meet the purpose of the integration. As in the control messages, there are two message types: single receiver and multiple receiver messages. These are transmitted in the following manner:

Single receiver
a. Bus controller to remote terminal
b. Remote terminal to bus controller
c. Remote terminal to remote terminal

Multiple receivers
a. Bus controller to multiple remote terminals
b. Remote terminal to multiple remote terminals

Each of these messages is transmitted using command and status words for control operation. The command word is used to:

a. Identify the receiving terminal(s)
b. Identify if data are to be received or transmitted by the receiving terminal(s)
c. Identify the specific message identification (sub-address) within the remote terminal(s)
d. Notify the terminal(s) of the number of data words to be received or transmitted

The command word, status word and data word format for accomplishing these messages are described in 4.3.3.5.1, 4.3.3.5.2, and 4.3.3.5.3.. Using these words,
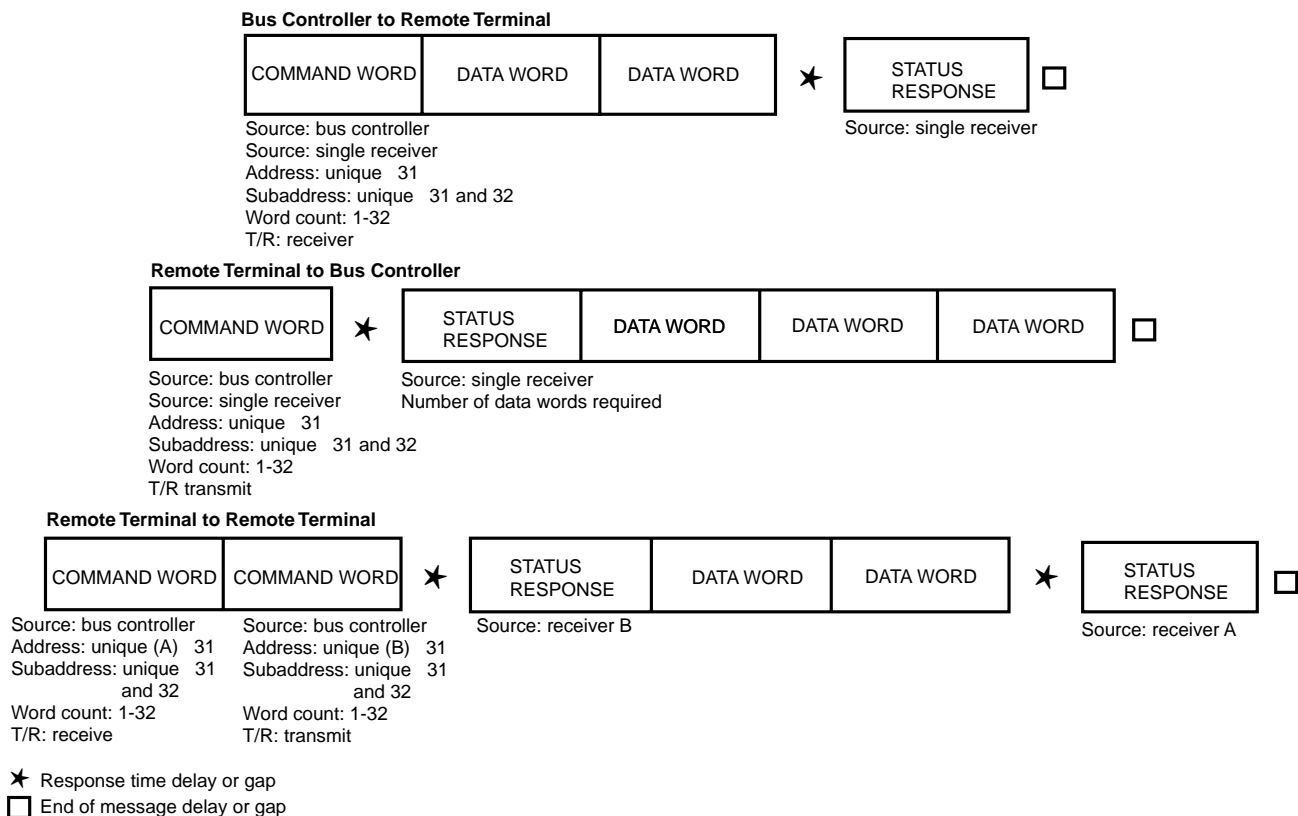
Bus Controller to Remote Terminal

| COMMAND WORD | DATA WORD | DATA WORD | ★ | STATUS RESPONSE | □ |

Source: bus controller
Source: single receiver
Address: unique   31
Subaddress: unique   31 and 32
Word count: 1-32
T/R: receiver

Source: single receiver

Remote Terminal to Bus Controller

| COMMAND WORD | ★ | STATUS RESPONSE | DATA WORD | DATA WORD | DATA WORD | □ |

Source: bus controller
Source: single receiver
Address: unique   31
Subaddress: unique   31 and 32
Word count: 1-32
T/R transmit

Source: single receiver
Number of data words required

Remote Terminal to Remote Terminal

| COMMAND WORD | COMMAND WORD | ★ | STATUS RESPONSE | DATA WORD | DATA WORD | ★ | STATUS RESPONSE | □ |

Source: bus controller
Address: unique (A)   31
Subaddress: unique   31
            and 32
Word count: 1-32
T/R: receive

Source: bus controller
Address: unique (B)   31
Subaddress: unique   31
            and 32
Word count: 1-32
T/R: transmit

Source: receiver B

Source: receiver A

★ Response time delay or gap
□ End of message delay or gap

Figure II-8. Single-Receiver Data Message Formats

Bus Controller to Remote Terminals

| COMMAND WORD | DATA WORD | DATA WORD | DATA WORD | □ |

Source: bus controller
Address: 31
Broadcast: 31
Subaddress: unique   31 and 32
Word count: 1-32
T/R: receive

Remote Terminals to Remote Terminals

| COMMAND WORD | COMMAND WORD | ★ | STATUS RESPONSE | DATA WORD | DATA WORD | DATA WORD | □ |

Source: bus controller
Address: 31
Broadcast: 31
Subaddress: unique   31
            and 32
Word count: 1-32
T/R: receive

Source: bus controller
Address: unique A
Subaddress: unique   31
            and 32
Word count: 1-32
T/R: receive

Source: receiver A

★ Response time delay or gap
□ End of message delay or gap

Figure II-9. Multiple-Receiver Data Message Formats

the format for data message transmissions is developed. The single receiver data message formats are shown in figure II-8. The message formats for multiple receiving terminals are shown in figure II-9.

Mode commands are used to manage the data bus system and are considered a necessary overhead requirement to properly control the data flow. The overhead requirements are provided by command words and status words. These header words to each data transmission are required to maintain system data flow within the multiplex system. Command and status words are associated with both control messages and data messages. Message formats within this protocol can be transmitted to a single receiver or to multiple receivers based upon the command word address for the message.

Mode commands are identified by the subaddress/mode field in the command word being set to 32 (00000) or 31 (11111). All control messages originate with the bus controller and are received by a single receiver or by multiple receivers (broadcast). A terminal address value of 31 (11111) in the command word indicates a broadcast message, and any other terminal address is used to identify unique mode commands to terminals on the bus. The mode command information is in the word count/mode code field of the command word and in the attached data word if allowed by the mode command.

The various legal mode commands without and with data word are illustrated in figure II-10.

4.3.3.7 Intermessage gap. The bus controller shall provide a minimum gap time of 4.0 microseconds (µs) between messages as shown on figure 6 and figure 7. This time period, shown as T on figure 8, is measured at point A of the bus controller as shown on figure 9 or figure 10. The time is measured from the mid-bit zero crossing of the last bit of the preceding message to midzero crossing of the next command word sync.

This paragraph in the 1553B expands the requirements of the response time (par. 4.3.1 in 1553A), by adding this intermessage gap paragraph. The purpose was to clearly identify that the bus controller shall not transmit contiguous messages (must have a gap) and that the maximum response time (12 µs par. 4.3.3.8) does not apply to gaps between messages. The bus controller may issue messages with a gap time greater than 4 µs.

4.3.3.8 Response time. The RT shall respond, in accordance with 4.3.3.6, to a valid command word within the time period of 4.0 to 12.0 µs. This time
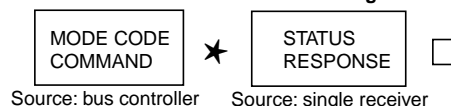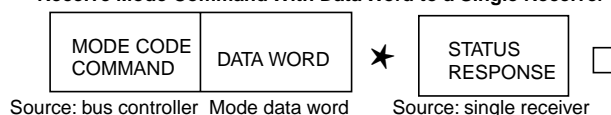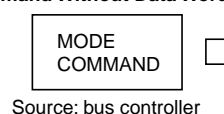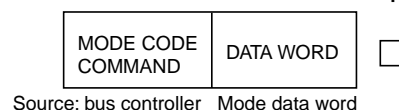
Figure II-10. Mode Command Transfer Formats

period, shown as T on figure 8, is measured at point A of the RT as shown on figure 9 or figure 10. The time is measured from the mid-bit zero crossing of the last word as specified in 4.3.3.6 and as shown on figure 6 and figure 7 to the midzero crossing of the status word sync.

This paragraph in 1553B relates to paragraph 4.3.1 in 1553A with the following changes:

a. The maximum response time was increased by 100% (5 to 10 µs or 7 to 12 µs when using the measurement techniques described below).

b. The point of measurement to establish the time was identified and was chosen to be a different point than was usually interpreted to be in 1553A. The 4 to 12 µs response time will allow more hardware
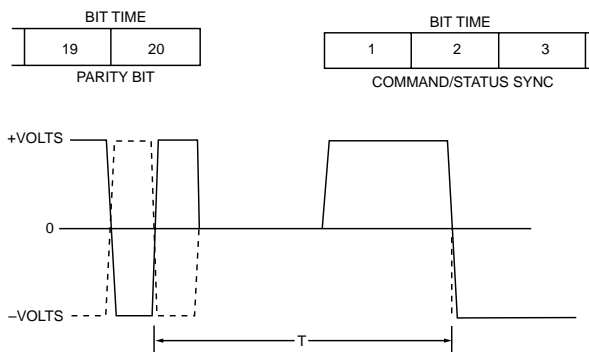
Figure 8 of 1553B. Intermessage Gap and Response Time

design flexibility in the multiplex interface area. Also, the measurement technique was undefined in 1553A and because it is hard to determine when the multiplex line is quiet (dead), the measurement is easier to make if the previous mid-bit (zero) crossing and next mid-bit crossing are examined.

4.3.3.9 Minimum no-response time-out. The minimum time that a terminal shall wait before considering that a response as specified in 4.3.3.8 has not occurred shall be 14.0 μs. The time is measured from the mid-bit zero crossing of the last bit of the last word to the mid-zero crossing of the expected status word sync at Point A of the terminal as shown on figure 9 or figure 10.

This new requirement of 1553B is provided to clarify the minimum time that a bus controller shall wait before concluding that the RT is not going to respond as requested. This is measured from the end of its transmission (last mid-bit crossing) to the expected response (first mid-bit crossing). Notice that this represents the minimum wait time on the same bus where the previous message was requested from the RT.

4.4 Terminal Operation

This paragraph in 1553B was provided to clarify the various terminals identified in the standard and their performance requirements. The first section covers common operational requirements that apply to all devices connected to the data bus system. Specific requirements include bus controller (par. 4.4.2), remote terminal (par. 4.4.3), and bus monitor (par. 4.4.4).

4.4.1 Common operation. Terminals shall have common operating capabilities as specified in the following paragraphs.

4.4.1.1 Word validation. The terminal shall insure that each word conforms to the following minimum criteria:

a. The word begins with a valid sync field.
b. The bits are a valid Manchester II code.
c. The word parity is odd.
When a word fails to conform to the preceding criteria, the word shall be considered invalid.

4.4.1.2 Transmission continuity. The terminal shall verify that the message is contiguous as defined in 4.3.3.6. Improperly timed data syncs shall be considered a message error.

4.4.1.3 Terminal fail-safe. The terminal shall contain a hardware implemented timeout to preclude a signal transmission of greater than 800.0 μs. This hardware shall not preclude a correct transmission in response to a command. Reset of this timeout function shall be performed by the reception of a valid command on the bus on which the timeout has occurred.

This paragraph describes the common operation associated with terminals connected to the data bus system. Performance requirements include: word validation, transmission continuity, and terminal fail-safe.

The word validation paragraph has been modified to explain more fully the detection and response required. These requirements are contained in 1553B paragraphs (word validation, 4.4.1.1), (transmission continuity, 4.4.1.2), and (invalid data reception, 4.4.3.6). The new and modified paragraphs should provide sufficient information concerning invalid words or invalid messages.

The terminal fail-safe requirement prevents excessive transmissions on a data bus by a single transmitter, which would preclude its effective use. Changes in 1553B to 800 μs (par 4.4.1.3) instead of 660 μs (par. 4.3.2) will allow less accurate analog or relaxed digital timers with more independence of the timer circuits to be used in the current design. In addition there were several mechanisms to reset this fail-safe timer described in 1553B. These include the following:

a. Reset of this timeout function shall be performed by the reception of a valid command on the bus on which the timeout has occurred (par. 4.4.1.3).

b. The mode command override transmitter shutdown (00101 for two-bus system or 10101 for multiple-bus system) on an alternative bus can also be used to reset the timer.

c. The mode command reset remote terminal (01000) causes the remote terminal to assume a power-up initialized state that can also be used to reset the timer.
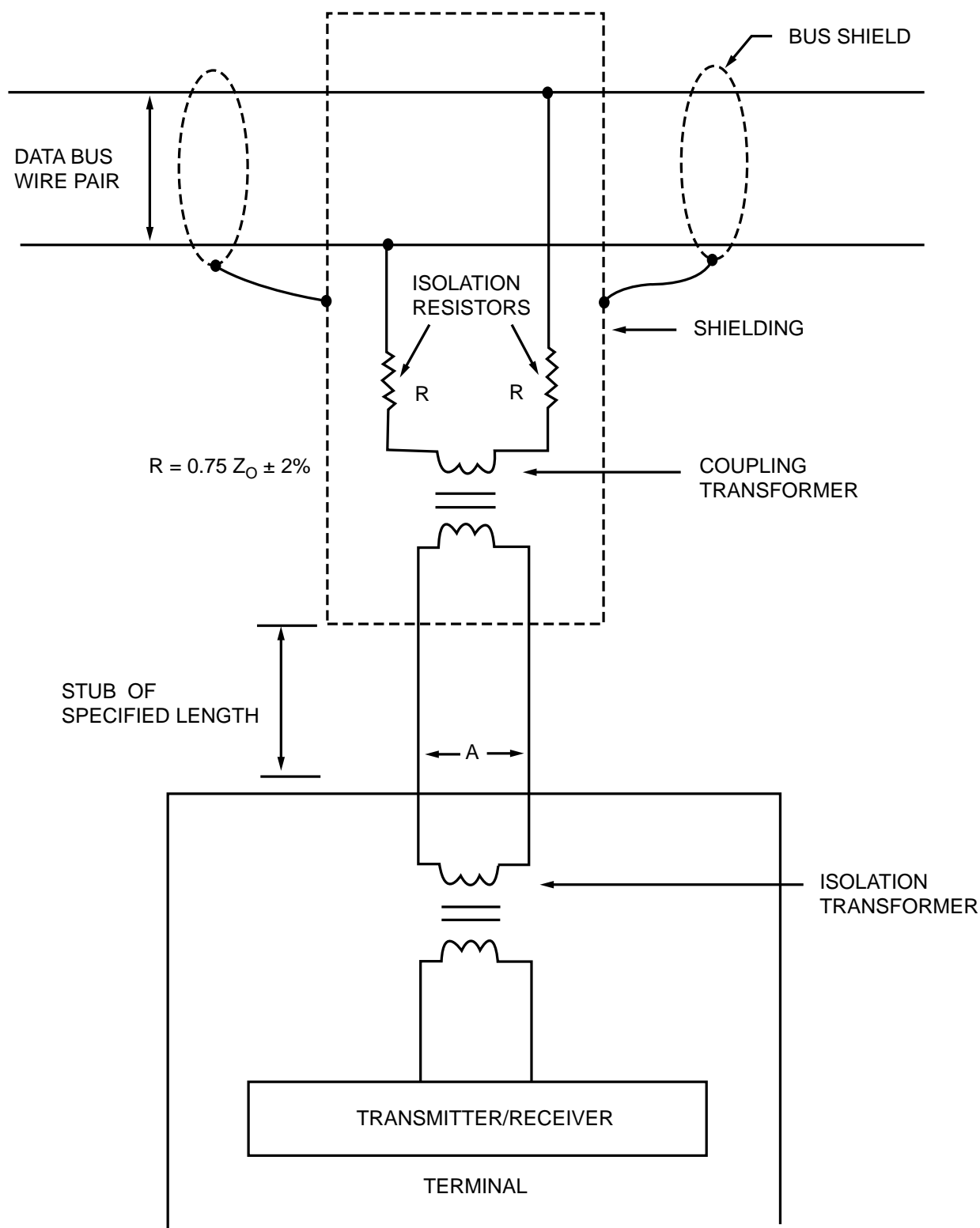
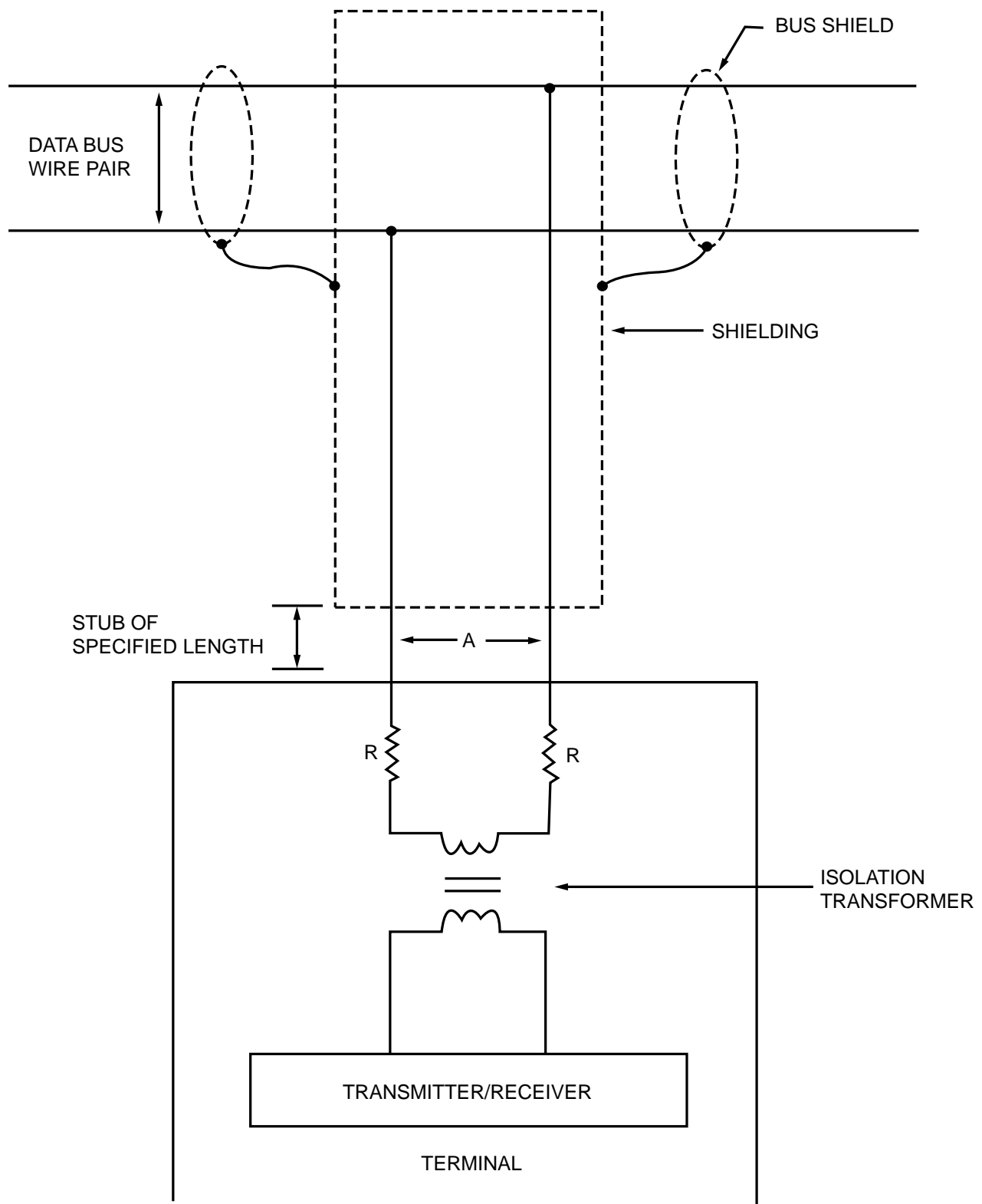Figure 9 of 1553B. Data Bus Interface Using Transformer Coupling

BUS SHIELD

DATA BUS
WIRE PAIR

SHIELDING

STUB OF
SPECIFIED LENGTH

A

R   R

ISOLATION
TRANSFORMER

TRANSMITTER/RECEIVER

TERMINAL

Figure 10 of 1553B. Data Bus Interface Using Direct Coupling

Both b and c are optional ways to reset the timer and may depend on the system and hardware implementations. However, the preferred reset approach is to transmit the appropriate mode code.

4.4.2 Bus controller operation. A terminal operating as a bus controller shall be responsible for sending data bus commands, participating in data transfers, receiving status responses, and monitoring system status as defined in this standard. The bus controller function may be embodied as either a stand-alone terminal, whose sole function is to control the data bus(s), or contained within a subsystem. Only one terminal shall be in active control of a data bus at any one time.

This paragraph is generally the same paragraph in both 1553A (4.5) and 1553B.

4.4.3 Remote Terminal

4.4.3.1 Operation. A remote terminal (RT) shall operated in response to valid commands received from the bus controller. The RT shall accept a command word as valid when the command word meets the criteria of 4.4.1.1, and the command word contains a terminal address which matches the RT address or an address of 11111, if the RT has the broadcast option.

The remote terminal operation has been expanded in 1553B to include the broadcast option and additional definitions associated with: superseding valid commands (par. 4.4.3.2), invalid commands (par. 4.4.3.3), illegal commands (par. 4.4.3.4), valid data reception (par 4i4.3.6), and invalid data reception (par. 4.4.3.6).

4.4.3.2 Superseding valid commands. The RT shall be capable of receiving a command word on the data bus after the minimum intermessage gap time as specified in 4.3.3.7 has been exceeded, when the RT is not in the time period T as specified in 4.3.3.8 prior to the transmission of a status word, and when it is not transmitting on that data bus. A second valid command word sent to an RT shall take precedence over the previous command. The RT shall respond to the second valid command as specified in 4.3.3.8.

The superseding valid command requirement clarifies the gap time issue in 1553A (par.4.3). "A second valid command word sent to a terminal after it is already operating on one shall invalidate the first command and cause the RT to begin operation on the second command." This phrase in 1553A can be misinterpreted to indicate that near back-to-back (no gap or only 4 µs gap) commands can be sent to a ter-

minal on the same bus and the terminal will respond to the second command.

However, this was not the intention because in certain cases an RT responding with a status word slightly greater than 4 µs would collide with the second command being transmitted by the bus controller. The intended purpose for this requirement is to allow the bus controller to reissue an identical transmission or issue a new transmission on the same bus to the same RT, when an RT fails to respond to a command on that bus. This method is described in the 1553B by requiring a minimum time T (greater than a gap time but less than a full 32-word message) to occur prior to transmitting the second command. Therefore, the bus controller is assured that the RT is not responding and a new command on the same bus is appropriate. Figure 8 in 1553B demonstrates this intermessage gap problem and solution.

4.4.3.3 Invalid commands. A remote terminal shall not respond to a command word which fails to meet the criteria specified in 4.4.3.1.

Command words that fail to meet the word validation requirement cause the system to continue to "look" for a valid command word. When this condition occurs, no change occurs to the status word and no response is transmitted by the RT. This operation is identified as invalid command. This paragraph is used to cover failure in the decoding process of the command word. To prevent multiple responses by two or more terminals to a command word (one without a failure and one with) the terminal that cannot absolutely validate a command word must take the safe approach and reset the circuitry and continue to look for a valid command word that meets it particular requirements (address). All RT's should use this approach of not responding, when there is a question about the commands. This approach is considered to be a fail-passive approach providing the least impact on the multiplex system.

4.4.3.4 Illegal command. An illegal command is a valid command as specified in 4.4.3.1, where the bits in the subaddress/mode field, data word count/mode code field, and the T/R bit indicate a mode command, subaddress, or word count that has not been implemented in the RT. It is the responsibility of the bus controller to assure that no illegal commands are sent out. The RT designer has the option of monitoring for illegal commands. If an RT that is designed with this option detects an illegal command and the proper number of contiguous valid data words as specified by the illegal command word, it shall respond with a status word only, setting

the message error bit, and not use the information received.

Illegal commands are command words that have passed the word validation test but do not comply with the system's capability. These include command words where the subaddress-mode field, data word/mode code field, or the T/R bit are set so that they represent conditions not allowed in the system. These include both conditions not allowed by the standard and any additional condition not allowed in a particular system design. The responsibility for not allowing illegal commands to be transmitted is given to the bus controllers. Since the bus controller is responsible for all command/ response message communications, it will be a design goal that the bus controller not transmit an invalid command.

Two methods can be provided to meet this requirement: (1) careful generation of bus controller commands in the development of the system and tight control of the change process during operational use and (2) examination of failure modes of the controller hardware and software to determine potentially illegal command generations and transmissions. An additional method of rejecting illegal commands in the multiplex system can only be provided by circuitry within the receiving remote terminal. This approach is an optional capability for remote terminals built to the 1553B standard. If an RT with this capability detects an illegal command that meets all other validation requirements, the RT shall respond with a status word with only the message error bit set and not use the information sent or disregard the request for information.

4.4.3.5 Valid data reception. The remote terminal shall respond with a status word when a valid command word and the proper number of contiguous valid data words are received, or a single valid word associated with a mode code is received. Each data word shall meet the criteria specified in 4.4.1.1.

The purpose of the valid data reception in 1553B was to clearly state when a message containing at least one data word would be responded to by the appropriate RT. Previous systems have taken different approaches to messages with various failures (e.g., under word count, over word count, parity errors in words, gaps in word transmissions). Therefore, this requirement was established to identify the only time status words would be transmitted by the RT after the reception of a data message with at least one data word. It should be noted that one other message format will produce a status word response: mode code without data word trans-mitted to a specific RT (not broadcast).

4.4.3.6 Invalid data reception. Any data word(s) associated with a valid receive command that does not meet the criteria specified in 4.4.1.1 and 4.4.1.2 or an error in the data word count shall cause the remote terminal to set the message error bit in the status word to a logic one and suppress the transmission of the status word. If a message error has occurred, then the entire message shall be considered invalid.

In contrast to the valid data reception status response, certain action is required when an invalid data reception occurs. This paragraph in 1553B is an expanded version of the two 1553A paragraphs 4.2.5.4.4 and 4.2.3.5.3.3. This again assumes message formats with associated data word(s), thus mode code commands without a data word are rightly excluded from this group however, in contrast to valid data reception where broadcast message protocol were excluded, here they are included. Therefore, all message formats containing a least one data word (e.g., broadcast data messages, nonbroadcast data messages, broadcast mode codes with a data word, and mode codes with a data word) are included in this requirement. As stated in the requirement, the message command word has been validated and the error occurs in the data word portion of the message. The withholding or suppression of the status response alerts the bus controller error detection electronics to the fact that an incomplete message has occurred and some level of error recovery must occur. The setting of the message error bit in the status that remains in the RT will provide additional information to the error recovery circuitry only if the bus controllers request the status word using the appropriate mode code.

Also notice that the requirement is that the entire received message be considered invalid. This message invalidation requirement may cause some systems like electrical multiplex (EMUX) a problem. Since the EMUX system usually have bit-oriented data rather than word or multiple words (message) oriented data, errors in a word following the reception of good data will invalidate good data. It has been proposed that such a system invalidate all data words from the failure to the end of the message and use previously good data words. This approach, however, has not been allowed. Regardless of the approach, some system mechanisms will store the data and then tag the message as being invalid; others will not allow the user to receive the data. In the first case, it is the responsibility of the user to examine the message valid indication prior to using the data; how-

ever, in the second case, the user must recognize that the data has not been updated.

4.4.4 Bus monitor operation. A terminal operating as a bus monitor shall receive bus traffic and extract selected information. While operating as a bus monitor, the terminal shall not respond to any message except one containing its own unique address if one is assigned. All information obtained while acting as a bus monitor shall be strictly used for off-line applications (e.g., flight test recording, maintenance recording or mission analysis) or to provide the back-up bus controller sufficient information to take over as the bus controller.

A terminal may operate in the bus monitor mode for two reasons: (1) information recording for off-line analysis and (2) information source for backup bus controller. The unique feature of this mode is that it has the ability to decode and accept for data storage any or all messages transmitted on the data bus without the knowledge of or without affecting the operation of multiplex system or the terminal(s) attached to the bus. It also has the option of not being addressable as a terminal attached to the bus. If this is the case, it acts in the "listen capability only" to the system. In this implementation, data cannot be sent to it specifically, but the monitor may collect data by recording message traffic. However, the same terminal may operate in the remote terminal and potential bus controller (backup bus controller) mode as well as having the additional capability to monitor and store all message traffic or an internally derived subset of all messages. It is because of this second capability, its nonpassive nature, that a terminal with the monitor mode is an extremely powerful device in the multiplex system.

4.5 Hardware Characteristics

The following discussion will provide a summary and comparison of MIL-STD-1553A/B requirements that have significant effect on the hardware characteristics (1553, par. 4.5). A detailed comparison of these subparagraphs in 1553A and 1553B is provided in tables II-2 and II-3.

The hardware characteristic section of 1553B examines data bus characteristics (par. 4.5.1) and terminal characteristics (par. 4.5.2). This section is similar to the terminal operation paragraphs (4.3) and the transmission line (4.2.4) of 1553A. Paragraph 4.4 in 1553A (Terminal to Subsystem Interface) has been deleted from 1553B completely. This deletion was consistent with the emphasis on a data bus protocol standard and an electrical multiplex interface requirement treating the terminals inter-facing to the multiplex bus as black box interfaces and not defining any internal interfaces.

4.5.1 Data Bus Characteristics

4.5.1.1 Cable. The cable used for the main bus and all stubs shall be a two conductor, twisted shielded, jacketed cable. The wire-to-wire distributed capacitance shall not exceed 30.0 picofarads per foot. The cables shall be formed with not less than four twists per foot where a twist is defined as a 360 degree rotation of the wire pairs; and, the cable shield shall provide a minimum of 75.0 percent coverage.

4.5.1.2 Characteristic Impedance. The nominal characteristic impedance of the cable (ZO) shall be within the range of 70.0 Ohms to 85.0 Ohms at a sinusoidal frequency of 1.0 megahertz (MHz).

4.5.1.3 Cable attenuation. At the frequency of 4.5.1.2, the cable power loss shall not exceed 1.5 decibels (dB)/100 feet (ft).

Table II-4 contains a summary listing of the data bus and coupling requirements contained in 1553A and 1553B. The characteristics of the twisted shielded pair cable have been relaxed to allow selection of cable types from a variety of manufacturers. It has been shown than minor variations from the specified cable characteristics do not significantly affect the system performance.

A great deal of concern and confusion has resulted because of the cable network requirements, including bus length, coupling, and stubbing. 1553 and 1553A did not provide adequate guidelines for bus network design, especially for the transformer coupled stub. 1553A defined a maximum cable length of 300 ft for the main bus while 1553B chose not to specify a maximum main bus length since it is reasoned that the cable length, number of terminals, and length of stubs are all subject to tradeoff and must be considered in the design for reliable system operation. In other words, an arbitrary limit of 300 ft should not be applied since all parameters of the network must be considered.

4.5.1.4 Cable termination. The two ends of the cable shall be terminated with a resistance, equal to the selected cable nominal characteristic impedance (Z_O) ± 2.0 percent.

4.5.1.5 Cable stub requirements. The cable shall be coupled to the terminal as shown on figure 9 or figure 10. The use of long stubs is discouraged, and the length of a stub should be minimized. However, if installation requirements dictate, stub lengths

Table II-2. Comparison of Data Bus Characteristics

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
1. Twisted, shielded, jacketed	Yes 4.2.4.1	Yes 4.5.1.1
2. Minimum cable shield coverage	80% 4.2.4.1	75% 4.5.1.1
3. Minimum cable twist	1 twist/in (12 twists/ft) 4.2.4.1	4 twists/ft 4.5.1.1
4. Wire-to-wire distributed capacitance (maximum)	30 pF/ft 4.2.4.1	30 pF/ft 4.5.1.1
5. Characteristic impedance of cable	70 ± 10% at 1 MHz 4.2.4.2	Nominal 70 to 85 at 1 MHz 4.5.1.2
6. Cable attenuation	1 dB/100 ft at 1 MHz 4.2.4.3	1.5 dB/100 ft at 1 MHz 4.5.1.3
7. Cable length	300 ft maximum 4.2.4.4	Unspecified —
8. Cable termination using a resistance at both ends	Characteristic impedance 4.2.4.6	Nominal characteristic impedance ± 2% 4.5.1.4
9. Cable stubbing	Transformer coupling for stubs longer than 1 ft but less than 20 ft; direct coupling if stub is less than 1 ft; maximum stub length of 20 ft 4.2.4.5 Figure II-14	Transformer coupling or direct coupling allowed; maximum stub length suggested 20 ft 4.5.1.5.1 or 4.5.1.5.2 Figures 9 or 10 or 1553B
10. Cable Coupling (connector)	Compatible with Amphenol type 31-235 or Trompeter type TEI-14949-E137 receptacles and Amphenol type 31-224 or Trompeter type TEI-14949-PL36 plugs 4.2.4.6	Unspecified —

Table II-2. Comparison of Data Bus Characteristics (continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
11. Cable coupling shielded box	Shielded coupler box 4.2.4.6	75% coverage, minimum 4.5.1.5.1.3 and 4.5.1.5.2.2
12. Coupling transformer turns ratio	Unspecified —	1:1.4 ± 3% higher turns on isolation resistor side of stub 4.5.1.5.1.1
13. Transformer open circuit impedance	Unspecified —	3,000 ohms over frequency of 75 kHz -1 MHz with 1V RMS sine wave 4.5.1.5.1.1.1
14. Transformer waveform integrity	Unspecified —	Droop not to exceed 20% overshoot and ringing less than ± 1V peak under test of figure 11 of 1553B 4.5.1.5.1.1.2
15. Transformer common mode rejection	Unspecified —	45dB at 1MHz 4.5.1.5.1.1.3
16. Fault isolation — Isolation resistor in series with data bus cable (coupler) Direct coupled case with the isolation resistor in the RT	$R = 0.75\ Z_O{*} \pm 5\%$ 4.2.5.2 Figure II-14	$R = 0.75\ Z_O{*} \pm 2\%$ 4.5.1.5.1.2 R = 55 ohms ± 2% 4.5.1.5.2.1 Figure 10 of 1553B
17. Impedance across the data bus for any failure of coupling transformer, cable stub, or terminal receiver and transmitter transformer coupling Direct coupling	No less than $1.5\ Z_O{*}$ 4.2.5.2	No less than $1.5\ Z_O{*}$ 4.5.1.5.1.2 No less than 110 ohms 4.5.1.5.2.3
18. Stub voltage requirements and input level transformer coupling	**Range of the 0.5V to 10V peak; 1.0V to 20V p-p, I-I 4.2.5.4.1 Figure II-14	**Range of 1.0V to 14.0V p-p**, I-I with one fault as stated in 17 above 4.5.1.5.1.4 Figure 9 of 1553B

*Z_O = cable normal characteristic impedance
**Assumes one fault of a coupling transformer, cable stub, or terminal receiver or transmitter

REVIEW AND RATIONALE OF MIL-STD-1553 A AND B

Table II-2. Comparison of Data Bus Characteristics (concluded)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
Direct coupling	**Range of the 0.5V to 10V peak; 1.0V to 20 V p-p, I-I 4.2.5.4.1 Figure II-14	**Range of 1.4V to 20V p-p, I-I with one fault as stated in 17 above 4.5.1.5.2.3 Figure 10 of 1553B
19. Wiring and cabling for electromagnetic capability	MIL-E-6051 4.2.4.7 MIL-STD-1553A	MIL-E-6051 4.5.1.5.3 MIL-STD-1553B

**Assumes one fault of a coupling transformer, cable stub, or terminal receiver or transmitter

Table II-3. Comparison of Terminal Characteristics

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
1. Output level — transformer coupling	± 3.0V to ± 10V peak (6.0V to 20.0V p-p) I-I with no faults: with one fault of a coupling transformer, cable stub, or terminal receiver- transmitter, ± 2.25V to ±11.25 V peak, (4.5V to 15V p-p) I-I 4.2.5.3.1	With RL = 70 ± 2%, 18.0V to 27.0V p-p, I-I 4.5.2.1.1.1 Figure 12 of 1553B
Direct coupling		With RL = 35 ± 2%, 6.0 V to 9.0V p-p, I-I 4.5.2.2.1.1 Figure 12 of 1553B
2. Output waveform — zero crossing deviation	± 25 ns 4.2.5.3.2 Point C, figure II-14 and figure 13 of 1553B	± 25 ns 4.5.2.1.1.2 Figure 12 of 1553B
Rise and fall time (10% to 90%)	100 NS 4.2.5.3.2 Figure 13 of 1553B	100 to 300 ns 4.5.2.1.1.2 Figure 13 of 1553B
Transformer coupling distortion (including overshoot and ringing)	Unspecified	± 90-mV peak, I-I 4.5.2.1.1.2 Point A, figure 12 of 1553B
Direct coupling distortion (including overshoot and ringing)	Unspecified	±300-mV peak, I-I 4.5.2.2.1.2 Point A, figure 12 of 1553B
3. Output noise — Transformer coupling	10-mV p-p, I-I 4.2.5.3.3 Point A, figure II-14	14mV, RMS, I-I 4.5.2.1.1.3 Point A, figure 12 of 1553B

Table II-3. Comparison of Terminal Characteristics (continued)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
Direct coupling		5-mV, RMS, I-I 4.5.2.2.1.3 **Point A, figure 12 of 1553B**
4. Output symmetry (after 2.5 µs of midbit crossing of the last parity bit — Transformer coupling	Unspecified	± 250-mV peak, I-I 4.5.2.1.1.4 **Point A, figure 12 of 1553B**
Direct coupling	Unspecified	± 90-mV peak, I-I 4.5.2.2.1.4 **Point A, figure 12 of 1553B**
5. Input waveform — Maximum zero crossing deviation	Unspecified	± 150 ns 4.5.2.1.2.1 **Point A, figures 9 or 10 of 1553B**
6. Input signal response range Transformer coupling	± 0.5V to ± 10.0V peak (1.0V to 20V p-p), I-I 4.2.5.4.1 **Point C, figure II-14**	0.86V to 14.0V p-p, I-I 4.5.2.1.2.1 **Point A, figure 9 of 1553B**
Direct coupling		1.2V to 20V p-p, I-I 4.5.2.2.2.1 **Point A, figure 10 of 1553B**
7. Input signal no response range Transformer coupling	Unspecified	0.0V to 0.2 V p-p, I-I 4.5.2.1.2.1 **Point A, figure 9 of 1553B**
Direct coupling	Unspecified	0.0V to 0.28V p-p, I-I 4.5.2.2.2.1 **Point A, figure 10 of 1553B**

Table II-3. Comparison of Terminal Characteristics (concluded)

Requirement	Requirement and MIL-STD-1553A paragraph	Requirement and MIL-STD-1553B paragraph
8. Common mode rejection	± 10.0V peak, line-to-ground, dc to 2 MHz 4.2.5.4.2 **Point A, figure II-14**	± 10.0V peak, line-to-ground, dc to 2 MHz 4.5.2.1.2.2.2 or 4.5.2.2.2.2 **Point A, figures 9 or 10 of 1553B**
9. Input Impedance — Transformer coupling	Minimum of 2,000 ohms over a frequency range of 100 kHz to 1 MHz, 1-1 4.2.5.4.3 **Point C, figure II-14**	Minimum of 1,000 ohms over a frequency range of 75 kHz to 1 MHz, I-I 4.5.2.1.2.3 **Point A, figure 9 of 1553B**
Direct coupling		Minimum of 2,000 ohms over a frequency range of 75 kHz to 1 MHz, I-I 4.5.2.2.2.3 **Point A, figure 10 of 1553B**
10. Noise rejection or error rate — Transformer coupling	Maximum bit error rate of 10^{-12} and a maximum incomplete message rate of 10^{-6} in a configuration of one bus controller on a 20-ft stub with a minimum of 100 ft of main bus cable between coupling boxes; test is conducted in presence of magnetic field per MIL-STD-462 method RS02 (spike test) with the limits of MIL-STD-461 RS02 4.3.3	Maximum of one part in 10^7 word error in the presence of additive white gaussian noise of 140-mV RMS over a bandwidth of 1.0 kHz to 4 MHz; input voltage 2.1V p-p, I-I **Point A, figure 9 of 1553B and accept/reject Table II-6** 4.5.2.1.2.4
Direct coupling		Maximum of one part in 10^7 word error rate in the presence of additive white gaussian noise of 200-mV RMS over a bandwidth of 1.0 kHz to 4 MHz; input voltage 3.0V p-p, I-I **Point A, figure 10 of 1553B and accept/reject Table II-6** **4.5.2.2.2.4**

REVIEW AND RATIONALE OF MIL-STD-1553 A AND B

Table II-4. Summary of Data Bus and Coupling Requirements

Parameter	MIL-STD-1553A	MIL-STD-1553B
Transmission line		
• Cable type	Twisted-shielded pair	Twisted-shielded pair
• Capacitance (wire-to-wire)	30 pF/ft, maximum	30 pF/ft, maximum
• Twist	1/in, minimum	4/ft (0.33/in), minimum
• Characteristic impedance (Z_o)	70 ohms ± 10% at 1.0 MHz	70 to 85 ohms at 1.0 MHz
• Attenuation	1.0 dB/100 ft at 1.0 MHz, maximum	1.5 dB/100 ft at 1.0 MHz, maximum
• Length of main bus	300 ft, maximum	Not specified
• Termination	Two ends terminated in resistors equal to Z_o	Two ends terminated in resistors equal to Z_o ± 2%
• Shielding	80% coverage, minimum	75% coverage, minimum
Cable coupling		
• Stub definition	Short stub < 1 ft Long stub > 1 to 20 ft (20 ft, maximum)	Short stub < 1 ft Long stub > 1 to 20 ft (may be exceeded)
• Coupler requirement	All connections use external coupler box; connectors specified (ref. **fig. 1-1.6)**	Direct coupled, short stub transformer coupled, long stub (ref. **fig. 1-1.7)**
• Coupler transformer		
• Turns ratio	Not specified	1 to 1.41
• Input impedance	Not Specified	3,000 ohms, minimum (75.0 kHz to 1.0 MHz)
• Droop	Not specified	20% maximum (250 kHz)
• Overshoot and ringing	Not specified	± 1.0V peak (250-kHz square wave with 100-ns maximum rise and fall time)
• Common mode rejection	Not specified	45.0 dB at 1.0 MHz
• Fault protection	Resistor in series with each connection equal to $(0.75Z_o)$ ± 5% ohms	Resistor in series with each connection equal to $(0.75Z_o)$ ± 2.0% ohms
• Stub voltage	± 0.5V to ± 10.0V, peak, I-I (1.0V to 20.0V p-p, I-I); nominal signal level for terminal response	1.0V to 14.0V p-p, I-I, minimum signal voltage (transformer coupled) 1.4V to 20.0V, p-p, I-I, minimum signal voltage (direct coupled)

exceeding those lengths specified in 4.5.1.5.1 and 4.5.1.5.2 are permissible.

4.5.1.5.1 Transformer coupled stubs. The length of a transformer coupled stub should not exceed 20 fees. If a transformer coupled stub is used, then the following shall apply.

4.5.1.5.1.1 Coupling transformer. A coupling transformer, as shown on figure 9, shall be required. This transformer shall have a turns ratio of 1:1.41 ± 3.0 percent, with the higher turns on the isolation resistor side of the stub.

A generalized multiplex bus network configuration is shown in figure 1 of 1553B. The main bus is terminated at each end in the cable characteristic impedance to minimize reflections caused by transmission line mismatch. With no stubs attached, the main bus looks like an infinite length transmission line and therefore there are no disturbing reflections. When the stubs are added for connection of the terminals, the bus is loaded locally and a mismatch occurs with resulting reflections. The degree of mismatch and signal distortions caused by reflections are a function of the impedance (Z) presented by the stub and terminal input impedance. In order to minimize signal distortion, it is desirable that the stub maintain a high impedance. This impedance is reflected back to the main bus. At the same time the impedance needs to be kept low so that adequate signal power will be delivered to the receiver input. Therefore, a tradeoff and compromise between these conflicting requirements is necessary to achieve the specified signal-to-noise ratio and system error rate performance. Two methods for coupling a terminal to the main bus are defined in 1553B, transformer coupling and direct coupling. The two methods are shown in figures 9 and 10 of 1553B. Transformer coupling is usually used with long stubs (1 to 20 ft) and requires a coupler box, separate from the terminal, located near the junction of the main bus and stub. Direct coupling is usually limited for use with stubs of less than 1 ft.

Fault isolation resistors (R) are included to provide protection for the main bus in case of a short circuit in the stub or terminal. The coupler-transformer characteristics defined in 1553B and listed in table 7-4 provide a compromise between the signal level and distortion characteristics delivered to the terminals. The coupler transformer turns ratio (1:1.41) provides impedance transformation for both l/ reception and transmission. The improvement of stub load impedance is a result of impedance transformation, which is proportional to the square of the turns ratio,

assuming an ideal coupler transformer.

As indicated above, the 1:1.41 transformer also provides ideal termination of the stub for transmission of signals from the terminal to the main bus. The impedance at main bus is

$$Z_B = \frac{Z_O}{2} + 2R \qquad (1)$$

where

$$R = 0.75\, Z_O$$

$$Z_B = 0.5 Z_O + 1.5 Z_O = 2 Z_O \text{ ohms} \qquad (2)$$

The reflected impedance, Z_R, from the bus to the stub due to the transformer impedance transformation is

$$Z_R = \frac{Z_B}{2} = \frac{2 Z_O}{2} = Z_O \qquad (3)$$

Therefore, the coupler transformer specified in 1553B provides the characteristics desired for reducing reflections and maintaining signal levels for systems where long stubs are required.

4.5.1.5.1.1.1 Transformer input impedance. The open circuit impedance as seen at point B on figure 11 shall be greater than 3000 ohms over the frequency range of 75.0 kilohertz (kHz) to 1.0 megahertz (MHz), when measured with a 1.0V root-mean-square (RMS) sine wave.

The transformer open circuit impedance (Zoc) is required to be greater than 3k ohms in 1553B systems. The measurement is made looking into the higher turns winding (1.41) with a 75 kHz to 1 MHz sine wave signal. The test amplitude at the transformer winding is adjusted to 1 V rms. The critical factors in achieving the 3k ohm Zoc is the distributed capacitance of the windings and the transformer primary inductance. The inductance of the transformer must be large enough to provide the open circuit impedance at 75 kHz and the distributed capacitance should be small enough to maintain the open circuit impedance at the 1 MHz test frequency. The inductance may obviously be increased by increasing the number of turns on the transformer. This technique, however, tends to increase the distributed capacitance, degrading high frequency performance and therefore causing waveform integrity and common mode rejection to suffer.

4.5.1.5.1.1.2 Transformer waveform integrity. The droop of the transformer using the test configuration shown on figure 11 at point B, shall not exceed 20.0 percent. Overshoot and ringing as measured at point
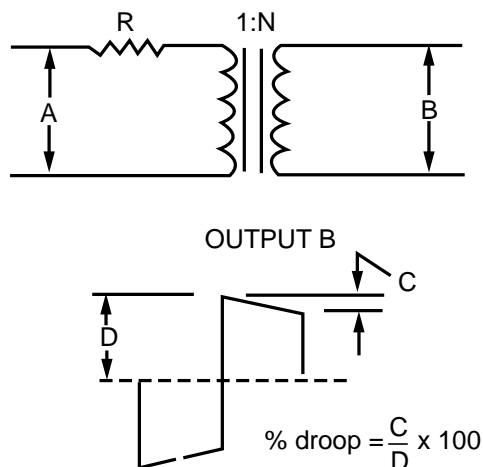
Figure 11 of 1553B. Coupling Transformer

$$\% \text{ droop} = \frac{C}{D} \times 100$$

B shall be less than ±1.0V peak. For this test, R shall equal 360.0 ohms + 5.0 percent and the input A of figure 11 shall be a 250.0 kHz square wave, 27.W peak-to-peak, with a rise and fall time no greater than 100 nanoseconds (ns).

The ability of the coupler transformer to provide a satisfactory signal is specified in the droop, overshoot, and ringing requirements of 1553B shown in figure II-11. Droop is specified at 20% maximum when driving the transformer with a 250 kHz, 27V pep square wave. The test for the droop characteristic is made by driving the low turns winding through a 360 ohm resistor and measuring the signal at the open-circuited high side winding. The droop of the transformer is determined mainly by the primary inductance. Since the primary inductance a/so provides the 3k ohm open circuit impedance, the inductance should be made as high as possible without degrading the high-frequency performance of the transformer. Ringing and overshoot on the transformer signal is also shown in figures 11-11. The ~ 1 V limit on these high-frequency perturbations can be achieved through careful attention to leakage inductance and transformer capacitance.

4.5.1.5.1.1.3 Transformer common mode rejection. The coupling transformer shall have a common mode rejection ratio greater than 45.0 dB at 1.0 MHz.

The command mode rejection of the isolation transformer is required to be greater than 45.0 dB. The common mode test shown in figure 11-12 consists of driving the high turns winding while measuring the differential signal across the high side. Common mode rejection can be improved by minimizing the interwinding capacitance and the core-to-winding capacitance.

4.5.1.5.1.2 Fault isolation. An isolation resistor shall be placed in series with each connection to the data bus cable. This resistor shall have a value of 0.75 Z_O plus or minus 2.0 percent, where Z_O is the selected cable nominal characteristic impedance. The impedance placed across the data bus cable shall be no less than 1.5 Z_O failure of the coupling transformer, cable stub, or terminal transmitter/receiver.

4.5.1.5.1.3 Cable coupling. All coupling transformers and isolation resistors, as specified in 4.5.1.5.1.1 and 4.5.1.4.1.2, shall have continuous shielding which will provide a minimum of 75 percent coverage. The isolation resistors and coupling transformers shall be placed at minimum possible distance from the junction of the stub to the main bus.

4.5.1.5.1.4 Stub voltage requirements. Every data bus shall be designed such that all stubs at point A of figure 9 shall have a peak-to-peak amplitude, line-to-line within the range of 1.0 and 14.0 V for a transmission by any terminal on the data bus. This shall include the maximum reduction of data bus signal amplitude in the event that one of the terminals has a fault which causes it to reflect a fault impedance specified in 4.5.1.5.1.2 on the data bus. This shall also include the worst case output voltage of the terminals as specified in 4.5.2.1.1.1 and 4.5.2.2.1.1.

4.5.1.5.2 Direct coupled stubs. The length of a direct coupled stub should not exceed 1 foot. Refer to 10.5 for comments concerning direct coupled

$$\% \text{ droop} = \frac{E_D}{E_P} \times 100$$
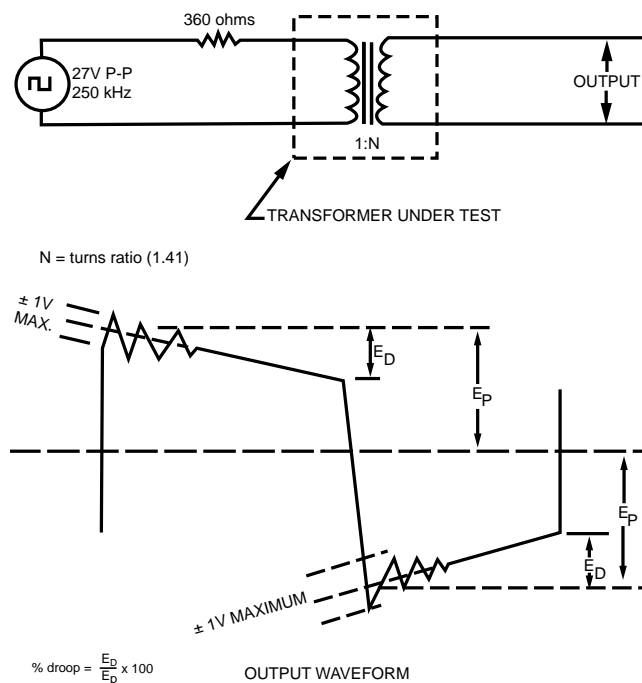
Figure II-11. Waveform Test
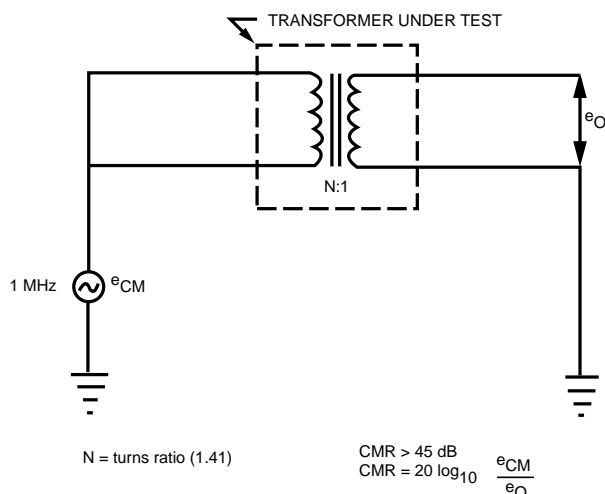
Figure II-12. Common Mode Test

stubs. If a direct coupled stub is used, then the following shall apply.

4.5.1.5.2.1 Fault isolation. An isolation resistor shall be placed in series with each connection to the data bus cable. This resistor shall have a value of 55.0 ohms plus or minus 2.0 percent. The isolation resistors shall be placed within the RT as shown on figure 10.

4.5.1.5.2.2 Cable coupling. All bus-stub junctions shall have continuous shielding which will provide a minimum of 75 percent coverage.

4.5.1.5.2.3 Stub voltage requirements. Every data bus shall be designed such that all stubs at point A of figure 10 shall have a peak-to-peak amplitude, line-to-line within the range of 1.4 and 20.0 V for a transmission by any terminal on the data bus. This shall include the maximum reduction of data bus signal amplitude in the event that one of the terminals has a fault which causes it to reflect a fault impedance of 110 Ohms on the data bus. This shall also include the worst case output voltage of the terminals as specified in 4.5.2.1.1.1 and 4.5.2.2.1.1.

4.5.1.5.3 Wiring and cabling for EMC. For purposes of electromagnetic capability (EMC), the wiring and cabling provisions of MIL-E-6051 shall apply.

The requirements for couplers specified in 1553A have been modified for 1553B and a comparison of the two requirements is shown in figure II-13. The major difference in the two requirements is the placement of the isolation resistors for the direct-coupled (short-stub) connection and the characterization of the coupling transformer in the long-stub (transformer-coupled) connection. With the isolation resistors located in the termi-

nal for the direct-coupled case, the need for a separate coupler box is eliminated as long as a reliable shielded splice can be maintained.

The terminal input and output specifications for the transformer-coupled and direct-coupled connections are required to be separated in 1553B because of the effects on signal levels and impedances of the transformer turns ratio being specified as 1:1.41 instead of the assumed 1:1 in 1553A.

The transformer in the 1553B coupler has the turns ratio of 1:1.41. This ratio, together with the 0.75Z fault isolation resistor provides the correct characteristic impedance for terminating the stub:

$$Z \text{ stub} = \left(\frac{1}{1.41}\right)^2 (.75 \, Z_O + .75 \, Z_O + .5 \, Z_O)$$

The stub capacitance is also effectively decreased by the square of the turns ratio to lessen the loading problem. The 1:1.41 ratio of 1553B is a compromise between stub matching and decreased stub loading.

The connector type specified is important for severe environment military aircraft applications. MIL-STD-1553A specified the use of two-pin polarized connectors such as TEI BJ37 (reference to "TEL-14949-E137" is in error). The two-pin polarized connector employs an interface configuration with one male and one female contact. The female contact is embedded in one side of a step dielectric and the male contact is exposed. There are several shortcomings inherent in this design. MIL-STD1553B does not specify connector types.

The coupling network provides bus connections for the transformer-coupled (external coupler) and direct-coupled cases defined in 1553B. Isolation resistors of 55 ohms value are included for the direct-coupled connection, and the proper transformer turns ratio is provided when the appropriate bus connection is selected. The turns ratio is different for the transformer-coupled and direct-coupled connections to compensate for the 1.41 to 1 reduction of signal level in the external coupler. This feature allows a threshold setting that is the same for both bus connections.

4.5.2 Terminal Characteristics

An additional concern is the specification for the bus and terminal interface. This area of 1553A was significantly reworked to provide a more complete definition of the terminal interface characteristics

Figure II-13. Coupler Characteristics
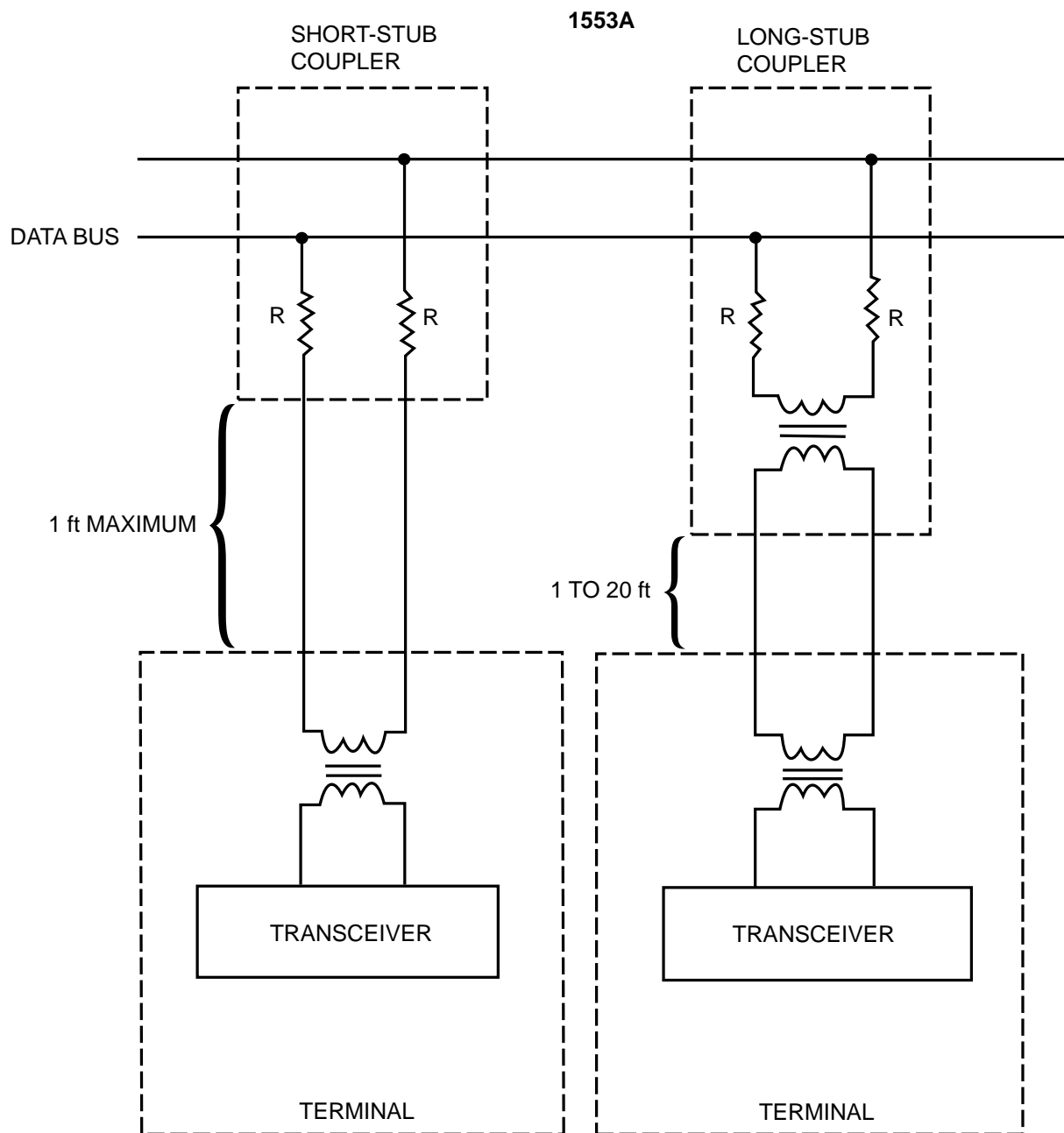
- Isolation resistors: R= 0.75 Z^*_O ± 5%
- Isolation transformer: (not specified)
  *Nominal characteristic impedance of bus cable: $Z_O = 70 ± 10\%$ at 1 MHz

- Isolation resistors: $R = 0.75 \, Z^*_O \pm 2\%$
- Isolation transformer: turns ratio $1{:}1.41 \pm 3\%$

(1—terminal winding)
(1.41—bus winding)
$Z_{OC} > 3K$ at 75 kHz to 1 MHz
1V rms sine wave

Droop: < 20%
Overshoot/ringing: < ± 1V } at 27V P-P 250 kHz square wave
CMR: > 45 dB at 1MHz

*Nominal characteristic impedance of bus cable: $Z_O = 70$ to 85 at 1MHz

Figure II-13. Coupler Characteristics (continued)

that are independent of network configuration. Figures II-14 and II-15 show the interface diagrams and the points where the signal measurement are defined in 1553A and 1553B. Table II-5 is a summary listing of the terminal and data bus interface requirements specified in the two versions of the standard. The following discussion will relate some of the rationale for this approach to development of the updated requirements in 1553B.
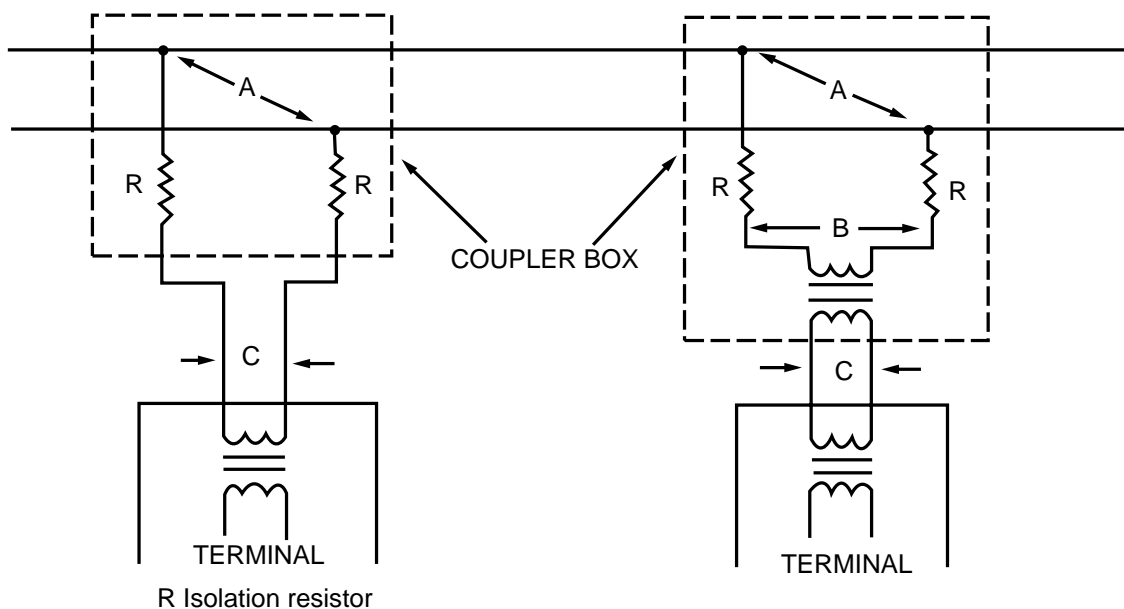
Figure II-14. MIL-STD-1553A Data Bus Interface
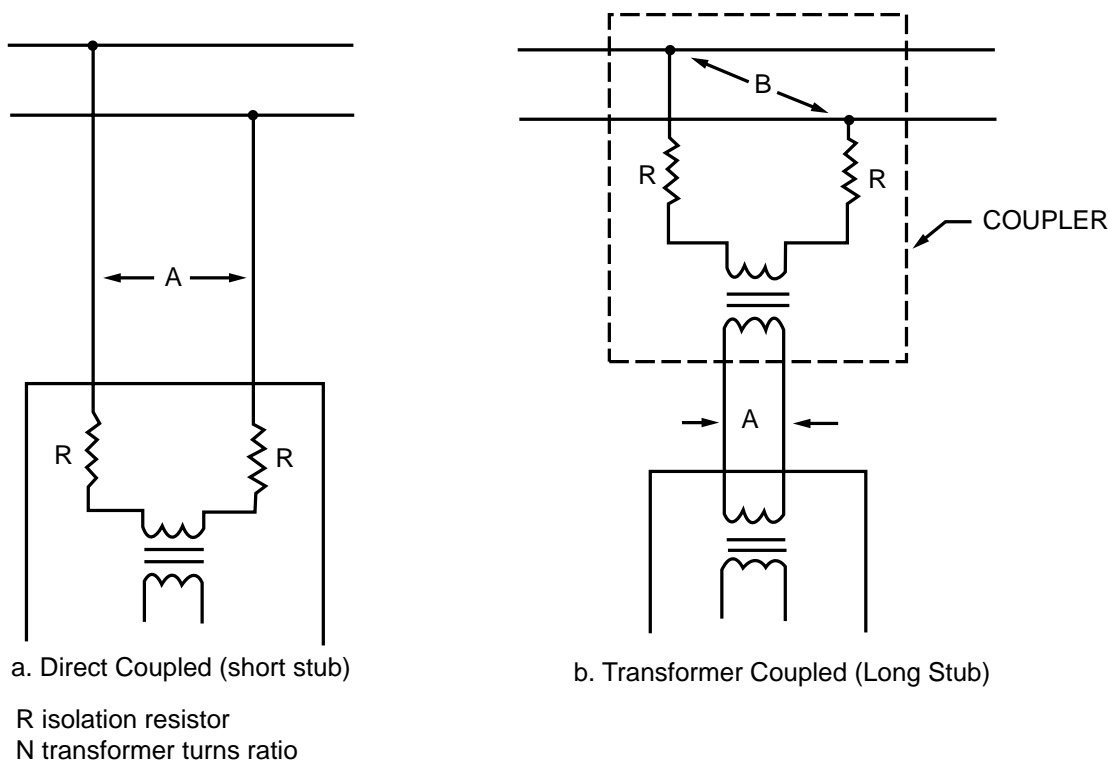
Figure II-15. MIL-STD-1553B Data Bus Interface

Table II-5. Summary of Terminal and Data Bus Interface Requirements

Parameter	MIL-STD-1553A	MIL STD-1553B
Terminal output characteristics		
• Output voltage	±3.0V to ±10.0V, peak, I-I (6.0 V to 20.0V, p-p, I-I) Point A, figure II-14.	18.0V to 27.0V, p-p, I-I (transformer coupled) Point A, figure II-16b
		6.0V to 9.0V, p-p, I-I (direct coupled) Point A, figure II-16a
• Output waveform	Zero crossing deviation ± 25ns; rise and fall time of this waveform shall be 100 ns	Zero crossing deviation ± 25ns, maximum, measured with respect to previous crossing; rise and fall times 100 to 300 ns
		Overshoot and ringing ± 900 mV peak; maximum, I-I Point A, figure II-16b (transformer-coupled stub)
		Zero crossing deviation and rise and fall times same as above overshoot and ringing — ± 300-mV peak, maximum, I-I Point A, figure II-16a (direct-coupled stub)
• Output symmetry	Not specified	Voltage at 2.5 µs after midpoint of parity bit; ± 250 mV peak, maximum, I-I Point A, figure II-16b (transformer-coupled stub)
		Voltage at 2.5 µs after midpoint of parity bit; ± 90-mV peak, maximum, I-I Point A, figure II-16a (direct-coupled stub)
• Output Noise	10 mV p-p, I-I Point A, figure II-14	14.0 mV, RMS, I-I Point A, figure II-16b (transformer-coupled)
		5.0mV, RMS, I-I Point A, figure II-16a, (direct-coupled)
Terminal input characteristics		
• Input Voltage	± 0.5 to ± 10.0V peak, I-I (1.0 V to 20.0V p-p, I-I) Point A, figure II-14 — terminal responds	0.86V to 14.0V, p-p, I-I, terminal response required; 0.0V to 0.2V, p-p, I-I, terminal no response (with transformer coupled stubs) Point A, figure II-15b
		1.2V to 20.0V, p-p, I-I, terminal response required; 0.0V to 0.28V, p-p, I-I, terminal no response (with direct-coupled stubs) Point A, figure II-15a

Table II-5. Summary of Terminal and Data Bus Interface Requirements (continued)

Parameter	MIL-STD-1553A	MIL-STD-1553B
• Input impedance	2,000 ohms, minimum, from 100 kHz to 1.0 MHz Point C, figure II-14	1,000 ohms, minimum, from 75 kHz to 1.0 MHz Point A, figure II-15b (transformer-coupled stub)
		2,000 ohms, minimum, from 75 kHz to 1.0 MHz Point A, figure II-15a (direct-coupled)
• Noise rejection	BER — 1 in 10^{12}; maximum, incomplete message rate 1 in 10^6; test condition — bus controller connected to RT over 100-ft data bus using 20-ft stubs	Maximum word error rate of 1 in 10^7, AWG noise 1.0 kHz to 4.0 MHz, 140 mV, RMS level; signal level — 2.1V, p-p, I-I Point A, figure II-15b (transformer-coupled stub)
		Maximum word error rate of 1 in 10^7, AWG noise 1.0 kHz to 4.0 MHz, 200 mV, RMS level; signal level is 3.0V, p-p, I-I Point A, figure II-15a (direct-coupled stub)
• Common mode rejection	± 10.0V peak, line-to-ground, dc to 2 MHz shall not degrade performance Point A, figure II-14	± 10.0V peak, line-to-ground, dc to 2.0 MHz, shall not degrade performance of the receiver Point A, figure II-15b (transformer-coupled stub)
		Same specification for the direct-coupled stub Point a, figure II-15a
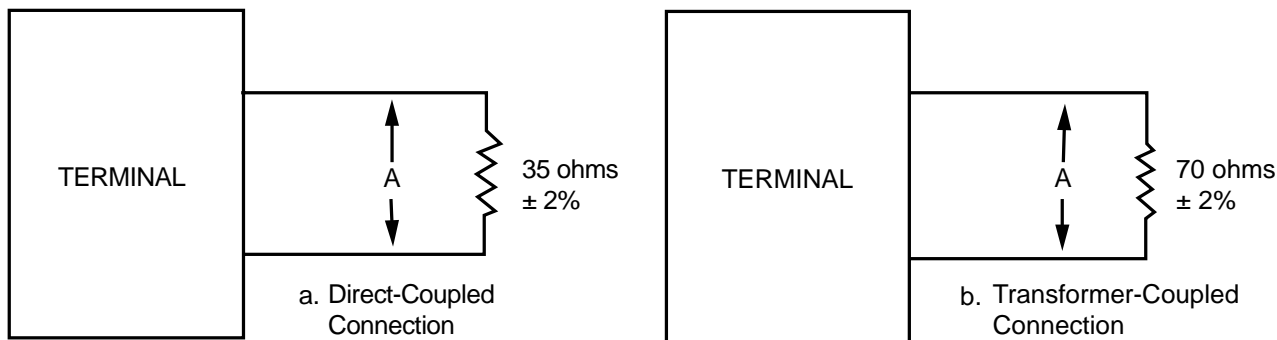
Table II-16. Direct-Coupled and Transformer-Coupled Terminal Output Test Configuration

4.5.2.1 Terminals with transformer coupled stubs

4.5.2.1.1 Terminal output characteristics. The following characteristics shall be measured with R_L, as shown on figure 12, equal to 70.0 Ohms ± 2.0 percent.

4.5.2.1.1.1 Output levels. The terminal output voltage levels shall be measured using the test configuration shown on figure 12. The terminal output voltage shall be within the range of 18.0 to 27.0 V, peak-to-peak, line-to-line, when measured at point A on figure 12.

The upper end of the bus voltage range (20V p-p) allowed by 1553A was considered to be excessive and if implemented would result in excessive power dissipation and most of the systems and hardware designed to 1553A use signal levels at or near the lower end (6.0V p-p) of the specified range. It should be noted that the measurement point in 1553A is at the main bus, point A on figure 7-14. This does not provide a specified level at the terminal connection point (c) and is especially troublesome to the terminal hardware designer since the characteristics of the coupler transformer are not specified. The approach taken for 1553B is to specify the terminal output for the two conditions, transformer-coupled and direct-coupled. This may require that each terminal have two sets of input-output pins for each bus cable connection. Therefore the 18V to 27V p-p transmitter output applied to the stub and coupler results in a nominal 6.0V to 9.0V p-p signal level at the stub and bus connection (point B). This range is equivalent to that specified for the direct-coupled case shown in figure II-15. Test configurations are provided for both direct coupled and transformer-coupled in figure II-16.

4.5.2.1.1.2 Output waveform. The waveform, when measured at point A on figure 12 shall have zero crossing deviations which are equal to, or less than, 25.0 ns from the ideal crossing point, measured

with respect to the previous zero crossing (i.e., .5 ± .025 μs, 1.0 ± .025 μs, 1.5 ± .025 μs, and 2.0 ± .025 μs). The rise and fall time of this waveform shall be from 100.0 to 300.0 ns when measured from levels of 10 to 90 percent of full waveform peak-to-peak, line-to-line, voltage as shown on figure 13. Any distortion of the waveform including overshoot and ringing shall not exceed ± 900.0 millivolts (mV) peak, line-to-line, as measured at point A, figure 12.

The transmitted waveform specified in 1553A is limited in the definition of signals that appear on the data bus. The zero crossing deviations allowed are not well defined for all possible patterns, and the rise and fall times specification is open ended. The waveform characteristics defined in 1553B provide control of the zero crossing deviations for all possible conditions and establishes a limit on distortion.

4.5.2.1.1.3 Output noise. Any noise transmitted when the terminal is receiving or has power removed, shall not exceed a value of 14.0 mV, RMS, line-to-line, as measured at point A, figure 12.
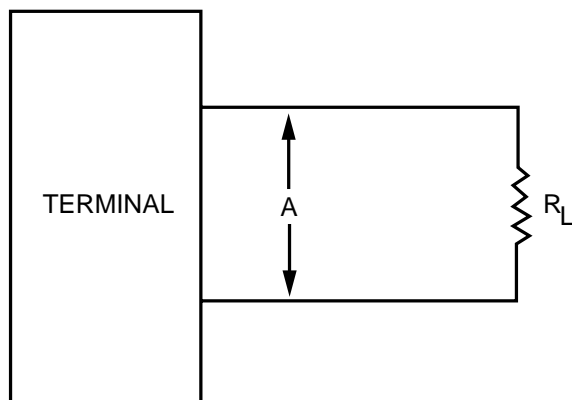
Figure 12 of 1553B. Terminal I/O Characteristics for Transformer-Coupled and Direct-Coupled Stubs
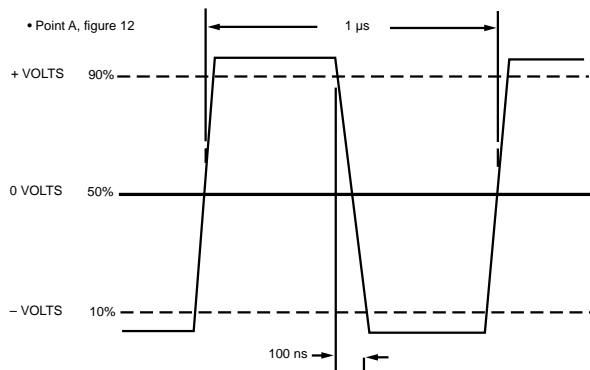
Figure 13 of 1553B. Output Waveform

MIL-STD-1553A specified value of 10 mV pep noise (4.25.3.3) is considered unrealistically low for practical hardware design. Also, noise is normally specified as an rms value since peak noise is difficult to measure. The output rms noise for the transformer-coupled and direct coupled cases are specified in 1553B (4.5.2.1.1.3 and 4.5.2.2.1.3) and are consistent with the required system performance and practical terminal hardware design. The requirement for low output noise of 14 mV rms and 5 mV rms when not transmitting also places significant constraints on the length and routing of input-output wiring because of the induced power supply and logic noise generated in the terminal.

4.5.2.1.1.4 Output symmetry. From the time beginning 2.5 µs after the mid-bit crossing of the parity bit of the last word transmitted by a terminal, the maximum voltage at point A of figure 12 shall be no greater than ± 250.0 mV peak, line-to-line. This shall be tested with the terminal transmitting the maximum number of words it is designed to transmit, up to 33. This test shall be run six times with each word in a contiguous block of words having the same bit pattern. The six word contents that shall be used are 8000_{16} $7FFF_{16}$, 0000_{16}, $FFFF_{16}$, 5555_{16} , and $AAAA_{16}$.

The output of the terminal shall be as specified in 4.5.2.1.1.1 and 4.5.2.1.1.2.

Symmetry of the transmitted waveform in time and amplitude was not adequately specified in 1553A. An ideal waveform is perfectly balanced so that the signal energy on both sides of zero (off) level is identical. If the positive or negative energy is not equal, problems can develop in the coupling transformers and the transmission line can acquire a charge that appears as a tail with overshoot and ringing when transmission is terminated. These considerations require that the symmetry of the transmitted waveform be controlled

within practical limits. This is accomplished in 1553B by specifying the signal level from a time beginning 2.5 µs after the midbit zero crossing of the parity bit of the last word in a message transmitted by the terminal under test The test messages contain the maximum number of words and defined bit patterns.

4.5.2.1.2 Terminal input characteristics. The following characteristics shall be measured independently.

4.5.2.1.2.1 Input waveform compatibility. The terminal shall be capable of receiving and operating with the incoming signals specified herein, and shall accept waveform varying from a square wave to a sine wave with respect to the previous zero crossing of ±150 ns, (i.e., 2.0 ± .15 µs, 1.5 ± .15 µs, 1.0 ± .15 µs, .5 ± .15 µs). The terminal shall respond to an input signal whose peak-to-peak amplitude, line-to-line, is within the range of .86 to 14.0 V. The terminal shall not respond to an input signal whose peak-to-peak amplitude, line-to-line, is within the range of 0.0 to .20V. The voltages are measured at point A on figure 9.

4.5.2.1.2.2 Common mode rejections. Any signals from direct current (DC) to 2.0 MHz, with amplitudes equal to or less than ± 10.0 V peak, line-to-ground, measured at point A on figure 9, shall not degrade the performance of the receiver.

The input voltage specifications in 1553B have been revised to reflect the output voltage ranges for the transformer-coupled and direct-coupled connections to the terminal. The terminal required response and no-response signal levels are specified so that the optimum threshold levels may be selected. It should be noted that the threshold setting has a significant effect on the noise rejection and error rate performance of the receiver. The maximum value for no-response signal level is 200 mV p-p, and 280 mV p-p allows optimum threshold settings of ± 125 and ± 175 mV, respectively, for minimum bit error rate (BER) performance when subjected to the specified noise rejection test conditions.

4.5.2.1.2.3 Input impedance. The magnitude of the terminal input impedance, when the RT is not transmitting, or has power removed, shall be a minimum of 1000.0 Ohms within the frequency range of 75.0 kHz to 1.0 MHz. This impedance is that measured line-to-line at point A on figure 9.

As indicated in the data bus network requirement, input impedance is required to be maintained at a reasonable level to reduce the signal distortion

effects when terminals are connected to the bus. Terminal input impedance is determined primarily by the following:

a. Transformer impedance maintains inductance required to support low-frequency component of signal while controlling interwinding capacitance for high frequencies.

b. Terminal wiring capacitance controls stray capacitance of wiring from terminal connector to receiver.

c. Secondary impedance transformation, for the transformer-coupled case, a transformer with a turns ratio of 1:1.41 is implied. The impedance at the secondary is reflected to the terminal input reduced by a factor of 2

The transformer is a very important element in determining the transceiver characteristics such as input impedance, signal waveform integrity, and common mode rejection required by 1553B. The considerations for transformer and associated input-output circuit design are to:

a. Provide the specified input impedance at high frequencies (terminal input impedance 1,000 ohms and 2,000 ohms at 1 MHz)

b. Design for low interwinding capacitance to achieve the common mode rejection (45 dB CMR at ± 10V peak, dc to 2.0 MHz)

These considerations are directly applicable to the design of the transceiver transformer. In addition to the transformer characteristics other considerations for maintaining the terminal minimum input impedance specified in 1553B are as follows:

a. Minimize stray capacitance of wiring from the external connector and on the circuit card to the buffer amplifier (every 100 pF results in approximately 1,600 ohms of shunt impedance).

b. Maintain high impedance at the receiver limiter and filter circuit inputs and transmitter driver outputs in the "off" state. These impedances must be maintained with the terminal (transceiver) power off.

The factor of 2 difference in the impedance specified for the transformer-coupled and direct-coupled cases is based primarily on the effect of c. The frequency range was changed to reduce the lower frequency limit from 100 kHz (1553A) to 75 kHz (1553B). This provides additional assurance that adequate transformer volt-time product (inductance) is available to support the lower frequencies of the signal without approaching saturation.

4.5.2.1.2.4 Noise rejection. The terminal shall exhibit a maximum word error rate of one part in 10^7, on all words received by the terminal, after validation checks as specified in 4.4, when operating in the presence of additive white Gaussian noise distributed over a bandwidth of 1.0 kHz to 4.0 MHz at an RMS amplitude of 140 mv. A word error shall include any fault which causes the message error bit to be set in the terminal's status word, or one which causes a terminal to not respond to a valid command. The word error rate shall be measured with a 2.1 V peak-to-peak, line-to-line, input to the terminal as measured at point A on figure 9. The noise tests shall be run continuously until, for a particular number of failures, the number of words received by the terminal, including both command and data words, exceeds the required number for acceptance of the terminal or is less than the required number for rejection of the terminal as specified in table 11. All data words used in the tests shall contain random bit patterns. These bit patterns shall be unique to each data word in a message and shall change randomly from message to message.

The noise rejection specification and test conditions defined in 1553A requires extensive system-type evaluation testing of the terminal employing a bus controller and data bus radiated with certain of the EMI fields specified in MIL-STD-461 and 462 Extensive test time is required to verify a BER of 10^{-12} and the test must be performed in a screen room.

The test conditions of signal and noise specified in 1553B were selected to produce a corresponding value of word error rate (WER) which is sufficiently high (10^{-7}) to permit performance verification of a terminal receiver within a reasonable test period. The noise rejection is a figure-of-merit test and can be performed in a normal laboratory environment using actual transmitters (Manchester waveform output) with a typical test setup as shown in figure II-17. The verification of detector performance should consider the measurement of both detected and undetected errors. To measure undetected errors that do not correlate with the transmitted signal and are not detected by the terminal under test, it is necessary to compare the transmitted and received data. Therefore, a reference of transmitted data is provided to the comparator for comparison with the deleted data from the terminal under test.

Externally generated noise on board an aircraft can take many forms with a wide variety of power and fre-

quencies. It is recognized that impulse noise having either random or periodic impulse duration, frequency of occurrences, and burst interval are more typical of noise sources that have major impact on aircraft digital data systems. Relay switching is generally regarded as the most severe source of impulse noise on a typical aircraft. This type of noise defies accepted forms of analysis, such as that per-

Table 11-6. Criteria for Acceptance or Rejection of a Terminal for the Noise Rejection Test

Total words received by terminal
(in multiples of 10^7)

No. of errors	Reject (equal or less)	Accept (equal or more)
0	NA	4.40
1	NA	5.21
2	NA	6.02
3	NA	6.83
4	NA	7.64
5	NA	8.45
6	0.45	9.27
7	1.26	10.08
8	2.07	10.89
9	2.88	11.70
10	3.69	12.51
11	4.50	13.32
12	5.31	14.13
13	6.12	14.94
14	6.93	15.75
15	7.74	16.56
16	8.55	17.37
17	9.37	18.19
18	10.18	19.00
19	10.99	19.81
20	11.80	20.62
21	12.61	21.43
22	13.42	22.24
23	14.23	23.05
24	15.04	23.86
25	15.85	24.67
26	16.66	25.48
27	17.47	26.29
28	18.29	27.11
29	19.10	27.92
30	19.90	28.73
31	20.72	29.54
32	21.43	30.35
33	22.34	31.16
34	23.15	31.97
35	23.96	32.78
36	24.77	33.00
37	25.58	33.00
38	26.39	33.00
39	27.21	33.00
40	28.02	33.00
41	33.00	NA

Note: NA — not applicable

formed utilizing additive white Gaussian (AWG) noise model. Because of the difficulty of error performance analysis using the impulsive noise model, a worst-case Gaussian model has been formulated. This model offers an analysis and test tool for evaluation of terminal receiver performance considering the effects of impulsive noise. This approach is reflected in the noise rejection test conditions and word error rate versus signal-to-noise ratio (SNR) performance requirements of 1553B, paragraphs 4.5.2.1.2.4 and 4.5.2.2.2.4.

4.5.2.2 Terminals with direct coupled stubs

4.5.2.2.1 Terminal output characteristics. The following characteristics shall be measured with RL, as shown on figure 12, equal to 35.0 Ohms ± 2.0 percent.

4.5.2.2.1.1 Output levels. The terminal output voltage levels shall be measured using the test configuration shown on figure 12. The terminal output voltage shall be within the range of 6.0 to 9.0 V, peak-to-peak, line-to-line, when measured at point A on figure 12.

4.5.2.2.1.2 Output waveform. The waveform, when measured at point A on figure 12, shall have zero crossing deviations which are equal to, or less than, 25.0 ns from the ideal crossing point, measured with respect to the previous zero crossing (i.e., .5 ±.025 µs, 1.0 ±.025 µs, 1.5 ±.025 µs and 2.0 ±.025 µs). The rise and fall time of this waveform shall be from 100.0 to 300.0 ns when measured from levels of 10 to 90 percent of full waveform peak-to-peak, line-to-line, voltage as shown on figure 13. Any distortion of the waveform including overshoot and ringing shall not exceed ± 300.0 mV peak, line-to-line, as measured at point A on figure 12.

4.5.2.2.1.3 Output noise. Any noise transmitted when the terminal is receiving or has power removed, shall not exceed a value of 5.0 mV, RMS, line-to-line, as measured at point A on figure 12.

4.5.2.2.1.4 Output symmetry. From the time beginning 2.5 µs after the mid-bit crossing of the parity bit of the last word transmitted by a terminal, the maximum voltage at point A on figure 12, shall be no greater than ±90.0 mV peak, line-to-line. This shall be tested with the terminal transmitting the maximum number of words it is designed to transmit, up to 33. This test shall be run six times with each word in a contiguous block of words having the same bit pattern. The six word contents that shall be used are 8000_{16} $7FFF_{16}$, 0000.0_{16}, $FFFF_{16}$, 5555_{16}, and
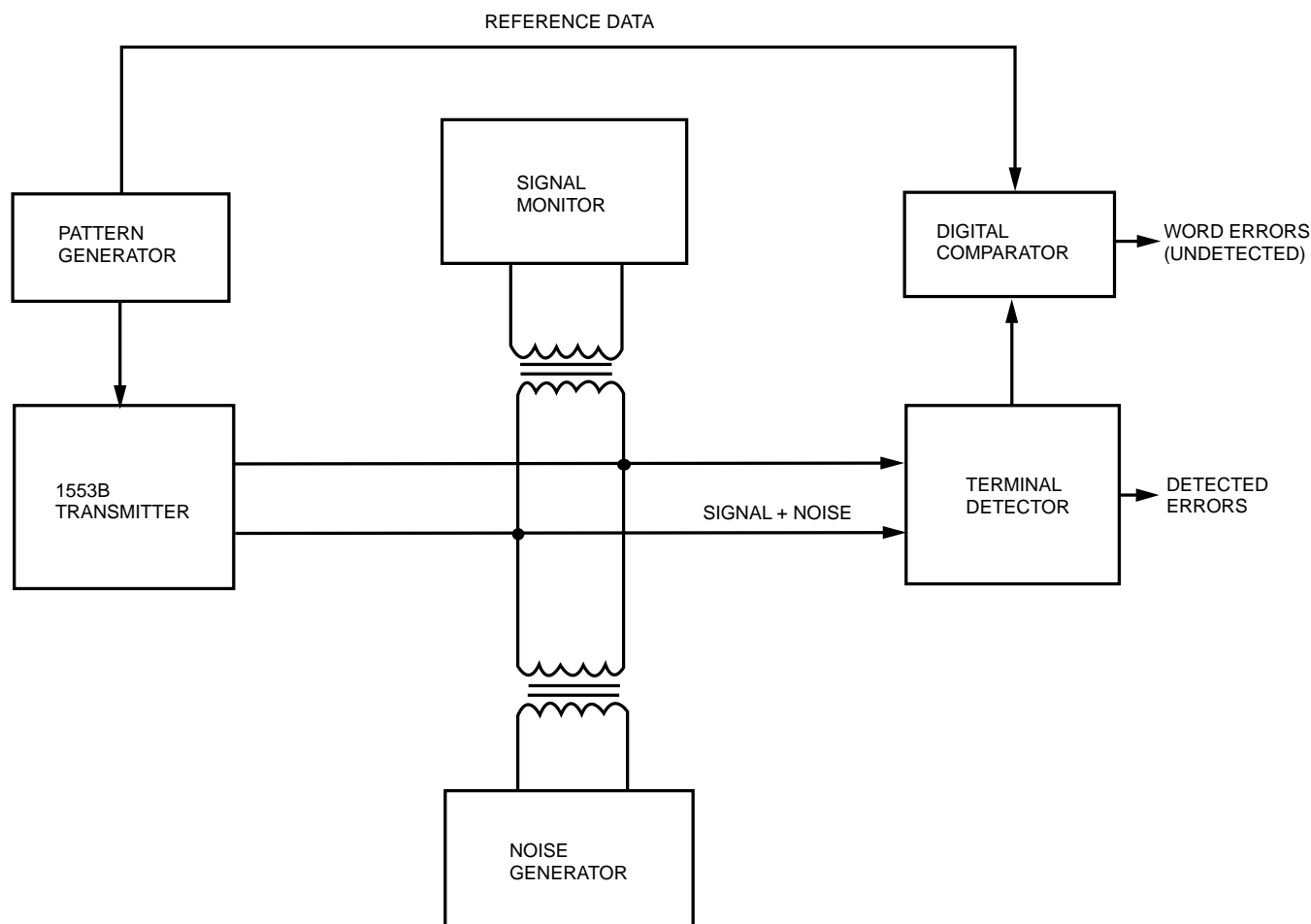
Figure II-17. Typical Noise Rejection Test Setup

AAAA$_{16}$. The output of the terminal shall be as specified in 4.5.2.2.1.1 and 4.5.2.2.1.2.

4.5.2.2.2 Terminal input characteristics. The following characteristics shall be measured independently.

4.5.2.2.2.1 Input waveform compatibility. The terminal shall be capable of receiving and operating with the incoming signals specified herein, and shall accept waveform varying from a square wave to a sine wave with a maximum zero crossing deviation from the ideal with respect to the previous zero crossing of plus or minus 150 ns, (i.e., 2.0 ± .15 μs 1.5 ± .15 μs, 1.0 ± .15 μs .5 ± .15 μs). The terminal shall respond to an input signal whose peak-to-peak amplitude, line-to-line, is within the range of 1.2 to 20.0 V. The terminal shall not respond to an input signal whose peak-to-peak amplitude, line-to-line, is within the range of 0.0 to .28 V. The voltages are measured at point A on figure 10.

4.5.2.2.2.2 Common mode rejections. Any signals from DC to 2.0 MHz, with amplitudes equal to or less than ± 10.0 volts peak, line-to-ground, measured at point A on figure 10, shall not degrade the performance of the receiver.

4.5.2.2.2.3 Input impedance. The magnitude of the terminal input impedance, when the RT is not transmitting, or has power removed, shall be a minimum of 2000.0 ohms within the frequency range of 75.0 kHz to 1.0 MHz. This impedance is that measured line-to-line at point A on figure 10.

4.5.2.2.2.4 Noise rejection. The terminal shall exhibit a maximum word error rate of one part in 10[7], on all words received by the terminal, after validation checks as specified in 4.4, when operating in the presence of additive white Gaussian noise distributed over a bandwidth of 1.0 kHz to 4.0 MHz at an RMS amplitude of 200 mV. A word error shall include any fault which causes the message error bit to be set in the terminal's status word, or one which causes a terminal to not respond to a valid command. The word error rate shall be measured with a 3.0V peak-to-peak, line-to-line, input to the terminal as measured at point A on figure 10. The noise tests shall be run continuously until, for a particular number of failures, the number of words

received by the terminal, including both command and data words, exceeds the required number for acceptance of the terminal, or is less than the required number for rejection of the terminal, as specified in table II. All data words used in the tests shall contain random bit patterns. These bit patterns shall be unique for each data word in a message, and shall change randomly from message to message.

A discussion of "terminals with direct coupled stubs" is covered in the preceding section, which discusses "terminals with transformer coupled stubs" (4.5.2 1).

4.6 Redundant data bus requirements. If redundant data buses are used, the requirements as specified in the following shall apply to those data buses.

This section of the standard was added to 1553B. The purpose was to clarify requirements relating to the electrical characteristics and operation of redundant data buses.

4.6.1 Electrical isolation. All terminals shall have a minimum of 45 dB isolation between data buses. Isolation here means the ratio in dB between the output voltage on the active data bus and the output voltage on the inactive data bus. This shall be measured using the test configuration specified in 4.5.2.1.1 or 4.5.2.2.1 for each data bus. Each data bus shall be alternately activated with all measurements being taken at point A on figure 12 for each data bus.

Although the data bus is commonly used in a dual-redundant manner, 1553A did not specify electrical isolation characteristics between redundant buses. This paragraph was added in 1553B to provide a ratio of the maximum transmitted signal on one bus to the minimum received threshold on the redundant bus. Test conditions that are specified in the referenced paragraphs correspond to terminals with

transformer-coupled stubs and with direct-coupled stubs.

4.6.2 Single event failures. All data buses shall be routed to minimize the possibility that a single event failure to a data bus shall cause the loss of more than that particular data bus.

This is an obvious requirement unique to military aircraft. This paragraph was added in 1553B.

4.6.3 Dual standby redundant data bus. If a dual redundant data bus is used, then it shall be a dual standby redundant data bus as specified in the following paragraphs.

4.6.3.1 Data bus activity. Only one data bus can be active at any given time except as specified in 4.6.3.2.

4.6.3.2 Reset data bus transmitter. If while operating on a command, a terminal receives another valid command from either data bus, it shall reset and respond to the new command on the data bus on which the new command is received. The terminal shall respond to the new command as specified in 4.3.3.8.

These new paragraphs in 1553B reflect the common practice in current aircraft of using the dual bus as one active, one standby, and it was the intent to restrict the operation of a dual data bus connected to terminal to a "one-at-a-time" operation. However, provision had to be made for a bus controller to override one bus to respond on the redundant bus. The requirement for this is in paragraph 4.6.3.2 above, and the reference to paragraph 4.3.3.8 is the response time requirement of a remote terminal to a valid command word.

5.0 DETAIL REQUIREMENTS (Not Applicable)

APPENDIX

10. General. The following paragraphs in this appendix are presented in order to discuss certain aspects of the standard in a general sense. They are intended to provide a user of the standard more insight into the aspects discussed.

10.1 Redundancy. It is intended that this standard be used to support rather than to supplant the system design process. However, it has been found, through application experience in various aircraft, that the use of a dual standby redundancy technique is very desirable for use in integrating mission avionics. For this reason, this redundancy scheme is defined in 4.6 of this standard. Nonetheless, the system designer should utilize this standard as the needs of a particular application dictate. The use of redundancy, the degree to which it is implemented, and the form which it takes must be determined on an individual application basis. Figures 10.1 and 10.2 illustrate some possible approaches to dual redundancy. These illustrations are not intended to be inclusive, but rather representative. It should be noted that analogous approaches exist for the triple and quad redundant cases.

10.2 Bus controller. The bus controller is a key part of the data bus system. The functions of the bus controller, in addition to the issuance of commands, must include the constant monitoring of the data bus and the traffic on the bus. It is envisioned that most of the routine minute details of bus monitoring (e.g., parity checking, terminal non-response time-out, etc.) will be embodied in hardware, while the algorithms for bus control and decision making will reside in software. It is also envisioned that, in general, the bus controller will be a general purpose airborne computer with a special input/output (I/O) to interface with the data bus. It is of extreme impor-

Note: RT — Remote Terminal
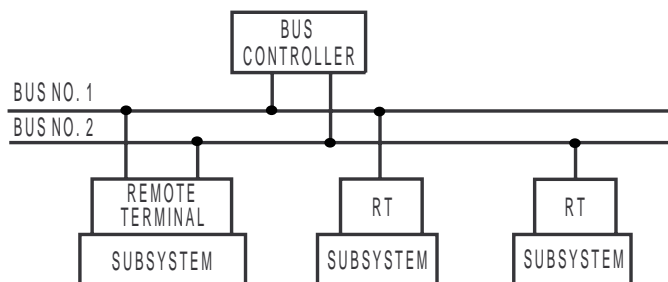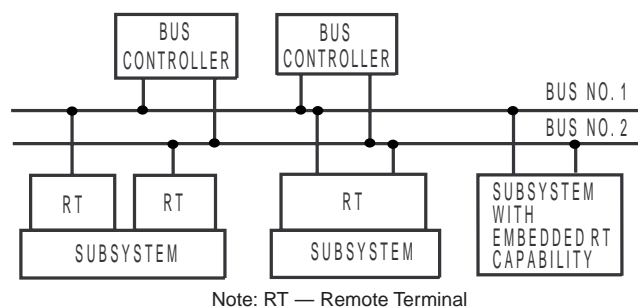
Figure 10.2 Illustration of possible redundancy.

tance in bus controller design that the bus controller be readily able to accommodate terminals of differing protocols and status word bits used. Equipment designed to MIL-STD-1553A will be in use for a considerable period of time; thus bus controllers must be capable of adjusting to their differing needs. It is also important to remember that the bus controller will be the focal point for modification and growth within the multiplex system, thus the software must be written in such a manner as to permit modification with relative ease.

10.3 Multiplex selection criteria. The selection of candidate signals for multiplexing is a function of the particular application involved, and criteria will in general vary from system to system Obviously, those signals which have bandwidths of 400 Hz or less are prime candidates for inclusion on the bus. It is also obvious that video, audio, and high speed parallel digital signals should be excluded. The area of questionable application is usually between 400 Hz and 3 kHz bandwidth. The transfer of these signals on the data bus will depend heavily upon the loading of the bus in a particular application. The decision must be based on projected future bus needs as well as the current loading. Another class of signals which in general are not suitable for multiplexing are those which can be typified by a low rate (over a mission) but possessing a high priority or urgency. Examples of such signals might be a nuclear event detector output or a missile launch alarm from a warning receiver. Such signals are usually better left hardwired, but they may be accommodated by the multiplex system if a direct connection to the bus controller's interrupt hardware is used to trigger a software action in response to the signal.

Figure 10.1 Illustration of possible redundancy.

10.4 High reliability requirements. The use of simple parity for error detection within the multiplex bus system was dictated by a compromise between the need for reliable data transmission, system overhead, and remote terminal simplicity. Theoretical and empirical evidence indicated that an undetected bit error rate of 10 can be expected from a practical multiplex system built to this standard. If a particular signal requires a bit error rate which is better than that provided by the parity checking, then it is incumbent upon the system designer to provide the reliability within the constraints of the standard or to not include this signal within the multiplex bus system. A possible approach in this case would be to have the signal source and sink provide appropriate error detection and correction encoding/decoding and employ extra data words to transfer the information. Another approach would be to partition the message transmit a portion at a time, and then verify (by interrogation) the proper transfer of each segment.

10.5 Stubbing. Stubbing is the method wherein a separate line is connected between the primary data bus line and a terminal. The direct connection of stub line causes a mismatch which appears on the waveforms. This mismatch can be reduced by filtering at the receiver and by using biphase modulation. Stubs are often employed not only as a convenience in bus layout but as a means of coupling a unit to the line in such a manner that a fault on the stub or terminal will not greatly affect the transmission line operation. In this case, a network is employed in the stub line to provide isolation from the fault. These networks are also used for stubs that are of such length that the mismatch and reflection degrades bus operation. The preferred method of stubbing is to use transformer coupled stubs, as defined in 4.5.1.5.1. This method provides the benefits of DC isolation, increased common mode protection, a doubling of effective stub impedance, and fault isolation for the entire stub and terminal. Direct coupled stubs, as defined in 4.5.1.5.2 of this standard, should be avoided if at all possible. Direct coupled stubs provide no DC isolation or common mode rejection for the terminal external to its subsystem. Further, any shorting fault between the subsystems internal isolation resistors (usually on a circuit board) and the main bus junction will cause failure of that entire bus. It can be expected that when the direct coupled stub length exceeds 1.6 feet, that it will begin to distort the main bus waveforms. Note that this length includes the cable runs internal to a given subsystem.

10.6 Use of broadcast option. The use of a broadcast message as defined in 4.3.3.6.7 of this standard represents a significant departure from the basic philosophy of this standard in that it is a message format which does not provide positive closed-loop control of bus traffic. The system designer is strongly encouraged to solve any design problems through the use of the three basic message formats without resorting to use of the broadcast. If system designers do choose to use the broadcast command, they should carefully consider the potential effects of a missed broadcast message, and the subsequent implications for fault or error recovery design in the remote terminals and bus controllers.

THIS PAGE INTENTIONALLY LEFT BLANK

III.
NOTICE I

MIL-STD-1553B
NOTICE 1(USAF)
12 February 1980

MILITARY STANDARD

AIRCRAFT INTERNAL TIME DIVISION
COMMAND/RESPONSE MULTIPLEX DATA BUS

TO ALL HOLDERS OF MIL-STD-1553B.

1. THE FOLLOWING PAGES OF MIL-STD-1553B HAVE BEEN REVISED AND SUPERSEDE THE
PAGES LISTED:

NEW PAGE   SUPERSEDED PAGE   DATE
iii   iii   21 September 1978
viii   viii   21 September 1978
34   34   21 September 1978

2. THE FOLLOWING NEW PAGES ARE TO BE INSERTED AS LISTED:

NEW PAGE

 viiia
  35

3. RETAIN THIS NOTICE PAGE AND INSERT BEFORE THE TABLE OF CONTENTS.

4. Holders of MIL-STD-1553B will verify that page changes indicated above have
been entered. The notice page will be retained as a check sheet. This
issuance, together with appended pages, is a separate publication. Each notice
is to be retained by stocking points until the Military Standard is completely
revised or cancelled.

5. This notice is applicable to all U.S. Air Force internal avionics
activities.

Custodian: Preparing activity
Air Force - 11   Air Force - 11

                                              Project MISC-FD32

*U.S. GOVERNMENT PRINTING OFFICE: 1980-603-121/1244   FSC MISC

MIL-STD-1553B
12 February 1980

FORWARD

    This standard contains requirements for aircraft internal time division command/response multiplex data bus techniques which will be utilized in systems integration of aircraft subsystems. Even with the use of this standard, subtle differences will exist between multiplex data buses used on different aircraft due to particular aircraft mission requirements and the designer options allowed in this standard. The system designer must recognize this fact, and design the multiplex bus controller hardware and software to accommodate such differences. These designer selected options must exist, so as to allow the necessary flexibility in the design of specific multiplex systems in order to provide for the control mechanism, architecture redundancy, degradation concept and traffic patterns peculiar to the specific aircraft mission requirements. * Appendix Section 20 selects those options which shall be required and further restricts certain portions of the standard for use in Air Force aircraft internal avionics applications.

Supersedes page iii of 21 September 1978

MIL-STD-1553B
12 February 1980

Supersedes page viii of 21 September 1978

MIL-STD-1553B
12 February 1980

viiia

MIL-STD-1553B
APPENDIX
12 February 1980

10.6 <u>Use of broadcast option</u>. The use of a broadcast message as defined in 4.3.3.6.7 of this standard represents a significant departure from the basic philosophy of this standard in that it is a message format which does not provide positive closed-loop control of bus traffic. The system designer is strongly encouraged to solve any design problems through the use of the three basic message formats without resorting to use of the broadcast. If system designers do choose to use the broadcast command, they should carefully consider the potential effects of a missed broadcast message, and the subsequent implications for fault or error recovery design in the remote terminals and bus controllers.

*20. <u>General.</u> This appendix is applicable to all U.s. Air Force aircraft internal avionics activities. The intent of the appendix is to select those options which shall be required and to further restrict certain portions of the standard for use in Air Force avionics. References in parenthesis are to the paragraphs in the standard that are affected.

*20.1 <u>Mode codes.</u> (4.3.3.5.1.7) The mode cotes for dynamic bus control, inhibit terminal flag bit, override inhibit terminal flag bit, selected transmitter shutdown and override selected transmitter shutdown shall not be transmitted on the data bus by bus controllers in Air Force avionics applications. However, these mode codes may be implemented in a remote terminal for Air Force avionics applications.

*20.2 <u>Broadcast command.</u> (4.3.3.6.7) The broadcast command shall not be transmitted on the data bus by bus controllers in Air Force avionics applications. However, this message format may be implemented in remote terminals. If the broadcast message format is implemented in a remote terminal, then that terminal shall also implement the transmit status word mode code as specified in 4.3.3.5.1.7.3. Note that the remote terminal address of 11111 is still reserved for broadcast, and shall not be used for any other purpose in Air Force Avionics applications.

*20.3 <u>Mode code indicators.</u>

*20.3.1 <u>Bus controllers.</u> (4.4.2) In Air Force avionics applications, the bus controller shall be able to utilize both 00000 and 11111 in the subaddress/mode field as defined in 4.3.3.5.1.7. In addition, if a bus controller is required to utilize any mode code in its operation, then it shall be required to implement the capacity to utilize all mode codes.

*20.3.2 <u>Remote terminals.</u> (4.4.3.1) All RT's which are designed for Air Force avionics applications, and which implement mode codes, shall respond properly to a mode code command, as defined in 4.3.3.5.1.7, with 00000 in the subaddress/mode field. In addition, such RT's may also respond to 11111 in the subaddress/mode field as a designer option. See section 20.8.1 for design consideration relating to the 11111 mode code indicator.

Supersedes page 34 of 21 September 1978

MIL-STD-1553B
APPENDIX
12 February 1980

*20.4 <u>Data bus cable</u>.

*20.4.1 <u>Shielding</u>. (4.5.1.1) The cable shield shall provide a minimum of 90.0 percent coverage.

*20.4.2 <u>Characteristic impedance</u>. (4.5.1.2) The actual (not nominal) characteristic impedance shall be within the range of 70.0 Ohms to 85.0 Ohms at a sinusoidal frequency of 1.0 megahertz (MHz)

*20.5 <u>Cable coupling.</u> (4.5.1.5.1.3) For Air Force avionics applications, the continuous shielding shall provide a minimum of 90.0 percent coverage.

*20.6 <u>Direct coupled stubs</u>. (4.5.1.5.2) Direct coupled studs shall not be utilized in Air Force avionics applications. .

*20.7 <u>Redundant data bus requirements</u>. (4.6) Dual standby redundant data buses as defined in 4.6.3 shall be utilized. There may be more than two data buses utilized but the buses must operate in dual redundant data bus pairs. 4.6.1 and 4.6.2 shall also apply.

*20.8 <u>Design considerations.</u> Avionics designed for Air Forces applications may be required to interface to existing avionics systems which were designed to preceding versions of the standard (e.g., the F-16 avionics suite). In this case, downward compatibility problems between the new avionics and the existing system can be minimized through the consideration of three key items:

*20.8.1 <u>Mode code indicator.</u> In some existing systems, such as the F-16, the bus controller uses 11111 to indicate a mode code command. The designer may wish to implement the capability in the new avionics to respond to 11111 mode code commands, in addition to the required capability for 00000 mode code commands.

*20.8.2 <u>Clock stability.</u> Since this version of the standard relaxed the transmission bit rate stability requirements (4.3.3.3), the avionics designer may wish to return to the stability requirements of the preceding version of the standard. The previous requirements were +0.01 percent long term and +0.001 percent short term stability.

*20.8.3 <u>Response time.</u> This version of the standard also expanded the maximum response time to 12.0 microseconds (4.3.3.8). The designer may also wish to return to the previous maximum response time of 7.0 microseconds as defined in 4.3.3.8 of this version of the standard.

THIS PAGE INTENTIONALLY LEFT BLANK

IV.
NOTICE II

CHANGE NOTICES ARE NOT CUMULATIVE
AND SHALL BE RETAINED UNTIL SUCH
TIME AS THE ENTIRE STANDARD IS
REVISED.

MIL-STD-1553B
NOTICE 2
8 September 1986

MILITARY STANDARD

DIGITAL TIME DIVISION

COMMAND/RESPONSE MULTIPLEX DATA BUS

TO ALL HOLDERS OF MIL-STD-1553B:

1. MAKE THE FOLLOWING PEN AND INK CHANGES:

Page i

Delete title and substitute: "DIGITAL TIME DIVISION COMMAND/RESPONSE MULTIPLEX
DATA BUS".

Page ii

Delete title and substitute: "Digital Time Division Command/Response Multiplex
Data Bus".

Page 1

Paragraph 1.1, second line: Delete "on aircraft".
Paragraph 1.2, third line: Delete "an aircraft" and substitute "a".

Page 3

Paragraph 3.11, first line, after "Bus controller": Insert "(BC)".
Paragraph 3.12, first line, after "Bus monitor": Insert "(BM)".

Page 21

Paragraph 4.4.3.1, add: "No combination of RT address bits, T/R bit,
subaddress/mode bits, and data word count/mode code bits of a command word
shall result in invalid transmissions by the RT. Subsequent valid commands
shall be properly responded to by the RT."

Page 30

Paragraph 4.6.3.2, first line: Delete "Reset data bus transmitter" and
substitute "Superseding valid commands".
Paragraph 4.6.3.2, second line: Delete "from either data bus" and substitute
"from the other data bus".

Page 31

Paragraph 10.2, eighth line: Delete "airborn".

FSC MCCR
DISTRIBUTION STATEMENT A.
Approved for public release;
distribution is unlimited

MIL-STD-15538
8 September 1986

2. THE FOLLOWING PAGES OF MIL-STD-1553B HAVE BEEN REVISED AND SUPERSEDE THE PAGES LISTED:

NEW PAGE	DATE	SUPERSEDED PAGE	DATE
iii	8 September 1986	iii	12 February 1980
iv	8 September 1986	iv	21 September 1978
vii	8 September 1986	vii	21 September 1978
viii	8 September 1986	viii	12 February 1980
viiia	8 September 1986	viiia	12 February 1980
33	21 September 1978	33	Reprinted Without Change
34	8 September 1986	34	12 February 1980
35	8 September 1986	35	12 February 1980
36	8 September 1986	INITIAL PUBLICATION	
37	8 September 1986	INITIAL PUBLICATION	
38	8 September 1986	INITIAL PUBLICATION	

3. RETAIN THIS NOTICE AND INSERT BEFORE TABLE OF CONTENTS.

4. Holders of MIL-STD-1553B will verify that page changes and additions indicated above have been entered. This notice page will be retained as a check sheet. This issuance, together with appended pages, is a separate publication. Each notice is to be retained by stocking points until the military standard is completely revised or cancelled.

Custodians: Preparing activity:
Army - ER   Air Force - 11
Navy - AS
Air Force - 11   Project No. MCCR-0008

MIL-STD-1553B
8 September 1986

FOREWORD

This standard contains requirements for a digital time division command/
response multiplex data bus for use in systems integration. Even with the use
of this standard, differences may exist between multiplex data buses in dif-
ferent system applications due to particular application requirements and the
designer options allowed in this standard. The system designer must recognize
this fact and design the multiplex bus controller hardware and software to
accommodate such differences. These designer selected options must exist to
allow the necessary flexibility in the design of specific multiplex systems in
order to provide for the control mechanism, architectural redundancy, degrada-
tion concept and traffic patterns peculiar to the specific application require
ments. Appendix, Section 30 selects those options which shall be required and
further restricts certain portions of the standard for the use in all dual
standby redundant applications for the Army, Navy, and Air Force.

Supersedes page iii dated 12 February 1980.

iii

```
   MIL-STD-1553B
   8 September 1986
```

CONTENTS

Supersedes page iv dated 21 September 1978

MIL-STD-1553B
8 September 1986

CONTENTS (Cont'd)

Supersedes page vii dated 21 September 1978.

MIL—STD—1553B
8 September 1986

CONTENTS (Cont'd)

Supersedes page viii dated 12 February 1980.

MIL-STD—1 553B
8 September 1986

CONTENTS (Cont'd)

Supersede page viiia dated 12 February 1980.

viiia

MIL-STD-1553B
21 September 1978

10.3 <u>Multiplex selection criteria</u>. The selection of candidate signals for multiplexing is a function of the particular application involved, and criteria will in general vary from system to system. Obviously, those signals which have bandwidths of 400 Hz or less are prime candidates for inclusion on the bus. It ia also obvious that video, audio, and high speed parallel digital signals should be excluded. The area of questionable application is usually between 400 Hz and 3 Khz bandwidth. The transfer of these signals on the data bus will depend heavily upon the loading of the bus in a particular applica-
tion. The decision must be based on projected future bus needs as well as the current loading. Another class of signals which in general are not suitable for multiplexing are those which can be typified by a low rate (over a mission) but possessing a high priority or urgency. Examples of such signals might be a nuclear event detector output or a missile launch alarm from a warning receiver. Such signals are usually better left hardwired, but they may be accommodated by the multiplex system if a direct connection to the bus controller's interrupt hardware is used to trigger a software action in response to the signal.

10.4 <u>High reliability requirements</u>. The use of simple parity for error detection within the multiplex bus system was dictated by a compromise between the need for reliable data transmission, system overhead, and remote terminal simplicity. Theoretical and empirical evidence indicates that an undetected bit error rate of 10^{-12} can be expected from a practical multiplex system built to this standard. If a particular signal requires a bit error rate which is better than that provided by the parity checking, then it is incumbent upon the system designer to provide the reliability within the constraints of the standard or to not include this signal within the multiplex bus system. A possible approach in this case would be to have the signal source and sink provide appropriate error detection and correction encoding/decoding and employ extra data words to transfer the information. Another approach would be to partition the message, transmit a portion at a time, and then verify (by interrogation) the proper transfer of each segment.

10.5 <u>Stubbing.</u> Stubbing is the method wherein a separate line is connected between the primary data bus line and a terminal. The direct connection of a stub line causes a mismatch which appears on the waveforms. This mismatch can be reduced by filtering at the receiver and by using bi-phase modulation. Stubs are often employed not only as a convenience in bus layout but as a means of coupling a unit to the line in such a manner that a fault on the stub or terminal will not greatly affect the transmission line operation. In this case, a network is employed in the stub line to provide isolation from the fault. These networks are also used for stubs that are of such length that the mismatch and reflection degrades bus operation. The preferred method of stubbing is to use transformer coupled stubs, as defined in 4.5.1.5.1. This method provides the benefits of DC isolation, Increased common mode protection, a doubling of effective stub impedance, and fault isolation for the entire stub and terminal. Direct coupled stubs, as defined in 4.5.1.5.2 of this standard, should be avoided if at all possible. Direct coupled stubs provide no DC isolation or common mode rejection for the terminal external to its subsystem. Further, any shorting fault between the subsystems internal isolation resistors (usually on a circuit board) and the main bus junction will cause failure of that entire bus. It can be expected that when the direct coupled stub length exceeds 1.6 feet, that it will begin to distort the main bus waveforms. Note that this length includes the cable runs internal to a given subsystem.

33

MIL-STD-1553B
APPENDIX
8 September 1986

* 10.6 <u>Use of broadcast option</u>. The use of a broadcast message as defined in 4.3.3.6.7 of this standard represents a significant departure from the basic philosophy of this standard in that it is a message format which does not provide positive closed-loop control of bus traffic. The system designer is strongly encouraged to solve any design problems through the use of the three basic message formats without resorting to use of the broadcast option. If system designers do choose to use the broadcast command, they should carefully consider the potential effects of a missed broadcast message, and the subsequent implications for fault or error recovery design in the remote terminals and bus controllers.

* 10.7 <u>Other related documents.</u> Several documents exist which are related to this standard. MIL-HDBK-1553 describes implementation practices for this standard and other related data. This standard is embodied in or referenced by the following international documents: NATO STANAG 3838, ASCC Air Standard 50/2, and UK DEF STAN 00-18 (PART 2)/Issue 1.

* 20. REFERENCED DOCUMENTS

Not applicable.

* 30. GENERAL REQUIREMENTS

* 30.1 <u>Option selection.</u> This section of the appendix shall select those options required to further define portions of the standard to enhance tri-service interoperability. References in parentheses are to paragraphs in this standard which are affected.

* 30.2 <u>Application.</u> Section 30 of this appendix shall apply to all dual standby redundant applications for the Army, Navy, and Air Force. All Air Force aircraft internal avionics applications shall be dual standby redundant, except where safety critical or flight critical requirements dictate a higher level of redundancy.

* 30.3 <u>Unique address (4.3.3.5.1.2).</u> All remote terminals shall be capable of being assigned any unique address from decimal address 0 (00000) through decimal address 30 (11110). The address shall be established through an external connector, which is part of the system wiring and connects to the remote terminal. Changing the unique address of a remote terminal shall not require the physical modification or manipulation of any part of the remote terminal. The remote terminal shall, as a minimum, determine and validate its address during power-up conditions. No single point failure shall cause a terminal to validate a false address. The remote terminal shall not respond to any messages if it has determined its unique address is not valid.

Supersedes page 34 dated 12 February 1980.

MIL-STD-1553B
APPENDIX
8 September 1986

* 30.4 <u>Mode codes (4.3.3.5.1.7)</u>

* 30.4.1 <u>Subaddress/mode (4,3.3.5.1.4).</u> An RT shall have the capability to respond to mode codes with both subaddress/mode of 00000 and 11111. Bus controllers shall have the capability to issue mode commands with both subaddress/mode of 00000 and 11111. The subaddress/mode of 00000 and 11111 shall not convey different information.

* 30.4.2 <u>Required mode codes (4.3.3.5.1.7)</u>

* 30.4.2.1 <u>Remote terminal required mode codes.</u> An RT shall implement the follow ing mode codes as a minimum:

Mode Code	Function
00010	Transmit status word
00100	Transmitter shutdown
00101	Override transmitter shutdown
01000	Reset remote terminal

* 30.4.2.2 <u>Bus controller required mode codes.</u> The bus controller shall have the capability to implement all of the mode codes as defined in 4.3.3.5.1.7. For Air Force applications, the dynamic bus control mode command shall never be issued by the bus controller.

* 30.4.3 <u>Reset remote terminal (4.3.3.5.1.7.9)</u>. An RT receiving the reset remote terminal mode code shall respond with a status word as specified in 4.3.3.5.1.7.9 and then react. While the RT is being reset, the RT shall respond to a valid command with any of the following: no response on either data bus, status word transmitted with the busy bit set, or normal response. If any data is transmitted from the RT while it is being reset, the information content of the data shall be valid. An RT receiving this mode code shall complete the reset function within 5.0 milliseconds following transmission of the status word specified in 4.3.3.5.1.7.9. The time shall be measured from the mid-bit zero crossing of the parity bit of the status word to the mid-sync zero crossing of the command word at point A on figures 9 and 10.

* 30.4.4 <u>Initiate RT self test (4.3.3.5.1.7.4).</u> If the initiate self test mode command is implemented in the RT, then the RT receiving the initiate self test mode code shall respond with a status word as specified in 4.3.3.5.1.7.4 and then initiate the RT self test function. Subsequent valid commands may terminate the self-test function. While the RT self test is in progress, the RT shall respond to a valid command with any of the following: no response on either data bus, status word transmitted with the busy bit set, or normal response. If any data is transmitted from the RT while it is in self test, the information content of the data shall be valid. An RT receiving this mode code shall complete the self test function and have the results of the self test available within 100.0 milliseconds following transmission of the status word specified in 4.3.3.5.1.7.4, The time shall be measured from the mid-bit zero crossing of the parity bit of the status word to the mid-sync zero crossing of the command word at point A on figures 9 and 10.

Supersedes page 35 dated 12 February 1980.

```
MIL-STD-1553B
APPENDIX
8 September 1986
```

* 30.5 <u>Status word bits (4.3.3.5.3)</u>

* 30.5.1 <u>Information content</u>. The status word transmitted by an RT shall contain valid information at all times, e.g., following RT power up, during initialization, and during normal operation.

* 30.5.2 <u>Status bit requirements (4.3.3.5.3)</u>. An RT shall implement the status bits as follows:

Message error bit (4.3.3.5.3.3) - Required

Instrumentation bit (4.3.3.5.3.4) - Always logic zero

Service request bit (4.3.3.5.3.5) - Optional

Reserved statue bits (4.3.3.5.3.6) - Always logic zero

Broadcast command received bit (4.3.3.5.3.7) - If the RT implements the broadcast option, then this bit shall be required.

Busy bit (4.3.3.5.3.8) - As required by 30.5.3

Subsystem flag bit (4.3.3.5.3.9) - If an associated subsystem has the capability for self test, then this bit shall be required.

Dynamic bus control acceptance bit (4.3.3.5.3.10) - If the RT implements the dynamic bus control function, then this bit shall be required.

Terminal flag bit (4.3.3.5.3.11) - If an RT has the capability for self test, then this bit shall be required.

* 30.5.3 <u>Busy bit (4.3.3.5.3.8).</u> The existence of busy conditions is discouraged. However, any busy condition, in the RT or the subsystem interface that would affect communication over the bus shall be conveyed via the busy bit. Busy conditions, and thus the setting of the busy bit, shall occur only as a result of particular commands/messages sent to an RT. Thus for a non-failed RT, the bus controller can, with prior knowledge of the remote terminal characteristics, determine when the remote terminal can become busy and when it will not be busy. However, the RT may also set the busy bit (in addition to setting the terminal flag bit or subsystem flag bit) as a result of failure/fault conditions within the RT/subsystem.

* 30.6 <u>Broadcast (4.3.3.6.7).</u> The only broadcast commands allowed to be transmitted on the data bus by the bus controller shall be the broadcast mode commands identified in table 1. The broadcast option may be implemented in remote terminals. However, if implemented, the RT shall be capable of distinguishing between a broadcast and a non-broadcast message to the same subaddress for non-mode command messages. The RT address of 11111 is still reserved for broadcast and shall not be used for any other purpose.

MIL-STD-1553B
APPENDIX
8 September 1986

* 30.7 Data wrap-around (4.3.3.5.1.4). Remote terminals shall provide a receive subaddress to which one to N data words of any bit pattern can be received. Remote terminals shall provide a transmit subaddress from which a minimum of N data words can be transmitted. N is equal to the maximum word count from the set of all messages defined for the RT. A valid receive message to the data wrap-around receive subaddress followed by a valid transmit command to the data wrap-around transmit subaddress, with the same word count and without any intervening valid commands to that RT, shall cause the RT to respond with each data word having the same bit pattern as the corresponding received data word. A data wrap-around receive and transmit subaddress of 30 (11110) is desired.

* 30.8 Message formats (4.3.3.6). Remote terminals shall, as a minimum, implement the following non-broadcast message formats as defined in 4.3.3.6: RT to BC transfers, BC to RT transfers, RT to RT transfers (receive and transmit), and mode command without data word transfers. For non-broadcast messages, the RT shall not distinguish between data received during a BC to RT transfer or data received during a RT to RT transfer (receive) to the same subaddress. The RT shall not distinguish between data to be transmitted during an RT to BC transfer or data to be transmitted during an RT to RT transfer (transmit) from the same subaddress. Bus controllers shall have the capability to issue all message formats defined in 4.3.3.6.

* 30.9 RT to RT validation (4.3.3.9). For RT to RT transfers, in addition to the validation criteria specified in 4.4.3.6, if a valid receive command is received by the RT and the first data word is received after 57.0 plus or minus 3.0 microseconds, the RT shall consider the message invalid and respond as specified in 4.4.3.6. The time shall be measured from the mid-bit zero crossing of the parity bit of the receive command to the mid-sync zero crossing of the first expected data word at point A as shown on figures 9 and 10. lt is recommended that the receiving RT of an RT to RT transfer verify the proper occurrence of the transmit command word and status word as specified in 4.3.3.6.3.

* 30.10 Electrical characteristics (4.5)

* 30.10.1 Cable shielding (4.5.1.1). The cable shield shall provide a minimum of 90.0 percent coverage.

* 30.10.2 Shielding (4.5.1). All cable to connector junctions, cable terminations, and bus-stub junctions shall have continuous 360 degree shielding which shall provide a minimum of 75.0 percent coverage.

* 30.10.3 Connector polarity. For applications that use concentric connectors or inserts for each bus, the center pin of the connector or insert shall be used for the high (positive) Manchester bi-phase signal. The inner ring shall be used for the low (negative) Manchester bi-phase signal.

* 30.10.4 Characteristic impedance (4.5.1.2). The actual (not nominal) characteristic impedance of the data bus cable shall be within the range of 70.0 ohms to 85.0 ohms at a sinusoidal frequency of 1.0 megahertz,

37

```
MIL-STD-1553B
APPENDIX
8 September 1986
```

.

* 30.10.5 <u>Stub coupling (4.5.1.5)</u>. For Navy applications, each terminal shall
have both transformer and direct coupled stub connections externally available.
For Navy systems using these terminals, either transformer or direct coupled
connections may be used. For Army and Air Force applications, each terminal
shall have transformer coupled stub connections, but may also have direct
coupled stub connections. For Army and Air Force systems, only transformer
coupled stub connections shall be used. Unused terminal connections shall
have a minimum of 75 percent shielding coverage.

* 30.10.6 <u>Power on/off noise.</u> A terminal shall limit any spurious output during
a power-up or power-down sequence. The maximum allowable output noise ampli-
tude shall be ±250 mV peak, line-to-line for transformer coupled stubs and
±90 mV peak, line-to-line for direct coupled stubs, measured at point A of
figure 12.

V.
PRODUCTION
TEST PLAN
FOR REMOTE TERMINALS

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Page 2

TABLE OF CONTENTS
(Continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

Page 3

1.   SCOPE:

This test plan is broken into two major sections for the production testing of remote terminals: Electrical and Protocol.

1.1   General: This production test plan defines the test requirements for verifying that Remote Terminals meet the requirements of MIL-STD-1553B, "Digital Time Division Command/Response Multiplex Data Bus."

1.2   Application: This is a general test plan for any remote terminal designed to meet the require ments of MIL-STD-1553B. These requirements shall apply to the remote terminal under test, when invoked in a specification or statement of work.

Tests specified may be performed in any order, combined with one another or combined with other tests of the subsystem in which the Remote Terminal may be embedded.

The Remote Terminal which is under test must also have its design validated per the "Validation Test Plan for the Digital Time Divislon Command/Response Multiplex Data Bus Remote Terminals."

The tests specified are to be used in production test of a remote terminal chip, module or function within an equipment. It is intended that all stimuli and measurements be made via external connections and/or test points. No required access to any other points is intended.

Special requirements such as those needed to meet armament bus requirements when specified in the individual equipment specification may, when within 1553B requirements, replace values or tolerances specified in this test plan for the purpose of generating a UUT production test procedure.

2.   APPLICABLE DOCUMENTS:

2.1   Standards, Military: MIL-STD-1553B, 21 Sep 1978, "Aircraft Internal Time Division Command/Response Multiplex Data Bus" with Notice 2, 8 September 1986.

2.2   Other Documents: MIL-HDBK-1553, Notice 1, 24 September 1986 "Validation Test Plan for the Digital Time Division Command/Response Multiplex Data Bus Remote Terminals."

3.   DEFINITIONS:

3.1   Responses: The following are definitions of the responses of the RT as used in this test plan. In each case, the status word must have the correct terminal address and unused status bits set to zero.

3.1.1   BCR - Broadcast Command Received: The broadcast command received bit (bit time 15) is set in status word.

3.1.2   BUSY - Busy: The busy bit (bit time 16) is set in the status word.

3.1.3 <u>CS - Clear Status</u>: The status word may have the busy bit or service request bit set, or both. All other status code bits in the status word must be zero and the associated message must have the proper word count.

3.1.4 <u>DBA - Dynamic Bus Acceptance</u>: The dynamic bus control acceptance bit (bit time 18) is set in the status word.

3.1.5 <u>ME - Message Error</u>: The message error bit (bit time 9) is set in the status word.

3.1.6 <u>NR - No response</u>: The addressed terminal does not produce any response to the command.

3.1.7 <u>Respond in Form</u>: A terminal is said to "respond in form" if its response to an illegal command as defined in 4.4.3.4 of MIL-STD-1553B consists of a response formatted as though it were a legal command.

3.1.8 <u>SF - Subsystem Flag</u>: The subsystem flag bit (bit time 17) is set in the status word.

3.1.9 <u>TF - Terminal Flag</u>: The terminal flag bit (bit time 19) is set in the status word.

3.2 <u>Other Definitions</u>:

3.2.1 UUT - Unit Under Test.

3.2.2 RT - Remote Terminal.

3.2.3 BC - Bus Controller.

4. <u>GENERAL TEST REQUIREMENTS</u>:

The following paragraphs define the configurations, pass/fail criteria, and general procedures for testing remote terminals. Specifically, this document contains the test configurations and procedures for the Electrical Tests (5.1) and the Protocol Tests (5.2) for MIL-STD-1553B remote terminals. The remote terminal under test is referred to as the unit under test (UUT). Proper terminal responses are defined in each test paragraph.

4.1   General Monitoring Requirements: In addition to the specific tests that follow, certain RT parameters must be continuously monitored throughout all tests. These parameters are:

1. response time
2. contiguous data
3. proper Manchester encoding
4. proper bit count
5. odd parity
6. proper word count
7. proper terminal address in the status word
8. reserved status and instrumentation bits in the status word are set to zero
9. proper sync

The UUT shall have falled the test if at any time during the test any of these parameters fail to meet the requirements of MIL-STD-1553B.

5.   DETAILED REQUIREMENTS:

5.1   Electrical Tests: All tests shall use Fig. 1, General Test Configuratlon, with all measure-mentstaken at point "A", unless otherwise noted. Each test paragraph contains the require-ments for both transformer and direct coupled stubs. A UUT which provides both trans former and direct coupled stubs must be tested on both stubs. Electrical tests shall be per-formed on all buses for UUTs with redundant bus configurations.

5.1.1   Output Characterstics: The following tests are designed to verify that all UUT output characteristics comply with MIL-STD-1553B. These tests shall be performed after establishing communications between the test equipment and the UUT.

If necessary, use the Optional Test Configuration, Fig. 2, In order to isolate the UUT output from any reactive load presented by the Tester.

5.1.1.1 Amplitude: A valid, legal transmit command shall be transmitted to the UUT, requesting the maximum number of words that it is capable of sending. The amplitude of the wave-form transmitted by the UUT shall be measured, peak-to-peak, as shown in Fig. 3. The pass criteria for Vpp for transformer coupled stubs shall be 18.0 V minimum and 27.0 V maximum. The pass criteria for Vpp for direct coupled stubs shall be 6.0 V minimum and 9.0 V maximum.

5.1.1.2 Rise and Fall Times: A valid, legal transmit command shall be sent to the UUT, requesting at least one data word. The rise and fall time of the UUT waveform shall be measured between the 10 and 90% points of the waveform as shown in Fig. 3. The measurements shall be taken at both the rising and falling edges of a sync waveform and a data bit waveform. The rise time (T_R) and the fall time (T_F) shall be measured. The pass criteria shall be 100 ns < = T_R < =300 ns and 100 ns < = T_F < = 300 ns.

5.1.1.2   (Continued):

Note: The rise time of the sync waveform shall be measured at the mid-crossing of a data word sync, and the fall time of the sync waveform shall be measured at the mid-crossing of the status word sync. The rise and fall times of the data bit waveform shall be measured at a zero crossing where the prior zero crossing and next zero crossing are at 500 ns intervals from the measured zero crossing.

5.1.1.3   <u>Zero Crossing Stability</u>: Selected valid legal transmit commands shall be sent to the UUT, requesting the UUT to transmit words having zero crossing time intervals of 500, 1000, 1500 and 2000 ns. The zero crossing time (T_{ZC}) shall be measured for both positve to negative and the negative to positive waveforms as shown if Fig. 4. The pass criteria shall be that T_{ZC} = 500 + 25, 1000 ± 25, 1500 2000 ± 25 ns.

5.1.1.4   <u>Distortion. Overshoot and Ringing</u>: A valid legal transmit command shall be sent to the UUT, requesting the UUT to transmit at least one data word. The distortion of the waveform, distortion voltage (VD), shall be measured as indicated in Fig. 3. Pass criterion shall be VD < = 900 mV, for transformer coupled stubs and VD < = 300 mV, for direct couple stubs.

5.1.1.5   <u>Output Symmetry</u>: A valid legal transmit command shall be sent to the UUT requesting the maximum number of data words that the UUT is capable of transmitting. The output symmetry is determined by measuring the waveform tail-off at the end of each message. The maximum residual voltage (VR) shall be measured as shown in Fig. 3. The intermessage gap time shall be a minimum of 1 ms. The pass criterion shall be VR < = 250 mV peak for transformer coupled stubs and VR < = 90 mV peak for direct coupled stubs after time TT (the time beginning 2.5 µs after the mid-bit zero crossing of the last parity bit). This test shall be run six times with each data word in the message having the same bit pattern.

The six data word patterns that shall be used are:

8000(HEX), 7FFF(HEX), 0000(HEX), FFFF(HEX) 5555(HEX), and AAAA(HEX)

Note: For terminals that cannot transmit data words, the output symmetry test shall be performed on the status word response to a receive command or mode command.

5.1.1.6   <u>Output Noise</u>: The test configuration shown in Fig. 5 shall be used to test the UUT inactive bus output noise levels. The test shall be conducted while the UUT is in the power-on receive state. The output noise (Vrms) shall be measured at point "A" as shown in Fig. 5. Measurements shall be made with an instrument that has a minimum frequency bandwidth of from DC to 10 MHz. The pass criterion shall Vrms < = 14 mV for transformer coupled stubs and Vrms < = 5.O mV for direct coupled stubs.

5.1.1.7   <u>Power On/Off Noise</u>: A UUT shall not emit any spurious differential output during a power-up or power-down sequence. Power shall be applied to the UUT and any outputs from the UUT to the bus shall be measured. Power shall be removed from the UUT and any output to the bus from the UUT shall be measured.

The pass criterion shall be:

1. For transformer coupled stubs, any spurious noise pulses produced shall be less than ±250 mV peak.
2. For direct coupled stubs, any spurious noise pulses produced shall be less than ±90 mV peak.

NOTE: Use normal on/off power sequence of UUT.

5.1.2   <u>Input Characteristics</u>: The input tests are designed to verify that multiplex devices can properly decode bi-phase data.

5.1.2.1   Input Waveform Compatibility:

5.1.2.1.1   <u>Amplitude Variations</u>: A sequence of valid legal receive commands shall be transmitted to the UUT. The test shall be performed for input voltage levels of 0.20, 0.86, and 6.0 Vpp for transformer coupled stubs, and at 0.28, 1.2, and 9.0 Vpp for direct coupled stubs. A minimum of 1000 commands shall be transmitted to the UUT at each input level. The response of the UUT shall be monitored for every message at each input level.

The pass criteria shall be:

1. NR for transformer coupled stub inputs of 0.20 Vpp.
2. CS for transformer coupled stub inputs of 0.86 Vpp.
3. CS for transformer coupled stub inputs of 6.0 Vpp.
4. NR for direct coupled stub inputs of 0.28 Vpp.
5. CS for direct coupled stub inputs of 1.2 Vpp.
6. CS for direct coupled stub inputs of 9.0 Vpp.

5.1.2.1.2   <u>Input Impedance</u>: The input impedance of the UUT shall be measured first with the power on and repeated with the UUT power off. The input impedance, Zin, shall be measured with a sinusoidal waveform having an amplitude 1.0 to 2.0 Vrms, at a frequency of 1.0 MHz. The pass criterion shall be Zin > = 1000   for transformer coupled stubs and Zin > = 2000   for direct coupled stubs.

For this test, do not use Fig. 1; remove all loads from UUT.

5.2   <u>Protocol Tests</u>: All tests in this section shall use the test configuration shown in Fig. 1. The test signal amplitude shall be 3.0 Vpp for direct coupled stubs and 2.1 Vpp for transformer coupled stubs measured at point A. For UUTs having both direct and transformer coupled stubs, the protocol tests need only be performed on one stub type per bus. Any condition, which causes the UUT to respond other than as called out in MIL-STD-1553B, to lock up or require a power cycle in order to recover from a failure, shall automatically cause that UUT to fail the test. The protocol tests shall be performed on all buses for UUTs with redundant bus configurations.

Page 8

5.2.1   RT Response to Command Words:

5.2.1.1   RT Address: The purpose of this test is to verify that the UUT responds only to valid legal commands, that is, those that contain the address of the UUT.

5.2.1.1.1   Valid RT Address: The test equipment shall transmit to the UUT one valid legal transmit command and one valid legal receive command with the minimum word count for the UUT. Associated data may be random or controlled depending on UUT requirements. The pass criteria shall be that the UUT responds with CS for each command. If the UUT has variable address capability, this test shall be repeated for the following addresses:

00001, 00010, 00100, 01000, 10000, 00000

5.2.1.1.2   Invalid RT Address: The test equipment shall transmit to the UUT receive commands and transmit commands using all RT addresses not assigned to the UUT (with the exception of address 11111). Subaddress and word count fields shall be identical to those used in 5.2.1.1.1. The pass criteria shall be NR for each command.

5.2.1.2   Word Count: The purpose of the test is to verify that the UUT responds properly to all word counts implemented. During the test, the test equipment will transmit one valid, legal, non-mode transmit command and one valid, legal, non-mode receive command for each implemented word count. Only one subaddress is needed for any word count. The pass criteria shall be CS for each command.

5.2.1.3   Subaddress: The purpose of this test is to verify that the UUT responds properly to all subaddresses implemented. During this test, the test equipment shall transmit one valid, legal, non-mode transmit command and one valid, legal, non-mode receive command for each implemented subaddress. Only one word GOunt is needed for any subaddress. The pass criteria shall be CS for each command.

5.2.1.4   Error Injection: These tests are intended to examine the UUT response when specific errors are forced into the message stream. The UUT shall not respond with a status word. If the data words it decodes fail any of the validity tests, the UUT should internally set the message error bit in its status word buffer. If the command word it decodes fails any of the validity tests, it shall ignore the command and wait for the next command (the ME bit should not be internally set in this case). For this test, valid legal commands will be sent to the UUT for Steps 1 and 3 in the test sequence below. Each error condition listed in Table 1 shall be injected in Step 2 of the test sequence. One error per test sequence shall be implemented.

1. Transmit a valid, legal command to the UUT.
2. Transmit to the UUT a message which contains one of the error conditions listed in Table 1.
3. Transmit a valid, legal command to the UUT.

5.2.1.4   (Continued):

The pass criteria shall be:

1. The UUT shall respond with CS for Steps 1 and 3 in the test sequence above.
2. The UUT shall not respond to commands for Step 2 in the test sequence above.

5.2.2   Optional Operation: This section provides for testing the optional requirements of MIL-STD-1553B. If a remote terminal implements any of the options, it shall be tested in accordance with the test identified for the option.

5.2.2.1   Dual Redundant Operation: This test shall be performed only if the UUT is configured with dual redundant buses. The requirements are as follows:

1. The UUT is required to accept a valid command received on the alternate bus while responding to a command on the original bus.
2. The UUT is required to respond to the valid command occurring later in time when overlapping valid commands are received on both buses.
3. When Step 1 or 2 occurs, the UUT shall reset and respond to the new command on the alternate bus.

Legal messages shall be used in this test. The following test sequence shall be performed twice for each interrupting command, once for each redundant bus:

Step 1. Send a valid command to the UUT requesting the maximum number of data words that the UUT is designed to transmit.
Step 2. Send the interrupting command on the alternate bus beginning no sooner than 4.0 µs after the beginning of the first command.
Step 3. Send a valid "Transmit Status" mode command on the first bus after the messages on both buses have been completed. (If "Transmit Status" is not used, delete this step.)

The pass criteria shall be: Step 1 - NR, truncated message or CS, Step 2 - CS, and Step 3 - CS if transmit status is implemented.

5.2.2.2   Mode Commands: The purpose of these tests is to verify that the UUT responds properly to implemented mode commands. The tests are not intended to verify the mission aspects stated in the equipment specification. The UUT shall be tested for each mode code implemented with a subaddress field mode code indicator of all zeros and repeated with a subaddress field of all ones. The pass criteria are defined in each test paragraph.

TABLE 1 - Error Formats

ERROR TYPE	LOCATION
Parity	Receive command word
Parity	Transmit command word
Parity	Data Word
Bit Count (−1 bit)	Receive command word
Bit Count (+2 bits)	Receive command word
Bit Count (−1 bit)	Transmit command word
Bit Count (+2 bits)	Between two data words
Bit Count (−1 bit)	Between two data words
Bi-phase (high)	Ripple through transmit command
Bi-phase (high)	Ripple through receive command
Bi-phase (low)	Ripple through transmit command
Bi-phase (low)	Ripple through receive command
Bi-phase (low)	Ripple through data word
Bi-phase (high)	Ripple through data word
Sync	Command word, patterns:
	111100
	110000
	111001
	011000
	000111
Sync	Data word, patterns:
	000011
	001111
	000110
	100111
	111000
Contiguous data	Between command and data words
Contiguous data	Between two data words
Message length (+)	Receive message plus one data word
Message length (−)	Receive message minus one data word
Message length (+)	Transmit command plus data word

NOTE: For contiguous data errors, receive commands shall be used and the contiguous data error shall be 4.0 μs as measured from the mid-bit zero crossing of the parity bit of the preceding word to the mid-sync zero crossing of the following word.

Page 11

5.2.2.2.1   <u>Dynamic Bus Control</u>: The purpose of this test is to verify that the UUT has the ability to recognize the "Dynamic Bus Control" mode command and to take control of the data bus.A valid legal dynamic bus control mode command shall be sent to the UUT. The pass criteria shall be that the UUT responds with a DBA upon acceptance of bus control or a CS upon rejection of bus control. The UUT shall take control of the data bus when its response is DBA as required in the UUT's design specification.

5.2.2.2.2 5   <u>Synchronize</u>: The following paragraphs provide the test criteria for the "Synchronize" mode commands:

5.2.2.2.2.1   <u>Synchronize (Without Data Word)</u>: The purpose of this test is to verify that the UUT has the ability to recognize a "Synchronize" mode command without a data word. A valid legal "Synchronize" (without data word) mode command shall be sent to the UUT. The pass criterion shall be that the UUT responds with CS.

5.2.2.2.2.2   <u>Synchronize (With Data Word)</u>: The purpose of this test is to verify that the UUT has the ability to recognize a "Synchronize" mode command with a data word. A valid legal "Synchronize" (with data word) mode command shall be sent to the UUT. The pass criterion shall be that the UUT responds with CS.

5.2.2.2.3   <u>Transmit Status</u>: The purpose of this test is to verify that the UUT has the ability to recognize the "Transmit Status" mode command and respond with its last status word. The following sequence shall be performed:

Step 1.   A valid legal message shall be sent to the UUT.
Step 2.   A legal receive command with a parity error in a data word
          shall be sent to the UUT.
Step 3.   A valid legal "Transmit Status" mode command shall be sent to
          the UUT.
Step 4.   Repeat Step 3.
Step 5.   A valid legal receive command shall be sent to the UUT.

The pass criteria for the above steps shall be as follows: Step 1 CS; Step 2 - NR; Step 3 - ME; Step 4 - ME; Step 5 - CS.

5.2.2.2.4   <u>Initiate Self-Test</u>: The purpose of this test is to verify that the UUT has the ability to recognize the "Initiate Self-Test" mode command. A valid legal "Initiate Self-Test" mode command shall be sent to the UUT. The pass criterion shall be that the UUT responds with CS.

Note: Normal UUT operation may be affected during the Self-Test execution time. Detailed requirements shall be as defined by the UUT's design specification.

5.2.2.2.5   <u>Transmit BIT Word</u>: The purpose of this test is to verify that the UUT has the ability to recognize the "Transmit BIT Word" mode command. A valid legal "Transmit BIT Word" mode command shall be sent to the UUT. The pass criterion shall be that the UUT responds with a CS followed by a valid data word.

5.2.2.2.6   <u>Transmitter Shutdown and Override</u>: The purpose of this test is to verify that the UUT has the ability to recognize the "Transmitter Shutdown" and "Override Transmitter shutdown" mode commands. The pass criterion for each individual test is contained in one of the paragraphs below. To test all bus combinations, in this sequence each bus in turn shall be identified as the primary bus. All other buses shall be tested in turn as the selected alternate bus using Steps 1 through 10.

Step 1.   A valid legal command shall be sent on the primary bus to the UUT. A CS response shall be verified.
Step 2.   A valid legal command shall be sent on the selected alternate bus to the UUT. A CS response shall be verified.
Step 3.   A valid legal mode command to shut down the transmitter shall be sent to the UUT on the primary bus. A CS response shall be verified.
Step 4.   A valid legal command shall be sent on the selected alternate bus to the UUT. A NR shall be verified.
Step 5.   A valid legal command shall be sent on the primary bus to the UUT. A CS response shall be verified.
Step 6.   A valid legal mode command to override the transmitter shutdown shall be sent to the UUT on the selected alternate bus. A NR shall be verified.
Step 7.   A valid legal command shall be sent to the UUT on the selected alternate bus. A NR shall be verified.
Step 8.   A valid legal mode command to override transmitter shutdown shall be sent to the UUT on the primary bus. A CS shall be verified.
Step 9.   A valid legal command shall be sent on the selected alternate bus to the UUT. A CS shall be verified.
Step 10. A valid legal command shall be sent on the primary bus to the UUT. A CS shall be verified.

5.2.2.2.6.1   <u>Dual Redundant Shutdowns and Overrides</u>: This test shall verify that the UUT recognizes the dual redundant mode commands to shut down the alternate bus transmitter and to override the shutdown.

A valid legal "Transmitter Shutdown" mode command shall be encoded into the command word to cause an alternate bus transmitter shutdown. A valid legal "Override Transmitter Shutdown" mode command shall be encoded into the command word to cause an override of the transmitter shutdown. The test sequence in 5.2.2.2.6 shall be used for each case including verification of the UUT response indicated. The pass criterion shall be that the UUT performs as defined in each step.

5.2.2.2.6.2 Selective Bus Shutdowns and Overrides: This test shall verify that the UUT recognizes the multi-redundant mode commands to shut down a selected bus transmitter and to override the shutdown. A valid legal "Selected Transmitter Shutdown" receive mode command shall be encoded, accompanied by the appropriate data word, to cause a selective bus transmitter shutdown. A valid legal "Override Selected Transmitter Shutdown" receive mode command shall be encoded accompanied by the appropriate data word to cause an override of the selected bus transmitter shutdown. The test sequence in 5.2.2.2.6 shall be used for each case including verification of the UUT response indicated. The pass criteria shall be that the UUT performs as defined in each step.

5.2.2.2.7 Terminal Flag Bit Inhibit and Override: This test verifies that the UUT recognizes and responds properly to the mode commands of "Inhibit Terminal Flag Bit" and "Override Inhibit Terminal Flag Bit." Beginning in Step 2 of the test sequence below, the UUT shall be caused to set the terminal flag bit.

Step 1. A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response verified.
Step 2. Introduce a condition that will set the terminal flag in the UUT status response. Send a valid legal receive command with at least one data word to the UUT and verify that the TF bit is set in the returned status word.
Step 3. A valid legal "Inhibit Terminal Flag" mode command shall be transmitted to the UUT and a CS or TF response verified.
Step 4. Repeat Step 1 and verify a CS response.
Step 5. A valid legal "Override Inhibit Terminal Flag" mode command shall be transmitted to the UUT and a CS or TF verified in the returned status word.
Step 6. A valid legal receive command with at least one data word shall be transmitted and a TF verified in the returned status word.
Step 7. Procedures, as defined for the UUT, shall be performed which reset the TF bit.
Step 8. Repeat Step 1.

The pass criterion shall be that the UUT performs as defined in each step.

5.2.2.2.8 Reset Remote Terminal: This test shall verify that the UUT has the capability to recognize the mode command to reset itself to a power up initialized state. For this test, a reset mode command shall be transmitted to the UUT and a CS response verified.

Note: Normal UUT operation may be affected during the reset execution time. Detailed requirements shall be as defined by the UUT's design specification.

Page 14

5.2.2.2.9   <u>Transmit Vector Word</u>: This test verifies the capability of the UUT to recognize and respond properly to a "Transmit Vector Word" mode command. A valid legal "Transmit Vector Word" mode command shall be transmitted to the UUT and a CS response shall be verified.

5.2.2.2.10   <u>Transmit Last Command</u>: This test verifies that the UUT recognizes and responds properly to a "Transmit Last Command" mode command. The following test sequence shall be used:

Step 1.   A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response verified.
Step 2.   A valid legal receive command, different from that used in Step 1 above with at least one data word, shall be transmitted to the UUT and a parity error shall be encoded into the first data word. A NR shall be the proper UUT action.
Step 3.   A valid legal "Transmit Last Command" mode command shall be transmitted to the UUT and a status response of ME followed by a data word containing the last command (command word from Step 2) shall be verified.
Step 4.   A valid legal "Transmit Status" mode command shall be transmitted to the UUT and a status response with a ME set shall be verified.
Step 5.   A valid legal "Transmit Last Command" mode command shall be transmitted to the UUT and a status response of ME followed by a data word containing the last command (command word from Step 4) shall be verified.
Step 6.   A valid legal "Transmit Last Command" mode command shall be transmitted to the UUT and a status response of ME followed by a data word containing the last command (command word from Step 4) shall be verified.
Step 7.   A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response verified.
Step 8.   A valid legal "Transmit Last Command" mode command shall be transmitted to the UUT and a status response of CS followed by a data word containing the last command (command word from Step 7) shall be verified.

5.2.2.3   <u>Status Word</u>: The following tests verify that all implemented status word bits are properly used and cleared. The UUT has failed a test sequence if it does not respond as indicated in each test paragraph for any implemented status word bit.

5.2.2.3.1   Service Request: This test verifies that the UUT sets the service request bit as necessary and clears it when appropriate. The UUT shall set bit time 11 of the status word when a condition in the UUT warrants the RT to be serviced. A reset of the bit shall occur as defined by each RT. The following steps shall be performed and the appropriate responses verified:

Step 1.   A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response with the service request bit reset verified.

Step 2.   A condition, which causes the service request bit to be set shall be introduced into the UUT. A valid legal receive command with at least one data word shall be transmitted to the UUT and a status response having the service request bit set shall be verified.

Step 3.   A valid legal receive command with at least one data word shall be transmitted to the UUT and a status response having the service request bit set shall be verified.

Step 4.   Procedures, as defined for the UUT, shall be performed which reset the service request bit. A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response with the service request bit reset shall be verified.

The pass criterion for this test shall be the successful completion of Steps l through 4.

5.2.2.3.2   Broadcast: This test verifies that the UUT sets the broadcast command received bit of the status word after receiving a broadcast command. The UUT shall set status bit 15 to a logic one after receiving the broadcast command. The following test sequence shall be performed using either the "Transmit Last Command" or "Transmit Status" mode command to verify the bit condition.

Step 1.   A valid legal broadcast receive message shall be transmitted to the UUT. A NR shall be verified.

Step 2.   A valid legal "Transmit Last Command" or "Transmit Status" mode command shall be transmitted to the UUT. In either case, verify BCR. If the "Transmit Last Command" mode command is used, the data word containing the broadcast command shall be verified.

Step 3.   A valid, legal, non-broadcast command shall be transmitted to the UUT and a status word with the broadcast command received bit reset shall be verified.

Step 4.   A broadcast receive message with a parity error in one of the data words shall be transmitted to the UUT. A NR shall be verified.

Page 16

5.2.2.3.2   (Continued):

Step 5.   A valid legal "Transmit Last Command" or "Transmit Status" mode command shall be transmitted to the UUT. In either case, a status word with the message error bit set shall be verified. (BCR may be set.) If the "Transmit Last Command" mode command is used, the data word containing the broadcast command shall be verified.

The pass criterion for this test shall be the successful completion of Steps 1 through 5.

5.2.2.3.3   <u>Busy</u>: This test verifies the capability of the UUT to set the busy bit of the status word. Bit time 16 of the status word shall be set when the UUT is busy.

Step 1.   Procedures as defined for the UUT shall be performed which set the busy bit.
Step 2.   A valid legal transmit command shall be transmitted to the UUT and only a status word response with the busy bit set shall be verified. The UUT shall not transmit the requested data words when the busy bit is set.
Step 3.   Procedures, as defined for the UUT, shall be performed which reset the busy bit.
Step 4.   A valid legal transmit command shall be transmitted to the UUT and a status word with the busy bit reset shall be verified. The UUT shall have responded with the correct number of data words.
Step 5.   Repeat Step 1.
Step 6.   A valid legal receive command shall be transmitted to the UUT and a status word response with the busy bit set shall be verified.
Step 7.   Repeat Step 3.
Step 8.   A valid legal receive command shall be transmitted to the UUT and a status word response with the busy bit reset shall be verified.

The pass criterion for this test shall be the successful completion of Steps 1 through 8.

5.2.2.3.4   <u>Subsystem Flag</u>: This test verifies the capability of the UUT to set the subsystem flag bit of the status word. Bit time 17 of the status word shall be set to a logic one when a subsystem fault has been determined. Prior to performing the test sequence below, a condition. which sets the subsystem flag bit, must be activated.

Step 1.   A valid legal transmit command shall be transmitted to the UUT and a status response having the subsystem flag bit set shall be verified.
Step 2.   Procedures, as defined for the UUT, shall be performed which reset the sub system flag bit.

5.2.2.3.4   (Continued):

Step 3. A valid legal transmit command shall be transmitted to the UUT and a status word with the subsystem flag bit reset shall be verified.

The pass criterion for this test shall be the successful completion of Steps 1 through 3.

5.2.2.3.5   Terminal Flag: This test verifies that the UUT sets the terminal flag bit as necessary and clears it when appropriate. The UUT shall set bit time 19 of the status word when an occurrence in the UUT causes a terminal fault condition. Prior to performing the test sequence below, a condition which sets the terminal flag bit must be activated.

Step 1. A valid legal receive command with at least one data word shall be transmitted to the UUT and a status response with the TF bit set shall be verified.
Step 2. Procedures, as defined for the UUT, shall be performed which reset the TF bit.
Step 3. A valid legal transmit command shall be transmitted to the UUT and a status word with the TF bit reset shall be verified.

The pass criterion for this test shall be the successful completion of Steps 1 through 3.

5.2.2.4   Broadcast Messages: The following tests verify that the UUT responds properly to broadcast messages.

Note: In the following test sequences, a "Transmit Last Command" mode command shall be used to verify the reception of the broadcast message by the UUT. If the "Transmit Last Command" mode command is not implemented, the "Transmit Status" mode command shall be used. If neither mode command is implemented, this test will not be performed.

5.2.2.4.1   Response to Broadcast Commands: The purpose of this test is to verify that the UUT responds properly to broadcast commands. The commands are classified into three categories. BC to RT broadcast commands broadcast mode commands, and RT to RT broadcast commands. Each category is tested separately in the following paragraphs. Use the following test sequence unless otherwise noted:

Step 1. A valid legal receive message shall be sent to the UUT.
Step 2. A valid legal broadcast message shall be sent to the UUT.
Step 3. A "Transmit Last Command" mode command shall be sent to the UUT.

5.2.2.4.1.1   <u>BC to RT Broadcast Commands</u>: All possible broadcast commands meeting the criteria of 4.4.1.1 of MIL-STD-1553B except broadcast mode commands shall be sent to the UUT. Each command word shall be followed by the proper number of contiguous valid data words.

The pass criteria are as follows: Step 1 - CS; Step 2 - NR; Step 3 - BCR and the data word contains the broadcast command sent in Step 2.

5.2.2.4.1.2   <u>Broadcast Mode Commands</u>: The purpose of this test is to verify that the UUT responds properly to implemented broadcast mode commands. This test is not intended to verify the mission aspects stated in the equipment specification. The UUT shall be tested for each mode code implemented with a subaddress field mode code indicator of all zeros and repeated with a subaddress field of all ones. The pass criteria are defined in each test paragraph.

5.2.2.4.1.2.1   <u>Broadcast Synchronize (Without Data Word)</u>: The purpose of this test is to verify that the UUT has the ability to recognize a broadcast "Synchronize" (without data word) mode command. The test sequence defined in 5.2.2.4.1 shall be used with a broadcast "Synchronize" (without data word) mode command in Step 2. The pass criteria for the above steps shall be as follows: Step 1 -CS; Step 2 - NR; Step 3 - BCR and the data word contains the broadcast command sent in Step 2.

5.2.2.4.1.2.2   <u>Broadcast Synchronize (With Data Word)</u>: The purpose of this test is to verify that the UUT has the ability to recognize a broadcast "Synchronize" (with data word) mode command. The test sequence defined in 5.2.2.4.1 shall be used with a broadcast "Synchronize" (with data word) mode command in Step 2. The pass criteria for the above steps shall be as follows: Step 1 - CS; Step 2 - NR; Step 3 - BCR and the data word contains the broadcast command sent in Step 2.

5.2.2.4.1.2.3   <u>Broadcast Initiate Self-Test</u>: The purpose of this test is to verify that the UUT has the ability to recognize the broadcast "Initiate Self-Test" mode command. The test sequence defined in 5.2.2.4.1 shall be used with a broadcast "Initiate Self-Test" mode command in Step 2. The pass criterion for each of the above steps shall be as follows: Step 1 - CS; Step 2 - NR; Step 3 - BCR and the data word contains the broadcast command sent in Step 2.

Note: Normal UUT operation may be affected during the Self-Test execution time. Detailed requirements shall be as defined by the UUT's design specification.

5.2.2.4.1.2.4 Broadcast Transmitter Shutdown and Overrides: The purpose of this test is to verify that the UUT has the ability to recognize and properly execute these broadcast mode commands. The pass criterion for each individual test is contained in one of the paragraphs below. The following sequence shall be performed for each test:

Step 1.   A valid legal command shall be sent on the primary bus to the UUT. A CS response shall be verified.

Step 2.   A valid legal command shall be sent on the selected alternate bus to the UUT. A CS response shall be verified.

Step 3.   A valid legal "Broadcast Transmitter Shutdown" mode command shall be sent to the UUT on the primary bus. A NR response shall be verified.

Step 4.   A "Transmit Last Command" mode command shall be sent on the primary bus to the UUT. A BCR response shall be verified, followed by the data word transmitted in Step 3.

Step 5.   A valid legal command shall be sent on the selected alternate bus to the UUT. A NR shall be verified.

Step 6.   A valid legal command shall be sent on the primary bus to the UUT. A CS response shall be verified.

Step 7.   A valid legal broadcast "Override Transmitter Shutdown" mode command shall be sent to the UUT on the selected alternate bus. A NR shall be verified.

Step 8.   A valid legal command shall be sent to the UUT on the selected alternate bus. A NR shall be verified.

Step 9.   A valid legal broadcast "Override Transmitter Shutdown" mode command shall be sent to the UUT on the primary bus. A NR shall be verified.

Step 10. A "Transmit Last Command" mode command shall be sent on the primary bus to the UUT. A BCR response shall be verified, followed by the data word from Step 9.

Step 11. A valid legal command shall be sent on the selected alternate bus to the UUT. A CS shall be verified.

Step 12. A valid legal command shall be sent on the primary bus to the UUT. A CS shall be verified.

5.2.2.4.1.2.4.1   Broadcast Dual Redundant Shutdowns and Overrides: This test shall verify that the UUT recognizes the dual redundant broadcast mode commands to shut down the alternate bus transmitter and to override the shutdown. In a dual redundant system, each bus must be tested as the alternate bus and as the primary bus. A valid legal broadcast "Transmitter Shutdown" mode command shall be used in Step 3. A valid legal broadcast "Override Transmitter Shutdown" mode command shall be used in Steps 7 and 9. The test sequence in 5.2.2.4.1.2.4 shall be used for each case, including verification of the UUT response indicated. The pass criteria shall be that the UUT performs as defined in each step.

5.2.2.4.1.2.4.2 <u>Broadcast Selective bus Shutdowns and Overrides</u>: This test shall verify that the UUT recognizes the multi-redundant broadcast mode commands to shut down a selected bus transmitter and to override the shutdown. In a multi-redundant system, each bus must be tested as the alternate bus and each as the primary bus. A valid legal broadcast "Selected Transmitter Shutdown" mode code shall be encoded accompanied by the appropriate data word in Step 3 to cause a selective bus transmitter shutdown. A valid legal broadcast "Override Selected Transmitter Shutdown" mode shall be encoded accompanied by the appropriate data word in Steps 7 and 9 to cause an override of the selected bus transmitter shutdown. The test sequence in 5.2.2.4.1.2.4 with the changes given shall be performed using each bus as the primary bus and each as the alternate bus, including verification of the UUT response indicated. The pass criteria shall be that the UUT performs as defined in each step.

5.2.2.4.1.2.5 <u>Broadcast Terminal Flag Bit Inhibit and Override</u>: This test verifies that the UUT recognizes and responds properly to the broadcast mode commands of "Inhibit Terminal Flag Bit" and "Override Inhibit Terminal Flag Bit". Beginning in Step 2 of the test sequence below, the UUT shall be caused to set the terminal flag bit.

Step 1. A valid legal receive command with at least one data word shall be transmitted to the UUT and a CS response verified.
Step 2. Introduce a condition that will set the terminal flag in the UUT status response. Send a valid legal receive command with at least one data word to the UUT and verify that the TF bit is set in the returned status word.
Step 3. A valid legal "Inhibit Terminal Flag" broadcast mode command shall be transmitted to the UUT and a NR verified.
Step 4. A "Transmit Last Command" mode command shall be sent to the UUT. A BCR response shall be verified, followed by the data word from Step 3.
Step 5. Repeat Step 1 and verify a CS response.
Step 6. A valid legal "Override Inhibit Terminal Flag" broadcast mode command shall be transmitted to the UUT and a NR verified.
Step 7. A "Transmit Last Command" mode command shall be sent to the UUT. A BCR shall be verified.
Step 8. A valid legal receive command with at least one data word shall be transmitted and a TF verified in the returned status word.
Step 9. Procedures, as defined for the UUT, shall be performed which reset the TF bit.
Step 10. Repeat Step 1.

The pass criteria shall be that the UUT performs as defined in each step.

5.2.2.4.1.2.6 <u>Broadcast Reset Remote Terminal</u>: This test shall verify that the UUT has the capability
to recognize the broadcast mode command to reset itself to a power-up initialized state.
For this test, the sequence in 5.2.2.4.1 shall be used with a broadcast "Reset Remote
Terminal" mode command in Step 2.

Note: Normal UUT operation may be affected during the Reset execution time. Detailed
requirements shall be as defined by the UUT's design specification.

The pass criteria shall be as follows: Step 1 - CS, Step 2 - NR; Step 3 - action and
response defined by UUT's design specifications.

5.2.2.4.1.3 <u>RT to RT Broadcast Commands</u>: The purpose of this test is to verify that the UUT
responds properly to broadcast RT to RT commands with the UUT as the receiving RT.
The following test sequence shall be used:

Step 1. A valid legal receive message shall be sent to the UUT.
Step 2. A valid legal broadcast RT to RT command pair shall be sent
to the UUT.
Step 3. A "Transmit Last Command" mode command shall be sent to the
UUT.
Step 4. Repeat Step 1.

The pass criteria shall be as follows: Step 1 - CS; Step 2 - NR from the UUT; Step 3 -
BCR and the data word contains the receive command of the RT to RT command word
pair of Step 2; Step 4 - CS.

5.2.2.5 <u>RT to RT Transfers</u>: This test is intended to verify proper UUT operation during RT to
RT transfers. For this test, the bus tester shall act as the bus controller and receiving or
transmitting remote terminal, as required.

5.2.2.5.1 <u>RT to RT Transmit</u>: A valid legal RT to RT command pair shall be sent to the UUT. The
transmit command shall be addressed to the UUT. The pass criterion shall be CS.

5.2.2.5.2 RT to RT Receive: A valid legal RT to RT command pair followed in 4 to 12 ps by a
valid status word with the RT address of the transmit command and the proper number
of data words shall be sent to the UUT. CS shall be verified.

5.2.2.5.3   <u>RT to RT Timeout</u>: The purpose of this test is to verify that the UUT functions properly when operating as the receiving RT in an RT to RT transfer. The UUT must not respond after receiving an RT to RT command pair if the data is not received within 54 to 60 µs as shown in Fig. 6. This time is measured from the zero crossing of the parity bit of the receive command to the mid sync zero crossing of the first data word. The following test sequence shall be performed:

Step 1.   A valid legal RT-to-RT command pair followed in 4 to 12 ps by a valid status word with the RT address of the transmit command and the proper number of data words shall be sent to the UUT. CS shall be verified.

Step 2.   The time between the command pair and status word of Step 1 shall be increased until the UUT (receiving RT) stops responding and the time "T" as specified in Fig. 6 shall be measured. This time shall be between 54 and 60 ps.

The pass criterion for this test shall be the successful completion of Steps 1 and 2.

5.2.2.6   <u>Illegal Commands</u>: This test shall be performed if the UUT has the illegal command detection option implemented. If this test is performed, the Word Count (5.2.1.2) and Subaddress (5.2.1.3) tests shall not be performed. The requirements are as follows:

With the UUT terminal address set for a specific code and all commands issued containing that code, perform the following tests:

Step 1.   Send a valid command to the UUT with subaddress 00001, T/R bit 0, and word count 00000, and appropriate number of data words.

Step 2.   Send any valid legal receive command with appropriate data word(s) to the UUT.

Step 3.   Repeat Step 1 for all other combinations of subaddress (except 00000 and 11111),
T/R and word counts with appropriate data words following for receive commands.

Step 4.   Repeat Step 2 following each of the commands of Step 3.

Step 5.   Send a valid illegal receive command to the UUT with a parity error in one of the data words.

Step 6.   Send a "Transmit Status" mode command to the UUT. If "Transmit Status" is not implemented, proceed to Step 7.

Step 7.   Repeat Step 2.

Step 8.   Send an illegal command to the UUT with a parity error in the command word.

Step 9.   Send a "Transmit Last Command" mode command to the UUT. If "Transmit Last Command" is not implemented, proceed to Step 10.

Step 10.   Repeat Step 2.

Step 11.   Repeat Step 1 for all combinations of T/R and mode code for subaddresses, 00000 and 11111 (mode commands) with a valid data word contiguously fol lowing each receive command.

Step 12.   Repeat Step 2 following each of the commands of Step 3.

5.2.2.6   (Continued):

The pass criteria shall be: Step 1 - CS for legal commands and status word only with ME set for illegal commands; Step 2 - CS; Step 3 - CS for legal commands and status word only with ME set for illegal commands; Step 4 - CS, Step 5 - NR; Step 6 - ME; Step 7 - CS; Step 8 - NR; Step 9 - CS and a data word containing the command word associated with Step 7; Step 10 - CS; Step 11 - status word only with ME bit set for unimplemented mode commands; for undefined mode commands, status word only with ME bit set, or NR if UUT does not respond to undefined mode commands; Step 12 - CS.
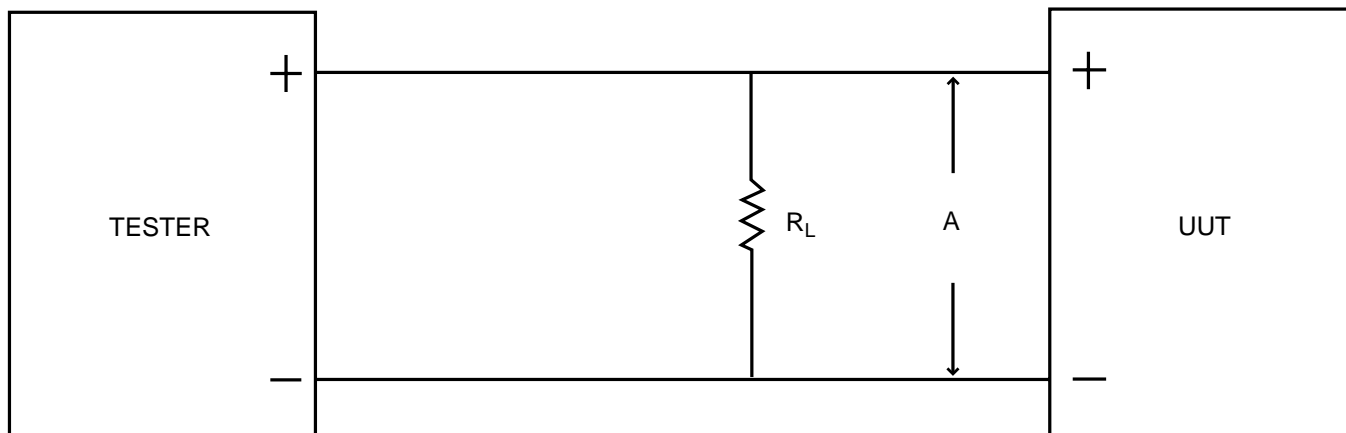
R_L	DIRECT COUPLED	TRANSFORMER COUPLED
	35 ohms ± 2%	70 ohms ± 2%

GENERAL TEST CONFIGURATION
Figure 1.
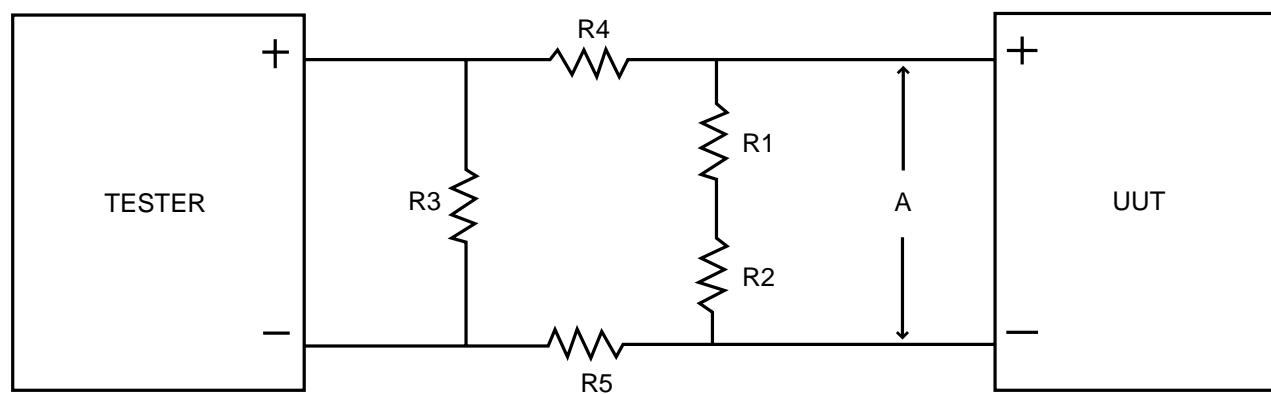
	DIRECT COUPLED	TRANSFORMER COUPLED
	35 ohms ± 2%	70 ohms ± 2%
R1, R2	20	46.5
R3, R4, R5	100	93.1

OPTIONAL TEST CONFIGURATION
Figure 2.

Page 25

T_F = FALL TIME 90% TO 10%
T_R = FALL TIME 10% TO 90%
T_T = WAVEFORM TAILOFF TIME = 2.5 ± µSEC
V_R = RESIDUAL VOLTAGE
V_D = DISTORTION VOLTAGE
V_{PP} = PEAK-TO-PEAK VOLTAGE
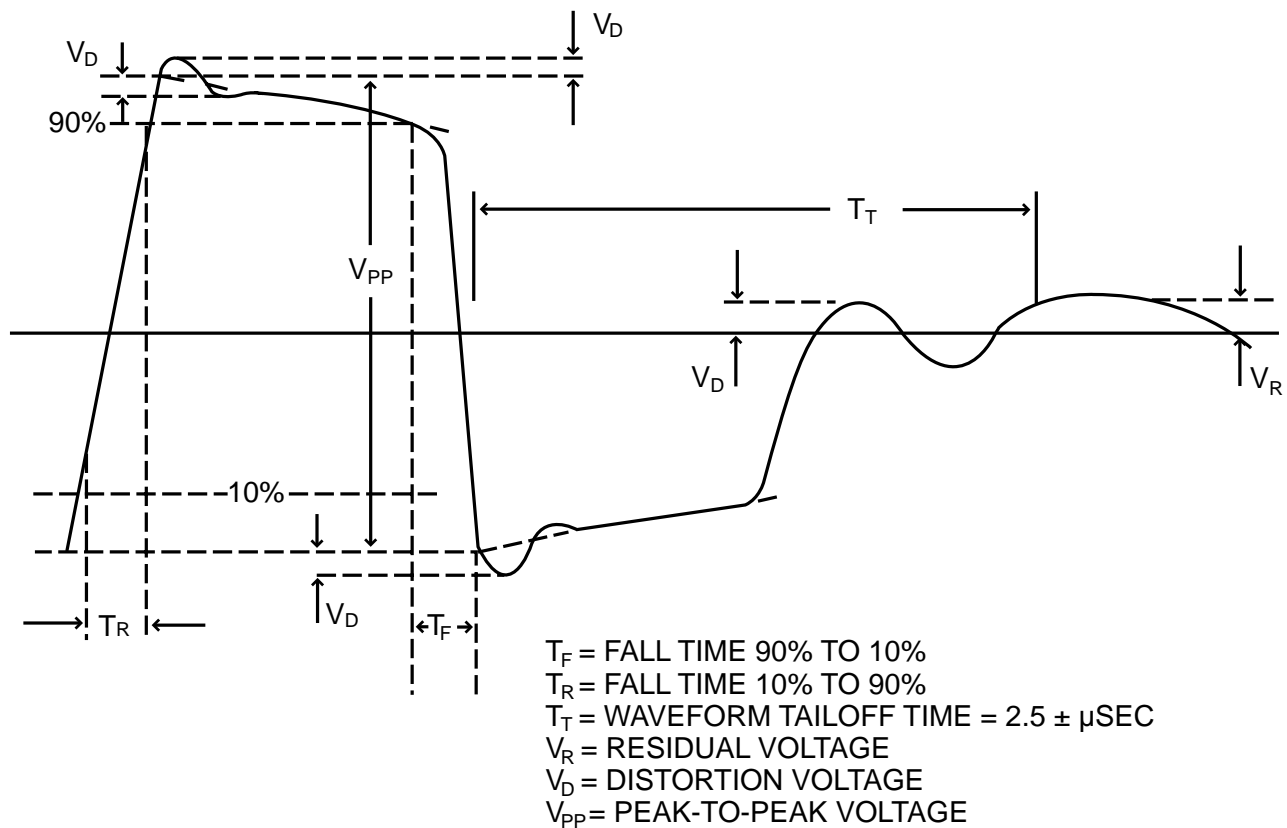
WAVEFORM MEASUREMENTS
Figure 3.
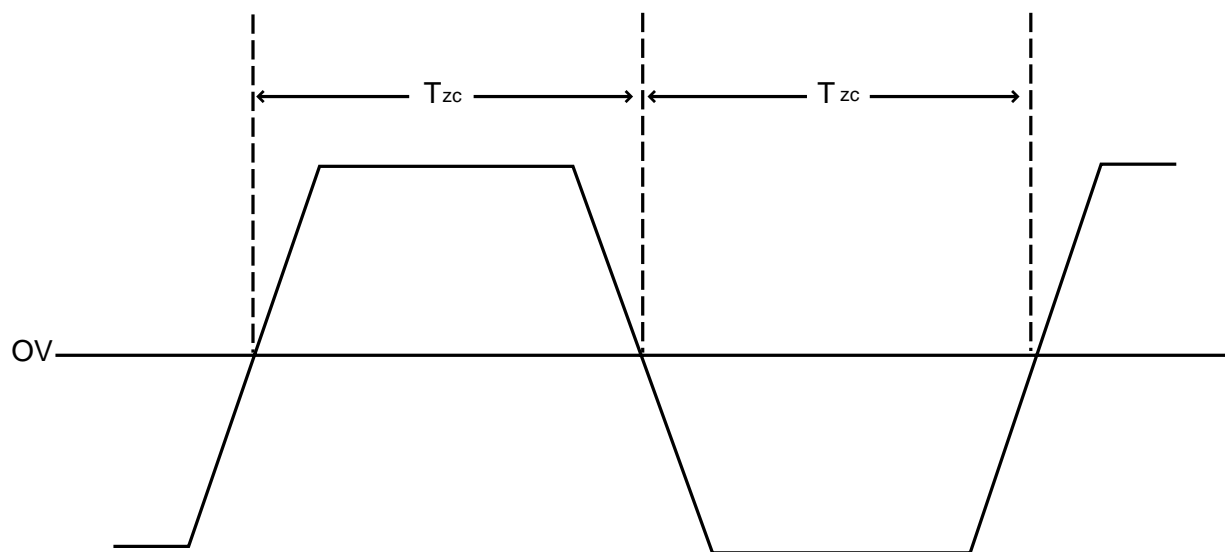
ZERO CROSSING INTERVAL MEASUREMENTS
Figure 4.

V-28

TRANSFORMER COUPLED R_L = 70.0 ohms ± 2%
DIRECT COUPLED R_L = 35.0 ohms ± 2%

OUTPUT NOISE CONFIGURATION
Figure 5.

T = 54 μsec measured as shown.
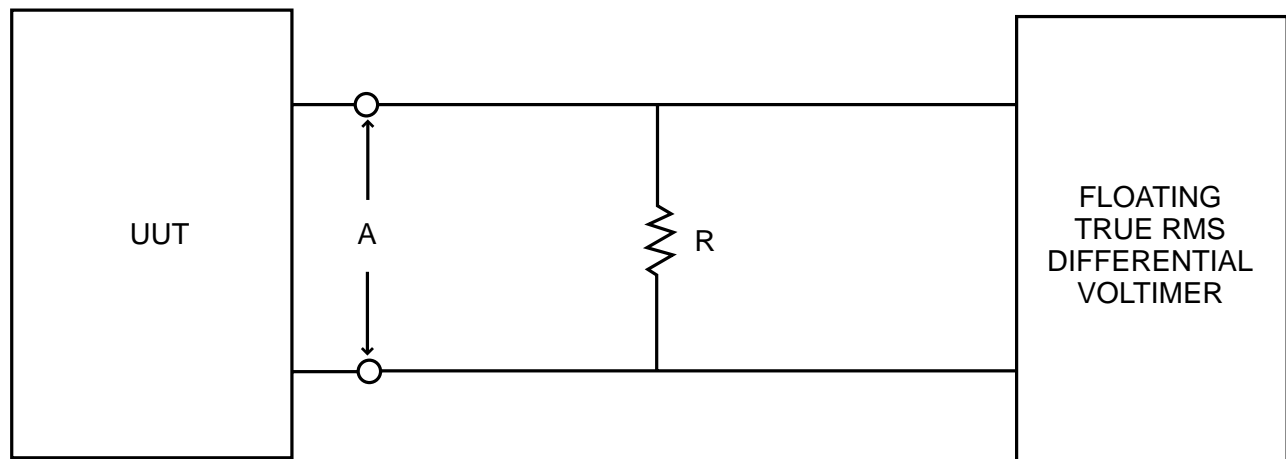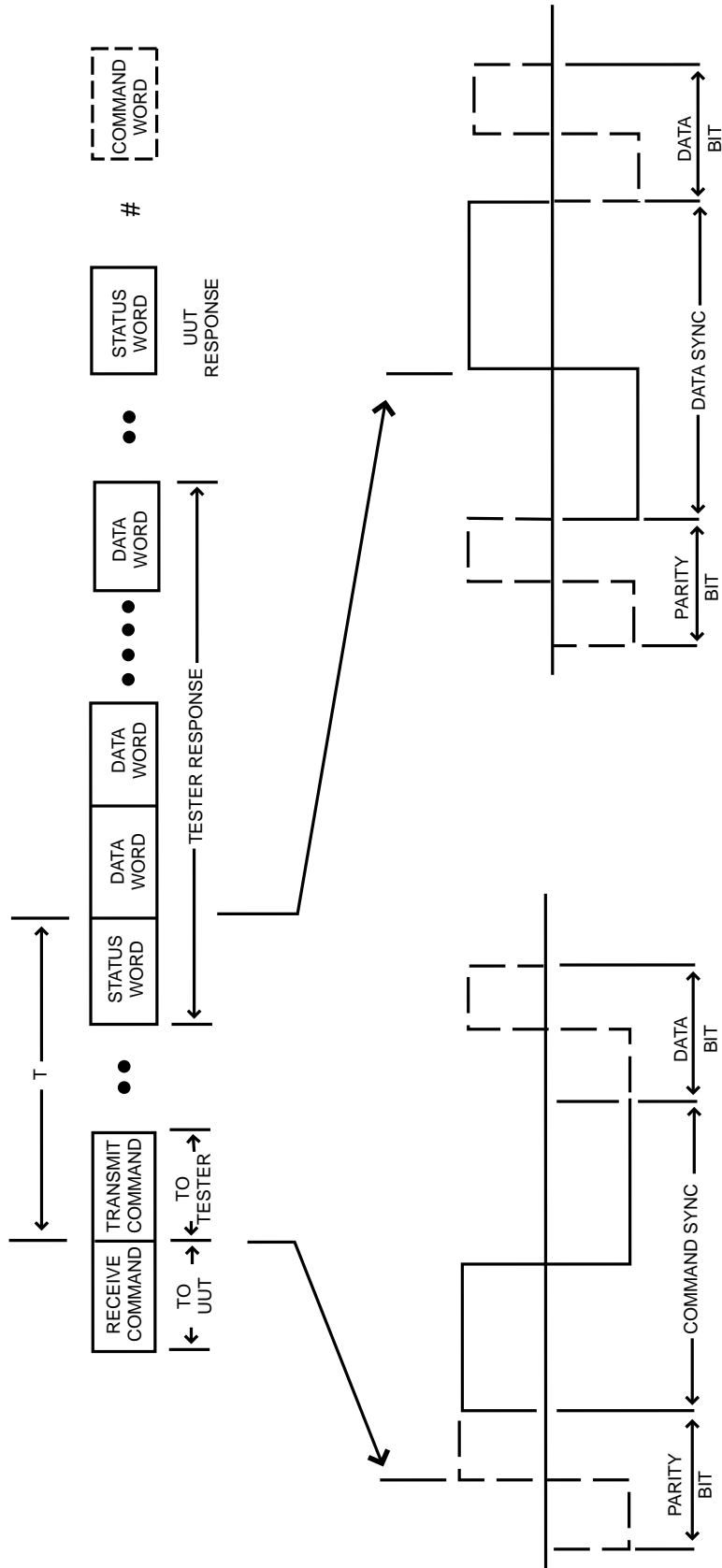**Delay tester response until UUT response goes away.**

RT — RT TIMEOUT MEASUREMENT
Figure 6.

APPENDIX A
TEST PLAN CHANGES FOR MIL-STD-1553B, NOTICE 2 RTs

For remote terminals destgned to comply with MIL-STD-1553B Notice 2, the following changes shall be made for tests or pass criteria, or both, in this document:

A1.   <u>Unique Address</u>: The purpose of this test is to verify that the UUT can be assigned a unique address from an external connector on the UUT. The following shall be performed for the UUT:

Step 1. Send a valid legal command to the UUT.
Step 2. After externally changing the RT address to simulate a single point address validation failure (for example, parity error on the address lines), repeat Step 1.

Note: Power cycling may be required after externally changing the RT address.

The pass criteria shall be: Step 1 - CS; Step 2 - NR.

A2.   <u>Mode Codes</u>: Compliance with the following paragraphs is no longer optional:

5.2.2.2.3   Transmit Status
5.2.2.2.6   Transmitter Shutdown and Override
5.2.2.2.8   Reset Remote Terminal

A3.   <u>Reset Remote Terminal</u>: The purpose of this test is to verify that the UUT has the ability to recognize the reset terminal mode command and the UUT to complete the reset function within its design time. The following sequence shall be performed once with time $T \geq T_{dr}$ and again with time $T \geq T_{dr}$ (see Step 2) where T_{dr} is the actual maximum design value for reset time. In no case shall T be greater than 5 ms.

Step 1. A reset remote terminal mode command shall be sent to the UUT on one bus.
Step 2. After time T from Step 1, as measured per Fig. A-1, a valid legal command shall be sent to the UUT on the same bus.
Step 3. A valid legal transmitter shutdown mode command shall be sent to the UUT on the same bus.
Step 4. A valid legal command shall be sent to UUT on the alternate bus.
Step 5. A reset remote terminal mode command shall be sent to the UUT on the first bus.
Step 6. After time T, repeat Step 4.

The pass criteria for each of the above steps shall be as follows: Step 1 CS; Step 2 - CS (with BUSY bit reset) for time $T \geq T_{dr}$, and CS, CS with BUSY bit or NR (whichever is the design requirement) for $T < T_{dr}$; Step 3- CS; Step 4 - NR; Step 5 - CS; Step 6 - CS.

A4.   Initiate Self-Test: The purpose of this test is to verify that the UTT has the ability to recognize the initiate self-test mode command and complete self test within its design time. The following sequence shall be performed once with time $T > T_{dt}$ and again with the time $T < T_{dt}$ (see Step 2) where T_{dt} is the actual maximum design value for self test time. In no case shall T be greater than 100 ms.

Step 1.   An initiate self-test mode command shall be sent to the UUT on one bus.
Step 2.   After time T from Step 1, as measured per Fig. A-1, a valid legal command shall be sent to the UUT on the same bus. The test shall be performed with T set to a value determined by actual design requirement.

The pass criteria for each of the above steps shall be as follows: Step 1 CS; Step 2 - CS (with BUSY bit reset) for all time T > the time used in Step 2, and CS, CS with BUSY bit set or NR (whichever is the design requirement) for T < the time used in Step 2.

A5.   Power On Response: The purpose of this test is to verify that the UUT responds correctly to commands after power is applied to the UUT. The following test sequence shall be per formed for the UUT using the normal power on sequence for the UUT.

Step 1.   Power the UUT off.
Step 2.   Send valid legal non-broadcast, non-mode commands to the UUT with a maximum intermessage gap of 1 ms.
Step 3.   Power on the UUT and monitor all the UUT responses for 2 s starting from the first transmission of the UUT after power up.

The pass criteria shall be: Step 3 - NR until the first UUT transmission, and CS for the first transmission and all responses thereafter.

A6.   Data Wrap-Around: The purpose of this test is to verify that the UUT properly implements the data wrap-around capability. The following sequence shall be performed with random data patterns for each data word. The messages used shall contain the maximum number of data words that the RT is capable of transmitting or receiving; that is, the maximum word count from the set of all messages defined for that RT.

Step 1.   Send a receive command with the appropriate number of data words to the UUT at the design requirement receive wrap-around subaddress defined for the UUT.
Step 2.   Send a transmit command to the UUT with the design requirement wrap-around subaddress and with the same word count as Step 1.

The pass criteria shall be: Step 1 - CS, Step 2 - CS with each data word having the same bit pattern as the corresponding data word in Step 1.

A7.   RT to RT Validation: The following step and pass criteria shall be added to 5.2.2.5.3:

Step 3.   Following the completion of Step 2, a Transmit Status mode command shall be
          issued to the UUT.

The pass criterion for Step 3 is "ME."

A8.   Connector Polarity: Add the following as 4.2 to this document.

4.2   Connector Polarty: When performing tests for UUT's that use concentric connectors
         or inserts for each bus, observe polarity conventions ln connecting to the UUT's
         such that the center pin of the connector or insert shall be used for the high
         (positive) Manchester bi-phase signal. The inner ring shall be used for the low
         (negative) Manchester bi-phase signal.
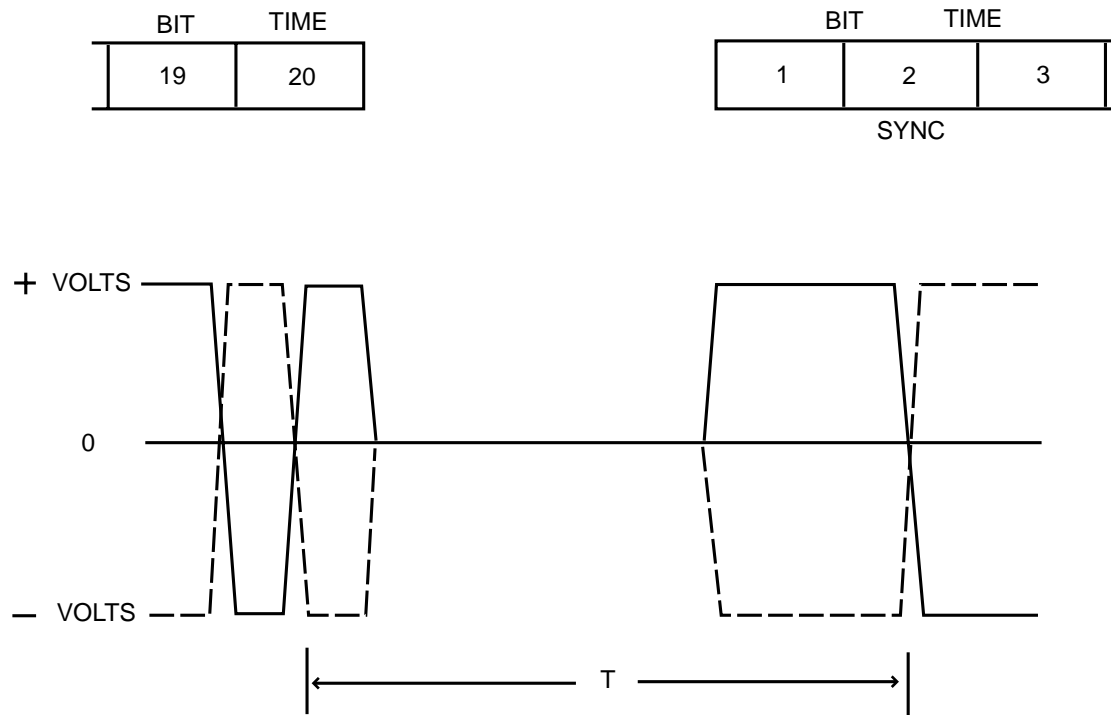
GAP TIME MEASUREMENT
Figure A-1

V-34

VI.
RT VALIDATION
TEST PLAN

CONTENTS

FIGURES

FIGURE   TITLE
PAGE

THIS PAGE INTENTIONALLY LEFT BLANK

100. RT VALIDATION TEST PLAN

Section 100 of this handbook revision supersedes Appendix A of Notice 1 for Mil-Hdbk-1553 and should be used in all applications where the previous test plan was specified. The only technical changes made are as listed below:

paragraph	revision
5.1 .2.1 .1	In second sentence "ideal" was added between "previous" and "zero". This was a clarification.
5.2.1.1.1	The first lines of items d and g were changed to: "Undefined" mode commands (See Table I): (for any one undefined mode command, any single set (1),(2),(3), or (4) is acceptable). This change was for clarification.
5.2.1.1.2	The introductory paragraph was modified to provide more clarification and alleviate misinterpretation.
5.2.1 .4.1	This is a new test: RT-RT superceding command.
5.2.1.5.1	In step 6 the alternate bus was changed to primary bus. This was an error in the original test plan.
5.2.1.5.3	This test was rewritten to better accomplish the original intent of the test.
5.2.1 .7	This was changed to an introductory paragraph. The original "RT to RT timeout" test was modified and became 5.2.1.7.1. A new test "RT to RT message format errors" was added as 5.2.1.7.2. A new test for "transmitting RT errors" was added is 5.2.1.7.3.
5.2.1.8	The pass criteria for a, step 1, under RT transmitting was changed to: "...for a step 1NR, truncated message, or CS...." This adds "NR" as permissible.
5.2.2.1.3	This test was rewritten to better accomplish the original intent of this test.
5.2.2.2.2	Step 6 and Step 7 were deleted as well as the criteria for these two steps. These steps duplicated 5.2.2.5.1.2 and inconsistently handled the BCR limit.
5.2.2.2.4	The pass criteria for step 3 was changed to CS only.
5.2.2.3	This test was rewritten.
53	The following two sentences were added to the end of this paragraph to be consistent with 5.1 and 5.2: "A unit under test (UUT) that provides transformer and direct coupled stubs shall be tested on both stubs. The noise test shall be performed on all busses for UUTs with redundant bus configurations."

APPENDIX A RT VALIDATION TEST PLAN

NOTE: THE REMAINDER OF THIS SECTION IS A REPRINT OF THE VERSION PUBLISHED AS PART OF MIL-HDBK -1553 DATED 24 SEPTEMBER 1986

FOREWORD

This section of the handbook provides a sample test plan for MIL-STD-1 553B that may serve several different purposes. This section is intended to be noncontractual when the entire MIL-HDBK-1553 is referenced in an equipment specification or SOW. In this case the test plan, as well as the rest of the handbook, provides guidance to both the DOD procuring engineer and the contractor design engineer. This section is intended to be contractual when specifically called out in a specification, SOW, or when required by a DID. If the contractor is required to submit a test plan for his RT to the government, he may remove this section from the handbook and submit a as a portion of his test plan. A better approach would be to simply reference this section. In either case, any and all contractor changes, alterations, or testing deviations to this section shall be separately listed for easy review by government personnel.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

1.0 SCOPE

1.1 <u>General.</u> This validation test plan defines the test requirements for verifying that the design of remote terminals meet the requirements of MIL-STD-1 553B "Digital Time Division Command/Response Multiplex Data Bus." A remote terminal is considered to have failed to meet the above requirements if that remote terminal fails any test or a portion of any test performed according to this test plan. Passing this test plan does not automatically mean that the remote terminal is acceptable for use by the government. The remote terminal must also meet all the requirements of MIL-STD-1553B over all the environmental EMI vibration and application requirements in the subsystem specification.

1.2 <u>Application.</u> This general test plan is intended for design verification of remote terminals designed to meet the requirements of MIL-STD-1553B Notice 2. Appendix A and B provide cross references between this test plan and MIL-STD-1 553B. For those remote terminals not required to meet Notice 2 Appendix C and D list the changes in this test plan for MIL-STD-1553B only and MIL-STD-1553B Notice 1. These requirements shall apply to the terminal under test when invoked in a specification or statement of work

2.0 APPLICABLE DOCUMENTS

2.1 <u>Standards</u>

MILITARY

MIL-STD-1553 Digital Time Division Command/Response Multiplex Data Bus

3.0 DEFINITIONS

3.1 <u>Responses.</u> The following are definitions of the responses of the RT as used in this test plan. In each case the status word must have the correct terminal address and unused status bits set to zero.

3.1.1 <u>Broadcast command received (BCR).</u> The broadcast command received bit (bit time fifteen) is set in status word (and no data words in response to a transmit status mode command or a single data word in response to a transmit last command mode command).

3.1.2 <u>Busy bit (BUSY).</u> CS with the busy bit (ten time sixteen) set in the status word and no data words.

3.1 .3 <u>Clear status (CS).</u> The status word may have the busy bit and/or service request bit set. All other status code bits in the status word must be zero and the associated message must have the proper word count.

3.1.4 <u>Dynamic bus acceptance bit (DBA).</u> CS with the dynamic bus control acceptance bit (bit time eighteen)) set in the status word.

3.1.5 <u>Service request bit (SRB).</u> CS with the service request bit (bit time eleven) set in the status word.

3.1 .6 <u>Message error bit (ME).</u> The message error bit (bit time nine) is set in the status word and no data words (except in response to a transmit last command mode command which requires on data word).

3.1.7 <u>No response (NR).</u> The addressed terminal does not produce any response to the command.

3.1.8 <u>Respond in form.</u> A terminal is said to "respond in form" if its response to an illegal command is defined in the paragraph titled "Illegal command. of MIL-STD-1553 consists of a response formatted as though it were a legal command.

3.1.9 <u>Subsystem flag bit (SF).</u> CS with the subsystem flag bit (bit time seventeen) set in the status word.

3.1.10 <u>Terminal flag bit (TF).</u> CS with the terminal nag bit (bit time nineteen) set in the status word.

100-7

4.0 GENERAL REQUIREMENTS

4.1 General test requirements. The following paragraphs define the configurations, pass/fail criteria, and general procedures for testing remote terminals (RT). Specifically, this document contains the test configurations and procedures for the Electrical tests (5.1), the Protocol Tests (5.2), and the Noise Rejection Test (5.3) for MIL-ST-1553 remote terminals. The remote terminal under test is referred to as the unit under test (UUT). Proper terminal responses are defined in each test paragraph. If the hardware/software design of the UUT does not permit a test to be performed, then adequate analysis shall be provided in place of the test results to demonstrate that the design meets the requirements of MIL-STD-1553 as stated in the test.

Any condition which causes the UUT to respond other than as called out in MIL-STD-1553, to lock up, or require a power cycle in order to recover for any reason shall automatically cause that UUT to fail the test. All occurrences of responses with the busy bit' set in the status word shall be recorded. If the U UT response does not match the pass criteria for a particular test, then the UUT has failed that test.

4.2 Test for optional requirements. All tests for optional requirements defined in 5.2.2 shall be executed if that MIL-STD-1553 option is required by the subsystem specification or interface control document (ICD) . Any optional capabilities implemented in the RT should also be tested, if possible. Within the constraints imposed by the hardware/software design, optional capabilities must be tested prior to use by system integrators.

4.3 General monitoring requirement. In addition to the specific tests that follow, certain RT parameters must be continuously monitored throughout all tests. These parameters are:

a.   response time
b.   contiguous data
c.   proper Manchester encoding
d.   proper bit count
e.   odd parity
f.   proper word count
9.   proper terminal address in the status word
h.   reserved status and instrumentation bits in the status word are set to zero
L   proper sync

The UUT shall have failed the test if at any time during the test any of these parameters fail 10 meet the requirements of MIL-STD-1553. Record the parameters for all failures.

5.0 DETAILED REQUIREMENTS

5.1 Electrical tests. Each test paragraph contains the requirements for both transformer and direct coupled stubs. A UUT which provides both transformer and direct coupled stubs must be tested on both stubs. Electrical tests shall be performed on all buses for UUTs with redundant bus configurations.

5.1.1 Output characteristics. The following tests are designed to verify that all UUT output characteristics comply with MIL-STD-1553. These tests shall be performed after establishing communications between the test equipment and the UUT. All output electrical tests shall use figure 1A, General Resistor Pad Configuration, with all measurements taken at point "A" unless otherwise noted.

5.1.1.1 Amplitude. A valid, legal transmit command shall be sent to the UUT, requesting the maximum number of words that it is capable of sending. The amplitude of the waveform transmitted by the UUT shall be measured, peak-to-peak, as shown on figure 2.

The pass criteria for Vpp for transformer coupled stubs shall be 18.0 V minimum, and 27.0 V maximum. The pass criteria for Vpp for direct coupled stubs shall be 6.0 V minimum and 9.0 V maximum. The maximum and minimum measured parameters, Vpp, shall be recorded.

5.1.1.2 <u>Risetime/falltime.</u> A valid, legal transmit command shall be sent to the UUT, requesting at least one data word. The rise and fall time of the UUT waveform shall be measured between the 10% and 90% points of the waveform as shown on figure 2. The measurements shall be taken at both the rising and falling edges of a sync waveform and a data bit waveform. The risetime (Tr) and the falltime (Tf) shall be recorded.

The pass criteria shall be 100 ns _ Tr   300 ns and 100 ns Tf   300 ns. The measured parameters, Tr and Tf, shall be recorded.

Note: The risetime of the sync waveform shall be measured at the mid-crossing of a data word sync, and the fall time of the sync waveform shall be measured at the mid-crossing of the status word sync.

5.1.1.3 <u>Zero crossing stability.</u> A valid legal transmit command shall be sent to the UUT, requesting the UUT to transmit words having zero crossing time intervals of 500 ns, 1000 ns, 1500 ns and 2000 ns. The zero crossing time shall be measured for both the positive (Tzcp) and the negative (Tzcn) waveforms as shown on figure 3.

The pass criteria for each case shall be that Tzcp end Tzcn = 500 ±25ns, 1000 1500 ± 25 ns and 2000 ± 25 ns. The measured parameters, Tzcp and Tzcn shall be recorded for each case.

5.1.1.4 <u>Distortion, overshoot and ringing.</u> A valid legal transmit command shall be sent to the UUT, requesting the UUT to transmit at least one data word. The distortion of the waveform, distortion voltage (VD) shall be measured as indicated on figure 2.

Pass criteria shall be VD ± 900 mV peak, line-to-line, for transformer coupled stubs or VD ±300 mV peak, line-to-line, for direct coupled stubs. The worst measured parameter, VD, shall be recorded.

5.1.1.5 <u>Output symmetry.</u> A valid legal transmit command shall be sent to the UUT requesting the maximum number of data words that the UUT is capable of transmitting. The output symmetry is determined by measuring the waveform tail-off at the end of each message. The maximum residual voltage (Or) shall be measured as shown on figure 2. This test shall be run six times with each data word in the message having the same bit pattern. The six data word bit patterns that shall be used are:

8000(HEX),7FFF(HEX), 0000(HEX)), FFFF(HEX), 5555(HEX), AND AAAA(HEX

The pass criteria shall be V4 ± 250 mV peak, line-to-line, for transformer coupled stubs and Vr ±90 mV peek, line-to-line, for direct coupled stubs after time Tt (the time beginning 2.5 µs after the mid-bit zero crossing of the last parity ten). The measured parameter, Vr, shall be recorded for each bit pattern.

5.1.1.6 <u>Output noise</u>. The test configuration shown on figure 4 shall be used to test the UUT inactive bus output noise levels. The test shall be conducted while the UUT is in the power-on receive state and the power off state. The output noise (Vrms) shall be measured at point "A" as shown on figure 4 for both states. Measurements shall be made with an instrument that has a minimum frequency bandwidth of DC to 10 MHz.

The pass criteria shall be Vrms   14.0 mV for transformer coupled stubs and Vrms   5.0 mV for direct coupled stubs. The measured parameters, Vrrns, shall be recorded for each case.

5.1.1.7 <u>Output Isolation.</u> This test shall be performed only if the UUT is configured with redundant buses. A valid legal transmit command shall be sent to the UUT requesting the maximum number of data words that it is capable of sending. The voltage of the output waveform transmitted by the UUT shall be measured on

100-9

the active and redundant bus (or buses). Each data bus shall be alternately activated and measurements taken.

The pass criteria shall be that the ratio in dB between the output peak-to-peak voltage on the active bus and the output peak-to-peak voltage on all inactive buses shall be greater than or equal to 45 dB (figure 5). The measured parameters, output isolation, expressed as a ratio in dB, shall be recorded for each bus combination.

5.1.1.8 <u>Power on/off</u>

5.1.1.8.1 <u>power on/off noise.</u> A UUT shall limit any spurious differential output during a power-up or power-down sequence. Power shall be applied to the UUT and any outputs from the U UT shall be measured. Power shall be removed from the UUT and any output from the UUT shall be measured. Repeat the test ten tunes.

The pass criteria shall be:

a.   For transformer coupled stubs any spurious noise pulses produced shall be less than or equal to ±250 mV peak, line-to-line.

b.   For direct coupled stubs any spurious noise pulses produced shall be less than or equal to ±90 mV peak, line-to-line .

All measured parameters, output noise amplitudes and pulse widths, shall be recorded.

Note: This test shall be performed using the normal on/off power sequence of the UUT.

5.1.1.8.2 <u>Power on response</u>. The purpose of this test is to verify that the UUT responds correctly to commands after power is applied to the U UT. Using the normal power on sequence for the UUT, repeat the following test sequence a minimum of ten times.

Step 1.   Power the UUT off.

Step 2.   Send valid, legal, non-broadcast, non-mode commands to the UUT with a maximum intermessage gap of 1 ms.

Step 3.   Power on the UUT and observe all the responses for a minimum of 2 s from the first transmission of the UUT after power on.

The pass criteria shall be: step 3 - NR until the first UUT transmission, and CS for the first transmission and ail responses thereafter.

5.1.1.9 <u>Terminal response time.</u> The purpose of this test is to verify that the UUT responds to messages within the proper response time. The test sequence shown below shall be performed:

Step 1.   A valid legal transmit command shall be sent to the UUT and the response time measured.

Step 2.   A valid legal receive command shall be sent to the UUT and the response time measured.

Step 3.   A valid legal RT-to-RT command, with the UUT as the receiving terminal, shall be sent to the UUT and the response time measured.

Step 4.   A valid legal mode command shall be sent to the UUT and the response time measured.

The pass criteria for step 1, step 2, step 3, and step 4 shall be a response time between 4.0 and 12.0 µs at point A of figure 1A and measured as shown on figure 7. The command words used and the response times shall be recorded.

5.1.1.10 <u>Frequency stability.</u> The purpose of this test is to verify that the transmitter clock in the UUT has the proper accuracy and long term stability and proper short term stability. The transmitter clock measured shall be either the main oscillator output or an appropriate derivative of that clock (e.g., either the 16 MHz oscillator or the 1-2 MHz transmitter shift clock). The test sequence shown below shall be performed on the clock output whose ideal frequency is Fi.

Step 1.   The short term transmitter clock frequency shall be measured for a single period of the waveform.

Step 2.   Repeat step 1 for at least 10,000 samples and record the minimum (Fsmin) and the maximum (Fsmax) frequency from the samples taken.

Step 3.   The transmitter clock frequency shall be measured with a gate time of 0.1 s and the mean frequency for at least 1,000 samples (Fav) shall be recorded.

The pass criteria shall be:

Step 1 and step 2 -Ss1 = 100 (Fsmax - Fav)/VFav   0.01 and Ss2 = 100 (Fav-Fsmin)/Fav   0.01;

Step 3 - the magnitude of S1 = 100 (Fav-Fi)/Fi   0.1. Record Ss1, Ss2 and S1.

5.1.2 <u>Input characteristics</u>. The input tests are designed to verify that multiplex devices can properly decode bi-phase data. All input electrical tests shall use figure 1A or figure 1B with all measurements taken at point "A," unless otherwise noted. For Air Force applications, all input electrical tests shall use figure 1B, with all measurements taken at point "A" unless otherwise noted.

5.1.2.1 <u>Input waveform compatibility.</u>

5.1.2.1.1 <u>Zero crossing distortion.</u> A legal valid receive message shall be sent to the UUT and the proper response verified. Positive and negative zero crossing distortions equal to N ns, with respect to the previous ideal zero crossing shall be introduced individually to each zero crossing of each word transmitted to the UUT. The transmitted signal amplitude at point "A" shall be 2.1 Vpp for transformer coupled stubs and 3.0 Vpp for direct coupled stubs. The rise and fall time of the transmitted message (measured at a data bit zero crossing with the prior zero crossing and the next zero crossing at 500 ns intervals from the measured zero crossing) measured at point "A" shall be 200 ns ±20 nS. Each zero crossing distortion shall tee transmitted to the UUT a minimum of 1000 times.

The pass criteria is the transmission of a CS by the UUT for each zero crossing distortion sent with N 150 ns. Positive and negative zero crossing distortions shall then be applied in turn to a single zero crossing and adjusted to determine the values at which the first NR of the UUT occurs; these values shall be recorded.

The fail criteria is the transmission of a NR by the UUT for any zero crossing distortion sent with N   150 ns.

5.1.2.1.2 <u>Amplitude variations</u>. A legal valid receive message shall be sent to the UUT. The transmit voltage, as measured red at point "A" of figure 1 A or figure 1 B, shall be decremented from 6.0 Vpp to 0.1 Vpp for transformer coupled stubs and from 9.0 Vpp to 0.1 Vpp for direct coupled stubs in steps no greater than 0.1 Vpp. The rise and fall time of the transmitted message (measured at a data bit zero crossing with the prior zero crossing and the next zero crossing at 500 ns intervals from the measured zero crossing) measured at point "A" shall be 200 ns ±20 ns. The response of the UUT shall be observed at each step. A minimum of 1000 messages shall be transmitted for each setting.

100-11

The pass criteria shall be:

a. A CS for 0.86   Vpp   6.0 for transformer coupled stubs and 1.2   Vpp   9.0 for direct coupled stubs

b. A NR for Vpp   0.20 for transformer coupled stubs and Vpp   0.28 for direct coupled stubs

The measured parameter, Vpp, at which NR first occurs shall be recorded.

5.1.2.1.3 Rise and fall time.

5.1.2.1.3.1 Trapezoidal. A minimum of 1000 valid receive messages shall be sent to the UUT with a signal amplitude of 2.1 Vpp for the transformer coupled stub and 3.0 Vpp for the direct coupled stub The rise and fall times of the signal shall be less than or equal to 100 ns.

The pass criteria shall be CS by the UUT for each message.

5.1.2.1.3.2 Sinusoidal. A minimum of 1000 valid receive messages shall be sent to the UUT with a signal amplitude of 2.1 Vpp for the transformer coupled stub and 3.0 Vpp for the direct coupled stub. The rise and fall times of the signal shall approximate that of a 1 MHz sinusoidal signal.

The pass criteria shall be CS by the UUT for each message.

5.1.2.2 Common mode rejection. The common mode test configuration, figure 6A or figure 6B, shall be used for this test. Legal valid receive messages with the UUT's maximum word count shall be sent to the UUT at a repetition rate which generates a bus activity duty cycle of 50% ±1 0% with a common mode voltage injected at point "C" and the UUT response observed. The voltage of the transmitted message measured at point "A" shall be .86 Vpp for transformer coupled stubs and 1.2 Vpp for direct coupled stubs. The rise and fail time of the transmitted message (measured at a data bit zero crossing with the prior zero crossing and the next zero crossing at 500 ns intervals from the measured zero crossing) measured at point "A" shall be 200 ns +20 ns. The following common mode voltage levels shall be applied in turn: +10.0 V.D.C. line-to-ground, -10.0 V.D.C. Line-to-ground and ±10 Vp line-to-ground sinusoidal signal that is swept through the range of 1 Hz to 2 MHz. Each test condition shall be present for a minimum time period of 90 seconds.

The pass criteria shall be a CS by the UUT for all messages at each setting. If a failure occurs, the measured parameters, common mode signal injected shall be recorded.

5.1.2.3 Input impedance. The input impedance of the U UT in a stand alone configuration (i.e., disconnected from figure 1A or 1 B) shall be measured with the UUT power on and with the UUT power off. The input impedance, Zin, shall be measured with a sinusoidal waveform having an amplitude 1.0 VRMS to 2.0 VRMS, at the following frequencies: 75.0 kHz, 100.0 kHz, 250.0 kHz, 500.0 kHz and 1.0 MHz.

The pass criteria shall be Zin   1000 ohms for transformer coupled stubs and Zin   2000 ohms for direct coupled stubs. The measured parameter, Zin, shall be recorded at each frequency.

5.2 Protocol tests. All tests in this section shall use the test configuration as shown on figure 1A or figure 1B. The test signal amplitude shall be 3.0 Vpp ±0.1 Vpp for direct coupled stubs and 2 1 V pp ±0.1 Vpp for transformer coupled stubs measured at point A. For UUTs having both direct and transformer coupled stubs, the protocol tests need only be performed on one stub type per bus. The protocol tests shall be performed on an buses for UUTs with redundant bus configurations.

5.2.1 Required remote terminal operation. The following tests verify required operations of a remote terminal.

100-12

5.2.1.1 <u>Response to command words</u>. The purpose of this test is to verify that the UUT responds properly to all commands.

5.2.1.1.1 <u>RT response to command words</u>. All possible command words (65, 536 combinations) meeting the criteria of the paragraph on "Word validation" of MIL-STD-1553 shall be sent to the UUT. Mode commands tested in 5.2.1.5, 5.2.2.1, or 5.2.2.4 may be omitted from this test since they are tested separately. Each command word shall be followed by the proper number of contiguous valid data words as defined in the paragraph on "Message formats. of MIL-STD-1553. Refer to table I for undefined mode commands. The associated data may be either random or controlled, depending on the UUT requirements. The following sequence shall be executed for all combinations of command words where the varying command word is sent as step 2.

Step 1. Send a valid legal non-broadcast non-mode command to the UUT.

Step 2. Send the variable command word to the UUT.

Step 3. Send a transmit last command mode command to the UUT. (If this mode command is not implemented, then transmit status mode command shall be used and the data word associated with transmit last command mode command shall be deleted from the pass criteria.)

The pass criteria given below is contingent on the type of command sent. All commands which cause the UUT to fail shall be recorded.

Non-Broadcast Commands (including mode commands):

a. Valid legal commands: step 1- CS; step 2-CS; step 3-CS and the data word contains the command word bit pattern from step 2 (except for transmit last command mode command where the data word contains the command word bit pattern from step 1).

b. Valid illegal commands:

    (1) If illegal command detection option is implemented: step 1-CS; step 2-ME with no data words; step 3-ME and the data word contains the command word bit pattern from step 2.

    (2) If the illegal command detection option is not implemented: step 1-CS; step 2-CS; step 3-CS and the data word contains the command word bit pattern from step 2.

c. Invalid command (wrong RT address): step 1-CS; step 2-NR; step 3-CS and the data word contains the command word bit pattern from step 1.

d. Undefined mode commands (see table 1) (for any one undefined mode command, any single set (1), (2), (3), (4), is acceptable):

    (1) step 1-CS; step 2-CS; step 3-CS and the data word contains the command word bit pattern addressed to the UUT from step 2.

    (2) step 1-CS; step 2-ME; step 3-ME and the data word contains the command word bit pattern addressed to the UUT from step 2.

    (3) step 1-CS; step 2-NR; step 3-CS and the data word contains the command word bit pattern addressed to the UUT from step 1.

    (4) step 1-CS; step 2-NR; step 3-ME and the data word contains the command word bit pattern addressed to the UUT from step 2.

T/R	MODE CODE	ASSOCIATED DATA WORD
0	00000	No
0	01111	No
0	10000	Yes
0	10010	Yes
0	10011	Yes
1	10001	Yes
1	10100	Yes
1	10101	Yes

Broadcast Commands (including mode commands):

e. If there are any broadcast commands that are considered as valid commands:

   (1) Legal commands: step 1 -CS; step 2-NR; step 3-BCR and the data word contains the commands word bit pattern from step 2.

   (2) illegal commands (if illegal command detection is implemented): step 1-CS; step 2-NR; step 3BCR and ME and the data word contains the command word bit pattern from step 2.

   (3) Illegal commands (if illegal command detection is not implemented): step 1-CS; step 2-NR; step 3BCR and the data word contains the command word bit pattern from step 2.

f. If there are no broadcast commands that are considered as valid commends: step 1-CS; step2-NR; step 3-CS and the data word contains the command word bit pattern from step 1.

g. Undefined broadcast mode commands (see table 1) (for any one undefined mode command, any single set (1), (2), (3) is acceptable):

   (1) step 1-CS; step 2-NR; step 3-BCR and the data word contains the command word bit pattern from step 2.

   (2) step 1-CS; step 2-NR; step 3-ME and BCR and the data word contains the command word bit pattern from step 2.

   (3) step 1-CS; step 2-NR; step 3-CS and the data word contains the command word bit pattern from step 1.

5.2.1.1.2 <u>RT-RT response to command words</u>. All possible command words (65,536 combinations) meeting the criteria of the paragraph on "Word validation: of MIL-ST-1553 shall be sent to the UUT embedded in an RT-RT message format. The test equipment shall supply the required responses for the other RT in order to properly complete the message formats as defined in paragraph on "Message format" of MIL-STD 1553. Refer to Table I for undefined mode commands. The associated data maybe either random or controlled, depending on the UUT requirements. The intent of this test is to send all combinations of RT to RT command pairs with a fixed receive command addressed to the UUT and a variable transmit command, which includes all combinations of command words with the T/R bit equal to one and terminal address field different from the UUT's, to which the status response and data is simulated. Then send all combinations

of RT to RT command pairs with a fixed transmit command addressed to the UUT and a variable receive command which includes all combinations of command words having the T/R bit equal to zero and the terminal address field different from the UUT s to which the status response is simulated. Note that this test includes 1024 transmit commands to RT address 31 (broadcast) which are invalid messages. Mode commands tested in 5.2.1.5 5.2.2.1 or 5.2.2.4 may be omitted from this test since they are tested separately. The following sequence shall be executed for all combinations of command words where the varying command word is sent as step 2.

Step 1.   Send a valid non-broadcast non-mode command to the UUT.

Step 2.   Send the variable command word to the UUT embedded in the RT-RT message format.

Step 3.   Send a transmit last command mode command to the UUT. (If this mode command is not implemented then a transmit status mode command shall be used and the data word associated with transmit last command mode command shall be deleted from the pass criteria.)

The pass criteria shall be as listed in 5.2.1.1.1 except the pass criteria for any RT-RT mode command is as specified in 5.2.1.1.1.d and the pass criteria for any broadcast RT-RT mode command shall be as specified in 5.2.1.1.1.g.

5.2.1.2 <u>Intermessage gap.</u>

5.2.1.2.1 <u>Minimum time.</u> The purpose of this test is to verify that the UUT responds properly to messages with a minimum intermessage gap. The message pairs listed in table 11 shall be sent to the UUT with the minimum intermessage gaps as defined in the paragraph on "intermessage gap" of MIL-STD-1553. Each message pair shall be sent to the UUT a minimum of 1000 times. Message pairs which include commands not implemented by the UUT shall be deleted from the test. Each message pair shall have an intermessage gap time (T) of 4.0 μs as shown on figure 7.

The pass criteria for this test is CS for each message. All message pairs used shall be recorded and message pairs which cause the UUT to fail the test shall be indicated.

5.2.1.2.2 <u>Transmission rate.</u> The purpose of this test is to verify that the UUT responds properly to messages sent for a sustained period with a minimum intermessage gap. The message listed in each step shall be sent with an intermessage gap of 7 μs ±3 μs i.e. a burst of messages with an intermessage gap of 7 μs ±3 μs between each message as shown on figure 7. Each step shall be performed for a minimum of 30 s.

Step 1.   A valid legal transmit message followed by a valid legal transmit message.

Step 2.   A valid legal receive message followed by a valid legal receive message.

Table II. Intermessage gap messages.

COMMAND TYPES

A)   BC to UUT Transfer (maximum word count)
B)   UUT to BC Transfer (maximum word count)
C)   UUT/RT (maximum word count)
D)   RT/RT (maximum word count)
E)   Mode Command Without Data Word
F)   Mode Command With Data Word (Transmit)
G)   Mode Command With Data Word (Receive)
H)   BC to UUT Transfer (Broadcast) (maximum word count)

I)   UUT/RT (Broadcast) (maximum word count)
J)   RT/UUT (Broadcast) (maximum word count)
K)   Mode Command Without Data Word (Broadcast)
L)   Mode Command With Data Word (Broadcast)

MESSAGE PAIRS

1)   A (GAP) A
2)   B (GAP) A
3)   C (GAP) A
4)   D (GAP) A
5)   E (GAP) A
6)   F (GAP) A
7)   G (GAP) A
8)   H (GAP) A
9)   I (GAP) A
10)   J (GAP) A
11)   K (GAP) A
12)   L (GAP) A

Note: This table defines the types end combinations of messages to be used infest 5.2.1.2, e.g., pair number 2 specifies a transmit command with the maximum word count to be followed (after the minimum intermessage gap time specified in the paragraph on   "Intermessage gap" of MIL-ST-1553) by a receive command with the maximum word count.

UUT/RT: denotes RT to RT transfer with UUT receiving
RT/UUT: denotes RT to RT transfer with UUT transmitting

Step 3.   A valid legal transmit message followed by a valid legal receive message.

The pass criteria for this test is a CS for each message. All messages which cause the UUT to fail the test shall be recorded.

Note: If the busy bit gets set, then increase the intermessage gap until the busy bit is reset. At this time record the intermessage gap and repeat steps 1 thru 3 until the test is completed without the busy bit getting set.

5.2.1.3 Error Injection. The purpose of these tests is to examine the UUT's response to specific errors in the message stream. Unless otherwise noted, the following test sequence shall be used for all error injection tests. The error to be encoded in step 2 for a given message is specified in each test paragraph.

Test sequence:

Step 1.   A valid legal message shall be sent to the UUT. A mode command shall not be used.

Step 2.   A legal message containing the specified error shall be sent to the UUT.

Step 3.   A transmit status mode command shall be sent to the UUT.

The pass criteria is defined in each test paragraph. All commands and responses shall be recorded.

5.2.1.3.1 Parity. The purpose of these tests is to verify the UUT's capability of detecting parity errors embedded in different words within a message.

5.2.1.3.1.1 <u>Transmit Command Word.</u> This test verifies the ability of the UUT to detect a parry error occurring in a transmit command word. The test sequence as defined in 5.2.1.3 shall be performed with a parity error encoded into a transmit command word for test step 2.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.1.2 <u>Receive Command Word.</u> This test verifies the ability of the UUT to recognize a parry error occurring in a receive command word. The test sequence as defined in 5.2.1.3 shall be performed with a parity error encoded in a receive command word for test step 2.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.1.3 <u>Receive Data Words.</u> This test verifies the ability of UUT to recognize a parity error occurring in a data word. The test sequence as defined in 5.2.1.3 shall be performed with a parity error encoded in a date word for test step 2. The message shall be a receive command with the maximum number of date words that the UUT is designed to receive. The test sequence must be sent N times where N equals the number of data words sent.

Individually each data word must have the parity bit inverted. Only one parity error is allowed per message.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.2 <u>Word Length.</u> This test verifies the ability of the UUT to recognize transmit command word length errors. The test sequence as defined in 5.2.1.3 shall be performed with the command word shortened as defined below for test step 2.

a. Transmit command shortened by one bit

b. Transmit command shortened by two bits

The pass criteria for this test shall be: step 1-CS step 2-NR; step 3-CS.

5.2.1.3.2.2 <u>Receive Command Word.</u> This test verifies the ability of the UUT to recognize receive commend word length errors. The test sequence as defined in 5.2.1.3 shall be performed with the command word as defined below for test step 2.

a. Shorten the receive command word by one bit

b. Shorten the receive command word by two bits

c. Lengthen the receive command word by two bits

d. Lengthen the receive command word by three bits

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS or alternately for c and d only the pass criteria may be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.2.3 <u>Receive Data Words.</u> this test verifies the ability of the UUT to recognize data word length errors. The test sequence as defined in 5.2.1.3 shall be performed as defined below for test step 2. The message shall be a receive command with the maximum number of data words that the UUT is designed to received.

a. Shortened the data word by one bit

b. Shortened the data word by two bits

100-17

c.   Lengthen the data word by two bits

d.   Lengthen the data word by three bits

The test sequence of 5.2.1.3 shall be performed N times for a and b and N-1 times for c and d, where N equals the number of data words sent. High bit errors shall not be tested in the last data word of a receive message. Only one data word shall be altered at a time. Steps a through d shall be performed for each data word in the message.

The pass criteria for this test shall be: stop 1-CS; step 2-NR; step 3-ME.

5.2.1.3.3 <u>Bi-Phase Encoding</u>. This test verifies the ability of the UUT to recognize bi-phase errors. A biphase encoding error is defined to be the lack of a zero crossing in the center of a bit time. A bi phase error occurs as either a logic high or low for the duration of a bit time. Each bit location, except the sync period of each word shall have a single bi-phase error encoded into it. Only a single bi-phase error shall be injected for each message.

5.2.1.3.3.1 <u>Transmit Command Word</u>. This test verifies the ability of the UUT to recognize bi-phase encoding errors in transmit command words. The test sequence as defined in 5.2.1.3 shall be performed with a bi-phase encoding error encoded into a transmit command word for test step 2. Each bit location shall have each of the bi-phase errors injected into it. Only one bi-phase error is allowed per command word. A test set involves performing the test sequence 17 times, once for each bit location. A complete test requires two test sets to be performed, one for injecting high bi-phase errors and another for injecting low bi-phase errors.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.3.2 <u>Receive Command Word</u>. This test verifies the ability of the UUT to recognize bi-phase encoding errors in receive command words. The test sequence as defined in 5.2.1.3 shall be performed with a bi-phase error encoded into a receive command word for test step 2. Each bit location must have each of the bi-phase errors injected into it. Only one bi-phase error is allowed per commend word. A test set involves performing the test sequence 17 times, once for each bit location. A complete test requires two test sets to be performed one for injecting high bi-phase errors and another for injecting low bi phase errors.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.3.3 <u>Receive Data Words</u>. This test verifies the ability of the UUT to recognize bi-phase encoding errors in data words. The test sequence as defined in 5.2.1.3 shall be performed with a bi-phase error encoded into each data word in the message for test step 2. The message shall be a receive command and the maximum number of data words that the UUT is designed to receive. Individually each bit location of each data word shall have a bi-phase error encoded into it. Only one bi-phase error is allowed for each message. A test set involves performing the sequence 17 times. The test set shall be repeated N times, where N equals the number of data words sent. A complete test requires 2*N test sets to be performed, once for high bi-phase errors and once for low bi-phase errors.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.4 <u>Sync Encoding</u>. This test verifies the ability of the UUT to recognize sync errors. The sync pattern, as defined for this test, is a waveform with six 0.5 µs divisions. The divisions are represented as a 1 or 0 to indicate the polarity of each division of each division on the data bus. A proper command sync is represented as 111000 and a proper data sync is represented as 000111.

5.2.1.3.4.1 <u>Transmit Command Word</u>. This test shall verify that the UUT rejects transmit commands with invalid sync waveforms. The test sequence as defined in 5.2.1.3 shall be performed with a sync error encoded

in a transmit command word for test step 2. The test sequence shall be performed for each of the following invalid sync patterns:

111100, 110000, 111001, 000111

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.4.2 <u>Receive Command Word</u>. This test shall verify that the UUT rejects receive commands with invalid sync waveforms. The test sequence as defined in 5.2.1 .3 shall be performed with a sync error encoded in a receive command word for test step 2. The test sequence shall be performed for each of the following invalid sync patterns:

111100, 110000, 111001, 011000, 000111

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-CS.

5.2.1.3.4.3 <u>Data Word</u>. This test shall verify that the UUT rejects invalid data sync waveforms. Perform the test sequence as defined in 5.2.1.3 with a sync error encoded into each data word for test step 2. The message is a valid receive command word and the maximum number of data words that the UUT is designed to receive. Only one data word per message shall have an invalid sync encoded into it. The test sequence shall be performed N times for each of the following invalid sync patterns: where N equals the maximum number of data words in the message.

000011, 001111, 000110, 100111, 111000

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-ME.

Note: Data words shall not be encoded such that bit times 4 thru 8 match the terminal address of the UUT or be 11111 when the invalid data sync pattern 111000 is being used.

5.2.1.3.5 <u>Message Length</u>. These tests shall verify that the UUT properly defects an error when an incorrect number of data words are received.

5.2.1.3.4.1 <u>Transmit Command</u>. This test verifies the ability of the UUT to respond properly if the data word is contiguous to a transmit command word. perform the test sequence as defined in 5.2.1.3 with a data word contiguously following a transmit command word for test step 2.

The pass criteria for this test shall be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.5.2 <u>Receive Command.</u> This test shall verify that the UUT recognizes an error in the number of data words that are received. Perform the test sequence as defined in 5.2.1.3 with a data word count error in a receive message for test step 2. This message is a valid legal receive command word with the word count field equal to the maximum number of data words that the UUT is designed to receive but with a different number of date words than specified in the command word. The test sequence shall tee performed N+1 times where N equals the maximum number of data words. The first sequence shall have N+1 data words. The second sequence shall have N-1 data words and each of the remaining sequence shall remove one additional data word until the number of data words equals zero.

The pass criteria shall be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.5.3 <u>Mode command Word Count Error.</u> This test verifies the ability of the UUT to respond properly when an incorrect number of words are sent with a mode command. Perform the test sequence defined in 5.2.1.3 using valid receive mode command in step 2 which would normally have an associated data word transmitted with it but send the number of data words equal to the mode code value used. Repeat the test sequence with the same mode command but with no data word in step 2. Repeat the test sequence using a valid transmit mode command except send a data word contiguously following the command word.

100-19

In all three cases the pass criteria shall be: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.5.4 <u>RT to RT Word Count Error</u>. This test verifies the ability of the UUT to respond properly when an incorrect number of words are sent to it as a receiving RT during an RT to RT transfer. Perform the following test sequence.

Step 1   Send a valid legal RT to RT command pair followed in 4.0 to 12.0 µs by a valid status word and N data words to the UUT, where N is the number of data words requested in the transmit command.

Step 2   Send the same RT to RT command pair followed in 4.0 to 12.0 µs by a valid status word and N-1 data words.

Step 3   A transmit status mode command shall be sent to the receiving RT.

Step 4   Repeat steps 1 through 3 using a word count of N+1 in step 2.

The pass criteria in both cases shall be that the receiving RT's status is: step 1-CS; step 2-NR; step 3-ME.

5.2.1.3.6 <u>Contiguous Data</u>. This test verifies that the UUT recognizes discontiguous data in a message. Perform the test sequence as defined in 5.2.1.3 with a 4.0 µs data word gap error in a receive message for test step 2. The gap is measured as on figure 7. The receive message shall be a receive command with the maximum number of data words that the UUT is designed to receive with a gap between the command word and the first data word or between a data word pair. The test sequence shall be performed N times, where N equals the maximum number of data words. Only one gap time insertion is allowed per message.

The pass criteria for this test shall be: step 1 -CS; step 2-NR; step 3-ME.

5.2.1.3.7 <u>Terminal Fail Safe.</u> The purpose of this test is to verify that the terminal fail-safe timer is properly implemented in the UUT. The UUT is required to contain a hardware implemented timer that will cause the transmitter to shutdown if the UUT transmits a message longer than 800.0 µs. A fail-safe time~ut occurring on one bus shall not affect the transmitter on any other bus. The reception of a valid command on the bus on which the time-out has occurred shall enable the transmitter. The test sequence below shall be performed for each bus:

Step 1.   Initiate a condition in the UUT which causes the failsafe timer to timeout. Measure the duration of the transmission.

Step 2.   Remove the condition initiated in step 1.

Step 3.   Send the UUT a valid legal message over the bus on which the time-out has occurred.

The pass criteria shall be that the timeout in step 1 occurs and the transmitter is shut down allowing the total transmission time to be between 660.0 µs and 800.0 µs. The response of the UUT in step 3 shall be CS. Record the measure parameter at which the failsafe time-out occurs. For test failures, record the test parameters at which the failure occurred.

5.2.1.4 <u>Superseding Commands</u>. This test verifies that the UUT will not malfunction and responds properly to possible occurrences of superseding commands. The following test sequence shall be used for this test.

Step 1.   A valid legal receive message shall be sent to the UUT with the maximum number of words that the UUT is designed to receive encoded in the word count field.

Step 2.   Before step 1 is completed, a superseding message shall be sent to the UUT.

100-20

Step 3. A transmit status mode command shall be sent to the UUT.

Record the UUT's response to each step when the test is performed with the following superseding command formats (step 2):

a. After at least one data word is transmitted in step 1, but before the last data word is transmitted, follow the selected data word with a gap of 4.0 μs (reference figure 7), then a valid legal transmit command requesting the maximum number of data words that the UUT is designed to transmits.

b. Proceed as in "a" above, except transmit a valid legal transmit status mode command as the superseding command.

c. After at least one data word is transmitted in step 1, but before the last data word is transmitted, follow the selected data word contiguously with a valid legal transmit command requesting the maximum number of data words that the UUT is designed to transmit

d. After the last data word is transmitted in step 1 follow it contiguously with a valid legal transmit command requesting the maximum number of data words that the UUT is designed to transmit.

The pass criteria shall be:

For a, step 1 - NR, step 2 - CS, step 3 - CS
For b, step 1 - NR, step 2 - ME, step 3 - ME
For c, step 1 - NR, step 2 - NR, step 3 - ME or, step1 - NR, step 2 - CS, step 3 - CS
For d, step 1 - NR, step 2 - CS, step 3 - CS or, step 1 - NR, step 2 - NR, step 3 - ME

For test failures, record the test parameters for which the failure occurred.

5.2.1.4.1 <u>RT to RT Superseding Command</u>. The purpose of this test is to verify that the UUT will not malfunction and responds properly to a superseding command when operating as the receiving RT in a RT to RT transfer.

Step 1. Send a valid legal RT to RT command pair followed in 4.0 to 12.0 μs by a valid status word and N data words to the UUT, where N is the number of data words indicated in the receive command.

Step 2. Send the same RT to RT command pair followed in 4.0 to 12.0 μs by a valid legal transmit command addressed to the UUT.

Step 3. A transmit status mode command shall be sent to the UUT.

The pass criteria for this test shall be: step 1 - CS, step 2 - CS, step 3 - CS.

5.2.1.5 <u>Required Mode Commands</u>. The purpose of these tests is to verify that the UUT responds properly to the required mode commands. The tests are not intended to verify the mission aspects stated in the equipment specification. The UUT shall be tested for each required mode code with a subaddress held mode code indicator of all zeros and repeated with a subaddress field of all ones.

The pass criteria is defined in each test paragraph. If any test fails, record the UUT response to that test.

5.2.1.5.1 <u>Transmit Status</u>. The purpose of this test is to verify that the UUT has the ability to recognize the transmit status mode command and to transmit its last status word. The following sequence shall be performed:

Step 1.   A valid legal message shall be sent to the UUT on the primary bus.

Step 3.   A valid legal message shall be sent to the UUT on the alternate bus.

Step 4.   A transmit status mode command shall be sent to the UUT on the alternate bus.

Step 5.   A valid legal receive command with a parity error is a data word shall be sent on the primary bus.

Step 6.   A transmit status mode command shall be sent to the UUT on the primary bus.

Step 7.   Repeat step 6.

Step 8.   Repeat step 4.

Step 9   Repeat step 1.

Step 10.   Repeat step 2.

Step 11.   Repeat step 4.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-CS; step 3-CS; step 4-CS; step 5 NR; step 6-ME; step 7-ME; step 8-ME; step 9-CS; step 10-CS; step 11-CS.

5.2.1.5.2 <u>Transmitter Shutdown and Override.</u> This test shall verify that the UUT recognizes the dual redundant mode commands to shutdown the alternate bus transmitter and to override the shutdown. In a dual redundant system each bus must be tested as the alternate bus and as the primary bus. A valid legal transmitter shutdown mode command shall be sent to the UUT to cause an alternate bus transmitter shutdown. A valid legal override transmitter shutdown mode command shall be sent to the UUT to cause an override of the transmitter shutdown. The following test sequence shall be used for each case including verification of the UUT response indicated.

Step 1.   A valid legal command shall be sent on the primary bus to the UUT.

Step 2.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 3.   A valid legal transmitter shutdown mode command shall be sent to the UUT on the primary bus.

Step 4.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 5.   A valid legal command shall be sent on the primary bus to the UUT.

Step 6.   A valid legal override transmitter shutdown mode commend shall be sent to the UUT on the alternate bus.

Step 7.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 8.   A valid legal override transmitter shutdown mode command shall be sent to the U UT on the primary bus.

Step 9.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 10.   A valid legal command shall be sent on the primary bus to the UUT.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-CS; step 3-CS; step 4-NR; step 5-CS; step 6-NR; step 7-NR; step 8-CS; step 9-CS; step 10-CS.

5.2.1.5.3 <u>Reset Remote Terminal</u>. The purpose of this test is to verify that the UUT has the ability to recognize end properly operate when the reset remote terminal mode command is received. Note that this test provides characterization of reset time as a first step. It the reset time is variable, the test must be performed with conditions in the UUT set such that a maximum reset time results. The following sequence shall be performed.

Step 1.   A reset remote terminal mode command shall be sent to the UUT on one bus.

Step 2.   After time T from step 1, as measured per figure 7, a valid legal command shall be sent to the UUT on the same bus.

Starting with time T not less than 5 ms, repeat step 1 and step 2 while decreasing time T to 4.0 µs in steps no greater then 10.0 µs.

The minimum time, T_R, between step 1 and step 2, as measured per figure 7, in which the UUT's response to step 2 is CS (with busy bit reset), shall be recorded.

Step 3.   A valid legal transmitter shutdown mode command shall be sent to the UUT on the same bus.

Step 4.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 5.   A reset remote terminal mode command shall be sent to the UUT on the first bus.

Step 6.   After an intermessage gap to T_R, a valid legal command shall be sent to the UUT on the alternate bus.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-CS (with BUSY bit rest) for all time T > 5.0 ms, and CS or NR for T < 5.0 ms; step 3-CS; step 4-NR; step 5-CS; step 6-CS.

Having established the time, T_R, that the UUT requires in order to complete its reset function, the following sequence shall be performed.

Step 7.   A reset remote terminal mode command shall be sent to the UUT on one bus.

Step 8.   Send a valid legal receive command to the UUT on the same bus time T after the status response in step 7, where (T_R - 40.0 µs < T < (T_R - 20.0 µs), but not less than 4 µs, as measured in figure 7

Step 9.   Send a valid legal command to the UUT on the same bus time T after the status response in step 8 (if the response of the UUT during the reset period is NR, then time T shall be measured after the last data word of step 8), where 4.0 µs < T < 5.0 µs, as measured in figure 7.

The pass criteria for each of the above steps shall be as follows: step 7-CS; step 8-CS or NR, step 9-CS (with BUSY bit reset).

5.2.1.6 <u>Data Wraparound</u>. The purpose of this test is to verify that the UUT properly implements the data wrap-around capability. The following sequence shall be performed 10,000 times, with random data patterns for each data word in each sequence. The messages used shall contain the maximum number d data words that the RT is capable of transmitting or receiving, e.e., Record the number of correct responses and the number of incorrect responses.

Step 1.   Send a receive message to the UUT at subaddress 30 (11110) or the appropriate receive wrap around subaddress defined for the UUT.

Step 2.   Send a transmit command to the UUT with the appropriate transmit wrap-around subaddress and with the same word count as step 1.

The pass criteria shall be: step 1-CS; step 2-CS with each data word having the same bit pattern as the corresponding data word in step 1.

5.2.1.7 <u>RT to RT Message Errors</u>. The purpose of this test is to examine the UUT's response to specific errors in the message stream when operating as the receiving RT in an RT to RT transfer. For these tests, the bus tester shall act as the bus controller and the transmitting remote terminal as required.

5.2.1.7.1 <u>RT to RT Timeout</u>. The purpose of this test is to verify that the UUT functions properly when operating as the receiving RT in a RT transfer. The UUT must not respond after receiving a RT to RT command pair if the data is not received within 54 µs to 60 µs as shown on figure 8. This time is measured from the zero crossing of the parity bit of the receive command to the mid sync zero crossing of the first data word. The following test sequence shall be performed:

Step 1.   Send a valid legal RT to RT command pair followed in 4.0 µs to 12.0 µs by a valid status word with the RT address of the transmit command and the proper number of data words shall be sent to the UUT.

Step 2.   Send a transmit status mode command to the UUT.

Repeat step 1 and step 2 incrementing the time in step 1 between the transmit command and the status word in no greater than 0.5 µs increments until the time is equal to 30.0 µs

When the UUT (receiving RT) stops responding, the time T, as specified in figure 8, shall be measured and recorded.

The pass criteria shall be: step 1 -CS for T < 54.0 µs CS or NR for 54.0 < T < 60.0 µs and NR for T > 60.0 µs; step 2 -CS if step 1 is CS, ME if step 1 is NR.

5.2.1.7.2 <u>RT to RT message format errors.</u> The purpose of this test is to verify that the UUT will not malfunction and responds properly to errors in the message stream associated with the transmitting RT when the UUT is the receiving RT in a RT to RT transfer. The following test sequence shall be used:

Step 1.   Send a valid legal RT to RT command pair followed in 4.0 to 12.0 µs by a valid status word and N data words to the UUT, where N is the number of date words indicated in the receive commend.

Step 2.   Send the same RT to RT message as in steps 1 injecting one of the errors indicated below.

Step 3.   Send a transmit status mode command to the UUT.

Record the UUT's response to each step when the test is performed with each of the following format errors (step 2):

a.   Invalid transmit command word. Inject an invalid sync pattern in the transmit command word followed in 4.0 to 12.0 µs with a receive command with the same address as the transmit command followed by N data words and no status response. This command combination will appear to the UUT as the same message format in step 1 with an invalid transmitting RT's command word.

b.   Invalid transmitting RT's status word. Inject an invalid sync pattern in the transmitting RT's status word.

c.   Wrong sync type in the transmitting RT's status word. Send the same RT to RT command pair as in step 1 followed in 4.0 to 12.0 µs by N+1 data words.

The pass criteria for each of the test formats shall be as follows: step 1-CS; step 2-NR, step 3-ME.

5.2.1.7.3 <u>Transmitting RT errors</u>. The purpose of this test is twofold: first to determine whether the UUT verities the proper occurrence of the transmitting RT's command and status word and, secondly, to verify that the UUT will not malfunction and responds properly to errors in the message stream associated with the transmitting RT when the UUT is the receiving RT in an RT to RT transfer. Although recommended by Notice 2, the standard does not require the receiving RT to verify the proper occurrence of the transmitting RT's command and status word. Therefore, the first purpose of this test is for characterization of the UUT only. The following test sequence shall be used:

Step 1.   Send a valid legal RT to RT command pair followed in 4.0 to 12.0 µs by a valid   status word and N data words to the UUT, where N is the number of date words indicated in the receive command.

Step 2.   Send the same RT to RT command pair as in step 1 followed in 4.0 to 12.0 µs by a status word containing a terminal address different from both the transmit command and the UUT, followed by N data words.

Step 3.   A transmit status mode command shall be sent to the UUT.

The pass criteria shall be the following: step 1-CS; step 2-CS or NR, step 3-CS U step 2 is CS; ME if step 2 is NR.

5.2.1.8 <u>Bus Switching.</u> This test shall be performed only if the UUT is configured with dual redundant buses. This test verifies that the dual redundant remote terminal properly performs the bus switching requirements of MIL-STD-1553 (pare on "Data Bus Activity"). The requirements are as follows:

a.   If the UUT is receiving or operating on a message on one bus, and another valid, Legal command to the UUT occurs on the opposite bus later in time, then the UUT is required to reset and respond appropriately to the later command on the opposite bus.

b.   An invalid command on the alternate bus shall not affect the response of the UUT to commands on the original bus.

Unless otherwise specified, legal messages are used in this test. The interrupting message on the alternate bus shall be swept through the command word, the response time gap, the UUT's status status word, and the UUT's data transmission on the first bus. For all tests, record the command words used. The following test sequences shall be performed twice for each interrupting command, once for each redundant bus.

RT transmitting:

Step 1.   Send a valid transmit command to the U UT requesting the maximum number of data words that the UUT is designed to transmit.

Step 2.   Send the interrupting command on the alternate bus beginning 4.0 µs after the beginning of the first command.

Step 3.   Send a valid transmit status mode command after the messages on both buses have been completed.

Step 4.   Repeat step 1 through step 3 increasing the time between step 1 and step 2 in no greater than 0.25 µs increments until the messages no longer overlap.

Perform the test with the following interrupting messages for step 2.

a.   A valid legal message

b.   A message with a parity error in the command word.

c.   A valid message with a terminal address different than that of the UUT.

The pass criteria shall be: for a, step 1 -NR, truncated message or CS, step 2 -CS and step 3 -CS; and for b and c, step 1 CS; step 2 -NR and step 3 -CS. For test failures, record the test parameters at which the failure occurred.

RT receiving:

Step 1.   Send a valid RT to RT message command to the UUT and a second RT with the UUT the receiving terminal with the maximum number of data words that the UUT is designed to receive.

Step 2.   Send the interrupting command on the alternate bus beginning 4.0 µs after the beginning of the first command.

Step 3.   Send a valid transmit status mode command after the messages on both buses have been completed.

Step 4.   Repeat step 1 through step 3 varying the time between step 1 and step 2 in no greater than 0.25 µs increments until the messages no longer overlap.

Perform the test with the following interrupting messages for step 2.

a.   A valid legal message

b.   A message with a parity error in the command word.

c.   A valid message with a terminal address different than that of the UUT.

The pass criteria shall be: for a, step 1-NR or CS, step 2-CS and step 3-CS; and for b and c, step 1-CS, step-NR and step 3-CS. For test failures, record the test parameters at which the failure occurred.

5.2.1.9 Unique Address. The purpose of this test is to verify that the UUT can be assigned any unique address from an external connector on the UUT. The following sequence shall be performed for the UUT:

Step 1.   Send a valid, legal command to the UUT.

Step 2.   Repeat step 1 thirty-one times with the same command word except use all other possible bit combinations in the RT address field of the command word.

Step 3.   Repeat step 1 and step 2 after externally changing the RT address for all possible combinations from 00000 thru 11110.

Step 4.   After externally changing the RT address to simulate a single point address validation failure (e.g., partly error on the address lines), repeat step 1 and step 2.

The pass criteria shall be: step 1-CS; step 2-NR for each combinations; step 3-same as step 1 and step 2; step 4-NR for each combination.

Note: Power cycling may be required after externally changing the RT address.

5.2.2 Optional Operation. This section provides for testing the optional requirements of MIL-STD-1553. If a remote terminal implements any of the options, it shall be tested in accordance with the test herein identified for the option. if the transmit status mode command shall be used.

5.2.2.1 Optional Mode Commands. The purpose of these tests is to verify that the UUT responds properly to implemented mode commands. The tests are not intended to verify the mission aspects stated in the equipment specification. The UUT shall be tested for each mode code implemented with a subaddress field mode code indicator of all zeros and repeated with a subaddress field of ail ones.

The pass criteria is defined in each test paragraph. If any test fails, record the UUT response to that test.

5.2.2.1.1 Dynamic Bus Control. The purpose of this test is to verify that the UUT has the ability to recognize the dynamic bus control mode command and to take control of the data bus. A valid legal dynamic bus control mode command shall be sent to the UUT. The UUT shall take control of the data bus when its response is DBA as required in the UUT's design specification

The pass criteria shall be that the UUT respond with a DBA upon acceptance of bus control or a CS upon rejection of bus control.

5.2.2.1.2 Synchronize. The following paragraphs provide the test criteria for the synchronize mode commands.

5.2.2.1.2.1 Synchronize (without data word). The purpose of this test is to verify that the UUT has the ability to recognize a synchronization mode command without using a data word. A valid legal synchronize (without data word) mode command shall be sent to the UUT.

The pass criteria shall be that the UUT respond with CS.

52.2.1.2.2 Synchronize (with data word). The purpose of this test is to verify that the UUT has the ability to recognize a synchronization mode command which uses a data word. A valid legal synchronize (with data word) mode command shall be sent to the UUT.

The pass criteria shall be that the UUT respond with CS.

5.2.2.1.3 Initiate Self-Test. The purpose of this test is to verify that the UUT has the ability to recognize and properly operate when the initiate self-test mode command is received. Note that this test provides characterization of self-test time as a first step. If the self-test ti me is variable, the test must be performed with conditions in the UUT set such that a maximum self test time results. The following sequences shall be performed:

Step 1.   An initiate self-test mode command shall be sent to the UUT on one bus.

Step 2.   After time T from step 1, as measured per figure 7, a valid legal command shall be sent to the UUT on the same bus.

Starting with time T not less than 100 ms, repeat step 1 and step 2 while decreasing time T to 4.0 µs in steps no greater than 1.0 ms. Finer granularity, 10.0 µs maximum steps, shall be used to more accurately determine the self-test time when the time of self-test is determined using the coarser steps.

The minimum time T_S between step 1 and step 2, as measured per figure 7 in which the UUT's response to step 2 is CS (with BUSY bit reset), shall be recorded.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-CS (with BUSY bit reset) for all time T 100 ms, and CS or NR for time T 100 ms.

100-27

Having established the time, T_S, that the UUT requires in order to complete its self-test function, the following sequence shall be performed.

Step 3.   An initiate self-test mode command shall be sent to the UUT on one bus.

Step 4.   Send a valid legal receive command to the UUT on the same bus time T after the status response in step 3, where (T_S-40.0 µs)   T   (T_S -20.0 µs), but not less than 4.0 µs, as measured in figure 7.

Step 5.   Send a valid legal command to the UUT on the same bus time T after the status response in step 4 (if the response of the UUT during the reset period is NR, then time T shall be measured red after the last data word of step 4), where 4.0 µs   T   5.0 µs, as measured in figure 7.

The pass criteria for each of the above steps shall be as follows step 3-CS, step 4-CS or NR, step 5-CS (with BUSY bit reset).

5.2.2.1.4 <u>Transmit BIT Word</u>. The purpose of this test is to verify that the UUT has the ability to recognize this mode command. A valid legal transmit BIT mode command shall be sent to the UUT.

The pass criteria shall be that the UUT respond with CS.

5.2.2.1.5 <u>Selective Transmitter shutdown and Override</u>. This test shall verify that the UUT recognizes the multi-redundant mode code commands to shut down a selected bus transmitter and to override the shutdown. In a multi-redundant system, each bus must be tested as the primary bus with the remaining busses as alternate busses. A valid legal selected transmitter shutdown mode command shall be sent to the UUT accompanied by the appropriate data word to cause a selective bus transmitter shutdown. A valid legal override selected transmitter shutdown mode command shall be sent to the UUT accompanied by the appropriate data word to cause an override of the selected bus transmitter shutdown. The following test sequence shall be performed using each bus as the primary bus and each of the remaining busses in turn as the alternate bus, including verification of the UUT response indicated.

Step 1.   A valid legal command shall be sent on the first bus to the UUT.

Step 2.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 3.   A valid legal selected transmitter shutdown mode command shall be sent to the UUT on the first bus with the data word encoded to shutdown the alternate bus.

Step 4.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 5.   A valid legal command shall be sent on the first bus to the UUT.

Step 6.   A valid legal override selected transmitter shutdown mode command shall be sent to the UUT on the alternate bus with the same data word as sent in step 3.

Step 7.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 8.   A valid legal override selected transmitter shutdown mode command shall be sent to the UUT on the first bus with the same data word as sent in step 3.

Step 9.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 10.   A valid legal command shall be sent on the first bus to the UUT.

Step 11.   Repeat step 3 except that the data word shall be encoded with a bit pattern that would normally shutdown the first bus if it was sent on the alternate bus.

Step 12.   Repeat step 4.

Step 13.   Repeat step 5.

The data words associated with step 3 and step 11 for each bus shall be recorded.

The pass criteria for each of the above steps shall be as follows: step 1-CS, step 2-CS, step 3 -CS. step 4-NR, step 5-CS, step 6-NR, step 7-NR, step 8-CS, step 9-CS, step 10-CS, step 11-CS, step 12-CS, step 13-CS.

5.2.2.1.6 Terminal Flag Bit Inhibit and Override. This test verifies that the UUT recognizes and respond properly to the mode code commands of inhibit terminal flag bit and override inhibit terminal flag bit. Beginning in step 2 of the test sequence below the UUT shall be caused to set the terminal flag bit.

Step 1.   A valid legal receive command with a t least one data word shall be sent to the UUT.

Step 2.   Procedures as defined for the UUT shall be performed that will set the terminal flag in the UUT status response. Send a valid legal receive command with at least one data word to the UUT.

Step 3   A valid legal inhibit terminal flag mode code command shall be sent to the UUT.

Step 4.   Repeat step 1.

Step 5.   A valid legal override inhibit terminal flag mode code command shall be sent to the UUT.

Step 6.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 7.   Procedures as defined for the UUT shall be performed which resets the TF bit.

Step 8.   Repeat step 1.

The pass criteria for each of the above steps shall be as follows: step 1-CS, step 2-TF, step 3-CS or TF, step 4-CS, step 5-CS or TF, step 6-TF, step 8-CS.

5.2.2.1.7 Transmit Vector Word. This test verifies the capability of the UUT to recognize and respond properly to a transmit vector word mode code command. A valid legal transmit vector word mode code command shall be sent to the UUT.

The pass criteria shall be that the UUT respond with CS.

5.2.2.1.8 Transmit Last Command. This test verifies that the UUT recognizes and response properly to a transmit last command mode code. The following test sequence shall be used:

Step 1.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 2.   A valid legal receive command different from that used in step 1 above with at least one data word shall be sent to the UUT and a parity error shall be encoded into the first data word.

Step 3.   A valid transmit last command mode command shall be sent to the UUT.

Step 4.   A valid transmit status mode command shall be sent to the UUT.

Step 5.   A valid legal transmit last command mode command shall be sent to the UUT.

Step 6.   A valid legal transmit last command mode command shall be sent to the UUT.

Step 7.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 8.   A valid legal transmit last command shall be sent to the UUT.

Step 9.   A valid legal transmit command shall be sent to the UUT.

Step 10. A valid legal transmit last command mode command shall be sent to the UUT.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-NR; step 3-ME, followed by a data word containing the command word from step 2; step 4-ME; step 5-ME, followed by a data word containing the command word from step 4; step 6-ME, followed by a data word containing the command word from step 4; step 7-CS; step 8-CS, followed by a data word containing the command word from step 7; step 9-CS; step 10-CS, followed by a data word containing the command word from step 9.

5.2.2.2 <u>Status Word Bits.</u> The following tests verify that all implemented status code bits are properly used and cleared. implementation of all status code bits in the status word except the ME bit is optional. In addition to the separate tests, for each of the following status bits service request, busy, subsystem flag, and terminal flag provide the analysis as listed below.

a.   What conditions set the status bit in the status word transmitted on the data bus.

b.   What conditions reset the status bit in the status word transmitted on the data bus.

c.   If the condition specified in item a. occurred end disappeared without intervening commands to the UUT,   list   the cases where the status bit is set and reset in response to a valid, non-node command to the UUT.

d.   Given that the status bit was set, and the condition which set the bit has gone away, list the cases where   the sta-tus bit is still set in response to the second valid, non-mode command to the UUT.

The UUT has failed a test sequence if it does not respond as indicated in each of the separate tests below.

5.2.2.2.1 <u>Service Request.</u> This test verifies that the UUT sets the service request bit as necessary and clears it when appro-priate. The UUT shall set bit time eleven of the status word when a condition in the UUT warrants the RT to be serviced. A reset of the bit shall occur as defined by each RT. The following steps shall be performed and the appropriate responses verified.

Step 1.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 2.   A condition which causes the service request bit to be set shall be introduce into the UUT. A valid
legal command that does not service the request shall be sent to the UUT.

Step 3.   A valid legal command that does not service the request shall be sent to the UUT.

Step 4.   Procedures, as defined for the UUT, shall be performed which resets the service request bit.

Step 5.   A valid legal receive command with at least one data word she! be sent' to the UUT.

The pass criteria for each of the above steps shall be as follows: step 1-CS, with the service request bit reset; step 2-SRB; step 3-SRB; step 5-CS, with the service request bit reset. All commands and UUT responses shall be recorded.

5.2.2.2.2 <u>Broadcast command Received</u>. This test verifies that the UUT sets the broadcast command received bit of the status word alter receiving a broadcast command. The UUT shall set status bit fifteen to a logic one after receiving the broadcast command. The following test sequence shall be performed using either the transmit last command or transmit status mode code command to verify the bit condition.

Step 1.   A valid legal broadcast receive message shall be sent to the UUT.

Step 2.   A valid legal transmit last command shall be sent to the UUT.

Step 3.   A valid, legal, non-broadcast command shall be sent to the UUT.

Step 4.   Repeat step 1.

Step 5.   Repeat step 3.

Step 6.   Deleted.

Step 7.   Deleted.

The pass criteria for each of the above steps shall be as follows: step 1-NR; step 2-BCR, and the data word contains the bit pattern of the commend word in step 1; step 3-CS; step 4--NR; step 5-CS; step 6-NR; step 7-ME and BCR, and the data word contains the bit pattern of the command word in step 6. All commands and UUT responses shall be recorded.

5.2.2.2.3 <u>Busy.</u> This test verifies the capability of the UUT to set the busy bit of the status word. Bit time sixteen of the status word shall be set when the UUT is busy. Prior to performing the test sequence below, a condition which sets the busy bit must be received.

Step 1.   A valid legal transmit command shall be sent to the UUT.

Step 2.   Procedures, as defined for the UUT, shall be performed which resets the busy bit

Step 3.   A valid legal transmit command shall be sent to the UUT.

The pass criteria for each of the above steps shall be as follows: step 1-BUSY; step 3-CS. All commands and UUT responses shall be recorded.

5.2.2.2.4 <u>Subsystem Flag</u>. This test verifies the capability of the UUT to set the subsystem tag of the status word. Bit time seventeen of the status word shall be set to a logic one when a subsystem fault has been determined. Prior to performing the test sequence below, a condition which sets the subsystem flag bit must be activated.

Step 1.   A valid legal transmit command shall be sent to the UUT.

Step 2.   Remove the condition which sets the subsystem flag bit. Cycling power to the UUT shall not be part of these procedures to reset the SF bit.

Step 3.   A valid legal transmit command shall be sent to the UUT.

Step 4.   Repeat step 3.

The pass criteria for each of the above steps shall be as follows: step 1-SF; step 3-CS; step 4-CS. All commands and UUT responses shall be recorded.

5.2.2.2.5 <u>Terminal Flag.</u> This test verifies that the UUT sets the terminal flag bit as necessary and clears it when appropriate. The UUT shall set bit time nineteen of the status word when an occurrence in the UUT shall set causes a terminal fault condition Prior to performing the test sequence below, a condition which sets the terminal flag bit must be activated.

Step 1. A valid legal receive command with at least one data word shall be sent to the UUT.

Step 2. Remove the condition which sets the terminal flag bit. Cycling power to the UUT shall not be part of this procedure.

Step 3. A valid legal transmit command shall be sent to the UUT.

Step 4. Repeat step 3.

The pass criteria for each of the above steps shall be as follows: step 1-TF; step 3-CS or TF; step 4-CS. All commands and UUT responses shall be recorded.

5.2.2.3 <u>Illegal Command.</u> This test verifies that the UUT recognizes and responds properly to illegal commands when the illegal command detection option is implemented. The following sequence shall be performed:

Step 1. Send an illegal receive command to the UUT.

Step 2. Send a transmit status mode command to the UUT.

Step 3. Send a valid legal transmit command to the UUT.

Step 4. Send an illegal receive command to the UUT with a parity error in one of the data words.

Step 5. Send a transmit status mode command to the UUT.

Step 6. Repeat step 3.

Step 7. Send an illegal transmit command to the UUT.

Step 8. Send a transmit status mode command to the UUT.

Step 9. Repeat step 3.

Step 10. Send an illegal command to the UUT with a parity error in the command word.

Step 11. Send a transmit last command mode command to the UUT. If the transmit last command mode command is not implemented in the RT, send a transmit status mode command instead.

The pass criteria shall be: step 1-ME; step 2-ME; step 3-CS; step 4-NR; step 5-ME; step 6-CS; step 7-status only with ME bit set; step 8-ME; step 9-CS; step 10-NR; step 11-CS if the transmit last command mode command was used the data word shall be the command word sent in step 9.

5.2.2.4 <u>Broadcast Mode Command.</u> The purpose of this test is to verify that the UUT responds properly to implemented broadcast mode commands. This test is not intended to verify the mission aspects stated in the equipment specification. The UUT shall be tested for each mode code implemented with a subaddress filed mode code indicator of all zeros and repeated with a subaddress field of all ones. Use the following test sequence unless otherwise noted.

Step 1. A valid receive message shall be sent to the UUT.

Step 2.   A valid legal broadcast message shall be sent to the UUT.

Step 3.   A transmit last command mode command shall be sent to the UUT.

5.2.2.4.4 Broadcast Transmitter Shutdown and Override. The purpose of this test is to verify that the UUT has the ability to recognize and properly execute these broadcast mode commands. The pass criteria for each individual test is contained in the paragraph below.

The following sequence shall be performed for each test:

Step 1.   A valid legal command shall be sent on the first bus to the UUT.

Step 2.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 3.   A valid legal broadcast transmitter shutdown command shall be sent to the UUT on the first bus.

Step 4.   A transmit last command mode shall be sent on the first bus to the UUT.

Step 5.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 6.   A valid legal command shall be sent on the first bus to the UUT.

Step 7.   A valid legal broadcast override transmitter shutdown mode command shall be sent to the UUT on the alternate bus.

Step 8.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 9.   A valid legal broadcast override transmitter shutdown mode command shall be sent to the UUT on the first bus.

Step 10. A transmit last command mode command shall be sent on the first bus to the UUT.

Step 11. A valid legal command shall be sent on the alternate bus to the UUT.

Step 12. A valid legal command shall be sent on the first bus to the UUT.

The pass criteria for each of the above steps shall be as follows: step 1 -CS; step 2-CS; step 3-NR; step 4 BCR (and the data word contains the command word of step 3); step 5 NR; step 6-CS; step 7-NR; step 8-NR; step 9-NR; step 10-BCR (and the data word contains the command word of step 9); step 11-CS; step 12-CS.

5.2.2.4.5 Broadcast Selective Transmitter Shutdown and Override. This test shall verify that the UUT recognizes the multi-redundant broadcast mode code commands to shutdown a selected bus transmitter and to override the shutdown. In a multi-redundant system each bus must be tested as the primary bus with the remaining busses as alternate busses. A valid legal broadcast selected transmitter shut down mode command shall be sent accompanied by the appropriate data word to cause a selective bus transmitter shutdown. A valid legal broadcast override selected transmitter shutdown mode command shall be sent accompanied by the appropriate data word to cause an override of the selected bus transmitter shutdown. The following test sequence shall be performed using each bus as the primary bus and each of the remaining busses in turn as the alternate bus including verification of the UUT response indicated.

Step 1.   A valid legal command shall be sent on the first bus to the UUT.

Step 2.   A valid legal command shall be sent on the alternate bus to the UUT

Step 6.   A valid legal override inhibit terminal flag broadcast mode code c command shall be sent to the UUT.

Step 7.   A transmit last command mode command shall be sent to the UUT.

Step 8.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 9.   Procedures as defined for the UUT, shall be performed which resets the TF tel.

Step 10.   Repeat step 1.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-TF; step 3-NR; step 4 BCR or (BCR and TF) and in either case the data word contains the command word of step 3; step 5-CS; step 6-NR; step 7-BCR or (BCR and TF) and in either case the data word contains the command word of step 6; step 8-TF; step 10-CS.

5.2.2.4.7 <u>Broadcast reset remote terminal</u>. The purpose of this test is to verify that the UUT has the ability to recognize the broadcast mode code command to reset itself to a power up initialized state. The following sequence shall be performed:

Step 1.   A broadcast reset remote terminal mode command shall be sent to the UUT on one bus.

Step 2.   After time T from step 1 as measured per figure 7 a valid legal transmit command shall be sent to the UUT on the same bus.

Step 3.   The time T shall be obtained by repeating step 1 and step 2 while varying the intermessage gap from 100.0 ms down to 4.0 μs in the following steps: from 100.0 ms to 6 ms in no greater than 1.0 ms steps and from 6.0 ms to 4.0 μs in no greater than 10.0 μs steps. When the time T is between 5.0 ms and 100.0 ms then in addition to each command sent in step 2 a minimum of one valid legal command shall be sent to the UUT positioned within 4.0 ms after step 1.

Step 4 .   A valid legal transmitter shutdown mode command shall be sent to the UUT on the same bus.

Step 5.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 6.   A broadcast reset remote terminal mode command shall be sent to the UUT on the first bus.

Step 7.   After 5 ms repeat step 5.

The minimum time between step 1 and step 2 as measured per figure 7 in which the UUT s response to step 2 is CS (with BUSY bit reset) shall be recorded.

The pass criteria for each of the above steps shall be as follows: step 1-NR; step 2-CS (with BUSY bit reset) for all time T 5 ms and CS or nr for T < 5 ms: step 4-CS; step 5-NR step 7-CS.

5.2.2.4.8   <u>Broadcast dynamic bus control</u>. The purpose of this test is to insure that the UUT does not take over bus control function in response to a broadcast mode command. The following sequence shall be performed:

Step 1.   A broadcast dynamic bus control mode command shall be sent to the UUT.

Step 2.   A transmit status mode command shall be sent to the UUT.

The pass criteria shall be: step 1-NR; step 2-CS (the BCR bit shall be set, ME bit may be set,, but the DBA bit shall not be set).

Step 3.   A valid legal broadcast selected transmitter shutdown mode command shall be sent to the UUT on the first bus with the data word encoded to shutdown the alternate bus.

Step 4.   A transmit last command mode shall be sent on the first bus to the UUT.

Step 5.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 6.   A valid legal command shall be sent on the first bus to the UUT.

Step 7.   A valid legal broadcast override selected transmitter shutdown mode command   shall be sent to the UUT on the alternate bus with the same data word as sent in step 3.

Step 8.   A valid legal command shall be sent to the UUT on the alternate bus.

Step 9.   A valid legal broadcast override selected transmitter shutdown mode command shall be sent to the UUT on the first bus with the same data as sent in step 3.

Step 10.   Repeat step 4.

Step 11.   A valid legal command shall be sent on the alternate bus to the UUT.

Step 12.   A valid legal command shall be sent on the first bus to the UUT.

Step 13.   Repeat step 3 except that the data word shall be en coded with a bit pattern that would normally shutdown the first bus if it was sent on the alternate bus.

Step 14.   Repeat step 4.

Step 15.   Repeat step 5.

Step 16.   Repeat step 6.

The data words associated with step 3 and step 13 for each bus shall be recorded.

The pass criteria for each of the above steps shall be as follows: step 1-CS; step 2-CS; step 3-NR; step 4BCR (and the data word contains the command word of step 3); step 5-NR; step 6CS; step 7-NR; step 8-NR; step 9-NR; step 10-BCR (and the data word contains the command word of step 9); step 11-CS; step 12-CS; step 13-NR; step 14-BCR (and the data word contains the command word of step 13); step 15-CS; step 16-CS.

5.2.2.4.6 <u>Broadcast terminal flag bit Inhibit and override.</u> This test verifies that the UUT recognizes and responds properly to the broadcast mode code commands of inhibit terminal flag bit and override inhibit terminal flag bit. Beginning in step 2 of the test sequence below, the UUT shall be caused to set the terminal flag bit.

Step 1.   A valid legal receive command with at least one data word shall be sent to the UUT.

Step 2.   Procedures, as defined for the UUT, shall be performed that will set the terminal flag in the UUT status response. Send a valid legal receive command with at least one data word to the UUT.

Step 3.   A valid legal inhibit terminal nag broadcast mode code command shall be sent to the UUT.

Step 4.   A transmit last command mode command shall be sent to the UUT.

Step 5.   Repeat step 1.

5.2.2.5 <u>Error injection - broadcast messages.</u> The purpose of this test is to verify the UUT's response to data specific errors in broadcast messages. Unless otherwise noted, the following test sequence shall be used for all error injection tests. The error to be encoded in step 4 for a given message is specified in each test paragraph. The pass criteria is defined in each test paragraph. All responses shall be recorded.

Test sequence:

Step 1.   A valid legal broadcast message shall be sent to the UUT.

Step 2   A transmit last command mode command shall be sent to the UUT.

Step 3.   A valid legal receive message shall be sent to the UUT.

Step 4.   A broadcast message containing the specified error shall be sent to the UUT.

Step 5.   A transmit last command mode command shall be sent to the UUT.

Step 6.   Repeat Step 3.

5.2.2.5.1 <u>Parity: bus controller (BC) RT broadcast.</u> The purpose of this test is to verify the UUT's capability to detect parity errors embedded in different words within a message.

5.2.2.5.1.1 <u>Command word error.</u> This test verifies the ability of the UUT to recognize a parity error in the broadcast command. The test sequence as defined in 5.2.2.5 shall be performed with a parity error encoded in a broad-cast command for test step 4.

The pass criteria for this test shall be: step 1-NR; step 2-BCR end the data word contains the command word of step 1; step 3-CS; step 4 NR; step 5-CS and the data word contains the command word of step 3; step 6-CS.

5.2.2.5.1.2 <u>Data word error.</u> This test verifies the ability of the UUT to recognize a parity error occurring in a data word. The test sequence as defined in 5.2.2.5 shall be performed with a parity error encoded in a data word for step 4. The message shall be a BC-RT (broadcast) command with the maximum number of data words that the UUT is designed to receive. The test sequence must be executed N times, where N equals the number of data words in the message. Each data word in the message will be transmitted with a parity error. Only one parity error is allowed per message.

The pass criteria for this test shall be: step 1-NR; step 2- BCR end the date word contains the command word of step 1; step 3-CS; step 4-N R; step 5-M E (BC R may be set) and the data word contains the command word of step 4; step 6-CS.

5.2.2.5.2 <u>Message length, BC to RT broadcast.</u> This test shall verify that the UUT recognizes an error in the number of data words that are received. Perform the test sequence as defined in 5.2.2.5 with the data word count error in a BC - RT (broadcast) message for test step 4. The message is a valid legal broadcast command word with the word count field equal to the maximum number of data words that the UUT is designed to receive and a different number of data words than specified in the command word. The test sequence shall be performed N+1 times, where N equals the maximum number of data words. The first sequence shall have N+1 data words. The second sequence shall have N-1 data words. Other sequences shall remove one additional data word until the number of data words equals zero.

The pass criteria for this test shall be: step 1- NR; step 2- BCR and the date word contains the command word of step 1; step 3- CS; step 4- NR; step 5- ME (BCR may be set) and the data word contains the command word of step 4; step 6-CS.

5.3 <u>Noise Rejection Test</u>. This test verifies the RT's ability to operate in the presence of noise. The maximum word error rate for a RT is one part in 107. While performing this test, all words received by the UUT shall be in presence of an additive white Gaussian noise distributed over a bandwidth of 1.0 kHz to 4.0 MHz at an RMS amplitude of 140 mV for transformer could stubs or 200 mV for direct coupled stubs measured at point A of figure 9A or figure 10A. This test shall be conducted with a signal level of 2.1 V peak-to-peak, line-to-line, for transformer coupled stubs or 3.0 V peak-to-peak, line-to-line, for direct coupled submeasured at point A of figure 9A or figure 10A. The rise and fall time of the transmitted message (measured at a data bit zero crossing with the prior zero crossing and the next zero crossing at 500 ns intervals from the ensured zero crossing) measured at point "A" shall be 200.0 ns ± ns. Figure 9A and figure 10A depict the configurations for conducting the noise rejection test. Air Force applications shall only use the configuration in figure 10A. Figure 9B and figure 10B depict suggest configurations for the noise rejection test. The noise test shall run continuously with intermessage gaps of 100.0 µs until the total number of all words received by the UUT exceeds the required number for acceptance of the UUT or is less than the required number for rejection of the terminal, as specified in table III. All data words used in the tests shall contain random bit patterns. These bit patterns shall be unique for each data word in a message and shall change randomly from message to message. A unit under test (UUT) that provides transformer and direct c coupled stubs shall be tested on both stubs. The noise test shall be performed on all buses for UUTs with redundant bus configurations

TABLE III. Criteria for acceptance or Rejection of a Terminal for the Noise Rejection Tests
total number of words received by the terminal (in multiples of 10^7)

NO. OF ERRORS	REJECT (EQUAL OR LESS)	ACCEPT (EQUAL OR LESS)
0	N/A	4.40
1	N/A	521
2	N/A	6.02
3	N/A	6.83
4	N/A	7.64
5	N/A	8.45
6	0.45	9.27
7	1.26	10.08
8	2.07	10.89
9	2.88	11.70
10	3.69	12.51
11	4.50	13.32
12	5.31	14.13
13	6.12	14.94
14	6.93	15.75
15	7.74	16.56
16	8.55	17.37
17	9.37	18.19
18	10.18	19.00
19	10.99	19.81
20	11.80	20.62
21	12.61	21.43
22	13.42	22.24
23	14.23	23.05
24	15.04	23.86
25	15.85	24.67
26	16.66	25.48
27	17.47	26.29
28	18.29	27.11
29	19.10	27.92
30	19.90	28.73
31	20.72	29.54
32	21.53	30.35
33	22.34	31.16
34	23.15	31.97
35	23.96	32.78
36	24.77	33.00
37,	25.58	33.00
38	26.39	33.00
39	27.21	33.00
40	28.02	33.00
41	33.00	N/A

100-38

APPENDIX A
TEST PLAN TO MIL-STD-1553B
CROSS REFERENCE

P= Primary Reference
R= Related Reference

Test Plan			MIL-STD-1553B
5.0	Detailed requirements		
5.1	Electrical tests		
5.1.1	Output characteristics		
5.1.1.1	Amplitude	(Transformer coupled)	4.5.2.1.1.1-P
		(Direct coupled)	4.5.2.2.1.1-P
5.1.1.2	Rise time/fall time	(Transformer coupled)	4.5.2.1.1.2-P
		(Direct coupled)	4.5.2.2.1.2-P
5.1.1.3	Zero crossing stability		
		(Transformer coupled)	4.5.2.1.1.2-P
		(Direct coupled)	4.5.2.2.1.2-P
5.1.1.4	Distortion overshoot & ringing		
		(Transformer coupled)	4.5.2.1.1.2-P
		(Direct coupled)	4.5.2.2.1.2-P
5.1.1.5	Output symmetry	(Transformer coupled)	4.5.2.1.1.4-P
		(Direct coupled)	4.5.2.2.1.4-P
5.1.1.6	Output noise	(Transformer coupled)	4.5.2.1.1.3-P
		(Direct coupled)	4.5.2.2.1.3-P
5.1.1.7	Output isolation		4.6.1-P
			30.10.6-P
5.1.1.8	Power on/off		
5.1.1.8.1	Power on/off noise	(Transformer coupled)	30.10.6-P
		(Direct coupled)	30.10.6-P
5.1.1.8.2	Power on response		30.5.1-P
			3.16-R
5.1.1.9	Terminal response time		4.3.3.6-P
5.1.1.10	Frequency stability		4.3.3.3-P
5.1.2	Input characteristics		
		(Transformer coupled)	4.5.2.1.2-P
		(Direct coupled)	4.5.2.2.2-P
5.1.2.1	Input waveform compatibility		
		(Transformer coupled)	4.5.2.1.2.1-P
		(Direct coupled)	4.5.2.2.2.1-P
5.1.2.1.1	Zero crossing distortion		
		(Transformer coupled)	4.5.2.1.2.1-P
		(Direct coupled)	4.5.2.2.2.1-P
5.1.2.1.2	Amplitude variations		
		(Transformer coupled)	4.5.2.1.2.1 -P
		(Direct coupled)	4.5.2.2.2.1-P
5.1.2.1.3	Rise and fall time		4.5.2.1.1-P
5.1.2.1.3.1	Trapezoidal		4.5.2.1.2.1-P
5.1.2.1.3.2	Sinusoidal		4.5.2.1.1-P

100-39

Test Plan			MIL STD-1553B
5.1.2.2	Common mode rejection		
		(Transformer coupled)	4.5.2.1.2.2-P
		(Direct coupled)	4.5.2.2.2.2-P
5.1.2.3	Input impedance		
		(Transformer coupled)	4.5.2.1.2.3-P
		(Direct coupled)	4.5.2.2.2.3-P
5.2	Protocol tests		4.4.1.3-P
5.2.1	Required remote terminal operation		4.4.3-P
5.2.1.1	Response to command words		
5.2.1.1.1	RT response to command words		4.3.3.6-P
5.2.1.1.2	RT-RT response to command words		4.3.3.6-P
5.2.1.2	Intermessage gap		
5.2.1.2.1	Minimum time		4.3.3.7-P
5.2.1.2.2	Transmission rate		4.3.3.7-P
			4.3.3.8-R
5.2.1.3	Error injection		4.3.3.5.1.6-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.1-R
			4.4.3.3-R
			4.4.3.5-R
			4.4.3.6-R
5.2.1.3.1	Parity		4.3.3.5.1.6-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.1-R
			4.4.3.3-P
5.2.1.3.1.1	Transmit command word		4.3.3.5.1.6-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.3-P
			4.4.3.5-R
			4.4.3.6-R
5.2.1.3.1.2	Receive command word		4.3.3.5.1.6-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.3-P
			4.4.3.5-R
			4.4.3.6-R
5.2.1.3.1.3	Receive data word		4.3.3.5.1.6-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.1-R
			4.4.3.3-P
			4.4.3.5-R
			4.4.3.6-R
5.2.1.3.2	Word length		4.3.3.4-R
			4.3.3.5.3.3-P
			4.4.1.1-R
			4.4.3.1-R
			4.4.3.3-P

100-40

Test Plan		MIL-STD-1553B
5.2.1.3.2.1	Transmit command word	4.3.3.4-P
		4.3.3.5.3.3-P
		4.4.1.1-R
		4.4.3.3-P
		4.4.3.5-R
5.2.1.3.2.2	Receive command word	4.3.3.4-P
		4.3.3.5.3.3-P
		4.4.1.1-R
		4.4.3.3-P
		4.4.3.6-R
5.2.1.3.2.3	Receive data words	4.3.3.4-P
		4.3.3.5.3.3-P
		4.4.1.1-R
		4.4.3.6-R
5.2.1.3.3	Bi-phase encoding	4.3.3.2-P
		4.3.3.5.3.3-R
		4.4.1.1-R
		4.4.3.5-R
		4.4.3.6-R
5.2.1.3.3.1	Transmit command word	4.3.3.2-P
		4.3.3.5.3.3-R
		4.4.1.1-R
		4.4.3.3-R
5.2.1.3.3.2	Receive command word	4.3.3.2-P
		4.3.3.5.3.3-R
		4.4.1 .1 -R
5.2.1.3.3.3	Receive data word	4.3.3.2-P
		4.3.3.5.3.3-R
		4.4.1.1-R
		4.4.3.5-R
		4.4.3.6-R
5.2.1.3.4	Sync encoding	4.3.3.5.1.1-P
		4.3.3.5.2.1-P
		4.3.3.5.3.1-P
		4.4.1 .1-R
		4.4.3.3-R
		4.4.3.5-R
		4.4.3.6-R
5.2.1.3.4.1	Transmit command word	4.3.3.5.1.1-P
		4.4.1.1-R
		4.4.3.3-R
5.2.1.3.4.2	Receive command word	4.3.3.5.1.1-P
		4.4.1.1-R
		4.4.3.3-R
5.2.1.3.4.3	Data word	4.3.3.5.2.1-P
		4.4.1.1-R
		4.4.3.6-R

100-41

<u>Test Plan</u>		<u>MIL-STD-1553B</u>
5.2.1.3.5	Message length	4.3.3.5.1.5-P
5.2.1.3.5.1	Transmit command	4.3.3.5.1.5-P
		4.3.3.6.~R
		4.3.3.6.2-R
5.2.1.3.5.2	Receive command	4.3.3.5.1.5-P
		4.3.3.6-R
		4.3.3.6.1-R
5.2.1.3.5.3	Mode command word count error	4.3.3.6-R
5.2.1.3.5.4	RT to RT word count error	4.3.3.5.1.5-P
		4.3.3.6-R
5.2.1.3.6	Contiguous data	4.3.3.6.1-P
		4.4.1.2-R
		4.4.3.5-R
5.2.1.3.7	Terminal fail-safe	4.4.1.3 -P
5.2.1.4	Superseding commands	4.4.3.2-P
5.2.1.5	Required mode commands	30.4.2.1-P
		4.3.3.5.1.7-P
5.2.1.5.1	Transmit status	4.3.3.5.1.7.3-P
5.2.1.5.2	Transmitter shutdown & override	4.3.3.5.1.7.5-P
		4.3.3.5.1.7.6-P
5.2.1.5.3	Reset remote terminal	4.3.3.5.1.7.9-P
		30.4.3-P
5.2.1.6	Data wrap-around	30.7-P
5.2.1.7	RT to RT timeout	30.9-P
		4.3.3.9-P
5.2.1.8	Bus switching	4.6.3-P
		30.2-R
5.2.1.9	Unique address	30.3-P
		4.3.3.5.1.2-R
5.2.2	Optional operation	
5.2.2.1	Optional mode commands	4.3.3.5.1.7-P
5.2.2.1.1	Dynamic bus control	4.3.3.5.1.7.1-P
		4.3.3.5.3.10-R
5.2.2.1.2	Synchronize	
5.2.2.1.2.1	Synchronize without data word	4.3.3.5.1.7.2-P
5.2.2.1.2.2	Synchronize with data word	4.3.3.5.1.7.12-P
5.2.2.1.3	Initiate self-test	4.3.3.5.1.7.4-P
		30.4.4-P
5.2.2.1.4	Transmit bit word	4.3.3.5.1.7.14-P
5.2.2.1.5	Selective transmitter	
	shutdown 8 override	4.3.3.5.1.7.15-P
		4.3.3.5.1 .7.16-P
5.22.1.6	Terminal flag bit inhibit and override	4.3.3.5.1.7.7-P
		4.3.3.5.1.7.8-P
5.2.2.1.7	Transmit vector word	4.3.3.5.1.7.11-P
5.2.2.1.8	Transmit last command	4.3.3.5.1.7.13-P
5.2.2.2	Status word bits	4.3.3.5.3-P
		30.5.2-P
5.2.2.2.1	Service request	4.3.3.5.3.5-P
5.22.2.2	Broadcast command received	4.3.3.5.3.7-P

Test Plan		MIL-STD-1553B
5.2.22.3	Busy	4.3.3.5.3.8-P
		30.5.3-P
5.22.2.4	Subsystem flag	4.3.3.5.3.9-P
5.2.2.2.5	Terminal flag	4.3.3.5.3.11-P
5.2.2.3	Illegal command	4.4.3.4-P
5.2.2.4	Broadcast mode commands	4.3.3.6.7.3-P
		4.3.3.6.7.4-P
5.2.2.4.1	Broadcast synchronize (without data word)	4.3.3.5.1.7.2-P
		4.3.3.6.7.3-P
5.2.2.4.2	Broadcast synchronize(without data word)	4.3.3.5.1.7.12-P
		4.3.3.6.7.4-P
5.2.2.4.3	Broadcast initiate self-test	4.3.3.5.1.7.4-P
		4.3.3.6.7.3-P
5.2.2.4.4	Broadcast transmitter shutdown and override	4.3.3.5.1.7.5-P
		4.3.3.5.1.7.6-P
		4.3.3.6.7.3-P
5.2.2.4.5	Broadcast selective transmitter shutdown and override	4.3.3 5.1.7.15 P
		4.3.3.5.1.7.16-P
5.2.2.4.6	Broadcast terminal flag bit inhibit and override	4.3.3.5.1.7.7-P
		4.3.3.5.1.7.8-P
		4.3.3.6.7.3-P
5.2.2.4.7	Broadcast reset remote terminal	4.3.3.5.1.7.9-P
		4.3.3.6.7.3-P
5.2.2.4.8	Broadcast dynamic bus control	4.3.3.5.1.7.1-P
5.2.2.5	Error injection - broadcast messages	4.3.3.5.3.3-P
		4.3.3.5.3.7-R
		4.4.1.1-R
		4.4.3.1-R
		4.4.3.3-R
		4.4.3.5-R
		4.4.3.6-R
5.2.2.5.1	Parity: BC-RT broadcast	4.3.3.5.1.6-P
		4.3.3.6.7.1-R
5.2.2.5.2	Command word error	4.3.3.5.1.6-P
		4.3.3.5.1-R
5.2.2.5.3	Data word error	4.3.3.5.1.6-P
		4.3.3.5.2-R
5.2.2.5.4	Message length, BC to RT broadcast	4.3.3.6.7.1-P
		4.4.3.6-P
		4.3.3.5.3.7-R
5.3	Noise rejection	(Transformer coupled) 4.5.2.1.2.4-P
		(Direct coupled) 4.5.2.2.2.4-P

100-43

APPENDIX B
MIL-STD-1553B TO TEST PLAN
CROSS REFERENCE

MIL-STD-1553B		Test Plan
4.	General requirements	
4.1	Test 8 operating requirements	none
4.2	Data bus operation	5.2
4.3	Characteristics	
4.3.1	Data form	none
4.32	Bit priority	none
4.3.3	Transmission method	
4.3.3.1	Modulation	none
4.3.3.2	Data code	5.2.1.3.3
4.3.3.3	Transmission bit rate	5.1.1.10
4.3.3.4	Word size	5.2.1.3.2
4.3.3.5	Word formats - command	5.2.1.1
	- data	none
	- status	5.2.2.2
4.3.3.5.1	Command word	5.2.1.1
4.3.3.5.1.1	Sync	5.2.1.3.4.1
4.3.3.5.1.2	Remote terminal address (not 11111)	5.2.1.1
	(11111)	5.2.1.1
4.3.3.5.1.3	Transmit/receive	5.2
4.3.3.5.1.4	Subaddress/mode - subaddress	5.2.1.1
	- mode	5.2.2.1
4.3.3.5.1.5	Data word count/mode code - word count	5.2.1.3.5
	- mode code	5.2.2.1
4.3.3.5.1.6	Parity	5.2.1.3.1
4.3.3.5.1.7	Optional mode control	5.2.2.1
4.3.3.5.1.7.1	Dynamic bus control	5.2.2.1.1
4.3.3.5.1.7.2	Synchronize (without data word)	5.2.2.1.2.1
4.3.3.5.1.7.3	Transmit status word	5.2.1.5.1
4.3.3.5.1.7.4	Initiate self test	5.2.2.1.3
4.3.3.5.1.7.5	Transmitter shutdown	5.2.1.5.2
4.3.3.5.1.7.6	Override transmitter shutdown	5.2.1.5.2
4.3.3.5.1.7.7	Inhibit T/F bit	5.22.1.6
4.3.3.5.1.7.8	Override inhibit T/F flag	5.2.2.1.6
4.3.3.5.1.7.9	Reset remote terminal	5.2.1.5.3
4.3.3.5.1 .7.1 0	Reserved mode codes (01001-01 1 11)	5.2.1.1
4.3.3.5.1.7.11	Transmit vector word	5.2.2.1.7
4.3.3.5.1.7.12	Synchronize (with data word)	5.2.2.1.2.2
4.3.3.5.1.7.13	Transmit last command word	5.2.2.1.8
4.3.3.5.1.7.14	Transmit built-in-test (BIT) word	5.2.2.1.4
4.3.3.5.1.7.15	Selected transmitter shutdown	5.2.21.5
4.3.3.5.1 .7.16	Override selected transmitter shutdown	5.2.2.1.5
4.3.3.5.1.7.17	Reserved mode codes (10110 to 111 11)	5.2.1.1
4.3.3.5.2	Data word	
4.3.3.5.2.1	Sync	5.2.1.3.4.2
4.3.3.5.2.2	Data	none
4.3.3.5.2.3	Parity	4.1
4.3.3.5.3	Status word	5.2.2.2
4.3.3.5.3.1	Sync	4.2

100-44

MIL-STD-1553B		Test Plan
4.3.3.5.3.2	RT address	4.2
4.3.3.5.3.3	Message error bit	5.2.1.3
4.3.3.5.3.4	Instrumentation bit	4.2
4.3.3.5.3.5	Service request bit	5.2.2.2.1
4.3.3.5.3.6	Reserved status bits	4.2
4.3.3.5.3.7	Broadcast command received bit	5.2.2.2.2
4.3.3.5.3.8	Busy bit	5.2.2.2.3
4.3.3.5.3.9	Subsystem flag bit	5.2.2.2.4
4.3.3.5.3.10	Dynamic bus control acceptance bit	5.2.2.1.1
4 33.5.3.11	Terminal flag bit	5.2.2.2.5
4 3.3.5.3.12	Parity bit	4.2
4.3.3.5.4	Status word reset	5.2.2.2
4.3.3.6	Message formats	n/a
4.3.3.6.1	BC to RT transfers	5.2.1.1.1
4.3.3.6.2	RT to BC transfers	5.2.1.1.1
4.3.3.6.3	RT to RT transfers	5.2 1.1.2
4.3.3.6.4	Mode command w/o data word	5.2.2.1
4.3.3.6.5	Mode command with data word (transmit)	5.2.2.1
4.3.3.6.6	Mode command data word (receive)	5.2.2.1
4.3.3.6.7	Optional broadcast command	5.2.1.1
4.3.3.6.7.1	BC to RT transfer (broadcast)	5.2.1.1
4.3.3.6.7.2	RT to RT transfer (broadcast)	5.2.1.1
4 3.3.6.7.3	Mode commands w/o data word (broadcast)	5.2.2.4
4 3.3.6.7.4	Mode commands with data word (broadcast)	5.2.2.4
4.3.3.7	Intermessage gap	5.2.1.2
4.3.3.8	Response time	4.2
		5.1.1.9
4.3.3.9	Minimum no-response time-out	5.2.1.7
4.4	Terminal operation	
4.4.1	Common operation	
4.4.1.1	Word validation	5.2.1.3
4.4.1.2	Transmission continuity	5 21.3.6
4.4.1.3	Terminal failsafe	5.2.1.3.7
4.4.2	Bus controller operator	n/a
4.4.3	Remote terminal	5.2
4.4.3.1	Operation	5.2.1.1
4.4.3.2	Superseding valid commands	5.2.1.4
4.4.3.3	Invalid commands	5.2.1.3
4.4.3.4	Illegal commands	5.2.2.3
4.4.3.5	Valid data reception	5.2.1.3
4 4.3.6	Invalid data reception	5.2.1.3
4 4.4	Bus monitor operation	n/a
4.5	Hardware characteristics	
4.5.1	Data bus characteristics	
4 5.1.1	Cable	n/a
4 5.1.2	Characteristics impedance	n/a
4.5.1.3	Cable attenuation	n a
4.5.1.4	Cable termination	n/a
4.5.1.5	Cable stub requirements	n/a
4.5.1.5.1	Transformer coupled stubs	n/a
4.5.1.5.1.1	Coupling transformer	n/a

MIL-STD-1553B		Test Plan
4.5.1.5.1.1.1	Transformer input impedance	n/a
4.5.1.5.1.1.2	Transformer waveform integrity	n/a
4.5.1.5.1.1.3	Transformer common mode rejection	n/a
4.5.1.5.1.2	Fault isolation	n/a
4.5.1.5.1.3	Cable coupling	n/a
4.5.1.5.1.4	Stub voltage requirements	n/a
4.5.1.5.2	Direct coupled stubs	n/a
4.5.1.5.2.1	Fault isolation	n/a
4.5.1.5.2.2	Cable coupling	n/a
4.5.1.5.2.3	Stub voltage requirements	n/a
4.5.1.5.3	Wiring 8 cabling for EMC	n/a
4.5.2	Terminal characteristics	5.1
4.5.2.1	Terminals with transformer coupled stubs	5.1
4.5.2.1.1	Terminal output characteristics	5.1.1
4.5.2.1.1.1	Output levels	5.1.1.1
4.5.2.1.1.2	Output waveform	5.1.1.2
		5.1.1.3
		5.1.1 4
4.5.2.1.1.3	Output noise	5.1.1.6
		5.1.1.8
4.5.2.1.1.4	Output symmetry	5.1.1.5
4.5.2.1.2	Terminal input characteristics	5.1.2
4.5.2.1.2.1	Input waveform compatibility	5.1.2.1
4.5.2.1.2.2	Common mode rejection	5.1.2.2
4.5.2.1.2.3	Input impedance	5.1.2.3
4.5.2.1.2.4	Noise rejection	5.3
4.5.2.2	Terminals with direct coupled stubs	5.1
4.5.2.2.1	Terminal output characteristics	5.1.1
4.5.2.2.1.1	Output levels	5.1.1.1
4.5.2.2.1.2	Output waveform	5.1.1.2
		5.1.1.3
		5.1.1.4
4.5.2.2.1.3	Output noise	5.1.1.6
		5.1.1.8
4.5.2.2.1.4	Output symmetry	5.1.1.5
4.5.2.2.2	Terminal input characteristics	5.1.2
4.5.2.2.2.1	Input waveform compatibility	5.1.2.1
4.5.2.2.2.2	Common mode rejection	5.1.2.2
4.5.2.2.2.3	Input impedance	5.1.2.3
4.5.2.2.2.4	Noise rejection	5.3
4.6	Redundant data bus requirements	5.2.1.8
4.6.1	Electrical isolation	5.1.1.7
4.6.2	Single event failures	n/a
4.6.3	Dual standby redundant data bus	5.2.1.8
4.6.3.1	Data bus activity	5.2.1.8
4.6.3.2	Superseding valid commands	5.2.1
30.	GENERAL REQUIREMENTS	n/a
30.1	Option selection	n/a
30.2	Application	n/a
30.3	Unique address	5.2.1.9

100-46

30.4	Mode codes	n/a
30.4.1	Subaddress/mode	5.2.1.5
MIL-STD-1553B		Test Plan
30.4.2.1	Remote terminal required mode codes	5.2.1.5
30.4.2.2	Bus controller required mode codes	n/a
30.4.3	Reset remote terminal	5.2.1.5.3
30.4.4	Initiate RT self-test	5.2.2.1.3
30.5	Status word bits	n/a
30.5.1	Information content	5.1.1.8.2
30.5.2	Status bit requirements	5.2.2.2
30.5.3	Busy bit	5.2.2.2.3
30.6	Broadcast	5.2.1.1
30.7	Data wrap-around	5.2.1.6
30.8	Message formats	n/a
30.9	RT to RT validation	5.2.1.7
30.10	Electrical characteristics	n/a
30.10.1	Cable shielding	n/a
30.10.2	Shielding	n/a
30.10.3	Connector polarity	n/a
30.10.4	Characteristic impedance	n/a
30.10.5	Stub coupling	n/a
30.10.6	Power on/off noise	5.1.1.8

APPENDIX C

TEST PLAN CHANGES FOR MIL-STD-1 553B ONLY RTs

For remote terminals designed to only comply with MIL-STD-1 553B, the following changes shall be made for the pass criteria in this document.

1. The following paragraphs are optional and are subject to the same requirements as 5.2.2.

5.2.1.5	Required mode commands
5.2.1.5.1	Transmit status
5.2.1.5.2	Transmitter shutdown and override
5.2.1.5.3	Reset remote terminal
5.2.1.6	Data wrap-around
5.2.1.7	RT to RT timeout
5.2.1.8	Bus switching
5.2.1.9	Unique address

2. For the following paragraphs, the pass criteria for step 2 shall be changed to delete the words "(with BUSY bit reset)..

5.2.2.1.3	Initiate self-test
5.2.1.5.3	Reset remote terminal
5.2.2.4.3	Broadcast initiate self-test
5.2.2.4.7	Broadcast reset remote terminal

3. For 5.1.1.8.1 Power on/off noise, the pass criteria shall not be defined in this document.

4. For the following paragraphs, the requirement to implement both mode code indicators of all zeros and all ones is optional and subject to the same requirements as 5.2.2. The RT must meet the pass criteria for either all zeros or all ones, but is not required to meet both.

5.2.1.5	Required mode commands
5.2.2.1	Optional mode commands
5.2.2.4	Broadcast mode commands

5. The following note shall be added to the end of 5.2.1.3.

Note: If transmit status mode command is not implemented, then transmit last command mode command shall be used. U neither mode command is implemented, then step 3 shall be deleted.

APPENDIX D

TEST PLAN CHANGES FOR MIL-STD-1553B, NOTICE 1 RTs

For remote terminals designed to comply with MIL-STD-1553B, Notice 1, the following changes shall be made for the pass criteria in this document.

1. The following paragraphs are optional, and are subject to the same requirements as 5.2.2.

    5.2.1.5   Required mode commands
    5.2.1.5.1   Transmit status
    5.2.1.5.2   Transmitter shutdown and override
    5.2.1.5.3   Reset remote terminal
    5.2.1.6   Data wrap-around
    5.2.1.7   RT to RT timeout
    5.2.1.9   Unique address

2. For the following paragraphs, the pass criteria for step 2 shall be changed to delete the words "(with BUSY bit reset)."

    5.2.2.1.3   Initiate self-test
    5.2.1 .5.3   Reset remote terminal
    5.2.2.4.3   Broadcast initiate self-test
    5.2.2.4.7   Broadcast reset remote terminal

3. For 5.1.1.8, the pass criteria shall not be defined in this document.

4. For the following paragraphs the requirement to implement mode code indicator of all ones is optional and subject to the same requirements as 5.2.2.

    5.2.1.5   Required mode commands
    5.2.2.1   Optional mode commands
    5.2.2.4   Broadcast mode commands

5. The following note shall be added to the end of 5.2.1.3.

100-49
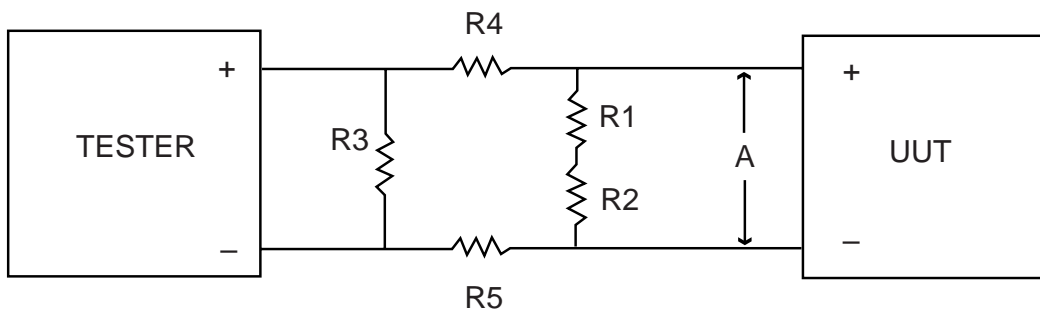
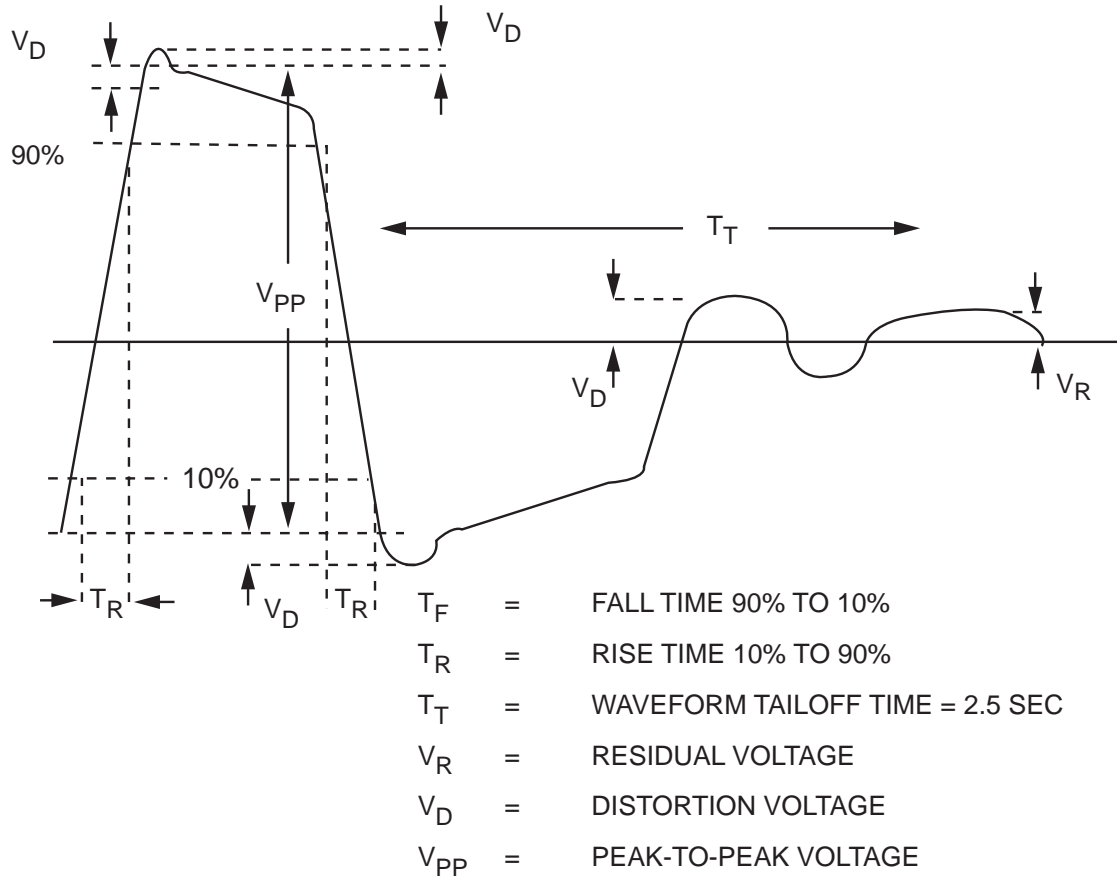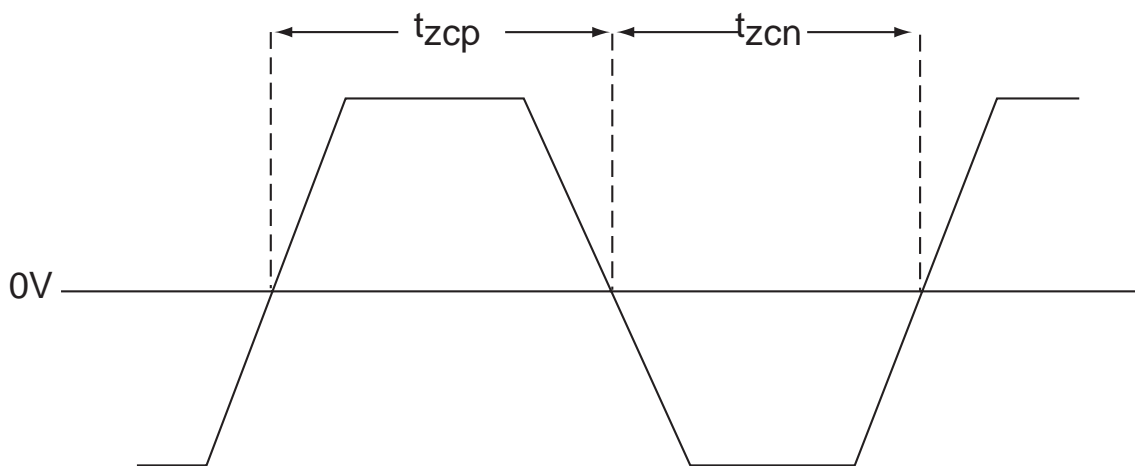|	DIRECT COUPLED	TRANSFORMER COUPLED
	35 ohms ± 2%	70 ohms ±2%
R1, R2	20	46.5
R3, R4, R5	100	93.1

GENERAL RESISTOR PAD CONFIGURATION. FIGURE 1A
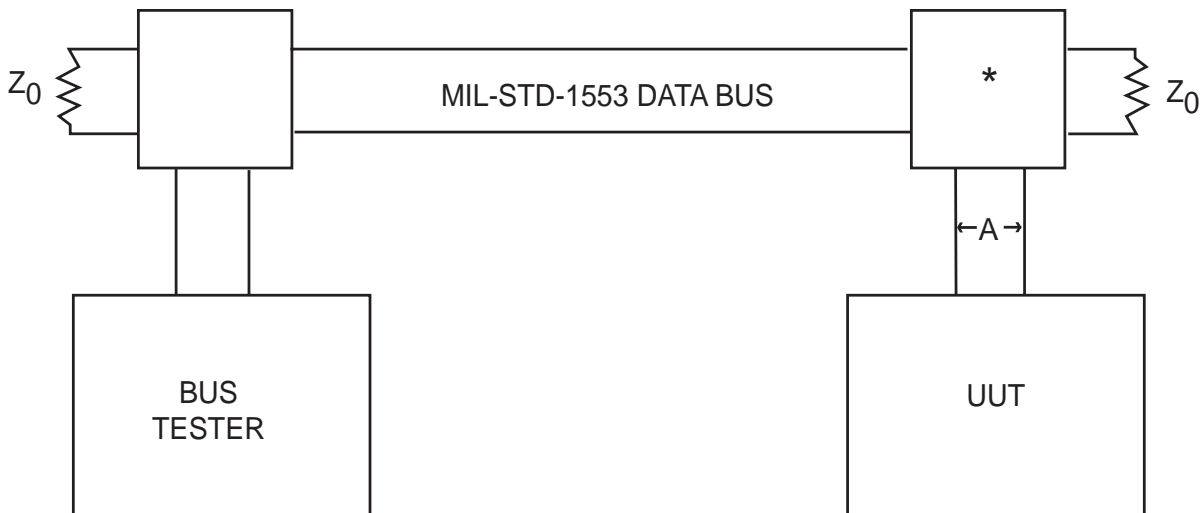
$*$ NOTE: REFERENCE FIGURES 9A AND 10A OF
MIL-STD-1553 FOR DATA BUS INTERFACE
COUPLING. REFERENCE FIGURE 10B OF THIS
TEST PLAN FOR SUGGESTED CABLE TYPE, BUS
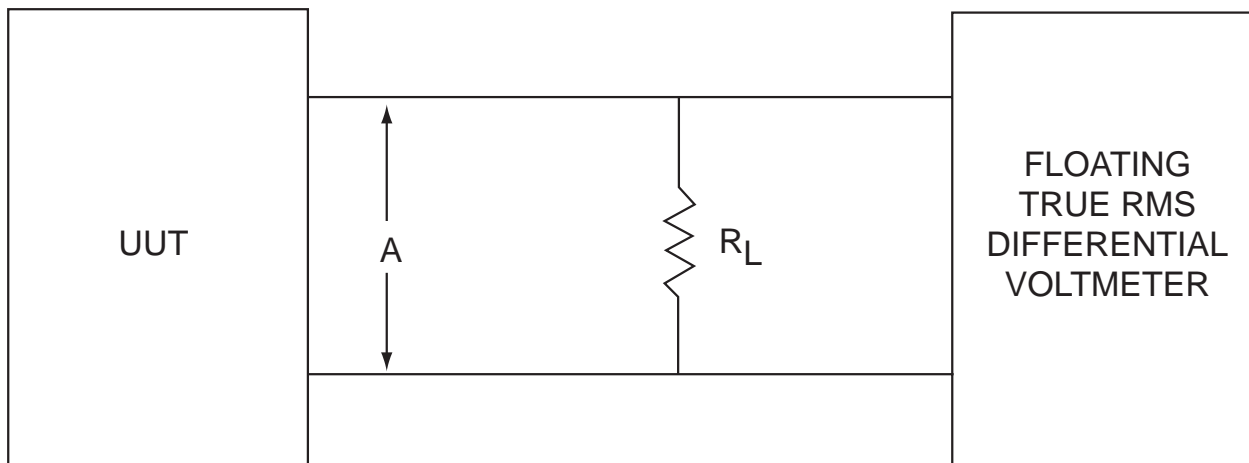AND SUB LENGTHS, ETC.
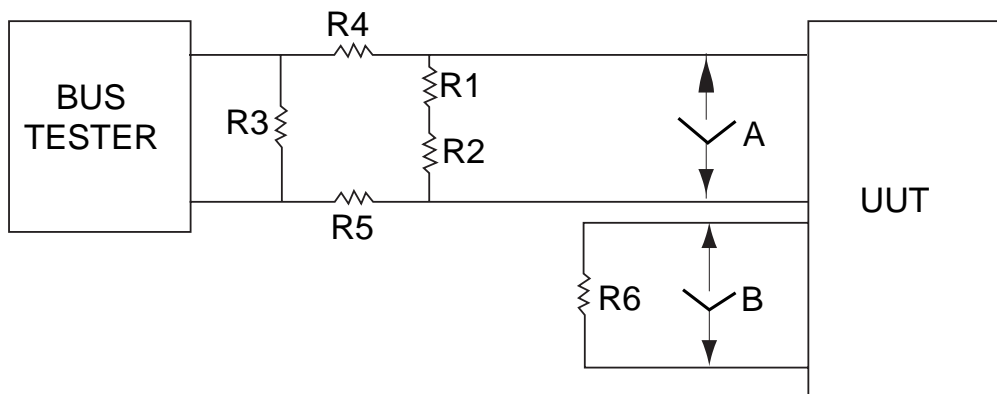
GENERAL BUS CONFIGURATION FIGURE 1B

100-50

T_F	=	FALL TIME 90% TO 10%
T_R	=	RISE TIME 10% TO 90%
T_T	=	WAVEFORM TAILOFF TIME = 2.5 SEC
V_R	=	RESIDUAL VOLTAGE
V_D	=	DISTORTION VOLTAGE
V_{PP}	=	PEAK-TO-PEAK VOLTAGE

WAVEFORM MEASUREMENTS. FIGURE 2.

ZERO CROSSING INTERVAL MEASUREMENTS. FIGURE 3.

100-51

TRANSFORMER COUPLED R_L = 70.0 ohms ± 2%
DIRECT COUPLED R_L = 35.0 ohms ± 2%

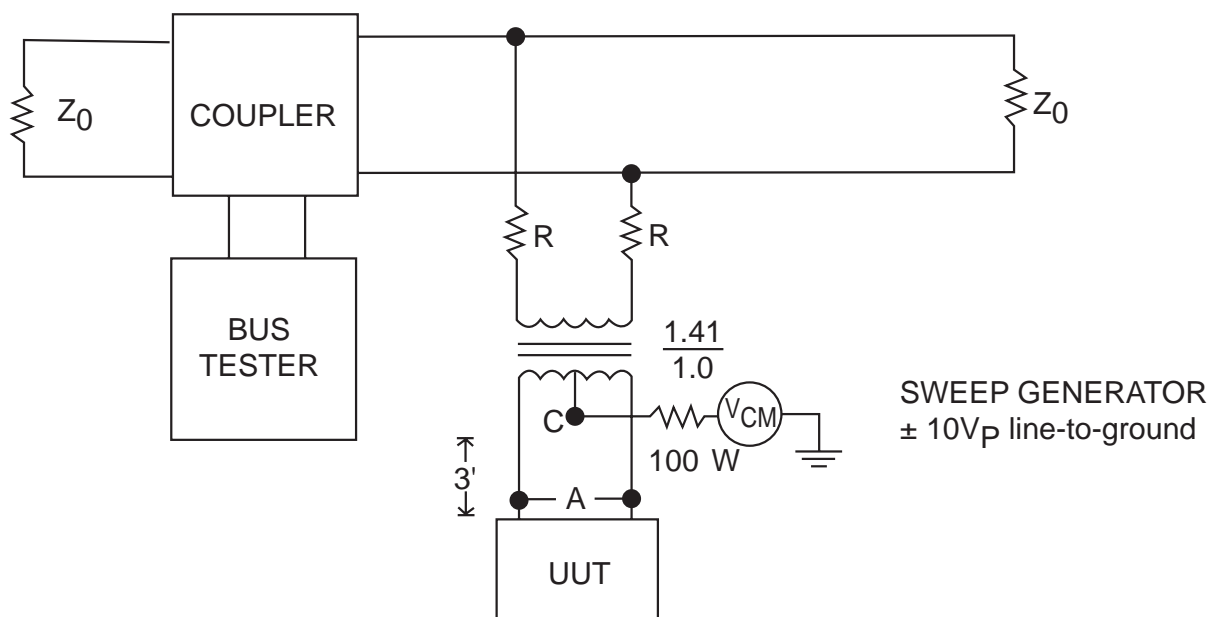**OUTPUT NOISE CONFIGURATION. FIGURE 4.**

	DIRECT COUPLED 35 OHMS ±2%	TRANSFORMER COUPLED 70 OHMS ±2%
R1, R2	20	46.5
R3, R4, R5	100	93.1
R6	35	70

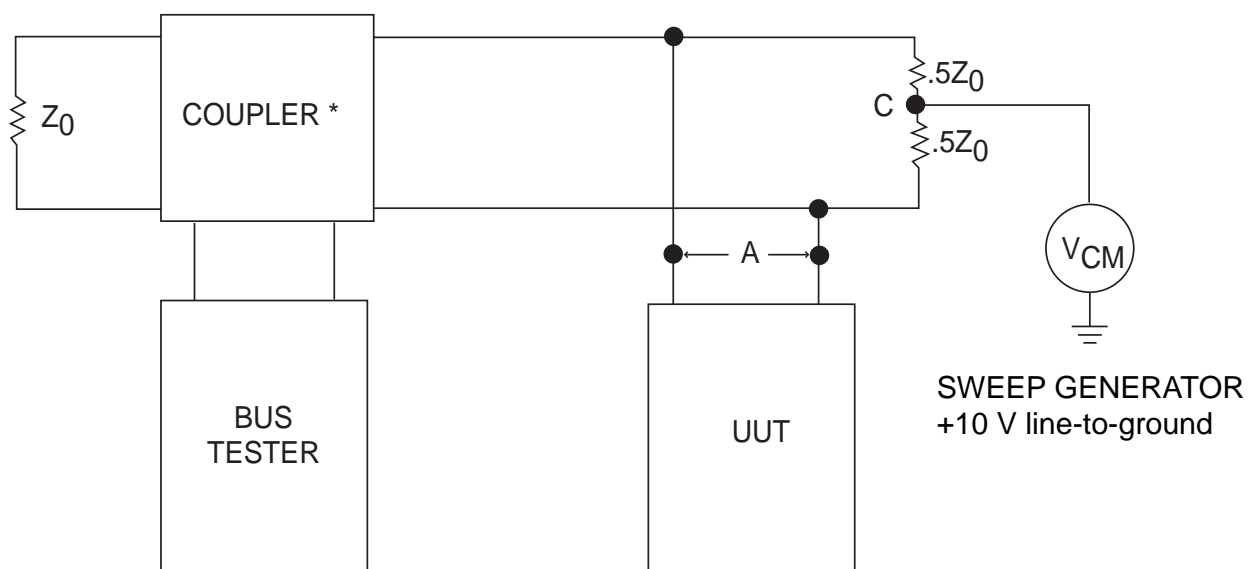$$dB = 20 \, LOG \, \frac{V_A}{V_B}$$

OUTPUT ISOLATION. FIGURE 5.

100-52

$R = 0.75 \ Z_0 \ \pm 2\%$

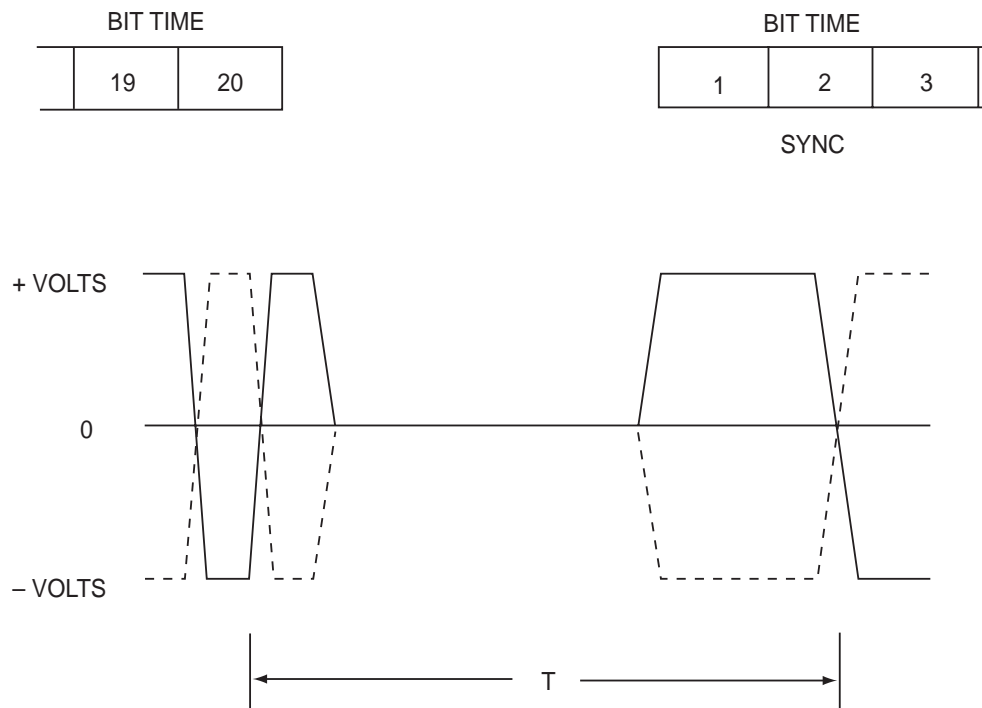Z   – Selected Cable Nominal Characteristic Impedance

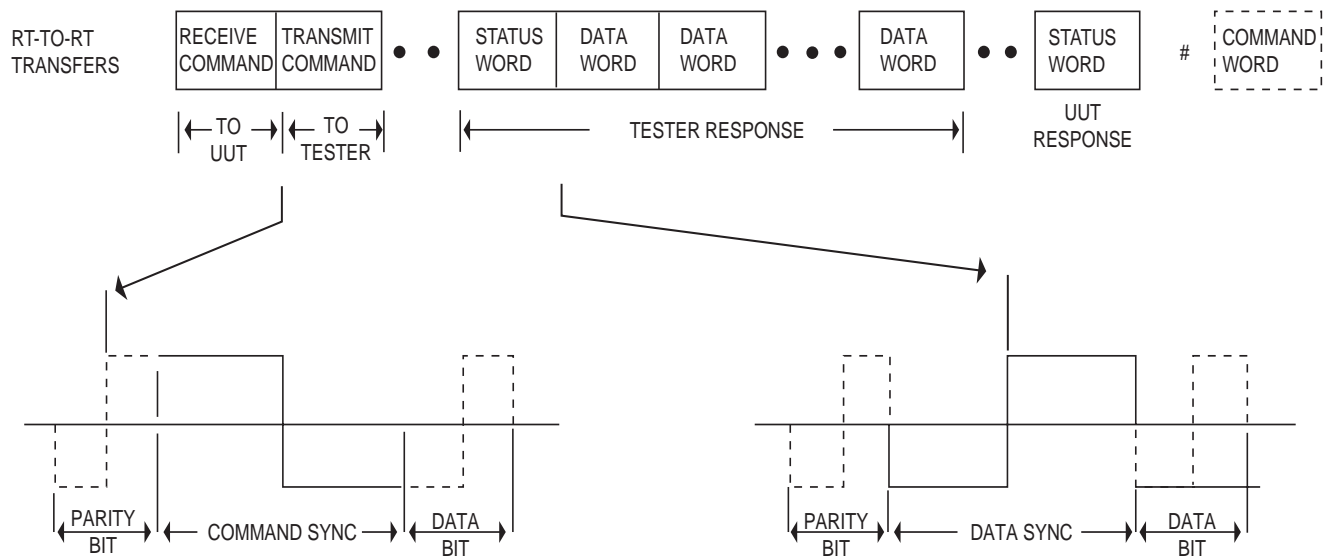**TRANSFORMER COUPLED. COMMON MODE CONFIGURATION. FIGURE 6A.**

* Transformer Coupled Stub Must Be Used

Z_0 Selected Cable Nominal Characteristic Impedance
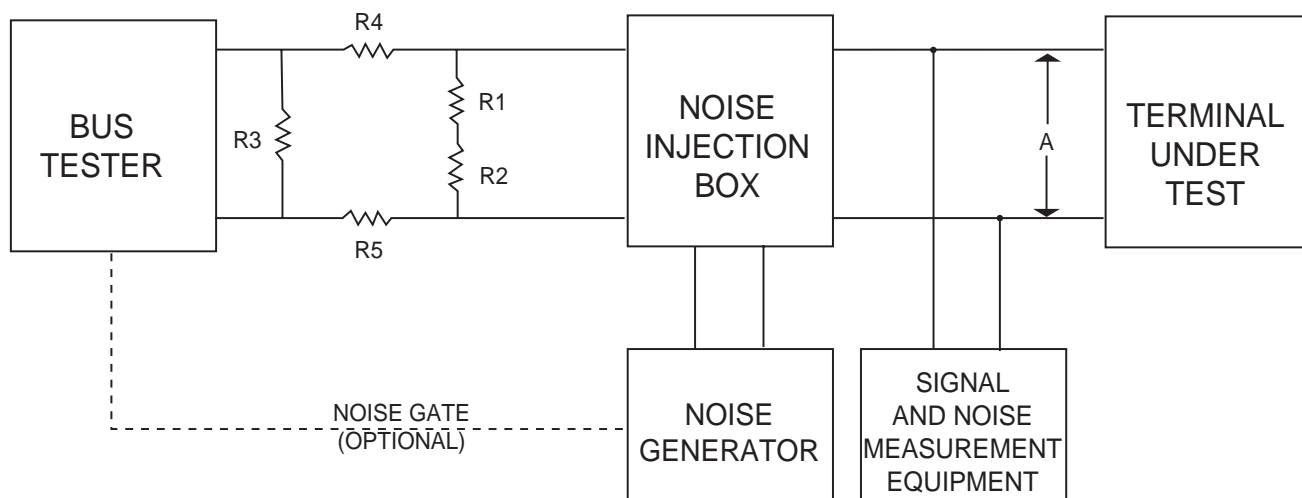
DIRECT COUPLED. COMMON MODE CONFIGURATION. FIGURE 6B.

100-53

BIT TIME

19	20

BIT TIME

1	2	3

SYNC

+ VOLTS

0

− VOLTS

T

GAP TIME MEASUREMENT. FIGURE 7.

RT-TO-RT
TRANSFERS

RECEIVE COMMAND	TRANSMIT COMMAND		STATUS WORD	DATA WORD	DATA WORD		DATA WORD		STATUS WORD	#	COMMAND WORD

TO UUT — TO TESTER

TESTER RESPONSE

UUT RESPONSE

PARITY BIT — COMMAND SYNC — DATA BIT

PARITY BIT — DATA SYNC — DATA BIT

T = 54 to 60 μsec measured as shown.
Delay tester response until UUT response goes away.

RT—RT TIMEOUT MEASUREMENT. FIGURE 8.

100-54

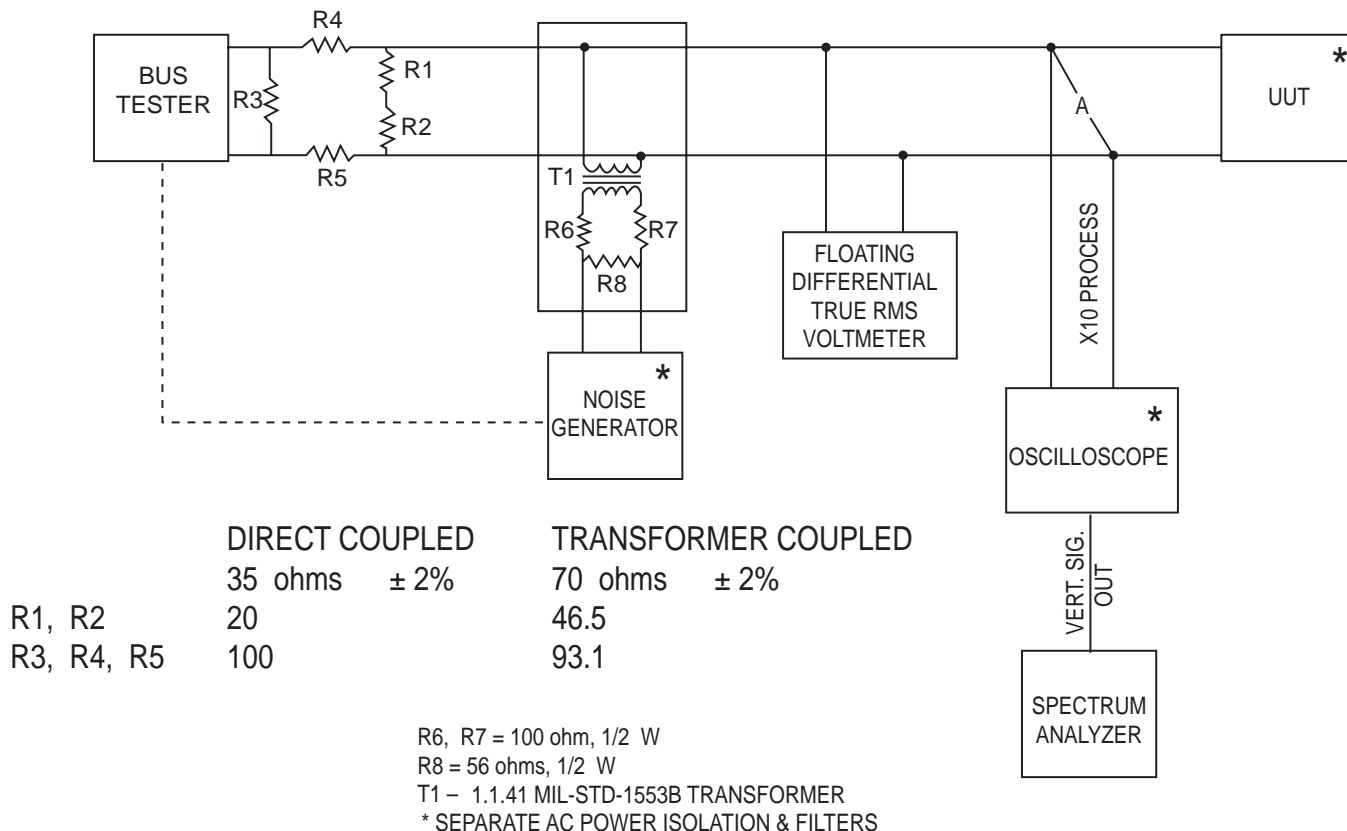DIRECT COUPLED   TRANSFORMER COUPLED

       35 ohms   ± 2%   70 ohms   ± 2%

R1, R2   20   46.5

R3, R4, R5   100   93.1

CONFIGURATION FOR NOISE REJECTION TEST. FIGURE 9A.

DIRECT COUPLED   TRANSFORMER COUPLED

35 ohms   ± 2%   70 ohms   ± 2%

R1, R2   20   46.5

R3, R4, R5   100   93.1

R6, R7 = 100 ohm, 1/2 W
R8 = 56 ohms, 1/2 W
T1 – 1.1.41 MIL-STD-1553B TRANSFORMER
* SEPARATE AC POWER ISOLATION & FILTERS

SUGGESTED CONFIGURATION FOR NOISE REJECTION TEST. FIGURE 9B.

100-55

CONFIGURATION FOR NOISE REJECTION TEST. FIGURE 10A.

BUS CABLE: MIL-C-17/176-00002

Z0 = 70 ohms, 1/2 W   L1 < 1 foot
R1 = 160 ohms, 1/2 W   L2 = 200 feet
R2 = 160 ohms, 1/2 W   L3 < 1 foot
R3 = 50 ohms, 1/2 W   LS < 3 feet

* SEPARATE AC POWER ISOLATION & FILTERS

SUGGESTED CONFIGURATION FOR NOISE REJECTION TEST. FIGURE 10B.

100-56

VII.
DATA BUS
PRODUCT INFORMATION

2 YEAR
WARRANTY

● TRANSCEIVERS — Single

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-63102	Universal Transceiver	1.26 x 1.26 x 0.17"	Universal transceiver conforms to MIL-STD-1553 A/B and McDonnell Douglas A-3818, A-5232, A-4905, and A-5690. Use BUS-27765 XFMR.	VII-149
BUS-63104	Transceiver	1.26 x 1.26 x 0.17"	Conforms to MIL-STD-1553 A/B. Improved filtering to enhance bit error rate. Use BUS-27765 XFMR.	
BUS-63105 (-15 V) BUS-63107 (-12 V)	Transceiver	24-pin DDIP 1.4 x 0.8 x 0.2"	Monolithic Transceiver. Listed on MIL DWG.No.5962-86049-02ZC. Conforms to MIL-STD-1553 A/B. High S/N ratio, internal filtering. Drives Harris 15530 directly. SEAFAC tested. Eliminates non-standard part approval when specified by the Military Drawing number. Uses BUS-25679(-15 V) and BUS-29854(-12 V) XFMR.	
BUS-63115 BUS-63117			Same as BUS-63105 but has RX and inverted RX outputs. Same as BUS-63107 but has RX and inverted RX outputs	

● TRANSCEIVERS — Dual

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-63125 (-15 V) BUS-63127 (-12 V) BUS-63126 (-15 V) BUS-63128 (-12 V)	Dual Transceiver	36-pin DDIP 1.90 x 0.78 x 0.20" 36-pin flat pack 1.905 x 0.785 x .165"	Dual redundant monolithic transceivers in a single hybrid. Conforms to MIL-STD-1553 A/B. High S/N ratio, internal filtering. Drives Harris 15530 directly. Complete isolation, including separate power and signal connections. SEAFAC tested. Eliminates non-standard part approval when specified by the MIL DWG. No. 5962-87579-02XC. Uses BUS-25679(-15 V) and BUS-29854(-12 V) XFMR.	VII-149
BUS-63135 BUS-63137			Same as BUS-63125 but has RX and inverted RX outputs. Same as BUS-63127 but has RX and inverted RX outputs	

● COMPLETE TERMINALS

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
★ BU-61688/89	Miniature Advanced Communication Engine (Mini-ACE) PLUS	1" Square MCM (Multi-Chip Module)	The Mini-ACE-PLUS integrates two 5 volt only transceivers, complete 1553A/B Notice 2 / STANAG3838 BC/RT/MT Protocol, Memory Management, Processor Interface Logic, and 64K x 16 words of RAM in a choice of Pin Grid Array (PGA) or Quad Flat Pack packages. This product operates with 10/12/16/20 MHz clock input.	VII-59
BU-65170/61580: 'X1 (-15 V) 'X2 (-12 V) 'X3 (+5 V) 'X6 (+5 V with TX_INH signals) '65171/61581XX (with programmable RT Address option)	Advanced Communication Engine (ACE)	1.9 x 1.0 x 0.165" 70-pin ceramic DIP, flat pack	The BU-65170 is a complete 1553A/B Notice 2/STANAG 3838 Integrated RT. Contains dual low-power transceiver (12 or 16 MHz), encoder/decoder, RT protocol, memory management logic, interrupt logic, processor interface logic, and 4K x 16 internal RAM. Flexible processor/memory interfacing and software operation, internal self-test. The BU-61580 is a Complete 1553A/B Notice2/STANAG 3838 Integrated BC/RT/MT. Contains dual low-power transceiver, encoder/decoder, protocol including simultaneous RT/MT mode, memory management logic, interrupt logic, processor interface logic, and 4K x 16 internal RAM. Flexible processor/memory interfacing and software operation, internal self-test. (DESC #5962-93065)	VII-17
BU-61585/6: 'X3 'X6	(ACE)	1.9 x 1.0 x 0.165" 70-pin DIP or flat-pack package	Meets low-profile height requirements. 70-pin fully cofired, 12K x 16 internal RAM; RAM may optionally be configured as 8K x 17, providing parity generation/checking on all RAM accesses.	VII-17
BU-61590X5	Universal ACE	2.0 x 1.87 x 0.210" 78-pin ceramic DIP or 2.1 x 1.8 x 0.210" 78-pin ceramic flat pack package	Contains dual sinusoidal low-power transceiver, encoder/decoder, protocol including simultaneous RT/MT mode, memory management logic, interrupt logic, processor interface logic, and 4K x 16 internal RAM. Provides compliance to MIL-STD-1553A/B Notice 2; STANAG 3838; McAir A3818, 5232, and A5690; G.D. and Grumman protocols.	VII-12

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

●COMPLETE TERMINALS (continued)

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
★ BU-65179	Miniature Advanced Communication Engine (Mini-ACE) PLUS	1" Square MCM (Multi-Chip Module)	The Mini-ACE-PLUS integrates two 5 volt only transceivers, complete 1553A/B Notice 2 / STANAG3838 BC/RT/MT Protocol, Memory Management, Processor Interface Logic, and 4K x 16 words of RAM in a choice of Pin Grid Array (PGA) or Quad Flat Pack packages. This product supports RT Auto Boot and 10/12/16/20 MHz clock input..	VII-59
BU-65178/61588	Mini-ACE	1" Square MCM (Multi-Chip Module)	The Mini-ACE integrates two 5 volt only transceivers, protocol, memory management, processor interface logic, and 4K x 16 words of RAM in a choice of pin grid array (PGA) or quad flat pack packages.	VII-59
BU-65620P0		1.589 x 1.589 x 0.112" 144-pin ceramic PGA	1553A/B Notice 2/McAir/STANAG 3838 Integrated BC/RT/MT digital controller. 4K x 16 internal RAM. May be interfaced to external electrical (1553) and/or fiber optic (1773) transceivers. Provides compliance to 1553A/B/McAir, G.D., and Grumman protocols. Supports up to 64K x 17 of buffered shared RAM with parity generation/checking capability.	
BUS-61553 (-15 V) BUS-61554 (-12 V) BUS-61555 (+5 V)	Advanced Integrated Multiplex (AIM) Hybrid AIM-HY	2.1 x 1.87 x 0.210" 78-pin ceramic DIP	Complete MIL-STD-1553 Bus Controller (BC), Remote Terminal (RT), or Bus Monitor (MT) device. Contains dual low power transceivers, complete BC/RT/MT protocol logic, MIL-STD-1553-to-host processor interface unit, and an 8K x 16 shared RAM. Provides memory mapped 1553 interface, on-line and off-line self-test, and is SEAFAC tested. Eliminate non-standard part approval when specifiied by the MIL DWG No. 5962-88692-01XC. Uses BUS-25679(-15 V), BUS-29854(-12 V), and BUS-41429(5 V) XFMR.	VII-145
BUS-61556 (-15 V)			- Same as above but it is transceiverless and is used with BUS-63102II.	
BUS-61563 (-15 V) BUS-61564 (-12 V) BUS-61565 (+5 V)		2.2 x 1.6 x 0.17" (flat pack)	- Flat pack version of BUS-61553. - Flat pack version of BUS-61554. - Flat pack version of BUS-61555	
BUS-61559 SERIES (-15 V) BUS-61560 (-12 V)	Enhanced Advanced Integrated MUX (AIM) Hybrid AIM-HY'ER	2.1 x 1.9 x 0.25" (DDIP)	Complete MIL-STD-1553B Notice 2 Bus Controller (BC), Remote Terminal (RT), or Bus Monitor (MT) device. Supports STANAG 3910. Functionally and software compatible to BUS-61553 Series including data and address buffers selection, enhanced RT Subaddress Memory Management, and additional interrupts, internal illegalization, and time tag.	VII-147
BUS-61569 (-15 V) BUS-61570 (-12 V) BUS-61571 (+5 V)		2.2 x 1.6 x 0.17" (flat pack)		
BUS-65112	Dual Redundant RT	2.1 x 1.9 x 0.25" (DDIP)	MIL-STD-1553 dual redundant RT includes transceivers, encoder/decoder and protocol. Supports 13 mode codes and has DMA type interface handshake.	VII-162
BUS-65117		2.2 x 1.6 x 0.17" (flat pack)	SEAFAC tested. Eliminated non-standard part approval when specified by the Military Drawing number (DESC) 5962-87535-01XC. Uses BUS-25679 XFMR.	
BUS-65142	Dual Redundant RT (Space Level Available)	1.9 x 2.1 x 0.25" (DDIP)	Complete dual redundant RT package in either DDIP or flat pack hybrid packages. Includes CMOS-SOS Monolithic RT protocol and bipolar low power Mark II transceivers, data buffers, timing control logic, supports mode codes, 16 MHz decoders that offer improved noise rejection and zero crossing detection, and has SEAFAC tested components. Uses BUS-25679 XFMR.	VII-166
BUS-65144		1.6 x 2.2 x 0.17" (Kovar flat pack)		
BUS-65153	Complete, dual redundant MIL-STD-1553B & MIL-STD-1760 Remote Terminal "STIC"	1.90 x 1.0 x 0.240" 70-pin DIP ceramic package	Provides the output voltage level required for MIL-STD-1760. Contains two (2) low power transceivers and a DDC custom designed protocol chip. This chip includes dual encoder/decoder, RT protocol logic, 3-state data buffers, and DMA transfer control logic. Supports all 13 dual redundant mode codes, programmable. Parallel data transfers (8-bit or 16-bit) are accomplished via a DMA type handshake. Programmable 16 or 12 MHz clock. No external processor required.	VII-173
BUS-65163/4		70-pin flat pack		

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

● COMPLETE TERMINALS — Rad Hard

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BU-61582	Space Level Advanced Communication Engine (SP'ACE)	1.9 x 1.0 x 0.215" (70-pin DDIP) 1.9 x 1.0 x 0.215" (70-pin flat pack)	The BU-61582 is available as a complete space level 1553A/B Notice 2/STANAG 3838 Integrated BC/RT/MT. Contains dual low-power transceiver, encoder/decoder, BC/RT/MT protocol including simultaneous RT/MT mode, memory management logic, interrupt logic, processor interface logic, and 16K x 16 internal RAM. Flexible processor/memory interfacing and software operation, and internal self-test. The SP'ACE can be supplied either with or without an internal transceiver. Since the transceiver is bipolar, and the digital logic and RAM are implemented using Honeywell's RICMOS process, the hybrids are inherently immune to latchup, and provide a gamma total dose rate of 1 MRad.	VII-8
BUS-65142/3-616 or BUS-65144-616	Dual Redundant RT	2.0 x 1.87 x 0.210" 78-pin ceramic DIP or 2.1 x 1.8 x 0.210" 78-pin ceramic flat pack package	Complete MIL-STD-1553 RT integrated Remote Terminal for space and other rad hard/hi-rel applications. Contains dual transceivers, RT protocol, memory management and processor interface logic. The -616 version includes bipolar transceivers, CMOS/SOS logic provide total dose rad hardness of .5 MRAD. Other rad hardness parameters include latchup immunity, and SEU rate (for -616 version). Hi-rel screening is available, per customer requirements. RICMOS 4 version supports 1 MRAD total dose.	VII-166

● CARD ASSEMBLIES

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BU-65528 **BU-65527**	VME/VXI MIL-STD-1553B BC/RT/MT Interface Militarized	6.3 x 9.2 x 0.6" VXI B Size Card	Provides interfacing between multiple, dual redundant MIL-STD-1553A/B Notice 2 data buses and the parallel VMEbus/VXIbus through the use of up to four BU-61586 "ACE" hybrids. Optionally, the BU-65528 can be configured for up to four channels. Software controls each independent channel as a BC/RT or MT, and each channel contains 12K x 16 of shared on-board RAM.	VII-66
BU-65529 (**BU-65539** recommended for new designs)	IBM® PC AT BC/ RT/ MT	6.5 x 4.5" Half-Size IBM® PC AT Card	Dual redundant BC,RT, or MT on a half-slot IBM AT interface card. Supports all MIL-STD-1553 Message formats, 13 mode codes, and built-in-test capability. It's AIM-HY'er features and 4K x 16 shared RAM interface minimizes CPU overhead.	VII-94
BU-65539	IBM® PC AT MIL-STD-1553B BC/ RT/ MT Interface	6.8 x 4.5" Half-Size IBM® PC AT Card	Provides interfacing between a dual redundant MIL-STD-1553A/B Notice 2 data bus using "ACE" hybrids. Software controls the BU-65539's operation as a Bus Controller (BC), Remote Terminal (RT) or Bus Monitor (MT). Card has 32K x 16 of shared on-board RAM.	VII-96
BU-65550 **BU-65551**	PCMCIA MIL-STD-1553 BC/ RT/ MT Interface PCMCIA, Ruggedized	Type II PCMCIA Module Type II with Internal Transformers	Provides interfacing between a dual redundant MIL-STD-1553A/B Notice 2 data bus and a PCMCIA socket. The BU-65550 features the BU-61586 "ACE" hybrid. Software controls the operation of the module as a BC/RT or MT and the BU-65550 contains 12K x 16 shared RAM. The PCMCIA interface includes 256 bytes of attribute memory as well as the four standard PCMCIA card configuration registers. Ruggedized Type II PCMCIA with internal transformers, 4K x 16 RAM and -25 to +70°C operating range.	VII-120
BUS-65517II	IBM® PC Card for Test and Simulation	13.125 x 4.2" IBM® PC Card	Simultaneously simulates a Bus Controller, and active Monitor and up to 30 Remote Terminals. Provides comprehensive Error injection with Bit/Word resolution. Fully dual redundant. Includes user-friendly menu driven software.	VII-180

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

●CARD ASSEMBLIES (continued)

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-65518	IBM® Half-Size PC Card MIL-STD-1553 Simulator & Tester	6.75 x 4.2" Half-Size IBM® PC Card	Provides full, intelligent interfacing between the serial dual redundant MIL-STD-1553 data bus and the IBM PC. Software allows the BUS-65518 to simultaneously simulate a Bus Controller (BC), multiple Remote Terminal Units (RTUs), and/or a selectable, triggerable Bus Monitor (MT). Can be either direct or transformer coupled to an external 1553 bus.	VII-183
BUS-65536	VXI/VME MIL-STD-1553 C-Size Simulator & Tester		Provides full, intelligent interfacing between the serial dual redundant MIL-STD-1553 data bus and the VXI Bus. Software allows the BUS-65536 to simultaneously simulate a Bus Controller (BC), multiple Remote Terminal Units (RTUs), and/or a selectable, triggerable Bus Monitor (MT). Can be either direct or transformer coupled to an external 1553 bus.	VII-195

●TESTERS

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-68005	Data Bus Exerciser	17 x 14.5 x 3.5" Instrument	Full function simulation of RTU and BC operation; implements protocol test of SAE AE-9 (see BUS-69005) proposed production test plan; has 1553 A/B System Tests; Real-Time Monitor output; programmable transmitter output level and channel A/B selection; can do superceding command test; error generation/detection; RS-232 and GPIB-488 external control.	Contact Factory
BUS-68010	Data Bus Tester	8.5 x 3.5 x 9.3" Instrument	Low cost, bench top error generating and detecting instrument for functional testing of MIL-STD-1553 Data Bus systems. Operational as Bus Controller, Remote Terminal, and Bus Monitor.	Contact Factory

●TEST EQUIPMENT COMPONENTS

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-8559	Variable Output Transceiver	24-pin DDIP 1.4 x 0.8 x 0.20"	Conforms to MIL-STD-1553 A/B. Has variable output level for use in test applications. Receiver has internal filtering for high S/N ratio. Uses BUS-25679 XFMR.	Contact Factory

●TRANSFORMERS

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
BUS-25679 BUS-27765 BUS-29854 BUS-41429	Bus Isolation Transformer	0.63 x 0.63 x 0.25" 0.63 x 0.63 x 0.30" 0.63 x 0.63 x 0.275" 0.63 x 0.63 x 0.220" Modules	Provides all of the turns ratio configurations, component isolation, and common mode rejection ratio characteristics necessary for MIL-STD-1553 A/B. Modular units that are multitapped to accommodate existing system configurations.	VII-243
B-2200 / B-2300 / B-3200 Series* *	Interface Transfomers (Beta* *)	Available in 0.63 x 0.63 x 0.30" 0.35 x 0.50 x 0.25" 0.63 x 0.63 x 0.275" Modules	Interface transformers listed on DESC Spec. No. 21038/27. Designed to couple MIL-STD-1553 A/B compatible drivers to the bus. **Low profile** devices multitapped to accommodate existing system configurations. Eliminates non-standard part approval when specified by the Military Drawing number. Listed on QPL-21038-31 from BETA transformers.	VII-243
HLP-6000 Series**	Interface Transfomers (Beta* *)	0.63 x 0.63 x 0.175" Flat Pack or Surface Mount Modules	Provides the turns ratio configurations, component isolation, and common mode rejection ratio characteristics necessary for MIL-STD-1553A and B. These transformers are **hermetically sealed, low profile**, modular units that are multitapped to accommodate existing system configurations.	VII-243
LPB-5000 Series**	Interface Transfomers (Beta* *)	0.63 x 0.63 x 0.150" Flat Pack or Surface Mount Modules	Provides the turns ratio configurations, component isolation, and common mode rejection ratio characteristics necessary for MIL-STD-1553A and B. These transformers are **low profile**, modular units that are multitapped to accommodate existing system configurations.	VII-243
TST-9000 Series**	Twin-Stacked Interface Transfomers (Beta* *)	0.63 x 0.63 x 0.280" Flat Pack or Surface Mount	Provides the turns ratio configurations, component isolation, and common-mode rejection ratio characteristics necessary for dual redundant MIL-STD-1553A and B data bus configurations. These transformers are dual modular units that are multitapped to accommodate existing system configurations. The twin-stacked design replaces two single transformers with one dual transformer, conserving PC board space.	VII-243

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

Data Bus Products continued.

* * Order directly from Beta Transformer Technology Corporation, 40 Orville Drive, Bohemia, NY 11716-2529
  Tel: (516) 244-7393, Fax: (516) 244-8893, Website: www.bttc-beta.com.

●*SOFTWARE*				
MODEL	**TYPE**	**PACKAGE**	**DESCRIPTION/FEATURES**	**PAGE**
BUS-69023 (used w BUS-65517II IDEA card) **BUS-69123 (used w BUS-65518 IDEA card)**	MIL-STD-1553 RT ProductionTest Plan (PTP) Software	Floppy Disk(s)	The Bus Interface Card and this software package is a turnkey system used to automate the protocol tests of the SAE RT Production Test Plan. A number of military contracts are imposing this test plan on MIL-STD-1553 suppliers. The test engineer characterizes the Unit Under Test (UUT) via user- friendly, menu-driven software. Once this is done, the configuration can be saved to disk. The Bus Interface Card will execute the ENTIRE protocol tests in under 20 seconds. A report detailing test results is automatically generated.	VII-202
BUS-69024 (used w BUS-65517II IDEA card) **BUS-69124 (used w BUS-65518 IDEA card)**	MIL-STD-1553 RT ProductionTest Plan (PTP) Software & Validation Test Plan (VTP) Software	Floppy Disk(s)	The Bus Interface Card and this software package supports, real time C and Pascal library software, Menu software, VTP and PTP tester software. The Validation Tester complies with SECTION 100 of the MIL-HNDBK-1553A RT Validation Test Plan. It is a turnkey system used to automate the electrical, protocol, and noise rejection tests. The VTP allows the user to specify the UUT configuration via the user-friendly menu driven software. The VTP executes and monitors complex communication sequences like bus switching, superseding and RT-RT mode commands. The VTP reduces the time to validate an RT from an estimated one week to a few hours. This includes BUS-69023/69123 software package described above.	VII-202
BUS-69035 (used w BUS-65517II IDEA card) **BUS-69135 (used w BUS-65518 IDEA card)**	DOS & Wndows Menu, 'C' Library, Pascal Library (w.o. Menus) MIL-STD-1553 RT ProductionTest Plan (PTP) Software & Validation Test Plan (VTP) Software & Parameter Monitor (PMON) Software & Reconstructor Menu Software	Floppy Disk(s)	Used with a bus interface card, this software package allows the user to program the card with either Microsoft C, Borland C++, or High C (w/ the Phar Lap Linker), Windows 95, NT for the MIL-STD-1553 environment. The Parameter Monitor (PMON) software with a bus interface card provides a powerful tool for analysis of MIL-STD-1553B data. The Parameter Monitor focuses on the system data domain. Selected parameters are stored on disk and displayed in several available formats. The stored data can be replayed using replay options as single-step, go to time, go to a marked event, or message level monitoring. This instrument is ideal for application engineers and system integrators doing integrated analysis and simulation. This is an indispensible tool for developing and displaying MIL-STD-1553B applications ranging in complexity from a single box to a complete communication system. The reconstructor software allows the user to "playback" monitored 1553 message traffic. In the playback mode, it can be programmed to emulate the Bus Controller and / or up to 31 Remote Terminals transferring actual data on a 1553 data bus. The ability to "learn" and "playback" 1553 message traffic from an existing system eliminates the need to develop complex software for system integration.	VII-202

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

MODEL	TYPE	PACKAGE	DESCRIPTION/FEATURES	PAGE
●SOFTWARE (continued)				
BUS-69080	"ACE" Software Library	Floppy Disk(s)	Consists of C run-time library which perform common MIL-STD-1553 functions. Supports all ACE series products, i.e., BU-61580, BU-65550/51 PCMCIA, BU-65539 IBM AT and BU-65528 VME/VXI products.	VII-227
BU-69081	Win 3.1 Menu		Menu Software consists of user-friendly pull-down screens that assist immediate 1553 communications with the ACE board products (BU-65550/51 PCMCIA, BU-65539 IBM-AT, and BU-65549 PCI card).	Contact Factory
BUS-69082	Windows 95 DLL/Drivers			VII-227
BUS-69083	Windows NT DLL/Drivers			VII-227
BUS-69084	Windows 95 Menu			Contact Factory
BUS-69085	Windows NT Menu			Contact Factory
BUS-69087	VxWorks Run-Time Library (RTL)			VII-227

★ New Product

Note: Most hybrid products are available with Military processing — contact factory.

Note: All Trademarks used herein are the property of their respective owners.

VIII.
PRODUCT BRIEFS

1553 TERMINALS FOR SPACE APPLICATIONS

BUS-65142, BUS-61553-605/606 and SP'ACE

INTRODUCTION

ILC Data Device Corporation (DDC) has products available for supporting MIL-STD-1553 data bus for Space and other High Reliability (Hi-Rel) applications. Two fully integrated terminals that are now being used in a variety of Hi-Rel applications are the BU-65142 Remote Terminal for simple non-CPU based applications and the BU-61582 Bus Controller, Remote Terminal and Bus Monitor (BC/RT/MT) Space Advanced Communications Engine (SP'ACE). The SP'ACE follows in the footsteps of its predecessor, the BUS-61553-606 Advanced Integrated Mux Hybrid (AIM-HY). The BUS-61553-606 was the first to be space qualified (SSQ) for the International Space Station Program (Drawing SSQ-22678).

The SP'ACE terminal is half the size of the former AIM-HY series and has many more enhancements and features to simplify the interface and off-load the host processor. The BU-61582 terminal is a complete integrated interface between a 1553 bus and a host processor. The BU-61582 is housed in a 1.9 square inch package that shares the same foot print, functionality, and software compatibility of our standard BU-61580 series terminal. Its additional screening and RAM (16K x 16) provide the SP'ACE with space level screening options.

DDC is committed to producing hybrids with enhanced processing and screening for space borne applications and other systems requiring the highest levels of reliability and radiation tolerance. These platforms include launch vehicles, satellites and the International Space Station.

DESIGN

The manufacturing of hybrids for space applications involves enhancements in the area of layout design, processing and screening. The principle goal of the hybrid mechanical design is to improve the ease of wire bonding and inspection. All of DDC's space grade terminals use ceramic packaging. Ceramic packages eliminate the potential hermeticity problems associated with the glass beads of KOVAR (metal) packages. In addition, ceramic packages facilitate improved thermal properties and have reduced weight and lower cost.

HYBRID SCREENING FOR HIGH RELIABILITY

Visual screening for space grade hybrids includes rigorous inspection for damaged die, broken or crossed wires, particles, and contamination. Additional screen-

ing procedures for the space hybrids include:
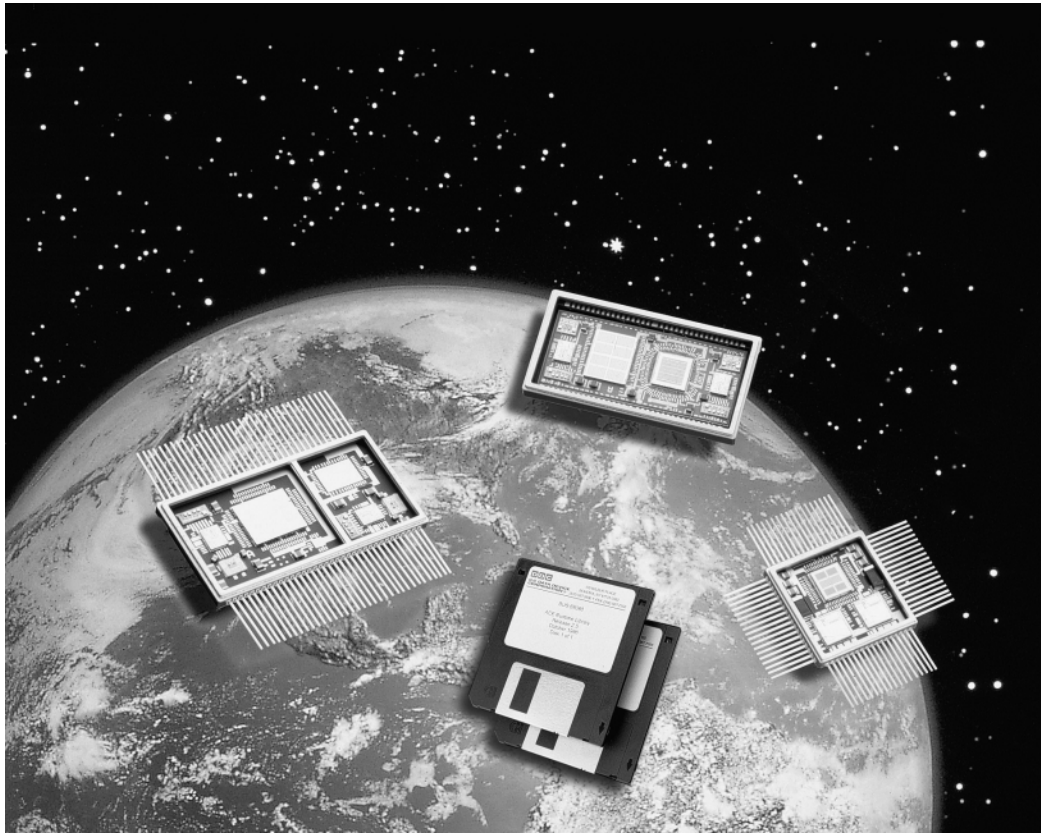
- Particle Impact Noise Detection (PIND) testing with the use of getter material

- 320 hour burn-in (the burn-in for Class H devices is 160 hours)

- 100% nondestructive wire bond pull

- Optional radiographic (X-ray) analysis

- Destructive Physical Analysis (DPA) testing.

The Percent Defective Allowable (PDA) after burn-in is:

- 2% for the space hybrids.

- 10% for standard screened hybrids.

RADIATION HARDNESS

In addition to the screening process, DDC pays particular attention to the technologies used in the design of space grade 1553 products. DDC's latest SP'ACE BU-61582 series products combine bipolar transceivers with the Honeywell Solid-State Electronics Center's (HSSEC) 0.8 micron (Radiation Insensitive CMOS) RIC-MOS rad hard process on the protocol (JRAD) chip and

RAM. The earlier AIM-HY series terminals combined Bipolar with CMOS-SOS (Silicon-on-Sapphire CMOS) logic and RICMOS HSSEC RAM (HC6364) to provide high radiation tolerance. The present SP'ACE with transceivers, protocol and RAM has 1 mega-rads(Si) total dose immunity.

The SP'ACE hybrids also exhibit a LET threshold of 100 Mev/mg/cm^2, providing a soft error rate of 10^{-7} errors/bit-day at 80°C and Adams 10% worst case environment. Since the transceivers are bipolar, and the digital logic and RAM is implemented in RICMOS technology, the hybrids are inherently immune to latchup.

THE TRUE HI-REL SOLUTION

Per MIL-HDBK-217F the lower power consumption and reduced number of nonhermetic PC board interconnects combine to give the BUS-61553-606 an MTBF of 4,100,000 hours in a space flight environment at 85°C. The corresponding MTBF for a 'discrete' 1553 implementation is about 2,270,000 hours. The SP'ACE which represents the latest in integration, offers more features and options with an MTBF of 3,890,000 hours in the comparison.

MIL-STD-1553 TRANSFORMERS

INTRODUCTION

Beta Transformer Technology Corporation (BTTC), a subsidiary of ILC Data Device Corporation (DDC), is a leader in isolation and coupling transformers for MIL-STD-1553 data bus. Beta is located within DDC's three-building complex in Bohemia, NY.

The close relationship between Beta and DDC enables the companies' engineers to work hand-in-hand in developing new products. Using advanced technology and manufacturing processes, Beta has become the recognized source for custom interface transformers that continually meet or exceed the requirements for military systems. Beta transformers provide the turns ratio configurations, component isolation, common mode rejection, and other electrical and mechanical characteristics necessary for MIL-STD-1553 A & B compliance.

Individual data sheets on all transformers are available through Beta's applications department. The step-up and step-down turns ratios that are available with our transformers complement DDC's entire 1553 product line and are compatible with competitors' drivers, receivers and transceivers.

Beta transformers are low-profile (down to 0.130" height), modular units that are multitapped to accommodate existing system configurations. Encapsulated and hermetically sealed interface transformers accommodate printed circuit board mounting. Special lead bending for surface mounted units, along with custom part marking, is also available. The waveform fidelity of Beta transformers make them an excellent choice for MIL-STD-1553 sinusoidal or trapezoidal applications.

DESC QUALIFIED

Beta is recognized as a qualified source of interface transformers specified by the Defense Electronics Supply Center (DESC) MIL-T-21038/27-01 through -20. These transformers are designed and manufactured in accordance with MIL-T-21038.

For space applications, transformers are manufactured and tested in accordance with MIL-STD-981 which details requirements for space-level transformers. Beta is a participating member of the MIL-STD-981 Industry Committee which is creating a comprehensive specification for space applications. Many programs, however, have unique requirements. Beta, like DDC, maintains the flexibility to satisfy these requirements on a program-by-program basis. Lead times for products are:

MIL-Level: 1-4 Months

Space-Level: 6-14 Months

Pricing: Contact factory directly

SSQ DRAWING

Beta passed an IBM Quality Survey for MIL-STD-981 "S" level manufacturing. Subsequently, Beta has delivered to IBM, along with other major suppliers for Space Station Freedom, both epoxy-cased and hermitically sealed pulse transformers. Beta, along with Grumman Space Station Integration in Reston VIRGINIA, developed the SSQ 22676 specification. This specification, entitled "Transformer MIL-STD-1553 Terminal Interface, Space Quality," standardizes the transformers used on Space Station Freedom. The products and part numbers listed in this specification are Beta's standard products. At this writing, Beta is listed as the only approved supplier to SSQ 22676.
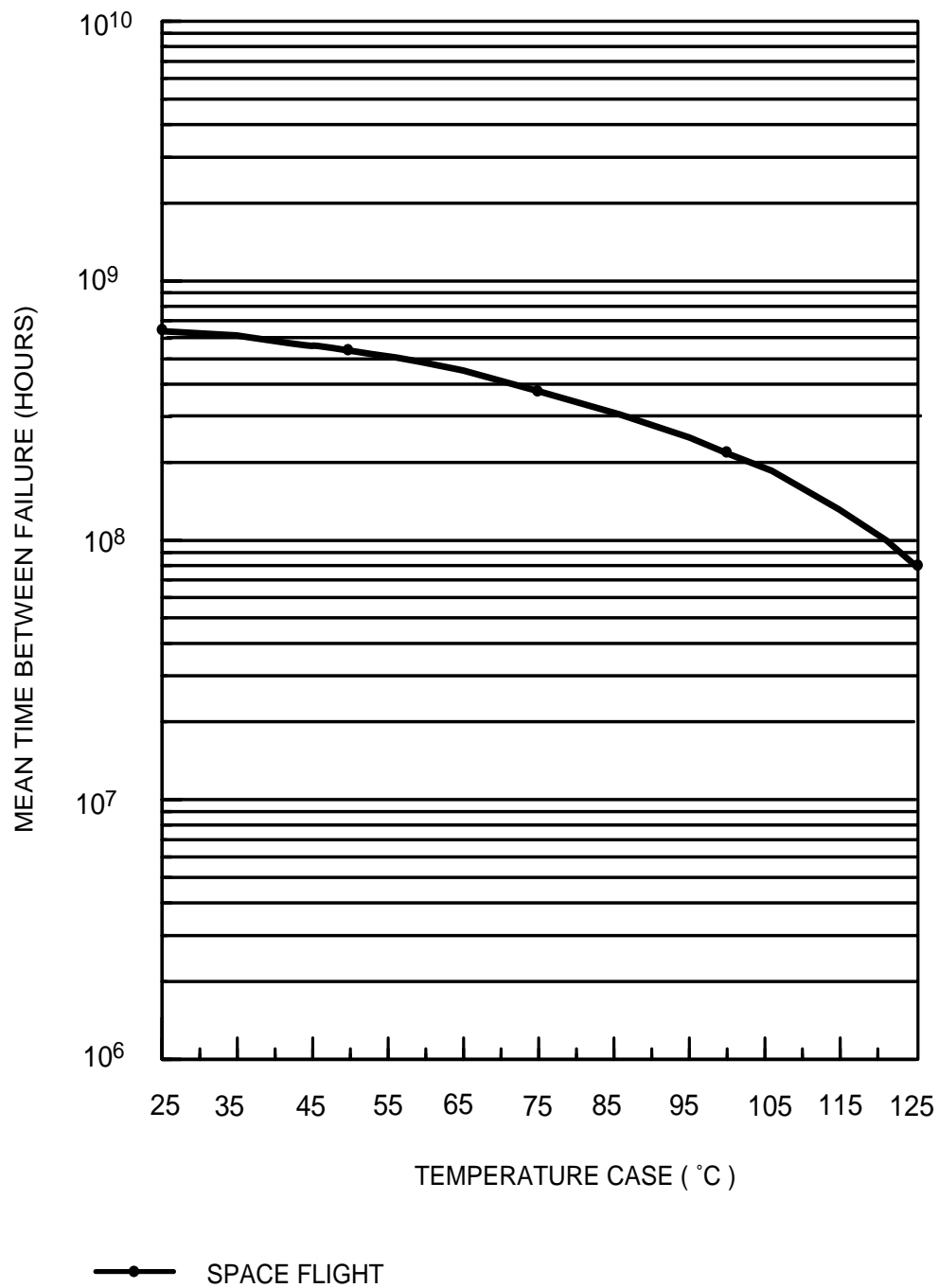
FIGURE 1. RELIABILITY PREDICTION FOR 1553 INTERFACE TRANSFORMERS

Beta's Statistical Process Control (SPC) program has been recognized and approved by General Dynamics and Loral.

ESD RATINGS

Electro Static Discharge (ESD) testing was performed on Beta's MIL-T-21038/27 QPL interface transformers according to MIL-STD-883 method 3015.7. The transformers were determined to have an ESD failure threshold classification of class 3 (in accordance with table III.) and a threshold of greater than 5 kV per the human body model. The IMCS 5000 automated ESD test system was used to perform this test.

An integral part of Beta's manufacturing process is its test department where both electrical parameters and environmental testing is performed. Custom-designed test fixtures facilitate fast, repetitive, accurate testing of all 1553 interface transformers. Beta's test department is also equipped to carry out burn-in and thermal shock testing. All other environmental tests are performed at a local testing laboratory.

Using the calculations from par. 5.1.3.5 of MIL-HDBK-217, the following mean time between failure, (MTBF) table, has been generated to cover all of Beta's 1553 interface transformers (see FIGURE 1).

MATERIALS

The materials used on all of Beta's 1553 interface transformers can be broken into two main categories: (1) epoxy-cased units and (2) hermetically sealed units.

TABLE 1. lists a breakdown of materials used on these two product groups.

TABLE 1. PRODUCT MATERIAL BREAKDOWN	
EPOXY CASED UNITS	**HERMETICALLY SEALED UNITS**
1) Diallyl Phthalate Case 2) Tin Coated Steel leads 3) Solderable Magnet Wire 4) Ferrite Core 5) Insulcast Potting 6) Core Isolation Coating 7) SN 60 Solder 8) Mylar Tape	1) Gold Plated Kovar Case 2) Gold Plated Kovar Lead 3) Solderable Magnet Wire 4) Ferrite Core 5) Insulcast Potting 6) Core Isolation Coating 7) SN 95 Solder 8) Mylar Tape

EQUIVALENT CIRCUIT

The nominal equivalent circuit shown in FIGURE 2 is for transformer BUS-25679, referred to terminals (1-2) or (2-3) for a 1 MHz sine wave input.

Parameters Rc and Lm, being functions of the ferrite core material, will vary over voltage level, frequency and temperature. Rp and Rs are copper wire resistances and will change over temperature as per the coefficient of standard annealed copper. L1 and Cc are generally functions of unit geometry and remain mainly unchanged over changing conditions compared to other parameters. This equivalent circuit is useful for computer modeling in conjunction with other circuit elements.
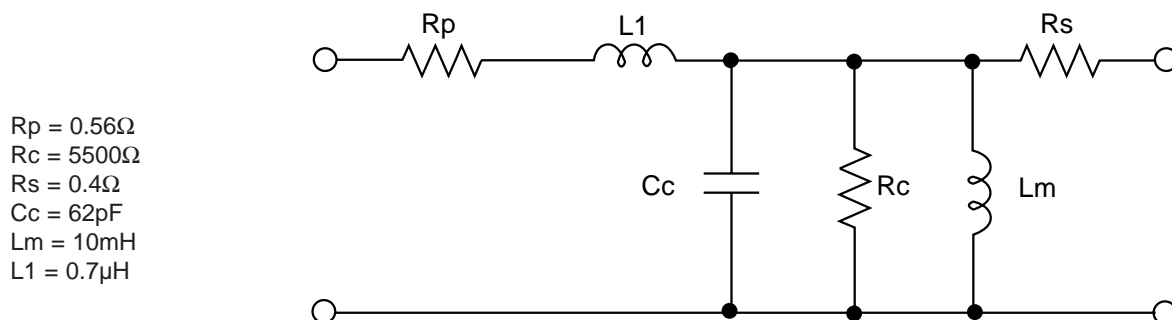

Rp = 0.56Ω
Rc = 5500Ω
Rs = 0.4Ω
Cc = 62pF
Lm = 10mH
L1 = 0.7µH

FIGURE 2. TYPICAL EQUIVALENT CIRCUIT

INPUT IMPEDANCE VERSUS FREQUENCY

The Impedance vs. Frequency plot shown in FIGURE 3 is also for transformer 25679, referred at terminals (1-3) and (5-7), from 75 kHz to 1 MHz, at 1V and a temperature of 25°C. MIL-STD-1553B requires a minimum value of 3 kΩ at terminals (1-3) for this impedance over the frequency range of 75 kHz to 1 MHz when measured with a 1 Vrms sine wave.

CONCLUSION

Beta transformer products provide the quality, reliability and precision needed to satisfy industry requirements. Beta's production capabilities ensure timely delivery of customer orders. This, coupled with broad design capabilities, makes Beta the ideal source for 1553 interface magnetics for military, space, and industrial applications.

FIGURE 3. IMPEDANCE VS. FREQUENCY PLOT

THIS PAGE INTENTIONALLY LEFT BLANK

IX.
APPLICATIONS INFORMATION

APPLICATION NOTES
CONTENTS

SYSTEMS ADVANTAGES OF THE INTEGRATED 1553 TERMINAL

ABSTRACT

There are a great many factors to consider in MIL-STD-1553 terminal design. In today's increasingly competitive climate for military systems, cost assumes a high degree of importance. Component prices, while significant, represent only a portion of the overall system cost picture. Some of the other prominent factors include functionality, ease of use, and the development costs associated with circuit design, printed circuit board layout design, and software development. High on the list of system design considerations are power consumption, board space, and thermal aspects.

Additional factors contributing to a system's life cycle costs (maintenance and logistics) include reliability, testability, ease of troubleshooting, mean time to repair, and component count.

COMMON 1553 ARCHITECTURE

The most common MIL-STD-1553 terminal architecture, a dual redundant bus interface, RT or BC/RT/Monitor protocol, with memory management, shared RAM processor interface, and buffer memory, is the subject of this application note. For most applications, a shared RAM type of interface is preferred over a direct memory access (DMA) configuration. The shared RAM configuration eliminates contention problems; moreover, it allows the 1553 terminal to access its buffer RAM while the host CPU is simultaneously able to access its ROM, RAM, or I/O.

Two alternative solutions to this interface design are considered:

(1) a fully integrated multi-chip module (MCM) implementation, and

(2) a multi-component discrete design.

As a preface to presenting the comparisons and trade-offs involving the two approaches, an attempt will be made to dispel some of the myths that prevail. A comparison summary of a fully integrated 1553 multi-chip module (MCM), DDC's BU-61580D1-110, vs. a "discrete" terminal implementation is detailed in Table 1. The balance of this application note will support the conclusions listed in the summary.

CHIPS VS. MULTI-CHIP MODULES

With respect to the design of 1553 interfaces, there exists considerable confusion regarding the trade-offs when comparing the use of a single multi-chip module (MCM) to that of multiple discrete packaged chips. The two fallacies that need to be dispelled are described as follows:

(1) **The Fully Monolithic 1553 Solution.** There is no such thing. As of this writing, there are no fully monolithic implementations of a 1553 terminal. To date, nobody has succeeded in integrating analog transceiver functions together with the various digital functions (protocol, RAM, etc.) on the same piece of silicon. Today, all 1553 transceivers are fabricated using bipolar chips, while all current 1553 digital implementations are CMOS chips or chip sets, to minimize power consumption.

Most 1553 transceivers are still fabricated as hybrid circuits utilizing thick film resistors and

chip capacitors external to active custom bipolar monolithic chips, or as discrete designs. The "Monolithic 1553 Solution" is, indeed, a specious claim.

(2) **Monolithics Are Less Expensive Than Hybrids.** In a first glance price comparison between a 1553 protocol chip and a fully integrated 1553 hybrid, the protocol chip may appear to be the more economical choice. **In point of fact, this is not the case.** In order to comprise a complete microprocessor/shared RAM to dual 1553 bus interface, today's single hybrid component is nearly autonomous. All that is needed externally are a clock oscillator and two isolation transformers. By contrast, in order to assemble a complete 1553 solution, the protocol chip requires (in addition to the oscillator and transformers) a separate 1553 transceiver, two packaged memory chips, two address buffers, and two data buffers.

In addition, most monolithic 1553 protocol chips provide only a direct memory access (DMA) host interface. Additional logic is required to derive the control signals required for a shared RAM interface: this logic is usually implemented using a programmable logic device (PLD). In addition to the raw cost of these additional components, by necessity they entail increased PC board space and complexity, and other added costs. This application note will proceed to show that the total cost of using these components is greater than that for the single fully integrated 1553 terminal.

NON-RECURRING COSTS

This application note takes a look at non-recurring engineering costs broken down into three areas: proposal costs, circuit design, and printed circuit board layout design.

Proposal Effort.

The use of a single component provides obvious advantages in terms of proposal preparation effort. Since many system proposals are done on a tight schedule, it is of urgent importance to minimize not only risk but also the number of components requiring selection, evaluation, and documentation. Typically, this entails analyses of functionality, cost, board space, power requirements, reliability, and other system parameters. If this evaluation is done on a comparative basis, it will inevitably favor the use of the single integrated terminal over the discrete design.

Electrical Design.

In contrast to an autonomous terminal, a design consisting of multiple components requires additional analysis and evaluation to verify operation and ensure an optimal design. This generally entails the time, cost, and risk of component selection, plus breadboard design, programmable device programming, build, and evaluation. Assuming all goes well in the breadboard stage, the amount of documentation required for a multicomponent bus interface design will necessarily be greater than that for a single MCM, which can be purchased to a standard military (DESC) drawing. The selection of the integrated terminal implementation eliminates this portion of the system breadboarding, allowing the design team to concentrate on higher level tasks in the system development.

Layout/Board Design.

There is little question that the time, cost, and risk of printed circuit board layout

PARAMETER	BU-625880-110 ACE MCM	"DISCRETE" 1553
Component Cost	Comparable	
Number of components	1	9
NRE costs	Moderate	Higher
Purchasing, inspection, & handling	Low	Higher
PC board cost	Low	Higher
Board assembly, inspection and rework	Low	Higher
Power dissipation (W)	2,250 (max)	4,865 (max)
Board space (plug-in package)	2.20 sq. in.	7.09 sq. in.
Number of PC board holes	70	276
Junction-to-case thermal resistance (°C/W)	5.54	6.00
Layout complexity	Low	High
MTBF @ 28, C (hours) @ 125°C (hours)	18,128,820 240,385	3,991,994 42,441
Troubleshooting effort required	Minimal	Complex
MTTR	Low	Higher
Life cycle costs	Moderate	Higher

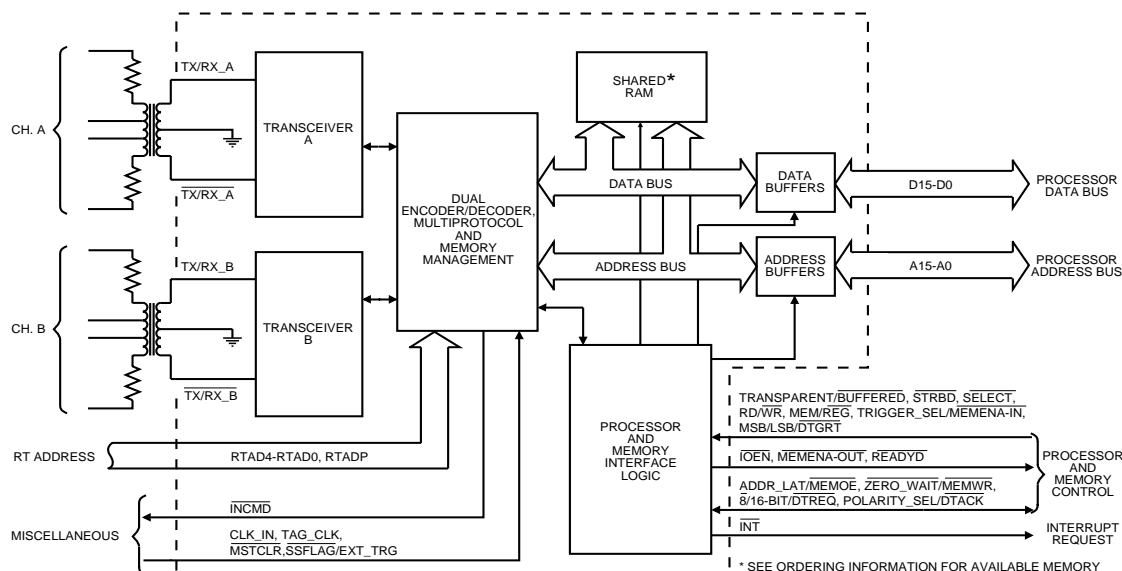TABLE 1. COMPARISON SUMMARY: BU-61580 ACE TERMINAL VS. "DISCRETE" 1553 TERMINAL IMPLEMENTATION

FIGURE 1. SIMPLIFIED BLOCK DIAGRAM

THE BU-65170 AND BU61580 ACE SERIES HYBRIDS

Figure 1 illustrates a simplified block diagram of the BU-65170 and BU-61580 Advanced Communications Engine (ACE) 1553 terminals. The ACE terminals comprise a family of complete integrated interfaces between a host processor and a MIL-STD-1553 bus. The ACE series of multi-chip hybrids (MCMs) provide RT-only or BC/RT/MT functions in the identical 1.9 square inch ceramic package footprint. The ACEs integrate dual transceiver, protocol, memory management and processor interface logic, and 4K words of internal buffer RAM. The BU-61585 version of the BC/RT/MT ACE includes an additional 8K X 17 of internal buffered RAM with RAM parity generation and checking. The ACE terminals are available in the choice of 70-pin quad in-line, flatpack, or J-lead packages.

Transceiver power supply options include +5V (only), +5/-15V, and +5/-12V.

As a means of minimizing PC board space and "glue" logic, the ACE terminals provide a high degree of flexibility in interfacing to a host processor and internal/external buffer RAM. The interface options include 8-bit or 16-bit buffered modes to 4K (or 12K) of internal RAM, 16-bit transparent and DMA configurations, and the capability to easily interface to processors that do not have an "acknowledge" input signal to enable wait states. The transparent and DMA configurations allow the use of up to 64K X 16 of external RAM.

The ACE components provide complete multiprotocol support of MIL-STD-1553A/B Notice 2 and STANAG 3838. A "Universal" version of the ACE, the BU-61590D5, which incorporates a dual sinusoidal transceiver, also provides compliance to McAir standard A3818. A5232, and A5690.

The advanced functional architecture of the ACE terminals provides software compatibility to DDC's previous generation AIM-HY and AIM-HY'er series hybrids. In addition, the ACE terminals incorporate myriad architectural enhancements supporting flexible operation while off-loading the host processor.

For BC mode, the ACE includes capabilities for programmable intermessage gaps, as well as automatic retries and frame repetition.

For RT mode, the ACE supports programmable illegalization and busy, and the choice of single message, double buffering and circular buffering modes, programmable by T/$\overline{R}$ bit, broadcast, and subaddress. For monitor. the ACE implements a true message monitor or combined RT/monitor mode with message selection based on RT address, T/$\overline{R}$ bit, and subaddress.

A wide array of maskable interrupts are provided for each of the three modes. The ACE hybrids may be operated with either a 16 MHz or 12 MHz clock input. The BU-65171 and BU-61581 versions allow for the use of a software programmable RT address.

The ACE terminals consume very low power, particularly the +5V only version (BU-65170/61580X3). The ACE MCMs have a standard military (DESC) drawing pending approval. They are manufactured in DDC's DOD-STD-1772 facility. 1772 processing is required for MCM or hybrid screening to 883B. An intermediate level ensuring operation over the full military temperature range of -55°C to +125°C. but without QCI testing, as well as units with a limited temperature range of 0 °C to 70°C, are also available.

design for a multiple component 1553 interface is much greater than that incurred when designing in a single MCM. Several factors favor the use of the integrated terminal:

(1) **PC board real estate.** This is often at a premium in avionics systems. The combination of a pin grid array 1553 protocol chip plus external transceiver, PLD, RAM, and buffer chips will require more board space than a single integrated MCM. This can be a restricting and complicating factor in PC board layout design,

(2) **Number of layers.** The use of a multiple component 1553 interface will, in all likelihood, lead to the added complication of more board layers. The probability of having to add PC board layers will be even greater if the protocol chip is packaged in a pin grid array (PGA) package.

(3) **Routing.** The use of the multicomponent 1553 interface will require approximately 200 more interconnects than will the integrated MCM. This is primarily attributable to the use of discrete static RAM and buffer chips, entailing the additional routing of 16-bit wide address and data buses. plus control signals. If the protocol chip is packaged in a PGA package, the routing task is further complicated by the difficulty of passing signals through the board region populated by the pin grid (four sides, multiple rows).

(4) **Component positioning.** This is an aspect that should not be trivialized. Some of the issues requiring careful attention are component spacing, trace length, crosstalk, ground planes, power distribution, and decoupling. The most risky interconnects, in terms of layout, are the protocol chip- to-transceiver and transceiver-to-transformer interfaces. Care must be taken in these areas to ensure proper transceiver and encoder/decoder operation. If the layout of the analog traces has excessive capacitance, the effective input impedance of the receiver as seen from the bus could fall below 1553's specifications. This can create problems in 1553 validation testing. Other possible effects of poor encoder/decoder-to-transceiver layout include crosstalk and/or timing skews. In some instances, these can lead to failure of the 1553 validation tests for noise and zero crossing distortion tolerance. The use of the integrated hybrid eliminates the protocol-to-transceiver interconnect on the board. Internal to an integrated 1553 hybrid, the transceiver-to-protocol connection distance is kept to an absolute minimum; the layout for this section is designed and optimized very carefully, tested and proven many times over.

RECURRING COSTS

Depending on production volume, recurring costs may assume a greater level of importance than NRE costs. For the 1553 interface, the two areas of recurring cost are component cost and miscellaneous system production costs.

The primary recurring cost is component cost. A cost analysis between the two approaches would need to compare the pricing of a single 1553 MCM terminal, such as a BU-65170 or BU-61580, against the total cost of a multicomponent design. The latter consists of a protocol chip plus dual transceiver, two discrete static RAM chips, two each tri-state data and address buffers, and a PLD chip for "glue" control logic.

The pricing comparison will vary somewhat as a function of quantity, screening levels, RAM, and logic chip prices, and other factors. Nevertheless, the price of the single MCM will generally be about 10 to 15% lower than the sum of the discrete component prices.

While the component cost advantage for the integrated 1553 MCM terminal solution over the multicomponent approach is relatively modest, the comparison is more apparent in other recurring aspects of system production. The reduced number of components results in cost advantages throughout the production cycle. These include:

1. Purchasing
2. Documentation
3. Material Handling
4. Incoming inspection
5. Component Test
6. Printed Circuit Board Cost
7. Board Assembly
8. Board Inspection
9 Testing and troubleshooting
10. Yield and re-work costs

ADVANTAGES OF THE ACE SERIES INTEGRATED 1553 SOLUTION

As mentioned, the use of the integrated hybrid results in advantages throughout the development and production cycles. In addition to the advantages listed above, it is important to consider that a 1553 terminal such as the BU-61580 is procured as a **fully integrated, tested, and inspected single component.** All other things assumed equal (they're not), this reduces the associated cost and overhead bur-

dens for the user.

Power and Thermal Considerations

Several important aspects of system design relate to power requirements. These include the number of supply voltages, current requirements (both instantaneous and average), dissipation, size, weight, complexity, heatsinking, and thermal budgeting. All weigh significantly in system MTBF and life cycle costs.

DDC, in the evolution of its 1553 product line, has been through several generations of transceiver and protocol design.

For the 15V and 12V ACE terminals (BU-65170/61580X1[2]), the active components for the transceivers for each channel are consolidated into a single monolithic chip. The need for a +15V or +12V supply, as required in earlier generation DDC products and some competitive transceivers, is eliminated. The latest generation of transceiver monolithics, which is incorporated into the 15V (12V) ACE terminals, not only eliminate the need for a positive supply rail, but greatly reduce the power consumption and power dissipation required for the 1553 interface. Going one step further, the fully monolithic +5V only dual transceiver integrated into the BU-65170/61580X3 ACE terminals not only eliminates the -15V (-12V) supply as well as the +15V (+12V) supply, but requires **an even lower level of input power** than the 15V or 12V ACE units. Specifically, the idle (non-transmitting) dissipation for the 5V ACE is about 60% that of the 15V or 12V ACE.

Going beyond low power consumption/dissipation, the ACE series hybrids provide a further advantage in the area of thermal management. This results from the larger surface area of the ACE ceramic MCM package. The larger package surface area, 1.90 sq. inches vs. 1.13 sq. inches for a 36-pin DIP transceiver, results in a lower junction-to-case thermal resistance, and thus improved reliability for the ACE over a multi-component implementation. The junction-to-case thermal resistance of the 15V and 12V ACE series terminals (I.0 X I.9 inch ceramic package) is 5.54°C/W. For a 36-pin DIP transceiver, the thermal resistance is 6.00°C/W.

The power dissipation comparison of the ACE series MCM to a discrete component 1553 interface is indicated in Table 2. Maximum power dissipation values are indicated for 25% transmitter duty

cycle. At this duty cycle, it should be noted that the total power consumed by the 1553 interface (for either implementation) is about 0.3W higher than the power dissipated. The 0.3W is dissipated externally, in bus isolation and termination resistors.

As shown, the ACE integrated solution requires less than half the power of the discrete 1553 interface.

To summarize, the low levels of power consumption/dissipation and inherent heatsinking of the ACE series hybrids provide several advantages. Decreased current requirements serve to reduce power supply size, complexity, weight, and heatsinking. Furthermore, the task of thermal management for the 1553 interface (transceiver) simplifies mechanical design and assembly costs by reducing the need for heatsinking rails and cooling fans.

PC Board Space, Cost

The recurring cost for a PC board accommodating a multi-component discrete 1553 terminal design will inevitably be higher than for a board incorporating a single bus interface MCM. As a bare minimum, the board for the multiple components will need to have **at least 200** additional holes. This is just to accommodate components.

Moreover, additional holes will likely be required for vias (feedthroughs) to interconnect the multiple components. An even larger increase in recurring board cost will be incurred if additional layers are required to accommodate the multi-component terminal. This is likely to be the case. In terms of yield and reliability, the board for the multi-component 1553 terminal, with its greater number of layers, holes and routing density will have the greater chance of failure due to faulty solder lands, joints, or solder bridging.

Board space analysis in Table 3 assumes 0.1 inch component spacing. The analyses do not consider the board space required for transformers or clock oscillators,

TABLE 2. ACE VS. DISCRETE POWER DISSIPATION	
ACE SERIES INTEGRATED TERMINALS (max)	**"DISCRETE" 1553 TERMINAL SOLUTION (typ/max)**
BU-61580X1 (5V/-15V): 2.25 W	Protocol/Interface Chip 0.250 W
	Dual -15/+5V Transceiver: 2.75 W
	Two 8K x 16 static RAM chips: 2 x 0.050 = 0.100 W
	Two 54LS244 address bus buffers: 2 x 0.27 = 0.54 W
	Two 54LS245 data bus buffers: 2 x 0.475 = 0.95 W
	PAL I6L8L8Q-25 PLD: 0.275 W
Total ACE Terminal: 2.25 W	**Total "Discrete" Terminal: 4,865 W**

which are common to both approaches.

The ACE MCM terminal occupies less than one third of the PC board area of the "discrete" implementation.

System Production Costs

The use of a single component, rather than several components, results in a number of cost advantages in the labor intensive areas of system manufacturing. The various costs of component procurement are reduced in proportion to the number of line items on the parts list. This includes the overhead associated with purchasing, documentation, and material handling. Similarly, the costs of incoming inspection and component test labor are minimized by the use of the integrated terminal. The efforts required for PC board assembly and inspection are similarly minimized. This can result in significant cost savings. If the 1553 protocol device for the discrete 1553 terminal is packaged in a pin grid array (PGA) package, inspection of PC boards becomes difficult. In all likelihood, the added complexity of the multi-component 1553 terminal will produce lower initial yield at board level assembly, resulting in higher rework costs.

Testing and Troubleshooting.

In terms of system costs, testing is a critical labor intensive operation. This being the case, testability assumes an important role not only in production economics, but also in terms of the total system life cycle costs. These costs are both non-recurring and recurring. NRE test costs include test equipment, fixturization design/build, test software development, and documentation. Since system and subsystem test methodologies are typically developed independently at factory, maintenance depot, and weapons platform sites, these development efforts represent a significant portion of the total support cost associated with any given system. The availability of 1553 production protocol testers and general purpose analog and digital test equipment facilitates the development of end-to-end functional tests. Nevertheless, the need for diagnostic fixturization, procedures, and software can add significantly to development costs.

In terms of testing and troubleshooting, the ACE series terminals provide advantages over multi-component 1553 interfaces. It is clearly evident that the use of an integrated 1553 hybrid greatly reduces these NRE costs by greatly reducing the need for diagnostic hardware/software.

Recurring costs are substantially reduced by eliminating the potentially tedious and lengthy troubleshooting efforts required to fault isolate from among a 1553 protocol chip, a transceiver and discrete RAM, buffer, and programmable logic chips. Some receiver/decoder problems can present subtleties in isolating the bad component. In many instances, identifying the faulty component on 16-bit data/address buses involves trial and error. These time consuming troubleshooting situations are eliminated by the use of an integrated 1553 hybrid; if the 1553 interface does fail, there is only the single component to replace.

The ACE hybrids contain a number of built-in self-test features. In addition to supporting the RT "wraparound" test required by Notice 2 of MIL-STD-1553B these include internal read/write registers and RAM, which are inherently self-testable, as well as a test mode that' allows for testing of over 99% of the ACE's internal logic. The "wraparound" feature which supports both on-line testing for BC and RT modes as well as an off-line self-test, entails a validity check of all words transmitted plus a comparison check of the last transmitted word. This facilitates an "end-to-end" on-line self-test, encompassing the processor interface logic, parallel and serial due paths, internal RAM, state machine, encoder/decoder, analog transceiver, plus external transformers, cabling, and resistors.

Reliability

For reasons of both cost and system availability, reliability is a very important parameter in military systems. For 1553-based systems, it is instructive to compare the mean time to failure (MTBF) of an integrated component to that of a "discrete" terminal design.

The reliability analyses, whose results

TABLE 3. ACE VS. DISCRETE PC BOARD & SPACE		
ACE SERIES MCM TERMINALS	**"DISCRETE" 1553 TERMINAL SOLUTION**	
$((1.0 + 0.1) \bullet (1.9 + 0.1))$	Protocol/Interface PGA: $((1.1 + 0.1) \bullet (1.1 + 0.1)) =$	1.44
= 2.20 sq. in.	Dual 36-pin DIP transceiver: $((0.6 + 0.1) \bullet (1.890 + 0.1)) =$	1.393
	Two 28-pin DIP RAM chips $(2) \bullet (1.4 + 0.1) \bullet (0.595 + 0.1) =$	2.085
	Two 54LS244s + two 54LS245s $(4) \bullet (.990 + 0.1) =$	1.722
	PAL 16 LBQ $(0.987 + 0.1) \bullet (0.311 + 0.1) =$	0.447
Total ACE = 2.20 sq. in	**Total "Discrete" = 7.087 sq. in.**	

TABLE 4. ACE VS. DISCRETE RELIABILITY ANALYSIS			
BU-6158001-110 INTEGRATED MCM SOLUTION		"DISCRETE" TERMINAL SOLUTION	
(Component @ 25° C)	Failures/10^6 hours	(Component @ 25° C)	Failures/10^6 hours
BU-61580D1-110	0.0451	Protocol Chip	0.08441
4 layer PC board	0.0100	Dual 15V Transceiver	0.04362
		8K x 8 Static RAM	0.01571
		8K x 8 Static RAM	0.01571
		54LS244	0.01101
		54LS244	0.01101
		54LS245	0.02203
		54LS245	0.22030
		PALC16L8Q-25	0.00802
		6 layer PC board	0.02406
Total = 0.0551 (MTBF = 18, 148, 820 hrs.)		TOTAL = 0.25760 (MTBF = 3,991,994 hrs.)	
(Component @ 125° C)	Failures/10^6 hours	(Component @ 125° C)	Failures/10^6 hours
BU-61580D1-110	4.1500	Protocol Chip	1.31104
4 layer PC board	0.0100	Dual 15V Transceiver	9.85045
		8K x 8 Static RAM	1.54528
		8K x 8 Static RAM	1.54528
		54LS244	1.54244
		54LS244	1.54244
		54LS245	3.08488
		54LS245	3.08488
		PALC16L8Q-25	0.03160
		6 layer PC board	0.02406
Total = 4.16 (MTBF = 240,385 hrs.)		TOTAL = 23.56235 (MTBF = 42,441 hrs.)	

are listed in Table 4, compare the MTBF of a BU-61580-110 AC terminal to that of a comparable "discrete" 1553 implementation. For each implementation, the failure rate of each component, including the respective PC board area, is factored in. The BU 61580-110 is screened to MIL-STD-883C Class B in DDC's QML-38510 (MIL-STD-1772 qualified) facility. The BU-61580 is assumed to be assembled on a four-layer printed circuit board. The "discrete" terminal is assumed to consist of a PGA protocol device that is screened fully compliant to 883, including qualification; a dual −15/+5V transceiver processed to MIL-STD-883, two 8K X 16 static RAM chips, two 54LS244 buffers, two 54LS245 buffers, and a PAL16L8Q-25 programmable logic device. It is assumed that the "discrete" terminal is assembled on a six-layer board. This is due to the additional layers required to accommodate the larger number of interconnects for the "discrete" terminal design. The analyses are performed in accordance with MIL-HDBK-217F. For both eases, 50% transmitter duty cycle is assumed at 25 and 125°C. in a ground benign environment.

Note that the BU-61580 has a significantly higher MTBF estimate; by a factor of 4.5 to 1 at 25°C and by a factor of about 5.6 to 1 at 125°C. The reliability advantage is primarily attributable to the processing of the hermetically sealed ACE ceramic MCMs in DDC's Qualified Manufacturers List (QML) facility. Additional factors favoring the ACE hybrid are the lower failure rate of the associated circuit board as well as the lower overall power dissipation and lower thermal resistance of the ACE hybrid in comparison to the transceiver used in the "discrete" terminal design.

Logistics and Life Cycle Costs.

In military system proposals, life cycle costs are assuming an increasingly high level of importance. In addition to raw component cost, there are a number of logistics related elements in total system life cost. These include component weight, power consumption, testability and ease of troubleshooting, MTBF, and MTTR. The number of different types of components required to be stocked for spare parts is viewed as a very important parameter by military logistics

people. Such factors as reliability, spare parts count, commonality throughout inventory and troubleshooting cost assume greater importance than raw component cost. The "postprocurement" percentage of total system life cycle costs is typically in the range of 70 to 80%. In the case of MIL-STD-1553 terminal design, the single component hybrid solution has advantages over a multi-component implementation in all of these areas.

CONCLUSION

The choice is abundantly clear. The single component 1553 terminal solution provides a number of advantages over a multi-component design. The use of an integrated 1553 terminal such as DDC's BU-65170/61580 ACE series MCMs provides advantages in the three major phases of system life cycle: development, production, and maintenance/logistics.

Advantages of the ACE series terminals in the NRE phase include reduction of proposal effort, simplified circuit design, simplified board layout design, and a reduction of the required effort for software development.

For the system production cycle, the use of the ACE integrated terminal provides significant advantages. In addition to lower component cost, these include reduced purchasing overhead, lower PC board costs and board space, a reduction in the labor-intensive areas of board assembly, troubleshooting, and rework.

In terms of maintenance/logistics, the most significant aspect of life cycle costs, the ACE series terminals provide lower power consumption, higher MTBF, fewer parts to inventory, and reduced troubleshooting time and thus lower mean time to repair (MTTR).

PROCESSOR-TO-ACE INTERFACES

INTRODUCTION

Among the salient features of DDC's Advanced Communication Engine (ACE) terminals is the flexibility of their processor and memory interface. In today's applications, there are designs entailing the use of multiple processors and/or multiple 1553 interfaces. Some types of systems employ both multiple processors and multiple 1553 interfaces. The two most common forms of the interface are direct memory access (DMA) and shared RAM. There are variations on each of these,

In addition, other configurations involve the use of external shared RAM or dual port RAM. This application note provides a detailed discussion of all of the listed interface configurations. In addition, the nuances of the ACE's 8-bit buffered mode are explained in detail. The application note provides an analysis of the effect of contention on CPU access time in the shared RAM mode.

A tradeoff study of the various configurations, considering the issues of cost, board space, contention, and CPU bandwidth is included.

Finally, a series of diagrams are provided which illustrate the buffered shared RAM interface between nine common microprocessors and an ACE 1553 terminal.

DMA CONFIGURATION

The DMA configuration (Figures I and 2) provides the advantage of supporting the use of up to 64K words of external RAM while not requiring tristate buffers in order to isolate CPU address and data buses from the 1553 memory buses.

In the DMA configuration, the host processor controls access to the address and data buses by means of a request/grant/acknowledge handshake. When the 1553 terminal needs to transfer a word or block of words to RAM, it requests use of the buses from the host CPU. In the DMA configuration, when the ACE needs to access RAM, it requests the use of the processor data/address by asserting the signal DTREQ (Data Transfer Request) low. When the host CPU completes its current instruction cycle, it relinquishes use of the buses by asserting $\overline{\text{DTGRT}}$ (Data Transfer Grant) low. When the ACE samples $\overline{\text{DTGRT}}$ low, it asserts $\overline{\text{DTACK}}$ (Data Transfer Acknowledge) low to indicate that the ACE has accepted control of the buses.

One clock cycle later, the ACE will assert the output signal $\overline{\text{MEMENA-OUT}}$. This signal can be connected to $\overline{\text{MEMENA-IN}}$ for internal RAM access or used as a $\overline{\text{CS}}$ input to external RAM. $\overline{\text{MEMENAOUT}}$ and $\overline{\text{DTACK}}$ stay asserted for 4 clock cycles (250 ns at 16 MHz, 333 ns at 12 MHz) to complete a single-word transfer cycle. For a read cycle, the ACE output $\overline{\text{MEMOE}}$ (Memory Output Enable) is asserted low; for a write cycle, $\overline{\text{MEMWR}}$ (Memory Write) is assented low. For a multi-word transfer cycle during a Start-of-Message (SOM) or End-of-Message (EOM) sequence, $\overline{\text{DTACK}}$ and $\overline{\text{MEMENA-OUT}}$ will remain asserted for an additional [4•(number of words)] clock cycles; either
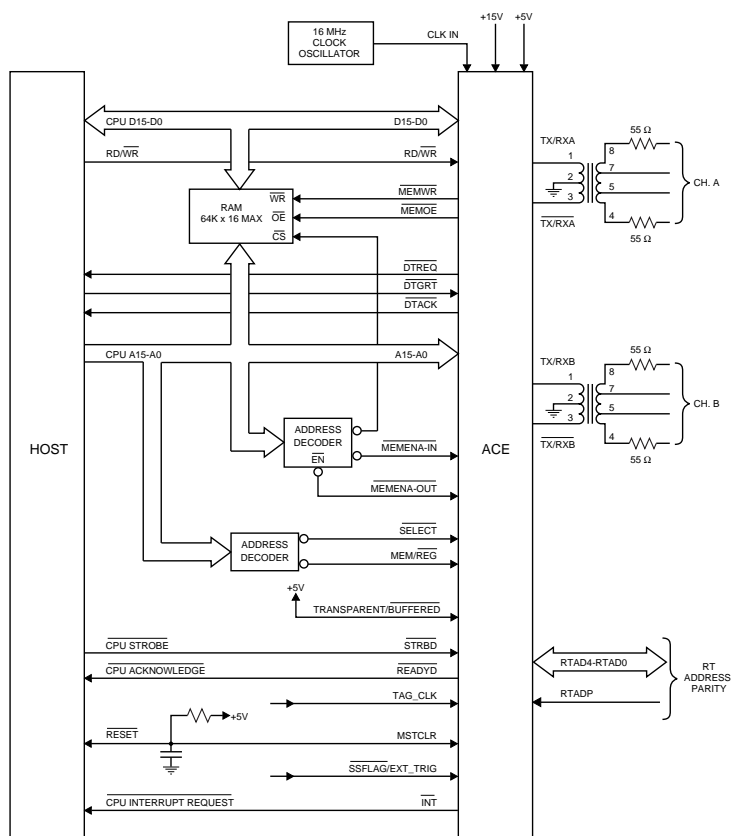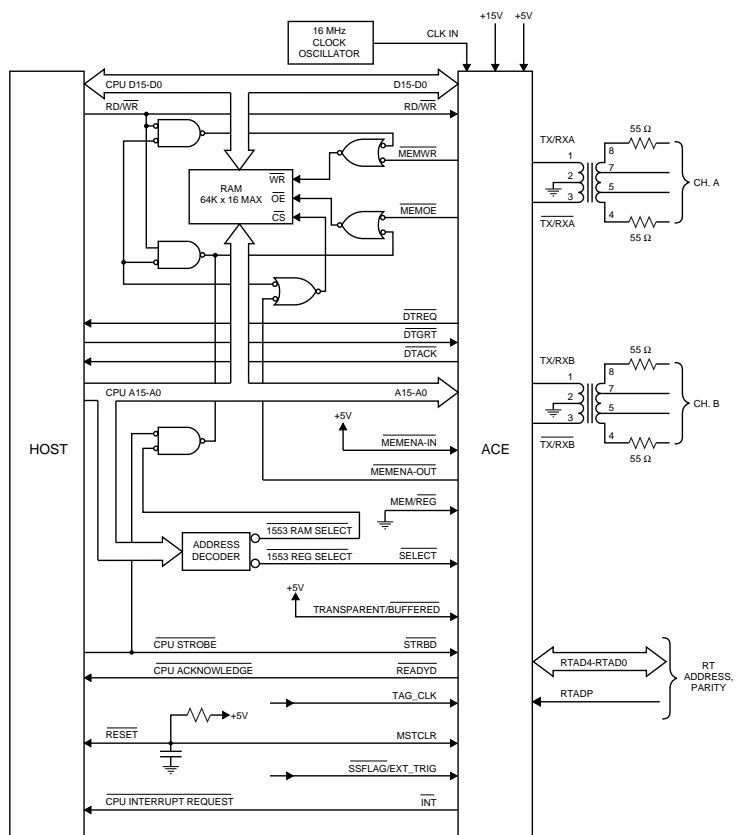
FIGURE 1. 16-BIT DMA CONFIGURATION

FIGURE 2. 16-BIT DMA MODE WITH LOGIC TO REDUCE CPU ACCESS TIME

$\overline{\text{MEMOE}}$ or $\overline{\text{MEMWR}}$ will be assented for each word cycle.

In the DMA configuration, the host has two mechanisms for accessing external RAM. As shown in Figure 1, the host can access the RAM by means of the ACE's control signals $\overline{\text{SELECT}}$, $\overline{\text{STRBD}}$, MEM/REG, RD/$\overline{\text{WR}}$, READYD, MEMENA-OUT, $\overline{\text{MEMWR}}$ and $\overline{\text{MEMOE}}$ While this requires no external control logic, it does entail about 250 ns of processor bandwidth to complete an access. This time allows the ACE to arbitrate between 1553 and CPU access, and between internal and external RAM access and to enable internal tri-state buffers.

Alternatively, in the configuration of Figure 2, the host processor's access time to read or write the external RAM is reduced by means of the extra logic gates shown. The only limitations are the CPU's strobe width, the access time of the RAM, and the gate delays. Note that in this configuration, the host must still activate the signals $\overline{\text{SELECT}}$, $\overline{\text{STRBD}}$. etc. in order to access the ACE's internal registers.

The major disadvantage of the DMA configuration is the 1553 terminal must steal bandwidth from the host processor each time a word or group of words needs to be transferred to/from RAM. This includes the "CPU-t-1553" arbitration time, the actual word(s) transfer time, and the "1553-to-CPU" arbitration time. In addition, the host must utilize additional bandwidth when it proceeds to access the same words in the system RAM.

Another problem involves the request/grant handshake. When the 1553 terminal issues a DMA request, it must receive a grant within an allotted time interval. With some processors, it's possible for the longest instruction cycle time (which cannot be interrupted) to be longer than the terminal's allowable request-to-grant time. If the terminal fails to receive a grant within the specified time window, a handshake timeout mechanism will abort processing of the current 1553 message. As a result, message data may be lost and/or messages may need to be retried.

The problems described above become further magnified when interfacing two or more 1553 terminals to the same processor. In such a scenario, the two terminals need to be granted access to the processor buses sequentially. The possibility of a DMA timeout for the second (or third, etc.) device requesting bus access is higher than for a single terminal. This is because, in addition to having to wait for the CPU grant, the second terminal must wait for the first terminal to complete its word transfer[s]).

SHARED RAM CONFIGURATION

Figures 3 and 4 illustrate shared RAM interfaces between 16-bit and 8bit processors, respectively, and an ACE terminal. In a shared RAM interface, tri-state buffers are provided between the processor address and data buses and the buses connecting to the 1553 buffer memory and terminal logic. In this configuration, the processor always has access to its own buses. That is, the ACE will never request the use of the CPU buses. This provides the advantage of allowing the ACE to access the buffer RAM while the CPU is able to simultaneously use its buses to access its memory or I/O. In this way, a shared RAM interface utilizes less of the CPU's bandwidth than does a DMA interface.

In the 16-bit buffered mode, note that TRANSPARENT /BUFFERED is strapped to logic "0", while 16/8-BIT is connected to logic " 1". The input signal ADDR_LAT may be used to demultiplex the address bus. For example, for some Intel microprocessors (eg., 8051, 80186), ALE should be connected to ADDR_LAT. If not used, ADDR_LAT should be connected to logic "1."

In the buffered configuration, the host initiates an access to the ACE's internal RAM or registers by asserting the signals $\overline{\text{SELECT}}$, $\overline{\text{STRBD}}$, MEM/REG and RD/$\overline{\text{WR}}$, $\overline{\text{SELECT}}$ and MEM/$\overline{\text{REG}}$ are generally the outputs of an address decoder. $\overline{\text{SELECT}}$ must be asserted low to access the ACE's RAM or registers. MEM/$\overline{\text{REG}}$ must be presented high for memory access and low for register access. If POLARITY_SEL is logic " I ", RD/$\overline{\text{WR}}$ must be presented high to read and low to write; if POLARITY_SEL is logic "O", RD/$\overline{\text{WR}}$ must be presented low to read and high to write. $\overline{\text{STRBD}}$ is the main processor control input to the ACE to control the length of an access cycle.

There are two variations of the buffered mode interface. Most microprocessors contain a "strobe/ acknowledge" (wait state) type of handshake for implementing a shared RAM interface. To support this type of handshake, the input $\overline{\text{ZERO\_WAIT}}$ must be strapped to logic "1." In this configuration, the ACE will assert its $\overline{\text{READYD}}$ output signal when write data has been latched or read data is available to the host. However, some 8-bit microcontrollers, such as Intel's 8051 (and a few 16-bit processors such as Analog Device's ADSP-2101) **do not** have a "strobe/ acknowledge" type of handshake mechanism. In this case, the ACE's "zero wait" interface configuration may be used by strapping $\overline{\text{ZERO\_WAIT}}$ to logic "0". In this mode, the
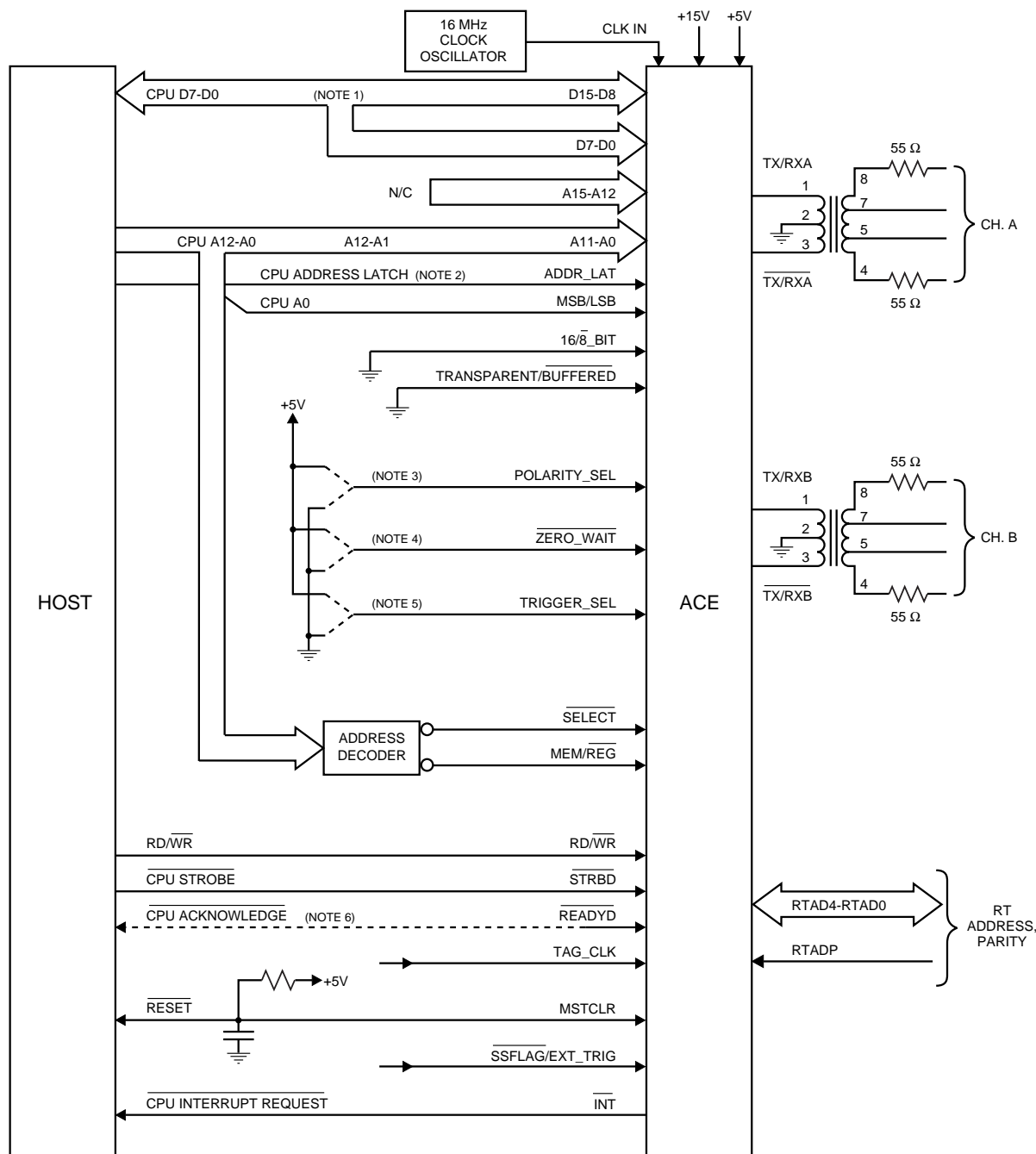
FIGURE 3. 16-BIT BUFFERED MODE
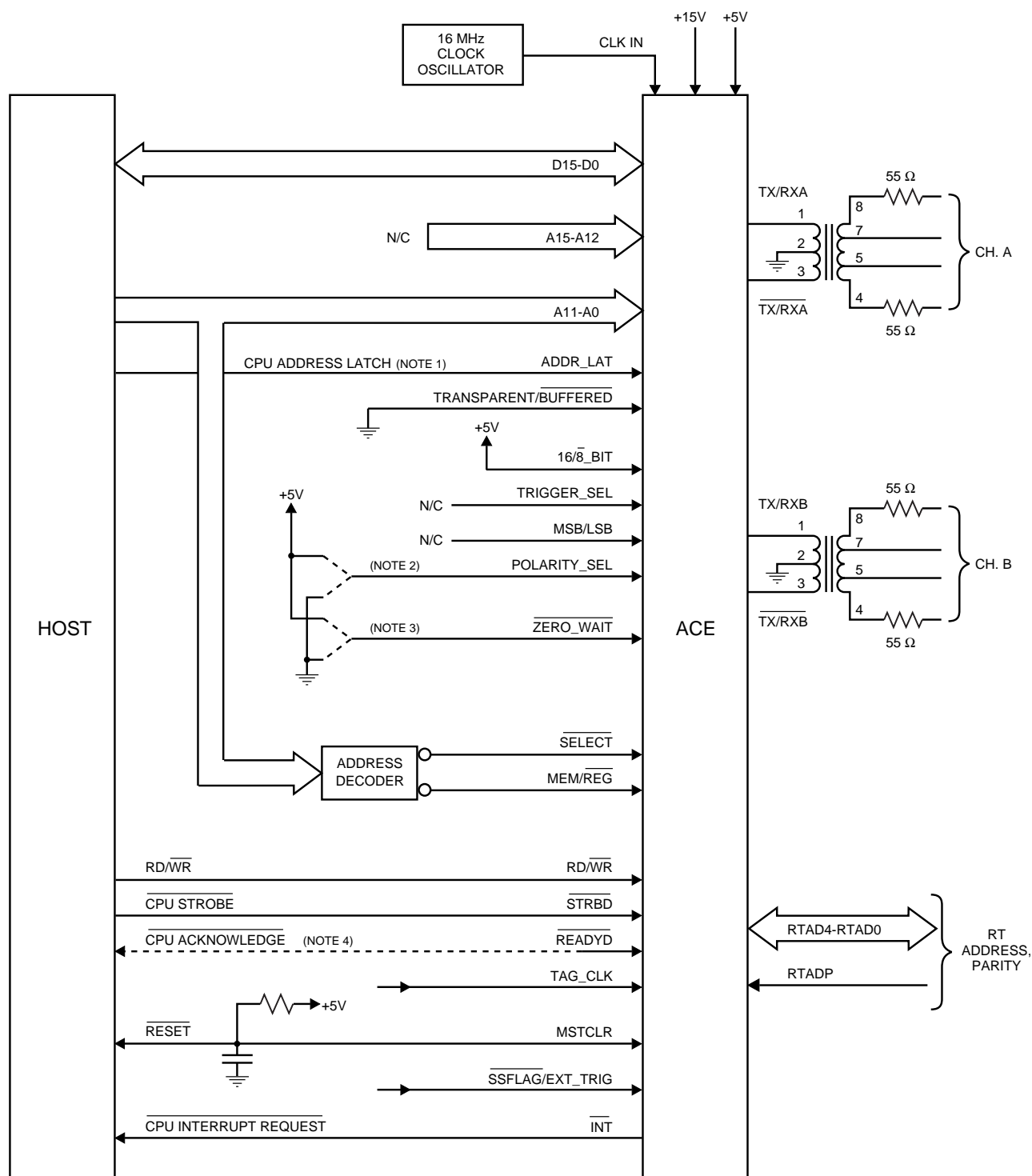
NOTES:

1. CPU ADDRESS LATCH SIGNAL PROVIDED BY PROCESSORS WITH MULTIPLEXED ADDRESS/DATA BUSES.
2. IF POLARITY_SEL = "1", RD/$\overline{\text{WR}}$ IS HIGH TO READ, LOW TO WRITE.
   IF POLARITY_SEL = "0", RD/$\overline{\text{WR}}$ IS LOW TO READ, HIGH TO WRITE.
3. $\overline{\text{ZERO\_WAIT}}$ SHOULD BE STRAPPED TO LOGIC "1" FOR NON-ZERO WAIT INTERFACE AND TO LOGIC "0" FOR ZERO WAIT INTERFACE.
4. CPU ACKNOWLEDGE PROCESSOR INPUT ONLY FOR NON-ZERO WAIT TYPE OF INTERFACE.

NOTES:

1. CPU D7-D0 CONNECTS TO BOTH D15-D8 AND D7-D0.
2. CPU ADDRESS LATCH SIGNAL PROVIDED BY PROCESSORS WITH MULTIPLEXED ADDRESS/DATA BUFFERS.
3. IF POLARITY_SEL = "1", THEN MSB/LSB SELECTS THE MOST SIGNIFICANT BYTE WHEN LOW, AND THE LEAST SIGNIFICANT BYTE WHEN HIGH.
   IF POLARITY_SEL = "0", THEN MSB/LSB SELECTS THE LEAST SIGNIFICANT BYTE WHEN LOW, AND THE MOST SIGNIFICANT BYTE WHEN HIGH.
4. ZERO WAIT SHOULD BE STRAPPED TO LOGIC "1" FOR NON-ZERO WAIT INTERFACE AND TO LOGIC "0" FOR ZERO WAIT INTERFACE.
5. OPERATION OF TRIGGER_SELECT INPUT IS AS FOLLOWS:
   FOR NON-ZERO WAIT INTERFACE (ZERO WAIT = "1"):
   IF TRIGGER_SEL = "1", THEN INTERNAL 16-BIT
   TRANSFERS ARE TRIGGERED BY THE MOST SIGNIFICANT BYTE TRANSFER READ ACCESSES AND BY THE LEAST SIGNIFICANT BYTE TRANSFER FOR WRITE ACCESSES.
   IF TRIGGER_SEL = "0", THEN INTERNAL 16-BIT TRANSFERS ARE TRIGGERED BY THE LEAST SIGNIFICANT BYTE TRANSFER FOR READ ACESSES AND BY THE MOST SIGNIFICANT BYT TRANSFER FOR WRITE ACCESSES.
   FOR ZERO WAIT INTERFACE (ZERO WAIT = "0"):
   IF TRIGGER_SEL = "1", THEN INTERNAL 16-BIT TRANSFERS ARE TRIGGERED BY THE LEAST SIGNIFICANT BYTE TRANSFER, FOR BOTH READ AND WRITE ACCESSES.
   IF TRIGGER_SEL = "0", THEN INTERNAL 16-BIT TRANSFERS ARE TRIGGERED BY THE MOST SIGNIFICANT BYTE TRANSFER, FOR BOTH READ AND WRITE ACCESES.
6. CPU ACKNOWLEDGE PROCESSOR INPUT ONLY FOR NON-ZERO WAIT TYPE OF INTERFACE.

FIGURE 4. 8-BIT BUFFERED MODE

TABLE 1. SUMMARY OF 8-BIT OPERATION (NON-ZERO WAIT MODE)					
ZERO WAIT	TRIGGER SEL	POLARITY SEL	RD/ WR	FIRST BYTE TRANSFER	SECOND BYTE TRANSFER
1	1	1	1	MSB/LSB = 0; Valid A15-A0, MEM/REG; CPU ← RAM 15-8; Buffer 7-0 ← RAM 7-0	MSB/LSB = 1; A15-A0, MEM/REG = "Don't Care"; CPU ← Buffer 7–0
1	1	1	0	MSB/LSB = 0; A15-A0, MEM/REG = "Don't Care" CPU → Buffer 15-8	MS8/LSB = 1; Valid A15-A0, MEM/REG; Buffer 15-8 → RAM 15-8; CPU → RAM 7-0
1	1	0	1	MSB/LSB = 1; Valid A15-A0, MEM/REG; CPU ← RAM 15-8; Buffer 7-0 ← RAM 7-0	MSB/LSB = 0; A15-A0, MEM/REG = "Don't Care"; CPU ← Buffer 7-0
1	1	0	0	MSB/LSB = 1; A15-A0, MEM/REG = "Don't Care" CPU → Buffer 15-8	MSB/LSB = 0; Valid A15-A0, MEM/REG; Buffer 15-8 → RAM 15-8; CPU → RAM 7-0
1	0	1	1	MSB/LSB = 1; Valid A15-A0, MEM/REG CPU ← RAM 15-8; Buffer 15-8 ← RAM 15-8	MSB/LSB = 0; A15-A0, MEM/REG = "Don't Care" CPU ← Buffer 7-0
1	0	1	0	MSB/LSB = 1; A15-A0, MEM/REG = "Don't Care" CPU → Buffer 7-0	MSB/LSB = 0; Valid A15-A0, MEM/REG; Buffer 7-0 → RAM 7-0; CPU → RAM 15-8
1	0	0	1	MSB/LSB = 0; Valid A15-A0, MEM/REG CPU ← RAM 7-0; Buffer 15-8 ← RAM 15-8	MSB/LSB = 1; A15-A0, MEM/REG = "Don't Care" CPU ← Buffer 15-8
1	0	0	0	MSB/LSB = 0; A15-A0, MEM/REG = "Don't Care" CPU → Buffer 7-0	MSB/LSB = 1; Valid A15-A0, MEM/REG; Buffer 7 0 → RAM 7-0 CPU → RAM 1-58

CPU is able to release $\overline{\text{STRBD}}$ high **before** the ACE asserts $\overline{\text{READYD}}$ low. This mode takes advantage of the ACE's internal address and data latches and added control logic and serves to minimize the amount of "glue" logic required.

8-BIT MODE

In the 8-bit buffered mode (Figure 4), the input 16/8-BIT must be strapped to logic "0" and the CPU's data bus must be connected to D15-D8 and D7-D0. The LSB of the processor address bus (processor A0) must be connected to the input MSB/LSB for upper/lower byte selection. In order to accommodate the different "A0" and "byte ordering" conventions of various 8-bit processor families, the inputs POLARITY_SEL and TRIGGER_SEL need to be strapped to logic "0" or " I ". See Figure 4 and Table 1.

The operation of the ACE's 8-bit buffered mode is essentially the same as the 16-bit mode, with the following exceptions:

(1) For the "non zero wait" mode ($\overline{\text{ZERO\_WAIT}}$ =

logic " I "): For a write cycle, the first byte is written to an internal buffer register, the second byte is written directly to RAM. When the second byte is written to RAM, the first byte is simultaneously transferred from the buffer register to RAM. The address (A15-A0 and MEM/$\overline{\text{REG}}$) is "don't care" during the first byte transfer, but must be valid during the second byte transfer. The handshake output $\overline{\text{READYD}}$ is asserted low after both byte transfers.

(2) For a read cycle in the "non zero wait" mode, the first byte is read directly from RAM. While the first byte is read by the CPU, the second byte is being stored in an internal buffer register. The processor may then read the second byte from the buffer. The address (A15-A0 and MEM/$\overline{\text{REG}}$) must be valid during the first byte read, but is "don't care" during the second byte read. The handshake output $\overline{\text{READYD}}$ is asserted after both byte transfers.

(3) The write cycle for the 8-bit "zero wait" mode

ZERO WAIT	TRIGGER SEL	POLARITY SEL	RD/ WR	FIRST BYTE TRANSFER	SECOND BYTE TRANSFER
				TABLE 1. SUMMARY OF 8-BIT OPERATION continued (NON-ZERO WAIT MODE)	
0	1	1	1	MSB/LSB = 0; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU ← Buffer 15-8 (last A15A0 and MEM/REG)	MSB/LSB = 1; A15-A0 and MEM/$\overline{\text{REG}}$ Valid for Next Word CPU← Buffer 7-0 (last A15A0 and MEM/REG) then, Buffer 15-0 ← RAM 15-0
0	1	1	0	MSB/LSB = 0; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU ← Buffer 15-8	MS8/LSB = 1; Valid A15-A0, MEM/$\overline{\text{REG}}$; CPU → RAM 7-0; then, Buffer → RAM 15-0
0	1	0	1	MSB/LSB = 1; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU ← Buffer 15-8 (last A15A0 and MEM/REG)	MSB/LSB = 0; A15-A0 and MEM/$\overline{\text{REG}}$ Valid for Next Word CPU ← Buffer 7-0 (last A15A0 and EM/REG) then, Buffer 15-0 ← RAM 15-0
0	1	0	0	MSB/LSB = 1; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU → Buffer 15-8	MSB/LSB = 0; Valid A15-A0, MEM/$\overline{\text{REG}}$; CPU → Buffer 7-0 then, Buffer → RAM 15-0
0	0	1	1	MSB/LSB = 1; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU ← Buffer 7-0 (last A15A0 and MEM/REG)	MSB/LSB = 0 A15-A0 and MEM/$\overline{\text{REG}}$ Valid for Next Word CPU ← Buffer 15-8 (last A15A0 and MEM/REG) then, Buffer 15-0 ← RAM 15-0
0	0	1	0	MSB/LSB = 1; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU → Buffer 7-0	MSB/LSB = 0; Valid A15-A0, MEM/$\overline{\text{REG}}$; CPU → Buffer 15-8 (last A15A0 and MEM/REG) then, Buffer 15-0 →RAM 15-0
0	0	0	1	MSB/LSB = 0; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU ← Buffer 7-0 (last A15A0 and MEM/REG)	MSB/LSB = 1; A15-A0 and MEM/$\overline{\text{REG}}$ Valid for Next Word CPU ← Buffer 15-8 (last A15A0 and MEM/REG) then, Buffer 15-0 ← RAM 15-0
1	0	0	0	MSB/LSB = 0; A15-A0, MEM/$\overline{\text{REG}}$ = "Don't Care" CPU → Buffer 7-0	MSB/LSB = 1; Valid A15-A0, MEM/$\overline{\text{REG}}$; CPU → Buffer 15-8 then, Buffer → RAM 15-0

($\overline{\text{ZERO\_WAIT}}$ = logic "0") is identical to that for the 8-bit "non zero wait" mode, except for the operation of the $\overline{\text{READYD}}$ output. In the "zero wait" mode, $\overline{\text{READYD}}$ stays low during the entire first byte transfer cycle. During the second byte transfer time, $\overline{\text{READYD}}$ stays low until the time that the control input is cleared high, or possibly while $\overline{\text{STRBD}}$ is still asserted low, depending on the width of the $\overline{\text{STRBD}}$ input. $\overline{\text{READYD}}$ then goes high (indicating "not ready") and remains high until the internal 16-bit transfer of data between the ACE's internal buffer registers and internal RAM has been completed.

When using the 8-bit (or 16-bit) "zero wait" mode, it is assumed there is no "acknowledge" input on the processor to connect the ACE's $\overline{\text{READYD}}$ output to. However, the CPU may still poll the READYD output. **It should not initiate another transfer cycle to the ACE while $\overline{\text{READYD}}$ is logic "1".** Alternatively, the CPU may insert software wait states between successive accesses to the ACE. The required wait time varies with

mode of operation (RT mode is generally the longest depending on what options are used), but as a rule of thumb: 2.56 µs is sufficient with a 16 MHz clock input; 3.4 µs is sufficient with a 12 MHz clock input.

(4) For the read cycle for the 8-bit "zero wait" mode: It is important to realize that for a given read cycle, the internal RAM address (including the value of MEM/$\overline{\text{REG}}$) is the address that was presented on the ACE's address inputs (A15-A0) during the **second byte read of the PREVIOUS (word) READ cycle.** Therefore, in order to read a word (2 bytes) in this mode, the first byte read access is a "dummy cycle" as far as the data bus (D15-8, D7-D0) is concerned; however, the address (A15-A0 and MEM/$\overline{\text{REG}}$) for the first word to be read must be presented at this time.

That is, in order to initiate a series of word accesses, the CPU must **FIRST perform a "second byte" read access (per Table 1)** with the valid address and value of MEM/$\overline{\text{REG}}$ for the first
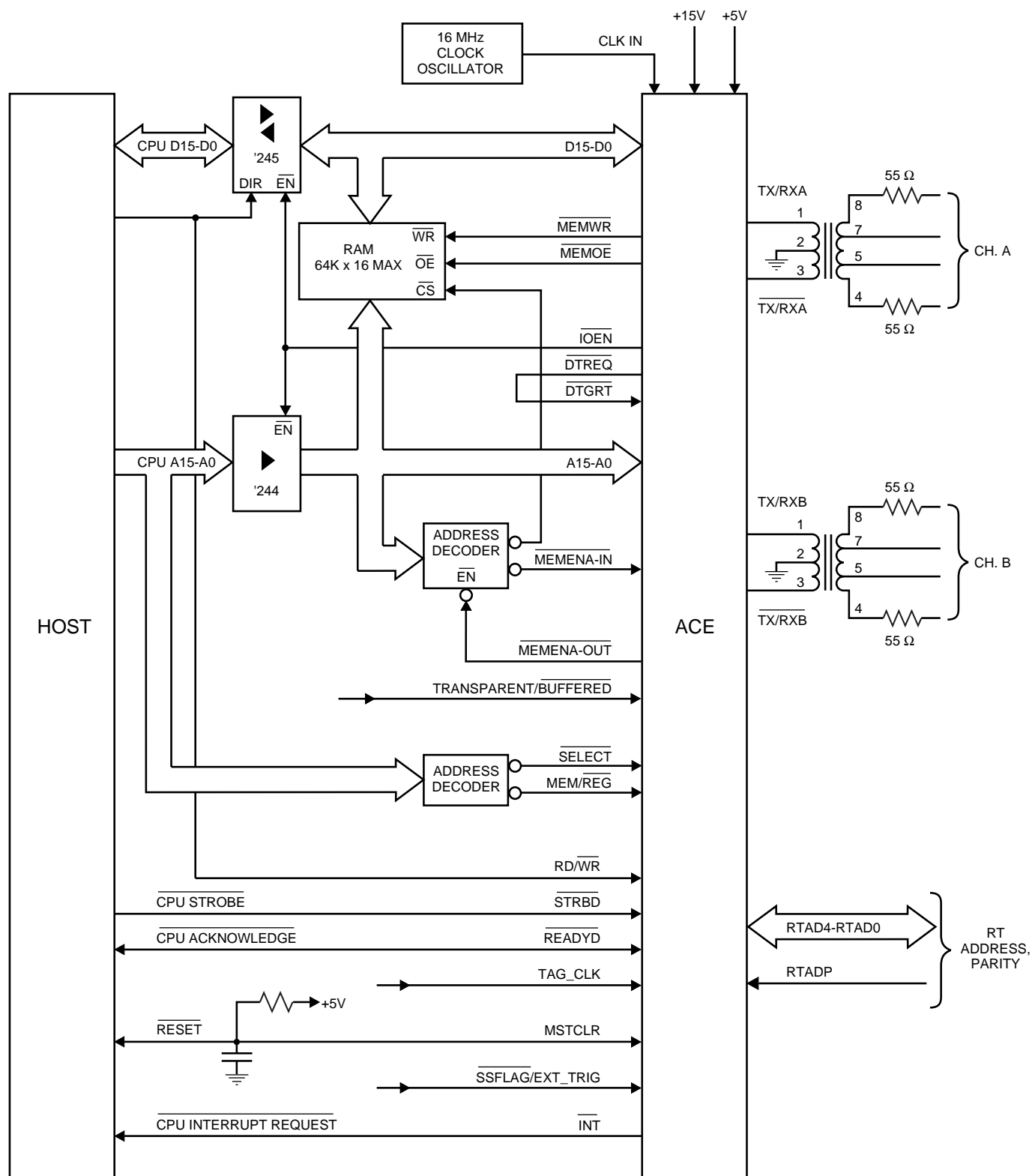
FIGURE 5. 16-BIT TRANSPARENT MODE

word presented. The CPU should then perform the "first byte" and "second byte" read cycles to acquire the data from the address (including MEM/$\overline{\text{REG}}$) that was previously presented. As far as the ACE is concerned, the address presented on A15-A0 and MEM/$\overline{\text{REG}}$ is "don't care" during the "first byte" transfer. On the second byte transfer, the valid address for the next word to be read must be presented on A15-A0 and MEM/$\overline{\text{REG}}$. The CPU may then read the subsequent word by performing a first byte read, followed by a second byte read, etc.

Note that $\overline{\text{READYD}}$ will remain low through the end of the first "zero wait" read cycle, but will transition to logic "1" during (or possibly after) the $\overline{\text{STRBD}}$ pulse for the second byte transfer cycle. During the time that $\overline{\text{READYD}}$ is logic "1", the RAM data for the address presented is transferred from the ACE's internal RAM to the two internal buffer registers. When $\overline{\text{READYD}}$ is re-asserted to logic "0" (2.6 μs max later with a 16 MHz clock), the CPU may then proceed to read the first byte, then the second byte. Note that the address inputs A15-A0 are "don't care" during the first byte read transfer. The address presented on A15-A0 during the second byte read transfer must be valid to provide the RAM address for the next 2-byte read cycle.

BOARD SPACE, RAM SIZE

An integrated 1553 terminal, such as the ACE, has an inherent advantage over a "discrete" 1553 mechanization in implementing a shared RAM interface. That is, it is able to minimize PC board space requirements by incorporating all of the elements within a single package: transceiver, protocol, RAM, and buffers. The BU-65170RT, BU-61580 BC/RT/MT, and BU-61590 (Universal) ACE terminals include 4K words of internal RAM. The BU-61585 BC/RT/MT ACE contains an additional 8K X 17 of internal RAM. The BU-61585 supports two different organizations for its internal memory: (1) 8K X 17, with parity generation and checking on all RAM accesses; and (2) 12K X 16 without the RAM parity functions.

While the 4K (or 8K or 12K) of internal RAM provided is sufficient for most applications, some systems will require a larger buffer memory.

While most current generation systems employ 16-bit or 32-bit processors, there are still many that make use of 8-bit microprocessors or 8-bit microcontrollers. In these applications, a small amount of processing power is typically needed to perform simple command and control functions, and system self-test. Some simple sensors are examples of such applications. In other cases, the low-end CPUs serve as 1553 I/O processors for a faster host.

In some applications, "zero wait" mode is used in

TABLE 2. ADDRESS AND DATA BUS BUFFER OPERATION FOR 16-BIT MODES					
TRANSPARENT BUFFERED MODE	CPU/1553 ACCESS	READ/WRITE	INT/EXT RAM (Note 1)	ADDRESS BUFFERS	DATA BUFFERS
Buffered	CPU	WRITE	INT	→	→
Buffered	CPU	READ	INT	→	←
Buffered	1553	WRITE	INT	Z	Z
Buffered	1553	READ	INT	Z	Z
Transparent	CPU	WRITE	INT	→	→
Transparent	CPU	WRITE	EXT	→	Z
Transparent	CPU	READ	INT	→	←
Transparent	CPU	READ	EXT	→	Z
Transparent	1553	WRITE	INT	←	←
Transparent	1553	WRITE	EXT	←	←
Transparent	1553	READ	INT	←	←
Transparent	1553	READ	EXT	←	→

NOTES:

(1) The ACE distinguishes between internal (INT) and external (EXT) by means of the input signal $\overline{\text{MEMENA-IN}}$. MEMENA-IN is low for internal RAM access, high for external RAM access.

(2) "→" indicates buffers enabled in direction from external processor bus towards internal memory bus.
    "←" indicates buffers enabled in direction from internal memory bus towards external processor bus.
    "Z" indicates buffers are in tri-state or disabled condition.

TABLE 3. INTERNAL TRANSFER SEQUENCE LENGTHS		
SEQUENCE	@ 16 MHz	@ 12 MHz
BC, RT, or Selective Monitor single word transfer	312 ns	416 ns
Monitor 2 word transfer	562 ns	750 ns
BC Start-of-Message (SOM) sequence	1.81 µs	2.41 µs
BC End-of-Message (EOM) sequence	1.31 µs	1.75 µs
BC Retry sequence	1.76 µs	2.35 µs
RT Start-of-Message (SOM) sequence	1.56 - 2.56 µs	2.08 - 3.41 µs
RT End-of-Message (EOM) sequence	0.56 - 1.31 µs	0.75 - 1.75 µs
Selective Monitor Start-of-Message (SOM) sequence	2.31 µs	3.08 µs
Selective Monitor End-of-Message (EOM) sequence	812 ns	1.08 µs

conjunction with a system DMA controller. In this configuration, the host is able to "steal" time from the DMA controller while the 1553 terminal is performing its internal transfers. In the same system, the host uses the normal "strobe/acknowledge" handshake ("non zero wait" mode) for its own programmed I/O accesses.

TRANSPARENT MODE

Figure 5 illustrates a variation of the ACE's shared RAM mode, the transparent configuration. In the transparent mode, the host is able to access up to 64K of external RAM. Like the buffered mode, the host is able to use its data/address buses while the ACE **simultaneously** accesses its internal or external shared RAM, Note that in the transparent configuration, external tri-state buffers are required to isolate the CPU's data/address buses from the ACE's RAM buses.

Table 2 summarizes the operation of the ACE's tri-state address and data bus buffers for the 16-bit buffered and transparent modes.

EFFECT OF CONTENDED ACCESSES ON PROCESSOR ACCESS TIME

For the shared RAM interface configuration (either buffered or transparent), the ACE contains internal contention resolution logic to arbitrate the use of its internal buses between RAM accesses by the host processor and the ACE's infernal 1553 memory access logic. For **most (in general, 99% or more)** accesses by the host processor, the 1553 memory access logic will not be accessing the ACE shared RAM. The 1553 logic accesses the RAM in multi-word Start-of-Message (SOM) and End-of-Message (EOM) sequences for the BC, RT, and Selective Message Monitor modes. For these modes, it also performs single word accesses for words transmitted or received on the

1553 bus. In the Word Monitor mode, the ACE performs a 2-word write operation to RAM following every word received on the 1553 bus.

Using a 16 MHz clock, the average time for an **uncontended** processor access, defined as being the time from the falling edge of the $\overline{STRBD}$ input to the falling edge of the $\overline{READYD}$ acknowledge output, is about 250 ns. That is, the $\overline{STRBD}$ (low)-to-$\overline{IOEN}$ (low) time of 62 ns typ, 85 ns max. plus the $\overline{IOEN}$ (low)-to-$\overline{READYD}$ (low) time of 187.5 ns typical.

If the processor tries to access the ACE's RAM or registers after the internal 1553 logic has begun a single-word or multi-word transfer sequence. the CPU will be held off until the internal access has been completed. The worst case holdoff occurs if $\overline{STRBD}$ is brought low **just after** the beginning of an internal transfer cycle. The lengths of the infernal transfer sequences are listed in Table 3.

To estimate the effect of contention access on processor access time, consider the following example: The ACE is configured for RT mode. It will transmit or receive a 32-word message every 10 ms. Assume the ACE uses a 16 MHz clock input.

The effect of contention will be to increase the average CPU access time. The effect of each single-word or multi-word internal transfer sequence is based on a statistical average. Define the periodicity of the overall message sequence to be T (T = 10 ms in this example).

Consider a given transfer sequence with a time duration tx. The worst case holdoff time due to this transfer sequence = t_X.

The probability of encountering a particular infernal transfer sequence during a CPU access (resulting in a contended access) is t_X/T.

If there is access contention, the host could be

held off for the full infernal transfer time or, more likely, **some portion** of the full transfer time. Therefore, assuming that there is a contended access due to a particular internal transfer cycle, the **average** holdoff time will be one half of the infernal transfer time, or $0.5 \cdot t_X$.

The average additional CPU holdoff time attributable to any particular internal 1553 transfer cycle is, therefore:

(probability of contended access)•(average holdoff time)

$$= (t_X/T) \cdot (0.5 \cdot t_X)$$

$$= (0.5/T) \cdot t_X 2$$

The overall average holdoff time, or composite increase in contention time is the sum of the contributions from all of the individual internal transfer cycles. That is,

$$t_{holdoff} = \text{i} \; (0.5/T, \cdot t_i 2$$

For the example given, there are three types of internal transfer sequences: RT SOM, RT EOM, and 32 single-word Data Word transfers. The contributory effect on average access time must be considered for each of the three.

RT SOM sequence:

$$(0.5/T) \cdot t_X 2 =$$

$$(0.5/10,000) \cdot (2.56)^2 =$$

0.33 ns

RT EOM sequence:

$$(0.5/T) \cdot t_X 2 =$$

$$(0.5/10,000) \cdot (1.31)^2 =$$

0.09 ns

The 32 single word transfer cycles (for the individual Data Words):

$$(32) \cdot [(0.5/T) \cdot t_X 2] =$$

$$(32) \cdot [(0.5/10,000) \cdot (0.312)^2] =$$

0.15 ns

Therefore, for the example given, the **average** CPU holdoff time will be:

$$0.33 + 0.09 + 0.15 = \textbf{0.57 ns.}$$

Note that the average holdoff time is inversely proportional to the message cycle time, T. If the RT is assumed to be involved in message activity 100% of the time, T drops to 700 µs. In this case, the average holdoff time would increase to:

$$(0.57 \text{ ns})/(700 \text{ µs}/10 \text{ ms}) = \textbf{8.1 ns.}$$

Note that this "worst case" average contention time (based on RT mode with 32-word messages) increases the average access time from 250 ns to 258.1 ns, **an increase of only 3.2%.**

To conclude, while the effect of contended access can result in a worst case holdoff time of 2.56 µs, the **average** holdoff time for most "real world" bus traffic scenarios will be less than 10 ns, usually much less.

DUAL PORT RAM

Another possible interface configuration involves the use of the ACE with a true dual port RAM. Reference Figure 6. Note that in this configuration tristate buffers **are required** on the address and data buses to allow the host processor to access the ACE's infernal registers. In terms of performance--that is, minimizing CPU bandwidth requirements--this technique is the "Cadillac" of all the methods discussed. Except for a rare concurrent access to the same address, this configuration allows simultaneous access by both the host and the ACE. In addition, since it does not entail the arbitration time required for the shared RAM configuration, the only limitations on the host's access time are the CPU instruction cycle time and the speed of the RAM. In terms of design tradeoffs, these performance advantages need to be weighed against the relatively high cost and PC board real estate of the dual port RAM chip(s) and the tri-state buffers.

TABLE 4. SUMMARY OF INTERFACE CONFIGURATIONS					
INTERFACE CONFIGURATION	MAXIMUM RAM SPACE (WORDS)	CONTENTION PROBLEMS	CPU BANDWIDTH REQUIRED	RELATIVE BOARD SPACE	RELATIVE COST
BUFFERED	Internal 4K (12 K)	None	Low/Medium	Minimal	Lowest
TRANSPARENT	64 K	None	Low/Medium	High	Medium
DMA	64 K	Possible	Highest	Low/Medium	Low/Medium
DMA with "Speedup" Logic	64 K	Possible	High	Medium	Medium
Dual Port RAM	64 K	None	Lowest	Highest	Highest

TRADEOFF SUMMARY OF INTERFACE CONFIGURATIONS

The particular requirements of a given application will dictate which interface configuration is best. For most situations where 4K (or possibly 1 2K) words of RAM is adequate, use of the ACE's internal RAM in the buffered mode will minimize both cost and board space and be the optimal solution. For applications where a larger buffer memory is required (up to 64K words), one of the other configurations must be used. As shown in Table 4, there are trade-offs between cost, board space and CPU bandwidth utilization among the various configurations.

For applications requiring external RAM, DMA is the lowest cost technique, but has potential contention problems and utilizes more CPU bandwidth. At the other extreme, the use of dual port RAM has no contention problems and minimizes the amount of CPU bandwidth required to access the 1553 RAM, but at a substantial penalty in cost and board space. Transparent mode represents a compromise between the two. It should be noted that "CPU BANDWIDTH REQUIRED" refers to the sum of the bandwidth that the ACE "steals" from the CPU when accessing external RAM (for 1553 transfers in a DMA configuration) plus the time required for the CPU to access infernal or external RAM, including contention time.

PROCESSOR INTERFACES

Figures 7 through 15 illustrate buffered mode interfaces between several microprocessors and an ACE terminal.

Figure 7 shows for the Intel 8051 8bit microcontroller. This diagram illustrates the 8-bit zero wait configuration. Figure 8 shows the Analog Devices ADSP-2101 DSP processor connected using the 16-bit zero wait configuration. The zero wait mode is required for these two processors since neither has an "Acknowledge" type of input signal to support wait states.

Figures 9 through 12 show the interface to Motorola 68000, Intel 80186 and 80286, and Performance (PACE) 1750 16-bit microprocessors. Figure 13 illustrates the interface to Intel's 80C196 16-bit microcontroller. Figures 14 and 15 show connections to the Motorola 68020 and Intel i960 32-bit processors. Since all of these processors (Figures 9 through 15) have an "Acknowledge" type input, all of these diagrams illustrate the 16-bit buffered non-zero wait interface configuration.

FIGURE 6. 16-PORT DUAL RAM INTERFACE

FIGURE 7. INTEL 8051-TO-ACE
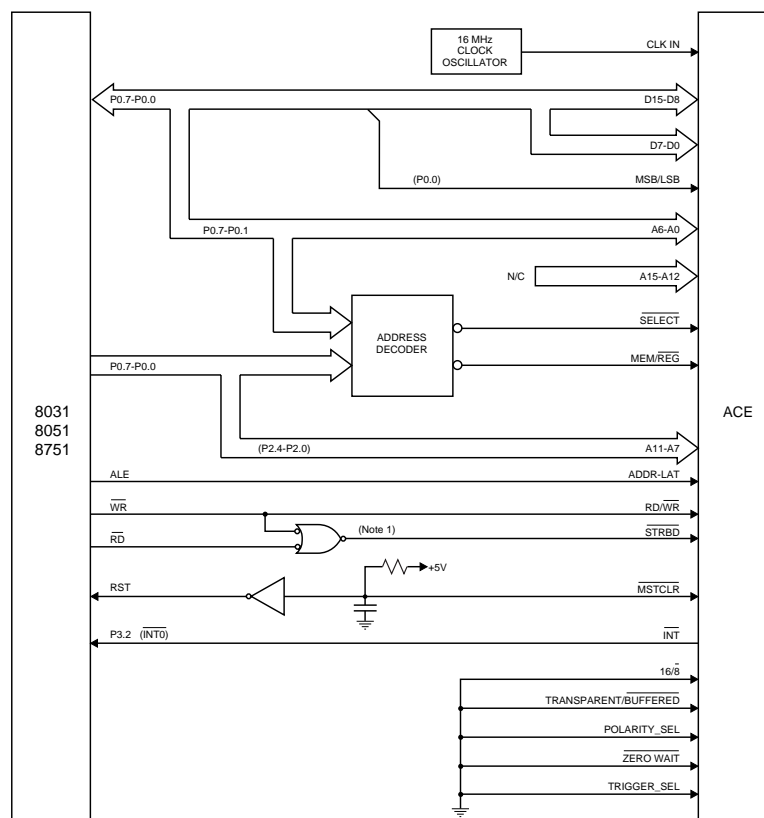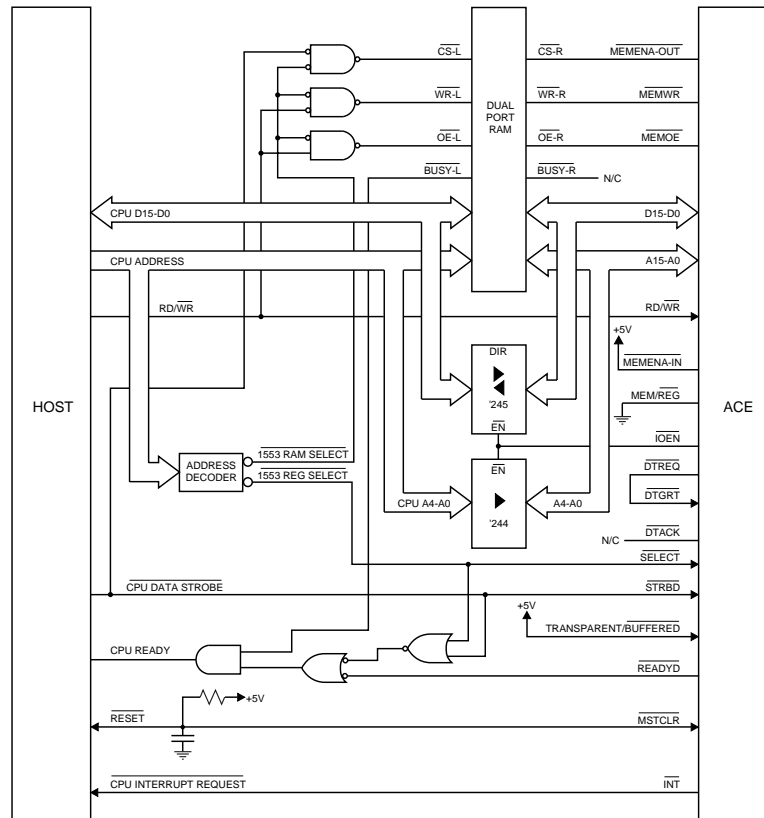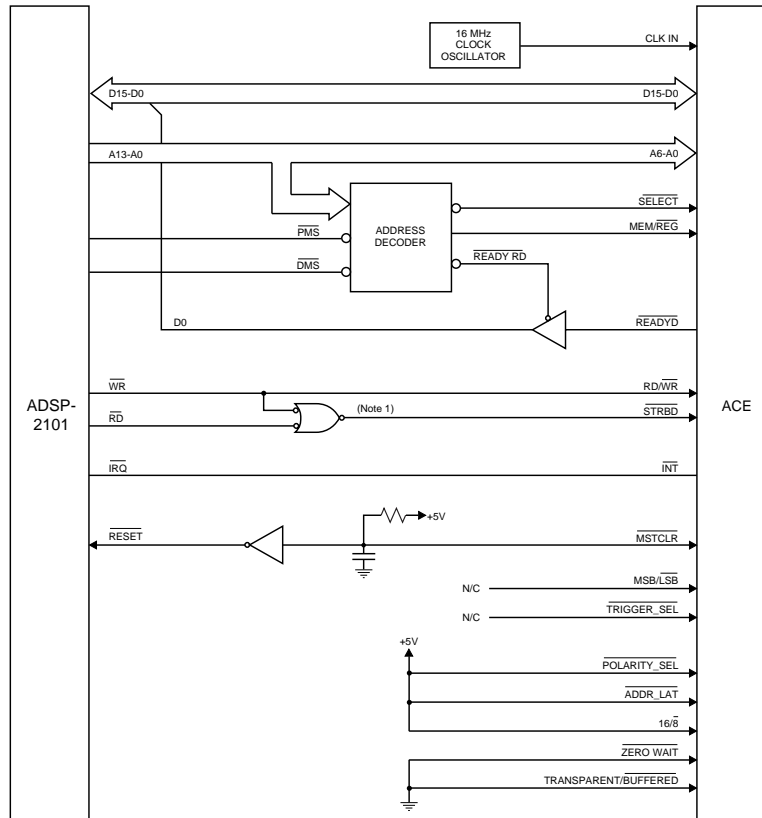
FIGURE 8. ANALOG DEVICES ADSP-2101-TO-ACE
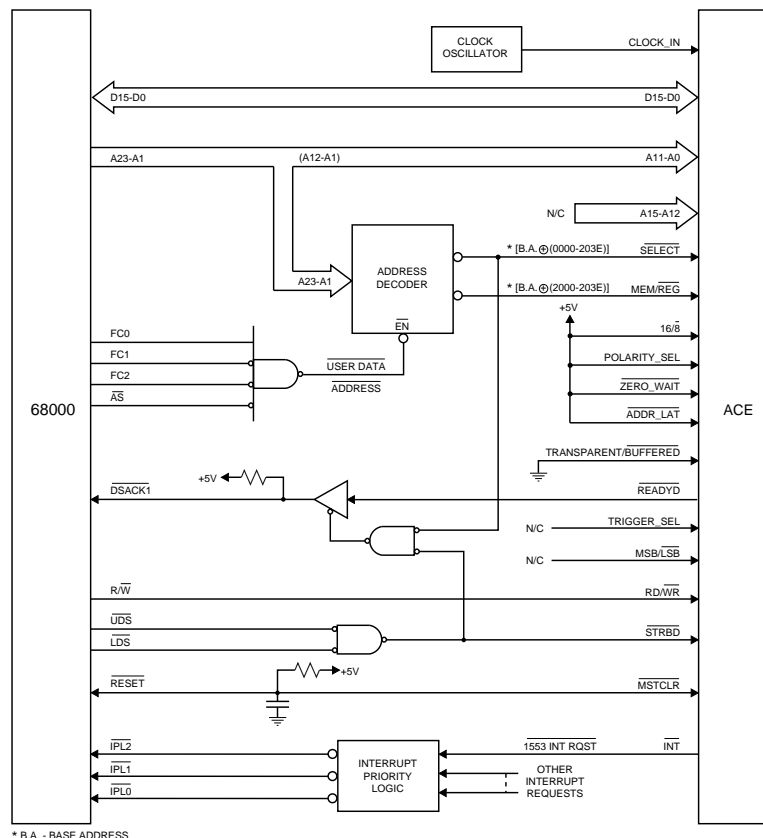
* B.A. - BASE ADDRESS

FIGURE 9. MOTOROLA 6100-TO-ACE
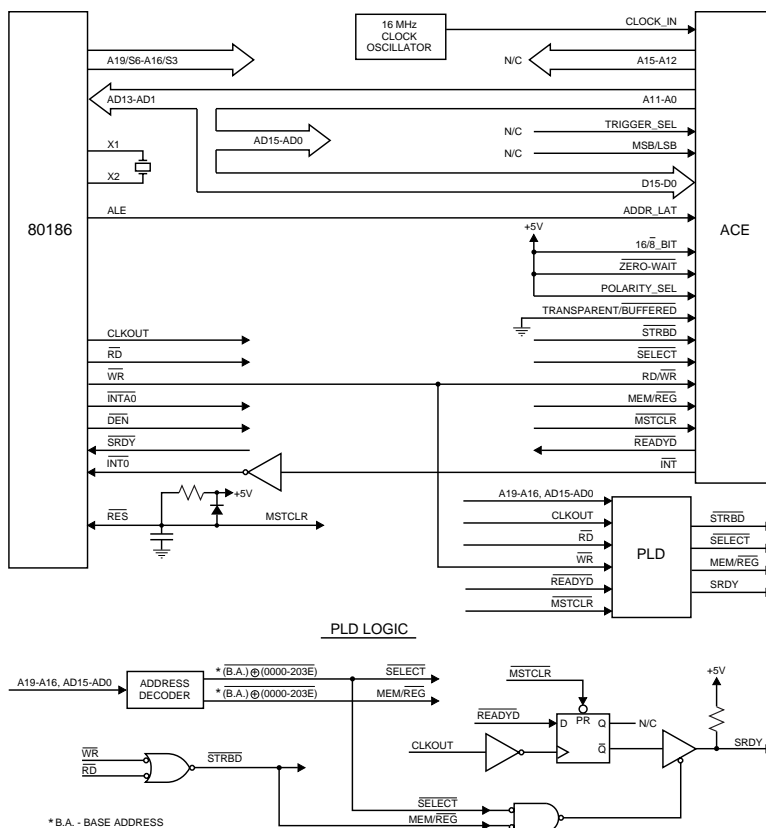
FIGURE 10. INTEL 80186-TO-ACE
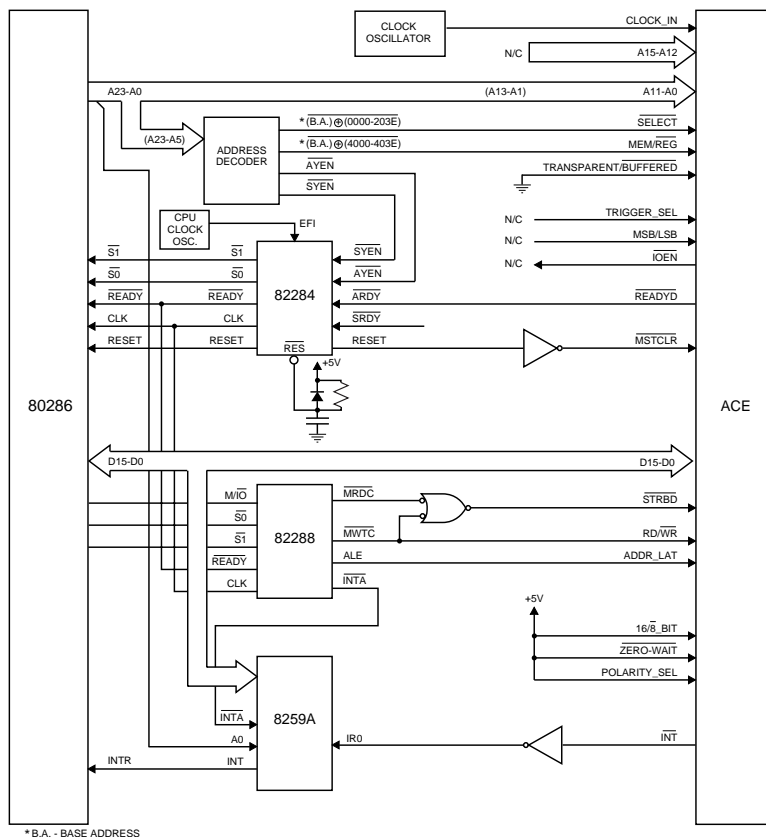
FIGURE 11. INTEL 80286-TO-ACE
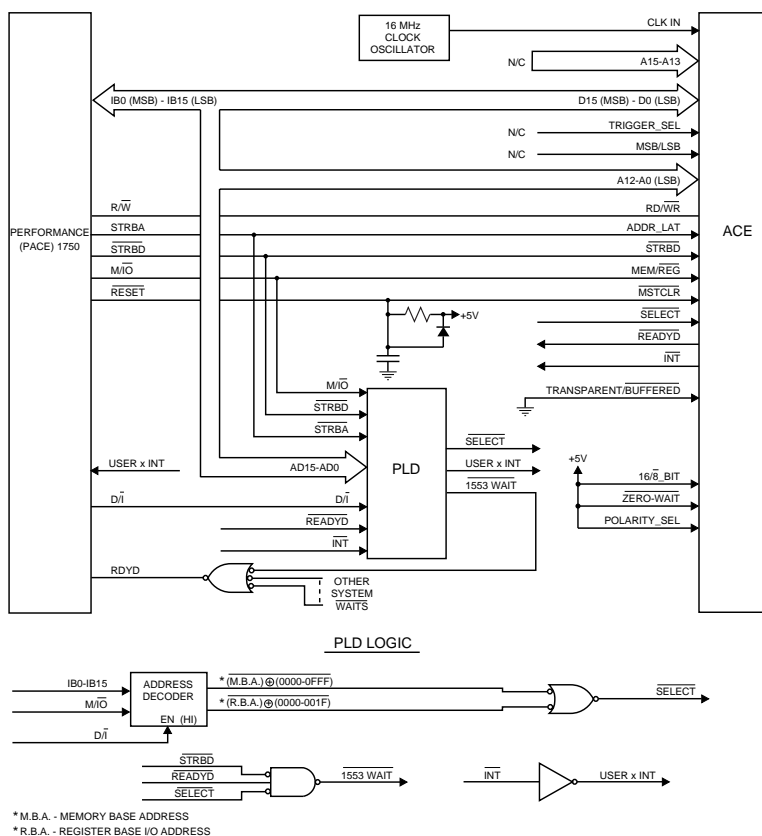
FIGURE 12. PERFORMANCE (PACE) 1750-TO-ACE
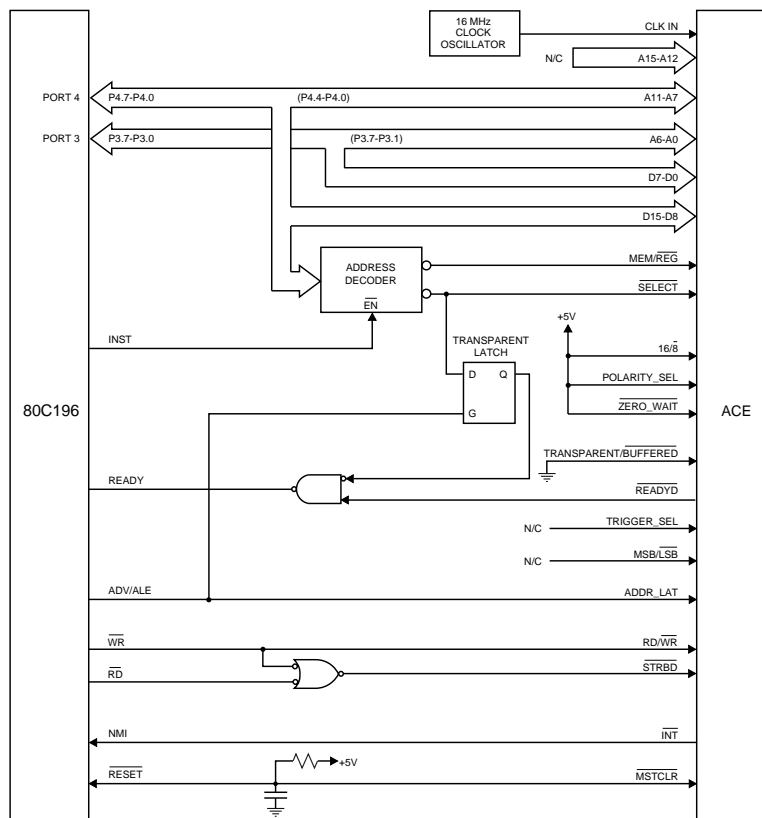
FIGURE13. INTEL 80C196-TO-ACE
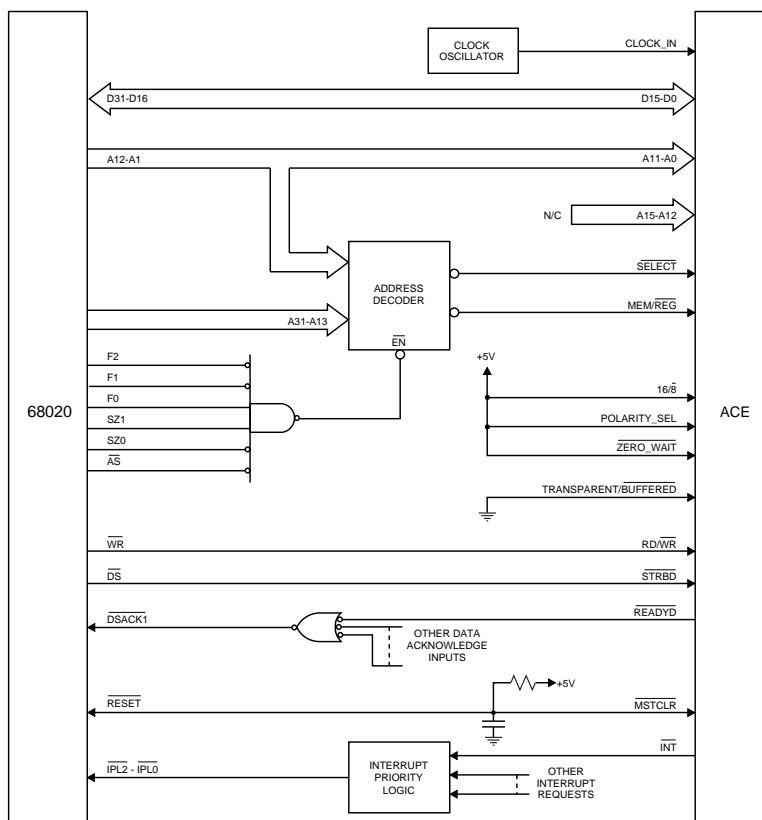
FIGURE 14. MOTOROLA 68020-TO-ACE
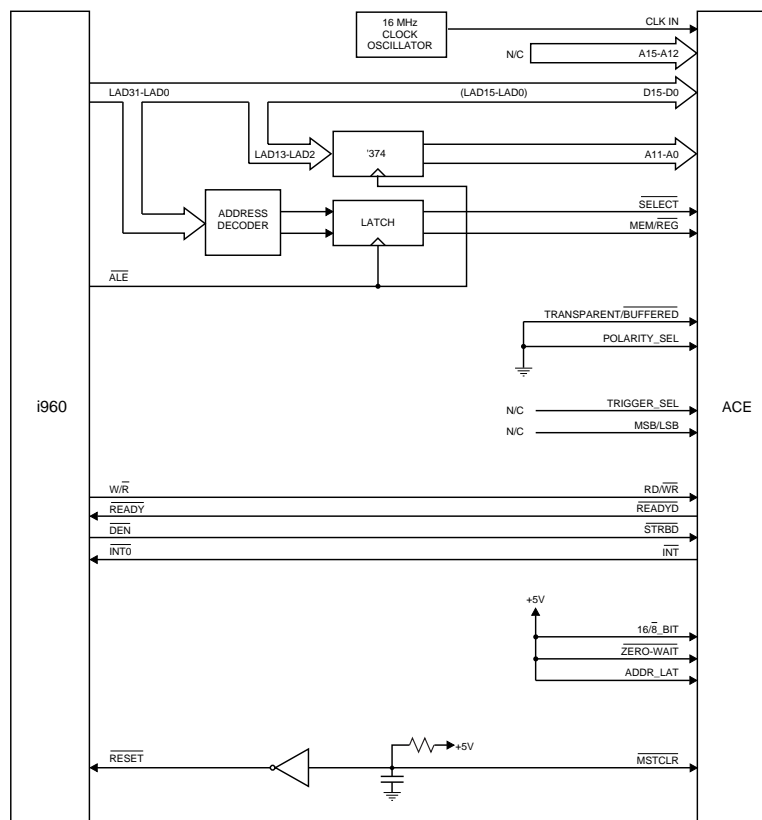
FIGURE 15. INTEL 1960-TO-ACE

ACE RT MEMORY MANAGEMENT OPTIONS

ABSTRACT

The software design of an RT for a microprocessor-based system is a common MIL-STD-1553 interface requirement. System requirements and Interface Control Drawings (ICDs) often provide a multitude of demands on the software engineer. Among these requirements are the need to ensure data integrity and data consistency, the need to perform bulk (multi-message) data transfers, the demand for interrupt-driven software, and the need to offload the operation of the host CPU to the greatest degree possible This latter requirement is reflected in such specifications as spare CPU bandwidth.

This application note provides a discussion of the application of the ACE's various memory management modes: single message, circular buffered, and double buffered. Other issues discussed are broadcast separation, considerations regarding data consistency and bulk data transfers, the use of polling techniques and interrupts, software initialization, and the servicing of synchronous and asynchronous messages.

INTRODUCTION

The ACE RT architecture offers a high degree of programmable flexibility. As a result, it is able to provide a solution to a wide variety of applications. One of the salient attributes, which is the main subject of this application note, is the ACE's memory management architecture. This includes a variety of options, most of which are programmable on a subaddress basis.

The ACE allows the memory management configuration for each T/$\overline{R}$ (/broadcast) subaddress to be selected between a single message buffer or a circular buffer The size of each circular buffer is programmable with choices between 128 and 8192 words, in integral powers of two.

For receive messages, there is a third option, providing a form of double buffering on a subaddress basis. In addition, for each transmit/receive/broadcast subaddress, there is a choice of interrupts: for any message to the T/$\overline{R}$(/brdcst)-subaddress, or following a circular buffer rollover for the T/$\overline{R}$(/brdcst)-subaddress.

MEMORY MAP, DATA STRUCTURES

Table 1 illustrates a typical memory map for the ACE RT with the enhanced mode features enabled. The two Stack Pointers reside in fixed locations in the shared RAM address space: address 0100 (hex) for the Area A Stack Pointer and address 0104 for the Area B Stack Pointer. In addition to the Stack Pointer, for the ACE enhanced RT, there are several other areas of the ACE address space that are designated as fixed locations. These are for the Area A and Area B Lookup Tables, the Mode Code Selective Interrupt Table, the Mode Code Data Table, the Busy Lookup Table, and the Command Illegalizing Table.

The ACE provides a global area A/B select mechanism, programmable by means of bit 13 of Configuration Register #1. This allows the host CPU to be able to toggle between two alternative sets of corresponding data structures. These data structures include the Stack Pointer and Lookup Table. Each Stack Pointer references an individual stack. Each lookup table references a set of data tables for the individual RT subaddresses. A provision of MIL-STD-1553B Notice 2 (30.6) requires that "...RT shall be capable of distinguishing between a broadcast and a non-broadcast message to the same subaddress for non-mode command messages." The ACE exceeds the letter of this requirement by including a programmable option for separate lookup tables for non-broadcast and broadcast receive messages. Use of separate lookup tables allows data words received from

TABLE 1. TYPICAL RT MEMORY MAP	
(SHOWN FOR 4K RAM WITH ENHANCED MODE FEATURES ENABLED)	
ADDRESS (HEX)	**DESCRIPTION**
0000-00FF	Stack A
0100	RT Command Stack Pointer A (fixed location)
0101-0103	RESERVED
0104	RT Command Stack Pointer B (fixed location)
0105-0107	RESERVED
0108-010F	Mode Code Selective Interrupt (fixed area)
0110-013F	Mode Code Data Table (fixed area)
0140-01BF	Lookup Table A (fixed area)
01C0-023F	Lookup Table B (fixed area)
0240-0247	Busy Bit Location Table (fixed area)
0248-025F	(not used)
0260-027F	Data Block 0
0280-029F	Data Block 1
• • •	• • •
02E0-02FF	Data Block 4
0300-03FF	Command Illegalizing Table (fixed area)
0400-041F	Data Block 5
0420-043F	Data Block 6
• • •	• • •
0FEO-0FFF	Data Block 100

broadcast messages to be stored in separate blocks from non-broadcast data in the ACE address space. BROADCAST SEPARATION may be enabled by programming bit 0 of Configuration Register #2 to logic "1".

The RT lookup tables provide a mechanism for mapping data blocks for individual Tx/Rx/Bcst-subaddresses to areas in the shared RAM. The two lookup tables are located in address range 0140 to 01BF for Area A and address range 01C0 to 023F for Area B. The RT lookup tables include Subaddress Control Words as well as the individual data block pointers. The actual Stack RAM area as well as the individual data blocks may be located in any of the non-fixed areas in the shared RAM address space.

Table 2 illustrates the organization of the RT Lookup Tables. It should be noted that if SEPARATE BROADCAST DATA, bit 2 of Configuration Register #2, is logic "0", the data block pointers for both broadcast and nonbroadcast receive messages will be stored in the first 32 locations of the lookup tables. If SEPARATE BROADCAST DATA is logic "I", the pointers to data words for broadcast messages are stored in the third block of 32 locations within the respective lookup table.

It should be noted that the Subaddress Control Words, located in the fourth block of 32 locations within each lookup table, are only used if ENHANCED RT MEMORY MANAGEMENT, bit 1 of Configuration Register #2, is logic "1".

SUBADDRESS MEMORY MANAGEMENT

The most salient attribute of the ACE's RT architecture is the flexibility offered by its options for subaddress memory management. The ACE supplies a host of programmable features to support ease of operation, guarantee data consistency, and facilitate bulk data transfers.

The ability to program the memory management mechanism on a subaddress basis is essential to most complex system designs since the data content that will be transferred to and from an RT generally varies in size (bulk data versus single messages). and CPU servicing requirements (received data is retrieved after immediately after every message versus asynchronous use of data). The key to the ACE's RT memory management architecture is that it allows the selection of single message. double buffering, and circular buffer (including circular buffer size) to be programmed on a transmit/ receive/broadcast-subaddress basis.

In its default (power turn-on) configuration, the ACE initializes to its non-enhanced mode. In this configuration, the data tables for broadcast and non-broadcast receive messages are referenced by the same set of lookup table pointers and the subaddress memory

TABLE 2. RT LOOKUP TABLE			
AREA A	**AREA B**	**DESCRIPTION**	**COMMENT**
0140	01C0	RX (Bcst) SA0	Receive (/Broadcast) Lookup Table
•	•	•	
•	•	•	
•	•	•	
015F	01DF	RX (Bcst) SA31	
0160	0150	TX SA0	Transmit Lookup Table
•	•	•	
•	•	•	
•	•	•	
017F	01FF	TX SA31	
0180	0200	Bcst SA0	Broadcast Lookup Table (Optional)
•	•	•	
•	•	•	
•	•	•	
019F	021F	Bcst SA31	
01A0	0220	SACWSA0	Subaddress Control Word Lookup Table (Optional)
•	•	•	
•	•	•	
•	•	•	
01BF	023F	SACW SA31	

management scheme defaults to the single message mode for all receive and transmit subaddresses. In order to use the various enhanced features. various configuration register bits need to be set. Reference Table 3. It should be noted that for most of the advanced features (broadcast separation is the exception), combinations of bits in the Subaddress Control Word must also be used.

SUBADDRESS CONTROL WORD

If ENHANCED RT MEMORY management has been enabled by means of bit I of Configuration Register #2, each of the 32 Subaddress Control Words specifies the memory management and interrupt schemes for the respective subaddress. Refer to Tables 4 and 5.

For each Subaddress Control Word, five bits specify the memory management scheme and interrupts for each of transmit, receive, and broadcast messages. In addition, bit 15 (MSB) of each Subaddress Control Word may be used to enable double buffering for individual receive (and broadcast receive) messages to the particular subaddress. Bit 15 of Configuration Register #3, followed by bit 12 of Configuration Register #2 **(in that order)** must be set to logic "1" in order to enable use of the subaddress double buffering mode.

For each transmit, receive, or broadcast subaddress, three bits are used to specify the memory management scheme. For each Tx/Rx/Bcst subaddress, the memory management scheme may be

selected for either the "single message" mode (as in the nonenhanced mode), or the "circular buffer" mode. In addition, each receive (and broadcast, if separated) subaddress may be configured for the subaddress double buffering mode if the RX:(BCST:) MM2, RX:(BCST:) MMI, and RX:(BCST:) MM0 bits (7, 6, and 5 for receive; bits 2, 1, and 0 for broadcast if separated) are programmed to logic "0" **and** Subaddress Double Buffering (bit 15) is logic "1".

In the single buffer mode, a single data block is repeatedly overread (for transmit data) or overwritten (for receive or broadcast data). Alternatively, in the circular buffer mode, Data Words for successive messages to/from any particular Tx/Rx/Bcst subaddresses are read from or written to the next contiguous block of locations in the respective circular buffer.

For the circular buffer mode, the size of the circular buffer for each transmit, receive, or broadcast subaddress may be programmed for 128, 256, 512, 1024, 2048, 4096, or 8192 words.

For each Tx/Rx/Bcst subaddress, two bits of the subaddress control word are used to enable interrupts. One of these bits will result in an interrupt following every message directed to the specific Tx/Rx/Bcst subaddress. The second interrupt control bit will result in an interrupt at the end of a message if the message resulted in the lookup table pointer for the respective Tx/Rx/Bcst-subaddress crossing the lower boundary of

TABLE 3. CONFIGURATION REGISTER BITS USED TO ENABLE ADVANCED RT MEMORY MANAGEMENT FEATURES					
MEMORY MANAGEMENT FEATURE	C.R. 2, BIT 12 SUBADDRESS DOUBLE BUFFER ENABLE	C.R. 2, BIT 11 OVERWRITE INVALID DATA	C.R. 2, BIT 1 ENHANCED RT MEMORY MANAGEMENT	C.R. 2, BIT 0 SEPARATE BROADCAST	C.R. 2, BIT15 ENHANCED MODE ENABLED
Default (power turn-on) Mode: "Single Message" for all tx/rx (/bcst) subaddresses.	0	0	0	0	0
"Single Message" for all transmit subaddresses: "Double Buffering" for all receive (and bdcast) subaddresses.	1 (See Note 2)	0	0	X	1 (See Note 2)
Broadcast separation	X	X	X	X	X
Enable Circular Buffer for individual subaddresses; NOT overwriting invalid data	X	0	1	X	X
Enable Circular Buffer for individual subaddresses; With overwriting invalid data	X	1	1	X	X
Enable subaddress double buffering for individual subaddresses	1 (See Note 2)	X	1	X	1 (See Note 2)

NOTES: (1) "C.R." denotes Configuration Register.
(2) In order to enable the subaddress double buffering modem bit 15 of Configuration Register 3 must be set to logic "1" before bit 12 of Configuration Register 2 is written as logic "1."

TABLE 4. SUBADDRESS CONTROL WORD BIT MAP	
BIT	**DESCRIPTION**
15 (MSB)	RCV DBL BUFF'NG ENA
14	TX: INT ON EOM
13	TX: INT BUF ROLLOVER
12	TX; MM2
11	TX: MM1
10	TX: MM0
9	RX: INT ON EOM
8	RX: INT BUF ROLLOVER
7	RX: MM2
6	RX: MM1
5	RX: MM0
4	B'CAST: INT ON EOM
3	B'CAST: INT BUF ROLLOVER
2	B'CAST: MM2
1	B'CAST: MM1
0 (LSB)	B'CAST: MM0

the circular buffer, rolling over to the top of the buffer.

SINGLE MESSAGE MODE

As discussed, the default (power turn-on) configuration for the ACE results in the single message mode for all receive and transmit subaddresses. In addition, individual transmit, receive, and broadcast subaddresses may be designated for the single message mode by means of the Subaddress Control Word.

The operation of the single message RT mode is illustrated in Figure 1. In the single message mode, the respective lookup table entry is loaded by the host

processor. At the start of each message, the lookup table entry is stored in the third address location of the respective message block descriptor in the stack area of RAM. Received Data Words are written to or transmitted Data Words are read from the Data Word block referenced by the lookup table pointer. In the single message mode, the current lookup table pointer is not written by the ACE memory management logic at the end of a message. Therefore, if a subsequent message is received for the same subaddress, the same block of Data Words will be overwritten or overread.

The chief advantage of the single message mode is its simplicity. For a given receive subaddress, the CPU does not need to reference the lookup table pointer prior to accessing received Data Words. In comparison to other techniques, the single message buffer uses an absolute minimum amount of memory space. This reduces cost and allows relatively more RAM area to be used for subaddresses where bulk data transfers are being performed (see section on circular buffers).

For applications where the RT synchronously receives a periodic message to a particular subaddress, either the End-of-Message (EOM) or Subaddress Control Word (RX: [BCST:] INT ON EOM) interrupt should be used if feasible. This allows the CPU to road the new data block with minimal software overhead. However, for asynchronous applications, it may be necessary to utilize the subaddress double buffering technique for received messages.

For transmitting of synchronous single-message data blocks, a software double buffering technique may be employed. That is, the host may alternate ("ping-pong") between two buffers of Data Words to be transmitted. The critical step is for the CPU to re-assign the lookup

TABLE 5. SUBADDRESS CONTROL WORD Management Subaddress Buffer Scheme				
MM2	**MM1**	**MM0**	**RCV PBL BUF ENL**	**DESCRIPTION**
0	0	0	0	Single Message	
0	0	0	1	Double Buffered: for receive (or broadcast) subaddress; if single message for transmit subaddress.	
0	0	1	X	128 Word	
0	1	0	X	256 Word	
0	1	1	X	512 Word	Circular Buffer
1	0	0	X	1024 Word	of
1	0	1	X	2048 Word	Specified Size
1	1	0	X	4096 Word	
1	1	1	X	8192 Word	

CONFIGURATION REGISTER

15   13   0

CURRENT AREA B/A

STACK POINTERS

DESCRIPTOR STACKS

BLOCK STATUS WORD

TIME TAG WORD

DATA BLOCK POINTER

RECEIVED COMMAND WORD

LOOK-UP TABLE ADDR
(See note)

LOOK-UP TABLE
(DATA BLOCK ADDR)

DATA BLOCKS

DATA BLOCK

DATA BLOCK

Note: Lookup table is not used for mode commands when enchanced mode codes are enabled.
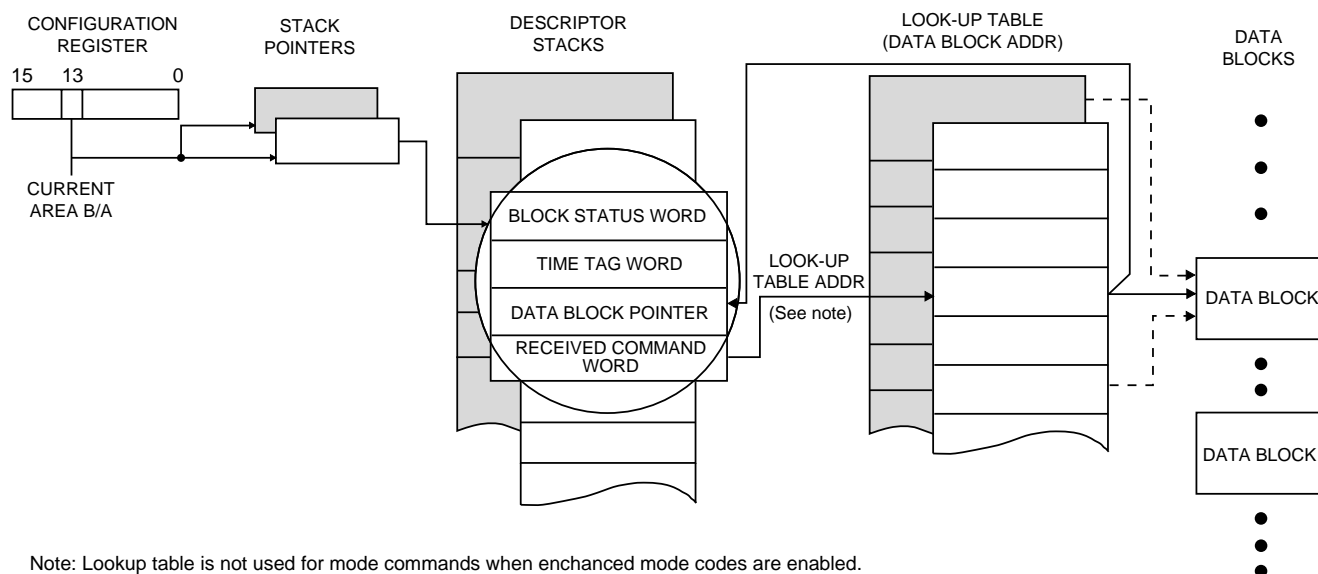
FIGURE 1. SINGLE MESSAGE MODE

table pointer **after it** has written a new block of Data Words to be transmitted. This ensures data consistency by eliminating the possibility of transmitting a mixture of old and new Data Words. It should be noted that the ACE's protocol logic samples (reads) the Lookup Table pointer **only at the start of a message sequence.** That is, after the Command Word is received. As a result, it will not switch blocks during the processing of a message.

Another application of the single message technique involves the data wraparound subaddress. Paragraph 30.7 of MIL-STD-1553B Notice 2 requires that an RT provide such a mechanism to allow for an "end-to-end" self-test by a bus controller. It is suggested that subaddress 30 (11110) be used for the wraparound function. The wraparound subaddress involves receiving a message of "one to N words of any bit pattern" to the wraparound subaddress. In order to comply to Notice 2, the RT must be able to transmit, from the wraparound subaddress, the re received word pattern back to the bus controller. In order to ensure that the same block of words are transmitted as well as received, the single message mode must be used for the data wraparound subaddress.

DESCRIPTOR BLOCK

As shown in Figure 1, the ACE RT stores a 4-word block descriptor in the active area stack for each message processed. The Block Status Word contains information of whether a message is ongoing or has been completed, what bus channel it was processed on, an indication of circular buffer rollover, and a mul-

titude of information identifying message error conditions. These include loop test (self-test) failure, illegal Command Words, errors in received Data Words, RT-to-RT transfer errors and other fault conditions. The Block Status Word may be used as a diagnostic tool in troubleshooting hardware and software system problems.

Following completion of a message, the Time Tag word reflects the contents of the ACE's 16-bit free-running Time Tag Register at the end of the message. One use of the Time Tag word is for the host to be able to determine how long a received message has been stored in a receive buffer; i.e., to determine data staleness or interrupt latency.

In the single message mode, the Data Block Lookup Pointer provides a convenience for the CPU to locate Data Word blocks. In the circular buffer and double buffered modes, its use is necessitated by the fact that the subaddress lookup table pointers are generally updated after every message. The received Command Word is stored in the fourth location of the descriptor for each message. In addition to providing an identifier for each message, this allows the CPU to be able to easily scan through the stack to locate particular received Command Words.

CIRCULAR BUFFER MODE

The operation of the ACE circular buffer RT memory management mode is illustrated in Figure 2. As in the single message mode, the individual lookup table entries are initially loaded by the host processor. At the start of each message, the lookup table entry is

stored in the third position of the respective message block descriptor in the stack area of RAM. Receive or transmit Data Words are transferred to (from) the circular buffer, starting at the location referenced by the Lookup Table pointer.

The size of the circular buffer for each transmit/ receive/broadcast subaddress is specified by means of the respective Subaddress Control Word. It is selectable from among 128, 256, 512, 1024, 2048, 4096, or 8192 words.

If the OVERWRITE INVALID DATA bit (bit 11) of Configuration Register #2 is logic "0", the location after the last word accessed for the message is stored into the respective Lookup Table location, **regardless of whether or not there were any errors in the just completed message.** By so doing, data for the next message for the respective Tx/Rx/Bcst subaddress will be accessed to/from the next lower contiguous block of locations in the circular buffer.

If the OVERWRITE INVALID DATA bit (bit 11) of Configuration Register #2 is logic "l", the location after the last word accessed for the message is written to the respective Lookup Table location **only following a valid received (or transmitted) message.** Assuming that the value of the Lookup Table pointer is updated, data for the next message for the respective Tx/Rx/Bcst subaddress will be accessed to/from the next lower contiguous block of locations in the circular buffer. Assuming that the OVERWRITE INVALID DATA bit is set, the Lookup Table pointer will not be updated at the end of the message, if there was an error in the message. This allows failed messages in a

bulk data transfer to be retried without corrupting the circular buffer data structure, and without intervention by the RT's host processor.

It is strongly recommended that OVERWRITE INVALID DATA be set to logic "1" when performing bulk data transfers.

When the pointer reaches the lower boundary of the circular buffer, the pointer moves to the top boundary of the circular buffer, as shown in Figure 3. It is important to note that the boundaries of the circular buffers are determined by 128-word, 256-word, etc. boundaries in the ACE address space. The boundaries are not dependent on the starting address of the buffer.

For example, if a circular buffer size is programmed for 256 and the lookup table pointer is initialized to 400, the bottom boundary of the circular buffer is established as 511, not 655 (400 + 255). That is, the buffer address rolls over from 511 to 256.

As in the single message mode, the pointer to the start of the data block is stored in the third location of the message block descriptor (in the stack). This provides a convenience for the CPU to locate Data Words received from individual messages.

Circular Buffer Interrupt.

An important feature of the ACE's circular buffer capability is the circular buffer interrupt. This interrupt may be enabled by means of bits 13,8, and 3 of the Subaddress Control Word for transmit, receive, and broadcast messages, respectively, The use of this interrupt serves to offload the host processor when
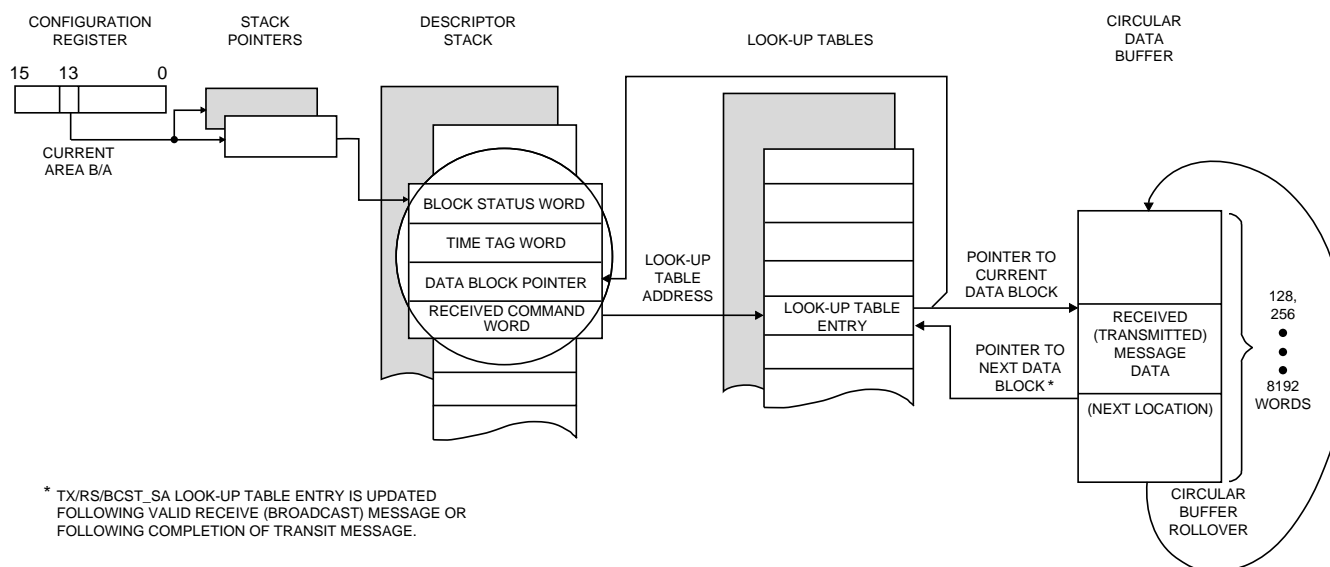
FIGURE 2. RT MEMORY MANAGEMENT – CIRCULAR BUFFER MODE

performing bulk data transfers. The interrupt will be issued after the circular buffer pointer has rolled over from the bottom to the top of the circular buffer address space. It should be noted that the operation of this interrupt request is effected by the OVER-WRITE INVALID DATA bit. If OVERWRITE INVALID DATA is logic "0", the interrupt request will occur **immediately** after the word at the bottom boundary of the circular buffer has been accessed. If OVER-WRITE INVALID DATA is logic "1", the interrupt request will occur **following the end of a valid (but not Invalid) message** in which the word at the bottom boundary of the circular buffer has been accessed.

Circular Buffer-Bulk Data Transfers.

The primary use of the circular buffer feature is for performing bulk data transfers, such as downloads of program data. Consider the example where the RT anticipates receiving 1000 Data Words. The CPU initializes for such a download by performing the following sequence:

(1) Write to Configuration Register #2, setting the ENHANCED RT MEMORY MANAGEMENT (bit 2) and OVERWRITE INVALID DATA (bit 11) bits to logic "1".

(2) The host should initialize the receive (or possibly, broadcast) bits of the respective Subaddress Control Word for a circular buffer of size 1024 words and enable the RX: (or BCST:) CIRCULAR BUFFER ROLLOVER interrupt request.

(3) The CPU should initialize the lookup table pointer exactly 24 words down from the top of the buffer.

(4) The host should then wait for a CIRCULAR BUFFER interrupt request. This may be verified by means of the Interrupt Status Register. By so doing, the host is able to determined that it has received exactly 1000 valid Data Words that may now be accessed.

The use of the circular buffer feature offloads the host CPU in a number of ways. These include:

(1) Aside from configuration registers, the only initialization required by the CPU is to write the Subaddress Control and Lookup Table pointer words.

(2) All "real time" pointer management is performed by the ACE RT, rather than by the host.

(3) There is no need to service interrupts, or poll for, individual received messages.

(4) There is no need for the host processor to be concerned about errors or message retries. If there is an error in a Received message (eg., a parity error), the ACE RT will not respond and the BC will retry the message. Since the ACE RT's OVERWRITE INVALID DATA bit has been programmed to logic "1", the value of the lookup table pointer will not be updated as a result of the failed message. When the BC retries the message, the valid received Data Words will overwrite the invalid received message.

(5) After initialization, the host need only wait for the Circular Buffer Rollover interrupt request. Following occurrence of the interrupt request, the host need only read the received multi-message data buffer. The operation of the circular buffer scheme enabled the 1000 valid words to be received with no host in intervention.

SUBADDRESS DOUBLE BUFFERING

The purpose of this mode is to provide the host processor with the highest possible degree of data consistency for received Data Words. As illustrated in Figure 3, this is accomplished by allocating two 32-bit Data Word blocks for each individual receive (and/or broadcast receive) subaddress. At any point in time, one of the blocks is designated as the active 1553 block while the other block is considered inactive.

The Data Words for the next receive message to that subaddress will be stored in the active block. Upon completion of the message, provided that the message was valid and Receive Double Buffering is enabled, the ACE will switch ("ping-pong") the active and inactive blocks automatically for that subaddress by writing the starting address of the inactive block to the respective lookup table address. This means that the latest, valid, complete received data block is always readily available to the host processor.

As illustrated in Table 3, subaddress double buffering may be invoked globally, for all subaddresses, or may be utilized on an individual subaddress basis. In the latter case, double buffering for any particular receive (and/or broadcast) subaddress is programmable via the Subaddress Control Word. To enable double buffering for receive messages, RX: MM2-MM0 (bits 7-5) must set to logic "0") and the DOUBLE BUFFER ENABLE bit (bit 15) must be set to logic "1". If SEPARATE BROADCAST is enabled (bit 0 in Configuration Register #2 is set to logic "1"), then BCST: MM2-MM0 (bits 2-0) should be set to logic "0" and DOUBLE

BUFFER ENABLE bit (bit 15) must be set to logic "1" to enable double buffering for a broadcast subaddress.

To make best use of the subaddress double buffering mode, it is best to invoke the ENHANCED RT MEMORY MANAGEMENT mode by setting bit 1 of Configuration Register #2 set to logic "1". This allows the double buffering for individual subaddresses to be disabled while reading the data.

The basic software algorithm for reading the latest, most consistent block of valid data words received to a given subaddress is outlined as follows:

(1) Disable double buffering for the desired subaddress by clearing RECEIVE DOUBLE BUFFER ENABLE, bit 15 of the Subaddress Control Word to logic "0".

(2) The host processor should then read the current value of the data block pointer in the lookup table. This pointer references the active block. The latest consistent data block can be referenced in the inactive block by inverting bit 5 of the current data block pointer. The current lookup table entry should not be modified.

(3) Read the Data Words from the inactive block. By definition, these represent the latest, most consistent sample of valid Data Words received to the particular subaddress (at the time of step 1).

(4) Re-enable double buffering for the accessed subaddress by seeing the RX DOUBLE BUFFER ENABLE bit (bit 15) in the appropriate Subaddress Control Word to logic "1".

Many system Interface Control Drawings (ICDs) require that a method be provided to ensure sample data consistency. The principal advantage of the subaddress double buffering mode is that it ensures data consistency. That is, it guarantees that the host processor will never read a mixture of data words from the previous sample and the most recent sample.
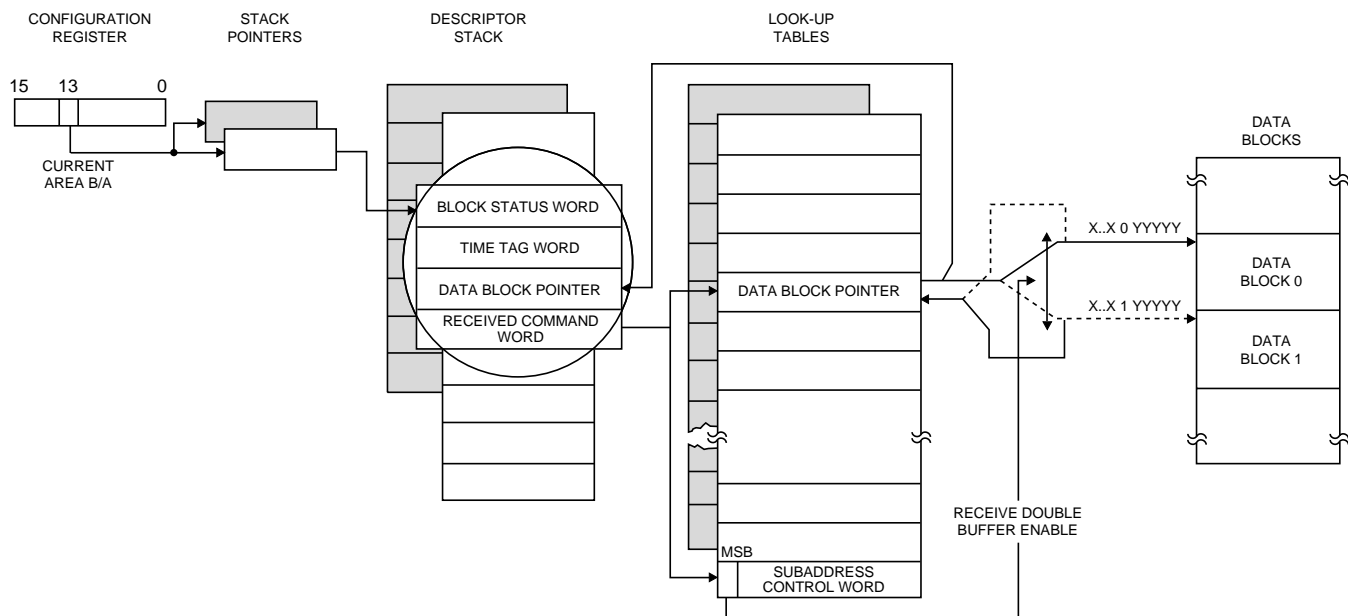
The use of the double buffering capability, particularly the capability to disable double buffering on a subaddress basis, is essential for the case in which the processor is asynchronously reading data from a particular RT subaddress. There are many applications in which the RT is receiving data from the BC to a particular subaddress, but the RT's host processor does not immediately need to make use of the information. The ACE's RT subaddress double buffering allows the host processor to ignore received messages until the information is actually needed and still guarantee data consistency with minimum processor overhead. This allows the host may to read the latest, complete, valid data block from a subaddress at any time.

RT SOFTWARE INITIALIZATION PROCEDURE

The following software sequence is typical of the steps the host CPU should perform following power turn-on to configure the ACE for RT operation. For most applications, it is possible to skip many of the steps indicated.

(1) Perform a software reset, by writing 0001 (hex) to the Start/Reset Register.

(2) If any of the Enhanced mode functions (eg., subaddress double buffering) are to be used, invoke the ACE RT's enhanced mode by writing

FIGURE 3. SUBADDRESS DOUBLE BUFFERING MODE

8000 (hex) to Configuration Register #3.

(3) Initialize the Interrupt Mask register. For many RT applications, the EOM interrupt will generally be enabled. In other instances, the RT Subaddress Control Word, RT Circular Buffer Rollover, and/or Format Error Interrupt Requests may also be enabled. The RT Subaddress Control Word Interrupt enables interrupt requests to be issued following messages to specified Transmit, Receive, or Broadcast subaddresses. The RT Circular Buffer Rollover Interrupt may be used to provide an interrupt request following a multi-message reception or transmission of a specified number of Data Words to/from a given subaddress.

(4) Load the starting location of the Stack into the Active Area Stack Pointer location in RAM.

(5) As an option, initialize the Active Area Stack. If there is a desire to poll the Stack RAM while 1553 messages are being processed, the Block Status Word locations for the respective message block descriptors (relative address locations 0, 4, 8...[stack size 4] in the stack) should be cleared to 0000. A Block Status Word of 0000 (SOM = EOM = 0) indicates that a message has not been processed yet.

(6) Initialize the Active Area Lookup Table. The Lookup Table address for each transmit, receive. and (optionally) broadcast subaddress should be initialized as the pointer value for each respective lookup table. If the RT is going to be used in the enhanced RT Memory Management mode, it will also be necessary to select the memory management and interrupt options for each subaddress by initializing the Subaddress Control Words for the Active Area.

If there are several unused subaddresses for an RT, it is recommended that the Lookup Table pointers for these be initialized to the same value in order to conserve memory space.

(7) If ENHANCED MODE CODE HANDLING (bit 0 of Configuration Register #3) is not used, the pointers for receive subaddresses 0 and 31 (for Synchronize with Data messages) generally get loaded with the same pointer value. Similarly, the Lookup Table addresses for transmit subaddresses 0 and 31 (for Transmit Vector Word messages) generally get loaded with the same pointer value.

If ENHANCED MODE CODE HANDLING is enabled. Data Words for these mode codes are stored in locations 0111 (for Synchronize with data) and 0130 (for Transmit Vector Word).

(8) Initialize Configuration Register #2. This involves selecting use of the following functions: ENHANCED RT MEMORY MANAGEMENT should be selected if it is desirable to select the circular buffer feature and/or subaddress double buffering features on an individual subaddress basis. SEPARATE BROADCAST should be set if it is necessary to comply to Notice 2 for broadcast messages.

ENHANCED INTERRUPTS should be enabled if the CPU needs to poll using the Interrupt Status Register and/or it is desired that one or more of the following conditions cause an interrupt: Transmitter Timeout, RT Command Stack Rollover, or RT mode code interrupt. OVERWRITE INVALID DATA should be set to logic "1" if the circular buffer mode is used with one or more subaddresses. SUBADDRESS DOUBLE BUFFERING should be enabled, if desired. Finally, BUSY LOOKUP TABLE ENABLE should be logic "1" if there is a need for the Busy bit in the RT Status Word to be set for particular T/R/Bcst subaddresses.

(9) Initialize Configuration Register #3. If one or more of the Enhanced Mode features are to be used, **bit 15 must be maintained at logic "1".** The RT Stack size is programmable with choices of 256 words (default, 64 messages), 512, 1024, or 2048 words (512 messages) by bits 14 and 13. Other RT features that may be selected by this register include ILLEGALIZATION (default = "0" = enabled), ALTERNATE STATUS (allowing software programming of **all 11** Status Word bits), the choice of storing or not storing words for illegal or "BUSY" messages, and ENHANCED MODE CODE HANDLING.

If ENHANCED MODE CODE HANDLING is selected, Data Words for mode codes are stored in address locations 0110-013F, and interrupt requests for individual mode codes may be enabled by means of a table in address locations 0108-010F. Other RT options selectable by Configuration Register #3 include 1553A MODE CODES ENABLED, and RTFAIL/RTFLAG WRAP ENABLED. 1553A

MODE CODES ENABLED causes only subaddress 00000 to be treated as a mode code subaddress. RTFAIL/RTFLAG WRAP ENABLE causes the RT FLAG Status bit to be automatically set following a failure of the loop test.

(10) Initialize Configuration Registers #4 and #5. If EXTERNAL BIT WORD ENABLE is logic "1", the Data Word for a Transmit BIT Word mode command is accessed from a RAM location, rather than from an internal register. INHIBIT BIT WORD IF BUSY prevents the BIT Word from being transmitted if the RT is Busy. If MODE CODE

OVERRIDE BUSY is logic '1", this enables the ACE RT to transmit a Data Word in response to a Transmit Vector Word or Reserved transmit mode command, even if the RT is busy.

For the BU-65171 and BU-61581 versions of the ACE, LATCH RT ADDRESS WITH C.R. #5 allows the ACE's RT Address to be software programmable by means of bits 5 through 0 of Configuration Register #5. If LATCH RT ADDRESS WITH C.R. #5 is logic "1", writing to Configuration #5 causes the RT Address to be read from pins RTAD4-0 and RTADP and latched internally. After the RT Address has been programmed, it is suggested that LATCH RT ADDRESS WITH C.R. #5 be cleared to logic "0" to prevent an erroneous overwrite.

(11) If RT Illegalization is used, the CPU should initialize the Illegalization Table, address locations 0300-03FF.

(12) If the BUSY LOOKUP TABLE is enabled, select the desired subaddresses to be Busy by programming the Busy table, address locations 0240 through 0247.

(13) If ENHANCED INTERRUPTS are enabled and ENHANCED MODE CODE HANDLING is enabled, interrupts for selective mode code messages may be enabled by programming locations 0108 through 010F.

(14) Data to be transmitted on the 1553 bus (in response to transmit commands) should be written into the appropriate data blocks. As an option, the locations for Data Words for anticipated receive Data Words may be initialized to zero.

(15) To configure the ACE as an on-line RT, write to Configuration Register # I, setting bit 15 (MSB) to logic "I" and bit 14 to logic "0". The current active

area is selected by the seeing of bit 13 (0 for A, 1 for B). If ALTERNATE RT STATUS is not enabled, bits 11 through 8 should be initialized to select the values for the RT Status Word bits Dynamic Bus Control Acceptance, Busy, Service Request, and Subsystem Flag. Also, in the enhanced mode, the RT FLAG Status Word bit is programmable by bit 7. These bits must be programmed for the logical inverse of their desired values in the RT Status Word.

If ALTERNATE RT STATUS is enabled, bits 10 through 0 of the ACE's RT Status Word are programmable via bits 11 through I of Configuration Register #1. In this case, the logical values (not the inverse) of the intended Status Word bit values must be programmed.

SERVICING COMPLETED RT MESSAGES

The ACE RT provides a number of techniques for determining when a message has been processed. These methods support both polling-driven and interrupt-driven software. There are several polling methods that may be used. These include:

(1) In the ENHANCED mode, the host may continuously poll RT MESSAGE IN PROGRESS, bit 0 of Configuration Register #I. This bit will return logic "0" while the ACE RT is not processing a message. During the time that the ACE RT is servicing a message (after the receipt of a Command Word), RT MESSAGE IN PROGRESS will return logic "1". When the message completes, RT MESSAGE IN PROGRESS will once again return logic "0".

(2) The CPU can poll the contents of the Stack Pointer RAM location. The active area Stack Pointer increments by four **at the beginning** of each message being processed (after receipt of a Command Word).

(3) If the host needs to determine the occurrence of a particular Command Word, it may do so by polling the RT Last Command Register. It should be noted that the contents of this register are updated **at the beginning** of a message being processed. The CPU may then poll the EOM (End-of-Message) bit of the Interrupt Status Register to determine when the message has been completed. See explanation below.

(4) If the ACE is programmed for the ENHANCED mode and ENHANCED INTERRUPTS (bit 15 of Configuration Register #2) are enabled, the CPU may poll the Interrupt Status Register. In

this mode, the various bits in the Interrupt Status Register will become set **regardless** of the programming of the corresponding bits in the Interrupt Mask register.

In this mode, the ACE may determine that a message has been **completed** by polling the Interrupt Status Register until the EOM (End-of-Message) bit returns logic " 1".

In this mode, the host processor may also poll to determine when a message has been processed for a particular transmit, receive, or broadcast subaddress. To enable do this, ENHANCED RT MEMORY MANAGEMENT (bit 1 of Configuration Register #2) must be invoked. To cause the bit to be set for a particular Tx /Rx /Bcst-subaddress, it is then necessary to set the appropriate bit ([TX:, RX:. or BCST:1] INT on EOM) bit in the desired Subaddress Control Word to logic "1". All other Subaddress Control Word ([TX:, RX:, or BCST:] INT on EOM) bits should be programmed to logic "0". This will cause the RT SUBADDRESS CONTROL WORD EOM bit in the Interrupt Status Register to be set to logic "1" after completion of the desired message.

Similarly, the host may poll for receipt of a particular mode code message. This feature is enabled by invoking ENHANCED MODE CODE HANDLING (bit 0 of Configuration Register #3). The desired mode code may be selected by setting the appropriate bit in the Mode Code Selective Interrupt Table (address range 0108-010F).). When the specific mode code message has been completed, the RT MODE CODE bit of the Interrupt Status Register will return logic "1".

Similarly, the Interrupt Status Register may be polled to determine the occurrence of FORMAT ERROR, CIRCULAR BUFFER ROLLOVER, and/or COMMAND STACK ROLLOVER conditions. FORMAT ERROR indicates any error in a received message, other than an invalid Command Word: sync or Manchester encoding, parity, bit count, word count, or RT-to-RT transfer errors. CIRCULAR BUFFER ROLLOVER may be used to signal completion of a multi-message bulk data transfer, as described above. COMMAND STACK ROLLOVER occurs when the Stack rolls over at an address boundary of 256, 512, 1024, or 2048 words, as programmed in Configuration Register #3.

If interrupts are used, the normal procedure

would be to not invoke ENHANCED INTER-RUPTS. This allows the Interrupt Mask Register to be used to enable bits in the Interrupt Status Register as well as the corresponding interrupt requests for only selected condition(s), as discussed above.

RT ERROR HANDLING

As discussed above, the preferred method for handling erroneous messages is to make use of the circular buffer and/or double buffering techniques. In the case of the circular buffer mode, the OVERWRITE INVALID DATA bit should be set to logic "1". With these techniques, Data Words received and stored from invalid messages will be automatically overwritten by the ACE RT. In most systems, the bus controller will retry failed messages. By so doing, the occurrence of errors and message retries is transparent to the RT's host processor.

If necessary, the RT's host processor may ascertain the occurrence of failed messages by several methods:

(1) Determine, by polling or interrupt techniques, when a FORMAT ERROR condition occurs.

(2) Read the Block Status Words for all messages processed. Bits 12-9 and 6-0 all indicate error conditions in received messages.

SUMMARY OF SUBADDRESS MEMORY MANAGEMENT TECHNIQUES

Table 5 provides a summary of the subaddress memory management techniques that may be used for various types of receive and transmit data transfers. This encompasses applications involving both synchronous and asynchronous systems. In this context, "synchronous" implies that the action of the RT's host processor is tightly coupled to activity from the 1553 bus. In such applications, which are typically interrupt-driven with low interrupt latency, the host CPU will always read a received data table or update a transmitted data table a short period of time after a 1553 message has been processed.

In an "asynchronous" environment, the actions of the host CPU are loosely coupled from the timing of 1553 bus messages. In these situations, data consistency is more of a concern, since it is possible for a 1553 message to be received to a given subaddress during the time that the RT's host processor is accessing the data table for that subaddress.

Referring to Table 5, there are two situations that

TABLE 5. SUMMARY OF SUBADDRESS MEMORY MANAGEMENT METHODS		
TRANSMIT OR RECEIVE	**DESCRIPTION OF DATA TRANSFER**	**RECOMMENDED SUBADDRESS MEMORY MANAGEMENT TECHNIQUE**
Receive	Synchronous. That is, the host processor is reliably able to access received messages prior to the start of the next receive message to the subaddress.	Single message.
Receive, Transmit	Wraparound subaddress.	Single message.
Receive	Bulk (multi-message) data transfer.	Circular buffer.
Receive	Asynchronous, such that the 1553 message rate (messages per second) is much greater than the rate that the RT's host CPU samples received data blocks.	Subaddress double buffering as described above.
Receive	Asynchronous, such that the 1553 message rate (messages per second) is approximately equal to or less than the rate with which the RT's host CPU samples received data blocks.	A modified technique using the subaddress double buffering feature. This is required in order to preclude the possibility that the RT loses a received message from the 1553 bus. Refer to "Asynchronous Received Messages" section below.
Transmit	Synchronous single-message transfer, such that the RT's host CPU is reliably able to rewrite the transmit data table before the beginning of the next transmit message to the subaddress.	Single message.
Transmit	Asynchronous single-message transfer. In this case a transmit message to the subaddress may begin during the time that the RT's host CPU is updating the transmit data table.	Single message. However, a form of software double should be implemented to estimate the possibility of transmitting a combination of old and new Data Words. The CPU should set up two transmit data blocks. The lookup table pointer should be alternated between the starting locations of the two blocks. The pointer should only be updated after the RT's host CPU has finished writing Data Words to the new transmit data table.
Transmit	Bulk (multi-message) data transfer.	Circular buffer. For explanation of error handling, refer to "Bulk Data Transfers" in the section below.

require special attention. The two situations are:

(1) For an RT receiving asynchronous messages such that the 1553 message rate is less than, or approximately equal to the rate that the host CPU is sampling data from the receive Data Word tables.

(2) For an RT transmitting a bulk (multi-message) data transfer.

ASYNCHRONOUS RECEIVED MESSAGES

First, consider the case of an RT receiving asynchronous messages (see Figure 4). If a message transfer to the respective receive subaddress is completed during the time that the host is accessing the previous message, the new message data will be stored in the current "active" data block. However, in order to ensure data consistency, double buffering was temporarily disabled for the subaddress. As a result, the lookup table pointer **will not toggle** to the starting location of the "inactive" block, being accessed by the CPU. When the CPU finishes accessing the "inactive" block, it re-enables the double buffering for the subaddress. Note, at this time, the ACE **will not update the value of the**

lookup table pointer. Consequently, if the host CPU goes to sample the subaddress a second time before a new 1553 message comes in, it will read the same "inactive" data block again. **The data from the most recent received message stored in the "active" data block, will be lost.** If the 1553 message rate is much greater than the CPU's sampling rate, the scenario described above should not present a problem. That is, by the time the CPU goes to read its next sample, the "active" data block will have been overwritten by the next (i.e., frequent) receive message to the subaddress, resulting in a toggling of the pointer. In this case, therefore, data consistency can be ensured by means of the subaddress double buffering mode. That is, the simple procedure outlined above—disabling the double buffering during the CPU "read" time—may be used to eliminate the possibility of reading a mixture of old and new Data Words. Losing the current "active" data block is not a concern.

However, the situation is a bit more complicated for the asynchronous case where the 1553 message rate is less than, or approximately equal to the CPU sampling rate. Reference Figure 5. In this scenario, there is a strong likelihood that a message which completed during the CPU's access time will not be
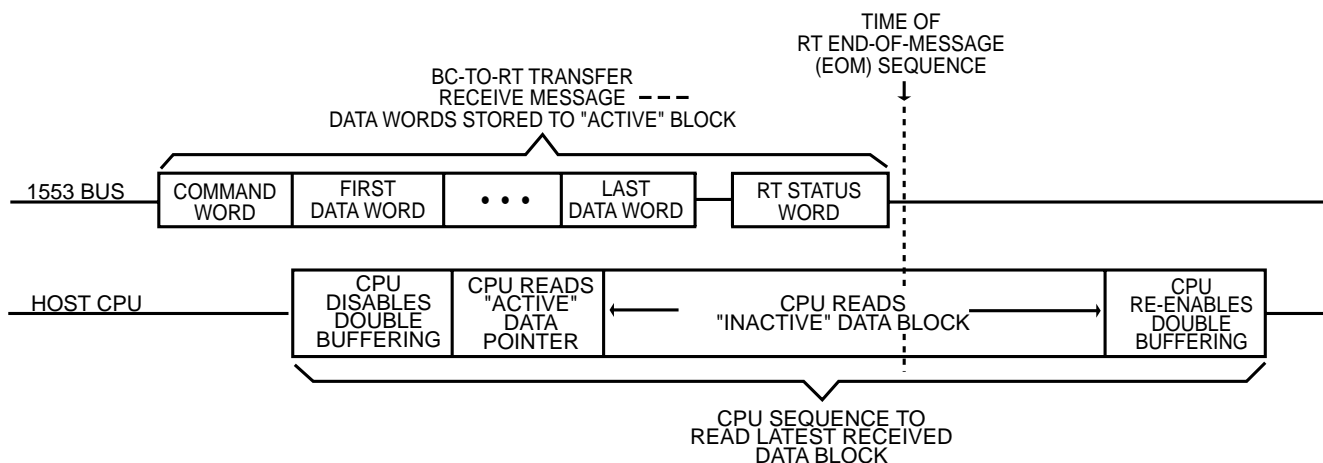
FIGURE 4. ASYNCHRONOUS RECEIVED MESSAGES

overwritten prior to the next sampling interval. In this case, the same data block will be read again. The more recent data, from the new received message, will be lost.

In this instance, a slight modification is needed in order to guarantee that the CPU always reads the most recent received data block. There are several ways of accomplishing this. A fairly simple technique is outlined in the steps below.

Initialization: For the case of a 1553 interface accessed asynchronously by the host, it is assumed that interrupt will not be used. Instead, to enable polling using the Interrupt Status Register ENHANCED INTERRUPTS should be enabled and the Interrupt Mask Register should be initialized to zero. In addition, the RX:, TX:, and BCST: INT ON EOM and INT ON CIRC BUFFER ROLLOVER bits of the 32 Subaddress Control Words should be initialized to logic "0". OVERWRITE INVALID DATA should be set to logic "1".

The sequence for accessing the most recent received data block is summarized as follows:

(1) Write to the Subaddress Control Word for the subaddress to be accessed. Disable double buffering for the desired subaddress by clearing RECEIVE DOUBLE BUFFER ENABLE, bit 15, to logic "0". During the same write cycle, set the RX: (or BCST:) INT ON EOM bit to logic "1".

(2) The host processor should then read the current value of the data block pointer in the lookup table. This pointer references the active block. The CPU should store the value of this pointer for future reference. The most recent data sam-

ple can be referenced in the inactive block by inverting bit 5 of the active data block pointer (ACTIVE PTR x or 32). The value of the lookup table entry should not be modified at this time.

(3) Read the Data Words from the inactive block. By definition, these represent the latest, most consistent sample of valid Data Words received to the particular subaddress (at the time of step [1]).

(4) Re-enable double buffering for the accessed subaddress by setting the RX DOUBLE BUFFER ENABLE bit (bit 15) in the appropriate Subaddress Control Word to logic "1". As part of the same write cycle, clear the RX: (or BCST:) INT ON EOM bit to logic "0".

(5) **IMMEDIATELY** following step (4), read the value of the Interrupt Status Register. If bit 4, RT SUBADDRESS CONTROL WORD EOM, is logic "1", this indicates that the end of a message to the accessed subaddress occurred **during the time that the CPU was reading the data block.** In this case, the CPU should **IMMEDIATELY** write the value of the inactive block as the updated value of the lookup table pointer for the receive (or broadcast) subaddress. This ensures that if another message is not received in the interim, the CPU will next read the latest block of received Data Words, rather than reread the previous sample.

BULK DATA TRANSMIT TRANSFERS

The OVERWRITE INVALID DATA feature provides an airtight mechanism for ensuring data validity for received multi-message data transfers. This is a result of MIL-STD-1553's command/response pro-
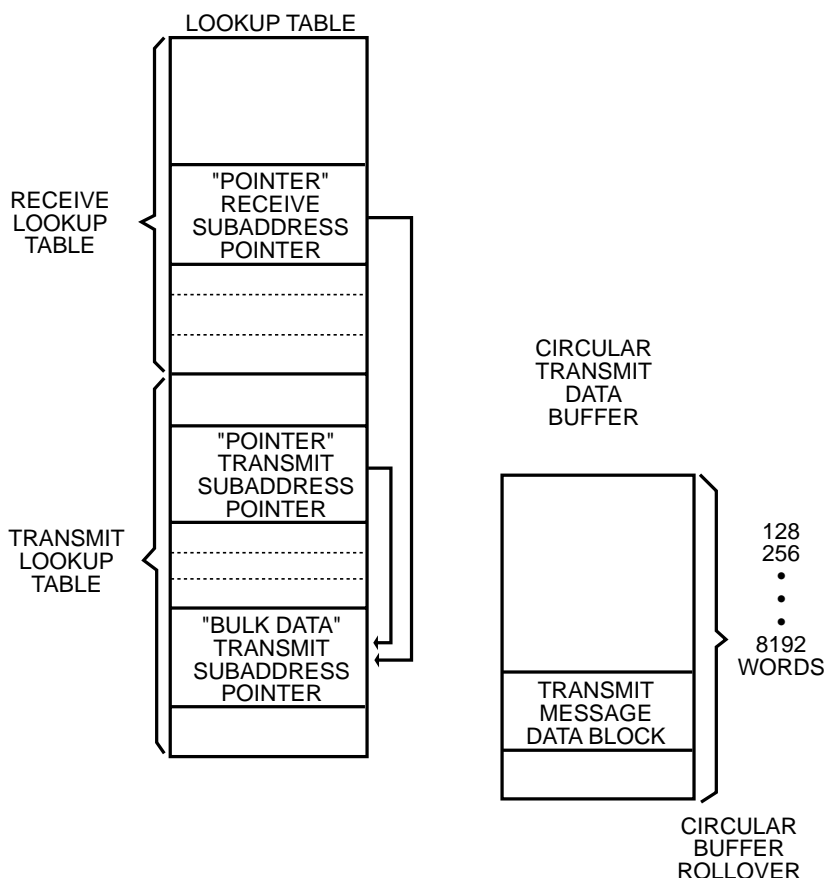
FIGURE 5. BC ACCESS TO AC RT LOOKUP TABLE FOR BULK TRANSMIT TRANSFERS

tocol. That is, a bus controller will be informed if an RT does not receive a completely valid message (the RT will no respond). As a reset, the BC can immediately retry the failed message, resulting in the RT overwriting the invalid data.

The situation is a bit more complicated for bulk data transfers in which the RT is transmitting, rather than receiving. This is due to the fact that MIL-STD-1553, for an RT-to-BC transfer, does not provide a direct mechanism for the bus controller to acknowledge receipt of a valid transmission by an RT. It is the responsibility of the BC to "tell" the RT that the message needs to be retried.

Similar to the problem of asynchronous received messages, there are a number of ways of dealing with the problem of bulk data transfers transmitted by an RT. By necessity, the problem must be managed by the bus controller. Ideally, the method used should be transparent to the RT's host processor.

One relatively simple method makes use of the ACE's circular buffer feature and the fact that the ACE's RT lookup tables are memory mapped. This provides a mechanism for the BC to access the lookup table pointer for the subaddress involved in the bulk data

transfer.

Very simply, the RT software should allocate one subaddress for the BC to access the lookup table pointer. As shown in Figure 5, the receive and transmit lookup table pointers for one subaddress (NOT the subaddress involved in the bulk data transfer) should both point to the location of the pointer for the bulk transfer subaddress. The Subaddress Control Word for this "pointer" subaddress should be initialized for "single message mode", for both transmitting and receiving.

By so doing, the BC is able to retry failed RT-to-BC transfer messages in a way that is transparent to the RT's host CPU.

It should be kept in mind that the BC must distinguish a "no response" from a "format error". A "no response" indicates the RT's non-acknowledgement of the transmit Command Word. In this case, the BC should simply retry the message. A "format error" indicates acknowledgement by the RT. However, it also indicates an error such as sync or Manchester encoding, parity, bit count, or word count in the RT's transmission or the BC's reception of the RT response.

For the case of a "Format Error," the BC's retry sequence is outlined as follows:

(1) After the BC has determined that the RT's response was invalid, it should send a transmit command to the RT's "pointer" subaddress. The RT should respond with a Data Word indicating the current value of the lookup table pointer for the "bulk data" subaddress.

(2) Assuming a message size of 32 Data Words for the "bulk data" subaddress, the BC's host CPU should subtract 32 from the received pointer value. That is,

$$PTR \leftarrow PTR - 32.$$

(3) The BC should send a receive command to the "pointer" subaddress, loading the new value (actually, the old value) of the bulk data pointer. This will cause the RT to retransmit the failed data block.

(4) The BC should retry the failed transmit message for the "bulk data" subaddress.

ILLEGALIZATION AND BUSY

Aside from memory management, another important feature for RT operation is programmable command illegalization. Commands to the ACE RT may be illegalized based on broadcast, T/R bit, subaddress, word count, and mode code. The illegalization scheme is programmable by the host processor in the ACE's RAM, providing a degree of self-testability.

A stipulation of MIL-STD-1553B Notice 2 (30.5.3) states that "...the setting of the busy bit, shall occur only as a result of particular commands/messages sent to the RT." This is subject to interpretation. The ACE RT provides compliance by including an option such that the Busy bit may be programmed to be set as a function of the command broadcast, T/R bit, and subaddress.

For both illegalization and busy, the ACE RT provides two different options for received messages. In the case of illegalization, MIL-STD-1553B (4.4.3.4) states that the RT "...not use the information received." For the case of setting of the Busy bit, the standard (4.3.3.5.3.8) states "...the RT or subsystem is unable to move data to or from the subsystem… ." In either case, there is the question of whether a shared RAM is part of the "RT" or part of the "subsystem". The ACE RT supports both interpretations by allowing for either storage or non-storage of received Data Words for messages in which the RT responded with its Busy and/or Message Error (indicating illegal) bit set. In the ACE's enhanced RT mode, these options are programmable by means of bits 4 and 3 of Configuration Register #3.

SIMPLE INTERFACES TO SIMPLE SYSTEMS

INTRODUCTION

ILC Data Device Corporation recognizes that a typical 1553 bus contains a mixture of terminals with varying degrees of complexity. One of the prominent architectural features of 1553 is that it allows complex computer systems—transferring large amounts of data—to coexist with "simple systems" transferring relatively small amounts of data. A "simple system" is defined as one that does not require a processor, refer to Figure 1. Many applications, such as pressure transducers, gauges, displays, switches, D/A converters, synchro converters, etc., do not require a microprocessor to perform their intended task.

So why add a processor just to satisfy the 1553 interface requirement? DDC offers a full line of "simple system" 1553 components that provide a complete 1553 interface with no processor.

Think about the cost savings by eliminating the processor, both recurring and non-recurring costs. Eliminating the CPU and its support components will reduce the bill of material cost of the system, will reduce the complexity of the board making it easier to build and troubleshoot, and will eliminate the need to develop expensive software and all the documentation that goes along with it (MIL-STD-2167 and ADA).

DDC offers a trio of "simple system" Remote Terminals. The BUS-65153 STIC (Small Terminal Interface Circuit) leads the pack as the smallest, complete low cost 1553 interface. The BUS-65142 is built using a radiation hardened SOS (Silicon-on sapphire)
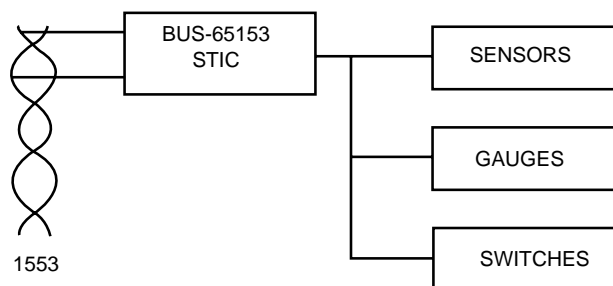
CMOS digital protocol chip and two bipolar analog transceivers making it especially well suited for space applications. The BUS-65149 incorporates a dual sinusoidal transceiver with full multi-protocol remote terminal logic (MIL-STD-1553A and 1553B) meeting the various McAir protocols.

The remainder of this discussion will be centered around the STIC (BUS-65153). Both the STIC and the BUS-65149 are built using the same digital monolithic, therefore, their functional operation is the same. The BUS-65142 is based on a different digital protocol chip but is very similar in operation to the STIC.

DEVICE INITIALIZATION

The STIC provides a system with a truly autonomous Remote Terminal Interface. The STIC contains no internal registers, pointers or Lookup tables that must be initialized. The STIC is ready to begin processing messages immediately following a power-up reset. No action is required by the subsystem to initialize this device. The "stand alone" operation of a 1553 Remote Terminal device is a crucial requirement for systems with no processor.
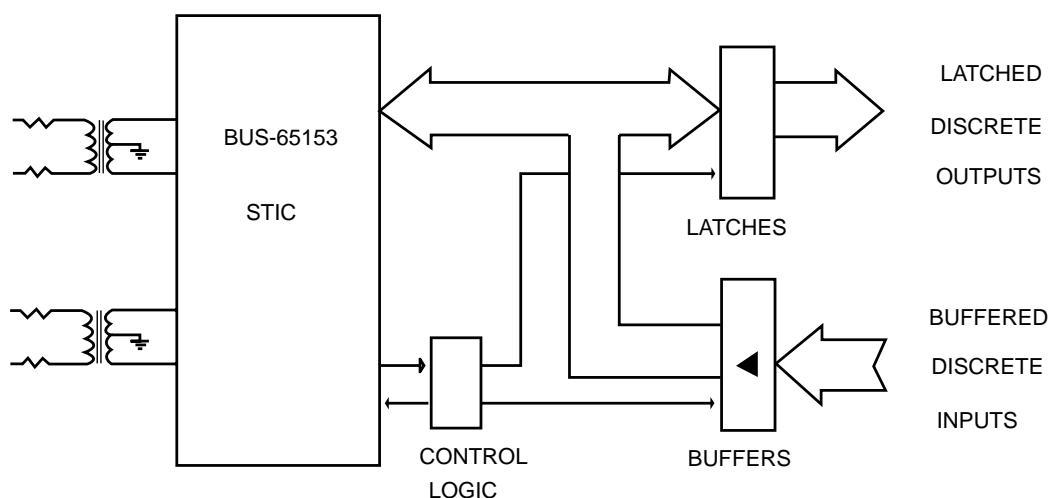
FIGURE 1. ST TO SIMPLE SYSTEM INTERFACE

DMA INTERFACE

All three of the components described make use of a DMA interface with a fixed address map (no lookup tables or other types of indirect addressing}. The STIC will assert the signal $\overline{DBREQ}$ to indicate to the subsystem that a data transfer to or from the subsystem is required. The subsystem must "grant" access to the STIC within the specified time (see data sheet for specific value).

You may wonder why a request/grant handshake mechanism is required for a simple system interface. Consider the case where the STIC is reading a data word directly from the output of an analog to digital (A/D) converter. Many A/D converters require a trigger to begin the conversion and provide a flag that indicates when the conversion is complete and the output data is ready. In this case the $\overline{DBREQ}$ signal may be connected to an active low conversion start signal and the conversion complete output of the A/D may be connected to the $\overline{DTGRT}$ input of the STIC. However, conversion time must be less than the maximum allowable request-to-grant time (refer to STIC data sheet for more details).

Once the grant is received, the STIC will write or read data to or from the subsystem. The first word that is transferred is always the Command Word. The 16 bit value of the Command Word is written to the subsystem. Most simple system applications do not make use of the command word. The command word may be ignored by using address bit A6 ($\overline{COMMAND}$ $\overline{WORD\ TRANSFER}$) as a qualifier in the subsystem's address decoding logic.

SUBADDRESS

Every 1553 command provides a 5-bit address field referred to as a subaddress. The subaddress field is used to map data in the subsystem. The STIC's address outputs A11-A7 provide the subaddress field. There are 30 valid data subaddresses (subaddress 0 and 31 are reserved for mode commands) and each message may contain up to 32 data words. This allows for a linear address space of 960 transmit data words and 960 receive words. The address space may be expanded using logical addressing methods but most simple system applications do not exceed the bounds of the linear address space.

RECEIVE COMMANDS

Upon receipt of a valid receive command to the programmed RT address, the STIC will begin processing the message by transferring the received command word to the subsystem. The Data Words will be written to the subsystem as they are received off the incoming serial 1553 data bus. This creates a gap of approximately 20µs between word transfers.

Ensuring data consistency places an extra burden on a Remote Terminal. Paragraph 4.4.3.6 of MIL-STD-1553B states that if any data word within a message is invalid, the entire message shall be considered invalid. This requires that receive data be "double buffered" until the **entire message** is validated to ensure data consistency. Suppose the STIC was processing a command to receive 30 words and a noise burst on the 1553 bus caused the STIC to miss the last data word. According to MIL-STD-1553B, the 29 other words that were received properly in that message must considered invalid.
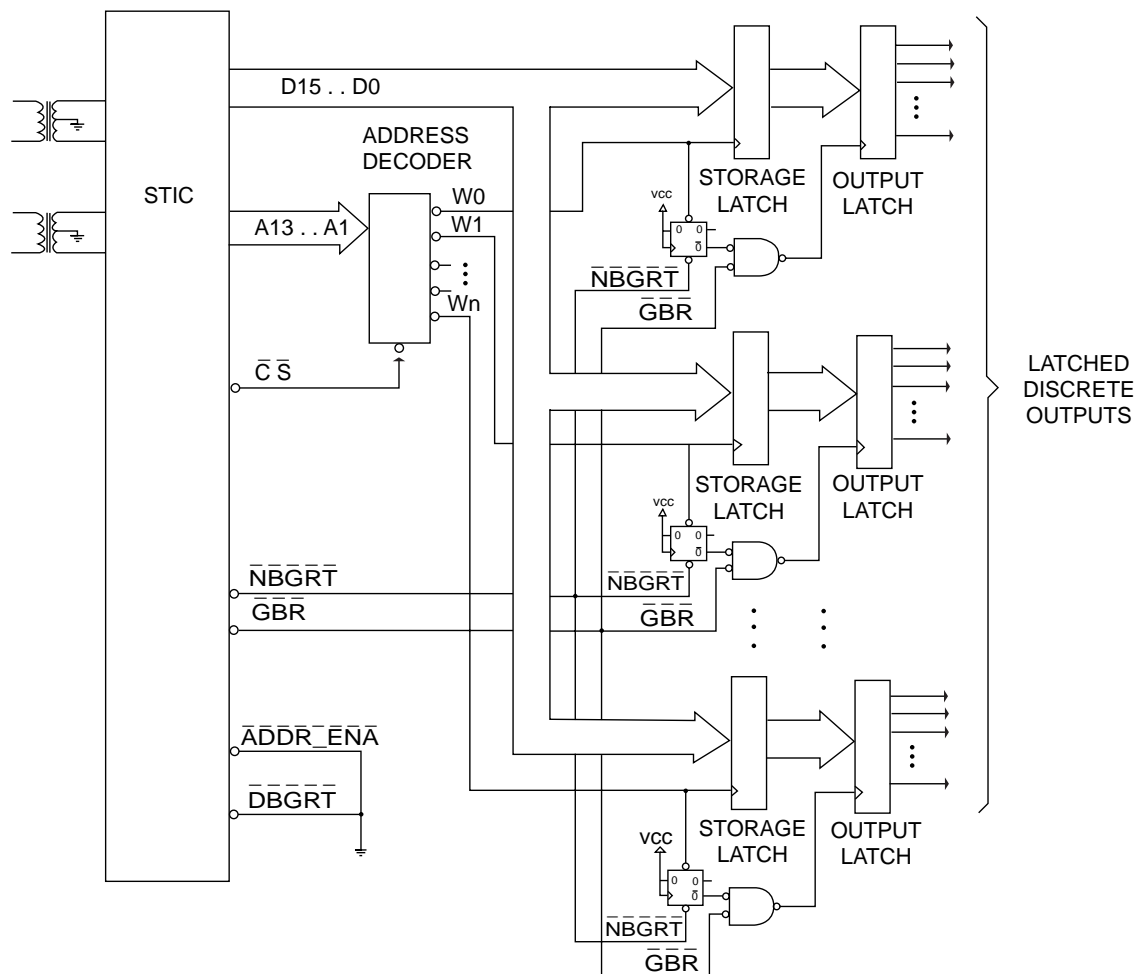
FIGURE 2.

The STIC provides an output signal called Good Block Received ($\overline{GBR}$) to flag to the subsystem that a valid, legal, non-mode receive command with the correct number of data words has been received without a message error and successfully transferred to the subsystem.

Figure 2 illustrates a receive data double buffering mechanism. Each 16-bit received data word is first clocked into a storage latch. Each latch contains a Set/Reset (S/R) latch to mark the fact that new data has been placed into the storage latch. The Good Block Received ($\overline{GBR}$) signal is then used to clock the stored data into the output latches.

At first glance, one might think that the S/R latch and logic gate may be eliminated. Why not use $\overline{GBR}$ to clock all the output latches, not just the ones that were modified? A message prior to the current command may have been a receive message to a different subaddress with one invalid data word. If the Bus Controller did not "retry" the failed message the

"invalid" data would still be stored in the temporary latches. This data **should not** be clocked to the output latches. The S/R latch, therefore serves to maintain data consistency by selectively clocking only the output latches that were written to. Note that the S/R latch is not required if only one receive subaddress is implemented.

Figure 3 illustrates a buffering scheme similar to the one shown in Figure 2, except the S/R latch has been eliminated. An assumption is made that the word count of the command word is the same as the number of latches associated with that subaddress. Based on this assumption the $\overline{GBR}$ pulse is passed to all latches for the current subaddress. To guarantee this assumption a PROM is used to implement command illegalization.

Figure 3 shows two separate address decoders for the purpose of illustration. The first address decoder is used to indicate which word within the current message is being transferred (based on word count
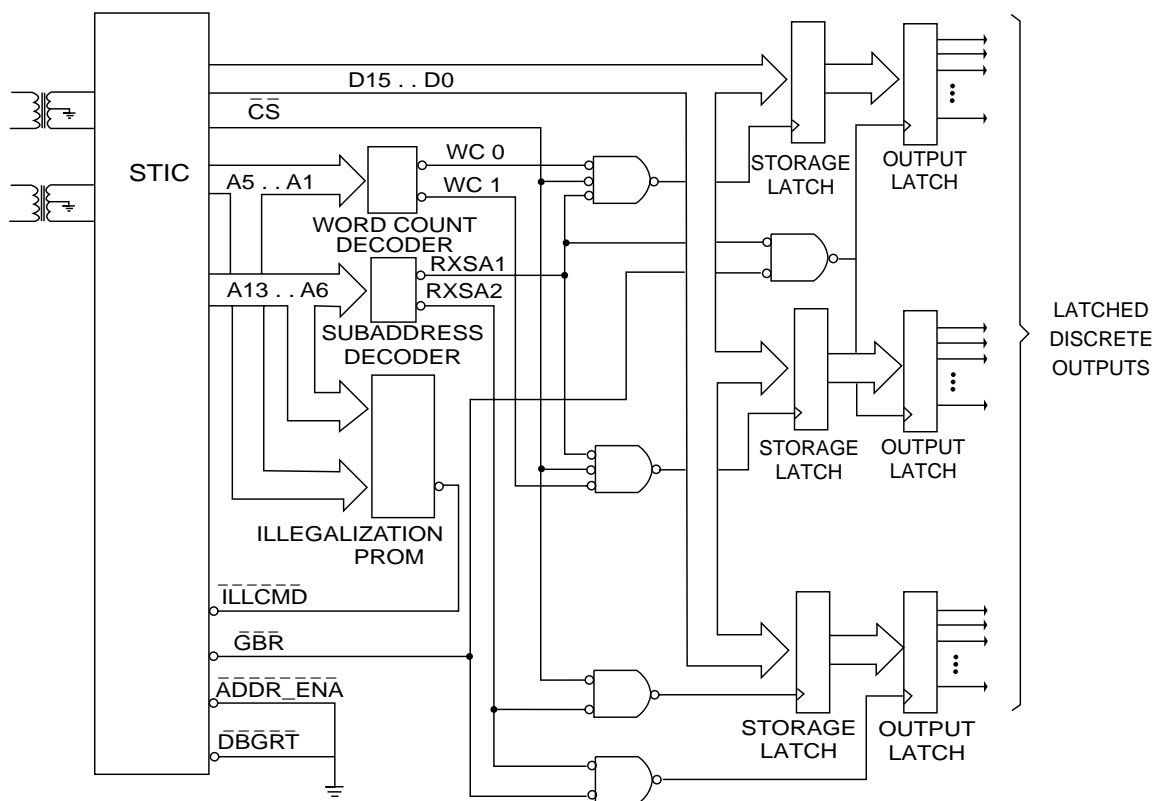
WC 0 = (WORD COUNT = 0)
WC 1 = (WORD COUNT = 1)
RXSA1 = (BROADCAST = 0) • (T /R = 0) • (SUBADDRESS = 1) • (DATA / COMMAND = 1)
RXSA2 = (BROADCAST = 0) • (T /R = 0) • (SUBADDRESS = 2) • (DATA / COMMAND = 1)

FIGURE 3. ADDRESS DECODERS

outputs A5..A I). The second address decoder generates two select outputs: receive subaddress #1 (RXSA1) and receive subaddress #2 (RXSA2). These select outputs will remain valid while the $\overline{GBR}$ output pulse is asserted. This allows the $\overline{GBR}$ pulse to clock the output latches for the current subaddress.

Note that all or most of the interface logic shown may be combined into a single programmable logic device (PLD).

Also note that Figures 2 or 3 do not use the STIC's write ($\overline{WRT}$) output signal. This is because the direction of the due word transfers is known from address bit 12 (Transmit/Receive) and is included as a term in the address decoding equations Transmit messages will always read due from the subsystem and receive messages will always write data to the subsystem.

Figure 4 illustrates an interface between the STIC and two DSC-11524s. A DSC-11524 is a 16-Bit digital to

synchro converter (DSC) from DDC. The concepts used in this example apply to any receive interface circuit.

The DSC-11524 input has 16-bit double buffered transparent latches built in. The first latch will be used to hold the 16-bit data value until the receive message is validated.

The address decoder used in this example generates three outputs: subaddress #1 select (SEL1), write pulse #O (WRO), and write pulse #1 (WR1). The circuit is designed to receive two data words to non-broadcast subaddress 1.

Upon detecting a valid receive command, the STIC will begin processing the incoming message. The address lines A13..A7 will be updated to reflect the current command (based on broadcast/non-broadcast,transmit/receive, and subaddress). Address lines A6..A1 will clear to logic 0 for the first subsystem transfer (the 16-bit command word). This example
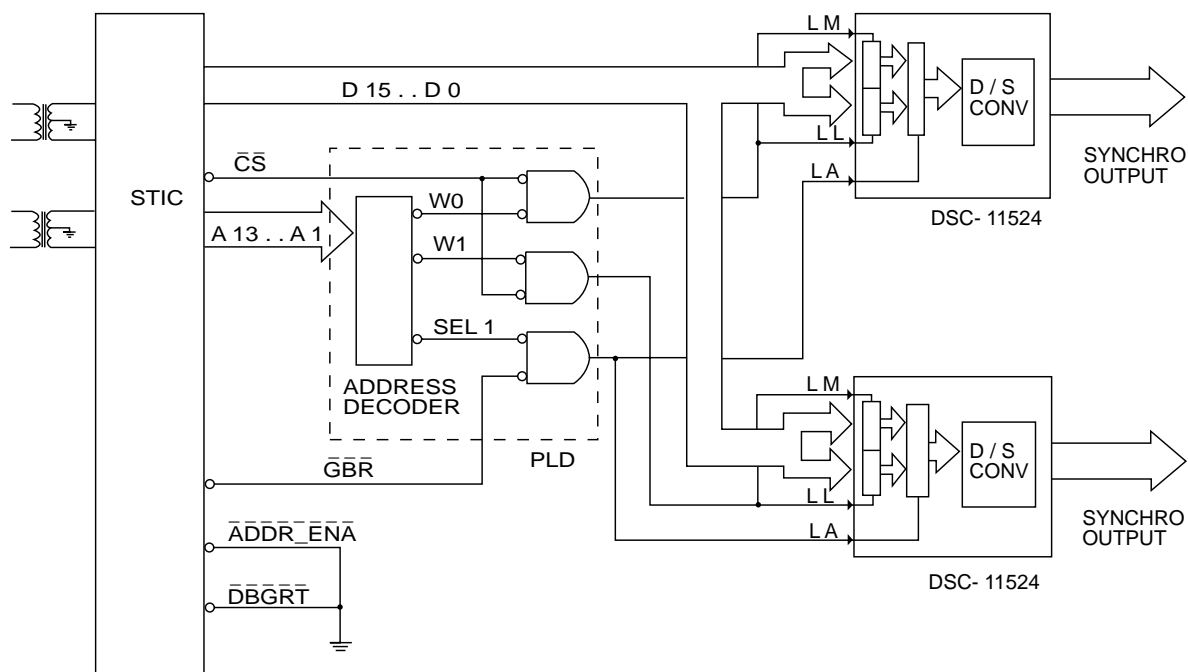
$$\overline{WR0} = \overline{SEL1} \bullet (\text{DATA} / \text{COMMAND} = 1) \bullet (\text{WORD COUNT} = 0)$$
$$\overline{WR0} = \overline{SEL1} \bullet (\text{DATA} / \text{COMMAND} = 1) \bullet (\text{WORD COUNT} = 1)$$
$$\overline{SEL1} = (\text{BROADCAST} = 0) \bullet (\text{T} / \text{R} = 0) \bullet (\text{SUBADDRESS} = 1)$$

FIGURE 4. STIC & DSC-11524 INTERFACE

ignores the Command Word by using address line A6 (data/command) as a qualifier in the address decoder.

After completion of the Command Word transfer, the STIC will set address line A6 (data/command) to logic I indicating that next transfer will be a Data Word. The STIC will present the first received Data Word with a value on address lines A5..A I (the word count) of 0. The value of the address bus during the first Data Word transfer will cause the address decoder to users the WRO select line. The STIC's Chip Select ($\overline{CS}$) output provides a pulse during the due transfer used to strobe the data into the selected latch.

Following each data word transfer, the value on address lines A5..A1 will increment by 1, preparing for the next data word transfer. At the end of a valid, legal receive message, the STIC will users the Good Block Received signal ($\overline{GBR}$) $\overline{GBR}$ is gated with the subaddress #I select line (SEL1) to generate a latch update pulse to the converters. If there was an error in the message, the $\overline{GBR}$ pulse would be suppressed and the latches in the convener would not be updated.

TRANSMIT COMMANDS

Transmit command circuitry is slightly simpler than that required for receive commands. Figure 5 makes use of a simple address decoder to generate output enable signals to the appropriate tri-state buffer as the transmit data words are required (approximately every 20μs during a transmit message). The tri-state buffers enable the subsystem's 16-bit data words onto the parallel data bus. The STIC in turn reads these values and transmits them ova the 1553 serial bus.

The use of a latching tri-state buffer to drive the subsystem data onto the STIC's data bus will provide stable parallel due during the transfer. The latched interface will meet the STIC's transfer timing specifications, and preclude any problems associated with metastability. Figure 5 makes use of 74LS373 buffers. Note that output enable and the latch control inputs to the '373 are tied together. This does not eliminate all time problems but does prevent the data from changing while the STIC is reading it. This requirement is very application dependent (e.g., not a problem for switch closures but could be a problem reading an 8-
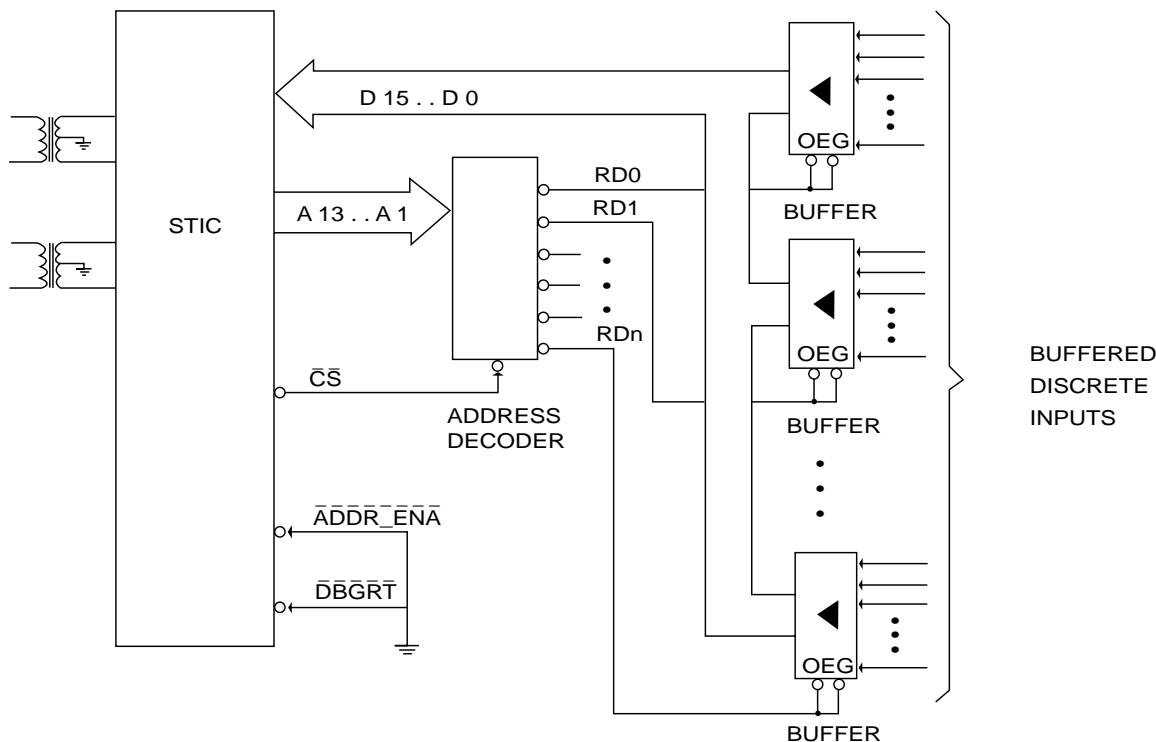
FIGURE 5. TRANSMIT COMMAND CIRCUITRY

bit or 16-bit binary number).

Many devices, such as analog to digital (AID) converters require an input trigger before due may be read The delay times on some of these devices may be very large. Figure 6 shows the STIC's data bus request output (DBREQ) being used to initiate a conversion by asserting the ENCODE input on an ADC-0300 A/D convener. The due ready output signal from the AID is used as the data bus grant (DBGRT) input to the STIC.

The STIC provides several signals that may be used to synchronize subsystem activity to the 1553 bus. New Bus Grant (NBGRT) is an active low output pulse asserted to indicate that the STIC is beginning to process a new command. In Command (INCMD) is an active low output that indicates that the STIC is currently processing a message. INCMD is driven low after the receipt of a valid command to the STIC and remains low until after the STIC has responded to the command

Figure 7 shows an interface between the STIC and two SDC-14620s. An SDC-14620 is a dual 16-Bit synchro-to-digital converter (SDC) from DDC. The SDC-14620's specifications state the output data is not valid for a minimum of 500 ns after the inhibit lines are asserted low. To meet this timing criteria an approach similar to that used in Figure 6 could be used, in which the DBREQ output would inhibit the converters. This method would, however, require an external circuit to generate the delay from DBREQ to DBGRT.

This interface uses the signal INCMD to inhibit the synchro converters during the entire message. This has the added effect of inhibiting all four SDC channels simultaneously, providing the same sampling time, which may be useful in some applications.

The address decoder used in Figure 7 is similar to those discussed in the previous examples. The read outputs (RD3..RD0) are generated based on the value of the address bus and are only driven active during data transfers (when CS is low).

The SDC-14620 provides four control signal for use in enabling its built in data buffers. EM is used to enable the Most Significant Byte, while EL is used to enable the Least Significant Byte. There are separate EM and EL signals for each of the two independent converters contained within the SDC14620.

The worst case propagation delay from EM and EL going low to the parallel output data valid is 150 ns maximum. Running at 16 MHz, the STIC allows for a propagation delay from going low until read due
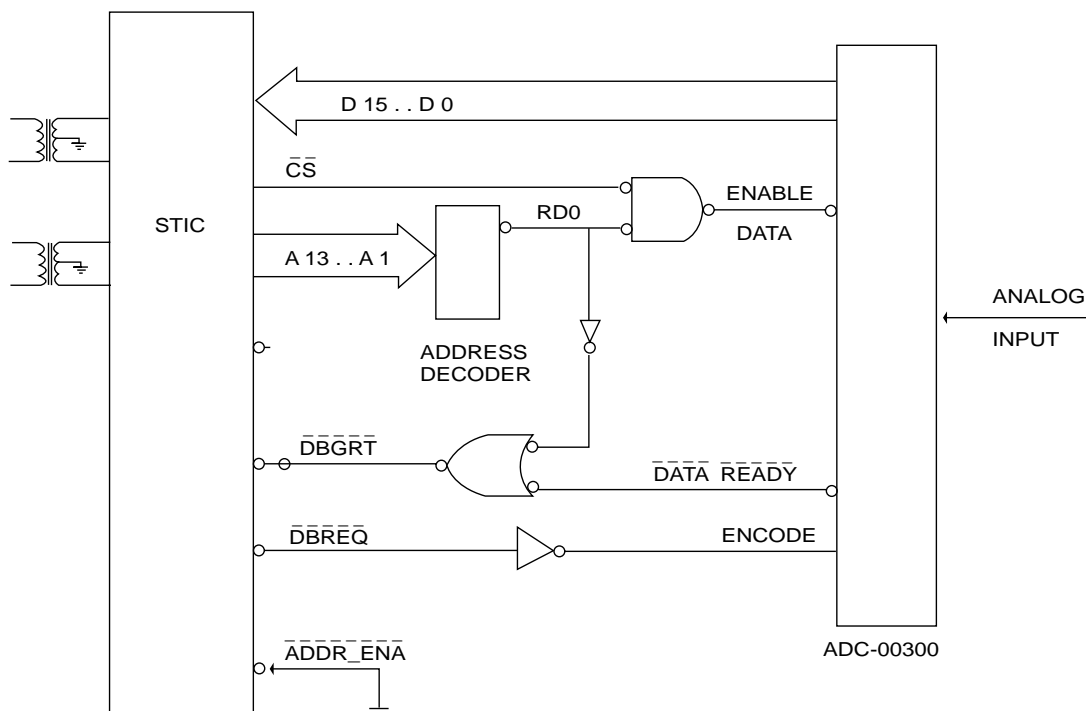
FIGURE 6. STIC & ADC-00300 INTERFACE

must be valid of 240 ns maximum. The propagation delay through the address decoder must be (240 ns - 150 ns) = 90 ns or less.

STATUS BITS

The STIC allows the subsystem to set or clear certain bits in the remote terminal status response. The subsystem flag (SSFLAG) bit is used to indicate that there is a failure within the subsystem. The SSFLAG input may be driven by the output of a built-in-test circuit.

The service request (SERVREQ) bit is used to indicate to the bus controller that an asynchronous event has occurred or that further processing is required. The bus controller will normally follow a service request with a mode transmit vector word. Therefore, implementation of the SERVREQ bit requires implementation of a vector word.

The BUSY bit may be set by the subsystem to indicate to the bus controller that the remote terminal is currently unable to process messages. MIL-STD-1553B Notice 2 discourages use of the Busy bit.

COMMAND ILLEGALIZATION

The STIC has the ability to implement command illegalization through the use of an external PROM or PLD. Upon receipt of an "illegal" command, the STIC will respond with the Message Error bit set in the Status Word, indicating to the bus controller that the command was illegal.

The primary purpose of illegal commands is in the area of system integration to help troubleshoot bus controller software. Most remote terminals are designed to accept a limited number of commands. If a bus controller sends a command to a remote terminal that the terminal was not designed to process, the terminal would then respond with a Status Word indicating that the command was illegal. An illegal command response is normally an indication of an error in the bus controller message.

MODE COMMANDS

But what about mode commands? It must require additional logic to implement mode commands, right? No, the STIC implements all required dual redundant 1553 mode codes with no external circuitry. All terminal related mode commands such as transmit last command, transmit status word, inhibit terminal flag, inhibit transmitter, etc are all handled internally by the STIC. Subsystem dependent mode commands such as mode transmit vector word may be implemented by the designer, if these functions are required.

SELF TEST

The STIC performs a continuous on-line loopback self test. The transmitter and receiver sections of the STIC are completely independent. While the STIC is actively transmitting on the 1553 bus, the receiver section is still active. The loopback test consists of a validity check (encoding, bit count, parity) on **every** word that is transmitted on the 1553 bus by the STIC, and a full 16-bit comparison of the last transmitted word with the last received word at the end of every non-broadcast message.

COMPLIANCE TO MIL-STD-1553B NOTICE 2

MIL-STD-1553B Notice 2 requires a remote terminal that implements the broadcast function have the ability of distinguishing between broadcast and non-broadcast messages. The STIC as address line A13 high to indicate a broadcast message.

MIL-STD-1553B Notice 2 paragraph 30.7 states that a remote terminal is required to implement a data wraparound subaddress. Subaddress 30 (11110) is recommended as the wraparound subaddress. The wraparound subaddress must be designed such that N number of data words may be received to that subaddress and then the same data words may be transmitted from that same subaddress, where N is the maximum word count defined by the remote terminal.

The wraparound subaddress requires that N readback latches be mapped into the address space of the remote terminal. To minimize the amount of circuitry required, it is desirable to keep the maximum word count as low as possible. A simple system may make use of 29 single word subaddresses (remember 00000 and 11111 are reserved for mode commands and 11110 is reserved for wraparound). The use of single word subaddresses will require only a single 16-bit read/write latch for wraparound.
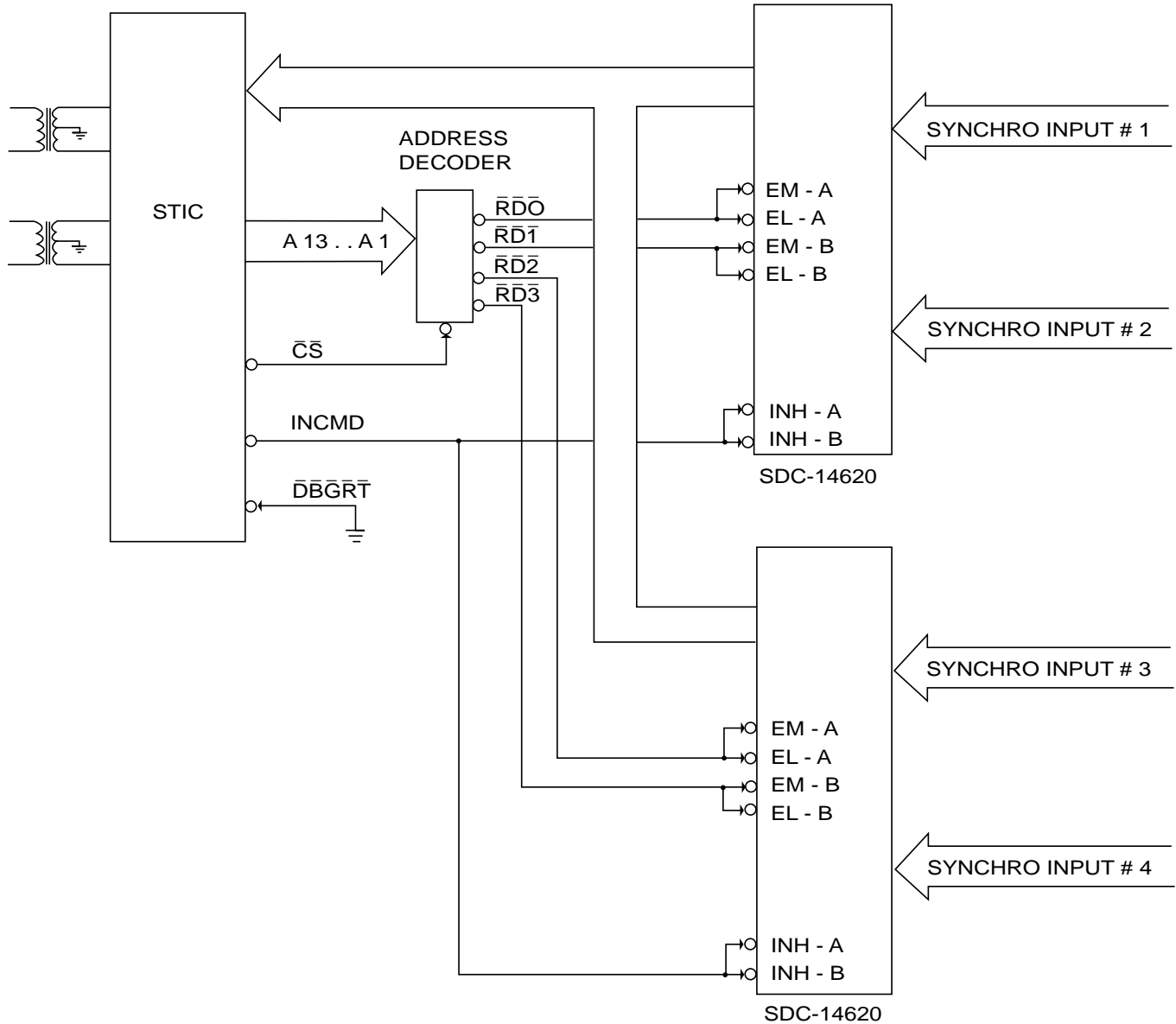
$$RD0 = (BROADCAST = 0) \bullet (T/R =1) \bullet (SUBADDRESS = 2) \bullet (DATA/COMMAND = 1) \bullet (WORD COUNT = 0)$$
$$RD1 = (BROADCAST = 0) \bullet (T/R =1) \bullet (SUBADDRESS = 2) \bullet (DATA/COMMAND = 1) \bullet (WORD COUNT = 1)$$
$$RD2 = (BROADCAST = 0) \bullet (T/R =1) \bullet (SUBADDRESS = 2) \bullet (DATA/COMMAND = 1) \bullet (WORD COUNT = 2)$$
$$RD3 = (BROADCAST = 0) \bullet (T/R =1) \bullet (SUBADDRESS = 2) \bullet (DATA/COMMAND = 1) \bullet (WORD COUNT = 3)$$

FIGURE 7. STIC & SDC-14520 INTERFACE

AVOIDING PITFALLS USING YOUR MIL-STD-1553 CARD

INTRODUCTION

Have you ever purchased a computer interface card, plugged it into your system, and it didn't work the first time? This application note discusses some of the most common installation problems and their solutions. Some of these techniques also apply to cards other than IBM PC cards (e.g. ,VME or PS/2).

MEMORY OR I/O?

First determine if the 1553 card is Memory or I/O mapped. The BUS-65515 and BUS-6517II cards are only memory mapped; the BUS-65529 card is Memory and I/O mapped. The IBM PC (ISA) architecture has some predefined locations in Memory and I/O for its own internal use.

Interrupt levels are also predefined. Interrupt level 6 (IRQ6) is usually not recommended to be used because it's for the hard and floppy disk controller card and would cause a conflict in many PCs. Tables I through 3 list some of the most common predefined locations in your ISA computer.

Your 1553 card should be mapped somewhere between the 640k and 1M boundary. Do not map a card to the reserved locations noted in Tables I and 2. If you do, then either the PC or 1553 card will not function properly. The BUS-65529 and BUS-65515 require 16K (1/4 segment) where the BUS-65517II requires 32K (one-half segment).

WHAT'S AVAILABLE IN MY PC?

There are a number of programs on the market that display Memory, I/O, and Interrupt usage. Some of the most common are Norton Utilities SI, Quarterdeck Manifest, and Checkit. Norton Utilities SI (System Information) program displays Memory segments and Interrupts that are being used.

Even though you map the 1553 card to an available segment this does not guarantee that it will work, but it is a step in the right direction.

The card is in a tree segment and doesn't work. What now? There are a few things to check.

EXTENDED MEMORY MANAGERS AND SHADOW/CACHE RAM

It is possible that the DOS operating system, specifically the Extended Memory Manager (EMM), may be the reason for the conflict. The EMM would try to use our shared RAM on the card for emulating expanded memory. You can exclude one-half segment from the EMM so that our card could be mapped to that location.

For example, if you are using DOS 5.0 and wanted to exclude HEX address D000 from the EMM, you would put the following in the CONFIG.SYS file:

```
DEVICE=EMM386.SYSNOEMS
X=D000-D7FF;
```

Remember, when making a change to the CONFIG.SYS or AUTOEXEC.BAT files, you must reboot the computer so the change will take effect.

If your computer is using Shadow or CACHE RAM,

TABLE 1. TYPICAL I/O MAP	
I/O HEX ADDRESS	**DESCRIPTION**
200-207	Game I/O
O20C-20D	Reserved
21F	Reserved
278–27F	Parallel Printer Port 2
2F8-2FF	Serial Port 2
300-31F	Available (Prototype Card)
360-363	PC Network (Low Address)*
368-36B	PC Network (High Address)*
378-37F	Parallel Printer Port 1
3B0-3BF	Monochrome Display Adapter
3C0-3CF	EGA Adapter
3D0-3DF	CGA Adapter
3F0-3F7	Disk Controller
3F8-3FF	Serial Port 1

* This I/O address may be located somewhere else based on the jumpers on the network card.

TABLE 2. TYPICAL MEMORY MAP	
MEMORY BASE HEX ADDRESS	**DESCRIPTION**
0000-9FFF	64K Base Memory
A000-A7FF	32K
A800-AFFF	32K
B000-B7FF	32K May be used for video RAM
B800-BFFF	32K May be used for video RAM
C000-C7FF	32K Reserved for CMOS
C800-CFFF	32K May be reserved for CMOS
D000-D7FF	32K
D800-DFFF	32K
E000-E7FF	32K
E800-EFFF	32K
F000-FFFF	64K Reserved for BIOS

Note: X000-X7FF and X800-XFFF are one-half segments in the PC.

TABLE 3. TYPICAL INTERRUPT MAP	
INTERRUPT LEVEL	**ISA DEFAULT**
IRQ 2	Cascade, IRQ8-15 mapped here also.
IRQ 3	COM 2
IRQ 4	COM 1
IRQ 5	LPT 2
IRQ 6	Floppy Disk Controller
IRQ 7	LPT 1

you may want to disable this to have a free segment in the PC.

PC BACKPLANE INTERFACE PROBLEMS

An Industry Standard Architecture (ISA) computer bus specification has been developed, based on the design of the original AT. The ISA specification is documented in a preliminary IEEE specification (IEEE 996). It specifies a maximum backplane speed of 8.33 MHz. The speed of the backplane, information being transferred between the card and the PC via the XT/AT connector, is not the same as the clock speed of the PC.

Some computers allow the user to modify the backplane speed from the CMOS Setup program. If you have a 386 (25 MHz) computer, the backplane speed may be running at (Processor Clock / 3) = 8.33MHz. Some computer manufacturers try to increase the performance of their PCs by increasing the backplane speed, violating the IEEE spec. This sometimes causes problems with the communication between the card and the PC (also see "Backplane Handshake Problem" below). A way to solve this problem is to modify CMOS so that the backplane speed is 8.33 MHz or lower.

8-BIT AND 16-BIT CARDS

Another possible problem is having a combination of 8-bit and 16-bit cards in a PC. The IBM AT standard breaks the 640K to IM boundary into three 128K memory blocks or segments. The original IBM PC and PC/XT provided an 8-bit data path with a 20-bit address bus on a single connector. To maintain compatibility, the IBM AT kept the same "XT connector" and added a second connector The second connector provided additional address lines to extend the address bus to 24-bits and provided a mechanism for performing 16-bit due transfers.

To maintain compatibility with existing 8-bit expansion boards, the ISA AT interface bus implements a 16-bit request mechanism that allows both 8-bit and 16-bit interface boards to co-exist in the same computer. The bus master (processor) asserts a signal called $\overline{\text{SBHE}}$ on the second connector (the AT connector) to indicate to the installed expansion boards that 16-bit data transfers may be requested. An expansion board may then request a 16-bit data transfer by asserting either the $\overline{\text{MEMCS16}}$ or the $\overline{\text{IOCS16}}$ signal at the beginning of a transfer cycle.

There are two sets of address lines on the XT and AT Connectors, SA 0-19 and LA 17-23. SA0 through SA19 are latched address lines that are available on

the 8-bit (XT) connector for both 8-bit and 16-bit interface cards. LA17 through LA23 are unlatched address lines that are only available to 16-bit cards through the AT connector. The ISA spec specifies worst case timing parameters for all interface signals. Based on timing specifications,an interface board may not use SA0 through SA 19 for decoding $\overline{MEMCS16}$ (MEMORY CHIP SELECT 16) since the standard does not guarantee that the address lines will be valid before the signal $\overline{MEMCS16}$ is sampled by the AT computer. Decoding $\overline{MEMCS16}$ off the unlatched address bus streamlines the memory transfer cycle and allows decoding of "16-bit" RAM segments that are smaller than 128K, but creates compatibility problems.

When $\overline{MEMCS16}$ is active, a straight decode of the LA 17-23 address lines occur causing all locations within the 128K memory block to be accessed with a single 16-bit read or write If $\overline{MEMCS16}$ is inactive (high), all locations within the 128K block will be accessed by two 8-bit reads or writes.

For example, if the BUS-65529 memory base address is placed at hex location D0000, 16-bit transfers will be performed in the memory range from hex location C0000 through DFFFF. This may or may not create a problem, based on the other interface cards that are installed. The lower boundary of this address range is determined by clearing bits 0 through 16 of the 24-bit physical memory base address of the board to logic zero (which yields a value of 0C0000 for this example). The upper limit of the address range is determined by setting bits 0 through 16 of the memory base address to logic I (which yields a value of 0DFFFF for this example).

As listed in Table 2, video display is usually located in paragraph B and Video BIOS in paragraph C. When you have an 8-bit video card installed and our 16-bit BUS-65529 card, problems will occur if the memory base address Is programmed to paragraphs A or D.

With the base address of the BUS-65529 memory programmed to paragraph A, 16-bit transfers will be performed on any access to both paragraphs A and B. This will cause problems accessing the video page (the host can only read or write the lower 8-bits of each location since the upper 8 data bits are located on the 16-bit connector during a 16-bit transfer).

Placing the BUS-65529 memory base in paragraph D will cause 16-bit transfers in paragraph C, where the BIOS extensions are located in a PROM on the graphics display card. In other words, the host proces-

sor will try to read words, but the board will only provide bytes. This causes the host to only be able to read the lower byte of each word location. The upper byte will be undefined since nothing is driving the upper 8 data bits on the 16-bit connector. The host will thus execute erroneous op-codes and "hang-up".

The BUS-65529 card should be moved to paragraph E, if available, in order for both the video card and 1553 card work properly. Another solution is to replace the 8-bit video card with a 16-bit video card.

BACKPLANE HANDSHAKE PROBLEMS

PCs are designed with an 80X86 processor which provides a handshaking mechanism through the use of a READY type of input. This READY input controls the insertion of "wait states" in a memory or I/O transfer. An active low level on a Memory or I/O Read or Write will indicate that a word transfer is taking place on the bus. The 1553 cards assert IOCHRDY (I/O Channel Ready) low to tell the host processor it is in the process of attempting to transfer information on the bus. The host processor will keep inserting wait states until the interface board asserts IOCHRDY high. According to the IEEE ISA BUS spec, the interface board must assert IOCHRDY low within 70 ns max after the start of the command. Some PC manufactures do not meet the IEEE spec with regards to IOCHRDY and present problems by sampling $\overline{IOCHRDY}$ earlier than 70 ns after asserting $\overline{MEMRD}$, $\overline{MEMWR}$, $\overline{I/ORD}$, or $\overline{I/OWD}$. It is possible to have the computer meet this timing by slowing down the backplane speed if the manufacture provides you this option in the CMOS Setup of the PC.

PRODUCT SPECIFIC PROBLEMS

If you install the board, power on the PC and the computer does not power up, remove the board and change the base memory and/or I/O address to a different address.

If the software package uses a configuration file, make sure it has the fame base and/or I/O address and interrupt values you have jumpered on the card. Check if the configuration file is located in the same directory as the program you are running. In some cases, the configuration file may be in a separate subdirectory.

BUS-65515 CARD INSTALLATION

This is our simplest card to set up. There are two sets of jumper blocks on the card, one for memory base address and one for interrupts. If you notice that the lower 8-bits are being written to or read from the cards memory, it is the 8/16-bit interface problem as dis-

cussed before. In other words, the BUS-65515 card has been located in the same 128K byte segment along with a 16-bit card on the backplane.

BUS-6517II CARD INSTALLATION

This card has one jumper block which is divided into the memory base address and the interrupt level. When you install the software, there is a configuration file that is located in the MB and LB subdirectories.

When you run either the menu software or a C Library program, and an error message appears "Self_Test_Failed", there are three possibilities for this failure:

{1} You did not set up the correct path for the configuration file. There is a batch file called "SETIDEA" that sets a path to the proper subdirectory when running either the menu software or a C Library program. Run "SETIDEA" and try again.

(2) The memory base address jumpered on the card does not match the configuration file. Modify either the file or the jumpers on the board so that the two addresses match.

(3) There is a memory conflict with the memory paragraph you selected. Remove the board, jumper a new address and modify the configuration file to match that address.

If you receive a message "Self_Test_Passed", then the base address is OK. If after you run the Bus Controller, view the communication stack. If the stack contains no data then one of the following two problems exists:

(1) The interrupt value in the configuration file does not match the value jumpered on the board. Modify either the file or the jumpers on the board so that the two interrupt levels match.

(2) There is an interrupt conflict. Remove the board, jumper a new address and modify the configuration file to match that address.

BUS-65529 CARD INSTALLATION

This is a challenging card to set up since it uses both I/O and Memory Base Addressing. There is a diagnostic menu software that accompanies the card. It is recommended that after you jumper the I/O and Interrupt level on the card, you should run this software

Select SELFTEST1 from the menu software and if the PC "hangs," there is an I/O conflict.

SELFTEST1 will test the registers and RAM on the card. The registers are located in the I/O space and the RAM is located in the memory space. You must have the register test pass before getting the memory test to pass. If the registers fail on SELFTEST1 then you need to remove the board, change the I/O address jumpers and modify the configuration file to match the new address. If the registers pass and the RAM fails, then you only have to modify the configuration file for the new base address. This can be done directly from the menu software. If you notice "strange" things happening to your display, you may be writing to the display memory of the video card or the 8/16-bit problem may be the cause.

INTERRUPTS

Interrupt conflicts can occur with any of the three boards mentioned above. The 1553 card outputs are two-state level type of interrupt (not tri-state or open collector like the ISA standard). This means that every card must have its own unique interrupt level.

The interrupts defined in Table 3 may or may not cause a conflict depending on what interrupt drivers are loaded in the PCs memory. If you have a mouse driver loaded, and the mouse is on COM 1, then you will not be able to use IRQ 4. Simply change the interrupt level on your card to the next available one. Some of these same steps can be used for any card being placed in a computer, not just a 1553 card.

CONCLUSION

Using the guidelines outlined, you will avoid the most common pitfalls of installing a computer interface card in your system.

ELECTRICAL AND LAYOUT CONSIDERATIONS FOR 1553 TERMINAL DESIGN

INTRODUCTION

MIL-STD-I553B defines a terminal as "The electronic module necessary to interface the data bus with the subsystem and the subsystem to the data bus… ." By definition, the terminal includes the isolation transformer as well as the analog transmitter/receiver and the digital protocol section.

Design of the terminal, therefore, includes selection and interconnection of the isolation transformer. In order to ensure proper terminal operation in compliance with the 1553 standard, there are a number of issues that need to be considered in this area.

Since most current terminal designs use integrated components such as a Small Terminal Interface Circuit (STIC) or Advanced Communications Engine (ACE), this application note will concentrate on the electrical and layout issues between the hybrid transceiver pins, the isolation transformers, and the system connector. Other matters discussed involve power distribution, layout strategy, grounding, decoupling capacitors, and transceiver-related issues.

ISOLATION TRANSFORMERS

Figure 1 illustrates the interface between the various versions of the ACE and STIC series hybrids to a 1553 data bus. Connections for both direct (short stub) and transformer (long stub) coupling, as well as the peak-to-peak voltage levels that appear at various points (when transmitting are indicated in the figure.

Table 1 lists the characteristics of the required isolation transformers for the various ACE hybrids and lists the DDC and Beta Transformer Technology Corporation (BTTC) corresponding part numbers, as well as the MIL (DESC) drawing number (if applicable). BTTC is a direct subsidiary of ILC Data Device Corporation.

For both coupling configurations. the transformer that interfaces directly to the ACE component is called the **isolation** transformer. As stated above, this is defined to be **part of the terminal.** For the transformer (long stub) coupling configuration. the transformer that interfaces the stub to the bus is the **coupling** transformer.

The turns ratio of the isolation transformer varies, depending upon the peak-to-peak output voltage of the specific ACE or STIC terminal. MIL-STD-1553B specifies that the turns ratio of the coupling transformer be 1.0 to 1.4.

The transmitter voltage of each model of the BU-65170/61580 or BUS-65153 varies directly as a function of the power supply voltage. The turns ratios of the respective transformers will yield a secondary voltage of approximately 28V p-p on the outer taps (used for direct coupling) and 20V p-p on the inner taps (used for stub coupling).

It should be noted that for the 15V or 12V ACE hybrids (BU-65170/61580X1[2]) or STIC (BUS-65153[/54]), the isolation transformer has a **step-down** turns ratio in going from the ACE (STIC) to

TABLE 1. RECOMMENDED ISOLATION TRANSFORMERS				
ACE, STIC PART NUMBERS	**TURNS RATIO**		**RECOMMENDED TRANSFORMER**	
	DIRECT COUPLED	**TRANSFORMER COUPLED**	**PLUG-IN**	**SURFACE MOUNT**
BU-65170/ 61580X1, BUS-65153	1.41:1	2:1	BUS-25679, B-2203, M21038/27-02	B-2387, M21038/27-12, B-2343, M21038/27-17, LPB-5002, LPB-5009, HLP-6002, HLP-6009
BU-65170/ 61580X2 BUS-65154	1:0.83	1:0.60	BUS-29854, B-2204 M21038/27-03	BU-2388, M21038/27-13, B-2344, M21038/27-18, LPB-5004, LPB-5011, HLP-6004, HLP-6011
BU-65170/ 61580X3(6)	1:2.5	1:1.79	B-3087	B-3072
BU-61590 BUS-63149	1:1	1.41:1	BUS-27765, B-2202, DESC M21038/27-01	B-2386, B-2342, DESC M21038/27-11, DESC M21038/27-16, LPB-5003, LPB-5010, HLP-6003, HLP-6010

Notes:

(1) The turns ratio for the direct coupled taps for the B-2204, B-2388, and B-2344 transformers varies slightly from that of the BUS-29854 transformer. They do, however, have the same turns ratios for transformer coupling. For transformer coupled applications, any of the transformers may be used.

(2) The transceiver in the BUS-65164(64), BU-65170X2, BU-61580X2 was designed to work with a 1:0.83 turns ratio for direct coupled applications.

(3) For direct coupled applications, the 1:0.83 turns ratio is recommended, but the 1.25:1.0 ratio may also be used. The 1.25:1.0 turns ratio will result in a slightly lower transmitter amplitude (approximately 4% lower) and a corresponding 4% decrease in the ACE's or STIC's receiver.

the stub. For the 5V ACE hybrids (BU-65170/61580X3), the isolation transformer has a **step-up** turns ratio in going from the ACE to the stub.

ISOLATION RESISTORS

For both coupling configurations, an isolation resistor is required to be in series with each leg connecting to the 1553 bus. This protects the bus against short circuit conditions in the transformers, stubs or terminal components.

For the direct coupled configuration, note that there is a 55 ohm isolation resistor in series with each transformer leg on the stub side. MIL-STD-1553B requires the isolation resistors to protect the bus from a short circuit condition in the stub path, isolation transformer, or transceiver. This allows the 1553 bus to continue operating in the event of a short circuit in the terminal.

As stated in MIL-STD-1553B Notice 2, only the Navy permits the use of direct coupling. Both the Army and the Air Force permit only stub coupling to be used. For almost all system applications, transformer (stub) coupling is preferred over direct coupling.

ADVANTAGES OF TRANSFORMER COUPLING

Some of the advantages of transformer coupling are:

(1) Looking from the 1553 bus towards the stub, the effect of the 1.4 to 1.0 stepdown ratio of the coupling transformer will be to **double** the impedance of the stub/terminal combination, as seen by the bus. Since the stub impedance decreases as a function of stub length due to distributed cable capacitance, this doubling effect serves to reduce the amount of impedance loading on the bus by individual terminals. Heavy stub loading can degrade bus performance by increasing reflections and reducing signal voltages. Stub coupled terminals may be located up to 20 feet from the bus; the distance for direct coupled terminals is limited to 12 inches.

(2) For a stub-coupled terminal, the impedance seen looking into the stub side of the bus coupling transformer is Z_0, assuming that the impedance of the coupling transformer is much higher than the bus impedance Z_0 (70 to 85 ohm). 1553B requires that the coupling transformer have a minimum impedance of 3k ohm, looking from the bus side. Therefore, the characteristic impedance of the stub cabling (78 ohm nominal) matches the stub's load impedance, minimizing reflections back toward the transmitter.

(3) In a direct coupled terminal, the main bus is not protected against a short circuit in the stub cabling. For the transformer coupled case, the
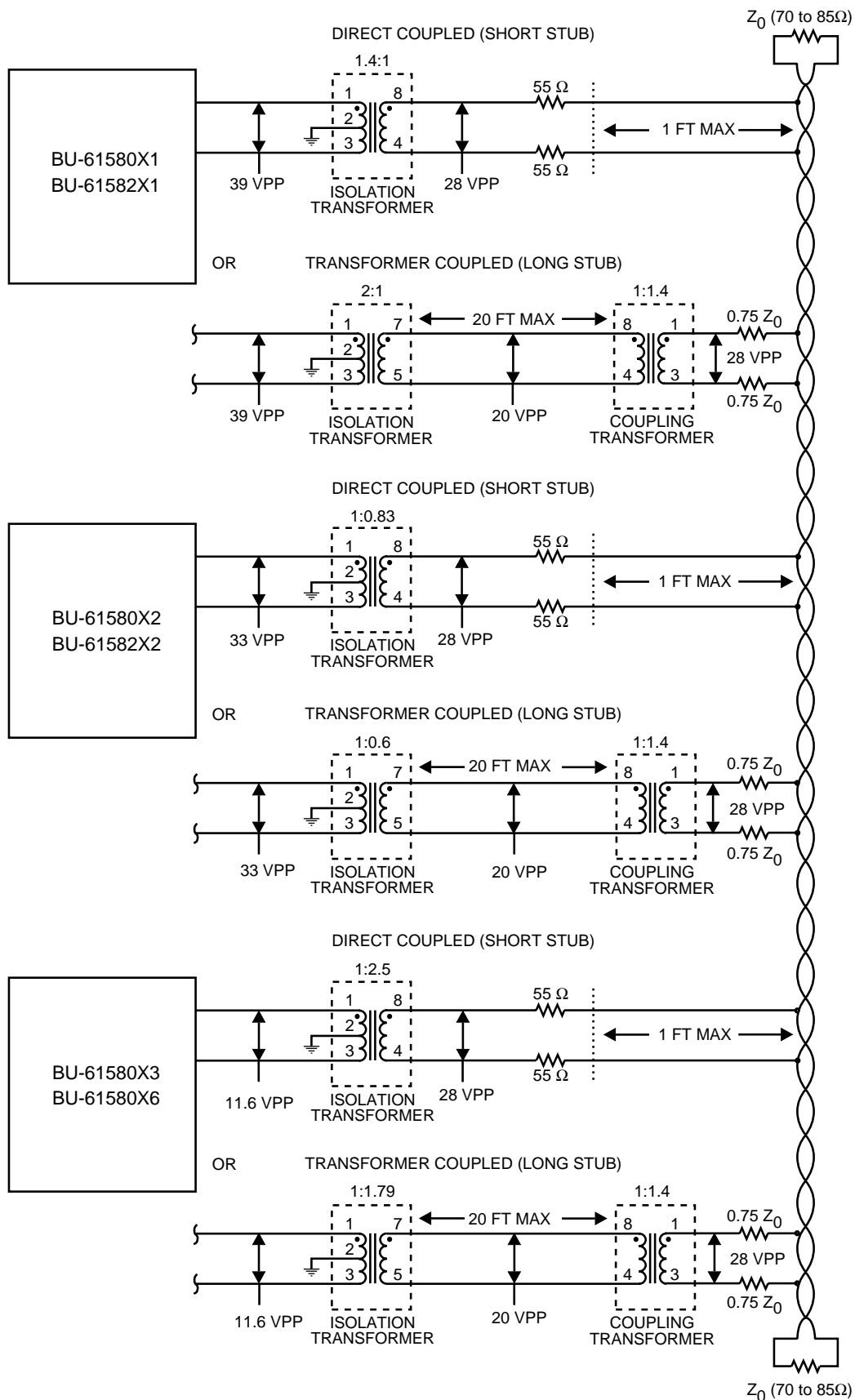
FIGURE 1. ACE/STIC, BUS-66549 INTERFACES TO 1553 BUS (continued next page)
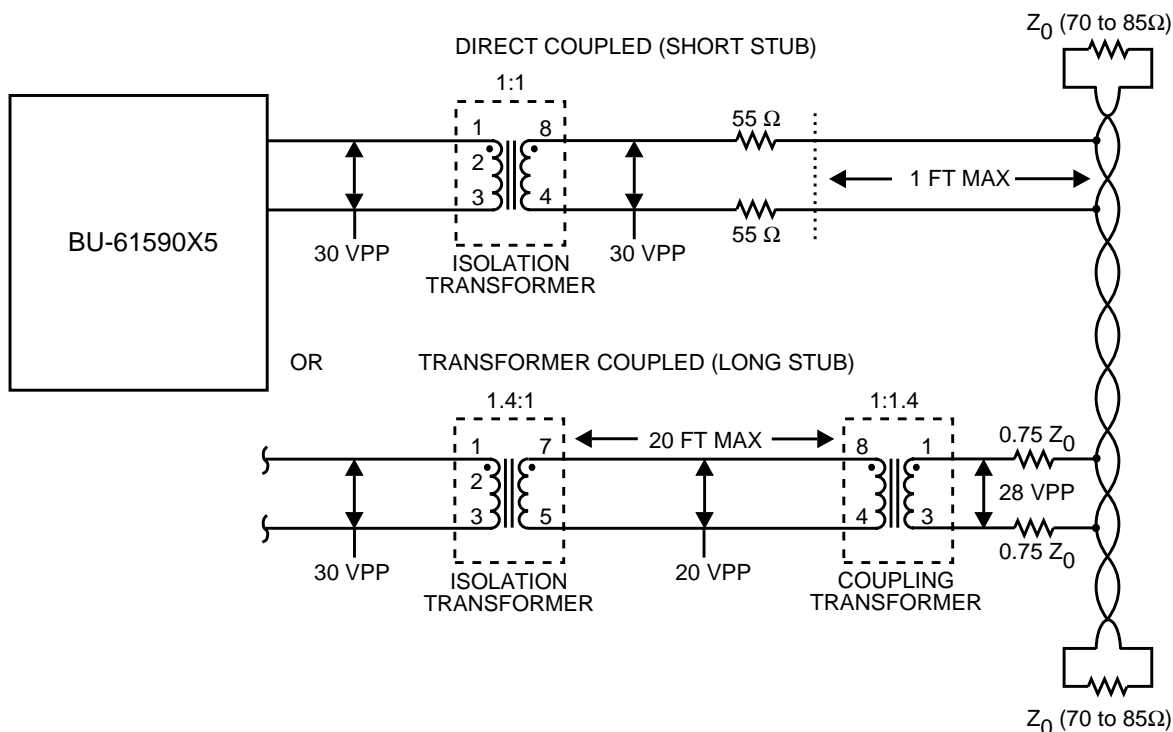
Notes:
(1) Shown for one of two redundant buses that interface to the BU-65170/61580 or BUS-65163.
(2) Transmitted voltage level on 1553 bus is 6 Vp-p min, 7 Vp-p nominal, 9 Vp-p max.
(3) Required tolerance on isolation resistors is ±2%. Instantaneous power dissipation (when transmitting) is approximately 0.5 W (typ), 0.8 W (max).
(4) Transformer pin numbering is correct for the DDC (e.g., BUS-25679) transmitters. For the Beta transformers (e.g., B-2203), or the QPL-21038-31 transformers (e.g., M21038/27-03), the winding sense and turns ratio are mechanically the same, but the pin numbering is reversed, Therefore, it is necessary to reverse pins 8 and 4 or pins 7 and 5 for the Beta or QPL transformers. (Note: DDC part numbers begin with a BUS- prefix, while Beta transformer part numbers begin with a B- prefix.

FIGURE 1. ACE/STIC, BUS-66549 INTERFACES TO 1553 BUS (continued from previous page)

bus is protected against such a fault.

(4) A transformer coupled terminal provides improved dc and common mode isolation over a direct coupled terminal.

TRANSFORMER CONNECTIONS

With the exception of the BU-61590 Universal ACE and the BUS-65149 RT, it is important to note that the transformer center tap on the "ACE" or "STIC" side, pin 2, **must be grounded.** The reason for this is that, at any point in time **only one transformer leg is actively driven** when the ACE (STIC) is transmitting. There is no instantaneous current in the alternate leg. Autotransformer action will result in an equal voltage excursion of the opposite polarity on the alternate leg of the transformer.

Using the BU-61580X1 as an example, during the first half of a Command/Status sync pulse, no current flows from TX/RX, while $\overline{TX/RX}$ is driven to approximately -

10V. Due to the autotransformer inductance TX/RX will swing to about +10V. The resulting primary voltage is nominally 20V peak, or about 40Vp-p.

For the BU-61590 Universal ACE and the BUS-65149RT hybrid, the transformer center tap, pin 2, **MUST NOT be grounded.** The transmitters in these hybrids include complimentary outputs **that drive both transformer legs simultaneously** (with the opposite signal polarity).

If the transformer center tap is grounded, there is a strong possibility that there will be a "dynamic offset" voltage (residual voltage) following the end of transmission by the BU-61590 or BUS-65149.

LAYOUT CONSIDERATIONS

Figure 2 illustrates the suggested PC board layout for the BU-61580D1 version (+5/-15V) of the ACE hybrid, two BUS-25679 (or equivalent) transformers, the clock oscillator (16 or 12 MHz), and decoupling capacitors.
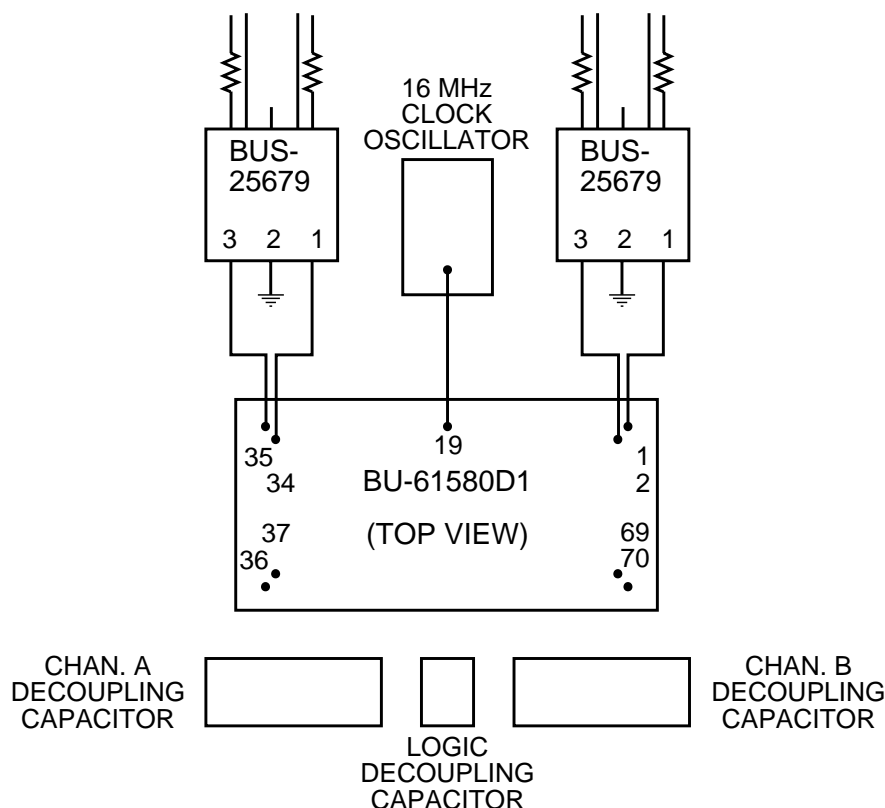
Notes:
(1) Physical spacing between ACE (STIC) transformers must be kept to a minimum.
(2) There must be **NO GROUND OR POWER SUPPLY PLANES** underneath the signal traces running to or from the transformers.
(3) The grounds for the analog and digital sections of the ACE (STIC) are not connected internally.

FIGURE 2. SUGGESTED PC BOARD LAYOUT

With regard to the suggested layout of Figure 2, there are a number of important factors to consider relating to component placement, circuit routing, power distribution, grounding, and decoupling capacitors.

Isolation Transformers

The isolation transformers should be located **as physically close as possible** to the respective TX/RX pins of the hybrid. When transmitting, the typical peak currents in the primary legs are 150 to 200 mA for a 15V or 12V transceiver or 450 to 500 mA for a 5V transceiver. Resistive and inductive voltage drops are minimized by providing widened traces and minimizing the length of the traces. This is particularly important for the 5V (only) ACE.

Crosstalk.

In addition to limiting the inductive and resistive voltage drops in the analog signal traces when transmitting, reducing the hybrid-to-transformer spacing serves to minimize crosstalk to the terminal's receivers from other signals on the board. Severe crosstalk can increase the terminal's word error rate above the maximum level of 10^{-7} allowed by MIL-STD-1553B. Another important precaution regarding crosstalk is to avoid running other analog and digital signal traces (particularly high-speed digital signals) in close proximity to the 1553 analog signal traces. This applies to the signal traces on **both** sides of the transformer.

It is most critical to avoid routing other signals on layers of the PC board that are adjacent to and in parallel with the 1553 analog signals. Such signals can result in the worst case crosstalk.

Ground Planes.

As is the rule in all high-speed digital circuits, it is a good practice to use ground and power supply planes under the ACE as well as the host processor and any digital "glue" logic.

However, it is very important that there be no ground and/or power supply planes underneath the analog bus signal traces. This applies to the

TX/RX signals running between the hybrid and the isolation transformer as well as the traces between the transformers to any connectors or cables leaving the board.

The reason for avoiding running supply or ground planes under the analog signal traces is that the effect of the distributed capacitance will be to lower the input impedance of the terminal. as seen from the MIL-STD-1553 bus. MIL-STD-1553B requires a minimum of 2k ohm input impedance for direct coupled terminals and 1k ohm for transformer (stub) coupled terminals. If there are ground planes under the analog signal traces, it is likely that the terminal will not meet this requirement. It has been found that placing a ground plane under the isolation transformers only slightly effects the input impedance. A ground and/or power plane may be placed under the transformers, if desired.

Power and Ground Distribution.

Another important consideration for 1553 transceiver operation is power and ground distribution. Refer to Figure 3.

For the ACE (STIC) hybrid/transformer combination, the high current path when the ACE is transmitting will be from the -15V (or -12V or +5V) power supply, through the ACE's (STIC's) transmitter output stage, through one leg of the isolation transformer to the transformer center tap.

It is important to realize that the high current path is **through the transformer center tap and not through the ACE's (STIC's) GNDA and GNDB pins.**

Two exceptions to the operation described above are the BU-61590 Universal terminal and the BUS-65149 RT hybrid. With these two units, it is important to note that the transmitter provides a **differential** output stage, actively driving both transformer legs with opposing polarity signals.

For the BU-61590 or BUS-65149, the transformer center tap must not be grounded. In this case, the heavy transmitter supply current runs from the +15/12V (-15/-12V) supply, through the transformer primary and back to the -15/-12V (+15/-15V) supply. Like the other ACE and STIC units, there will not be a large transmitter return current flowing through GNDA and GNDB.

A worst-case system design should ensure that with minimum supply voltage and calculated voltage drops, the transceiver voltage provided between the ACE's (STIC's) transceiver supply pins and the center tap of the respective isolation transformer will be no
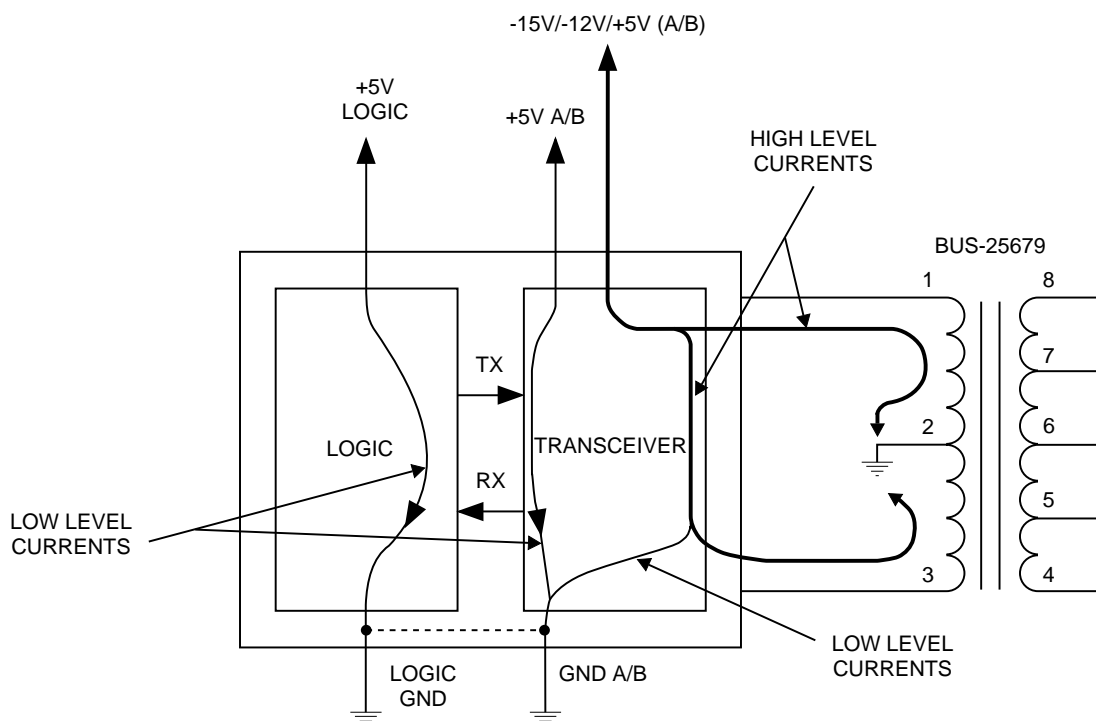
FIGURE 3. POWER/GROUND CURRENT DISTRIBUTION

less (in absolute value) than the specified minimum (-14.25V, -11.6V, or +4.75V).

In some cases, the voltage drop may be reduced by means of large decoupling capacitors, but the best practice is to minimize voltage drops in the power supply distribution.

Analog and Digital Grounds.

It is important to note that the logic ground and transceiver grounds are connected together internally in the STIC or ACE hybrids. These grounds must be connected together externally.

As far as the ACE (STIC) and its associated isolation transformer are concerned, the optimal circuit layout would entail a single ground plane for both the digital and analog (transceiver) circuits. While this is sometimes possible, in many applications, there are system requirements for separate analog (-15/-12/+5 (analog)) and digital (+5V) power supply returns,

In this case, the transformer center tap should be connected through a low impedance path to the analog return, not the digital return.

This provides the advantage of separating the analog and digital ground currents. It is assumed that the two return paths are ultimately bonded with low impedance connections to the system ground near the power supply.

In order to minimize the possibility of ground noise corrupting the protocol/transceiver interface within the ACE (STIC), it is best that both the LOGIC GND pin as well as the GNDA and GNDB pins be connected to the logic ground as close as possible to the hybrid.

Decoupling Capacitors.

When the ACE (or STIC) is transmitting, it is drawing

relatively large pulsating currents from the active transceiver power supply. For the -15V or -12V unit, this current will generally be in the range of 150 to 200 mA peak and can be as high as 300 mA. For a +5V unit, the peak current can be as high as 800 mA. The frequency content of the power supply current can include components in the ranges of DC to 30 kHz on the low end, and 500 kHz to 2 MHz (and above) on the high end.

Resistive and inductive voltage drops in the power distribution network can result in ripple voltages on the ACE's (STIC's) power supply inputs. It is strongly recommended that decoupling capacitors be used to reduce the level of the ripple voltage.

For the transceiver power inputs, it is generally necessary to use small decoupling capacitors to eliminate the high frequency (500 kHz to 2 MHz) power supply ripple. For -15V and -12V power inputs, 2.2 µF low ESR/ESL capacitors should be sufficient. For the 5V only ACE, 6.8 µF low ESR/ESL capacitors should be used.

If good power distribution practices are maintained, the small capacitors specified in the preceding paragraph should be sufficient. If, however, the trough of the ripple falls more than 5% below the nominal supply voltage, the transmitter output may fall below the minimum level of 6 Vp p for a direct coupled output, 18 Vp p for a MIL-STD-1553B transformer coupled output, or 20 Vp p for a MIL-STD1760B transformer coupled output. in this case, a larger capacitor may be needed to sustain the minimum voltage level.

For the +5V LOGIC input and +5V A/B inputs (for a -15 or -12V ACE (STIC unit), 0.01 µF is generally sufficient.

Avoid Switching Power Supplies and DC-to-DC Converters.
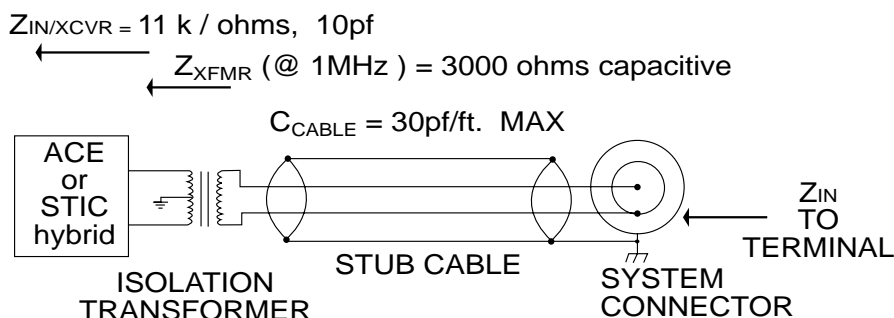
FIGURE 4. ISOLATION TRANSFORMER-TO-SYSTEM CONNECTOR IMPEDANCES

(A)

(B)

(C)
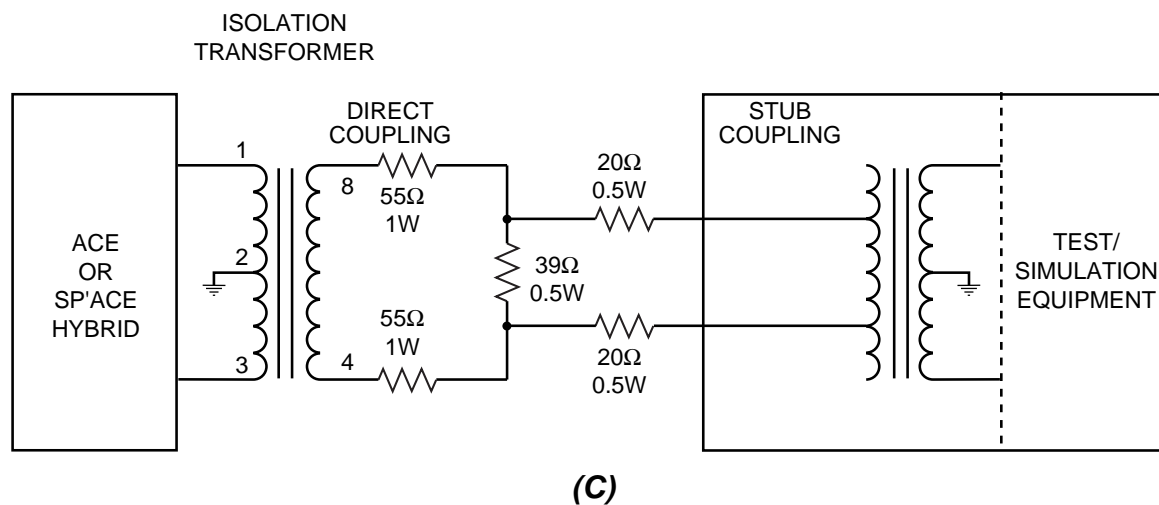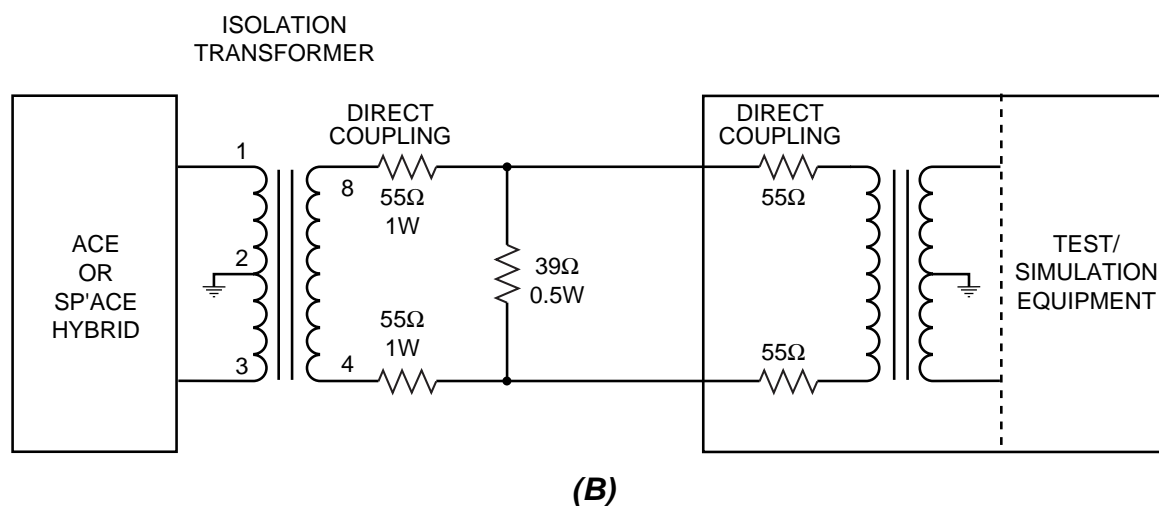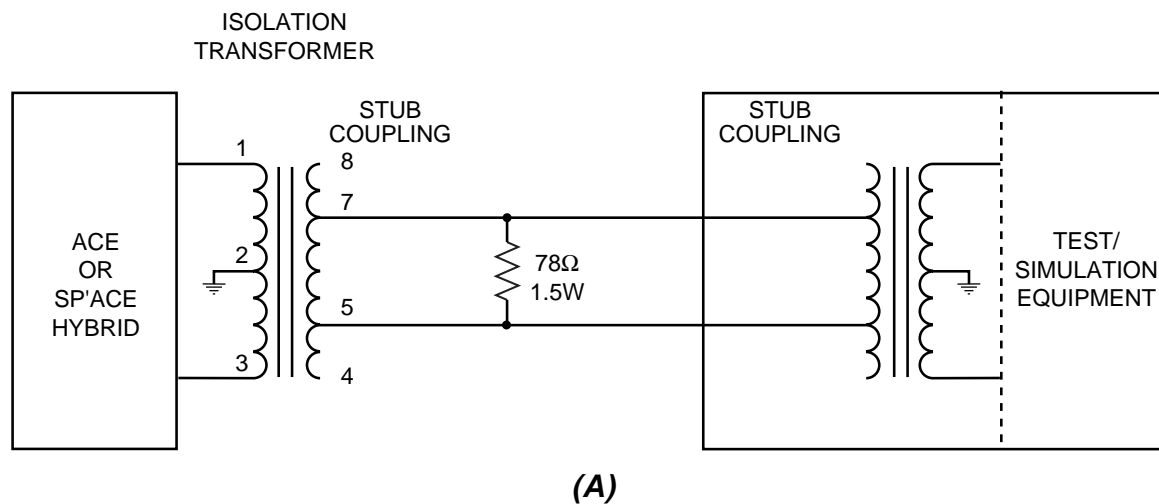
FIGURE 5. "SIMULATED BUS" INTERCONNECTIONS

(A) DIRECT-COUPLED-TO-DIRECT-COUPLED.

(B) TRANSFORMER-COUPLED-TO-TRANSFORMER-COUPLED.

(C) DIRECT-COUPLED-TO-TRANSPARENT COUPLED

The use of switching power supplies or DC-to-DC converters on the same board as the ACE or STIC hybrid should be avoided. The switching noise may result in poor performance of the ACE's (STIC's) transceiver. Some of the potential problems include high output noise as well as degraded zero-crossing tolerance and word error rate.

Location of Clock Oscillator

As illustrated in Figure 2, the 16 MHz or 12 MHz clock oscillator should be located as close as possible to the ACE's (or STIC's) CLK IN input pin in order to minimize attenuation, distortion, and corrupting crosstalk of the clock signal. The duty cycle of the clock input should be between 40% and 60% in order to optimize operation of the Manchester decoders and processor/memory interface control logic. Some of these circuits utilize both edges of the clock input.

ISOLATION TRANSFORMER INTERFACE TO SYSTEM CONNECTOR; INPUT IMPEDANCE CONSIDERATIONS.

The general practice in connecting the stub side of a transformer (or direct) coupled terminal is to make use of 78 ohm twisted-pair shielded cable. This serves to minimize impedance discontinuities.

The decision of whether to isolate or make connections between the center tap of the isolation transformer's secondary, the stub shield, the bus shield, and/or chassis ground must be made on a system basis, as determined by an analysis of ESD, EM/RFI. and lightning considerations.

There is some controversy within the 1553 community as to where a terminal ends and a stub begins. The issue is not purely academic, since it has a profound effect on the testing of terminal input impedance. In order to pass the 1553 RT Validation Test Plan, a transformer (stub) coupled terminal must have an input impedance of at least 1k ohm over the frequency range from 75 kHz to 1 MHz. Paragraph 3.10 of MIL-STD-I553B asserts the definition of a terminal as "The electronic module necessary to interface the data bus with the subsystem and the subsystem to the data bus. Terminals may exist as separate line replaceable units (LRU's) or be contained within the elements of the subsystem."

For a transformer (stub) coupled terminal, it is the author's interpretation of Paragraph 3-10 (particularly the first sentence) that the terminal *ends at the pins of the isolation transformer.* This interpretation is consistent with Figures 9 and 10 in the I553B standard. However, the armed services, faced with the practical task of testing and maintaining equipment, take a different interpretation. They contend that the terminal *includes the wiring path up to and including the system connector.* As a result, the distributed capacitance of PC board and backplane traces, cabling and connectors are included in the impedance measurement.

For the -15V/+5V ACE and STIC hybrids the transceiver input impedance is specified as follows: 11k ohm min, 10 pF max. Reference Figure 4. This characterizes a differential measurement taken between the TX/RXA(B) and $\overline{TX/RXA(B)}$ pins of the hybrid. It includes both the transmitter and receiver (tied together internally), is applicable to both powered and unpowered conditions with power and ground pins connected, and assumes a 2 Vrms balanced, differential, sinusoidal input for making the impedance measurement.

At 1 MHz, 11k ohm and 10 pF represent a composite impedance of 9.05k ohms at -34.6°. Assuming an ideal (Z_{in} =) 2:1 isolation transformer for a transformer coupled terminal, this reflects an impedance of 2.26k ohms at -34.6° on the stub side. Note this is "worst case." Typical numbers are about twice as high.

The -34.6° phase angle indicates a reflected ACE (STIC) input impedance that is somewhat more resistive than capacitive. The impedances of all the other elements in the path — the isolation transformer, PC board and backplane traces, cable, and connector — are capacitive. These capacitances add in parallel to form the "terminal's" input impedance (as defined by the armed services).

Assuming the worst case parameters for the ACE (STIC), these elements must have a minimum parallel impedance of 1.47k ohm (capacitive) to keep the parallel impedance of the "terminal" above 1k ohm. At 1 MHz, the highest frequency required for the 1553 impedance test, this translates to a maximum capacitance of 108 pF. Since most 1553 isolation transformers specify an input impedance of at least 3k ohms (some are higher), representing about 53 pF, this allows about 55 pF for the rest of the signal path. MIL-STD- I 553B specifies that the cable capacitance of the bus cable be less than 30 pF/ft. 55 pF represents 1.8 feet of such "worst case" cable.

In order to ensure that the "terminal" input impedance complies with MILSTD1553B, it is important to try to
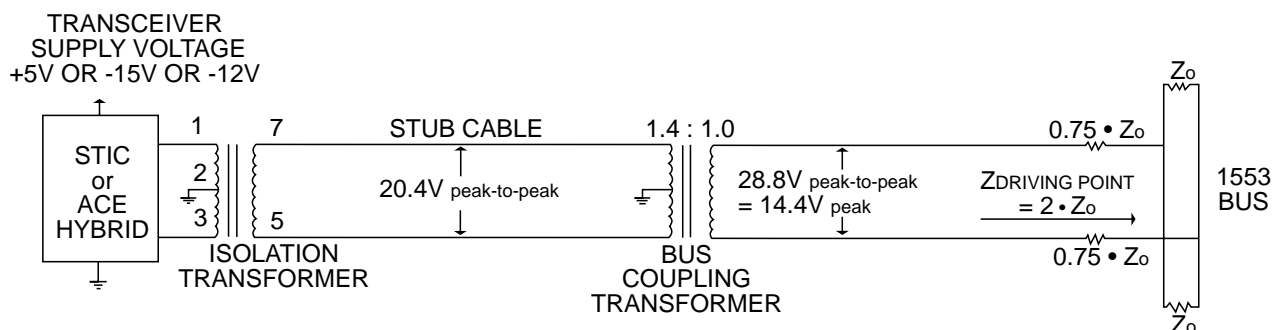
FIGURE 6. STUB COUPLED TERMINAL TO DATA BUS

minimize the distributed capacitance between the 1553 signals and ground. To achieve this, the length of internal stub wires must be kept to the shortest length possible. If this is not possible, approaches that can be used include not grounding the cable shield or using unshielded twisted pair cable.

Another approach is to use a pair of wires, rather than a twisted/shielded pair. In this case, the effect of the bus signals coupled from the unshielded wires on the rest of the system performance must be considered as a tradeoff against the benefit of the increased terminal impedance.

"Simulated Bus" (lab bench) Interconnections

For purposes of software development and systems integration, it is generally not necessary to integrate all of the required couplers, terminators, etc. that comprise a complete MIL-STD-1553B bus. In most instances, a simplified electrical configuration will suffice.

The three connection methods illustrated in Figure 5 allow an ACE (or STIC) to be interfaced over a "simulated bus" to simulation and test equipment, such as a BUS-6517II board. It is important to note that if a compliant I553 bus configuration is not used (as in Figure 1), the **resistors shown in the diagram are necessary** in order to ensure reliable communications between the ACE or STIC and the simulation/test equipment.

TRANSCEIVER ISSUES

This application note considers a number of transceiver related issues that are commonly asked about. These include: voltage vs. current drivers. the effect of receiver threshold on MILSTD-1553B validation testing, and consumed vs. dissipated power.

Voltage vs. Current Drivers.

The transmitter in the −15/+5V and −12/+5V ACE and

STIC terminals are voltage type transmitters. The transceiver in the +5V only ACE is a current type of driver. For both of these transceivers, only one leg of the isolation transformer primary is actively driven at any point in time. For the BUS-65149 and BU-61590 universal terminals, a true differential type of transmitter output stage is provided: for these hybrids, both legs are driven simultaneously.

Voltage source transmitters provide superior line driving capability for driving long cables and heavy amounts of stub/terminal loading. On the other hand, current source drivers, due to their simpler design, tend to consume and dissipate lower levels of power.

The transmitter requirements are a bit different for MIL-STD-1760B. In MIL-STD-1553B, the output of a transformer coupled terminal is required to be in the range of 18 to 27Vp-p. MIL-STD-1760B is more stringent, requiring a transmitter stub voltage of 20 to 27Vp-p. This higher voltage ensures robust operation over a range of network topologies (stub and bus lengths, separation distances) for various aircraft. It is also recommended that a voltage, rather than a current source transmitter, be used in 1760 applications. A voltage source transmitter is more robust on a stores bus with its detaching stubs and the resulting variations in load impedance.

The transmitter in the −15/+5V versions of the STIC and ACE terminals provides a voltage source with a minimum stub voltage of 20Vp p, making these products suitable for −1760 applications.

MIL-STD-1553B specifies that a transformer coupled terminal must be able to operate with an incoming line-to-line signal in the range of 0.86 to 14.0Vp-p. The standard also requires that a terminal not respond to a signal in the range of 0.0 to 0.20Vp p. In DDC's terminals, the receiver threshold is generally trimmed for between 0.550 and 0.700 Vp p. This range provides optimal performance for the Manchester decoders in the ACE and STIC termi-
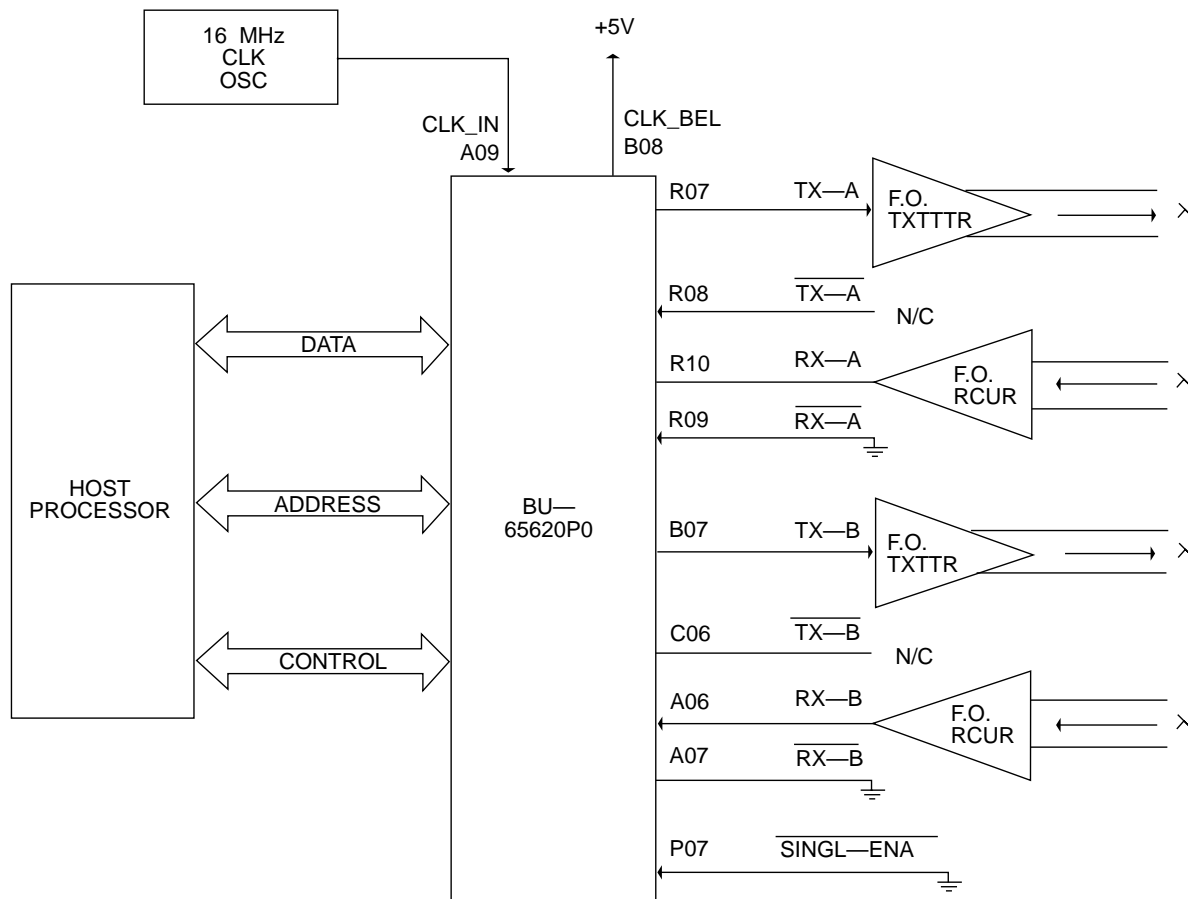
FIGURE 7. BU-65620 TO FIBER OPTIC TRANSCEIVER INTERFACE

nals.

Receiver Threshold.

If receiver threshold is sufficiently below 0.550Vp-p, there is an increased possibility of failing the word error rate (noise) test of the RT Validation Test plan, even if the threshold is in spec. A 1553 terminal's word error rate is primarily a function of the threshold-to noise ratio. Going to the other extreme, a receiver threshold that is too high (too close to 860 mV p-p) may cause problems in passing the zero crossing deviation and/or common mode rejection tests.

In the zero-crossing deviation test, the terminal must accept as valid a zero crossing-to-zero crossing time of 350 ns (500-150). If the receiver's threshold is too high, the pulse width (as seen by the terminals' Manchester decoder) will be reduced to an unacceptably short interval.

The validation test for common mode rejection involves superimposing a common mode voltage on a stub voltage of 0.86 Vp-p. Since the terminal's transformer has a finite common mode rejection ratio, this is tantamount to lowering the received signal level below 0.86V. Therefore, a threshold too close to 0.86 Vp-p may result in a failure of the validation common mode test.

Power Consumption vs. Dissipation

Another common point of misunderstanding regarding transceivers involves power consumption and power dissipation. Figure 6 illustrates a typical configuration of a stub coupled terminal and a 1553 bus. When the terminal is not transmitting, the power consumption and dissipation are essentially equal.

However, when the terminal is transmitting, there will be a difference between the consumed and dissipated power. The difference in the two power numbers is the power dissipated in external isolation and termination resistors.

As shown in Figure 6, the voltage on the "bus" side of the coupling transformer is approximately 28.8 Vp-p, or about 14.4Vpk.

That is, $V_{pk} = 14.4$.

The load resistance, as seen from this point, consists of the two termination resistors (in parallel), in series with the two isolation resistors.

This is equal to

$$(0.75 \cdot Z_0) + (0.75 \cdot Z_0) + (Z_0/2) = 2 \cdot Z_0.$$

If the transmitter waveform is assumed, for the moment to be a square wave, its RMS value will be equal to its peak value, Vpk. With this assumption, the combined instantaneous dissipation of the isolation and termination resistors, when the terminal is transmitting, is calculated as follows:

$$P_{LOAD} = (V_{pk})^2/(2 \cdot Z_0)$$

Since the 1553 waveform is a trapezoidal signal, rather than a square wave, the actual P_{LOAD} will be slightly less than that calculated by the equation given above. That is,

$$P_{LOAD} = K \cdot (V_{pk})^2/(2 \cdot Z_0).$$

The exact value of K is a function of the transmitter's rise and fall times as well as the actual transmitted data pattern K decreases with an increase in rise/fall times and the number of signal transitions in the transmitted waveform. As a first approximation, a value of $K = 0.9$ may be used.

Consider a transformer coupled terminal with a transmitter stub voltage of 20.4Vp-p and a bus with a characteristic impedance of $Z_0 = 78$ ohm. These are typical values. The voltage on the "bus" side of the coupling transformer will be:

$$2 \cdot 20.4 = 28.85 \ V_{p-p}, \text{ therefore,}$$

V_{pk} will be $28.85/2 = 14.425V$. As a result,

$$P_{LOAD} = (0.9) \cdot (14.425)^2/(2 \cdot 78) = \textbf{1.2W}$$

1.2 W represents the instantaneous dissipation of the isolation and termination resistors when the terminal is transmitting.

For a typical terminal, the actual transmitter duty cycle (portion of time transmitting) is typically between 1% and 10%. The average external dissipation is given by:

$$P_{LOAD} /_{AVG} = D \cdot 1.2W,$$

where D = transmitter duty cycle.

Both power supply current and terminal power dissipation vary linearly as functions of the transmitter duty cycle.

For example, consider a BU-61580D1 ACE terminal operating at 25% duty cycle. At 25% duty cycle, the average load dissipation is $(0.25) \cdot (1.2) = 0.3W$. Based on the "max" spec numbers, the total consumption:

$$= (5V) \cdot (190 \ mA) + (1 \ 5V) \cdot (108 \ mA)$$

$$= 0.95 + 1.62 = 2.57W.$$

This is consistent with the "total hybrid" power dissipation spec, listed as 2.25W max, at 25% duty cycle.

BU-65620-TO-FIBER-OPTIC-TRANSCEIVER INTERFACE

For MIL-STD-1773 applications involving a fiber optic transceiver, the BU-65620 digital monolithic version of the ACE may be easily interfaced to a fiber optic transceiver. To facilitate this interface, the Manchester decoders in the BU-65620 provide a pin-programmable option allowing them to accept the single-ended input signals from a fiber optic receiver.

As shown in Figure 7. this option is activated by strapping the input signal $\overline{SNGL\_ENA}$ to ground.

UNDERSTANDING THE ACE'S
SELF TEST CAPABILITIES

INTRODUCTION

One of the salient features offered by ILC Data Device Corporation's ACE series 1553 terminals is its TEST MODE option. This option allows a comprehensive in-line functional test of the 1553 interface portion of a system. All testing is performed through the CPU interface; no additional logic is required.

In the TEST MODE, the ACE's logic is divided into major sections and functions are tested independently. Read and write access to internal control signals is available through register locations. With this option, a minimum of 99% test coverage of the logic portion of the ACE terminal can be achieved.

For test purposes, it would be helpful to think of the ACE's logic in terms of the major building blocks. These include: 1553 decoders, protocol sequencer, 1553 encoder, registers, memory, and CPU interface.

All of these sections need to be tested.

Without TEST MODE, the host CPU can exercise the memory section, some of the registers, and the CPU interface by doing a number of read and write operations.

However, to fully verify the correct operation of the 1553 interface (decoders, protocol sequencer, and encoder sections), a 1553 bus simulator/tester was required.

Now, through the use of TEST MODE, 1553 bus activity can be simulated through the CPU interface by performing a series of read and write operations. This checks out these portions of the circuit without disturbing the actual 1553 bus.

TEST MODE PROVIDES HOST CPU COMPLETE ACCESS

The main concept behind the TEST MODE is to provide the host CPU with complete access to all sections. Register locations are used so additional control lines are not required. The 1553 portion of the ACE is divided into three major sections. Control input signals from the other sections are replaced by inputs that the user's CPU can control.

In this way, the CPU exercises the individual sections directly without going through a lot of preconditioning. Also, to verify the correct operation of each section the CPU is provided with a mechanism for reading the section's output control lines by means of register locations so that it may monitor the section directly.

For those "non-1553" sections (e.g., registers that are accessible to the processor in the normal operating mode), features were added to simplify testing.

ACCESSING TEST REGISTER LOCATIONS

In order to access the test register locations:

(1) Enable enhanced mode operation by setting the bit 15 (MSB) of the Configuration Register #3 (register address 0007h) to logic " 1 ".

(2) Program Configuration Register #4 (register

address 0008h).

The last three bits of Configuration Register #4 enable TEST MODE operations. The value contained in these bits determine what portion of the circuit the user has access to and what portions are kept in reset. If one or more of these bits are set, TEST MODE is enabled and the user has access to register locations 16 through 23. Table I lists the actual breakdown of the different test modes available to the user by means of Configuration Register #4. Each mode is detailed in the following paragraphs.

REGISTER MAPPING

Before delving deeper into the TEST MODE features of the ACE, it is important to understand a little bit about the register mapping.

Only five address lines are really required. Register addressing is module 32.

The first 16 register locations (00–15) are optional registers, as defined in the ACE User's Guide. The next eight locations (16–23) are used for test purposes only. The last eight locations (24–31) are not used by the ACE but are reserved for future expansion.

Writing the last eight locations, or the test registers when they are not enabled, will not change any internal information. Reading these locations will yield whatever data was last put on the internal data highway.

The actual definitions of the test mode controls, register locations 16 through 23 will depend upon what section has been selected and whether the user is doing a read or a write operation.

Write-Operation

For write operations, locations 16 and 17 are used for latched signals, locations 18, 19, and 22 are used to generate key strobes, and locations 20 21, and 23 are used to preload internal registers.

It is important to note that the definition of the latched signals in location 17, during a write operation, depends on the value of the last three bits of Configuration Register #4. These bits are multiplexed for the various test modes.

Read Operation.

For read operations, locations 16, 17, 19, and 23 are used to read groups of key test signals and locations 18, 20, 21, and 22 are used to gain access to the values in internal registers.

The definition of register location 23 during a read operation will depend upon the mode of operation selected (RT, BC or Monitor) in Configuration Register #1.

1553 DECODER TESTING

When the last three bits of Configuration Register #4 are set to 001, the 1553 decoder section has been selected for testing.

In this mode of operation, the user has the ability to take the 1553 decoders offline, to input the RX and RX_L data that normally come from the 1553 transceivers, and to single step the internal shift clocks.

Data can be examined as it is being shifted. All the decoding logic can be verified. Most of the circuit can be tested at reduced speeds where timing is not critical.

The only portion that would be time critical is if the user is inputting the RX and RX_L data and not single stepping the internal shift clocks. In this case, the registers would need to be updated every 500 ns.

When testing the decoder section, the user should keep the protocol section in Idle mode (by setting bits 15 and 14 of Configuration Register #l to logic "1") to prevent any response by the protocol sequencer.

1553 ENCODER TESTING

When the last three bits of Configuation Register #4 are set to 010, then the 1553 encoder testing is enabled.

The user is able to input all control signals normally supplied by the protocol sequencer, as well as to set up the encoder for offline wraparound testing, specifying the channel, sync type, data and the number of words. He may verify the correct operation through his 1553 decoders.

TABLE 1. TEST MODE OPERATION			
CONFIGURATION REGISTER #4			
BIT 2	BIT 1	BIT 0	TEST MODE SELECTION
0	0	0	NORMAL OPERATION (TEST MODE NOT ENABLED)
0	0	1	1553 DECODER SECTION
0	1	0	1553 ENCODER SECTION
0	1	1	PROTOCOL SECTION
1	0	0	FAIL SAFE TIMER (668µs)
1	0	1	REGISTER TESTING
1	1	0	MEMORY TESTING
1	1	1	GENERAL TEST MODE

When testing this section, the RT mode of the protocol sequencer should not be enabled in order to prevent any response on the 1553 bus. The simple word monitor could be enabled and then all words transmitted would be stored in memory along with the identification word for later examination. The only timing requirement for the CPU is that it must be able to update the control register at least once every 20 µs. If it is unable to do this. then the last word transmitted would be repeated until it can write the control register, or a transmitter timeout occurred. On those processors with a fast transfer rate, the output signals that normally feed the protocol section could be monitored to maintain synchronization with the encoder. Most current microprocessors should be able to maintain the pace of the encoder self-test.

PROTOCOL TESTING

When the last three bits of Configuration Register #4 are set to 011, then protocol testing has been enabled. Control signals and data that normally come from the 1553 decoder and encoder sections now come from test registers.

Register location 16 is used to supply the parallel data word that would have been received by the decoder section from the 1553 bus. Register location 17 is used to supply the control information that would have been supplied by the decoder and the encoder sections.

The actual encoder section is kept in reset to prevent transmission on the 1553 bus. All other functions are still operational. That is, register updates and data transfers still function normally.

By writing these to two register locations, the user can exercise the complete operation of the protocol sequencer. It takes only three data transfers to simulate the reception of any 1553 word as compared to the 20 µs required if it was received from the 1553 bus.

Test register bits are available to allow the CPU to read the control signals and data the protocol section supplies to the decoder and encoder sections.

The user can now verify complete functional operation of the protocol sequencer without going through the 1553 bus. He uses his host processor data bus. Timing is not critical. The only thing the user cannot simulate is a superseding command in RT mode that occurs during a data transfer since the CPU cannot get access to the data bus.

He can simulate Manchester encoding and parity

errors, incorrect sync types, high and low word counts, no response timeout, and superseding commands that do not occur during the data transfer window.

FAILSAFE TIMER TEST

When the last three bits in Configuation Register #4 are set to 100, then the FailSafe Timer test mode has been enabled. This timer has been built into the encoder section, as required by MIL-STD-1553, to prevent transmission on the 1553 bus greater than 800 µs.

This test blocks the signal from the protocol logic that normally stops the encoder from transmitting any more words on the 1553 bus. Therefore, once the Encoder section has started to transmit, it will continue to transmit until it shuts itself down at approximately 668 µs. This is a real time simulation that was intended to be used for the purpose of system characterization testing or 1553 validation testing to prove the existence of the timer. It is not required to actually verify the timer operation.

REGISTER TEST

When the last three bits of Configuration Register #4 are set to 101, the Register Test mode has been selected. This mode was added to provide easier test access to certain registers and counters. For example, most of the internal counters are followed by a latch to prevent the data from changing (due to the counter incrementing) during a CPU read cycle. The CPU is now able, through the test registers, to keep these signals latched and verify their operation.

Also, the Interrupt Status register can be written in test mode to verify the proper operation of the Interrupt Mask Register and associated logic without having to go through all the normal protocol sequences necessary to generate the actual interrupts.

For the BU-65620 (digital monolithic) and BU-61585/86 (with 8K X 17 additional RAM) versions of the ACE involving the use of 17-bit buffered RAM with the RAM parity function enabled, the CPU can force incorrect (even) parity to be written and verify the proper operation of the parity checker.

In addition, hooks are available to speed up the internal counters to reduce the test time required.

MEMORY TEST

When the last three bits in Configuation Register #4 are set to 110, then the memory test mode has been enabled. In this configuration, the CPU is able to write all 4K words of RAM by means of 1K write accesses.

This is possible because the internal memory has been partitioned into four 1K X 16 blocks. Normally, to verify memory operation, the CPU does not require any additional hooks but would require a lot of read and write operations. Patterns would have to be loaded in and then read back to verify the memory operation. All bits should be toggled and isolated from adjacent bits. A walking pattern of ones surrounded by all zeroes seems to be the most efficient pattern to use and still yield a high probability of confidence. Based on the memory core structure, we recommend a minimum of six patterns to be used.

Without TEST MODE, this would have require

48,752 data transfers. With TEST MODE, only 30,728 data transfer are required.

TEST MODE

When the last three bits of Configuration Register #4 are programmed to 111, the general test mode has been enabled. This feature is not required to perform a complete functional test of the ACE, but was left in so that we, the vendor, could utilize some of the key strobes during normal operations if it became necessary.

CONCLUSION

In conclusion, there are tremendous advantages to using the built-in test mode features of the ACE. The only additional cost to the user is the cost of the PROM to store the test program. In return, the user gets improved test time, a more comprehensive self-test which can be performed at any time, and does not require a 1553 bus simulator/tester.

FACTORS TO CONSIDER WHEN USING A PC IN A MIL-STD-1553 ENVIRONMENT

WHY USE A PC?

VME, MULTI BUS, UNIBUS — these are the systems that are typically considered when choosing a platform for 1553 development. However, it is well worth a system designer's time to consider the use of the IBM PC or compatible. One reason is the amount of software that is available for the PC. Just about any language, from assembler through Pascal and LISP can be hosted on a PC.

Another reason is the large number of interface cards that are available for the PC. There are cards available to allow the PC to communicate over Ethernet, IEEE-488 and VME busses, to name just a few. There are also cards already available to allow the PC to communicate over the 1553 bus.

But, you're asking yourself, "what makes the PC different than, say, the Multibus system? Many of these same caveats can be equally applied to an established system such as that one." One factor that allows the PC to stand out is the fact that it has so many other uses in an engineer's day. It can be used for word processing, electronic mail, etc.

EFFICIENT USE OF COMPANY RESOURCES

In fact, it's quite possible that your engineers already have access to PCs - possibly right on their desks. (Can you say that about a Multibus system?) This means that existing company resources can be used

for the 1553 development activity. This lowers initial costs and allows you to begin work immediately without having to wait for the delivery of a VME or Multibus system. It also reduces the problems associated with time-slicing engineers onto a scarce resource (such as a VME development system).

Another reason is the PC comes in so many easily acquired configurations. Ruggedized versions are available, making the PC ideal for such applications as "suitcase" type testers. Single board versions are available to allow you to meet even more stringent packaging requirements.

A world of peripherals is available for the PC. The system designer can choose the graphics card which provides the best human interface for a particular application. This can vary from a standard monochrome character display, to a high resolution Enhanced Graphics Adapter (EGA) system, to a flat plasma display for highly portable systems. Software packages are available to allow the user to develop highly sophisticated graphic interfaces in a fraction of the time it would otherwise take.

A RAM disk architecture is supported by DOS 3.0 and higher. This allows you to simulate a disk drive with RAM. It dramatically decreases the time the PC spends performing data retrieval. (It's a lot faster to retrieve data from RAM than from a mechanical floppy or hard disk.) The PC's system memory can be used, or special purpose RAM disk cards can be purchased.

MAKE VS. BUY DECISION

So far we've only talked about the options that can be purchased for a PC based system. What about designing your own custom card? The IBM PC has an "Open Architecture" which facilitates the design of custom cards. Eight-bit and 16-bit slots are available for these peripheral cards.

The 8-bit slots are "XT" compatible. In addition to performing data transfers a byte at a time, these slots only decode address bits A0-A19. The 16 bit slots are AT compatible, and they decode an extra 4 bits of addressing, from A0-A23. It is important that the system designer takes this into account when configuring the system. For instance, a board which decodes 20 bits of addressing that sits at address C000 will crash with a board which decodes 23 bits and sits at address 1C000. Many graphics boards are memory mapped only up to the 1 M byte boundary, so the entire system, including purchased peripherals, must be carefully configured.

The use of the memory mapped architecture provided by the PC allows the user to simplify the software development and integration of the PC based system. A memory mapped card appears to the application program as RAM. Normal read/write statements can be used to access the card — no hardware driver is necessary.

This means that if there is a problem, there are only two variables: the board could be bad, or there is a problem with the user's application software. There's no mysterious piece of code sitting between the application code and the board. (To make things even more complicated, the driver is usually not written by the person who is writing the application program. It's often not even written by someone in the same company, and is usually written in assembly language. This makes it very difficult to debug the system.)

Another advantage of a memory mapped card is that application software can be written and tested without the card being installed. (Simply assign the starting address of the program to an existing area of system memory.) This allows the engineer to begin development while the custom card is still being designed, or while waiting for the card to be delivered from the vendor.

EXAMPLE OF A MEMORY MAPPED PC TO 1553 INTERFACE CARD

The BUS-65515 is an example of a memory mapped

card that allows the PC to act as either a 1553 Bus Controller (BC), Remote Terminal Unit (RTU) or Bus Monitor (MT). Since it is memory mapped, the engineer utilizes normal read/write statements from Pascal, Fortran, even BASIC to load the 1553 messages into the board's onboard RAM.

The engineer loads either a stack or a Look Up table with the addresses of the messages. If the card has been programmed by the user to act as a BC, the engineer loads a message count into a reserved word of the onboard RAM. The user issues a start command by setting a particular bit in another reserved word, and the 1553 transmission begins.

In BC mode, the board will automatically issue commands over the 1553 bus, and interrogate the response from the RTU. In RTU mode, it responds to commands containing the boards' user-programmed RTU address. The card automatically transfers data to/from the onboard RAM, without requiring any overhead from the CPU. Upon completion of a 1553 message, a user maskable interrupt is generated to the PC. Other interrupts are also available to inform the PC of extraordinary events, such as error conditions.

INTERRUPT HANDLING

An elegant feature of the PC is its interrupt handling capability. The PC utilizes the INTEL 8259A Programmable Interrupt Controller. It handles eight interrupts, with IR0 having the highest priority. IR2 is typically used for specialized interface cards. The system designer defines the interrupt routine for the particular interrupt by loading it's starting address in the location 4* Interrupt number.

The PC has reserved six of these interrupt levels for itself, leaving IR2 for the user. This is generally not a problem, as each interrupt line can support multiple devices. However, the interrupt handler must then poll each device sharing the interrupt to determine which unit requested service.

If this proves to be a hindrance for very time critical events, the user can take over one of the "reserved" interrupt lines. For instance, DOS uses a discrete interrupt to indicate that a key has been pressed on the keyboard. The user can take over this interrupt for the custom board, and poll the keyboard using the Real Time Clock interrupt. Since most engineers cannot type faster than 18.2 keystrokes a second (the polling rate using the real time clock), there will be no noticeable difference in the human interface.

TYPICAL 1553 APPLICATION: ATE SYSTEM

The availability of IEEE-488 interface cards, plus the computing power of the PC itself, makes the PC an ideal platform for 1553 ATE applications. An example of this is the BUS-69005 software package. This software, written in Turbo Pascal, generates and runs the RTU Production Test Plan (PTP). It characterizes the particular RTU through user friendly menus. Once the capability of the particular RTU has been logged, the program automatically generates the IEEE commands necessary to run the RTU PTP. The user may run the entire test plan, or any subset. Low cost test equipment is used to implement the most economical solution.

Ruggedized and portable PC's can be used to implement suitcase tester applications. A truly portable ATE system can be developed using a single board or portable PC and a 1553 interface card with error injection capabilities, such as the BUS-65517. This card can also be used in a full PC to simulate the entire 1553 bus.

TYPICAL 1553 APPLICATION: SIMULATION/ TESTING

The BUS-65517 allows the PC to simultaneously act as a 1553 BC, multiple RTU's and an intelligent 1553 message monitor. Error injection capability is supported in both BC and RTU modes. Each message may be programmed with a different type of error condition. These include bit and word count errors, response timing errors, no responses, Manchester, parity and encoding errors.

Thus, the user can simulate an entire 1553 system, using just one card and one PC. The board can also be used as a tester, to evaluate an external BC or RTU. The intelligent message monitor reconstructs the 1553 traffic, indicating error conditions where applicable.

User friendly menus allow the engineer to quickly "get up and go" with the BUS-65517. The BUS65517 is memory mapped, thus the application software can be run without the board being installed. This allows the user to become familiar with the system and begin development work while waiting for the card to be delivered.

CONCLUSION

Due to the wealth of the software and hardware options that are available for the PC, it is well worth the system designer's time to consider the use of a PC in a 1553 application. The open architecture and interrupt handling available facilitates the use of custom cards. The use of memory mapping eases software development, test and interrogation. The availability of 1553 interface cards and software packages for the PC allows the system designer to mix and match available products with custom designed boards and software. The low cost of PC's reduces the problem of providing enough development tools for the project, as existing PC's can be used. Finally, the human interface can be tailored to utilize either highly sophisticated graphics, or simple monochrome screens, depending on what suits the particular application. The PC is definitely a contender for a MIL-STD-1553 CPU.

APPLICATION NOTES

The following Application Notes are available. You may:

Call us at: (631) 567-5600 ext. 7633

Fax us at: (631) 567-3402

E-mail us at: Info@ddc-web.com

Send your request on company letterhead to:

Data Device Corporation
105 Wilbur Place
Bohemia, N.Y. 11716-2482
Attention: Literature Department

COMMERCIAL AEROSPACE:

Order Number	Title
AN/A-1	Redundant Discrete Interface Devices.
AN/A-3	ARINC 429 Microprocessor Interface.
AN/A-4	Avionics Data Bus Basics.
AN/A-5	DDC's DD-03282 and the HS-3282 ARINC 429 Transceivers: A Technical Comparison
AN/A-6	DD-42900 Frequently Asked Questions

LINEAR DATA CONVERTERS:

Order Number	Title
AN/L-20	Analog/Digital Dynamic Testing Of High-Speed A/D.
AN/L-21	Noise Power Ratio Characterizes A/D Performance For Communications.
AN/L-22	Reconstruct with the ADC-00300-605 Evaluation Board.

MIL-STD-1553 DATA BUS:

Order Number	Title
AN/B-1	Designing a Remote Terminal to Talk with a MIL-STD-1553 Network.
AN/B-3	MIL-STD-1553 B: Military Standard for Avionics Integration.
AN/B-7	Avoiding Pitfalls in MIL-STD-1553 Transceivers.
AN/B-8	A Standard Local Network for Industrial Use.
AN/B-12	Test Whether a Noise Source is Gaussian.
AN/B-13	Hybrid Modules Help You Implement a 1553B Interface.
AN/B-14	Processor Interfaces to 1553 Buses.
AN/B-17	8088 Microprocessor to BUS-61553 Interface.

APPLICATION NOTES

(CONTINUED)

MIL-STD-1553 DATA BUS:

Order
Number   **Title**

AN/B-19   Upgrading from the BUS-61553 AIM-HY to the BUS-61559 AIM-HYer.
AN/B-20   Multi-Protocol RT with Shared RAM Interface and Memory Management.
AN/B-21   Stanag 3910 NATO Standard for 20MHz Fiber Bus.
AN/B-22   Systems Advantages of the Integrated 1553 Terminal.
AN/B-23   Processor-to-ACE Interfaces.
AN/B-24   ACE RT Memory Management Options.
AN/B-25   Simple Interfaces to Simple Systems.
AN/B-26   Avoiding Pitfalls Using Your MIL-STD-1553 Card.
AN/B-27   Electrical and Layout Considerations for 1553 Terminal Design.
AN/B-28   Understanding the ACE's Self-Test Capabilities.

POWER HYBRIDS:

Order
Number   **Title**

AN/H-3   Universal 3-Phase Drive for Closed-Loop Speed Control.
AN/H-4   Smart Power H-Bridges for Speed and Torque Control of DC Brush Motors.
AN/H-5   PW-82350 Motor Drive Power Supply Capacitor Selection

SOLID-STATE POWER CONTROLLERS:

Order
Number   **Title**

AN/P-1   Hybrid Solid-State Power Controller Outperforms Conventional Circuit Breakers.
AN/P-2   Solid-State Power Controllers.
AN/P-3   Solid-State Power Controllers Meet System Demands.

SYNCHRO CONVERTERS:

Order
Number   **Title**

AN/S-16   Let a Modern R/D Converter Simplify Your Design Effort!

X.
DDC TO DESC PART NUMBERS

MIL-STD-1553 DATA BUS PRODUCTS
DDC to DESC Part Numbers

The majority of DDC's products are DESC certified. New members of each product family become certified on a continuing basis.

The Table below lists the first member of each products family. The associated Mil-DWG lists all family members that are currently DESC certified with their respective DESC part number.

DDC PART NUMBER	MIL-DWG (DESC) NUMBER
BUS-61553-110	5962-8869201XC
BUS-61554-110	5962-8869202XC
BUS-61555-110	5962-8869203XC
BUS-61556-110	5962-8869204XC
BUS-61559-110	5962-9173401-HXC
BUS-61560-110	5962-9173402-HXC
BUS-61563-110	5962-8869201TC
BUS-61564-110	5962-8869202TC
BUS-61565-110	5962-8869203YC
BUS-61569-110	5962-9173401HZC
BUS-61570-110	5962-9173402HZC
BU-61580-S3-110	5962-9306511-HXC
BU-61580-S6-110	5962-9306512-HXC
BU-61580-V3-110	5962-9306511-HYC
BU-61580-V6-110	5962-9306512-HYC
BU-61580-S1-110	5962-9306503-HXC
BU-61580-S2-110	5962-9306504-HXC
BU-61580-V1-110	5962-9306503-HYC
BU-61580-V2-110	5962-9306504-HYC
BU-61581-S1-110	5962-9306507-HXC
BU-61581-S2-110	5962-9306508-HXC
BU-61581-S3-110	5962-9306515-HXC
BU-61581-S6-110	5962-9306516-HXC
BU-61581-V1-110	5962-9306507HYC
BU-61581-V2-110	5962-9306508-HYC
BU-61581-V3-110	5962-9306515HYC
BU-61581-V6-110	5962-9306516-HYC
BU-61582-D0-110	5962-9688705-HXC
BU-61582-D1-110	5962-9688701-HXC
BU-61582-D2-110	5962-9688702-HXC
BU-61582-F0-110	5962-9688705-HYC
BU-61582-F1-110	5962-9688701-HYC
BU-61582-F2-110	5962-9688702-HYC
BU-61582-G0-110	5962-9688705-HZC
BU-61582-G1-110	5962-9688701-HZC
BU-61582-G2-110	5962-9688702-HZC
BU-61583-D0-110	5962-9688706-HXC
BU-61583-D1-110	5962-9688703-HXC
BU-61583-D2-110	5962-9688704-HXC
BU-61583-F0-110	5962-9688706-HYC
BU-61583-F1-110	5962-9688703-HYC
BU-61583-F2-110	5962-9688704-HYC
BU-61583-G0-110	5962-9688706-HZC
BU-61583-G1-110	5962-9688703-HZC
BU-61583-G2-110	5962-9688704-HZC
BU-61585S1-110	5962-9306517-HXC
BU-61585S2-110	5962-9306518-HXC
BU-61585S3-110	5962-9306521-HXC

DDC PART NUMBER	MIL-DWG (DESC) NUMBER
BU-61585S6-110	5962-9306522-HXC
BU-61585V1-110	5962-9306517-HYC
BU-61585V2-110	5962-93065218-HYC
BU-61585V3-110	5962-9306521-HYC
BU-61585V6-110	5962-9306522-HYC
BU-61586S1-110	5962-9306519-HXC
BU-61586S2-110	5962-9306520-HXC
BU-61586S3-110	5962-9306523-HXC
BU-61586S6-110	5962-9306524-HXC
BU-61586V1-110	5962-9306519-HYC
BU-61586V2-110	5962-9306520-HYC
BU-61586V3-110	5962-9306523-HYC
BU-61586V6-110	5962-9306524-HYC
BUS-63105-110	5962-8604902ZC
BUS-63106-110	5962-8604902TC
BUS-63125-110	5962-8757902XC
BUS-63126-110	5962-8757902YC
BUS-63127-110	5962-8982601XC
BUS-63128-110	5962-8982601YC
BUS-63135-110	5962-8944704HXC
BUS-63136-110	5962-8944704HYC
BUS-65112-110	5962-8753501XC
BUS-65117-110	5962-8753501YC
BUS-65142-110	5962-8979801-XC
BUS-65143-110	5962-8979802XC
BUS-65144-110	5962-8979801YC
BUS-65145-110	5962-8979802YC
BUS-65153-110	5962-9216201HXC
BUS-65154-110	5962-9216202HXC
BUS-65163-110	5962-9216201HYC
BUS-65164-110	5962-9216202HYC
BU-65170S1-110	5962-9306501-HXC
BU-65170S2-110	5962-9306502-HXC
BU-65170S3-110	5962-9306509-HXC
BU-65170S6-110	5962-9306510-HXC
BU-65170V1-110	5962-9306501-HYC
BU-65170V2-110	5962-9306502-HYC
BU-65170V3-110	5962-9306509-HYC
BU-65170V6-110	5962-9306510-HYC
BU-65171S1-110	5962-9306505-HXC
BU-65171S2-110	5962-9306506-HXC
BU-65171S3-110	5962-9306513-HXC
BU-65171S6-110	5962-9306514-HXC
BU-65171V1-110	5962-9306505-HYC
BU-65171V2-110	5962-9306506-HYC
BU-65171V3-110	5962-9306513-HYC
BU-65171V6-110	5962-9306514-HYC
BUS-66300-110	5962-8858601XC
BUS-66301-110	5962-8858601YC

XI.
INDEX