OPEN ACCESS

International Journal Of Modern Engineering Research (IJMER)

Implementation of MIL-STD-1553 Data Bus

Elizabeth Sunny[1]

1(Federal Institute of Science and Technology, India)

Abstract: *MIL-tSTD-1553 is a military standard that defines the electrical and protocol characteristics for a data bus. This standard defines requirements for digital and time division multiplexing techniques for a one Megahertz serial data bus and specifies its interface electronics. The effective bandwidth of the systems using this standard can be improved by various data compression techniques. Run-length encoding is one of the lossless data compression techniques suitable for implementation in real-time and mission critical systems. This paper describes the better utilization of the available bandwidth using Run-length encoding algorithm.*

Keywords: *data compression, MIL-STD-1553, run-length encoding, serial data bus, time-division - Multiplexing.*

I. Introduction

MIL-STD-1553 was developed from the growing complexity of avionics systems and the subsequent increase in the interconnections between terminal devices. In the 1950s and 1960s, the navigation, communications, flight controls, and displays consisted of analog systems. Most of these systems were composed of multiple subsystems, connected to form a single system [1]. Various subsystems were connected with point-to-point wiring. As more and more systems were added, the cockpits became more crowded, and the overall weight of the aircraft increased. As time and technology progressed, a data transmission medium, which would allow all systems and subsystems to share a single and common set of wires, came in to existence. By sharing the use of this interconnects, the various subsystems could send data between themselves and to other systems, one at a time, and in a defined sequence, hence a data bus. The first draft of a standard in 1968, by the Aerospace Branch of the Society of Automotive Engineers laid the foundation for the US Air Force's adoption of MIL-STD-1553 in 1973 [5].

Researches on the standard have shown that a significant amount of the data sent over the MIL-STD-1553 data bus during any time slice carries no new information. In such a case, the effective bandwidth of systems using the MIL-STD-1553 serial data bus can be improved by various data compression techniques. Compression may be implemented through many techniques that result in either lossy or lossless compression. If lossless compression techniques are employed, no information is lost as part of the compression and decompression process. Run-length encoding is one such type which takes advantage of situations where the data contains relatively long runs of consecutive identical numbers [2].

II. Bus Architecture

The 1553 data bus is a dual-redundant, bidirectional and Manchester II encoded data bus with a high bit error reliability. The bus architecture consists of remote terminals, bus controllers and bus monitors.Remote terminals are defined within the standard as, all terminals not operating as the bus controller or as a bus monitor. A remote terminal typically consists of a transceiver, an encoder/decoder, a protocol controller, a buffer or memory, and a subsystem interface. The bus controller is the terminal that initiates information transfers on the data bus. All bus communications are controlled and initiated by a bus controller. It sends commands to the remote terminals which reply with a response. The bus will support multiple controllers, but only one may be active at a time. It is the key part of the data bus system, and the sole control of information transmission on the bus shall reside with the bus controller. The bus controller is responsible for directing the flow of data on the data bus. The bus controller is the only one allowed to issue commands onto the data bus [4].

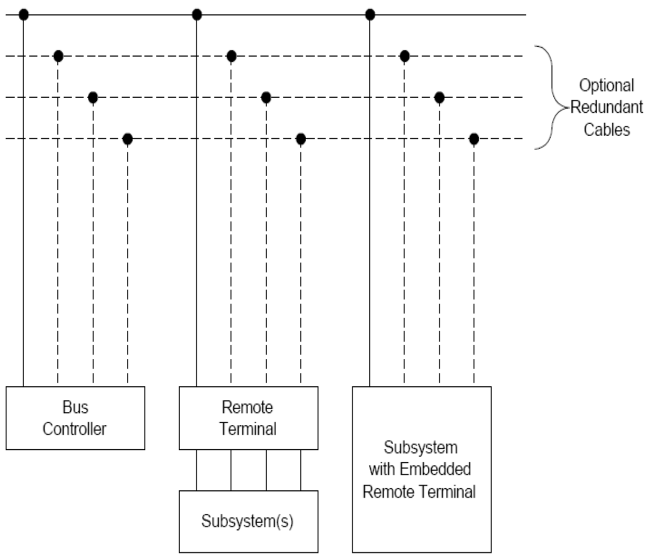

Fig. 1 Bus Architecture

A bus monitor is a terminal that listens to the exchange of information on the data bus. It may collect all the data from the bus or may collect selected data. It is the terminal that has assigned the task of receiving bus traffic and extracting selected information to be used at a later time.

III. WORD TYPES

The control, data flow and management of the bus are provided by the word types .There are three word types, Command word, Data word and Status word. Each word is twenty bits in length. The first three bits are used as a synchronization field. The next sixteen bits are the information field and are different between the three word types. The last bit is the parity bit and is based on odd parity for the single word.

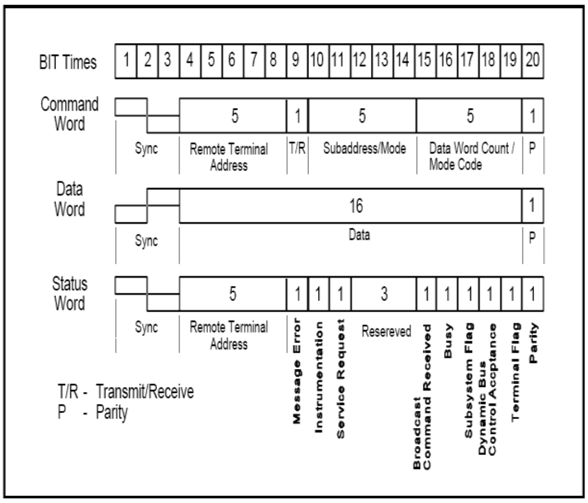

Fig. 2 Word formats

1. Command Word

The Command Word (CW) specifies the function that remote terminal is to perform. Only the active bus controller transmits this word. The word begins with command sync in the first three bit times followed by sixteen-bit information field and a parity bit. The sync field of command word and the status word are identical and the inverse of the data word sync pattern. Therefore, command and status words, can always be distinguished from data word sync patterns.

2. Data Word

The Data Word (DW) contains the actual information that is being transferred within a message. The first three-bit time contains a data sync. This sync pattern is the opposite of that used for command and status words and therefore is unique to the word type. Data words can be transmitted by either a remote terminal (transit command) or a bus controller (receive command).

3. Status Word

A remote terminal in response to a valid message transmits only the status word (SW). The status word is used to convey to the bus controller whether a message was properly received or to convey the state of the remote terminal (i.e., service request, busy, etc.).The optional status bits are, instrumentation, service request, broadcast command received, busy, subsystem flag, dynamic bus control acceptance and terminal flag.

IV. MESSAGE FORMATS

The standard defines ten types of message transmission formats. The message formats have been divided into two groups. These are referred to within the standard as the "information transfer formats" and the "broadcast information transfer formats". The information transfer formats are based on the command/response philosophy in that all error free transmissions received by a remote terminal are followed by the transmission of a status word from the terminal to the bus controller. This handshaking principle validates the receipt of the message by the remote terminal.

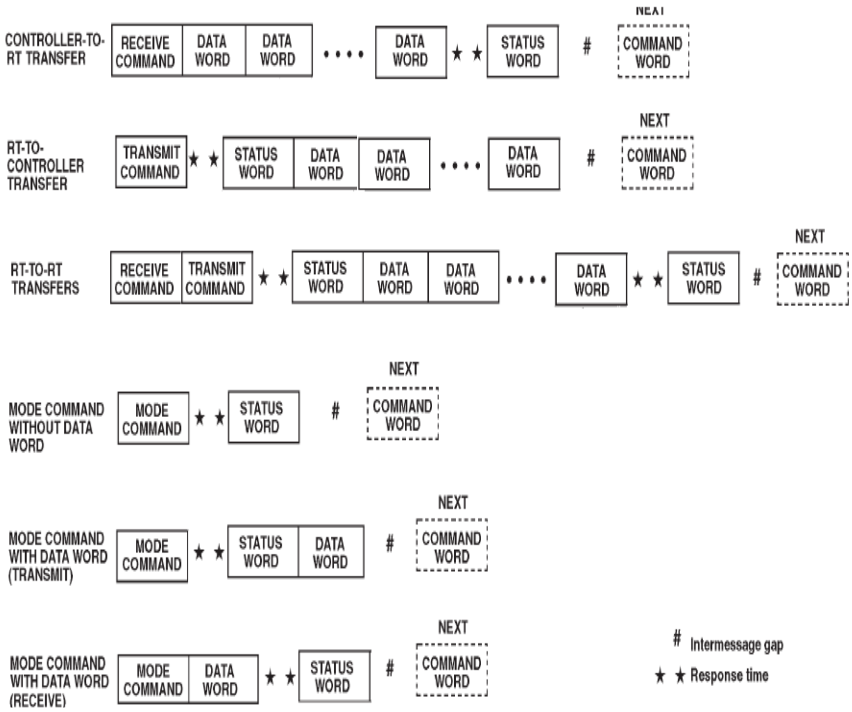

Fig. 3 Information transfer formats

Broadcast messages are transmitted to multiple remote terminals at the same time. The terminals suppress the transmission of their status words. In order for the bus controller to determine if a terminal received the message, a polling sequence to each terminal must be initiated to collect the status words. The broadcast option can be used with the message formats where the remote terminal receives data. Obviously, multiple terminals cannot transmit data at the same time, so the RT-BC transfer format and the transmit mode code with data format cannot be used. The broadcast RT-RT allows the bus controller to instruct all remote terminals to receive and then instructs one terminal to transmit, thereby allowing a single subsystem to transfer its data directly to multiple users.

1. Bus Controller to Remote Terminal

The bus controller to remote terminal (BC-RT) message is referred to as the receive command since the remote terminal is going to receive data. The bus controller outputs a command word to the terminal defining the Sub address of the data and the number of data words it is sending. Immediately (without any gap in the transmission), the number of data words specified in the command word is sent. The remote terminal upon validating the command word and all of the data words issues its status word within the response time requirements .The remote terminal must be capable of processing the next command that the bus controller issues.

2. Remote Terminal to Bus Controller

The remote terminal to bus controller (RT-BC) message is referred to as a transmit command. The bus controller issues only a transmit command word to the remote terminal. The terminal, on validating the command word, transmits its status word followed by the number of data words requested by the command word. The remote terminal doesn't know the sequence of commands to be sent and doesn't normally operate on a command until the command word has been validated.

3. Remote Terminal to Remote Terminal

The remote terminal to remote terminal (RT-RT) command allows a terminal to transfer data directly to another terminal without going through the bus controller. However, the bus controller may also collect the data and use them. The bus controller issues a command word to the receiving terminal immediately followed by a command word to the transmitting terminal. The receiving terminal is expecting data, but instead of data after the command word it sees a command sync (the second command word). The receiving terminal ignores this word and waits for a word with data sync.

4. Mode Command

Three mode command formats are provided. This allows for mode commands with no data words and for the mode commands with one data word (either transmitted or received). The status/data sequencing is the same as the BC-RT or RT-BC messages except that the data word count is either one or zero.

V. DATA COMPRESSION

Data compression has many benefits depending on its various applications. A few of these advantages include reduced time for communication, reduced transmission errors, and increased security. For a serial communication system, data compression allows greater bus utilization by allowing more data to be sent during a given transmission time interval, reducing the transmission time. Compression may be implemented through many techniques which is lossy or lossless compression. Lossy compression reduces the data size by identifying data that carries the least information and removing it away. This means that when the data is uncompressed, the result will not be identical to the original pre-compressed data, resulting in a loss of information. Lossy compressing routines are frequently used for picture and video compression. Examples of lossy compression are JPEG and MPEG compression. Lossless compression reduces redundancy in input data and is reversible resulting in decoded data that is identical to its pre-compressed form. Lossless compression is necessary for critical systems such as avionics.

Run-length encoding is a suitable lossless compression technique [2]. It takes advantage of situations where the data contains relatively long runs of consecutive identical numbers. In this scheme, a position-indicating word is used along with data words. The encoded data consists of the position-indicating word followed by non-zero data words listed in the original order of their appearance. A zero bit in the position-indicating word implies that the value does not repeat. A one indicates that the value repeats. When the position bit with a value of zero is encountered, the data word is read from the data section of the block. After that, if value one is encountered this value is repeated as the consecutive data word. At the decoder if the position bit with a value zero is read , the data words following the position-indicating word are taken in the original order of their appearance. If a bit

One is found, the previous data word is repeated. The TABLE 1 shown below is an example for the above scheme.

Table 1 Run-length encoding scheme

Word count (Hex)	Input Data (Hex)	Position-indicating Word	Encoded Data (Hex)
0	0	0	6BDF
1	0	1	0
2	0	1	FFFF
3	FFFF	0	1204
4	FFFF	1	5342
5	1204	0	
6	1204	1	
7	1204	1	
8	1204	1	
9	1204	1	
A	5342	0	
B	5342	1	
C	5342	1	
D	5342	1	
E	5342	1	
F	5342	1	

VI. RESULTS AND DISCUSSION

The software designing of the data bus is done in VHDL using Xilinx 1SE design suit 14.2 and the simulation is done using Xilinx ISE Simulator. Referring to the example shown in TABLE 1 instead of transmitting the entire 16 data words, we are transmitting 5 Run-length encoded data words .The simulation results shows that the time taken to encode and transmit the data words can be reduced effectively and the effective utilization of the bandwidth during data transmission is improved.

VII. CONCLUSION

MIL-STD-1553 is a military standard which is used as one of the basic tool by the Department of Defense for the integration of weapon systems. The serial data bus has solved the complex scenario of a crowded system and finds application in the areas of military aircrafts, submarines, tanks and launch vehicles. Since the amount of the data sent over the data bus during any time slice carries no new information, the paper describes a data compression technique called Run-length encoding. Comparing the time taken to encode and transmit the data words with compression and without compression shows that the above method is a better option to be used for the effective utilization of the band-width.

REFERENCES

[1.] Duren, Russell W, June 2006, “Options for upgrading legacy avionics systems,” Journal of Aerospace Computing, Information, and Communication, vol. 3, pp. 251-259

[2.] Duren, Russell W. and Michael W. Thompson, 1-8 March 2008, Application of Data Compression to the MIL-STD-1553 Data Bus, Proc. of the 2008 IEEE Aerospace Conf., Big Sky, MT.

[3.] Duren, Russell W, Algorithmic and architectural methods for performance enhancement of avionics systems, Proc. of the 28th Digital Avionics Systems Conf., October 25-29, 2009

[4.] Data Device Corporation, “ MIL-STD-1553 designer’s guide ” , sixth edition .

[5.] E. C. Gangl, “Evolution from analog to digital integration in aircraft avionics - a time of transition,”Aerospace and Electronic Systems, IEEE Transactions on, vol. 42, 1163-1170, 2006.