Send Orders for Reprints to reprints@benthamscience.ae

330 *The Open Cybernetics & Systemics Journal,* 2014, *8,* 330-334

Open Access

Design and Realization of FlexRay Communication Unit in Vehicle Network

Yi Wang[1*] Hui Pan[2] Jiawen Liang[1]

[1]*School of Physics and Electronics Science, Guizhou Normal University, Guiyang Guizhou 550001, China;* [2]*School of Electronics and Information, Guizhou University, Guiyang Guizhou 550025, China*

Abstract: The conventional FlexRay communication unit has complex wiring and low reliability, in order to improve the network rate, reliability and flexibility, this paper adopts the method of hardware and software combination. This article designed the FlexRay bus communication unit with the background of high-performance MC9S12XF microcontroller. The hardware structure and software design processes of the FlexRay communication unit are proposed respectively. In this paper, certain scientific basis for FlexRay communication exploitation is provided for the circuit with the characteristics of real-time, simple hardware design and high reliability.

Keywords: Communication unit, FlexRay, hardware design, MC9S12XF microcontroller, software, vehicle network.

1. INTRODUCTION

With the rapid development of modern automobile electronic technology, the function of automobiles is increasing continuously. Increasing number of electronic systems is applied to the car, especially the application of X-by-wire system, which promotes the development of vehicle bus and network technology. Traditional in-vehicle network systems such as LIN(Local Interconnection Network) and CAN(Controller Area Network) bus is replaced by FlexRay bus, because of its time-triggered communication method, fault-tolerance and high data rates. It can integrate and coexist with current network systems, including CAN, LIN and J1850 protocol etc. FlexRay has become a new standard for vehicle bus communication among electronic devices. Nowadays, almost all the leading vehicle, semiconductor and electronic systems manufacturers have become the member of the FlexRay Consortium, who contribute their product solutions and applications for FlexRay to further its development [1, 2].

Communication unit ECU (Electronic Control Unit) is the core of FlexRay, which is a function control unit in the vehicle network. This paper describes the hardware and software design of FlexRay communication unit based on the high-performance MC9S12XF microcontroller of Freescale and TJA1080 bus driver of NXP [3-5]. It simplifies the design of hardware circuit and increases the reliability of the system.

2. BASIC FEATURES OF FLEXRAY

FlexRay communication protocol has become popular for its real-time and fault-tolerant features with time division multiple access (TDMA) scheme of bus scheduling [6]. It not only simplifies vehicle communication system architecture, but also contributes to car communication system and achieves higher stability and reliability.

Compared with other field bus systems like CAN, LIN etc, it has the following advantages. FlexRay has the high data rate with dual channels, up to 10 Mbit/sec (up to 20 Mbit/sec with two channels) [7, 8]. It can be arranged in different physical topologies such as bus, star, point-to-point or hybrid. It is more accurate for fault tolerance and redundancy. Real-time capabilities guarantee message latency and reduced message jitter [9].

For high bitrates or increased safety two independent physical channels should be preferably used. The physical topology of each channel can be different. FlexRay can be used for the next-generation backbone network of a vehicle network with its efficient network utilization and flexibility [10, 11].

3. FLEXRAY NETWORK TOPOLOGY

FlexRay network can apply different network topologies to connect each node, whose structure mainly consists of bus, star and hybrid network topologies.

3.1. Bus Network Topology

In bus network topology, each ECU is connected to every other ECU by one cable. All ECUs share a data channel and any ECU information can be transferred in both directions along the bus. The bus-based network topology is shown in Fig. (**1**).

In bus network topology, each bus interface has the function to receive and send information. The receiver accepts serial information from the bus and converts it into appropriate information to send to the ECU. The transmitter converts node information into serial information and sends to the bus.

*Address correspondence to this author at the School of Physics and Electronics Science, Guizhou Normal University, No.166, Baoshan North Road, Guiyang City, Guizhou Province, China; Tel: 86-0851-6700647;
E-mail: wyigz@126.com

1874-110X/14 2014 Bentham Open

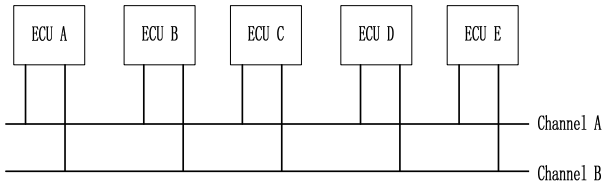

Fig. (1). Bus-based network topology.

3.2. Star Network Topology

In star network topology, each node connects through point-to-point to a central node, messages are transmitted from the central node to the destination node. Any two nodes in a star network topology have to go through the control of central node. The star network topology is shown in Fig. (**2**).

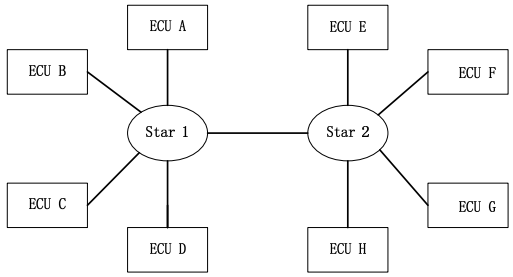

Fig. (2). Star network topology.

Under a high transmission rate and long distance circumstance, the use of star network ensures better stability and successful transmission rate than the use of passive bus. The significant strength of Star Network is fault decoupled. If problems occur in the transmission channel, caused by break off or short-circuit, the feature of Star Network can decouple fault branches and ensures that other nodes function regularly without interruption.

3.3. Hybrid Network Topology

Hybrid network topology refers to the structure made up of several topologies, such as bus structure, star structure. The hybrid network topology is shown in Fig. (**3**).

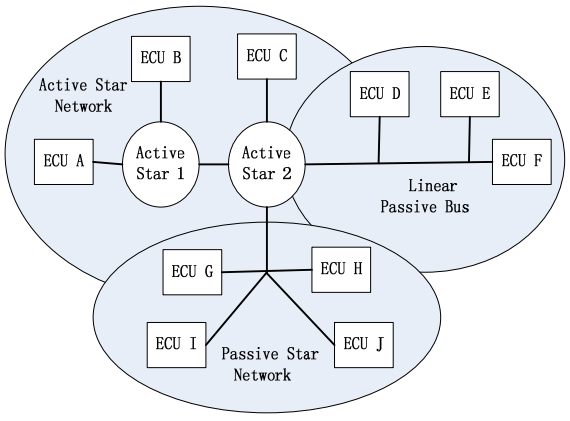

Fig. (3). Hybrid network topology.

As shown in Fig. (**3**), FlexRay can be used with multiple network topologies. The most commonly used topology is the hybrid network topology, in which multiple branches are connected together by using a central active star device to extend linear passive bus, active star network or passive star network. Each branch can contain one or more ECUs [12].

4. CHARACTERISTICS OF MC9S12XF INTEGRATED CONTROLLER

MC9S12XF512 is based on the structure of Freescale's S12X. It provides a high-performance, cost-effective distributed control solution for nodes on FlexRay networks [13]. The system includes these distinctive features:

• 16-Bit CPU12X

Additional (superset) instructions to improve 32-bit calculations and semaphore handling

Enhanced indexed addressing

Access to large data segments independent of PPAGE

• Enhanced Interrupt Module

Flexible assignment of interrupt sources to each interrupt level

One non-maskable high priority interrupt (XIRQ)

Wake-up Interrupt Inputs

• Memory Protection Unit (MPU)

4 address regions definable per active program task

Address range granularity as low as 256-bytes

Non-maskable interrupt on access violation

• XGATE

Programmable, high performance I/O coprocessor module

Up to 100 MIPS RISC performance

Transfers data to or from all peripherals and RAM without CPU intervention

Enables Full CAN capability when used in conjunction with MSCAN module

Full LIN master or slave capability when used in conjunction with the integrated LIN SCI module

• System Integrity Support

Illegal address detection with reset

Low-voltage detection with interrupt or reset

Clock monitor supervising the correct function of the oscillator

• Memory Options

128K, 256k, 384K and 512K byte Flash

2K, 4K byte Emulated EEPROM

16K, 20K, 24K and 32K Byte RAM

64 data bits plus 8 syndrome ECC (Error Correction Code) bits allow single bit fault correction and double fault detection

Up to 32K bytes of D-Flash memory with 256-byte sectors for user access.

• Clock and Reset Generator (CRG)

Phase-locked-loop (IPLL) clock frequency multiplier

Fast wake up from STOP in self clock mode for power saving and immediate program execution

• Real Time Interrupt (RTI)

Real Time Interrupt for task scheduling purposes or cyclic wake-up

Can be active in Pseudo Stop mode for low power precision timing tasks

• Background Debug Module (BDM)

Background debug controller (BDM) with single-wire interface

Supports in-circuit programming of on-chip non-volatile memory

• 50MHz maximum CPU bus frequency, 100MHz maximum XGATE bus frequency

Based around an enhanced S12X core, the MC9S12XF-Family runs 16-bit wide accesses without wait states for all peripherals and memories. The MC9S12XF-Family also features a new flexible interrupt handler, which allows multi-level nested interrupts.

Targeted at actuators, sensors and other distributed nodes in the FlexRay network for Chassis and Body Electronics, the MC9S12XF-Family delivers 32-bit performance with all the advantages and efficiencies of a 16-bit MCU.

5. HARDWARE DESIGN OF FLEXRAY COMMUNICATION UNIT

A FlexRay communication unit is made up of a controller module and a driver module. The block diagram of the designed ECU is shown in Fig. (**4**). The controller module consists of a host processor and a communication controller (CC), and for the driver module consists of a bus guardian (BG) and a bus driver (BD). Time slots of CC assigned transfer to BG in the function of host processor. The BG is in charge of security and safety for FlexRay communication systems. It protects channel from interference caused by any message that is not aligned with the communication schedule [14]. The BD connects the FlexRay bus and CC and BG.

There are two architectures of FlexRay ECU hardware circuit design. One is composed by the host processor, communication controller and the bus driver, for example the architecture of MPC5567+MFR4310+TJA1080. The other is composed by the host processor with integrated communication controller and bus driver [15, 16]. The former is an early FlexRay communication unit mode and the hardware circuit structure is more complex. This paper uses the latter mode with processor MC9S12XF and bus driver TJA1080.

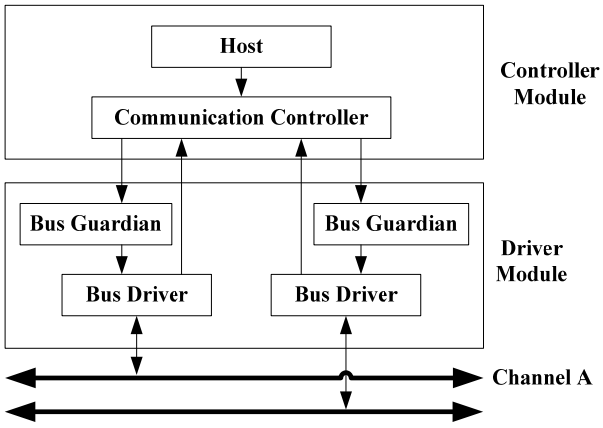

Fig. (4). Structure diagram of the FlexRay communication unit.

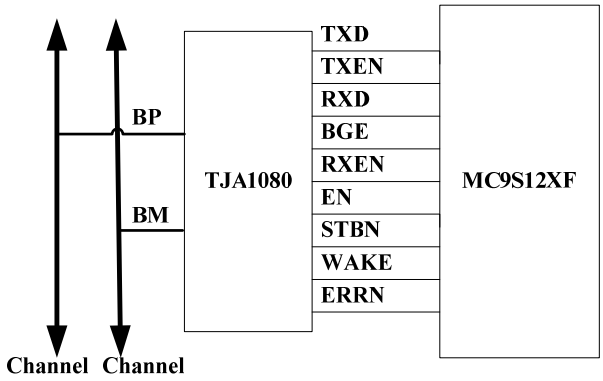

Fig. (5). Circuit diagram of the FlexRay communication unit.

Fig. (**5**) depicts the circuit diagram of FlexRay communication unit mainly composed of MC9S12XF and TJA1080. MC9S12XF is the core block, whose peripheral circuits are mainly composed of the power management module, reset circuit, and clock generator, etc. The peripheral circuit diagram of MC9S12XF is shown in Fig. (**6**), whose function is primarily used for AD signal acquisition, watchdog reset, permanently record data, the global clock signal processing, etc.

6. SOFTWARE DESIGN PROCESS

Software design is the key to system. In order to improve reliability and comprehensibility, the software part of the system is modular in design. Each module is connected to form a complete software system [17]. Software design mainly completes the parameter setup and initialization for FlexRay communication controller module, which is the process of building the microcontroller operating environment. The main program module is the main part of the software design, whose tasks are system initialization, data signal acquisition/sending/ processing and data communication/transmission, etc.

The main program flow chart of FlexRay is shown in Fig. (**7**). Due to the internal clock of MC9S12XF is utilized in FlexRay communication controller, so the clock phase-locked loop (PLL) of MC9S12XF should be configured before the

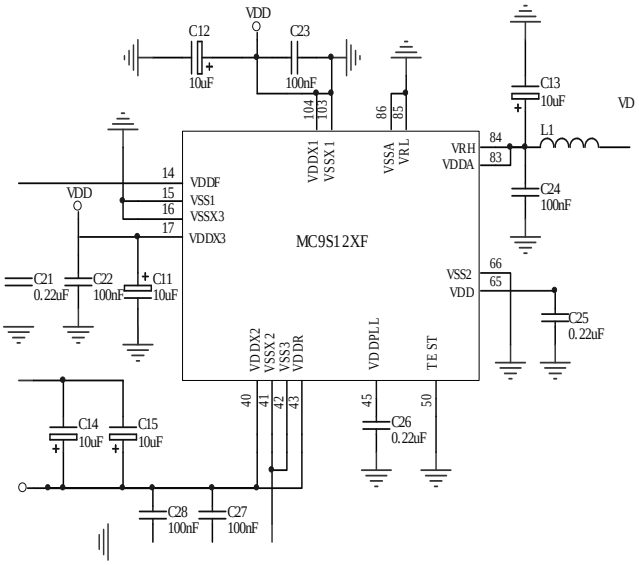

Fig. (6). Peripheral circuit diagram of MC9S12XF.

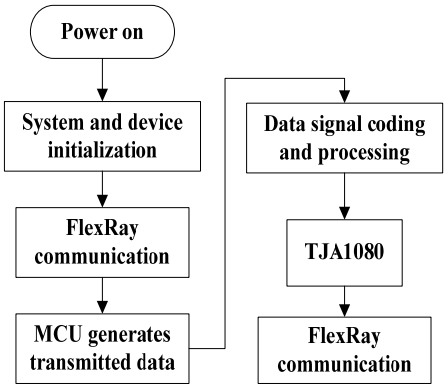

Fig. (7). Main program flow chart.

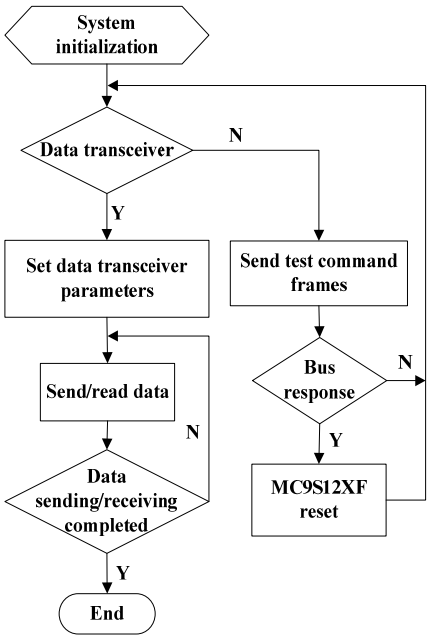

Fig. (8). Data communication flow chart.

MC9S12XF FlexRay module initialization. After initialization of MCU, FlexRay module enters the FlexRay protocol configuration and starts up the FlexRay communication controller. FlexRay module, clock module, phase-locked loop (PLL) module, general purpose I/O port, real-time interrupt module etc needs to be initialized. After complete initial configuration of FlexRay communication unit, FlexRay needs to send and receive frames information to implement FlexRay data communication. Then the data transfer to bus transceiver module TJA1080 through the data signal coding and processing of MC9S12XF. Finally the FlexRay bus completes the entire communication process. Detailed FlexRay data communication process shown in Fig. (**8**).

CONCLUSION

As the next-generation automotive control and communication protocol, FlexRay has higher transfer rates, is more secure and reliable than traditional CAN bus. And BMW already uses the FlexRay protocol in BMW X5, X6 and X7. The paper has presented a FlexRay communication unit to hardware design and software process. Hardware design is given based on the mainstream of FlexRay communication unit MC9S12XF series single-chip. The structure of FlexRay communication unit and its application peripheral circuits are designed. Software is designed based on modular ideology, which improves the reliability and maintainability of the system. The FlexRay communication unit operates in real-time and has high integration performance. The system meets the design requirements very well, so it has great practical value.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported in part by the Guizhou Science and Technology Department School Cooperation Project

(Qian Bureau No.[2013] 7001), in part by Guiyang Science and Technology Department Platform for innovation plan (2012303),in part by Project supported by the National Natural Science Foundation of China (Grant No. 61262007).

REFERENCES

[1] FlexRay-Protocol Specification V2.1. Rev.A.

[2] FlexRay- EPL-Specification V2.1 Rev B.

[3] M. Mang, L. Chu, L. Feng, "*Multichannel FlexRay simulation system based on FlexRay buses,*" P.R.C Patent CN201310354726, August 14, 2013.

[4] Deb A K, Kordes A, Vermeulen H G H,"*Flexray network runtime error detection and containment,*" U.S. Patent US2014047282A1, February 13, 2014.

[5] Y. Kim, K. Brown, "*FlexRay communications using Ethernet,*" U.S. Patent EP2690826A2, January 29, 2014.

[6] C. Viraktamath, G. Peitz, U. Strauss, J. Ressler, and O. Strobel, "Evaluation of software performance characteristics of flexray communication protocol for its suitability in avionics", In 3rd *CTON Mediterranean Winter Conference*, Angers 2009, pp. 1-4.

[7] C. C. Wang, G. N. Sung, C. L. Wang, P. C. Chen, "Physical layer design for ECU nodes in FlexRay-based automotive communication systems", In 17th *Digest of Technical Papers International Conference on Consumer Electronics*, Las Vegas, 2009, pp.1-2.

[8] C. Z. Yuan, S. Y. Liu, F. Yang, "*Satellite-borne FlexRay bus dynamic segment communication method,*" P.R.C Patent CN103281321A, September 4, 2014.

[9] M. Gossner, O. Augustin, J. Braun, M. Fischer, "*FlexRay Gateway and Method for Operating a FlexRay Gateway,*" U.S. Patent US2013311695A1, November 21, 2013.

[10] F. Luo, Z. Chen, J. Chen, and Z. Sun, "Research on FlexRay communication system", In 4th *Vehicle Power and Propulsion Conference*, Harbin, 2008, pp. 1-5.

[11] GOTZHEIN REINHARD,BRAUN TOBIAS, "*Time and priority-controlled transmission/reception node for FlexRay and LIN,*" U.S. Patent EP2614996A1, July 17,2013.

[12] Y. H. Sheu, and C. M. Ku, "The Intelligent FlexRay Safety Monitoring Platform Based on the Automotive Hybrid Topology Network", In 8th I*ntelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, Piraeus, 2012, pp. 289-292.

[13] X. Chen, Y. Zhang, "*Four-channel FlexRay bus communication module,*" P.R.C Patent CN103034609A, April 10, 2013.

[14] Z. W. Zhang, X. Chen, and X. H. Lv, "The design of internal bus communication node of a distributed flight control computer", *Application of Electronic Technique*, Vol.37, pp.40-43, March 2011.

[15] I. Park, M. Sunwoo, "FlexRay network parameter optimization method for automotive applications", *IEEE Transactions on Industrial Electronics*, Vol.58, pp.1449-1459. April 2011.

[16] T. Pop, P. Pop, P. Eles, & A. Andrei (2008). "*Timing analysis of the FlexRay communication protocol*", Real-time systems, Vol.39, pp. 205-235, 2008

[17] M. Lukasiewycz, M. Glaß, J. Teich, P. Milbredt, "*FlexRay schedule optimization of the static segment,*" In 7th *IEEE/ACM international conference on Hardware/software co-design and system synthesis,* New York, 2009, pp. 363-372.

Received: September 22, 2014 Revised: November 30, 2014 Accepted: December 02, 2014