International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 4, April 2016

Design of Flex Ray Communication Controller for Automotive Communication System

[1] Neethu Varghese, [2] Abhila R Krishna
[1] PG Student [VLSI] [2] Assistant professor,
Department of ECE, TKM Institute of Technology, Kollam
[1] neethu5454@gmail.com [2] abhilaktvm@gmail.com

Abstract: **— the communication bandwidth, flexibility and bit rate has been a serious issue for the serial communication protocols like Controller Area Network (CAN), Time-Triggered CAN (TT-CAN) and Byte-flight in implementing real time applications. The lack of a suitable networking scheme that could handle the requirements such as more sophisticated safety systems and advanced networking application leads to the development of the new protocol, the Flex ray. The main target for the new protocol is the growing demand for massive and complex data exchange in the environment where real time requirements are critical. The Flex-Ray is a new time-triggered protocol which is designed to provide large bunches of data to be exchanged in real-time with high dependability between electronic control units, sensors and actuators installed in a vehicle .The Flex ray node consists of a host, communication controller and a bus driver in which the communication Controller (CC) is an electronic component in a node which is responsible for implementing the protocol aspects of the Flex ray communications system. Flex ray communication controller (CC) is the core of the Flex ray protocol specification. It performs all communication tasks such as transmission and reception of messages in a Time-Triggered protocol (TTP) cluster. The purpose of the the introduced system is to generate a Flex ray protocol frame and the design of Flex ray protocol operational states with Finite State Machine(FSM).The work is coded by Verilog and simulated by Xilinx ISE 13.2.**

Index Terms**—Communication Controller (CC), Electronic Control Unit (ECU), Finite State Machine (FSM), Flex ray**

I. INTRODUCTION

In recent years, the amount of electronics introduced into automobiles has increased significantly. As the number of Electronic Control Units increase in vehicles the need for faster and more reliable communication is required. This trend will continue as automobile manufactures initiate further advances in safety, reliability and comfort. Time-triggered protocols like Flex Ray have been gaining ground as the standard for high-speed reliable communication in the automotive industry. Flex ray is a new communication protocol designed to provide message and data exchange between electronic devices installed in a vehicle. It was developed by the Flex ray Consortium as flexible and fast communication system by coupling several companies like Bavarian Motor Work, Diamler Chrysler, Volkswagen, Philips, general motors, Robert Bosch, NXP semiconductors and Freescale semiconductors operating in the fields of automotive and electronic industry. The main target for the new protocol introduction is the growing demand for massive and complex data exchange in the environment where real-time requirements are sometime critical. Flex ray is based on frame exchange, equipped with several supporting mechanisms providing protection of the data integrity and the whole system stability. From topological point of view, the Flex ray communication system consists of nodes, where each node may be a communication interface of a specific electronic component of a car. These nodes may be connected in a serial bus, a star or the combination of the two. In Controller Area Network protocol, it is quite difficult to transmit two data streams at a time with equal priorities. Flex ray succeeds this problem by dividing the medium into static and dynamic time slots. During the static segment, a deterministic access schema is used to transmit periodic messages while dynamic segment allows priority based Transmission of occasional messages. Another great feature of Flex ray is the capability to use two physical transmission channels named, channel A and channel B. Each channel can transmit up to 10Mbits/s on the physical layer, which is ten times the factor of the maximum rate of a CAN channel. The two channels can be used to double the overall data rate, provide redundancy, by transmitting only important messages on both the channel. To connect to a Flex ray cluster, a Flex ray Communication Controller is required. It handles the protocol according to the specifications and connected to the same host.

II. LITERATURE REVIEW

Data communication protocols for embedded systems can be divided into two main categories: Event-Triggered and Time-Triggered [12]. Event-triggering was the first paradigm used for data communication and at any time an event can trigger data transmission. Time-triggered communication systems usually use the Time -Division Multiple Access (TDMA) approach to share the transmission medium between different nodes in a static, pre-defined manner. Safety-relevant applications require reliable, real-time systems, infotainment applications require a high speed data transmission, and the networking of sensors and actuators in car body require a cost-efficient system with minimum wiring costs. This section introduces some of the communication protocols that are used in real-time embedded systems.In this paper[15], although the controller area network(CAN) serves as a multi-master bus, it is not deterministic , insufficient for real-time applications and supports data rate upto 1Mbps only. Time-Triggered Controller Area Network (TTCAN) is a higher layer protocol above the event based CAN protocol. Data communication in time-triggered suffers from lack of bandwidth and flexibility [14].Moreover the Flex ray protocol allows a time-triggered communication for several control function with real-time requirements.

III .PROPOSED SYSTEM

A. *Flex ray Protocol*

Flex ray is a new paradigm of network communication system which provides high speed serial communication, time triggered and fault tolerant communication between the electronic devices for automotive applications. The Flex ray supports both time-triggered and event triggered scheme. The upper bound of the data rate is 10Mbps and it provides two channels for redundancy. Support for on-demand communication, and make sure that it does not interfere with the deterministic communication. Flex ray is based on frame exchange, equipped with several supporting mechanisms providing protection of the data integrity and the whole system stability. . It has a message exchange service that provides cycle based message transport. The Flex ray protocol has been developed by an industry consortium with four founding members (BMW, Diamler-Chrysler, Philips and freescale).The Flex ray protocol is expected to become the de-facto industry standard in future automotive systems. The Flex ray is based on a TDMA approach. If a node wishes to transmit a message, it must wait until its communication slot comes around.It may then transmit a single message that can consist a data section of between 0 and 254 bytes. This protocol supports different network topologies such as passive bus,passive or active star, cascaded star and hybrid topologies. The Flex ray supports dual channel communications, which can be employed in fault-tolerant systems.

B. *Flex ray ECU Architecture*

In Flex ray communication network, each Flex ray network can be represented as a node .The Flex ray node is also called as Electronic Control Unit (ECU). Each Flex ray node consists of Host, one Communication Controller, one power supply unit, two Bus guardians and two Bus drivers.

As shown in fig1, the host microcontroller controls all the software operations of Flex ray controller. It sends configuration data to the communication controller and receives status from communication controller. Host

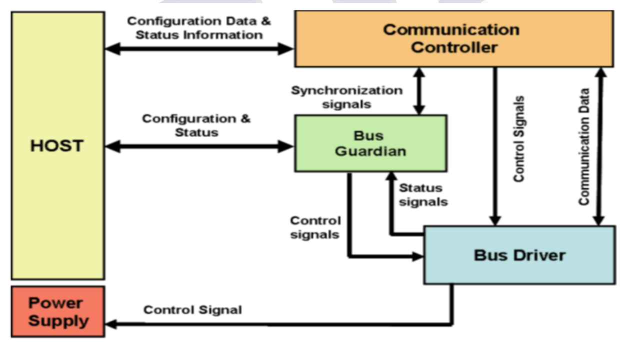

Fig.1 Flex ray node architecture

Microcontroller also transmits configuration data to the bus guardian module through serial peripheral interface and it also receives status from bus guardian module. The host controls the operating modes of bus driver and to read error condition and status information from the bus driver..The host is also-called an Electronic Control Unit (ECU), which consists of a processor with its memory interface and attached I/O-devices. Communication controller (CC) is an electronic element in a node that is responsible for implementing the protocol aspects of the Flex ray communication system. It execute all communications tasks such as reception and transmission of messages in a Time Triggered Protocol (TTP) cluster without interaction of the host CPU.The core of the communication controller is the protocol engine which consists of protocol operational states. The communication controller (CC) provides status information to the host microcontroller and delivers payload data received from communication frames. The Flex ray communication protocol is defined for a dependable automotive network.

C. Communication Cycle

The basis for the media access in the Flex ray protocol is its communication cycle. The Flex ray protocol transfer data and divides time into equally repeating, long communication cycles. They are numbered from 0 to 63 and sent one after another. If cycle 63 is reached, the cycle counter is reset to 0. In this protocol, the communication cycle is a concatenation of a time-triggered window (or static), an event-triggered window (or dynamic), a symbol window and a network idle time (NIT).

In the Flex ray protocol, the message frames are sent in the static slots or in the dynamic slots of each communication cycle. The static segment consists of time slots with fixed duration and are time-triggered. Each slot has ID assigned to a specific control unit. The static segment is equipped with Time-Division Multiple Access (TDMA) mechanism. The Dynamic segment uses flexible time trigger, which is known as Flexible Time Division Multiple Access (FTDMA), for the media access control. If task is triggered by a significant change of state, it needs to use FTDMA to arbitrate the media and transmitted. FTDMA enables frames, when it has chance to send data if needed. There are no static slot allocations in advance. Except the slot size FTDMA is similar to TDMA. The symbol window is a time slot of regular and fixed duration, in which special symbols such as wake-up pattern used to wake up the sleeping nodes to initiate communication

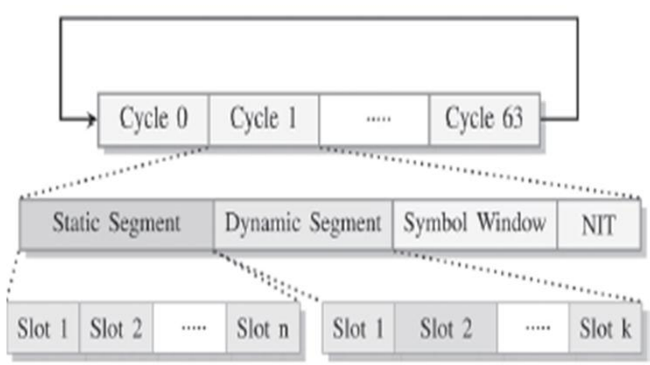

Fig.2 Flex ray communication cycle

Symbols are used for network management purposes. In the symbol segment, Flex ray sends internal control information. Most high-level applications do not interact with the symbol window. Network idle time is used to isolate two successive communication cycles. The network idle time is a communication-free period that completes every communication cycle. The duration of the network idle time is a global parameter that has to be consistent between all controllers in a network.

D. Flex ray Frame Format

Data transmission in a Flex ray node is performed using a uniform message frame. Each Flex ray message is composed of three parts .An overview of the Flex ray frame format is shown in fig.3

First is the header segment which begins by the reserved bit and is followed by a series of indicator bits. Frame ID determines the time slot of the communication cycle in which the frame is being transmitted. It can either be a static slot or a dynamic slot but no other frames are allowed to have the same Frame ID in the same communication cycle. The payload length is indicated by a number of half-words (16 bits).A maximum of 254 user bytes (payload) can be transported by one frame. The payload data is secured by the cyclic redundancy check (CRC) method. The header CRC is calculated from the last two indicator bits, Frame ID and payload length. It serves as a means of verification of the transmissions correctness. The cycle count denotes the cycle number. It ranges from 0 to 63. It starts from zero again, when it reaches its maximum value This is useful mostly for

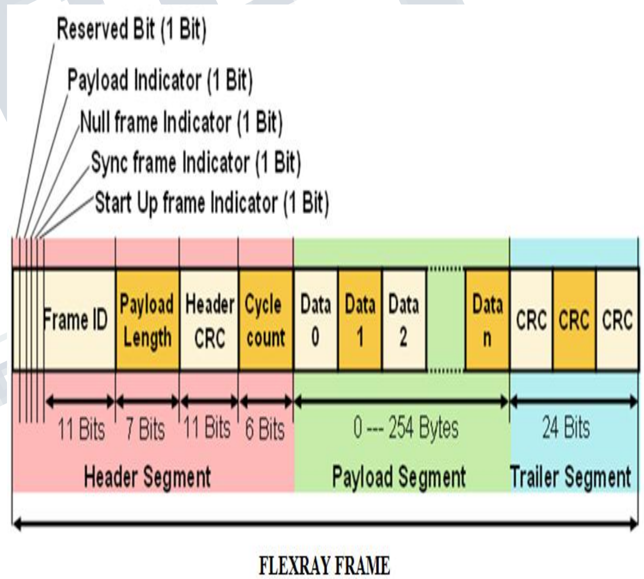

Fig.3 Flex ray Frame Format

The so called cycle filtering .In the same manner, the protocol supports also filtering of received frames. The data segment contains 0 to 254 bytes of payload data.. The CRC segment contains a value calculated over the whole header and data segments. The presence of two CRCs in a single frame is the Flex rays security features. The Flex ray trailer segment consists of 24-bit (CRC) for the frame.

E. Flex ray Communication Controller

Communication controller (CC) also referred to as Flex ray controller. It is a part of Flex ray ECU and is responsible for implementing the protocol aspects defined in the Flex ray specification.

The Controller Host Interface (CHI) module is used for transmitting operation message to host and receiving data and command from host. The Protocol Operation Control (POC) module is the status control unit and controls the operational modes of the Flex ray modules. The Clock Synchronization Process (CSP) module is responsible for synchronizing timing units with other nodes in the Flex ray communication controller. The Media Access Control (MAC) controls the commands to the coding and decoding unit (CODEC). The CODEC module, encodes the communication elements in the form of bit stream and it decodes the communication elements by making bit streams and then checks the correctness of the bit streams. The Frame and Symbol processing (FSP) module is responsible for managing the correct timing of received frames and symbol.

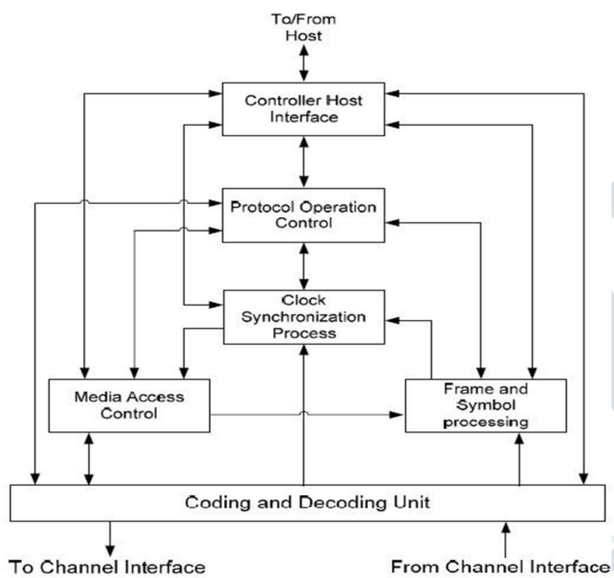

Fig.4 Block diagram of Flex ray communication controller

F. Flex ray Protocol Operation Control unit

The Protocol Operation Control (POC) acts as an interface between the Host and the sub-modules of Flex ray Protocol Engine.The POC indicates the operational state of the controller. POC state transitions are controlled synchronously and occur as a consequence of a host command or as a result of an error condition. . The state transitions cause an alteration in the operation of the protocol engine.. Operational state transitions can occur with reset conditions, host commands, and internal protocol error conditions.The host can issue commands by writing to the Controller Host Interface (CHI) command register. The Flex ray controller supports the following protocol operational states as shown in fig. 5

i) DEFAULT CONFIG state: This is the startup state of the communication controller.In this state, no communication is possible, since information concerning the connected Flex ray network are not available. To leave DEFAULT CONFIG state, the Host has to write the Command [4:0] = 00001 then the controller transits to CONFIG state.

ii) CONFIG state: The host submits specific data, which is necessary for the operation of thecommunication controller. After unlocking CONFIG state and writing theb command [4:0] = 00010 the communication controller enters READY state. In this state , the communication controller moves to wakeup state and perform a cluster wakeup .

iii) READY State: The Communication Controller has been successfully set up and is ready for starting the communication process. It can join either an already existing system or wakeup an idle network.

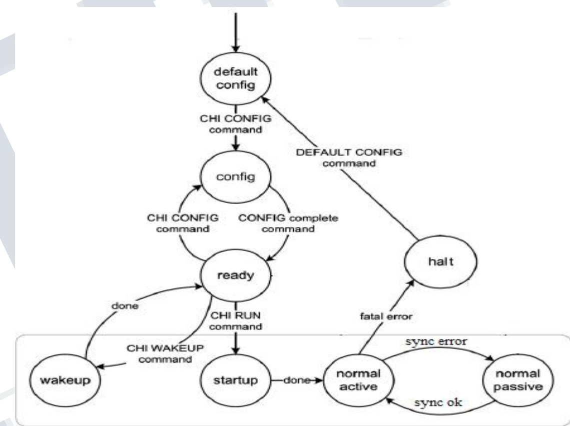

Fig.5 Flex ray Protocol Operation States

iv) WAKEUP State: The communication controller leaves this state when exiting from READY state by writing command [4:0] = ”00100” (wakeup command).The Flex ray controller leaves from this state to READY state after complete non-aborted transmission of wakeup pattern.

v) STARTUP State: In this state, the node tries to synchronize
with the bus. Depending on the role of the node, there are two different approaches. If the node is a Coldstart Node, it actively participates in the synchronization process of the network. Otherwise, it uses the already existing bus communication to synchronize itself.

vi) NORMAL ACTIVE state: This is the main operating state of the communication controller, in which communication with the bus is possible.

vii) NORMAL PASSIVE state: In case of specific errors, the node autonomously switches to this state or the halt state.The node is only allowed to listen to the bus, but is restricted from transmitting data.

viii) HALT state: All processes in the Communication Controllers are stopped. This state is reached, if an internal error of the Flex ray Controller occurs. It can only be left via the default config state.

IV. EXPERIMENTAL RESULTS

The communication controller was designed by using Finite State Machine (FSM) methodology and the states are explained with the help of state diagram and communication controller commands. The frame structure, protocol operation using verilog code are shown below.

- ❖ Simulation of Flex ray protocol frame
- ❖ Simulation of Flex ray protocol communication controller states

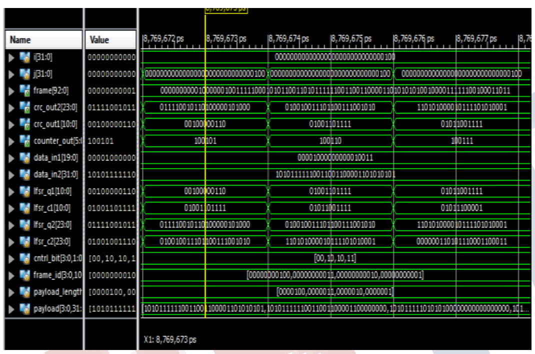

Fig.6 Simulation result of Flex ray protocol frame

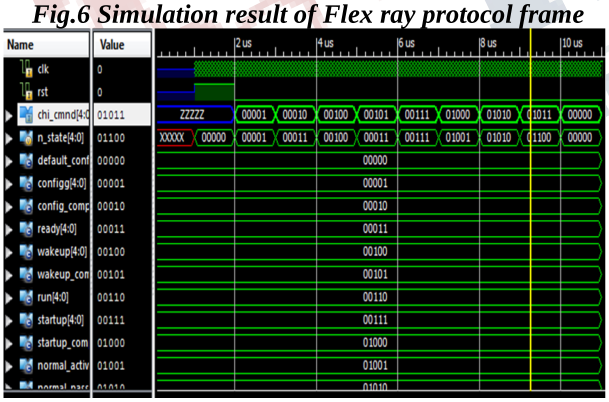

Fig 7: Simulation result of controller states

V. CONCLUSION

As a result of increasing number of electronic components inside vehicles, the demands on the communication have increased as well. The automotive network communication protocol Flex ray was developed to meet the continuously growing demands, offering a time-triggered functions and a higher bandwidth than existing protocols. Flex ray protocol is a deterministic, fault-tolerant and high-speed bus system with high performance, and it has more and more promising future in automotive industry. The proposed technique is based on generating the Flex ray protocol frame and monitoring the protocol operational states of a communication controller, where the protocol operational states occur as a consequence of a host command or an error condition occurring. The Flex ray communication controller and the protocol frame were modeled by Verilog HDL at the behavioral level.

REFERENCES

[1] Shanker Shreejith Student Member, IEEE and Suhaib A. Fahmy Senior Member, "Extensible Flex ray Communication Controller for FPGA-Based Automotive Systems", no.99,May 2014.

[2] Wissam Kafa, "The Flex ray protocol, Master of control systems", june 2014.

[3] Avinash K R, P Nagaraju, Surendra S, Shivaprasad S, "Flex ray protocol based an automotive application", International journal of Emerging technology and Advanced Engineering(IJETAE), vol. 2, no.5, May 2012.

[4] Antonio Cappiello, Omar Jaradat, "Flex ray Fault-Tolerance: Capabilities,weaknesses and proposed enhancements", March 2011.

[5] Yasser Sedaghat, Seyed Ghassem Miremadi, "An FSM-based monitoring technique to differentiate between follow-up and original errors in safety-critical distributed embedded systems",May 2011.

[6] J. Y. Hande, M. Khanapurkar, and P. Bajaj,"Approach for VHDL and FPGA implementation of communication controller of Flex ray controller", in Proc. Int. Conf. Emerging Trends Eng. Technol., ICETET, Dec. 2009, pp. 397401

[7] Awani Gaidhane, Milind Khanapurkar, Dr. Preeti Bajaj, "Design Approach for FPGA Implementation of Flex-ray Controller Using VHDL for Intra Vehicular Communication Application",ICETET- July 2008.

[8] Y.-N. Xu, I. Jang, Y. Kim, J. Chung, and S.-C. Lee, "Implementation of Flex ray protocol with an automotive application",in Proc. Int. SoC Des. Conf. (ISOCC), Nov. 2008.

[9] Yi-Nan Xu, Y. E. Kim, K. J. Cho, J. G. Chung, and M. S. Lim, "Implementation of Flex ray Communication Controller Protocol with Applicationto a Robot System,"

15th IEEE International Conference on Electronics, Circuits and Systems, pp. 994-997, 2008.

[10] R. Shaw and B. Jackman, "An Introduction to Flex ray as an Industrial Network," *IEEE International Symposium on Industrial Electronics*, pp. 1849-1854, 2008..

[11] Flex ray Communications System, Protocol Specification Version 2.1 Revision A, Flex ray Consortium Std., Dec.2005.

[12] A. Albert, "Comparison of event-triggered and time-triggered concepts with regard to distributed control systems," Embedded world, pp. 235-252,Feb.2004

[13] Yi-Nan Xu, Chang-Ha Jeon and Jin-Gyun, "Chung,Design of Flex ray communication network using active star".

[14] Steve C. Talbot and Shang ping Ren, "Comparison of FieldBus Systems,CAN, TTCAN, Flex ray and LIN in Passenger Vehicles".

[15] Andreas Forsberg and Johan Hedberg, "Comparison of Flex ray and CAN bus for real-Time Communication".