Review Paper on Design and Implementation of CAN Bus Application Layer Protocol for Vehicle Control System

[1]Heena Rajnikant Brahmbhatt, [2]Prof. Madhuri D. Thakkar
Department of Electronics & Communication Engineering
Parul Institute of Technology, Limda.
[1]heenarbrahmbhatt5789@gmail.com,[2]madhuri4288@gmail.com

Abstract - **Present Automobiles are being developed by more of electrical parts for efficient operation. Generally, many vehicles were build with an analog driver-vehicle interface for indicating various vehicle statuses like speed, fuel level, Engine temperature etc., This paper presents the development and implementation of a digital driving system for a semi-autonomous vehicle to improve the driver-vehicle interface. It uses an ARM Cortex based data acquisition system that uses ADC to bring all control data from analog to digital format and visualize through LCD. The communication module used in this project is embedded networking by CAN which has efficient data transfer rate. It also takes feedback of vehicle conditions like Vehicle speed, Engine temperature etc., and controlled by main controller.**

Index Terms - ECU (Engine Control Unit), CAN (Controller Area Network), Embedded C, serial communication, security system

I. INTRODUCTION

With rapidly changing in computer and information technology, much of the technology finds various ways into vehicles system for different applications. Vehicles are undergoing dramatic changes in their capabilities and how they interact with the drivers. Although some vehicles have provisions for deciding to either generate warnings for the human driver or controlling the vehicle autonomously, they usually must make these decisions in real time with only incomplete information. So, it is important that human drivers still have some control over the vehicle. Advanced in-vehicle information systems provide vehicles with different types and levels of intelligence to assist the driver. The introduction into the interior design of vehicle systems has allowed an almost symbiotic relationship between the driver and vehicle by providing a sophisticated & intelligent driver-vehicle interface through an intelligent information network.

Proposed system will apply for the use of two wire system from CAN protocol to interconnect between control terminals of the system. There will be several nodes designed in the system including main controller, different HMI modules such as sensors, keypad, touchpad etc. and few others. Each node will consists of its own controller and CAN transceiver. The communication between the nodes and controller will make use of CAN bus protocol. This paper discusses the development of such a control framework based on CAN for the vehicle which is called the digital-driving behavior, which consists of a joint mechanism between the driver and vehicle for perception, decision making and control.
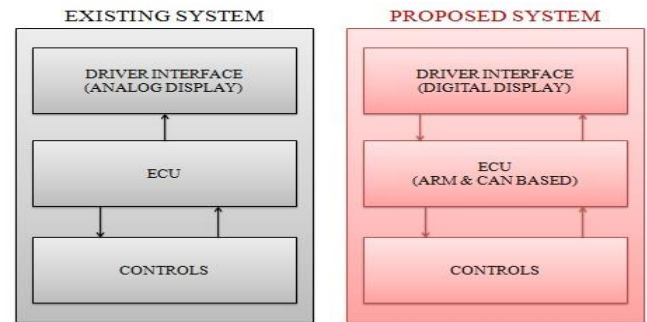

Figure 1: Existing and Proposed vehicle control system

Fig.1 shows the vehicle control of existing and proposed system. A vehicle system was generally build with an analog driver-vehicle interface for indicating various parameters of vehicle status like temperature, pressure and speed etc. To improve the driver-vehicle interface, an interactive digital system is designed. A microcontroller based data acquisition system that uses ADC to bring all control data from analog to digital format is used. Since the in-vehicle information systems are spread out all over the body of a practical vehicle, a communication module supports implementation of CAN frame for one stop control of the vehicle through the master controller of the digital driving system.

II. LITERATURE REVIEW

A. Benefits of Network Architecture

The technology of designing machines according to network architecture started some ten years ago. At that time, the first micro-controllers available in the market made possible to design the advanced electronics small enough to be integrated into mechanical machine parts, for e.g., a servo controller could be physically integrated into a standalone actuator. The basic idea of the network approach is to divide a machine into Functional Units, for e.g., actuator and sensor units. Each Functional Unit has its own micro-controller and the mechanics needed to perform a specific task. A complete machine system is then constructed out of several functional units connected to an information network, often called a Controller Area Network, and a power network. Through these networks, the Functional Units get all information and power needed to make them work. So that, their performance can be equivalent or better compared to traditionally designed machine.

The network architecture has many advantages in comparison with a standard design using a central microprocessor to carry out the control work:

- Fewer wires in the system make the wiring cheap and easy. In many cases, the wiring can be reduced by 90% or more.
- It is a flexible system. Functional Units can be added or removed in a simple way.
- Simple maintenance. The electronics in many Functional Units can be identical and fully interchangeable.
- Each Functional Unit can be developed and tested individually according to a specification given from the system demand.
- The network architecture is very well suited for project oriented development work.

B. CAN Open the Way for Machine Controller Area Networks

In 1988 Intel and Bosch presented a high speed serial protocol designated CAN. This was aimed for use in cars but it also turned out to be very well suited for industrial real-time machine control systems. CAN have many good features:

- It provides faster data rate of communication. 1 Mbps if the bus length is less than 40 meters.
- Non-destructive collision detection by bitwise arbitration.
- Any message has a specific priority on the bus.
- The messages have a predictable maximum latency time. A trigger message with no data and the highest priority can have a maximum latency time of 54 μs on the bus if 1 Mbps transfer rate is used.
- Messages can be sent point to point or multi-casted.
- Remote messages supported. A Functional Unit can always be prepared to immediately transmit the latest available data upon request from any other unit.
- 2032 (Standard CAN) or 536.870.912 (Extended CAN) different messages are available, each containing 0 - 8 bytes of data.
- Powerful error detection and handling. If there is one corrupted message in 1000 transmissions the total residual error probability amounts to $8.5 * 10{-14}$.
- Programmable transfer rate.
- Programmable output driver configuration.
- ISO STANDARD (ISO 11898)

With CAN, we have at last got the missing piece required for a new development era. The protocol is almost perfect for high speed real-time control systems. It is possible to transmit full messages with eight byte data at a rate of 7600 messages or 18000 trigger signals per second over the network. The CAN-Controllers take care of all error checking and if a message is corrupted for any reason it will automatically be re-transmitted at once.

III. OBJECTIVES

The research and development of this project was driven by robustness of the CAN communication and its practicality of application. Therefore, the completion of this project is expected to fulfil the following objectives:

1. To apply a CAN bus system in control and data acquisition system in industry.
2. To build a model of a system utilizing CAN Bus protocol.
3. To utilize the capabilities of a CAN communication as well as the network system.

4. To design and develop a new module control system based on the implementation of CAN Bus communication.

IV. HARDWARE STRUCTURE

The hardware structure mainly integrates the CAN bus controller, ARM as the main control module, LCD display to provide Digital interface and other accessories.

A. CAN bus

1. CAN Bus in an Automobile

CAN is a LAN (Local Area Network) controller CAN bus can transfer the serial data one by one. Fig 2 shows a typical architecture from an automotive. All participants in the CAN bus sub-systems are accessible via the control unit on the CAN bus interface for sending and receiving data. CAN bus is a multi-channel transmission system. When a unit fails, it does not affect others. The data transfer rate of CAN bus in a vehicle system is different. For example, the rate of engine control system and ABS is high speed of real-time control fashion of 125Kbps to 1M bps. While, the rate of movement adjustment is low-speed with transmission rate of 10 to 125Kbps. Others like multimedia systems use medium-speed rate between the previous two. This approach differentiates various channels and increases the transmission efficiency.
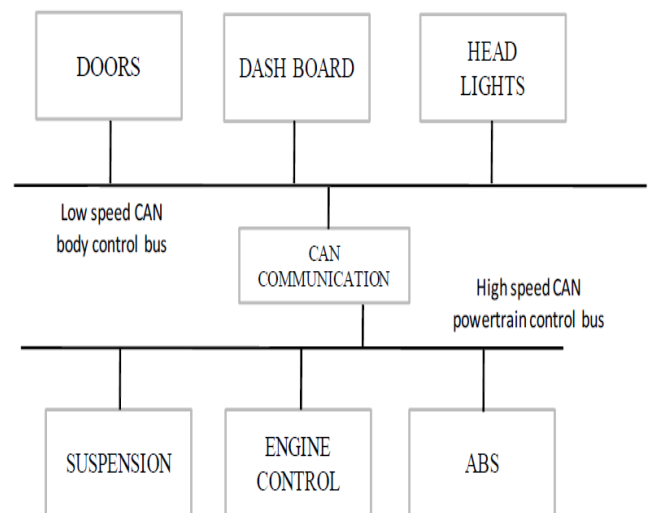

Figure 2: CAN bus system in an Automobile

2. CAN Bus for vehicle drive control System

A typical drive system with the control unit has electronic fuel injection system, automatic transmission systems, antilock breaking system (ABS), airbag systems etc. These units are the core components in a modern car system. They are sensitive for time and closed to the reliability and security of the entire system. As each control unit for real-time requirement is based on the data update rate and the control period varies, in order to meet the real-time requirements of each subsystem, it is necessary to achieve the implementation of public data sharing, such as engine speed, wheel speed, and throttle pedal location. The contents include the completion of speed measurement, fuel measurement, A/D conversion, the calculation conditions, the control modules and a series of

processes. That means the sending and receiving data in 1ms must be completed within the electrical control of gasoline in order to achieve real-time requirements. Therefore, the data exchange network must be a priority-based competitive mode, and has a very high speed communication fashion.

3. CAN Bus for accessories control system

CAN bus for vehicle system is a leading control network that connects several objects. They are central controller, 4-gates controller, memory modules and other components. There are several items controlled by the CAN bus. They are locker, windows, luggage locker, mirrors and interior dome light. In the case of remote control, it involves the remote control signal receiving and processing the anti-theft and warning systems.

B. Main control module

1. ARM Cortex Architecture

The LPC1769/68/67/66/65/64/63 are ARM Cortex-M3 based microcontrollers for embedded applications featuring a high level of integration and low power consumption. The ARM Cortex-M3 is a next generation core that offers system enhancements such as enhanced debug features and a higher level of support block integration.

The LPC1768/67/66/65/64/63 operates at CPU frequencies of up to 100 MHz. The LPC1769 operates at CPU frequencies of up to 120 MHz. The ARM Cortex-M3 CPU incorporates a 3-stage pipeline and uses Harvard architecture with separate local instruction and data buses as well as a third bus for peripherals. The ARM Cortex-M3 CPU also includes an internal pre-fetch unit that supports speculative branching.

The peripheral complement of the LPC1769/68/67/66/65/64/63 includes up to 512 kB of flash memory, up to 64 kB of data memory, Ethernet MAC, USB Device/Host/OTG interface, 8 -channel general purpose DMA controller, 4 UARTs, 2 CAN channels, 2 SSP controllers, SPI interface, 3 I^2C-bus interfaces, 2-input plus 2-output I^2S-bus interface, 8-channel 12-bit ADC, 10-bit DAC, motor control PWM, Quadrature Encoder interface, four general purpose timers, 6-output general purpose PWM, ultra-low power Real-Time Clock (RTC) with separate battery supply, and up to 70 general purpose I/O pins.

The LPC1769/68/67/66/65/64/63 is pin-compatible to the 100-pin LPC236x ARM7-based microcontroller series.

C. Other accessories

IR sensor, Motor speed sensor, Alarm, Temperature sensor, Pressure sensor, Level sensor, and LCD display.

Fig 3 shows the block diagram of CAN vehicle control system. It consists of one master node and two slave nodes.ARM as the master controller (Engine Control Module) which controls the vehicle status with various sensors. Two PIC ICs are used as slave nodes to receive the inputs of vehicle status. The communication between these sensors is done by using CAN controller. Slave controller receives the signals from vehicles like pressure, temperature, fuel level, and IR obstacles etc., send to master controller with high speed rate. Master controls the status of vehicle and sends the feedback to operator panel by providing digital information's via LCD display and alarms. Here Operator interface is digital type. By this operator can easily see the signals and able to control the vehicle. IR obstacle sensor helps in identifying the

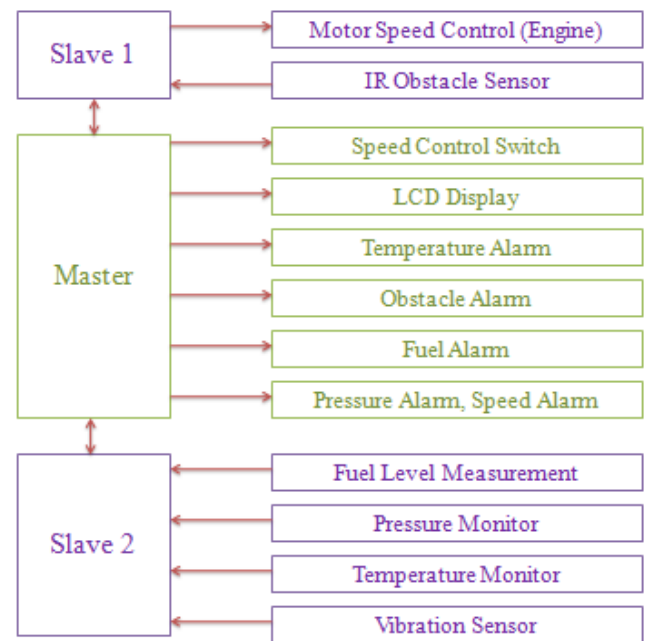obstacles presence around the vehicle.

Figure 3: Block diagram of proposed system

V. SOFTWARE STRUCTURE

In order for the hardware to function, the firmware code for the system has to be written. Theoretically, software that resides in the non-volatile memory and handles the operation as well as function of a system is known as firmware. The firmware holds the information that the Microcontroller needs to operate or run. Thus, it needs to be free of bugs and errors for a successful application or product. There are various types of software that could be used to program a ARM controller. Program can be written in a variety of languages such as C, Basic, Pascal or even Assembler. In this project, the program would be written in C language to generate the required firmware for the system.

The program code is written in C language using CCS C compiler (by Custom Computer Services) to generate the hex file. This hex file is then downloaded into the microcontroller for it to function as programmed. There are several advantages of using CCS as the compiler because functions related to CAN system are already available and ready to be used. This reduces the time to write the program code as well as ease the development process.

The CAN protocol supports two message frame formats. The "CAN base frame" supports a length of 11 bits for the identifier, and the "CAN extended frame" supports a length of 29 bits for the identifier.

The vehicle control system is programmed using the Embedded C and debugged with keil-4 tool.

Based on the CAN Standard being used either standard or extended format, the data being transmitted or received through the CAN bus has to be programmed according to the protocol. Figure 4 shows the flowchart of proposed system.

Figure 4: Flowchart for program

VI. CONCLUSION

This project introduces an embedded system with a combination of CAN bus systems. Vehicle control is an important criterion of modern technology. With the rapid development of embedded technology, high-performance embedded processor is penetrated into the automobile industry. CAN is having low cost, high reliability and other features to meet the needs of the modern automobile industry. The proposed high-speed CAN bus system solves the problem of automotive system applications, also has certain practical values and significance. With the use of ARM cortex as the main controller, it provides high-speed reduction of CAN bus communication control networks and any device control system. Hardware platform of the system will achieve full sharing of data between nodes to control any module and enhance their collaborative work. This system features efficient data transfer among different nodes in the practical applications. It also provides more security to the system and also increases system robustness.

REFERENCES

[1] Kumar, M. A.Verma, and A. Srividya, Response-Time "Modeling of Controller Area Network (CAN). Distributed Computing and Networking,
Lecture Notes in Computer Science Volume 5408, p 163-174, 2009.

[2] Tindell, K., A. Burns, and A.J. Wellings, Calculating controller area network (CAN) message response times. Control Engineering Practice, 3(8): p. 1163-1169, 2005.

[3] Li, M., Design of Embedded Remote Temperature Monitoring System based on Advanced RISC Machine. Electrotechnics Electric, 06, p. 273, 2009.

[4] Prodanov, W., M. Valle, and R. Buzas, A controller area network bus transceiver behavioral model for network design and simulation. IEEE Transactions on Industrial Electronics, 56(9): p. 3762-377, 2009.

[5] ISO (1993). Road Vehicles: Interchange of Digital Information: Controller Area Network (CAN) for High Speed Communication. ISO 11898:1993.

[6] B.Gmbh, "CAN specification" vol 1 Version 2.0, 1991.

[7] Pazul, "Controller Area Network (CAN) Basics", Microchip technology Inc., AN713, May 1999.

[8] Wilfried Voss, A comprehensive guide to controller area network, Copperhill Media Corporation, 2005-2008.

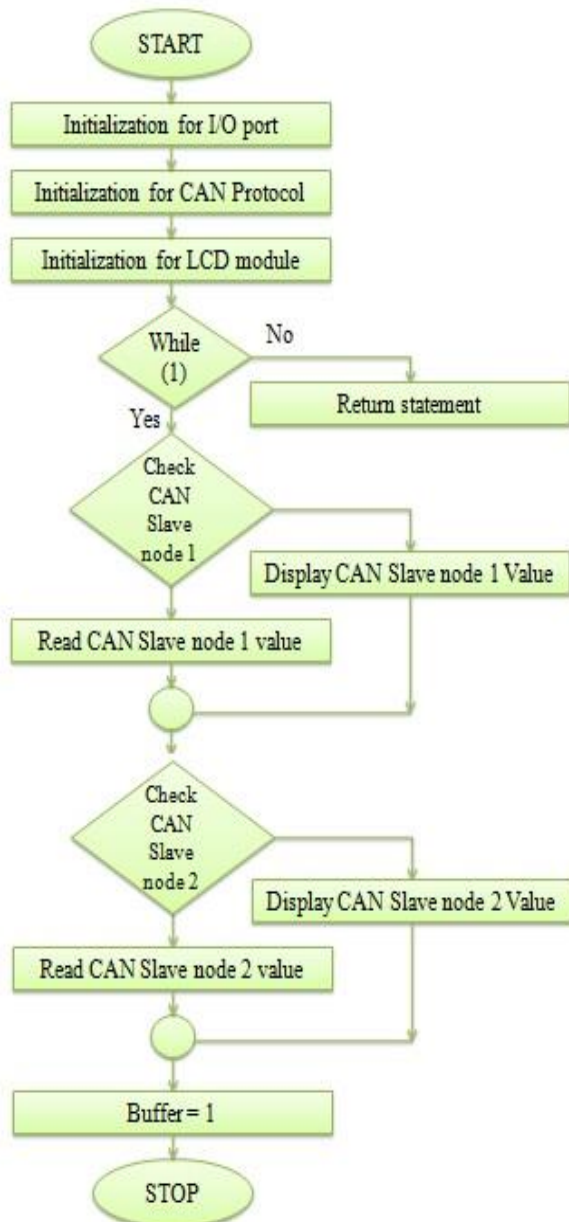[9] Benjamin C Kuo, M. Farid Golnaraghi, Automatic Control systems, Eight edition, John wiley & sons., Inc 2003.