Volume : 2 | Issue : 6 | June 2013 • ISSN No 2277 - 8179

Research Paper

Approach for Automotive Control Systems for Driver Assistance Using Flex-Ray Controller

Engineering

KEYWORDS : ECU – Electronic Control Units, FPGA: Field Programmable Gate Array, LIN: Local Interconnect Network, CAN: Controller Area Network, TDMA: Time Division Multiple Access,

Pravin P. Bhad Assistant Professor, ETC Dept, Priyadarshini J.L. College of Engineering, Nagpur, India

Jayant Y. Hande Assistant Professor, EN Dept, Priyadarshini J.L.College of Engineering, Nagpur, India

ABSTRACT

Automotive OEMs are expected to deliver the latest technologies with a high degree of integration while keeping pace with faster design cycles. From driver assist to infotainment to e-vehicles, automotive systems call for leading performance, best-in-class quality, and highly reliable underlying technology, as well as design flexibility to meet emerging needs. Automotive software is one of the most challenging fields of software engineering; it must meet real time requirements, and is distributed over multiple processors. Flex Ray is a fast, deterministic and fault tolerant bus system for automotive use. Flex Ray is recently developed communication standards used between electronic modules installed in a vehicle. This paper gives information of Automotive Control Systems using Flex Ray controller.

I. INTRODUCTION

Flex-Ray is a new communication protocol designed to provide large bunches of data to be exchanged in real-time and with high dependability between electronic control units, sensors and actuators. According to this perspective, the Flex Ray protocol, which features data-rates up to 10 Mb/s, time and event triggered transmissions, as well as scalable fault-tolerance support was developed and it is now expected to become the future standard for in-vehicle communication. 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 vehicle. These nodes may be connected in serial bus or in a star configuration or a combination of the two. This paper discusses data communication of input and output signals for advanced applications like as vehicle driver assistance systems. The rapid growth in advanced automotive control systems requires a new, more powerful communications protocol to ensure a network that works together to accomplish critical tasks within the vehicle. The Flex Ray Consortium—made up of a group of industry heavyweights, formed to enable the development of next-generation by-wire systems and to promote an open-source network communications protocol for the automotive industry. First three sections discuss a brief overview of the Flex-Ray protocol along with flex-ray data frame and block diagram of a flex-ray communication node. Authors are trying to implement driver assistance systems module using flex-ray on FPGA which along with general purpose processors can be integrated to allow hardware to be synthesized according to the application needs.

II. STANDARD FLEX-RAY FRAME

Flex-Ray protocol is based on frames, containing data organized in bytes, but transmitted serially. Figure 1 shows the block diagram of flex-ray frame, consists of 3 segments –header, Payload and Trailer. Fig.1 shows details of flex-ray frame. (a) The Header begins with 5 indicators –the single bits defining basic features of the frame. (b)Payload section contains the main data; its length may be variable between 0 and 254 bytes. (c) The Trailer section contains 24 bits of CRC, calculated for the Header and Payload section together.

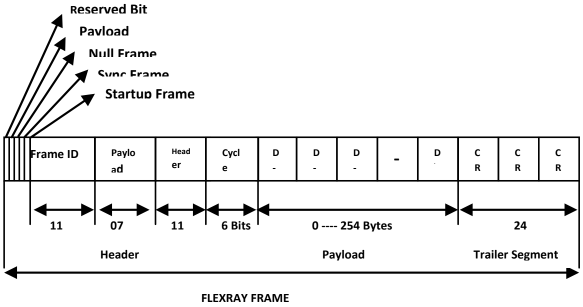

Fig1. Flex-ray Frame Format

From the timing point of view Flex-ray communication is based on equal length cycles. Each cycle is a complex structure, containing static segment, dynamic segment, symbol window and idle time. Static segment the first part of cycle contains series of static slots each of these slots is allotted to transmit a single frame. Another part of the cycle is the dynamic segment consists of mini slots. This part of cycle may be used for the frames transmission again but the amount of time allotted a current frame may vary, depending on its length. The macroticks consists of series of microticks generated by the oscillator in communication control module of each node.

III. ARCHITECTURE OF THE FLEX-RAY ECU

Each flex Ray node consists of a host, communication controller, power supply unit, two bus guardians, and two bus drivers. Each communication channel has one bus driver to connect the node to the channel. The Host is an embedded microprocessor running the software which controls the communication process. Communication Controller provides the operation (via the Bus Driver) and is responsible for synchronization. Bus Guardian provides mainly the error detection, generating interrupts and blocking the line when critical problems occur. Flex-Ray node is equipped with two physical Channels – A and B.

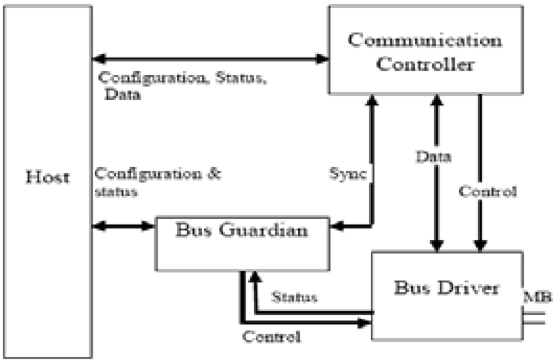

Figure 2. Architecture of Flex-ray Node

IV Advanced Driver Assistance Systems

FPGAs can help to keep pace with the rapid advancements in driver assistance technology:

- Higher levels of bandwidth and performance needed to process high-definition (HD) video streams from multiple cameras
- Complex, real-time processing required combining the camera and radar/laser/LIDAR sensor data, and applying advanced analytics for object identification and signature recognition
- Transmit, receive, and translate between multiple communication standards (i.e., CAN, local interconnect network

(LIN), FlexRay, MOST, Ethernet, LVDS)

Because manufacturer's ADAS requirements differ for each vehicle type, you are facing a challenge to provide a platform solution that can scale across an entire vehicle lineup.

FPGAs provide an ideal platform for developing high-performance, low-power, low-cost ADAS systems with the optimal level of integration and flexibility. You can even incorporate changes late in the design cycle, and in-field upgrades let you differentiate your products and keep pace with your customers' expectations.

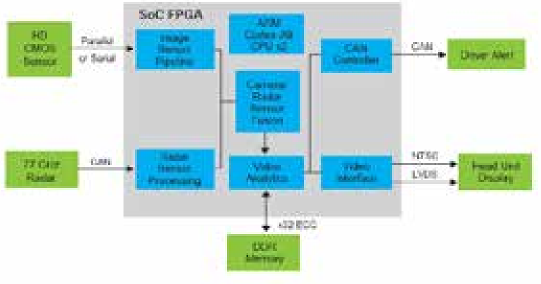

Fig 3. Forward-Looking Sensor Fusion ECU

Automotive infotainment systems are an integral part of modern vehicle design and greatly influence sales. You need to continually create compelling new technology that both rivals and complements the latest consumer devices while keeping up with the pace of product obsolescence common in consumer electronics. It is more important than ever to select the right main system processor to differentiate the system's user interface with the latest graphics. Should you choose a high-performance system on a chip (SoC) with a graphics processing unit (GPU) or a CPU with applications that can be upgraded with software? With multiple models to support, you may need to select several different SoCs due to system variations and emerging interfacing technologies. This is not cost effective or an efficient use of engineering resources. By using our FPGAs as an I/O companion, you can support any combination of I/O interfaces. You may also leverage the FPGA as an efficient coprocessor to offload functions from your host main processor such as video scaling and graphics acceleration. With an FPGA, your system becomes easily scalable, enabling you to upgrade firmware on the fly to support multiple manufacturers, regions, and models with minimum changes to the hardware.

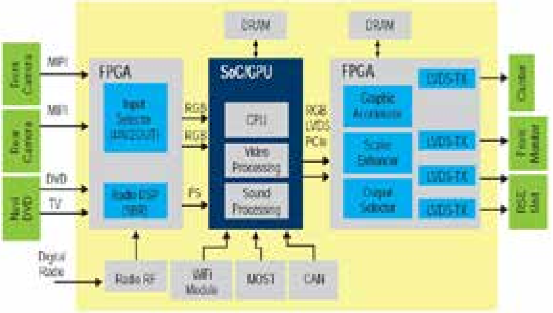

Fig 4 System I/O Companion

Benefits of FPGAs in Infotainment Applications

- Support multiple camera inputs in any bit rate or resolution via video select and front-end processing
- Support multiple display interfaces and offloads GPU
- Optimize system performance by integrating 2D/3D graphic accelerator, scalar, image enhancer, and interface protocol bridge
- Reduce cost by integrating radio digital signal processing (DSP) into FPGA with software-based radio IP
- Accelerate time to market by supporting the latest generation of interface standards in programmable logic
- Minimize PCB spins and future-proof your system with in-field upgrades to the FPGA.

V. ALGORITHMS:

Design approach to FPGA Implementation of Driver Assistance Systems using flexray controller will be done and the approach towards this followed by author is given below:

- Designing of System I/O Companion
- Design of interfacing Module
- Interfacing these with Host, BG, BD and achieving communication between those using FlexRay protocols.
- Coding of complete system in simplest possible using VHDL.
- Analysis of the design.
- Synthesis of the Design
- Implementing the complete design on FPGA platform

VI. CONCLUSIONS:

This paper has highlighted the concept of the FlexRay protocol and the authors have used this protocol for a new application, which is intra vehicular communication between modules of a car. The design approach for the same and the algorithm is added in this paper. It is recommended to use VHDL tools for the proposed architecture of the application. This will lead to cost effectiveness, simplicity, versatility and other related advantages. The result of interlinking will be presented in future papers.

REFERENCE

[1] Jayant Y. Hande, Milind Khanapurkar, Dr. Preeti Bajaj, "Design Approach for VHDL Implementation of Communication Controller of Flex-Ray Controller", PICETE – Dec. 2008, Pravara International Conference Pravara Rural Engg. College, Loni. (Maharashtra). | [2] Jayant Y. Hande, Milind Khanapurkar, Dr. Preeti Bajaj, " Approach for VHDL and FPGA Implementation of Communication Controller of Flex-Ray Controller", ICETET- 2009, | International Conference, G. H. Raisoni College of Engineering, Nagpur. (Maharashtra). | [3] "On Hardware Implementation of FlexRay Bus Guardian Module", Proc. 12th MIXDES JUNE-2007. FlexRay Communication System - Protocol Specification, v2.0, FlexRay Consortium, June -2004. | [4]FlexRay Consortium, Homepage, http://www.flexray-group.org | [5]FlexRay Consortium, FlexRay Communications System: Protocol Specification Version 2.0, 2004. | [6] FlexRay Communications System: Protocol Specification Version 2.1 Revision; FlexRay; Consortium; 2006. | [7] FlexRay Communications System Data Link Layer Conformance Test Specification Version 2.1.1 | [8] ISE 9.1i Release Notes and Installation Guide, Xilinx Corporation. | [9] TTP/C-C2 Communication Controller – Preliminary Data Sheet; Rev. 16, May 2002. | [10] Message Handling Concept for a FlexRay Communication Controller- Special Edition FlexRay automotive; 2004. | [11] Multiplexed Networks for Embedded Systems CAN, LIN, FlexRay, Safe-by-Wire. Copyright 2007 John Wiley & Sons Ltd. | [12] Driving Innovative Automotive Solutions, br-automotive_broch-1005.