# AUTOMATIC VEHICLE CONTROL AND SAFETY USING ARDUINO

[1] Kishan Dash, [2]Debasish Mohapatra, [3]Manas Ranjan Das, [4]AsishTripathy
[1]Assistant Professor, [2]Assistant Professor, [3]Assistant Professor, [4]Assistant Professor
[1]Department Of Mechanical Engineering,
[1]Raajdhani Engineering College, Bhubaneswar, India

***Abstract:*** An automatic vehicle human-voice-based safety control is proposed to decrease the happening of vehicle accidents due to an eye or hand activities while driving. The design is implemented using an Arduino microcontroller. One of the most important safety control features is the automotive speed control (ASC). In this safety control function, which prevents a vehicle sudden hit, the necessary action is issued based on the vehicle-to- vehicle distance measure. To prevent the driver from attention withdraw and maintain a proper road and situation focus (as especially is the case at night), several other automated safety control operations are provided (i.e., radio, flasher, left flasher, right flasher, wipers, drive, and stop). These features are controlled according to human voice commands using an EasyVR module. The proposed model has been designed, built, and tested providing suitable results as desired.

***Index Terms*** - Safety-vehicle-control, Arduino microcontroller, voice recognition.

## Introduction

The problem of vehicle accident is part of an never-ending list of disasters that could occur anywhere anytime. According to the Association for Safe International Road Travel [1], about 1.24 million die and 50 million are injured on the roads of the world every year. There are many reasons to why vehicle accidents may happen, for example, negligence while turning a radio on/off, searching for temperature control, searching for light control, etc. Fig. 1 [2] shows a study for vehicle accident accordance in India. In many parts of the world (i.e., in India), traffic accidents constitute a major social problem. Statistically, they are considered the second leading cause of death.

To overcome such problems, many vehicle manufacturers and automobile device companies have attempted to develop speed control systems to maintain a vehicle safe distance.

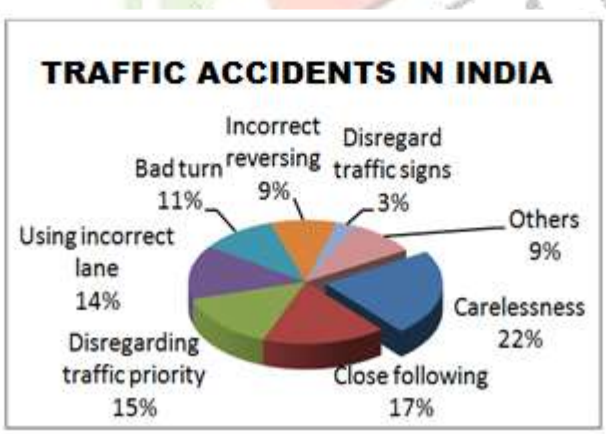


Fig. 1 Causes of car accidents in India

They generally used two methods; one is the Laser-based approach while the other is the Radar-based approach. In the Laser-based type, the systems do not track vehicles in difficult weather conditions nor do they reliably track extremely dirty vehicles. Another implication imposed is that the sensors must be exposed. In the Radar-based type, the system can be hidden behind plastic fascias. The Mitsubishi Inc. [3] was the first automaker to offer a laser-based ASC system in 1995 on the Japanese Mitsubishi Diamante marketed as "Preview Distance Control". More manufactures then used these features such as BMW, Mercedes-Benz, Audi, Honda, and Hyundai.

Researchers have recently started to explore the introduction of additional objectives into the automated vehicle control, e.g., fuel economy and driver desired speed control. Shengbo et al. [4,5] suggested the use of automated speed control to reduce fuel consumption of vehicles and considered issues of driver desired response and hence indicated the necessity to adapt to drivers' individual driving requirements.

Automated vehicle control systems have been recently developed using various detection sensors for speed control, such as long-and-short range radar, wide dynamic range cameras, ultrasonic sensors and laser scanners. Among these detection sensors, radars can quickly provide drivers with reliable information about the velocity, distance, and direction of a target obstacle, as well as information about the vehicle in changing weather conditions. Jeoung et al. [6] stated that three radar sensors are usually needed because two short range radars are used to detect objects in the adjacent lane and one long range radar to detect objects in the desired path. However, it is found that ultrasonic and laser sensors have the advantage of providing fast responses which convince most of the design requirements. The main difference between ultra sonic and laser sensors is the sensing range, i.e. range of ultra sonic is wider, which is a major problem in the ASC systems. Payman and Andrzej [4] have also investigated the issue of automated vehicle control and suggested control algorithms considering the stop & go function with adaptive speed control. Shengbo et al. [5] discussed the economy benefits using automated speed control based on reducing fuel consumption. Others suggest that the best sensors for this objective are the radar sensors [6]. Moreover, Rogowski [7] discussed the main advantages of voice control systems in industrial applications. Englund [8] discussed the effect of G- load on voice control systems and proposed some solutions.

## I. Design Procedure

### *2.1 Automated Speed Control (ASC)*

The general design for any ASC system is implemented as seen in Fig. 2. In this paper, the Arduino Mega is used to give all required functions for the proposed vehicle safety control. The Arduino Mega system is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator along with other crucial connections. It contains all necessary parts needed to support the microcontroller and is powered by a computer with a USB cable [9]. Even though the Arduino system is an educational tool to test new ideas, however, for real realization, it can be easily converted using an ATmega2560 microcontroller (used in the Arduino mega board) with the suitable connections.

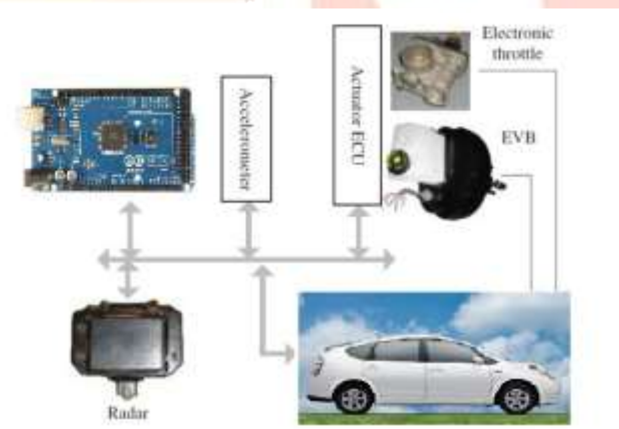


Fig. 2 Configuration of experimental platform

In the proposed work, the ASC is performed using the Arduino map function which, in a way, is similar to a fuzzy controller. The map function construction is: "map (value, from Low, from High, to Low, to High)". The function remaps a given number from one range to another. For example, a given value of "from Low" would get mapped into "to Low", a value of "from High" into "to High", values in-between into different value [9]. However, value from **above** "from High" will be mapped into **above** "to High" and hence another role is used "constrain" to keep the value in a proper range [9]. The value provided by the map function represents the distance measured by the ultra sonic sensor as it falls in a certain range. If the value falls into a lower range, then the vehicle is to be stopped. The proposed ASC can be summarized by the flowchart shown in Fig. 3.

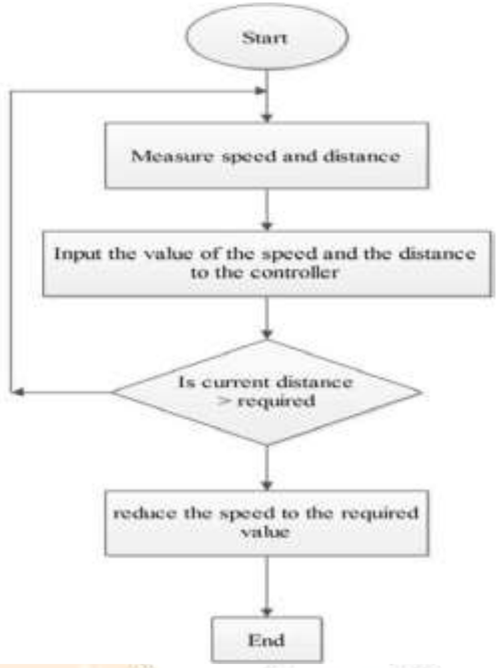


Fig. 3 Speed controlling algorithm

As generally illustrated in Fig. 3, a command is sent to trigger the ultra sonic signal where the time required for the echo signal to return is calculated. The distance is calculated using the following sound equation:

$$Range = T_h(V / 2) \qquad (1)$$

Where $V$= 340 m/s and $T_h$ is the High level time. In the next step, the vehicle speed is calculated using the appropriate sensor, e.g. accelerometer. The values of the distance and speed are entered to the controller algorithm where a test algorithm is made to test if the required distance is greater than the current distance. If the required distance is less than the current distance, then the vehicle should maintain its speed, otherwise, the speed should be reduced to a value that is less than the maximum required speed.

### *2.2 Voice Recognition*

There are various applications for voice gratitude including voice user interfaces such as voice dialing, call routing, demotic appliance control, search, simple data entry, preparation of structured documents, and voice-to-text processing. In this paper, the purpose of using voice recognition is to maximize the driver safety by allowing the line with various functions via voice commands to overcome different distraction while driving. Some of the distractions maybe looked at as flasher control, washers, radio etc.

Generally, the performance of voice recognition systems is evaluated in terms of accuracy and speed. Accuracy is usually rated with Word Error Rate (WER), whereas speed is measured with real time factor. However, voice recognition (by a machine) is a very complex problem. Vocalizations vary in terms of accent, pronunciation, articulation, roughness, nasality, pitch, volume, and speed. Voice maybe distorted by a background noise, echoes, or electrical characteristics. Hence, the accuracy of voice recognition may generally vary according to the followings:

- Vocabulary size and confusability
- Speaker dependence vs. independence
- Isolated, discontinuous, or continuous voice
- Task and language constraints
- Read vs. spontaneous speech
- Adverse conditions

In this work, a speaker dependent module called simple VR is used for the voice recognition operation. Easy VR is used as a user interface to allow for voice recognition training. It has 16 groups each group can take up to 32 user-defined Speaker Dependent commands. The groups can be used for different systems or subsystems depending on the number of commands that the user needs to activate.

Using the simple VR module, the Algorithm of recognizing a given voice is performed for two operations; training and recognition. For training, a spoken word is stored using certain software so as to make the system ready to recognize different words. To recognize a word, the word is spoken into a microphone to provide a certain vector that can be translated into a current (High) to make a certain function. A general part of the vehicle safety control algorithm is given by the following:

```
#include "EasyVR.h"
#define TP 12 //Trig_pin
#define EP 13 //Echo_pin
boolean flasher= true;
boolean radio= true;
 boolean wiper= true;
 boolean left = true;
 boolean right = true;
boolean move1 = true;
 boolean reduce = true;
EasyVR easyvr(Serial);
Serial.begin(9600);
}
void loop()
{
  check_distance();
easyvr.recognizeCommand(1); while
(!easyvr.hasFinished()){
check_distance();
  }
  int index = easyvr.getCommand();
   if(index==0||index==1||index==2){
   if(flasher==true){
      digitalWrite(30,HIGH);
      flasher=false;
    }
  else{ digitalWrite(30,LOW);
  flasher=true;
    }
  }
if(index==3||index==4||index==5){
 if(radio==true){ digitalWrite(32,HIGH);
 radio=false;
    }
    else{
   digitalWrite(32,LOW);
   radio=true;
     }
   }
```

The Easy VR shield is a "slave" module communicate via an asynchronous serial interface (commonly known as UART interface) designed with the following specifications:

- Baud Rate: 9600 (default), 19200, 38700, 57600, 115200.
- Frame: 8 Data bits, No parity, 1 Stop bit.

The receiver input data line is ERX, while the transmitter output data line is ETX (no handshake lines are used).

The microphone, provided with the Easy VR module, is a unidirectional electric condenser microphone. It is designed with the following specifications:

- Sensitivity = -38dB.
- Load Impedance 2.2K
- Operating Voltage 3V
- Almost flat frequency response in range 100Hz – 20 kHz
- Adapter board

To recognize a voice using the Easy VR module, a "dB" (present as a character) is first sent to the Easy VR to begin the credit process. Then, a check is performed to assure that there is a sequential communication response. Once the reply is positive, the module checks for the received nature "r" from the simple VR which indicates that a rule has been recognized. Then, a "20" hexadecimal code is sent for the rule index and is required to wait until the command is sent from the Easy VR serial communication. As a result, the appropriate rule related to the received index is performed. This process may simply be illustrated as seen Fig. 4.

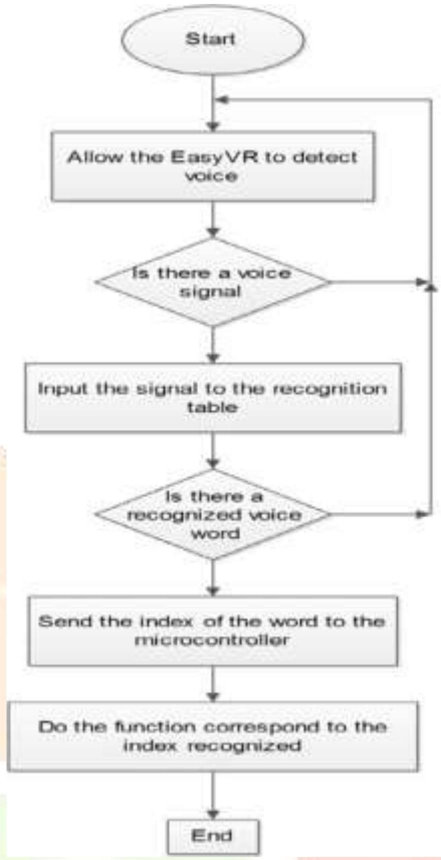


Fig. 4 Voice recognition algorithm

The Easy VR module can save up to 32 different commands. In this work, 7 functions were designed which can be stated as following:

- Flasher
- Left Flasher
- Right Flasher
- Radio
- Wipers
- Go
- Stop

Communication with the simple VR module uses a standard UART interface well-suited with 3.3-5VTTL/CMOS logical levels according to the powering voltage VCC. A representative link to an MCU-based horde is typically specified as seen in Fig. 5 [10].

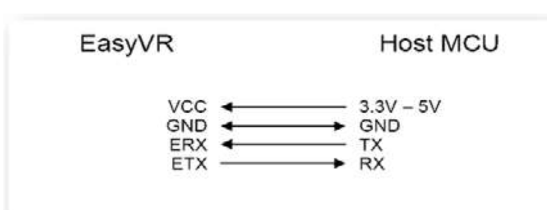


Fig. 5 MCU based host connection

The configuration at power/On is 9600 baud, 8 bit data, No parity, 1 bit stop. The baud rate can be altered as necessary to operate in the range 9600 - 115200 baud.

The communication protocol only uses printable ASCII characters, which can be divided in two main groups:

- Command and status characters, respectively on the TX and RX lines, chosen among lower-case letters.
- Command arguments or status details, again on the TX and RX lines, spanning the range of capital letters.

Each command is sent on the TX line with zero or more additional fight bytes and receives an answer on the RX line in the form of a status byte followed by a zero or more arguments. The proposed design model is obtainable as shown in Fig. 6.

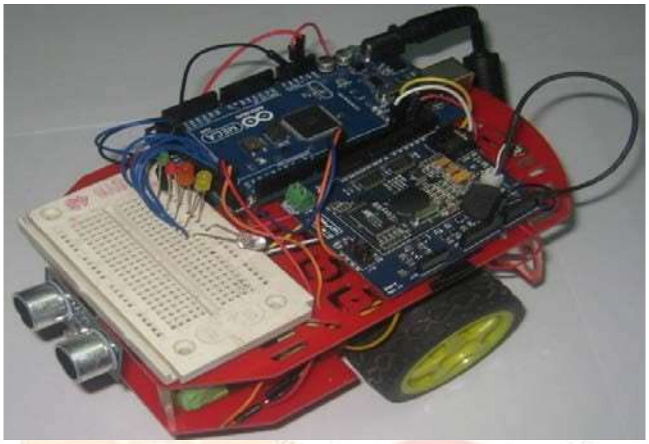

Fig. 6 Design of a vehicle safety control model

## II. CONCLUSION

As it can improve vehicle safety issues and provide the driver with more vehicle operation ease, which leads to minimizing the number of vehicle accidents, an automated vehicle safety control is presented in this paper. The design is implement using an Arduino microcontroller. An Easy VR is used for voice recognition process and training. For the vehicle speed control, the essential action is issue based on the vehicle-to-vehicle space measure. To retain a suitable road and position hub, the design has been provided with different other preset control functions; radio control, flasher control, light control. All of these control options are manage according to human voice instructions provided by the Easy VR module. The projected model has been designed, built, and tested providing suitable outcome as expected, which confirm the probable for further future work with better vehicle safety robotic operations.

.

## III. ACKNOWLEDGEMENT

We would like to thank the Raajdhani Engineering College for supporting this research.

## REFERENCES

[1] ASIRT, “http://www.asirt.org/home.aspx”, Association for Safe. International Road Travel, 20

[2] Mitsubishi,“http://www.mitsubishi.com/e/mitsubishihp/ ”, Mitsubishi.com Committee, 2013

[3] P. Shakouri, A. Ordys, and M.R. Askari, “Adaptive cruise control with stop and go function using the state-dependent nonlinear model predictive control approach,” ISA Transaction, vol. 51, no. 5, pp. 622-631, 2012.

[4] S.E. Li, K. Li, and J. Wang, “Economy-Oriented Vehicle Adaptive Cruise Control With Coordinating Multiple Objectives Function,” International Journal of Vehicle Mechanics and Mobility, vol. 51, no.1, pp. 1-17, 2013.

[5] S. H. Jeong, J. E. Lee, S. U. Choi, J. N. Oh and K. H. Lee,“TechnologyAnalysis And Low-Cost Design Of Automotive Radar http://dx.doi.org/10.1007/s12239-012-0116-2

[6] A. Rogowski, “Industrially oriented voice control system,” Robotics and Computer-Integrated Manufacturing, vol. 28, no. 3, pp. 303–315, June 2012http://dx.doi.org/10.1016/j.rcim.2011.09.010

[7] C. Englund, “Speech recognition in the JAS 39 Gripen aircraft - adaptation to speech at di_erent G-loads,” Master Thesis, Royal Institute of Technology, Stockholm, 2004.

[8] Arduino mega2560, “http://arduino.cc/en/Main/arduinoBoardMega2560”, 2013.

[9] Easy VR Arduino Shield, “http://www.veear.eu/products/easyvr-arduino- shield/”, Veear company, 2013.