Collision Avoidance System in Cars

Electronic Design Lab II Project Report

Group No. D-06

Chandrakant Agarwal (06d07030)

Anish Garg (06d07026)

Nekman Verma (06d07022)

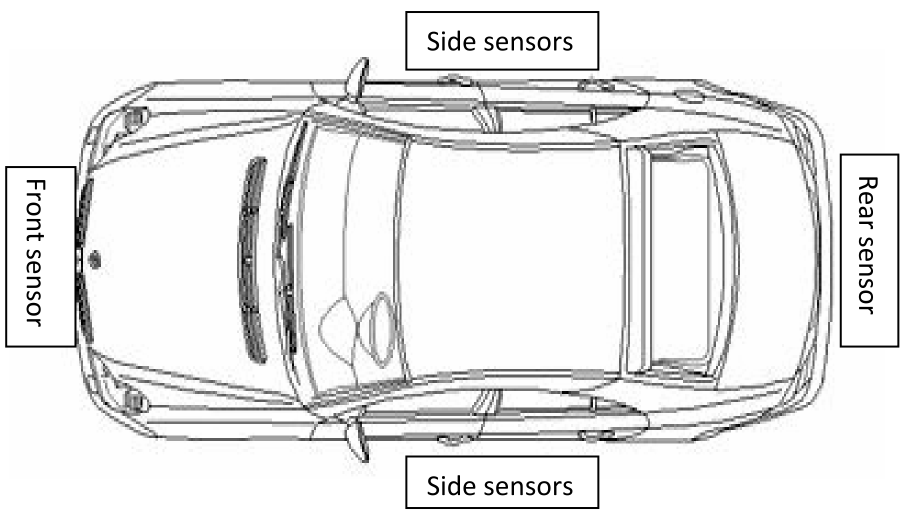
Kinjal Bhavsar(06d07003)

Department of Electrical Engineering

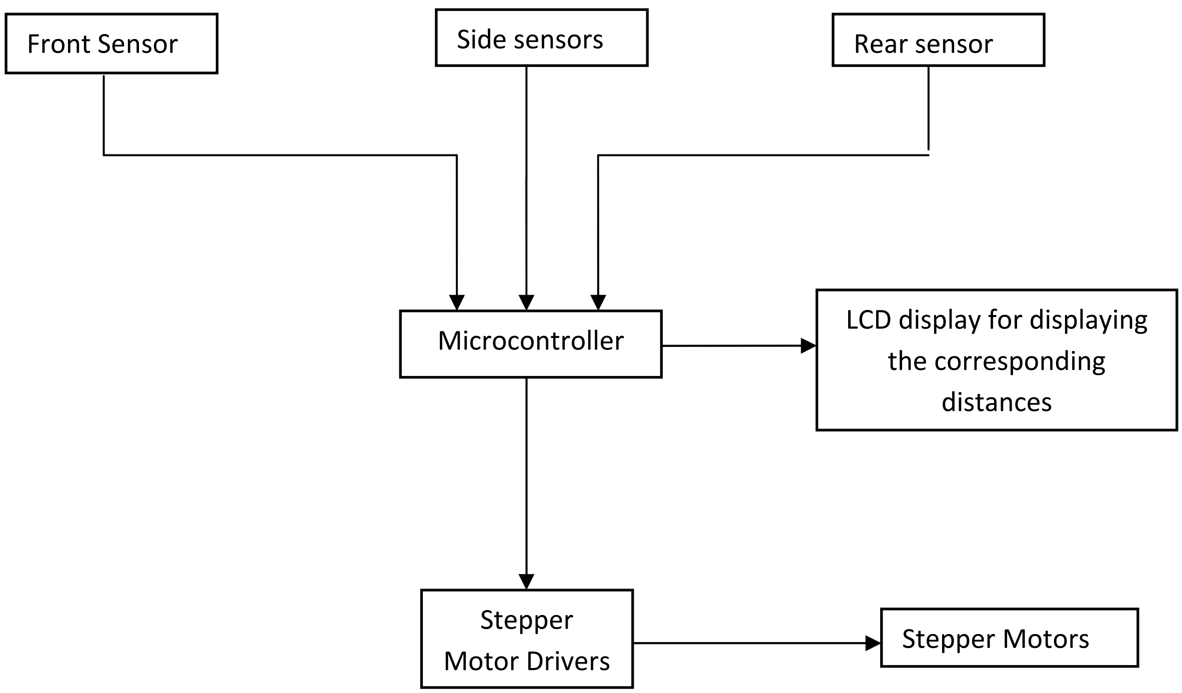
Indian Institute of Technology Bombay

Abstract

The objective of this project is to develop a safety feature in cars to avoid colliding with a vehicle or an obstacle in the way. The main objective of this system is to help driver prevent car collisions due to blind spots and their carelessness while driving. Collision avoidance systems are especially useful in bad weather conditions. The sensors in the car would be capable of detecting even in the poor conditions and would inform the driver distance from the various objects around the car which will help the driver to drive safely in such poor conditions and a central microcontroller would also be able take decisions according to different situations. For example, if a car is approaching from back and there is no obstacle in the front then the microcontroller will accelerates the car by itself and also fog affects visibility, the sensors would recognize another car and alert the driver of any dangers that lie ahead, giving the driver enough time to slow down, allowing him to escape from what could have been a bad accident.



Block Diagram



1st model

This model that we proposed earlier and tested involved msp430.

Main Components:

1. MSP430F413 – Microcontroller
2. 40 KHz Ultrasonic Transducers
3. SN74194 – Universal Shift Register
4. ULN2003 – Darlington Arrays

Concept behind this model

40-kHz ceramic ultrasonic transducers are to transmit and receive the ultrasonic sound waves. The MSP430 drives the transmitter transducer with a 12-cycle burst of 40-kHz square-wave signal derived from the crystal oscillator, and the receiver transducer receives the echo.

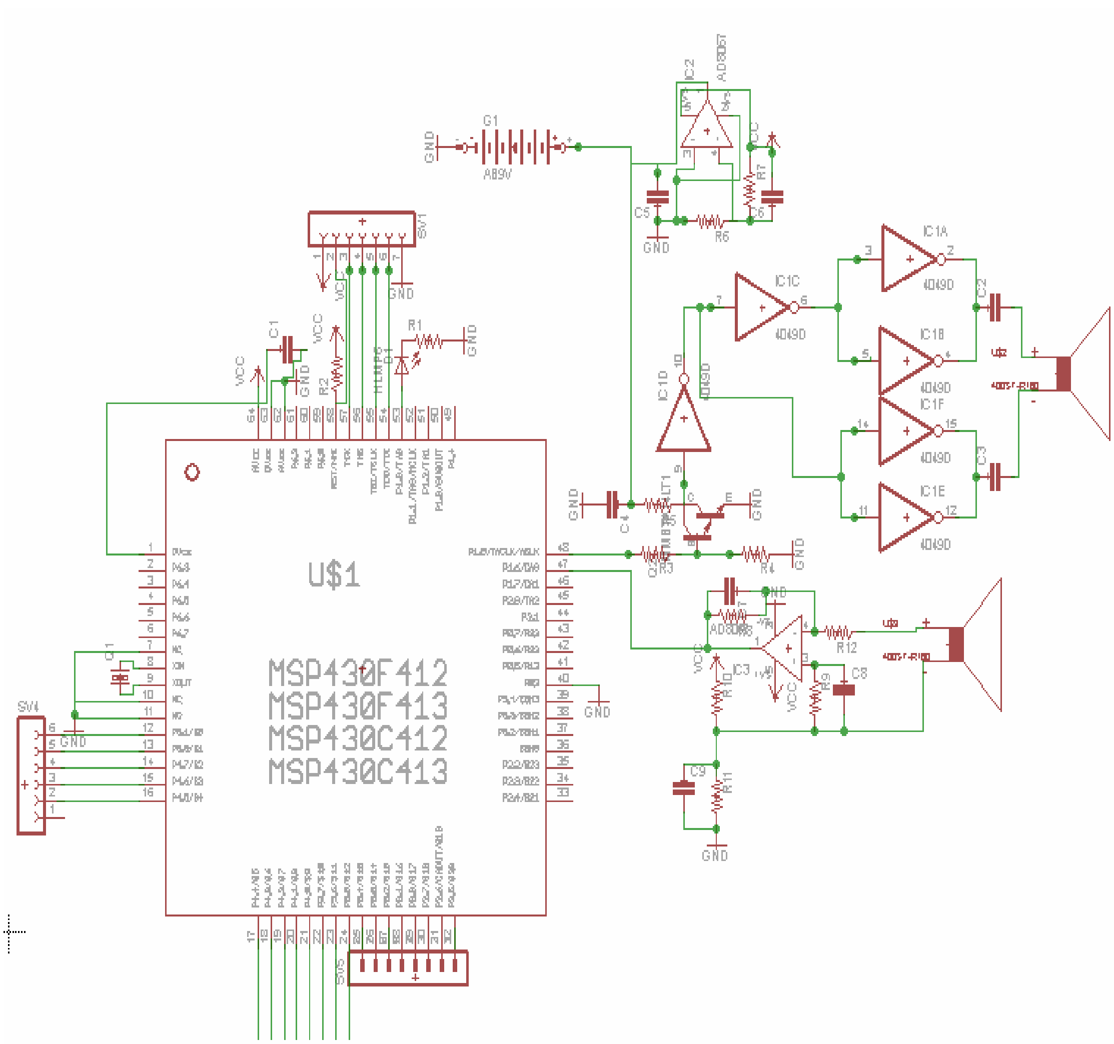
The measurement time base is very stable as it is derived from a quartz-crystal oscillator. The echo received by the receiver transducer is amplified by an operational amplifier and the amplified output is fed to the Comparator_A input. The Comparator_A senses the presence of the echo signal at its input and triggers a capture of Timer_A count value to capture compare register CCR1. The captured count is the measure of the time taken for the ultrasonic burst to travel the distance from the system to the subject and back to the system. The distance in inches from the system to the subject is computed by the MSP430.

The Basic Timer1 is programmed to interrupt the MSP430 every 205 milliseconds. The interrupt signal from the Basic Timer1 wakes up the MSP430 to repeat the measurement cycle and update the display.

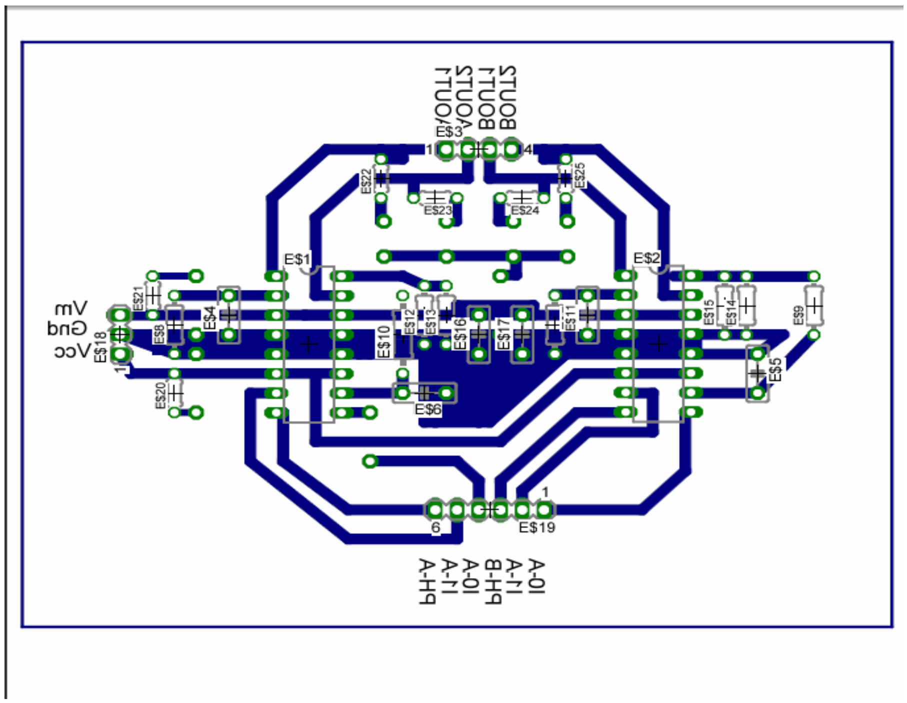
The output drive circuit for the transducer is powered directly from the 9-V battery and provides 18 VPP to drive the ultrasonic transmitter. The 18 VPP is achieved by a bridge configuration with hex inverter gates U4-CD4049. TLV2771 has a high-gain bandwidth and provides sufficiently high gain at 40 kHz.

In this case 0f 4 sensors, 40 khz bursts are sent to all the four US transmitters together, then polling is done via a Multiplexer/Demultiplexer to obtain the US transmitter where the echo has been received and accordingly the stepper motors are moved to avoid the obstacles.

Schematics for the msp-sensor PCB



BRD design of the stepper motor driver



The stepper motor driver works fine but msp430 circuit did not give efficient results with the US transducer, therefore post-midsem we shifted our focus and started to work with ATMEGA32 building the circuit on similar lines and concept as the msp430 circuit.

2nd model

In this model instead of msp430 we use ATMEGA 32 and use a simpler version of the circuit with same guidelines as before.

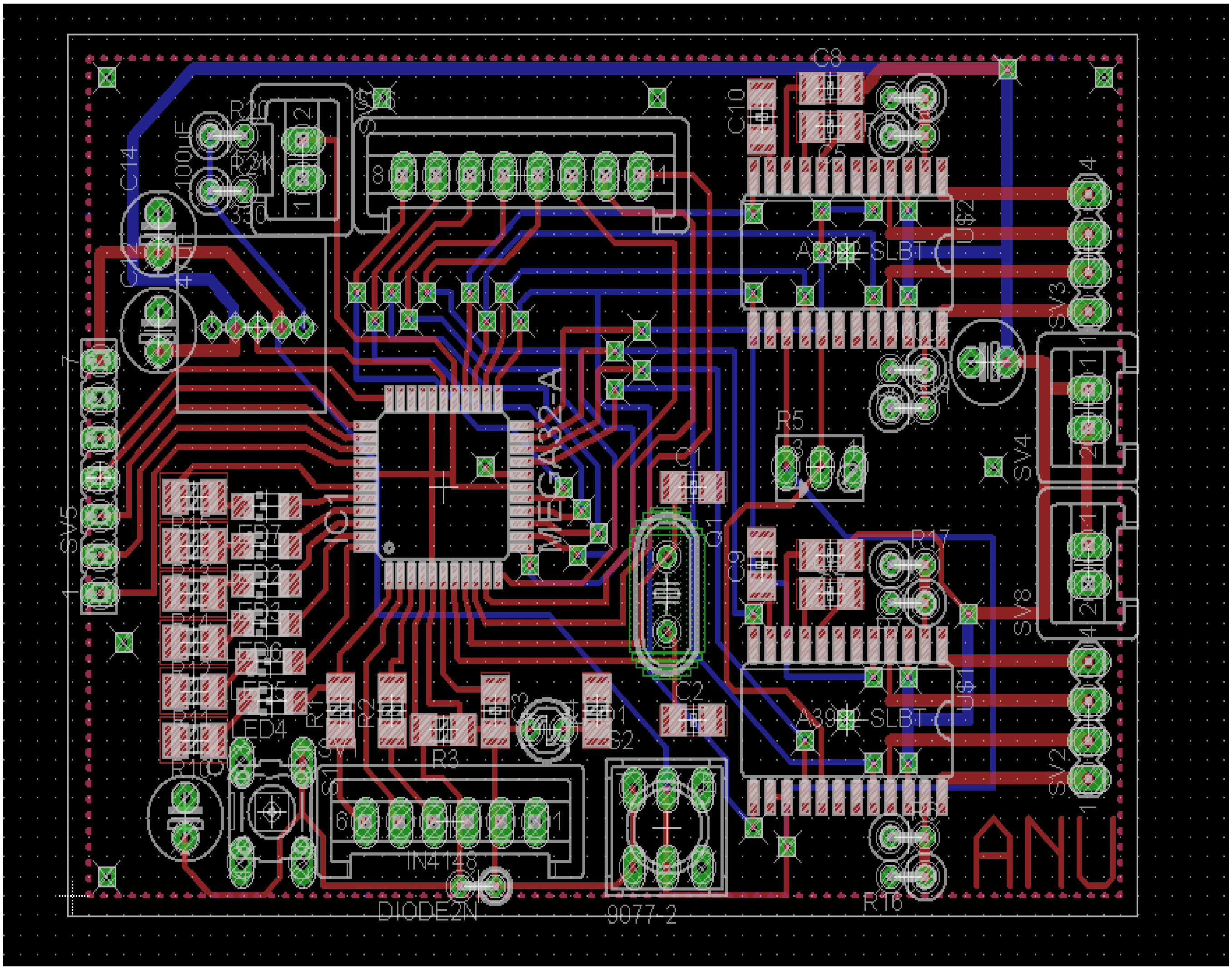
Main components

1. Atmega 32 microcontroller
2. 2 A3982 mototor drivers
3. 40 Khz US transducer
4. A3982 stepper motors

Concept behind this model

The input from the sensor are sent to the ADCs of port A namely PA0, PA1, PA2 and PA4. The interfacing of Atmega 32 with A3982 is as follows. PC0 and PC6 are connected to the 20th pin i.e the DIR pin of the 2 A3982 motor drivers respectively. PC1 and PC5 are connected to the STEP. PC3 and PC4 act as the common Enable and Reset and PC2 and PC7 as the MS1 respectively. Now the microcontroller follows a logic depending upon the state of the 4 sensors. There are 5 sets of things that the motor is allowed to perform namely moving straight, turning right, turning left, pausing and moving 180 degrees. And hence the code has been designed using these 4 things to successfully avoid collisions according to our problem statement. For instance when the sensor at the left goes high, the motor is paused and then moved right. It is allowed to go straight until the sensor at the back of the motor goes low. Then it is again paused and allowed to move left. Then it is allowed to continue its motion straight. Similar logic is applied if the sensor on the right goes high. If the sensor at the back goes high the machine is allowed to accelerate. And if the sensor at the front goes high, the machine is paused, turned right and moved straight until the sensor at the left goes low. Then it is paused and turned left. After that it is allowed to move straight.

Brd design of the main board consisting of Atmega 32 and the motor drivers



ANU
MEGA32-A
IC1
SV1
SV2
SV3
SV4
SV5
SV8
U$1
U$2
C2
C8
C9
C10
C14
R3
R5
R16
R17
LED4
DIODE2N
IN4148
9077-2
100uF
Q1