INTERFACING NAVIGATION AND FLIGHT CONTROL COMPUTER USING MIL STD 1553B

Dr. Murali Kante
HOD, Department of Electronics and Communication Engineering (JNTUA)
Narayana engineering college (JNTUA)
Nellore , India
hodece@necn.ac.in

Mahija Neelapareddy
Department of Electronics and Communication Engineering (JNTUA)
Narayana engineering college (JNTUA)
Nellore , India
nmahija16@gmail.com

Divya Anantha
Department of Electronics and Communication Engineering (JNTUA)
Narayana engineering college (JNTUA)
Nellore , India
ananthadivya49@gmail.com

ABSTRACT **-This paper aims to establish the communication between navigation computer(NC) and flight control communication (FCC) using popular MIL STD 1553B protocol. Flight control computer is the safety critical system and executes Longitudinal, lateral-directional control laws for manual/automatic takeoff and Landing and the control outputs are sent to various actuators. NC senses the aircraft attitude, rates, position and velocity. The data from navigation computer is used by flight control computer to control and navigate the aircraft so that aircraft can complete the desired mission. The paper starts with analyzing the interfacing requirement specifications to navigation computer and then developing the application software for 1553B Remote terminal (RT) module. 1553B Bus controller (BC) module application software for the flight control computer will be developed. The data integrity and communication channel checks between both navigation and flight control computer will be verified using portable avionics test system(PATS) which acts as bus monitor.**

***Keywords--MIL-STD-1553B, Bus Controller (BC), Remote Terminal (RT), Flight control , Navigation*.**

I.INTRODUCTION

MIL-STD-1553B is a military standard that defines the electrical and protocol characteristics for a data bus .A data bus is used to provide a medium for exchange of data and information between various system. It is widely used as an internal time division command/response multiplex data bus is for the military aircrafts. It has become one of the basic tools being used today by defence establishments for integration of weapon and aircraft systems.

This bus protocol includes various hardware characteristics such as bus controller (BC), remote terminal (RT), bus monitor (BM). The protocol, including the message formats, word types and command and status words which are responsible for the control, data flow, status reporting, and management of the bus. These bus connectors are available in standard (BNC size), *miniature* and sub-*miniature* size. The three Enhanced Mini-ACE chips are used each chips are used each has specific operation act as bus controller(BC), remote terminal(RT) and bus monitor respectively.

Navigation Computer is a GPS based Inertial Navigation System and it has got three High Accuracy RLG's and three Pendulum Accelerometers. It is interfaced with 1553B bus, for data exchange with external avionics packages. Navigation Computer expects two 1553B messages from BC. The first message issues a command to NC for altitude update, zero velocity update, and satellite hybridization. The second message issues command for navigation initialization parameter and selection of specific mode of navigation for NC.

This project aims to develop the application software for the bus controller and remote terminal which acts as flight control computer and navigation computer respectively. The software is developed using the Code warrior IDE.

II. MIL-STD-1553B

A. Architecture:

MIL-STD-1553B defines the data bus architecture in which a maximum of 32 devices can be connected to the 1553B data bus which is shown in Figure 1. It has a dual-redundant architecture and extremely low error rate of one word fault per 10 million words make MIL-STD-1553 a highly reliable bus**.** Only a single device is allowed to transmit on the bus at a given time. Terminal connected in the 1553B data bus has a master/slave relationship. The computer that controls all the communication on the bus and acts as the master is called Bus Controller (BC)[10]. In addition to initiating all data transfers, the BC must transmit, receive and coordinate the transfer of information on the data bus. All information is communicated in command/response mode, i.e., the BC sends a command to the RTs, which reply with a response. The BC can control multiple slave computers which are called Remote Terminals (RT) by sending commands to them. The remote terminal (RT) is a device designed to interface various subsystems with the 1553 data bus and provide the sources and sinks of data. There may also be one or more passive Bus Monitors (BM) deployed on the bus which are only used to monitor or record the messages on the bus but can't transmit any messages. As shown in Figure 1, a maximum of thirty-one remote terminals can be connected to the bus in addition to the bus controller [2]. Only the bus controller can initiate a transmission on the bus.

The RT receives and decodes commands from the bus controller and responds accordingly within the strictly defined time of 12 microseconds. An RT only performs transmission or reception of data when instructed to do so by the BC. The BC is responsible for directing communications over the data bus. Although several of the systems connected to the bus may have the ability to perform the role of BC. Only one BC is allowed to active at any given time and that only may issue commands over the data bus. These commands may be related to the management of the bus or may be oriented toward data communication to and from subsystem[10].

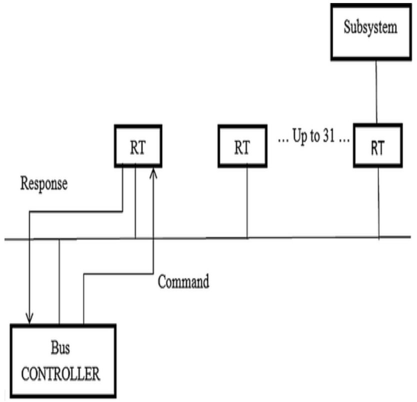

Fig.1 1553B bus control Architecture

B. WordFormats

There are three different words that form the messages that are transmitted on the bus; command word (CW), data word (DW) and status word (SW) . Each word is formed by a three-bit time sync, sixteen bits for the information field itself, and a parity bit at the end, which makes a total of twenty bits [5]. Fig. 2 shows an illustration of the three word formats. Each bit is timed as one microsecond, resulting in one megabit per second transmission rate for the bus[10]. The communications between two elements in the data bus system, the bus controller and the remote terminal is allowed only through information transfer formats of 1553B bus which is shown in Figure 2. In 1553, the bus controller controls 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. The actual information that is to be transferred through the 1553 bus is contained within the data word of a message [4]. It is transmitted by the bus controller after it sends a receive command or by the remote terminals after they receive a transmit command. The sixteen bits of payload of the data word is application specific and is defined by the interface designers. The standard only requires that the most significant bit (MSB) of the data must be transmitted first. The status of the remote terminal to the bus controller is indicated in the status word. Remote terminals must send a status word as the first word of a response to a valid message from bus controller[10]. The only time when a status word is suppressed is when the optional broadcast operation is performed.

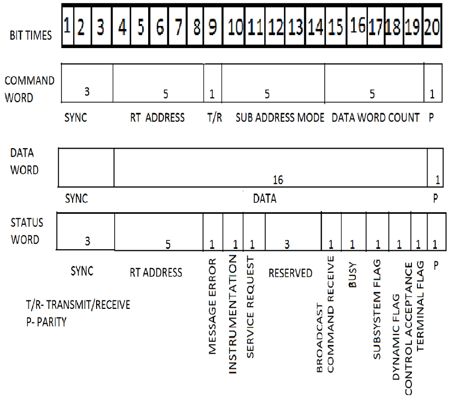

Fig 2. Word Formats of 1553B Protocol

C. Message Formats

The primary purpose of the 1553 data bus is to provide a common media for the exchange of data between systems. All control messages originate with the active bus controller and are received by single receiver or by multiple receivers (broadcast). A command word with a terminal address value of 31 (11111) indicates a broadcast message, while any other terminal addresses are to identify unique messages to a terminal on the bus. The exchange of data is based on message transmissions. A single message is the transmission of a command word, status word, and a data word if they are specified. Sole control of information transmission on the bus shall reside with the bus controller, which shall initiate all transmissions. The 1553 standard defines different types of information/message transmission formats [3]. All of these formats are based on the three word types just defined. Here we consider three important information formats which are shown in Fig.3. This 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. 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 [8]. The bus controller outputs a command word to the terminal defining the subaddress of the data and the number of data words it is sending[8]. The remote terminal upon validating the command word and all of the data words issues its status word within the response time requirements which is maximum of 12µs. 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[8]. 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 to remote terminal (RT-RT) command allows a terminal (the data source) to transfer data directly to another terminal (the data sink) without going through the bus controller[8]. However, the bus controller may also collect the data and use them. (This is sometimes called a RT-RT-M command, since the Bus Controller monitors the data). The bus controller issues a command word to the receiving terminal immediately followed by a command word to the transmitting terminal [5]. Here, the message shall include the two command words, the two status words, and the data words.

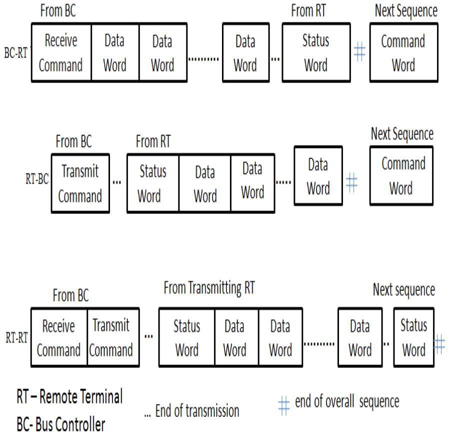

Fig.3 Message Formats of 1553B

III. PROPOSED SYSTEM

The proposed system makes use of a 1553B protocol for interfacing the navigation computer and flight controller for transmitting and receiving the messages. It makes use of an PCB board fabricated with three Enhanced mini ACE 1553B chips and MPC5554 microcontroller as the main components. The project aims to develop the software using the embedded C programming for bus controller and remote terminal in the Code warrior platform. The hardware PCB board is used to test the developed software. The main principle involved in 1553B protocol is master and slave relationship between the bus controller and remote terminals. It is composed of a bus controller and one or more remote terminals, connected by the 1553B serial data bus. To provide multiple data paths for redundancy, a mission-critical system typically utilizes several 1553B buses. All bus transmissions are accessible to all units connected to the bus, but only one unit can speak at a time. The BC initiates all bus transfers by sending a command word to individual RTs, and each RT is required to respond with a status message acknowledging receipt of an error free BC's message. The entire control of the bus system is incorporated in the bus controller. Thus the bus controller is said to be the heart of the system. The developed software is tested by using the portable avionics test system (PATS) which acts as bus monitor**.**

IV. HARDWARE BOARD

This section describes the hardware board use for the software implementation. The hardware board consists of Enhanced Mini ACE 1553B chips and MC554 as the Key components.

A. Enhanced Mini-ACE 1553B terminal

The Enhanced Mini-ACE family of MIL-STD-1553 terminals provides complete interfaces between a host processor and 1553 bus, integrating dual transceiver, protocol logic, memory management and 4K or 64K words of RAM.

The Enhanced Mini-ACE is available in square inch flat pack or gull wing package. The µ-ACE is available as a 0.815 square inch BGA package as shown in the Fig.4. These terminals are nearly 80% software compatible with the previous generation Mini-ACE and Mini-ACE Plus terminals with the original ACE series. These includes a 5V voltage source transceiver for improved line driving capability, for reducing power consumption, there are versions for which the logic is powered by 3.3V.

The salient feature of the Enhanced Mini-ACE/µ-ACE is its Enhanced bus controller architecture. The Enhanced BC's highly autonomous message sequence control engine provides a means for offloading the host processor for implementing multi-frame message scheduling, message retry schemes, data double buffering, and asynchronous message insertion. For the purpose of performing messaging to the host processor, the Enhanced BC mode includes a General Purpose Queue, along with user-defined interrupts. The another features includes the incorporation of a fully autonomous built-in self-test. This test provides comprehensive testing of the internal protocol logic. A separate test verifies the operation of the internal RAM. Since the self-tests are fully autonomous, they eliminate the need for the host to write and read stimulus and response vectors. The Enhanced Mini-ACE/µ-ACE RT offers the same choices of single, double, and circular buffering for individual sub addresses as ACE and Mini-ACE (Plus). New enhancements to the RT architecture include a global circular buffering option for multiple (or all) receive sub addresses, a 50% rollover interrupt for circular buffers, an interrupt status queue for logging up to 32 interrupt events, and an option to automatically initialize to RT mode with the Busy bit set. The interrupt status queue and 50% rollover interrupt features are also included as improvements to the Enhanced Mini-ACE/µ-ACE's Monitor architecture.

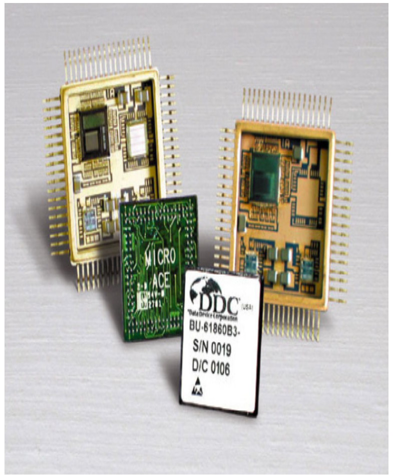

Fig.4 Enhanced Mini-ACE 1553B Terminals

B.MPC5554 Microcontroller:

The MPC5554 microcontroller (MCU) is a member of the MPC5500 family of microcontrollers built on the Power Architecture TM embedded technology. This family of parts has many new features coupled with high performance CMOS technology to provide substantial reduction of cost per feature and significant performance improvement over the MPC500 family. The host processor core of this device complies with the Power Architecture embedded category that is 80% user-mode compatible (including floating point library) with the original Power PCTM user instruction set architecture (UISA). The embedded architecture enhancements improve the performance in embedded applications. The core also has additional instructions, including digital signal processing (DSP) instructions, beyond the original Power PC instruction set. The MPC5554 has two levels of memory hierarchy. The fastest accesses are to the 32-kilobytes (KB) unified cache. The next level in the hierarchy contains the 64-KB on-chip internal SRAM and two-megabyte (MB) internal flash memory. The internal SRAM and flash memory hold instructions and data. The 416 package has 40-channels. The system integration unit (SIU) performs several chip-wide configuration functions. The less complex timer functions of the MPC5554 are performed by the enhanced modular input/output system (eMIOS).

V.SOFTWAREDEVELOPMENT

The software is developed for both Bus Controller and remote terminal which acts as master and slave respectively. The code warrior IDE is software tool used for the development. The source code is developed for memory read , memory write , register read and register write which are basic and common for the both BC and RT operation. Two source files are created one for storing the main program which is .C file and for storing the memory and register locations uses .h file.

A.Remote Terminal

The remote terminal software is used for the transmitting and receiving messages between the bus controller. There are thee RT-BC messages which acts navigation computer messages NC-01, NC-02,NC-03 as shown in Table I. The remote terminal act as device-1 uses the basic four functions. The RT software initialization procedure starts with start /reset register , interrupt mask register and then to configure six registers. The stack pointer register is initialize to zero .To configure each register function contains its device number, offset address and data to be initialized.

B.Bus Controller

The Bus Controller software is used for the transmitting and receiving messages between the remote terminal. It is operated in master mode in 1553B bus. Only one Bc is activated it sends messages to all remote terminals. The data transformed is in the form of frames major and minor frames. There are two BC-RT messages which acts FCC messages BC-01, BC-02 as shown in TableI. The Bus Controller act as device2 uses the basic four functions. The BC software initialization procedure starts with start /reset register , interrupt mask register and then to configure six registers. The stack pointer register is initialize to zero .To configure each register function contains its device number offset address and data to be initialized.

TABLE I

TRANSMIT AND RECEIVE MESSAGES IN MIL-STD-1553B

MESSAGE ID	MESSAGE	SOURCE	DESTINATION	WORDS	FREQUENCY	T/R_S/A
NC-01	PURE INTERTIAL	NC	FCC	24	80 Hz	Tx-01
NC-02	SATELLITE NAVIGATION	NC	FCC	20	1 Hz	Tx-02

NC-03	HYBRID NAVIGATION	NC	FCC	17	80 Hz	Tx-03
BC- 01	CONTROL	FCC	NC	4	12.5 Hz	Rx-01
BC-02	INSTALLATIO N	FCC	NC	8	12.5 Hz	Rx-02

C.Software Debugging

To debug the code uses three tools such as Comport 4.0 tool kit. Boot loader PC tool , Trace 32 ICD PPC USB. In the debugging process each tool has its specific operation. Comport tool is used to reset the system , serial connection is set and output is seen using it. Boot loader PC tool is set at the baud rate of 115200 and program is uploaded into the boot loader. Trace 32 USB port is used set required memory locations and base address and the output memory and register address can seen using it

D. Testing

The final software is tested by using portable avionics test system (PATS) which acts as bus monitor.

VI. RESULTS

The proposed system works in which bus controller acts as flight control computer and remote terminal acts as navigation computer which controls the data transfer through the bus. For BC to RT transfer a command word followed by data word is send by the BC. Upon receiving this, RT sends SW indicating an error free reception. The RT to BC transfer includes a command word which is send by the BC and upon receiving this Command word, RT sends status word followed by the data word.

VII. CONCLUSION AND FUTURE SCOPE

It can be concluded that the objectives of this project were achieved. MIL-STD-1553B remains relevant for military product upgrade programmes. MIL-BUS will also be relevant communication system in the next generation weapon system. Due to very long product life cycle on military we do not expect MIL-STD-1553B to be replaced in the coming decades.

VIII. REFERENCES

[1] Department of Defence, "MIL-STD-1553B, Aircraft internal time- division multiplexing data bus", Washington, D.C., 1978.

[2] Data Device Corporation, (2003,August) "MIL-STD-1553 Designer's guide", Bohemia, NY, 2003.

[3] Condor Engineering, "MIL-STD-1553 Tutorial" (1600100- 0028), 2000.

[4] Jemti Jose and Sharone Varghese, “Design of 1553 protocol controller for reliable data transfer in aircrafts”, In *Proceedings of the 12th International Conference on Intelligent Systems Design and Applications* (ISDA 2012), Cochin, India, pp 686- 691.

[5] MIL-STD-1553A *Multiplex Applications Handbook*, 1988, Washington, D.C., Department of Defense.

[6] J. Furgerson, "MIL-STD-1553 Tutorial", *AIM-Avionics Data Bus Solutions*, U.K, November 2010.

[8] Carey B. R (2007, Dec), Avionics magazine, [online]. Available: http://www.AvionicsToday.com

[9] D.R Bracknell, "Introduction to the MIL-STD-1553B serial multiplex data bus", Ministry of Defence, September 1988. [10]K.Siji and N.R saritha “FPGA Implementation of MIL-STD-1553B Bus Protocol Controller for Aircrafts”, for *International Journal of Science and Research*(IJSR,2015),India.