The Design and Implementation of Car Alarm Bus Testing System

Cheng Wei, Wang Zhixue, Zhang Xiaobo
Shandong Provincial Key Laboratory of Automotive Electronic Technology
Institute of Automation, Shandong Academy of Sciences

Zhang Yun
School of Electrical Engineering
Shandong University

***Abstract*-The mainly problem this thesis will solve is how to meet the time limits in functional testing of car alarm based on CAN bus. To effectively solve this problem, the bus testing system do a scheduling every 2ms with timer. Thus control of resources can be transferred to important tasks. By this method, the system completely satisfies the time limits in functional testing of car alarm. In practical, the system plays an important role in the production process, which possesses complete functions, good stability, friendly software interface, and so on.**

Keywords-ECU, CAN Bus, BSI

I. INTRODUCTION

The self-developed car alarm with CAN bus is applied to the foreign model, which has been favorable at home. It passes through the function and performance test by foreign car maker and professional development company. This production development process complies with TS16949 which ensure the production quality. That has begun on batch production and put the products into use in March 2008. Theory and Practice of setting test in automobile demonstrate that this product has stable performance and reliable quality.

II. DESIGN PRINCIPLE OF THE ALARM

A. Design The Network Structure

The Controller Area Network(CAN) is a serial communicating network which can effectually sustain the distributed control or real-timing control, which applied most popular in worldwide [1]. In the early 1980s, CAN is originally developed by the German company Robert Bosch in order to solve data exchange between equipment-to-equipment in contemporary automobiles. The emergence of CAN bus so that the networked control system possible, is recognized internationally as a field-bus protocol. CAN Bus has widely used in the industrial control fields because of its outstanding reliability and performance/cost ratio[2-4]. CAN Bus is now a popular form of industrial bus, which is going to maturity now.

CAN bus are widely used in communication of the control systems between checkpoint and execute institution, a CAN debugger with good function and convenience to use could give CAN bus developers a great deal of convenience.

As awareness of the requirements of constantly improving the quality of automotive safety and intelligence, electron of car is increasingly becoming more complex. CAN bus technology is widely applied in systems or vehicle to achieve complete customer's satisfaction. As a kind of reliable car computer network bus, it widely used in many advanced vehicles[5-6]. And it provides great convenience for high-efficient sharing of resources and information of vehicle electronic control units. One approach, in particular, is to use CAN bus with all the advantages, it offers in terms of less complex wiring, reducing the amount of acoustic sensors, increasing system reliability and performance optimization. Having been aware of the great advantages of CAN compared to traditional data communication means, more and more engineers begin to apply CAN bus in all kinds of automotive electronic control system applications.

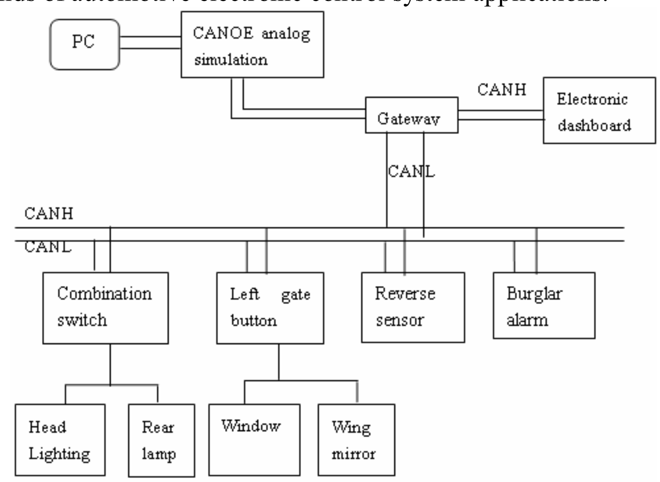

Figure 1. The networks topological structure of the system

The system which uses CAN bus and LIN bus is designed in the paper. Again, CAN bus is built on ISO 11898-3 standard, and LIN bus is based on LIN 2.0. The networks topological structure of the system is shown in Figure 1.By gateway can convert low- speed CAN into high-speed CAN. Four nodes are realized by low- speed CAN, including car light combination switch, the left gate, reverse sensor, burglar alarm and so on. Among them, combination switch and the left gate serve as primary nodes of LIN and form LIN Network. And the burglar alarm functional prototype has been designed at present.

B. Design Principle

As shown in Figure 2, alarm system has five main components, power management module, MCU, bull horn, hood switch and CAN bus communication module. It can warn the hood, four doors, sunroof, the back window, trunk, and so on. As long as any one state has changed, alarm immediately notify the BSI module through CAN bus and beep alarm. Also alarm with fault self-diagnostic features can record fault codes which are detected and transmit to BSI.

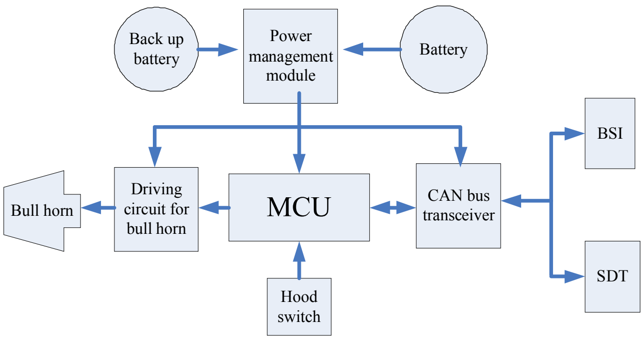

Figure 2. The alarm block diagram

Its main functions are as follows. Firstly, it can receive BSI monitoring alarms. Alarm as soon as intrusion message is received. Secondly, it can monitor mains power (storage battery). It is automatically switched to the standby power and generates alarm signals when the main source is turned off. Thirdly, it can monitor state of engine switches. When the engine switch is trespassed, it can generate engine switch intrusion alarm. Furthermore, it can monitor state of CAN bus. When CAN bus got cut off, it can generate bus intrusion alarm signals. In addition, the alarm has been designed with self diagnostic capability.

The system fully considers any conditions the alarm might encounter, such as power supply invasion, bus invasion, and so on. It must deal with it according to such illicit operations. The real-time performance is guaranteed by the interrupt operation.

According to the function of this alarm product, the matching low-speed and fault-tolerant CAN bus tester facilitate automatic test, which was fully self-development.

III. TECHNICAL SOLUTIONS ON ALARM CAN BUS TESTER

The low-speed and fault-tolerant CAN bus tester was designed to settle full-scale test during alarm batch production. The hardware platform of tester adopts AT91RM9200 processor made by Atmel, which give more stable performance and typically run faster than other singlechip because the highest frequency is 180 MHz. And the main program is realized by using modularized software design method which adopts open source embedded Linux environment. The next, both hardware and software design are discussed, and software design is given in detail.

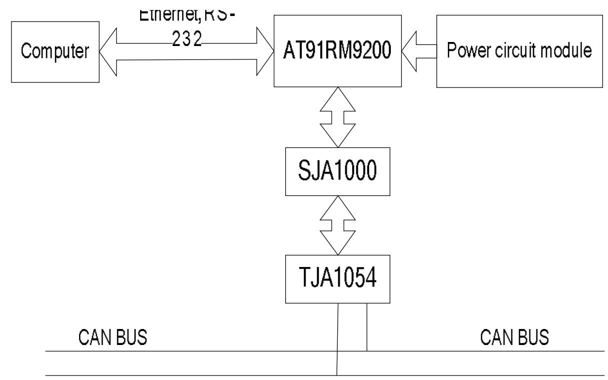

Figure 3. The tester hardware block charts

A. *Hardware* Design

Hardware design is mainly divided into five parts as shown in Figure 3., which AT91RM9200 processor as core.

1）The system extends 32MB flash and 32MB SDRAM in order to realize the storage and operation of Linux operating system and the application software.

2）CPU is connected to SJA1000 by buses, which realize protocols in the CAN link layer. And it realizes functionalities of the CAN physical layer by means of TJA1054.

3）The peripheral communication interface realizes communication functionalities of Ethernet and RS-232 that is convenient for remote data transmission and control.

4）The use of 12~24 DC power supply up to 1A, safe and reliable. With the help of DC-DC chips, it can be suitable for application of 3.3V or 1.8V mains voltage.

B. *Software Design*

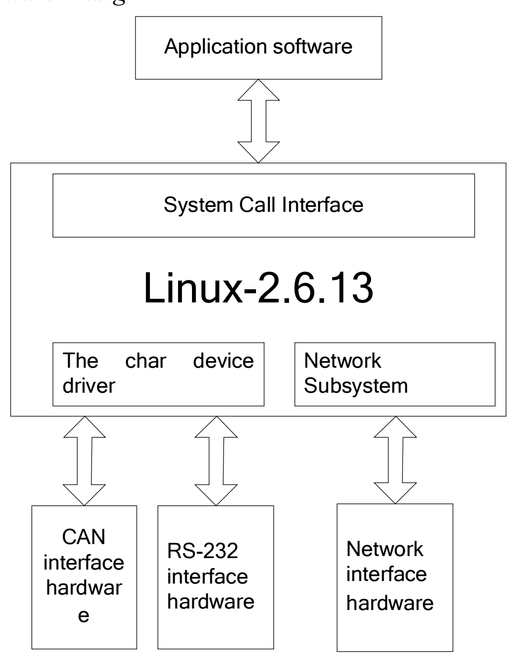

Figure 4. The tester software design framework

The software platform of tester is derived from the Linux 2.6.13 operating system which has been successfully ported, thus enhancing the running stability and safety of the system. The software architecture is clearly divided into two parts which are the Linux drivers and the applications. The drivers complete the operations of hardware, and the applications mainly fulfill the CAN application layer, network application layer and RS-232 application layer. To ensure the drivers run reliably and stably, the hardware must transmit and receive data by the method of interrupt processing. So make the tester possible to accurately process large amount of data in time. If fails to send, it can cancel in time or send again. Software diagram is shown in Figure 4.

C. Key Technology

The time limits in alarm protocols are very strict, so it is required that the system must poll every 2ms. The tester must be able to test this feature thoroughly,and if it fails to send BSI data, it must be able to cancel within 17ms[7]. So it is required that the system must do a scheduling every 2ms.However, as we know, the system scheduling is designed every 10ms in Linux operating system. We must modify Linux operating system to achieve scheduling every 2ms. To solve this problem, the common approaches are as follows: 1) using real-time Linux kernel. 2) Modifying preemptive kernel patching and low-latency patching. The experimental result shows that preemptive kernel patching can highly improve the system response time. By low-latency patching, long-running kernel operation made a concession to critical task and release resources. 3) According to the principles of the operating system, the system do a scheduling every 2ms with timer[8-11]. Thus control of resources can be transferred to important tasks[12].

After comparison revealed that any of the above three methods can be able to meet the requirements of the system. However, the difference between cost and workload is big. Cost of the first schemes, although with minimize workload, will be high. If the second scheme is adopted, LInux-2.6.13 kernels without a doubt require patching but with no guarantee of reliability. It has low cost, but work load is heavy. The last method has the characters of simple principle, less workload and low costs, etc. The shortage of the last scheme is that the frequent scheduling must overburden the system. Especially in more tasks situation, task with lower precedence is difficult to obtain control of the CPU. Based on the preceding analysis and taking the features of the system into account, the third option is like the best option. Considering the situation of the system, the shortage of the last scheme does not exist because there are only 3 tasks in this system at most. In fact, only CAN task is strict in needs of time, the other two tasks are not strict requirements on time. So the system adopt the last solution to design. Currently it has been shown that this system is stable and credible in performance, which is completely satisfied for time demands while testing CAN bus functions.

D. Test Records

The tester also includes a set of PC software, which can operate the tester through the human-machine interface. The software test kit is composed by power-up, alarm, LED, ARM engine switches, battery, horn, and so on.

In particular, the alarm with CAN bus passes through the function and performance test by foreign car maker and professional development company. The three CAN test records are shown in Figure 5. And the test is conducted by NCS Corporation which is specializing in testing CAN bus protocol in France. The Functional testing fie is shown in Figure 6.

4 Validation history

Author	Project code	Date	ALM Identification
A. BENACHOUR	003-06-0067-1	2006/11/03	Validation for ALM configuration: • **ECU description :** ALM • **Customer :** HISENSE • **Project name :** T53/B53 • **Hardware release :** 3.0 • **Software release :** 8I
A. BENACHOUR	098-07-0001	2007/02/02	Validation for ALM configuration: • **ECU description :** ALM • **Customer :** HISENSE • **Project name :** T53/B53 • **Hardware release :** 4.0 • **Software release :** 8I
I. DELBASSEZ	098-07-0002	2007/04/26	Validation for ALM configuration: • **ECU description :** ALM • **Customer :** HISENSE • **Project name :** T53/B53 • **Hardware release :** PIC-ALARM V5.0 • **Software release :** v9.10

Figure 5. The three CAN test records by NCS Corporation

VANUXEM Freddy 01 57 59 92 32 / 20 92 32 Fax : 01 57 59 31 44 freddy.vanuxem@mpsa.com Site : Vélizy	Référence : **AEEV_IVE07_0340** Date de création : **19/02/07**	PSA PEUGEOT CITROËN **DPTA/DMOV/ELE/QCE** Applicable au(x) projet(s) : **[B53]**

COMPTE RENDU DE VALIDATION DE L'ORGANE

Module Alarme Sirène CHINE

DESTINATAIRES		EN COPIE	
DMOV/ELE/QCE/EQVV	G.THOMAS	DMOV/ELE/QCE/EQVV	H.DE LANGLE
DMOV/ELE/QCE	F.LUBIN		
DMOV/ELE/EEH	V.ROO		
DMOV/ELE/EEH	M.HANNOURI	POUR INFO	
		DMOV/ELE/QCE	P.FUNCK

Figure 6. The Functional testing fie

IV. SUMMARIES

The low-speed and fault-tolerant CAN bus test system designed in the thesis has the advantages such as high precision, good stability, friendly software interface and so on. And for users, testing alarm can simply use PC software without having an insight into the details of hardware and software design. This not only provides the manufacturers with the great practical value, but also the workers with easy operation. In this regard, the tester also plays a very important role on the production line.

REFERENCES

[1] DENG Zun-Yi, NING Yi. Research and Applications of CAN Bus in Automobile Control System Development & Innovation of Machinery & Electrical Products，2010, 23(4): 137-139.
[2] HUO Shu Zhen. Development and research of multi-media product for vehicle based on ARM11, Microcomputer & Its Applications,2010,8:66-68.
[3] Jorg Kaiser, et al. "Evaluation of a Hybrid Real-time Bus Scheduling Mechanism for CAN". Workshop on Parallel and Distributed Real-Time Systems. San Juan, Puerto Rico. 1999.
[4] Mendoza, P., et al. "Developing CAN based networks on RT-Linux ". Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation ETFA'2001. IEEE Press. Antibes Juan-Les-Pins, Francia, 2001
[5] P. Pedreiras et al. "A Practical Approach to EDF Scheduling on CAN," Proc. IEEE Workshop Real-Time Distributed Embedded Systems (WRTDES), 2001.
[6] Pedreiras P., et al. "Combining Eventtriggered and Time-triggered Traffic in FTT-CAN: Analysis of the Asynchronous Messaging System". Proc. of WFCS 2000, 3rd IEEE Workshop on Factory Communication Systems, Porto, Portugal, September 2000.
[7] YUAN Wen-ju,ZHAO Kong-xin,LIU Li-wei,QIAN Feng.Wireless remote car alarm based on GPRS, Journal of Changchun University of Technology,2008,29(4):391-394.
[8] CHEN Jingbo,HU Jinchun. Distributed fault-tolerant system based on CAN bus, Journal of Tsinghua University(Science and Technology,2009,49(7):1023-1027.
[9] Laprie J C, Arlat J, Beounes C, et al. Definition and analysis of hardware-and software-fault-tolerant architectures [J].Computer, 1990,23(7): 39 - 51.
[10] Kopetz H, Damm A, Koza C, et al. Distributed ault-tolerant real-time systems: The Mars approach [J].IEEE Micro, 1989,9(1): 25 - 40.
[11] Tanenbaum A S, Steen M V. Distributed Systems:Principles and Paradigms [M]. Beijing: Tsinghua University ress, 2002.
[12] CAI Hong-yu, LI Yun, CHEN Li-rong. An In-vehicle network software solution and design based on CAN bus, Microcomputer Information, 2010, 26(7):111-113.

Published by Atlantis Press, Paris, France.