The Application of Microcontroller MC9S08DZ60 in Automotive CAN Bus Electronic Control Unit

Yi Wang[1,*] and Liren He[2]

[1]*College of Physics and Electronics Science, Guizhou Normal University, Guiyang, Guizhou, 550001, China*

[2]*College of Science, Guizhou University, Guiyang, Guizhou, 550025, China*

Abstract: Being an excellent field-bus, CAN bus has increasingly been used in automotive electronic systems. MC9S08DZ60, an MCU of the S08D Series of Freescale company is an 8-bit universal automotive electronics controller with a CAN module. In this paper, the design of automotive electronic control unit is introduced based on the MC9S08DZ60. The hardware structure and software design processes have been shown. This circuit has characteristics of simple, low cost, high reliability and real-time. It has provided a scientific basis for the development of the CAN communication electronic control unit based on a microprocessor.

Keywords: Automotive Vehicle Network, CAN Bus, Electronic Control Unit, MC9S08DZ60.

1. INTRODUCTION

CAN (Controller Area Network) is a serial communication bus network and has attracted more and more attention for its high performance, reliability, and flexible design. Due to its lower cost, higher real-time processing capabilities, as well as reliably in the strong electromagnetic interference environment, its application range from the automotive electronic control extended to various industrial control fields, such as Industrial automation, a variety of control equipments, transport, medical equipments and construction, environmental control, etc. And it has been certified under international standard (ISO 11898), and has become the most promising field-bus [1].

Using an MCU and a CAN controller is the most common way to design a CAN bus automotive electronic control system. Paper [2] constitutes a core of the intelligent node with a microcontroller 8051 and Intel Corporation's 82527 CAN bus controller. Intel 82527 is the replacement product of Intel Corporation 82526 CAN controller chip, and is the first chip that supports the CAN specification 2.0B standard and extended message format. Intel 82527 is initialized and controlled by microcontroller 8051. In this paper [3], the P87C591 microcontroller and SJA1000CAN bus controller have been used as the core designs of control systems. The microcontroller is responsible for the initialization and control of SJA1000 to achieve the receiving and sending of data communication tasks. The design of such chips is difficult to system integration.

Freescale is one of the world's largest semiconductor companies and a major supplier of the vehicle controller field. The S08D Series 8-bit microcontrollers provide more memories and components on single chips with enhanced performance and reduced power consumption.

Furthermore, with CAN controllers embedded in these MCUs, we can greatly reduce the cost of design, improve the quality of the product and save development time.

In this paper, we have designed an automotive CAN Bus electronic control unit with MC9S08DZ60. The hardware circuit and software program process of the automotive electronics control unit based on CAN bus are introduced. As the data communication between microcontroller MC9S08DZ60 and CAN transceiver MCP2551 only needs two data lines TXD and RXD, it has greatly simplified the hardware design and improved the reliability of the system.

2. THE TECHNICAL PERFORMANCE OF CAN BUS

CAN bus is a serial data communication bus, which was developed by a German company BOSCH, aiming to solve the problems of car control and data exchange. It has the advantages of good reliability, real-time, and low-cost, and is simple and practical. It is especially suitable for automotive work environment and situation with electromagnetic radiation and vibrations. The signal transmission medium of a CAN bus may be twisted-pair, coaxial cable or optical fiber. CAN bus could provide a data transfer rate up to 1Mbps (communication distance being under 40m). The direct transmission distance can be up to 10km at most (the transfer rate is up to 5kbps).

CAN bus communication interface integrates functions of the physical layer and the data link layer of the CAN protocol. It can complete the process of communication data frame, including fill, data block code, cyclic redundancy check and priority discrimination etc. [4]. CAN bus is a priority-based serial communication network, and uses carrier sense multiple conversion/collision avoidance (CSMA / CA) protocol. The start of frame portion of the data transmission

*Address correspondence to this author at the College of Physics and Electronics Science, Guizhou Normal University, Guiyang, Guizhou, 550001, China; Tel: 13007888851; Fax: 0851-3621198; E-mail: wyigz@126.com

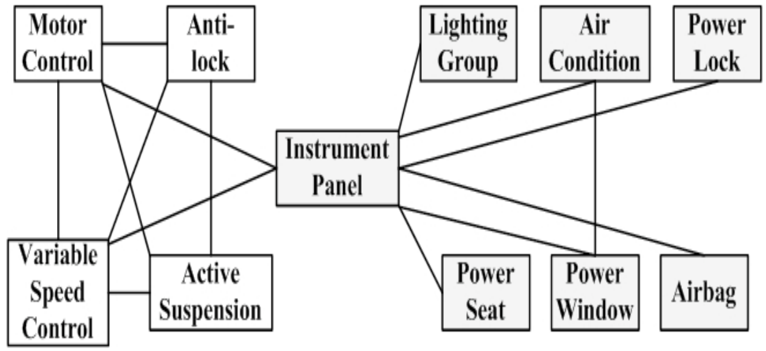

Fig. (1). Traditional Automotive Wiring Network Diagram.

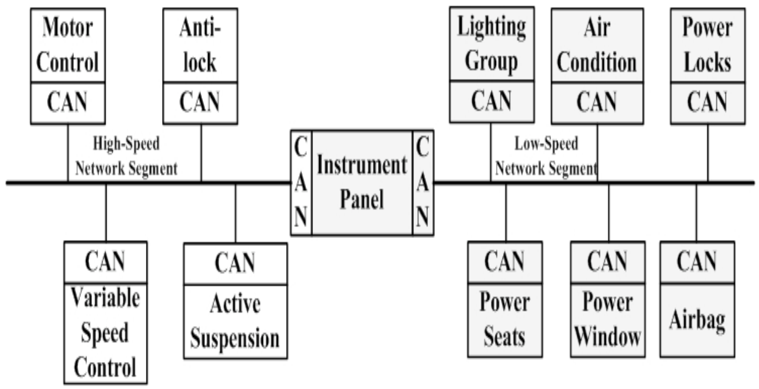

Fig. (2). Automotive CAN Bus Wiring Network Diagram.

in the CAN bus is a data identifier. The identifier can distinguish message and has the ability to indicate the priority of a message (0 being the highest priority). CAN bus has a multi-master way of working. As long as the bus is idle, any node can be active at any moment to send a message to other nodes on the network. If two or more nodes start sending packets at the same time, a bus access conflict arises. Using the principle of bit-wise arbitration, and with the help of frame identifier, low-priority nodes will stop sending data. The nodes with higher priority can continue to send information. During arbitration, each node performs the level comparison between the level sent by other node and bus level through a "and" operation. If they share the same level, the node can continue to transmit. For example, providing that the priority of 0 is high, when a node sends a 1 but detects a 0, this node knows that the higher priority message is in transmission, it stops sending information to the network until the network is idle. The biggest characteristic of CAN protocol is its communication data block coding [5], which replaced traditional station address coding. Theoretically, this method has an unlimited number of nodes in the network. The application ranges of CAN bus have extended gradually from the automotive electronic control to various areas of industrial control. CAN bus has broad prospects.

As CAN bus has been widely used in various industry fields, its communication format standardization also proposed more stringent requirements. In 1991, CAN Bus Specification Version2.0 was developed and published. The technical specification consists of part A and part B. 2.0A gave the CAN packets standard format, while 2.0B gave two formats of the standard and extended. There are two frame formats of CAN 2.0B, and the length of the identifier field was different. The frame which contains 11 frame identifiers is called the standard frame, and the frame containing 29 frame identifiers is called extended frame. The standard data frame format and remote frame format are equivalent. And the extended format is a new feature of CAN2.0B agreement. Our controller design is relatively simple and does not require extended format. And the controller designed must support the standard format without any restrictions. Class D always uses multimedia bus.

3. VEHICLE CAN NETWORK ARCHITECTURE

From a wiring point of view, the traditional vehicle's electrical system often uses a single "point to point" means of communication which has little contact with each other. This will inevitably lead to a huge amount of wiring systems, exacerbated the contradictions of thick wiring harness and limited available space in the car. Therefore, from the cost of materials and work efficiency point of view, the traditional wiring methods will not be able to meet the need of modern car development. Traditional wiring diagram and CAN bus cabling network diagram for the same control nodes are shown in Fig. (**1**) and Fig. (**2**) respectively. From the figures we can see the differences of two methods.

According to the protocol features, the vehicle network committee of the U.S. Society of Automotive Engineers divided the network into four categories A, B, C and D. Class A is a low-speed network for sensor or executive management. Its transmission data rate is typically less than 10kb/s. It is mainly used to improve the ease of application, such as adjusting electric mirrors, electric windows and lighting equipment of a car. The preferred standard of class A is LIN (Local Interconnect Network), which is mainly used for distributed electronic control system in a car, especially for digital communications of intelligent sensors or actuators. Class B is a medium-speed network for the independent control of the sharing of information between modules, such as the interior temperature control, air conditioning unit and display instruments. International standards for class B are CAN bus. Class C is the transmission network for real-time closed-loop control, the bit rate is often more than 125-1Mb/s. Class C is mainly used for the power train and dynamic synthesis and other systems that require high real-time. Class C mainly adopts high-speed CAN bus communication standard ISO11898, SAE J2284, SAE J1939, etc. Class D is a high-performance network of high-speed data stream for oriented multimedia devices. Its bit rate is generally more than 2Mb/s. It is mainly used in CD players, digital televisions, DVD players and LCD devices.

Fig. (**2**) shows a vehicle CAN bus wiring network diagram. The control unit of body systems includes lanterns, power windows, power door locks, air conditioning etc. As most of these parts are low-speed motor and switch devices, their control units are selected as the low-speed bus connection. And to ensure real-time and reliability of communication in power & transmission system, it is separated from the power & transmission system. Meanwhile, the low-speed bus can increase the transmission distance, improve the anti-jamming capability, and reduce cost. The control unit of power system includes motor control system, anti-lock control system and variable speed control systems etc. It directly relates to the running state of the car. So it requires high real-time communication and high reliability. Therefore a high-speed bus is used. Information of various statuses is posted to high-speed bus in the form of broadcast. Each node can access the high-speed bus to receive information on demand at the same time. This approach greatly improves the real-time nature of the system communication.

4. FEATURE OF MC9S08DZ60 INTEGRATED CAN MICROCONTROLLER

A car is a consumer goods and vulnerable to bad environment, so the automotive electronics require highly efficient mechanism and processor. Nowadays, there are various vehicle processors. MC9S08DZ60 is one of S08D series high-performance microprocessor with a CAN interface produced by American's Freescale Company. Due to the characteristics of ultra-small, low-power, low-cost, multi-variety and no requirement for an external CAN controller, it has a wide range of applications. The microprocessor is designed for automotive applications. It supports two ultra-low-power modes. Its clock speed is 40MHz. It has 60KB on-chip Flash memory, 4K of RAM, 2KB of E2PROM, 53 I/O ports, and 24 interrupt pins. Besides, MC9S08DZ60 is embedded on an MSCAN module which supports CAN2.0A/B protocol. The vehicle control system designed with MC9S08DZ60 can work properly in the range of -40°C~ +85°C. Meanwhile the microprocessor has high reliability and low power consumption [6]. The CAN module of MC9S08DZ608 does not only follow CAN bus protocol, but also has its own characteristics. The parameters of MC9S08DZ608 are as follows [7]:

(1) 8bit HCS08 central processor (CPU);

● 40-MHz HCS08 CPU (20MHz bus);

● HC08 instruction set, with additional BGND instructions;

● Support for up to 32 interrupt / reset sources.

(2) on-chip memory

● 60KB Flash memory; ● 2K EEPROM Online programmable memory; ● 4KB RAM;

(3) The power-saving mode

● supports two ultra-low-power stop mode;

● waiting mode to reduce power consumption;

● ultra-low-power real-time clock interrupt.

(4) clock source options

● the oscillator (XOSC) - closed-loop control Pierce oscillator, support a range of 31.25 kHz to 38.4 kHz or 1MHz to 16MHz crystal or ceramic resonator;

● multifunction clock generator (MCG) - PLL and FLL mode (FLL could achieve a deviation internal 1.5% while adopt temperature compensation); internal reference clock source with fine-tuning, external reference clock source that integrated a selectable crystal oscillator or ceramic resonator.

(5) Development Support

● single-wire background debug interface

● on-chip and online simulation (ICE), with the function of bus real-time capture

(6) Peripherals

● ADC, ACMPx, MSCAN, SCIx, SPI, IIC, TPMx, RTC;

(7) input/output

● 53 general-purpose input/output (I / O) pins and a dedicated input pin;

● 24 interrupt pins, each pin could select trigger polarity.

● All input pin with voltage hysteresis and can be configured with up/down device.

● All input pins can be configured with output slope

and drive strength.

(8) Package Options

● 64-pin low-profile quad flat package (LQFP) -10 × 10mm

● 48-pin quad flat package (LQFP) - 7 × 7 mm

● 32-pin quad flat package (LQFP) - 7 × 7 mm

Most interfaces of MC9S08DZ60 have double or even multiple functions. These interfaces can be configured as input/output functions or special interface functions. The 60KB of MC9S08DZ608 internal flash memory is used for storing a user program and data, and they stay read-only state in normal operation. The 2KB of E2PROM is used to save semi-permanent data, such as the configuration and setup information etc. The 4KB of RAM is used for processing stack to save intermediate results and dynamic data, if necessary they can even run on debugger. MC9S08DZ60 is mainly used in the products that need to integrate CAN network.

5. THE CIRCUIT DESIGN OF CAN COMMUNICATION SYSTEM

The circuit of CAN communication system mainly consists of three parts: MC9S08DZ608 micro-controller, MCP2551 high-speed CAN devices and DC power supply circuit. MC9S08DZ60 is the core of the entire system. It mainly receives and sends data, and achieves other communication tasks. We use a CAN bus driver MCP2551 to increase the CAN signal transmission distance and the integrity of signal transmission. The DC power supply circuit provides a stable power supply for the entire system.

MC9S08DZ608 SCM is the core of the CAN bus interface circuit, mainly completes the sending and receiving of CAN bus data, and achieves the decomposition and combination of serial data, and ensures normal smooth flow of communication. MCP2551 is a high-speed fault-tolerant CAN device made by Microchip. It functions as CAN protocol controller and physical bus interface, providing the ability of differential transceiver for CAN protocol controller. It is fully compliant with the ISO-11898 standard, including 24V requirement. It can rate up to 1Mb/s. MCP2551 has three modes of operation: high-speed, slope control and standby. We can select one mode via RS pin, according to the actual situation. In our design, a high-speed work mode was selected. The chip is pin-less and simple to use. The CANH and CANL pins of MCP2551 are connected to the CAN bus through a resistor, which limits the current and protects MCP2551 from flow impact. We parallel two small capacitors one between CANH and ground, and the other between CANL and ground, to filter high-frequency interference and to prevent electromagnetic radiation on the CAN Bus.

A DC power provides a stable power supply for the system. As most of the devices in the system use 5V voltage, we need to convert automotive battery 12V to 5V DC through a regulator. Then we supply power to microcontrollers and other electronic control units. We adopt a basic adjustment tube circuit to achieve the power conversion, while ensure the range of input voltage and stability of output voltage. Filter capacitors are used in the circuit to filter out high frequency interference, and to improve the stability of the power supply. The characteristics of the power circuit are that it can supply a large output current and has a long service life.

6. SOFTWARE DESIGN PROCESS OF CAN COMMUNICATION SYSTEM

Software design is a key to system design. We adopt Freescale Code Warrior development kit and C language for developing our process. C language is flexible and convenient. Code Warrior is an embedded application development software specifically designed for Freescale devices. Code Warrior integrates a variety of tools for microcontroller development, such as integrated development environment IDE, a processor expert database, full-chip simulation, visualization parameters tools, project management, a source file editor, a C compiler, an assembler, connectors, and a debugger. It supports development of third-party software cooperation.

During the code compilation process, Code Warrior can automatically check the error code. By scanning the original program code through the integrated debugger and editor, we are able to detect and reduce the glaring error easily. Then by compiling and linking, we get an executive program. In order to improve the reliability and understandability, the modular programming method is used. In accordance with the functions, the program is divided into different modules, and each module is relatively independent to accomplish a specific function [8]. The program mainly includes three parts: the MSCAN initialization subroutine, the MSCAN message sending subroutine, and the MSCAN packet reception subroutine.

MSCAN initialization is an extremely important part of the system. It is the premise of the system, which determines whether the entire system can work. To properly use the MSCAN module, we must configure the relevant registers of the MSCAN. And in order to avoid damage to the module and CAN agreement contrary, some registers of MSCAN can only be modified in MSCAN initialization mode. So the initialization of MSCAN is very important. The right process is as follows: enter the initialization mode, set the baud rate register, set the control register, set the packet filtering register, set the interrupt register, clear the INITRQ to leave the initialization mode and get into the normal operating mode and so on. Setting packet filtering register is an important part of the initialization module. During CAN bus transmission, there may be a lot of data in the communication process. As each frame might compare the ID with all the messages, it will spend a lot of CPU resources. In fact, many of the messages of the node are meaningless, so we should refuse the useless match to save CPU processing time and to reduce the load on the CPU. After setting packet filtering register, the CPU just needs to handle the packet whose ID is matched. Note should be initialized, if MSCAN module starts working before initialization, we should confirm the transmission queue is empty and make MSCAN work in sleep mode, otherwise it will result in an error, and affect other CAN bus nodes. Initialization process is as the following:

(1) If MSCAN module is running, make it work in sleep mode.

(2) Enter the initialized mode;

(3) Configure appropriate registers, to decide baud rate of the register packet filtering register;

(4) Clear INITRQ, return to normal mode;

(5) Exit sleep mode if it has entered previously.

(6) Set the rest MSCAN register, such as the CANCTL0, CANRIER, CANTIER.

The process is shown in Fig. (**3**).

The sending subroutine of MSCAN packet: In order to send message to CAN bus, first we need to select an available transmit buffer, and then we need to write the data to the transmit buffer, and finally to set the send flag of the buffer. MSCAN has three transmit buffers. To optimize the real-time performance of the system, we establish a number of packets in advance. In sending process only one address is

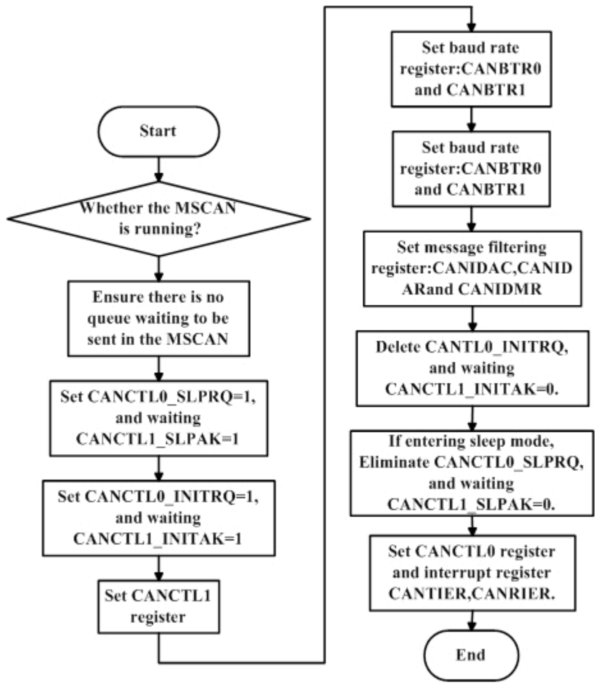

Fig. (3). MSCAN Initialization Subroutine Flowchart.

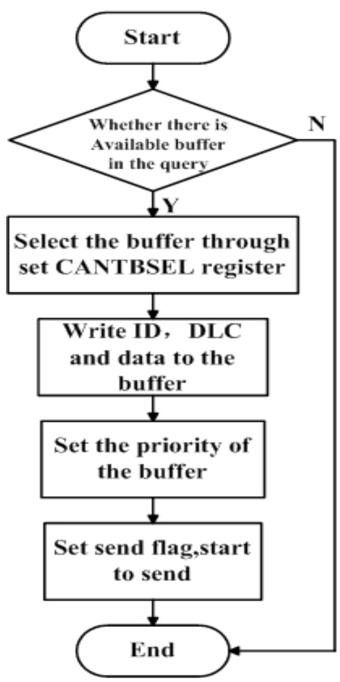

Fig. (4). MSCAN Packets Sent Subroutine Flowchart.

needed, this greatly simplifies the software process and can save memory address space.

MSCAN subroutine mainly includes:

To detect whether there is available buffer in the query, to select a transmit buffer, to set the remote frame flag, to write data, to set priorities, and to initiate the transmission process.

Before sending data, it is needed to detect whether the value of (CANTFLG & 7) is 0, to determine the availability of buffer. If the value is 0, it indicates that the three buffers

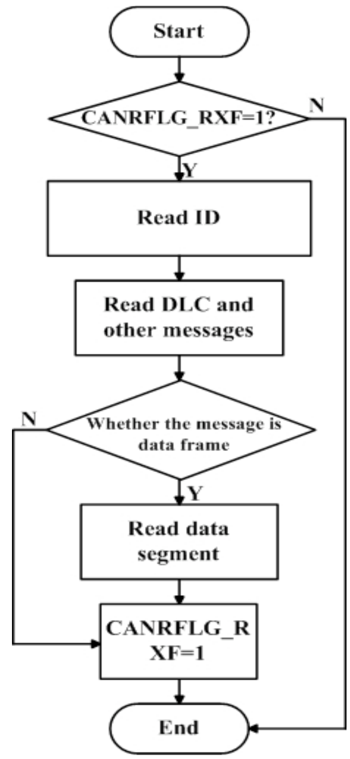

Fig. (5). MSCAN Packet Reception Subroutine Flowchart.

are occupied. If there are available buffers, the CPU will select a buffer to transmit data. When there is more than one buffer that is idle, the lowest number of buffer is selected. After selecting the buffer, identifier and data length, data is written to the buffer zone by CPU. And the sent flag is set. The buffer is marked as the sending state. If more than one buffer is ready to be sent, then the MSCAN module determines which buffer is transmitted first via the internal priority. The process is shown in Fig. (**4**).

The reception subroutine of MSCAN packet: The data reception is similar to data transmission, and it is done automatically by the MSCAN controller. MSCAN has five receiving buffers, similar to the transmit buffer. This mechanism greatly simplifies the preparation of applications, and improves the efficiency of the received frame. The received data is read from the received buffer and is treated appropriately. There are two ways in receiving data. One is the query mode and another one is interrupt mode [9]. When using query mode, all data must be read before the flag is cleared, including ID, data field and time marker lamps, otherwise it will lead to a receiving error.

The state of foreground received buffer is indicated by RXF bit in CANRFLG Register. When the buffer data is received correctly, RXF bit is set to 1. On opening the receiver interrupt, the CPU will automatically jump to execute the user's interrupt code. The user's interrupt service routine can read data from the front buffer, set RXF flag in response to the interrupt and release the front buffer. If new CAN frames are received prior to release of the front buffer, they will be stored in the background buffer that is available. If an error occurs during transmission, the data in the background buffer will be overwritten by new data, and will not be moved to the front buffer. The data flow of receiver is shown in Fig. (**5**).

7. CONCLUSIONS

Because of the high-performance, high-reliability and unique design, CAN bus has been widely used in advanced automotives [10]. Now various electronic control units in a car are able to share information resources through CAN bus, so the coordination and match between the control units are much more intact. To improve the safety and stability of the car, some of the well-known international car manufacturers have their products using CAN bus technology and research in this area is growing more and more and becoming more mature, while the research applications start late in domestic. Based on MC9S08DZ60, the CAN bus electronic control unit can be highly integrated providing stability, and strong anti-electromagnetic interference. Both hardware and software aspects of the embedded CAN controller MC9S08DZ608 are introduced in this paper. The circuit design and software development of CAN bus control unit are discussed. There are certain practical values for the development of CAN bus system.

CONFLICT OF INTEREST

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

ACKNOWLEDGMENTS

This work was financially supported by the National Natural Science Foundation of China (Grant No. 61262007), Guizhou Science and Technology Department School Cooperation Project (Qian Bureau No: [2013] 7001) and Guiyang Science and Technology Department Platform for innovation plan (2012303).

REFERENCES

[1] L. Cheng-zhi, D. Guo-Liang, Y. Liang, "Analysis for real-time performance of improved CAN protocol", *Comput. Technol. Dev.*, vol. 22, no. 7, pp. 81-84, 2012.

[2] G. Fei-fei, W. Yun, "Design of car lamp control system based on CAN bus", *Mech. Eng. Autom.*, vol. 4, pp. 152-154, 2008.

[3] G. Jun-fei, L. Jun-wei, Z. Yu-ying, "Development of the lighting control system for electric vehicle based on CAN bus", *Agric. Equipment & Vehicle Eng.*, vol. 3, pp. 15-18, 2009.

[4] G. Xiao-song, W. Zhen-ye, Y. Chuang-qiang, "Research of the fault tolerant redundant technology based on CAN bus", *Comput. Measur.Control.*, vol. 17, no. 1, pp. 60-62, 2009.

[5] J. Jing-hong, Y. Fang, C. Guang-da, "application of lighting control system to car based on CAN bus", *J. Northeast Agric. Uni.,* vol. 40, no. 1, pp. 108-111, Jan 2009.

[6] MC9S08DZ60 Data Sheet [OL].2007.10

[7] L. Feng, S. Ze-chang, *Principle, Design and Application of Automotive CAN Bus System.* Beijing: Publishing House of Electronics Industry, 2011.

[8] N. Zhang, F. Xin, P. Mao-hui, "Design of Electric Vehicle Light Contol system Based on CAN Bus", *Transducer Micro-syst Technol.*, vol. 30, no. 12, pp. 82-84, 2011.

[9] Y. Guo-guang, L. Jin-gao, "Application of CAN Bus Technology in Vehide System", *Comput. Sci.*, vol. 36, no. 4A, pp. 276-278, 2009.

[10] L. Yu, Z. Yi-min, S. Gui-qiu, "The Data Transmission Reliability Analysis in Automotive CAN Bus", *Mach Des Manuf*, vol. 12, pp. 130-132, 2011.

Received: July 02, 2013 Revised: July 07, 2013 Accepted: July 14, 2013