ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

CAN Based Accident Avoidance System for Automobiles

M. Suresh
PG Scholar,
Department of ECE,
Siddartha Educational Academy Groups of Institutions,
Tirupati, AP, India.

T.Venkata Ramana
Assistant Professor,
Department of ECE,
Siddartha Educational Academy Groups of Institutions,
Tirupati, AP, India.

ABSTRACT

Safety is traditionally the most relevant property for automotive systems, and it is further enhanced by Advanced Driver Assistance Systems (ADAS) in modern automotive systems. To support ADAS and other advanced autonomous functions, automotive electronic systems become more distributed and connected than ever, no matter from the perspective of in-vehicle architecture or Vehicle-to-X (V2X) communication. These connections create a variety of interfaces, such as direct or indirect physical access, short-range wire-less access, and long-range wireless channels, which become breeding grounds for security attacks. Accordingly, security becomes a rising issue for automotive systems.

The Controller Area Network (CAN) protocol has been the main focus of automotive security studies, and it has no direct support for security projection.

Hoppe et al. demonstrated that the operations of electric window lifts, warning lights, and airbag control systems may be affected through the CAN protocol.

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.

INTRODUCTION

Cars on the same direction in highway usually keep a safe distance one another with a similar speed. However, due to the driver's distraction, long-time driving fatigue, flake out, or even a sudden deceleration of the previous car, a serious collision accident may occur if the driver can not react in time to brake. On the other hand, drivers need the mirrors to know other approaching cars from two-side or from the rear end. Even the driver check around carefully, he cannot take an immediate respond, except push the horn, to a sudden approaching car and an accident is thus unavoidable. Therefore, developing a front-obstacle warning system and a rear end collision avoidance system subject to all directions are important in collision avoidance. For the front-end collision avoidance subsystem, Ultrasonic sensor is adopted 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.

When there are more electrical control devices in the modem cars, such as power train management system, antilock braking system (ABS), and acceleration skid control (ASC) system, etc, the functionality and wiring of these electric control units (!XU) are getting more complicated. Therefore, it is of great concern to upgrade the traditional wire harness to a smart & car network. In 198Os,a Germany car component provider

Robert Bosch Co. introduced an in-car network;the controller area network (CAN) bus, to replace the complex and expensive traditional in-car wiring [5]. In thisstudy, a high-level protocol CAN open is adopted to interconnect those CAN nodes with reliable communications among sensors.

STATEMENT OF THE PROBLEM

In this project we aim at Designing of CAN based Accident avoidance system using two Philips LPC2129 32 bit microcontroller which is having ARM7TDMI processor with many onboard interfaces like memory, LCD, I/O , CAN controller, serial port ,I2C interface, UART, 10 bit ADC, and standard JTAG interface. These two microcontroller are connected by CAN bus for transmission of data between them.

PRAPOSED SYSTEM

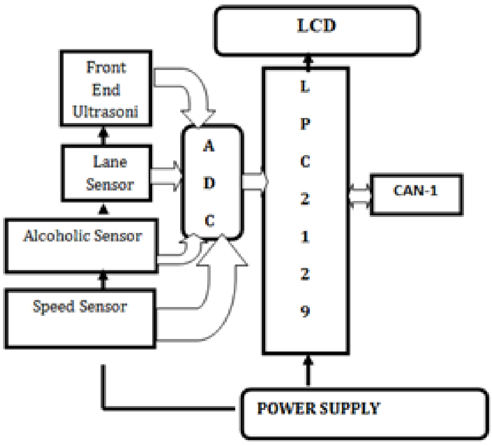


Fig.1: Block diagram

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.

Ultrasonic sensor is adapted to measure the distance with respect to the previous car. Forrear-end end collision avoidance subsystem, the currently available ultrasonic sensors for vehicles are adopted for approaching cars with relatively low speed. While the roughreading 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.

CONTROLLER AREA NETWORK PROTOCOL (CAN):

Controller Area Network (CAN) is an advanced serial bus system that efficiently supports distributed control system with a very high level of security.

Robert Bosch, Germany[1][2][3] initially developed it for the use in motor vehicles in the late 1980’s.It’s domain of application ranges from high-speed network to low cost multiplex wiring.

To improve the behavior of the vehicle, it was necessary for the different control system (and their sensor) to exchange information. This was usually done by discrete interconnection of the different system (i.e. point -to - point wiring). The requirement for the information exchange has then grown to such an extent that a cable network with a length up to several miles and many connectors were required. This leads to growing problems concerning material cost, production time and reliability.

The solution to this problem was the connection of the Control system via a serial bus system. With the use of CAN, point - to - point wiring is replaced by one serial bus connecting to all control systems. This is accomplished by adding some CAN specific hardware to each control unit that provides the “rules " or the protocol for transmitting and receiving information via the bus.

The CAN protocol uses the data link layer and the physical layer in the ISO_OSI model.
CAN is a multi - master bus with an open, linear structure with one logic bus line. The number of nodes is not limited by the protocol.

In CAN protocol, two versions are available. They are version 2.0A CAN and version 2.0B CAN. Version 2.0A is original CAN specifications specify an 11 bit identifier which allows 2^11(=2048) different message identifiers and is known as standard CAN. Version 2.0B CAN contain 29 bit identifiers which allows 2^29 (over 536 million) message identifiers.

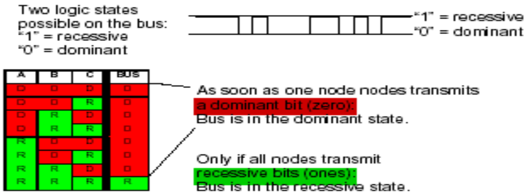
CAN has the following properties:

- Prioritization of messages.
- Guarantee of latency times.
- Configuration flexibility.
- Multicast reception with time synchronization.
- System wide data consistency.
- Error detection and error signaling.

The CAN protocol handle bus accesses according to the concept called "Carrier Sense Multiple Access with arbitration on message priority ". This arbitration Concept avoids collisions of messages whose transmission was started by more than one node simultaneously and makes sure the most important message is sent first without time loss.

If two or more bus nodes start their transmission at the same time after having found the bus to be idle, collision of the messages is avoided by bitwise arbitration. Each node sends the bits of its message identifier and monitors the bus level.

BUS CHARACTERISTICS



A	B	C	BUS
D	D	D	D
D	D	R	D
D	R	D	D
D	R	R	D
R	D	D	D
R	D	R	D
R	R	D	D
R	R	R	R

Fig.2: Bus characteristics

There are two bus states, called "dominant" and "recessive". The bus logic uses a "Wired-AND" mechanism, that is, "dominant bits" (equivalent to the logic level "Zero") overwrite the "recessive" bits (equivalent to the logic level "One").

BUS ACCESS AND ARBITRATION

The CAN protocol handles bus accesses according to the concept called "Carrier Sense Multiple Access with Arbitration on Message Priority". This arbitration concept avoids collisions of messages whose transmission was started by more than one node simultaneously and makes sure the most important message is sent first without time loss.

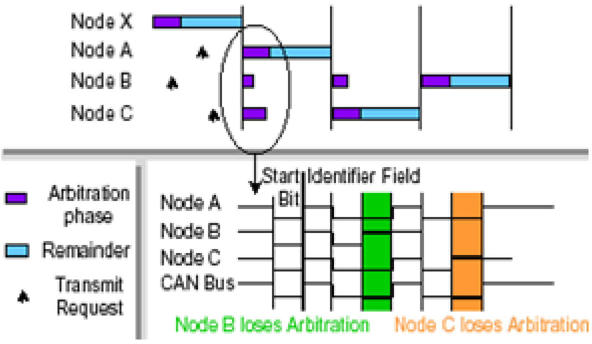


Fig.3: Bus Access And Arbitration

In the picture above you see the trace of the transmit pins of three bus nodes called A, B and C, and the resulting bus state according to the wired-AND principle.

If two or more bus nodes start their transmission at the same time after having found the bus to be idle, collision of the messages is avoided by bitwise arbitration. Each node sends the bits of its message identifier and monitors the bus level.

At a certain time nodes A and C send a dominant identifier bit. Node B sends a recessive identifier bit but reads back a dominant one. Node B loses bus arbitration and switches to receive mode. Some bits later node C loses arbitration against node A. This means that the message identifier of node A has a lower binary value and therefore a higher priority than the messages of nodes B and C. In this way, the bus node with the highest priority message wins arbitration without losing time by having to repeat the message.

Nodes B and C automatically try to repeat their transmission once the bus returns to the idle state. Node B loses against node C, so the message of node C is transmitted next, followed by node B's message.

It is not permitted for different nodes to send messages with the same identifier as arbitration could fail leading to collisions and errors

REMOTE FRAME

As shown in fig 4 is transmitted by a bus unit to request the transmission of the data frame with the same identifier. It is composed of six different bit fields. Start of frame, Arbitration field, control field, crc field, Ack field, End of frame.

A "Data Frame" is generated by a CAN node when the node wishes to transmit data. The Standard CAN Data Frame is shown above. The frame begins with a dominant Start of Frame bit for hard synchronization of all nodes.

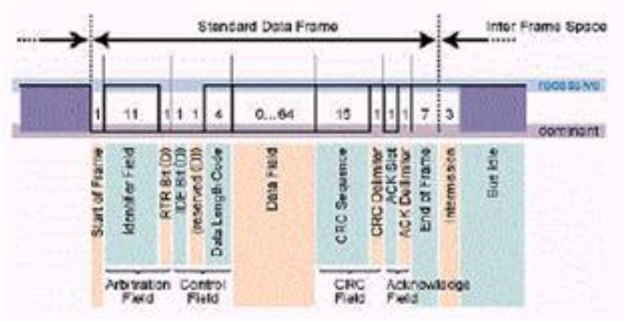


Fig.4: Data Frame

The Start of Frame bit is followed by the Arbitration Field consisting of 12 bits. The 11-bit Identifier, which reflects the contents and priority of the message, and the Remote Transmission Request bit. The Remote transmission request bit is used to distinguish a Data Frame (RTR = dominant) from a Remote Frame (RTR = recessive).The next field is the Control Field, consisting of 6 bits. The first bit of this field is called the IDE bit (Identifier Extension) and is at dominant state to specify that the frame is a Standard Frame. The following bit is reserved and defined as a dominant bit. The remaining 4 bits of the Control Field are the Data Length Code (DLC) and specify the number of bytes of data contained in the message (0 - 8 bytes).

The data being sent follows in the Data Field which is of the length defined by the DLC above (0, 8, 16, 56 or 64 bits).The Cyclic Redundancy Field (CRC field) follows and is used to detect possible transmission errors. The CRC Field consists of a 15 bit CRC sequence, completed by the recessive CRC Delimiter bit.

The next field is the Acknowledge Field. During the ACK Slot bit the transmitting node sends out a recessive bit. Any node that has received an error free frame acknowledges the correct reception of the frame by sending back a dominant bit (regardless of whether the node is configured to accept that specific message or not). From this it can be seen that CAN belongs to the "in-bit-response" group of protocols. The recessive Acknowledge Delimiter completes the Acknowledge Slot and may not be overwritten by a dominant bit. Seven recessive bits (End of Frame) end the Data Frame.

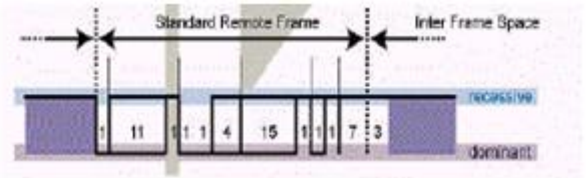


Fig.5: Remote Frame

Generally data transmission is performed on an autonomous basis with the data source node (e.g. a sensor) sending out a Data Frame. It is also possible, however, for a destination node to request the data from the source by sending a Remote Frame.There are 2 differences between a Data Frame and a Remote Frame. Firstly the RTR-bit is transmitted as a dominant bit in the Data Frame and secondly in the Remote Frame there is no Data Field. In the very unlikely event of a Data Frame and a Remote Frame with the same identifier being transmitted at the same time.

The Data Frame wins arbitration due to the dominant RTR bit following the identifier. In this way, the node that transmitted the Remote Frame receives the desired data immediately.

ERROR HANDLING

The CAN Controllers count and handle transmit and receive errors as specified in CAN Spec 2.0B. The Transmit and Receive Error Counters are incremented for each detected error and are decremented when operation is error-free. If the Transmit Error counter contains 255 and another error occurs, the CAN Controller is forced into a state called Bus-Off. In this state, the following register bits are set: BS in CANSR, BEI and EI in CANIR if these are enabled, and RM in CANMOD. RM resets and disables much of the CAN Controller.

Also at this time the Transmit Error Counter is set to 127 and the Receive Error Counter is cleared. Software must next clear the RM bit. Thereafter the Transmit Error Counter will count down 128 occurrences of the Bus Free condition (11 consecutive recessive bits). Software can monitor this countdown by reading the Tx Error Counter. When this countdown is complete, the CAN Controller clears BS and ES in CANSR, and sets EI in CANSR if EIE in IER is 1.

The Tx and Rx error counters can be written if RM in CANMOD is 1. Writing 255 to the Tx Error Counter forces the CAN Controller to Bus-Off state. If Bus-Off (BS in CANSR) is 1, writing any value 0 through 254 to the Tx Error Counter clears Bus-Off. When software clears RM in CANMOD thereafter, only one Bus Free condition (11 consecutive recessive bits) is needed before operation resumes.

LPC 2129 MICROCONTROLLER

The LPC2119/2129/2194/2292/2294 are based on a 16/32 bit ARM7TDMI-STM CPU with real-time emulation and embedded trace support, together with 128/256 kilobytes (kB) of embedded high speed flash memory. A 128-bit wide internal memory interface and a unique accelerator architecture enable 32-bit code execution at maximum clock rate.

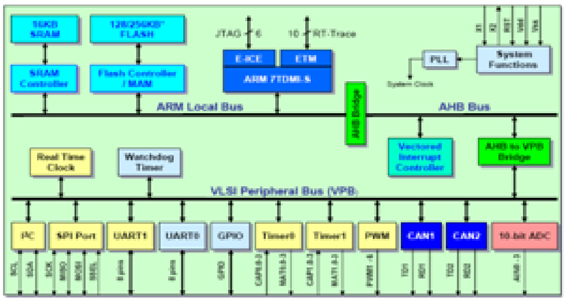


Fig.6: LPC2129 internal Block diagram

For critical code size applications, the alternative 16-bit Thumb Mode reduces code by more than 30% with minimal performance penalty. With their compact 64 and 144 pin packages, low power consumption, various 32-bit timers, combination of 4-channel 10-bit ADC and 2/4 advanced CAN channels or 8-channel 10-bit ADC and 2/4 advanced CAN channels (64 and 144 pin packages respectively), and up to 9 external interrupt pins these microcontrollers are particularly suitable for industrial control, medical systems, access control and point-of-sale. Number of available GPIOs goes up to 46 in 64 pin package. In 144 pin packages number of available GPIOs tops 76 (with external memory in use) through 112 (single-chip application). Being equipped wide range of serial communications interfaces, they are also very well suited for communication gateways, protocol converters and embedded soft modems as well as many other general-purpose applications.Number of Components Used: Two KNOWX ARM BOARDS.

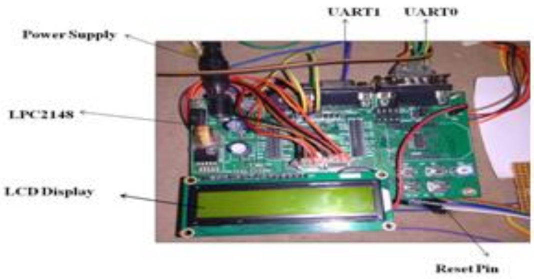


Fig.7: LPC2129 BLOCK DIAGRAM

Features of LPC2129

- 16/32-bit ARM7TDMI-S microcontroller in a 64 or 144 pin package.
- 16 kB on-chip Static RAM
- 128/256 kB on-chip Flash Program Memory. 128-bit wide interface/accelerator enables high speed 60 MHz operation.
- External 8, 16 or 32-bit bus (144 pin package only)
- In-System Programming (ISP) and In-Application Programming (IAP) via on-chip boot-loader software. Flash programming takes 1 ms per 512 byte line. Single sector or full chip erase takes 400 ms.

ULTRASONIC SENSOR

The sensor is primarily intended to be used in security systems for detection of moving objects, but can be effectively involved in intelligent children’s toys, automatic door opening devices, and sports training and contact-less-speed measurement equipment.

Modern security systems utilize various types of sensors to detect unauthorized object access attempts. The sensor collection includes infrared, microwave and ultrasound devices, which are intended to detect moving objects. Each type of sensor is characterized by its own advantages and drawbacks.

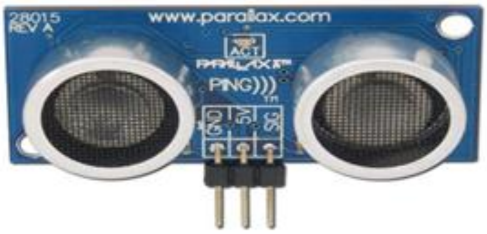


Fig.8: ultrasonic sensor

Microwave sensors are effective in large apartments because microwaves pass through dielectric materials. But these sensors consist of expensive super-high frequency components and their radiation is unhealthy for living organisms. Infrared sensors are characterized by high sensitivity, low cost and are widely used. But, these sensors can generate false alarm signals if heating systems are active or temperature change speed exceeds some threshold level. The ultrasound transmitter **TX** is emitting ultrasound waves into sensor ambient space continuously. These waves are reflecting from various objects and are reaching ultrasound receiver **RX**.

BUZZER

The piezo buzzer produces sound based on reverse of the piezoelectric effect. The generation of pressure variation or strain by the application of electric potential across a piezoelectric material is the underlying principle. These buzzers can be used alert a user of an event corresponding to a switching action, counter signal or sensor input. They are also used in alarm circuits. The buzzer produces a same noisy sound irrespective of the voltage variation applied to it.

Fig.9: Buzzer

ALCOHOL SENSOR (MQ2):

The Grove - Gas Sensor(MQ2) module is useful for gas leakage detection (in home and industry). It is suitable for detecting H2, LPG, CH4, CO, Alcohol, Smoke or Propane. Due to its high sensitivity and fast response time, measurements can be taken as soon as possible. The sensitivity of the sensor can be adjusted by using the potentiometer.

The sensor value only reflects the approximated trend of gas concentration in a permissible error range, it DOES NOT represent the exact gas concentration. The detection of certain components in the air usually requires a more precise and costly instrument, which cannot be done with a single gas sensor. If your project is aimed at obtaining the gas concentration at a very precise level, then we do not recommend this gas sensor.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

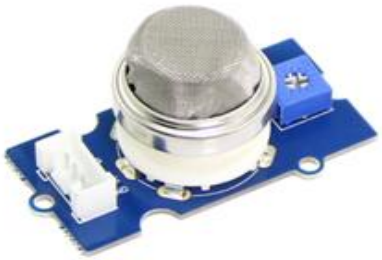
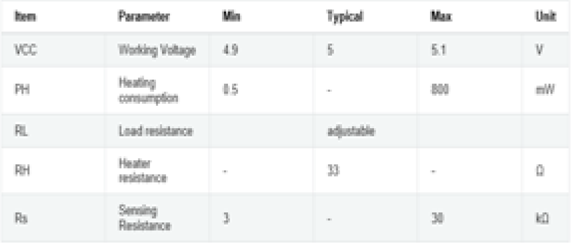


Fig.10: MQ-2 SENSOR

TABLE.I:MQ2 PIN CONFIGURATION



Item	Parameter	Min	Typical	Max	Unit
VCC	Working Voltage	4.9	5	5.1	V
PH	Heating consumption	0.5	-	800	mW
RL	Load resistance		adjustable		
RH	Heater resistance	-	33	-	Ω
Rs	Sensing Resistance	3	-	30	kΩ

IR- SENSOR

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion.These types of sensors measures only infrared radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infrared sensor.The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, Theresistances and these output voltages, change in proportion to the magnitude of the IR light received.

Fig.11: IR- SENSOR

An infrared sensor circuit is one of the basic and popular sensor module in an electronic device. This sensor is analogous to human's visionary senses, which can be used to detect obstacles and it is one of the common applications in real time.This circuit comprises of the following components

- LM358 IC 2 IR transmitter and receiver pair
- Resistors of the range of kilo ohms.
- Variable resistors.
- LED (Light Emitting Diode).

In this project, the transmitter section includes an IR sensor, which transmits continuous IR rays to be received by an IR receiver module. An IR output terminal of the receiver varies depending upon its receiving of IR rays. Since this variation cannot be analyzed as such, therefore this output can be fed to a comparator circuit. Here an operational amplifier (op-amp) of LM 339 is used as comparator circuit.

When the IR receiver does not receive a signal, the potential at the inverting input goes higher than that non-inverting input of the comparator IC (LM339). Thus the output of the comparator goes low, but the LED does not glow. When the IR receiver module receives signal to the potential at the inverting input goes low. Thus the output of the comparator (LM 339) goes high and the LED starts glowing. Resistor R1 (100), R2 (10k) and R3 (330) are used to ensure that minimum 10 mA current passes through the IR LED Devices like Photodiode and normal LEDs respectively. Resistor VR2 (preset=5k) is used to adjust the output terminals. Resistor VR1 (preset=10k) is used to set the sensitivity of the circuit Diagram. Read more about IR sensors.

IR sensors are classified into different types depending on the applications. Some of the typical applications of different types of sensors areThe speed sensor is used for synchronizing the speed of multiple motors. The temperature sensor is used for industrial temperature control. PIR sensor is used for automatic door opening system and Ultrasonic sensor are used for distance measurement.

IR SENSOR APPLICATIONS

IR sensors are used in various Sensor based projects and also in various electronic devices which measures the temperature that are discussed in the below.

Radiation Thermometers

IR sensors are used in radiation thermometers to measure the temperature depend upon the temperature and the material of the object and these thermometers have some of the following features

- Measurement without direct contact with the object
- Faster response
- Easy pattern measurements

DC MOTOR

DC motors are configured in many types and sizes, including brush less, servo, and gear motor types. A motor consists of a rotor and a permanent magnetic field stator. The magnetic field is maintained using either permanent magnets or electromagnetic windings. DC motors are most commonly used in variable speed and torque.

Motion and controls cover a wide range of components that in some way are used to generate and/or control motion. Areas within this category include bearings and bushings, clutches and brakes, controls and drives, drive components, encoders and resolves, Integrated motion control, limit switches, linear actuators, linear and rotary motion components, linear position sensing, motors (both AC and DC motors), orientation position sensing, pneumatics and pneumatic components, positioning stages, slides and guides, power transmission (mechanical), seals, slip rings, solenoids, springs.

Motors are the devices that provide the actual speed and torque in a drive system. This family includes AC motor types (single and multiphase motors, universal, servo motors, induction, synchronous, and gear motor) and DC motors (brush less, servo motor, and gear motor) as well as linear, stepper and air motors, and motor contactors and starters.

SOFT WARE DESIGN TOOLS

KEIL SOFTWARE

Keil compiler is software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. Keil compiler also supports C language code.

PROLOAD

Proload is software which accepts only hex files. Once the machine code is converted into hex code, that hex code has to be dumped into the microcontroller placed in the programmer kit and this is done by the Proload. Programmer kit contains a microcontroller on it other than the one which is to be programmed. This microcontroller has a program in it written in such a way that it accepts the hex file from the keil compiler and dumps this hex file into the microcontroller which is to be programmed. As this programmer kit requires power supply to be operated, this power supply is given from the power supply circuit designed above. It should be noted that this programmer kit contains a power supply section in the board itself but in order to switch on that power supply, a source is required. Thus this is accomplished from the power supply board with an output of 12volts or from an adapter connected to 230 V AC.

EXPERIMENTAL RESULTS:

CONCLUSION

This project CAN BASED ACCIDENT AVOIDANCE SYSTEM is intended for secure and smooth journey. The car/ vehicle itself is aware of its movement. If the driver himself is not concentrating on driving or any other parameters, which may cause damage to vehicle as well a life, this intelligent car/ vehicle warn the driver regarding the danger ahead. As the value of a human life is countless times more than the cost of this project, we are proud to be behind the success of this project.

FUTURE ENHANCEMENT

As for the rear end, side-end end collision avoidance subsystem can be adopted wSSSith the use of the currently available ultrasonic sensors for vehicles. Further, the relative speed between two cars can be estimated by applying a one-dimension **Kalman** filter [5]with great efficiency. From experimental data, a D/V curve can be further obtained to reliably generate a warning signal in advance of the accidental collision. As the car in traffic with pretty low speed or in a waiting state at the intersection, the warning signals should be terminated after a certain time since no collision warning is required under such circumstances.

REFERENCES

1. R. Anderson, F. Bergadano, B. Crispo, J.-H. Lee, C. Manifavas, and R. Needham, “A new family of authentication protocols,” ACM SIGOPS Operating Systems Review, vol. 32, no. 4, pp. 9–20, Oct. 1998.

2. P. Axer, D. Thiele, R. Ernst, J. Diemer, “Exploiting shaper context to improve performance bounds of Ethernet AVB networks,” ACM/IEEE Design Automation Conference, pp. 1–6, San Francisco, CA, USA, Jun. 2014.

3. R. Bloomfield, K. Netkachova, and R. Stroud, “Security-informed safety: if it’s not secure, it’s not safe,”International Workshop on Software Engineering for Resilient Systems, Kiev, Ukraine, Oct. 2013.

4. S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, “Comprehensive experimental analyses of automotive attack surfaces,” USENIX Conference on Security, pp. 6–6, San Francisco, CA, USA, Aug. 2011.

5. J. Diemer, D. Thiele, and R. Ernst, “Formal worst-case timing analysis of Ethernet topologies with strict-priority and AVB switching,” IEEE International Symposium on Industrial Embedded Systems, pp. 1–10, Karlsruhe, Germany, Jun. 2012.

6. I. Gashi, A. A. Povyakalo, L. Strigini, M. Matschnig, T. Hinterstoisser, and B. Fischer, “Diversity for safety and security in embedded systems,” IEEE International Conference on Dependable Systems and Networks, Atlanta, GA, USA, Jun. 2014.

7. B. Groza, S. Murvay, A. Van Herrewege, and I. Verbauwhede, “LiBrA-CAN: a lightweight broadcast authentication protocol for Controller Area Networks,” International Conference on Cryptology and Network Security, pp. 185–200, Darmstadt, Germany, Dec. 2012.

8. T. Hoppe, S. Kiltz, and J. Dittmann. “Security threats to automotive CAN networks—practical examples and selected short-term countermeasures,” InternationalConference on Computer Safety, Reliability and Security, pages 235–248, Newcastle upon Tyne, United Kingdom, Sep. 2008.