

# Controller Area Network Based Accident Avoidance System

Kashyap M. Joshi, Vipul Gohil

**Abstract**— Based on requirements of modern vehicle, in-vehicle Controller Area Network (CAN) architecture has been implemented. In order to reduce point to point wiring harness in vehicle automation, CAN is suggested as a means for data communication within the vehicle environment. The benefits of CAN bus based network over traditional point to point schemes will offer increased flexibility and expandability for future technology insertions.

This paper describes system which uses sensors to measure various parameters of the car like speed, distance from the other car, presence of alcohol in car and accidental change of lane and sends a warning signal to the driver if any of the parameter goes out of range to avoid accidents. A situation that provides a good example of how the system works is when a driver is about to change lanes, and there is a car in his blind spot. The sensors will detect that car and inform the driver before he starts turning, preventing him from potentially getting into a serious accident.

**Index Terms**— Control Area Network (CAN), collision avoidance system

## I. INTRODUCTION

Nowadays accidents occur due to mistakes done by driver. An intelligent system needs to be developed to overcome these mistakes. So this system is proposed where mistakes done by driver are eliminated. Most of the intelligent car systems have monitoring system only. Antilock brakes, speed sensors and other automatic systems are present in sports cars and other luxury cars only. But these cars are not affordable to everyone. So, a system needs to be developed which can be implemented in every car. A collision avoidance system is a system of sensors that is placed within a car to warn its driver of any dangers that may lie ahead on the road. Some of the dangers that these sensors can pick up on include how close the car is to other cars surrounding it, how much its speed needs to be reduced while going around a curve, and how close the car is to going off the road. The system uses sensors that send and receive signals from things like other cars; obstacles in the road, traffic lights, and even a central database are placed within the car and tell it of any weather or traffic precautions. A situation that provides a good example of how the system works is when a driver is about to change lanes, and there is a car in his blind spot. The sensors will detect that car and inform the driver before he starts turning, preventing him from potentially getting into a serious accident.

Manuscript published on 28 February 2014.

\* Correspondence Author (s)

**Kashyap M. Joshi** is pursuing Mtech in Electronics and Telecommunication from MPSTME, NMIMS, Mumbai. He has completed BE from Pune, India.

**Vipul J. Gohil** is assistant professor in Electronics and Telecommunication in MPSTME, NMIMS, Mumbai. He has completed BE from Mumbai, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

Ultrasonic sensor is adapted to measure the distance with respect to the previous car. For rear-end end collision avoidance subsystem, the currently available ultrasonic sensors for vehicles are adopted for approaching cars with relatively low speed. While the rough reading of distance data cannot be applied directly, an intelligent approach is proposed to process the raw distance readout of sensors to produce appropriate warning signals. We also include the alcoholic sensors in it to monitor the person in the car; if the person appears to be drunk the transmission will be automatically switched off.

## II. OVERVIEW OF CAN PROTOCOL

Controller area network (CAN) provide high reliability and good real-time performance with very low cost. Due to this, CAN is widely used in a wide range of applications, such as in-vehicle communication, automated manufacturing and distributed process control environments. CAN bus is a serial data communication protocol invented by German BOSCH Corporation in 1983. CAN is a network protocol which is designed for the car industry [1]. Since data communication in car often have many sensors transmitting small data packets, CAN supports data frames with sizes only up to 8 bytes as shown in figure I. Meanwhile, the 8 bytes will not take the bus for a long time, so it ensures real-time communication. CAN use a large amount of overhead, which combined with a 15-bit CRC makes CAN very secure and reliable.

CAN protocol use non-destructive bitwise arbitration process to access shared resource. CAN protocol define a logic bit 0 as a dominant bit and a logic bit 1 as a recessive bit, each transmitting node monitors the bus state and compares the received bit with the transmitted bit [2]. If a dominant bit is received when a recessive bit is transmitted then the node stops transmitting (i.e. it lost arbitration).

Arbitration is performed during the transmission of the identifier field. There are two message formats: Base frame format with 11 identifier bits and extended frame format with 29 identifier bits [3].

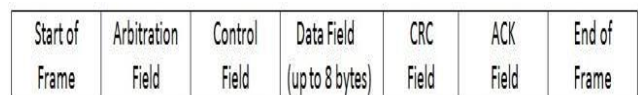


Figure 1. CAN Data Frame

### A. CAN Bus Electrical Characteristics

CAN transmission medium formed by the two, One is called high-level transmission line CANH and another is called low-level transmission line CANL, connected to CANH and CANL pins of MCP2551 CAN transceiver.

$V_{CANH}$  and  $V_{CANL}$  be the voltage level of CANH and CANL lines with respect to ground. The difference between them is called difference voltage  $V_{diff}$ .

**B. Hierarchical structure of CAN BUS**

Architecture of CAN protocol based on OSI reference model is as shown in figure II. CAN protocol contain only three layers, physical layer, data link layer and application layer. Application layer has different protocols such as SAE J1939, CANopen, Device Net, etc [4].

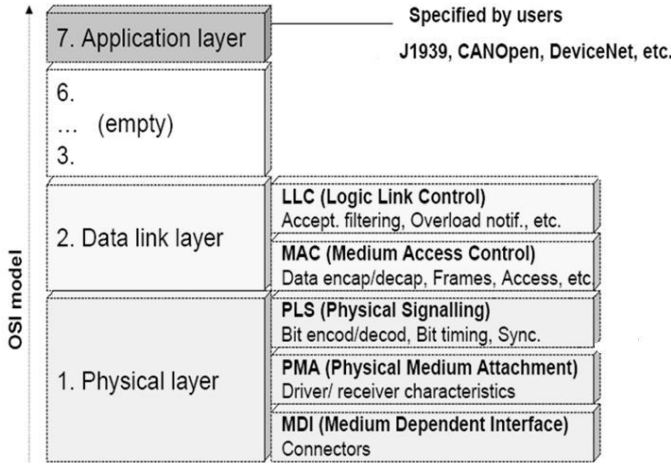


Figure II. Hierarchical structure of CAN BUS

**C. Scheduling of CAN BUS**

CAN protocol implements fixed priority scheduling of CAN messages. Higher priority node has lower node ID. If available bandwidth is scarce, problems come with traditional fixed-priority based scheduling [5]. It is possible that low priority control loops cannot access the network all the time, since limited resources have been consumed by high priority loops. As a result of tremendous delays, low priority control loops may be destabilized.

The problem of fixed priority is overcome using direct feedback scheduling algorithm, namely MUF (maximum urgency first) is integrated in the network scheduler.

Upon invocation, the scheduler calculates the urgency of each control loop based on the set points and current system outputs. According to the MUF algorithm, the scheduler produces new priorities based on these urgency values. And then, messages in different loops will be transmitted in accordance with the newly assigned priorities[6]. A new mixed traffic schedule (MTS) is based on the communication principle of controller area network, network scheduling and analysis of schedule.

The core idea of MTS is set the relative deadline information into the identifier. Use the earliest deadline first (EDF) message scheduling algorithm for high priority information and the RMS (rate monotonic scheduling) message scheduling algorithm for low priority information.

**D. Reliability**

Reliability is defined as the probability of no failures in an operational interval. High error handling capability of CAN improves system reliability. If any message transmitting node has detected an error, the node forcibly aborts transmission. Then it attempts to retransmit repeatedly until its message is

transmitted successfully. This functionality may let the CAN bus be hogged, if the node of high priority is failed. It is the designer's responsibility to ensure that no any message node hogs the bus. To avoid such crisis, the faculty of the transmit error counter (TEC) and the receive error counter (REC) are started to diagnose the conditions of CAN controller [7]. MCP2515 CAN controller has TEC and REC which enhances reliability of CAN bus system.

A CAN controller can be in one of three states: error active, error passive or bus off state. The operating state of the controller is controlled by two error counters – TEC and REC. The CAN controller is in error active state if TEC less than 127 and REC less than 127 [8]. Passive state is used if (TEC greater than 127 or REC greater than 127) and TEC less than 255. Bus off state is entered if TEC is greater than 255. Once the CAN controller has entered bus off state, it must be reset by the host microcontroller in order to be able to continue operation.

**III. HARDWARE DESIGN**

The proposed block diagram for CAN bus communication system is as shown in figure III. In this system the ultrasonic sensor is mounted on the front and backside of the car for measuring the distance between the two cars and if the distance is less then to avoid accident warning signal will be given to the driver on the LCD. The alcoholic sensor will sense whether the driver is drunk and if the driver is drunk then the driver will not be allowed to start the car . If the car accidentally changes its lane then it will be detected using an IR sensor and buzzer will be turned on.

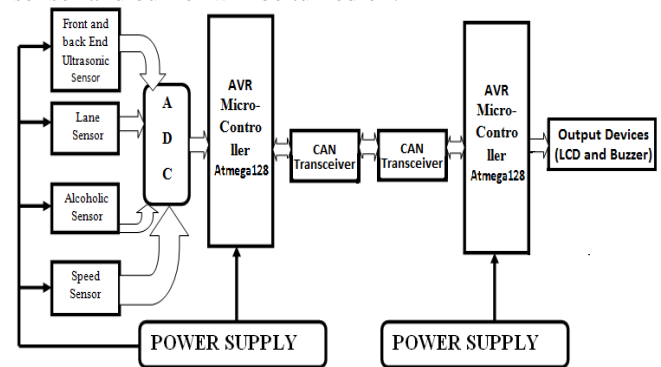


Figure III Block Diagram

The speed sensor will monitor the speed of the car and if found high then warning will be given to the driver using an alarm.

Here the sensors will communicate with the output devices using CAN (Control Area Network) protocol which will be implemented in the AVR controller

**A. Ultrasonic sensor**

Ultrasonic sensor is adapted to measure the distance with respect to the previous car. While the car is in motion the distance of another car is measured and accordingly warning signals are given to the driver.

**B. Alcoholic sensor**

Alcoholic sensors in it to monitor the person in the car. If the person appears to be drunk the transmission will be automatically switched off.



### C. Speed sensor

This sensor monitors the speed of the car and if the speed is found to be more than a prescribed level then a warning signal will be given to the driver.

### D. Lane sensor

Lane sensor will detect whether the car is in the same lane it is travelling and if accidentally the car changes its lane then accordingly a warning will be given to the driver.

### E. Microcontroller

The controller will take input from the sensors and depending on the various sensor inputs output devices will be driven using the other microcontroller. The two microcontroller will communicate with one another using can communication protocol.

### F. CAN controller

The CAN controller stores received bits serially from the bus until an entire message is available, which can then be fetched by the host processor. The host processor stores it and transmit messages to a CAN controller, which transmits the bits serially onto the bus.

### G. Transceiver

It adapts signal level from the bus to level that the CAN controller expects and has protective circuitry that protects the CAN controller. It converts the transmit-bit signal received from the CAN controller into a signal that is sent onto the bus.

### H. Output Devices

According to the inputs from various sensors warning signals are given to the driver to lower the speed of the car or stop turning to get into wrong lane.

MCP2515 implements the CAN specification, version 2.0B It is capable of transmitting and receiving both standard and extended data and remote frames with 0 – 8 byte length of the data field. The MCP2515 has two acceptance masks and six acceptance filters that are used to filter out unwanted messages, thereby reducing the host MCUs overhead. The MCP2515 is interfaced with microcontroller (MCU) via an industry standard Serial Peripheral Interface (SPI) [9]. MCP2515 has two receive buffers with prioritized message storage and three transmit buffers with prioritization and abort features.

The MCP2551 is a high-speed CAN, fault-tolerant device that serves as the interface between a CAN protocol controller and the physical bus. The MCP2551 provides differential transmit and receive capability for the CAN protocol controller. It will operate at speeds of up to 1 Mbs<sup>-1</sup> at 40m length.

Typically, each node in a CAN system must have a device to convert the digital signals generated by a CAN controller to signals suitable for transmission over the bus cabling (differential output). It allows a maximum of 112 nodes to be connected and a nominal termination resistor value of 120Ω. A dominant state occurs when the differential voltage between CANH and CANL is greater than a defined voltage (e.g., 1.2V). A recessive state occurs when the differential voltage is less than a defined voltage (typically 0V). The RXD output pin reflects the differential bus voltage between CANH and CANL.

## IV. PROPOSED ALGORITHM

Algorithm for the proposed system is divided in two parts as follows: Transmitter and Receiver

### A. Transmitter

Algorithm for transmitter side which consists sensors, AVR microcontroller and CAN (MCP2515) is as follows:

1. Initialize SPI (Serial Peripheral Interface).
2. Initialize LCD.
3. Initialize CAN (MCP2515).
4. Provide impulse to ultrasonic sensor.
5. Measure distance from other car and display on LCD.
6. Transmit distance via CAN (MCP2515).
7. If alcohol sensed send X.
8. If lane cutting detected send Y.
9. Sense speed and if speed goes beyond range send Z.

### B. Receiver

Algorithm for transmitter side which consists output devices, AVR microcontroller and CAN (MCP2515) is as follows:

1. Initialize SPI (Serial Peripheral Interface).
2. Initialize LCD.
3. Initialize CAN (MCP2515).
4. Send acknowledgment to the transmitter.
5. Receive distance data from CAN of transmitter and if distance is less then display warning signal on LCD.
6. If X is received then display “Car cannot be started”.
7. If Y is received then display “Wrong lane”.
8. If Z is received then turn on buzzer.

## IV. DESIGN SCHEME OF COMMUNICATION PROTOCOL

The design scheme of communication protocol is explained in this section. Identifier of the message is the unique character for the application program to distinguish messages. In this communication system, when a node receives a message correctly (until the last bit of the EOF area is right), the configured filter box message, and then save the messages with matched ID in receiving box. By using this feature, communication protocol can be made. Different identifiers are set for every data type or control command in this system, then distinguish the received messages conveniently, and choose corresponding processing mode[10].

The standard format of identifier is used in this system. It has 11 bits. Use of standard identifier can reduce the data length and improve data transmission efficiency. In this system, the 11 bit identifier is designed for the “address code + type code” format. Bits D7 to D4 of identifier is the address field, providing at most 16 address codes, and every address code corresponds to a individual node. Bits D3 to D0 is the type field, which can also provide 16 type codes. And the bits D10 to D8 is the backup filed which is used for system expansion. By configuring the value of the filter ID, each node would only receive the messages with the matched address code.



D10 D9 D8	D7	D6	D5	D4	D3	D2	D1	D0
000	0	0	0	1	0	0	1	0
backup	address code			type code				

Figure IV. Identifier format

V. CAN BUS NETWORK SOFTWARE DESIGN

The software part of main program unit is divided into: System initialization unit, CAN initialization unit, message sending unit, message receiving unit and interrupt service unit[10].

System initialization unit includes initialization of microcontroller i.e. Atmega128. Flow chart for CAN node is shown in figure V.

A. CAN Controller Initialization

CAN controller initialization process include initialization of internal register of CAN controller MCP2515, such as initialization of transmit and receive buffer. When initializing the CAN registers in the MCP2515, the system immediately clear the read and write buffer, Configures the operating mode, bit timing register, filter and mask register, and interrupt enable register.

B. CAN Message Sending

After initialization, the MCP2515 is in normal communication status and is ready to work. The CAN message sending adopts inquiry mode or it sends message when remote request is received. In inquiry mode, after specific time interval, Atmega128 creates packet and sends data to CAN Controller transmit buffer for further transmission [10].

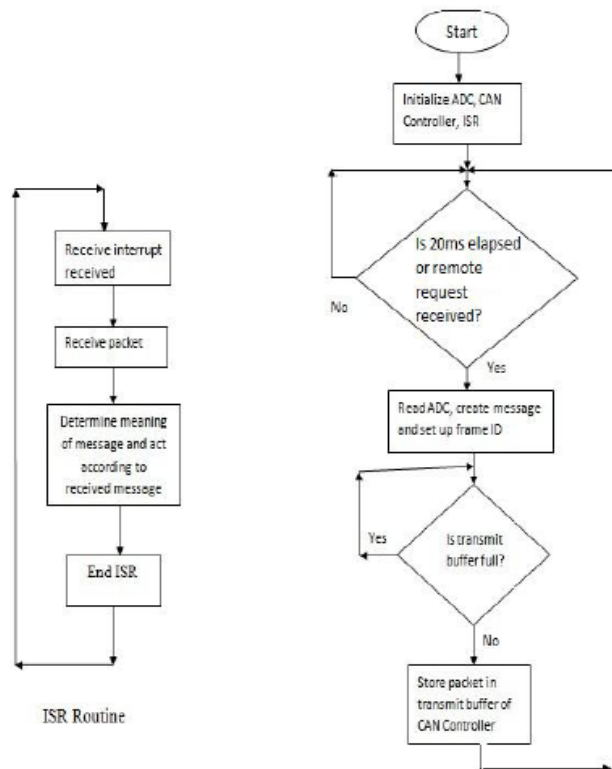


Figure V. CAN bus node design process

C. CAN Message Receiving

There are two ways for messages receiving: inquiry mode and interrupt mode. In system design, interrupt mode is implemented. In interrupt mode, if MCP2515 CAN controller receives valid message in receive buffer, it generates interrupt signal on INT pin. Message is read by Atmega128 by using SPI commands of MCP2515[10].

VI. CONCLUSION

In this paper, the CAN-bus based communication system for accident avoidance system is designed. System can be upgraded easily and use of CAN reduces wiring to a great extent. Real-time, reliability and flexibility, all these characteristics make CAN BUS an indispensable network communication technology applied in automobile network communication field.

VII. ACKNOWLEDGEMENT

I Kashyap M. Joshi would like to thank everyone, including: parents, teachers, family, friends, and in essence, all sentient beings for their help and support this paper would not have been possible. Especially, I dedicate my acknowledgment of gratitude toward my mentor and Co-author Prof. Vipul J. Gohil for his guidance and support.

REFERENCES

- [1] Li Ran, Wu Junfeng, Wang Haiying, Li Gechen. "Design Method of CAN BUS Network Communication Structure for Electric Vehicle", IFOST 2010 Proceedings IEEE.
- [2] Yujia Wang, Hao Su, Mingjun Zhang., "CAN-Bus-Based Communication System Research for Modular underwater Vehicle", 2011 IEEE DOI 10.1109/ICICTA.2011.
- [3] Chin E. Lin, S. F. Tai, H. T. Lin, T. P. Chen, P. K. Chang, C. C. Kao "Prototype Of A Small Aircraft Avionics Using Hybrid Data Bus Technology" 2005 IEEE
- [4] Chin E. Lin, Hung-Ming Yen, "A Prototype Dual Can-Bus Avionics System For Small Aircraft Transportation System" 2006 IEEE.
- [5] V. Claesson, C. Ekelin, N. Suri, "The event-triggered and time-triggered medium-access methods", 6th IEEE International Symposium on Object-Oriented Real-Time Distributed computing, May 14-16, 2003, pp. 131-134.
- [6] Chris Quigley, Richard McLaughlin, "Electronic System Inte-Gratation For Hybrid And Electric Vehicles".
- [7] Chin E. Lin, H. M. Yen, "Reliability And Stability Survey On Can-Based Avionics Network For Small Aircraft" -7803-9307-4/05/2005 IEEE.
- [8] Feng Xia, Xiaohua Dai, Zhi Wang, and Youxian Sun, "Feedback Based Network Scheduling of Networked Control Systems" 2005 IEEE.
- [9] Ashwini S. Shinde, Prof. Vidhyadhar B. Dharmadhikari, "Controller Area Network for Vehicle Automation" International Journal of Emerging Technology and Advanced Engineering www.ijetae.com ISSN 2250-2459, Volume 2, Issue 2, February 2012.
- [10] Mazran Esro, Amat Amir Basari, Siva Kumar S, A. Sadhiqin M I, Zulkifli Syariff, "Controller Area Network (CAN) Application in Security System" World Academy of Science, Engineering and Technology 35 2009.



**Kashyap M. Joshi** is pursuing Mtech in Electronics and Telecommunication from MPSTME, NMIMS, Mumbai. He has completed BE from Pune University in 2010. He has published a paper in national conference NCETETA 2013. He is currently working as teaching assistant in the department of Electronics and Telecommunication Engineering at MPSTME, NMIMS, Mumbai.





**Vipul J. Gohil** is assistant professor in Electronics and Telecommunication in MPSTME, NMIMS, Mumbai. He has completed BE from Mumbai University in 2001 and M.Tech from NMIMS in 2013. He has a teaching experience of around 11 years. He has published a paper in International journal IJLRST-2013. He is currently working as assistant professor in the department of Electronics and Telecommunication Engineering at MPSTME, NMIMS, Mumbai from past 5 years.