**CONTENTS**

**ABSTRACT i**

**LIST OF ABBREVATIONS ii**

**CHAPTER 1 INTRODUCTION 1**

1.1 Problem Definition 1

1.2 Objective 1

**1.3** Abstract1

1.4 Existing System 2

1.5 Proposed System 2 **CHAPTER 2 ULTRASONIC 3** 2.1 Introduction 3

2.2Principle 4 2.3 General Over View of Working System 4

2.4 Methodology 6

**CHAPTER 3 HARDWARE and SOFTWARE DESIGN 7**

3.1 Block Diagram 7

3.2 Micro Controller 7

3.3 Features of Micro Controller 9

3.4 Specification of Ultrasonic System 10

3.5 Peripheral Features of Micro Controller 11

3.6 HC-SR04 Ultrasonic Module 113.7 SIM800 Hardware Design

3.7.1 Introduction

3.7.2 SIM800 Overview

3.7.3 SIM800 Key Features

3.7.4 Serial Port and USB Interface

3.8 Transistor BC107

3.8.1 Introduction

3.8.2 Transistor Interface

3.8.3 Terminals and Functions

3.8.4 Configuration of NPN Transistor

3.8.5 Transistor BC107 Module

3**.**9 Integration of Hardware and Software 13

**CHAPTER 4 IMPLEMENTATION 14**

4.1 Introduction 14

4.2 Plan 15

4.3 Building the System 18 4.4 Software Environment 20

**CHAPTER 5 TESTING AND VALIDATION 21**

5.1 Introduction 21

5.2 Types of Tests 21

5.3 Design of Test Cases 23

**CHAPTER** **6 RESULT 25**

6.1 Output Screen 25

**CHAPTER 7 ADVANTAGES 27**

**CHAPTER 8 FUTURE SCOPE 28**

**CHAPTER 9 CONCLUSIONS 29**

**APPENDICES 30 REFERENCES 39**

**CHAPTER 1**

**INTRODUCTION**

**1.INTRODUCTION**

**1.1 PROBLEM DEFINITON:**

Today we all know that many accidents are being taking place at steep curves, bends and hilly areas and ruining out many lives of people and Wildlife. We can prevent these accidents by proposing an idea on SENSORS.

**1.2 OBJECTIVE :**

The main objective of this project is that with the help of Ultra-Sonic Sensors, Micro Controller(Aurdino) and LED’s we can prevent accidents at steep curves and hilly areas.

By this project we can prevent many accidents irrespective of the vehicles having Horns and Lights.

**1.3 ABSTRACT:**

Driving is an essential part in the life of many people. With the immense use of vehicles in day today life accidental death has shown a tremendous growth rate. Due to these accidents, in most of the cases, people lose their life. The main reason for this lies behind the delay in medical facilities to reach to the accidental place. This project ensures the losses of life will be reduced rapidly. Intelligent Transportation Systems covers a wide range of applications that include communication system, positioning, sensing, and other information related to technologies to improve the safety, efficiency and environmental aspect of the surface transportation. The various transport technologies are car navigation, traffic signals, container management system, variable message signs, automatic number plate recognition or speed cameras to monitor applications such as security, CCTV systems. The purpose of ITS to increase mobility, to make driving safer, more efficient Eco - friendly, cost effective, energy saving, convenient and comfortable.

**1.4 EXISTING SYSTEM:**

**1.4.1ABS (Anti-Locking Braking System)**:

ABS works with your regular braking system by automatically pumping them. In vehicles not equipped with ABS, the driver has to manually pump the brakes to prevent wheel lockup. In vehicles equipped with ABS, your foot should remain firmly planted on the brake pedal, while ABS pumps the brakes for you so you can concentrate on steering to safety.

**1.4.2 EBD (Electronic brake-force distribution)**:

Electronic brake-force distribution (EBD or EBFD), Electronic brake-force limitation (EBL) is an automobile brake technology that automatically varies the amount of force applied to each of a vehicle's brakes, based on road conditions, speed, loading, etc. always coupled with anti-lock braking systems.

**1.4.3 SRS Air Bags (Supplemental Restraint System Air Bags)**:

An airbag is a vehicle safety device. It is an occupant restraint consisting of a flexible envelope designed to inflate rapidly during an automobile collision, to prevent occupants from striking interior objects such as the steering wheel or a window, the sensors may deploy one or more airbags in an impact zone at variable rates based on the type and severity of impact; the airbag is designed to only inflate in moderate to severe frontal crashes.

**1.5 PROPOSED SYSTEM:**

In this project with the help of Ultra-Sonic Sensors, Micro Controllers(Aurdino), and LED’s we can prevent accidents at steep curves and hilly areas.

By this project we can prevent many accidents irrespective of the vehicles having horns and lights.

**CHAPTER 2**

**ULTRASONIC**

# 2.1 INTRODUCTION

The easy and accurate measurement of distance has been major subject of study in the field of engineering and physics from the time men began to walk on earth. Many different techniques have been devised till date for the measurement of distance from the observer to a target, for the purposes of surveying, navigation, determining focus in photography, or accurately aiming a weapon. Manual distance measuring is always done at the expense of human error. Employing electromagnetic waves for distance measurement gathered significance with the advent of research in the field of electromagnet.

The use of infra-red rays presents a simpler solution for the question at hand, while the cost of this simplicity arises in the form of a very low range and a significant error in the result. Here is when the technique proposed in the paper comes in. The proposed system employs ultrasonic waves for distance measurement, the highly directional properties of the wave and comparatively lower attenuation encountered makes it highly suited for distance measurement.

Objective of the proposed technique was to develop a device based on a highly that can be used to measure the distance of the target with high precision using Micro Controller. Focus has been given on lower ranges considering the range 1 cm to 2 sm with the precision of 10.1 cm using standard ultrasonic of transducer HC-SRO4.For contact less measurement of distance, the device has to rely on the target to reflect the pulse back to itself .

The target needs to have a proper orientation that is, it needs to be perpendicular to the direction of propagation of the pulses. The amplitude of the received signal gets significantly attenuated and is a function of nature of the medium and the distance between the transmitter and target.

The pulse or flight method of range measurement is subject to high  
echo time of levels of signal attenuation when used in an air medium, thus limiting its distance range.

**2.2 PRINCIPLE**  
An Ultrasonic transducer uses the various physical properties of ultrasound of a specific frequency. The ultrasonic transducers are available in piezoelectric and electromagnetic makes. The piezoelectric type is generally preferred due to its lower cost and simplicity to use It is capable to transmit and receive the ultrasonic signal of a required strength. The Ultrasonic wave propagation velocity is subject to temperature of the medium, in the air it is approximately 340 mis at 15 c of air or atmospheric temperature, the same as sonic velocity. The velocity (v) may be computed as a function of temperature as shown.  
  
 **V=(340+0,60T-15)m/s,**  
  
Where, The temperature Celsius, Here a room temperature of 20 oC is assumed; hence the velocity of ultrasound in the air is taken as 343 m/s. Because the travel distance is very short, the travel time is not affected by temperature to a greater extend. The distance traveled by the wave at time t seconds can be given by

**Distance (D) = k\*v their k- correction factor**

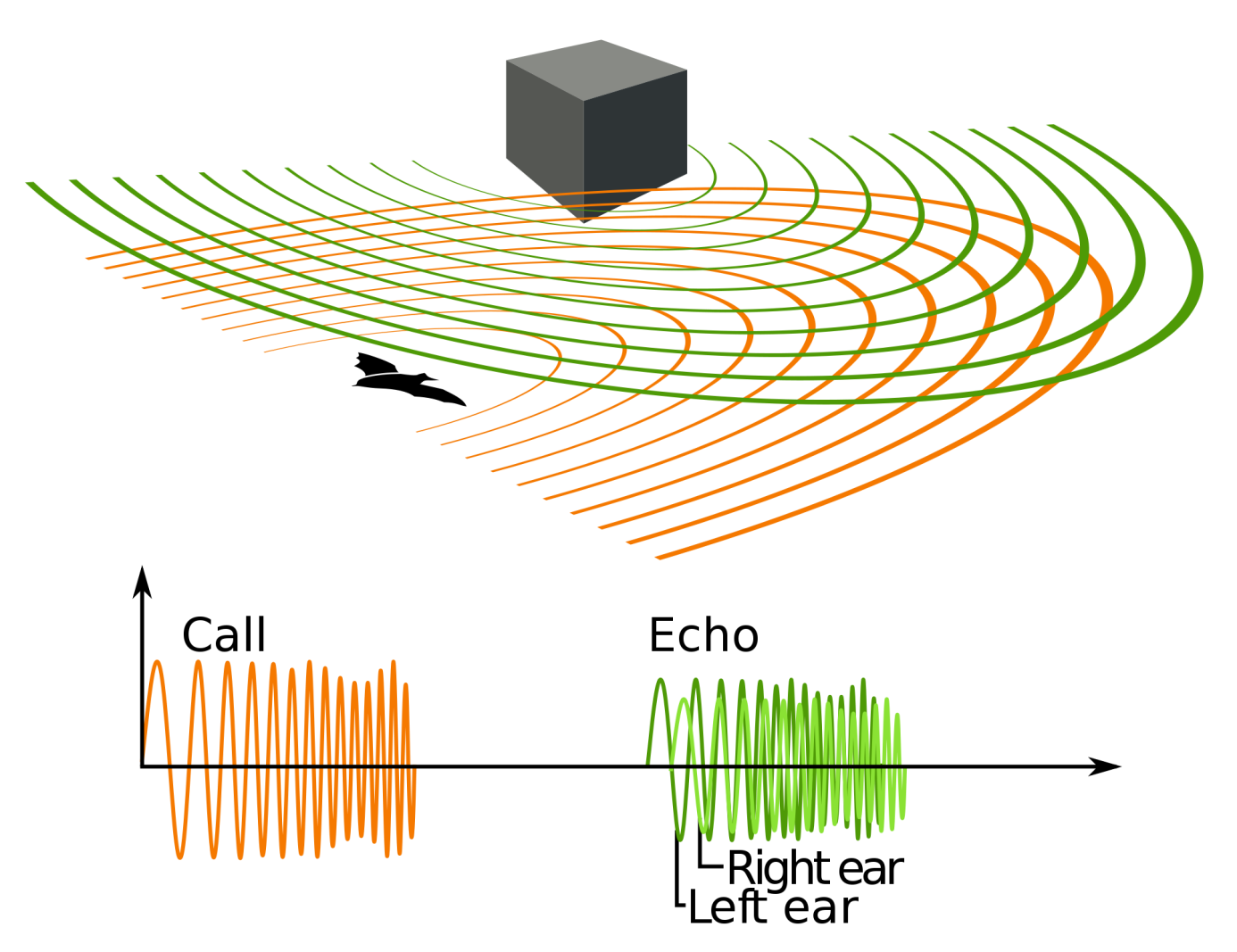
**2.3 GENERAL OVERVIEW AND WORKING OF THE SYSTEM**

The measurement process is initiated by sending a trigger signal to the ultrasonic module. When the resot pulse is given to the processor, it produces a trigger pulse of l us and transfers to the HC-SR04( ultrasonic module ). The trigger signal must be a pulse with 10ps high time. When the module receives a valid trigger signal it issues 8 pulses of 40 KHz ultrasonic sound from the transmitter. The echo of this sound is picked by the receiver, after getting the echo of the ultrasonic sound, the module produce a signal at the echo pin who's high time is proportional to the distance to be measured.

**Test distance = (high level time x velocity of sound (340M/S))/2**

**Distanceincm=EchopulsewidthinµS/58**  
  
**Distanceininch=EchopulsewidthinµS/148**

Finally the distance calculated based upon the pulse width of the echo signal is send to the LCD segment and the range is displayed in centimeters. This ultrasonic rangefinder can measure distances up to 2.5 meters at accuracy of 0.1 Centimeter.  
  
**How do bats use ultrasound to navigate? .**  
  
They can hear high-frequency sounds that the bats make continually while flying Bats can determine the direction and distance of objects in the area. This is called echolocation, each species of bat uses echolocation to make its own kind of noises.



**fig:1**

**2.4.METHODOLOGY**  
  
The system design required familiarization with adopted technologies for  
contact less range determination. The flaws in the infrared based system and the  
advantages of employing an ultrasonic wave based system was analyzed in detail by referring to the systems designed based on these technologies.

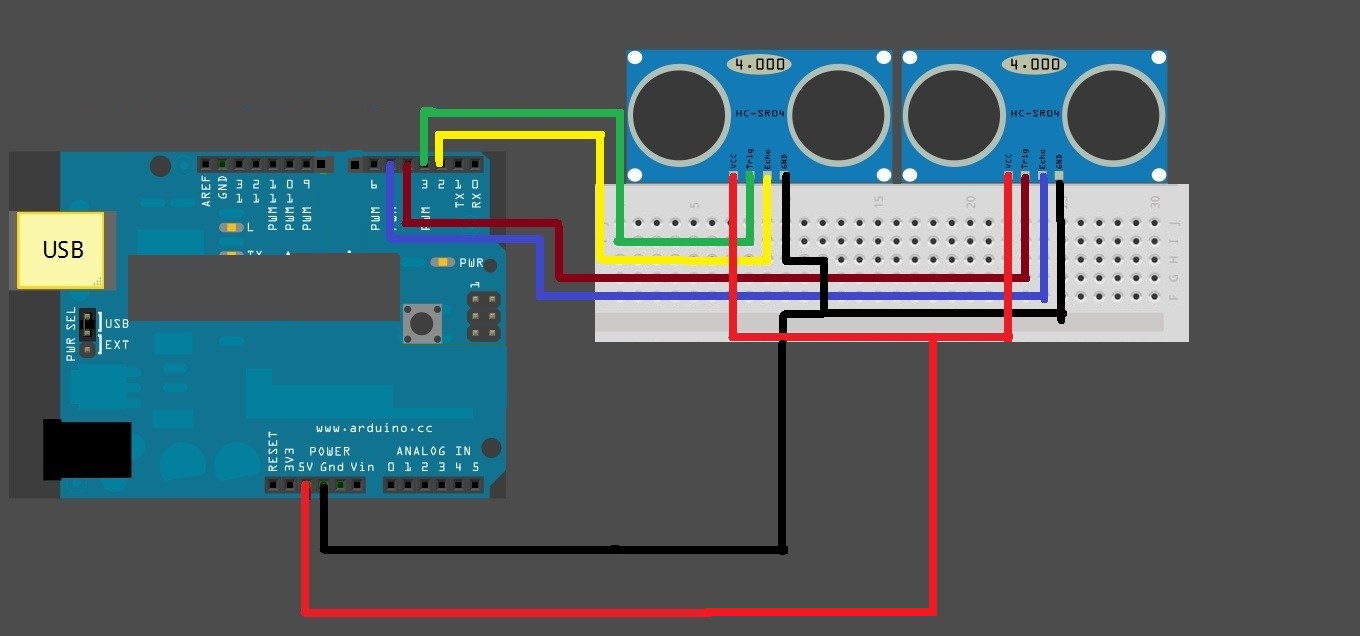
Comparisons were made and the technique was finalized. Extensive literature survey was conducted to determine the best suited components for the system to be fabricated. Program code and the hardware design were completed after considering the required system performance and the constraints like range temperature etc.

**CHAPTER 3**

**HARDWARE and SOFTWARE DESIGN**

# 

**3.1 BLOCK DIAGRAMS**

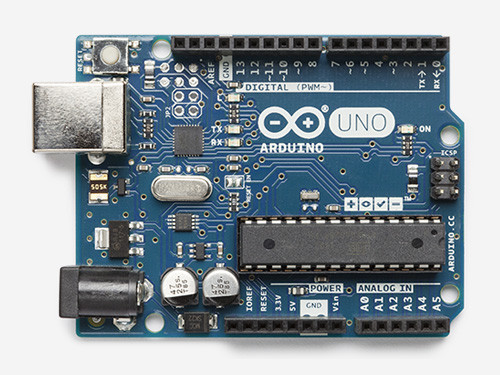


**fig:2**

**3.2 MICRO CONTROLLER**

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the micro controller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great.help.to.novices.and.experts.alike.  
Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

****

**fig:3**

**3.3 FEATURES OF MICRO CONTROLLER**

* Micro Controller - ATmega328
* Operating Voltage - 5V
* Input Voltage (Recommended) - 7-12V
* Input Voltage (Limits) - 6-20V
* Digital I/O Pins - 14 (of which 6 are output)
* Analog Input Pins - 6
* DC Current per I/O Pin - 40mA
* DC Current for 3.3V Pin - 50mA
* Flash Memory - 32 KB of which 0.5 KB used
* SRAM - 2 KB
* EEPROM - 1 KB
* Clock Speed - 16MHz

**3.4 SPECIFICATION OF ULTRASONIC SENSOR:**

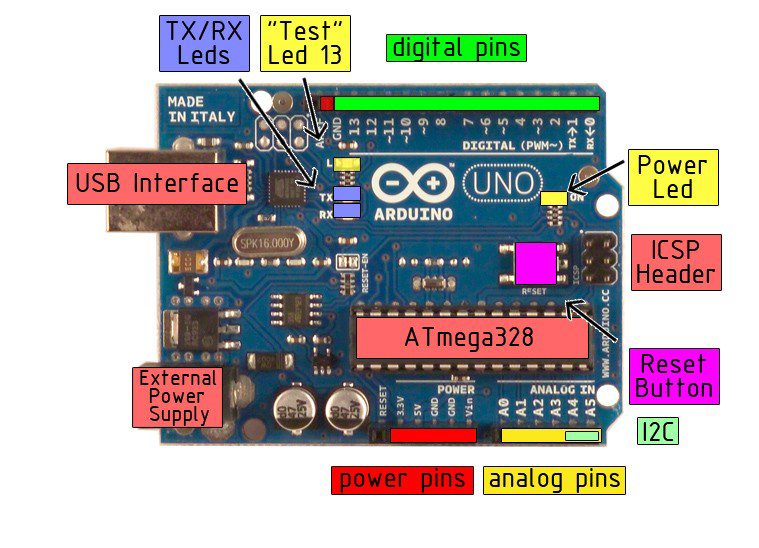
The SRF005 ultrasonic range sensor detects objects in it's path and can be used to calculate the range to the object. It is sensitive enough to detect a 3cm diameter broom handle at a distance of over 3m.

* Voltage 5V.
* Current 30mA Typ.50mA Max.
* Frequency 40 Khz.
* Max Range 3 m.
* Min Range 3 cm.
* Sensitivity-Detect 3cm diameter broom handle at > 3 m.
* Input Trigger 10us Min TTL level pulse
* Echo Pulse Positive TTL level signal.
* Width proportional to range.

**The module can be used in two different modes:**

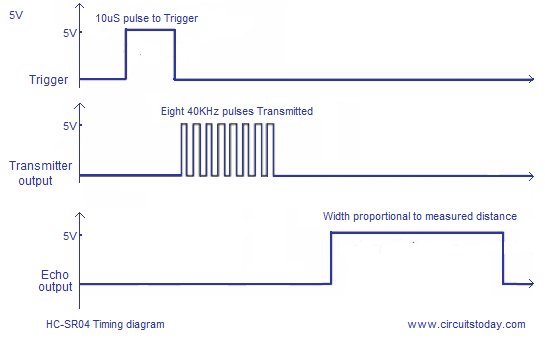
Single Pin Single micro controller pin (08M, and all M2 and X2 parts)  
Dual Pin Separate PICAXE micro controller trigger and echo pins Most  
users using the latest generation (M2 and X2) PICAXE parts should select  
single pin' connection mode.

**3.5 PERIPHERAL FEATURES OF MICRO CONTROLLER**

** fig:4**

**3.6 HC-SR04 ULTRASONIC MODULE**

HC-SRO4 is an ultrasonic ranging module designed for embedded system projects like this. It has a resolution of cm and the ranging distance is from 2cm to 500cm. It operates from a 5vDc supply and the standby current is less than 2mA . The module transmits an ultrasonic signal, picks up its echo, measures the time elapsed between the two events and outputs a waveform who's high time is modulated by the measured time which is proportional to the distance. The supporting circuits fabricated on the module makes it almost stand alone and what the programmer need to do is to send a trigger signal to it for initiating transmission and receive the echo signal from it for distance calculation. The HR-SRO4 has four pins namely Vcc, Trigger, Echo, GND .



**fig:5**

From the timing diagram, you can see that the 40 KHz pulse train is transmitted just after the 10µs triggering pulse and the echo output is obtained after some more time.  
The next triggering pulse can be given only after the echo is faded away and this time period is called cycle period. The cycle period for HC-SR04 must not be below 50ms. According to data sheet , the distance can be calculated from the echo pulse width using the following equations.

**Distance in cm = echo pulse width in µs/58**

**3.7 SIM800 HARDWARE DESIGN**

**3.7.1 Introduction**

This document describes SIM800 hardware interface in great detail. This document can help user to quickly understand SIM800 interface specifications, electrical and mechanical details. With the help of this document and other SIM800 application notes, user guide, users can use SIM800 to design various applications quickly.

**3.7.2 SIM800 Overview**

Designed for global market, SIM800 is a quad-band GSM/GPRS module that works on frequencies GSM 850MHz, EGSM 900MHz, DCS 1800MHz and PCS 1900MHz. SIM800 features GPRS multi-slot class 12/ class 10 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4.

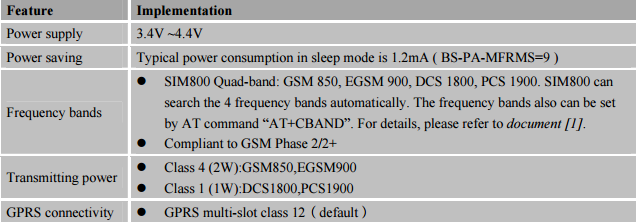
With a tiny configuration of 24\*24\*3mm, SIM800 can meet almost all the space requirements in users’ applications, such as M2M, smart phone, PDA and other mobile devices.

SIM800 has 68 SMT pads, and provides all hardware interfaces between the module and customers’ boards.

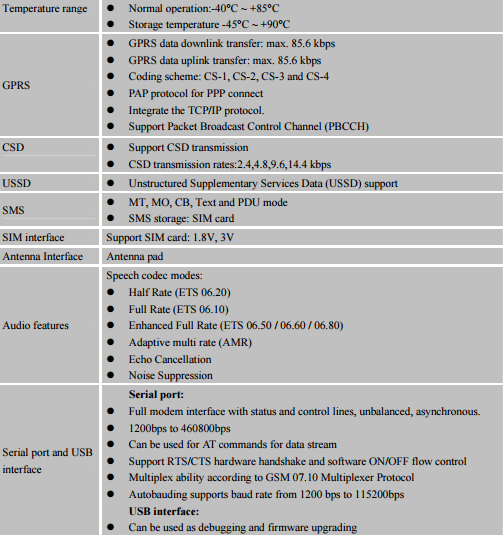
* + Support up to 5\*5\*2 Keypads.
  + One full function UART port, and can be configured to two independent serial ports.
  + One USB port can be used as debugging and firmware upgrading.
  + Audio channels which include a microphone input and a receiver output.
  + Programmable general purpose input and output.
  + One SIM card interface.
  + Support Bluetooth function.
  + Support one PWM.
  + PCM/SPI/SD card interface, only one function can be accessed synchronously.

SIM800 is designed with power saving technique so that the current consumption is as low as 1.2mA in sleep mode.

**3.7.3 SIM800 Key Features**



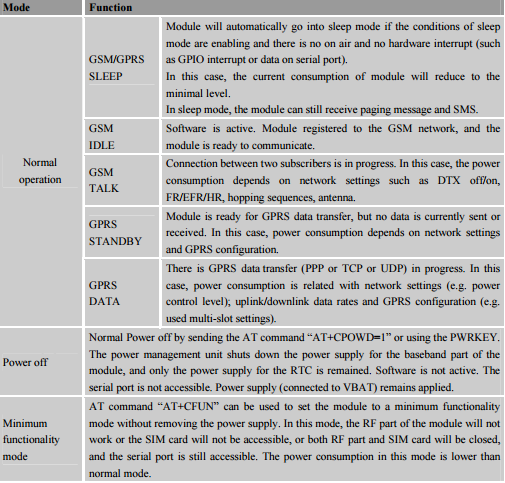
**Table 1**: SIM800 key features

****

**Table 1**: SIM800 key features

**3.7.4 SIM800 Operating Methods**

The table below summarizes the various operating modes of SIM800.This also deals with Normal Operation Mode, Power Off Mode and Minimum Functionality Mode.

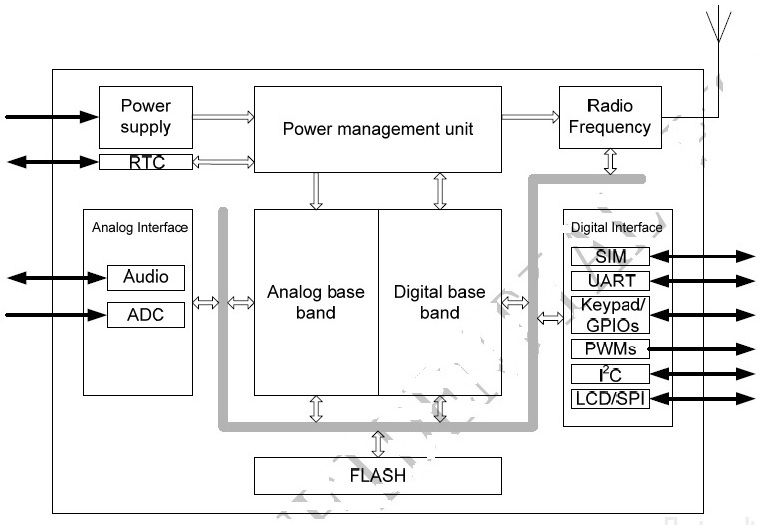


**Table 2**: Overview of operating modes

**3.7.4 SIM800 Functional Diagram**

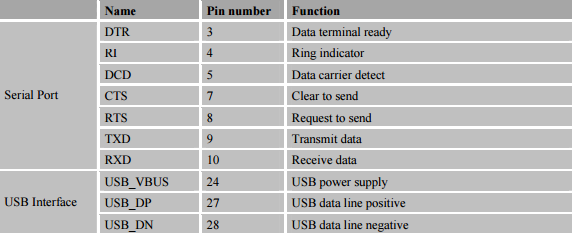
The following figure shows a functional diagram of SIM800:

* + GSM baseband engine
  + PMU
  + RF part
  + Antenna interfaces



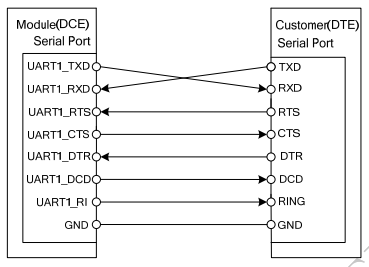
**3.7.5 Serial Port and USB Interface**

SIM800 provides one unbalanced asynchronous serial port. The module is designed as a DCE (Data Communication Equipment). The following figure shows the connection between module and client (DTE).

****

**3.7.5.1 Serial Port**

The following figure shows the connection between module and client (DTE).

****

**3.7.5.2 Function of Serial Port**

Serial port:

* Full modem device.
* Contains data lines TXD and RXD, hardware flow control lines RTS and CTS, status lines DTR, DCD and RI.
* Serial port can be used for CSD FAX, GPRS service and AT communication. It can also be used for multiplex function.
* Serial port supports the following baud rates:
* 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200bps, 230400 and 460800bps;

Autobauding allows SIM800 to automatically detect the baud rate of the host device. Pay more attention to the following requirements:

* **Synchronization between DTE and DCE**:

When DCE powers on with autobauding enabled, firstly, user must send character “A” or “a” to synchronize the baud rate. It is recommended to send “AT” until DTE receives the “OK” response, which means DTE and DCE are correctly synchronized. For more information please refer to the AT command “AT+IPR”.

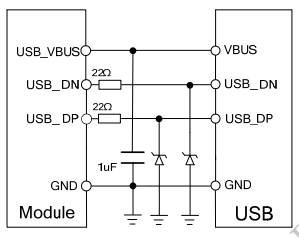
* **Restrictions of autobauding operation**:

The DTE serial port must be set at 8 data bits, no parity and 1 stop bit. The URC such as "RDY", "+CFUN: 1" and "+CPIN: READY” will not be reported.

**3.7.5.2 USB Interface**

USB interface supports software debug function. When power on the module, connect USB\_VBUS, USB\_DP, USB\_DN and GND to PC, then install the driver successfully, a UART port could be recognized by the PC, customer could achieve the software Debug purpose with this UART port.

The following diagram is recommended:

****

**3.7.6 Bluetooth Interface**

SIM800 supports Bluetooth function, customer only needs to design the Bluetooth antenna, and then customer can operate Bluetooth conveniently by AT commands.

* Fully compliant with Bluetooth specification3.0.
* Support operation with GPS and GSM/GPRS worldwide radio systems.
* Fully integrated PA provides 10dbm output power.
* Up to 4 simultaneous active ACL links.
* Support sniff mode.
* Supports PCM interface and built-in programmable transcoders for liner voice with transmission.

**3.8 Transistor BC 107**

**3.8.1 Introduction**

First letter denotes type of semiconductor used. B is for silicon and A is for germanium. Second letter indicates frequency of operation. C is for audio and F is for high frequency (RF) application. Therefore BC107 is a silicon transistor for low frequency (AF) applications. Normally after the number 107 there is one more letter. Which may be A, B or C. This indicates gain of the transistor. Here letter C means highest and A shows lowest gain.

**3.8.2 Transistor Interface**

****

Basically BC 107 is a single purpose Bipolar Junction Transistor (BJT) which is of a NPN type. The working of NPN transistor is quite complex. In the above circuit connections we observed that the supply voltage VB is applied to the base terminal through the load RB. The collector terminal connected to the voltage VCC through the load RL. Here both the loads RB and RL can limit the current flow through the corresponding terminals. Here the base terminal and collector terminals always contain positive voltages with respect to emitter terminal.

If the base voltage is equal to the emitter voltage then the transistor is in OFF state. If the base voltage increases over emitter voltage then the transistor becomes more switched until it is in fully ON state. If the sufficient positive voltage is applied to the base terminal i.e. fully-ON state, then electrons flow generated and the current (IC) flows from emitter to the collector. Here the base terminal acts as input and the collector-emitter region acts as output.

To allow current flow between emitter and collector properly, it is necessary that the collector voltage must be positive and also greater than the emitter voltage of transistor. Some amount of voltage drop presented between base and emitter, such as 0.7V. So the base voltage must be greater than the voltage drop 0.7V otherwise the transistor will not operate. The equation for base current of a bipolar NPN transistor is given by,

**IB = (VB-VBE)/RB**

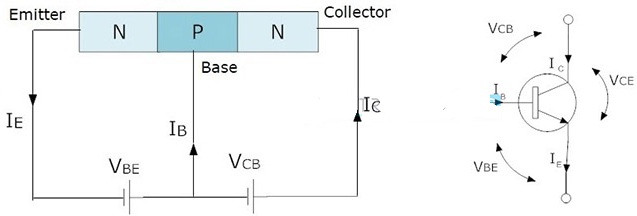
Where,

IB = Base current  
VB = Base bias voltage  
VBE = Input Base-emitter voltage = 0.7V  
RB = Base resistance

The output collector current in common emitter NPN transistor can be calculated by applying Kirchhoff’s Voltage Law (KVL).

**3.8.3 Terminals and Functions**

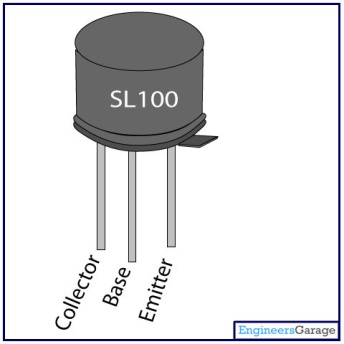
The NPN transistor has three terminals – emitter, base and collector. The NPN transistor is mostly used for amplifying and switching the signals.

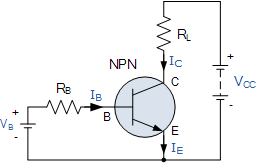


The NPN transistor can be used in two different modes: forward biased mode and the reverse biased mode. In forward biased mode, the electric current *can easily flow through it*. So it acts like a CLOSED SWITCH. However, in reverse biased mode, the current through it is practically zero and thus, it acts like an OPEN SWITCH.

**3.8.4 Configuration of NPN Transistor**

The construction and terminal voltages for a bipolar NPN transistor are shown above. The voltage between the Base and Emitter ( VBE ), is positive at the Base and negative at the Emitter because for an NPN transistor, the Base terminal is always positive with respect to the Emitter. Also the Collector supply voltage is positive with respect to the Emitter ( VCE ). So for a bipolar NPN transistor to conduct the Collector is always more positive with respect to both the Base and the Emitter.



****

Then the voltage sources are connected to an NPN transistor as shown. The Collector is connected to the supply voltage VCC via the load resistor, RL which also acts to limit the maximum current flowing through the device. The Base supply voltage VB is connected to the Base resistor RB, which again is used to limit the maximum Base current.

So in a NPN Transistor it is the movement of negative current carriers (electrons) through the Base region that constitutes transistor action, since these mobile electrons provide the link between the Collector and Emitter circuits. This link between the input and output circuits is the main feature of transistor action because the transistors amplifying properties come from the consequent control which the Base exerts upon the Collector to Emitter current.

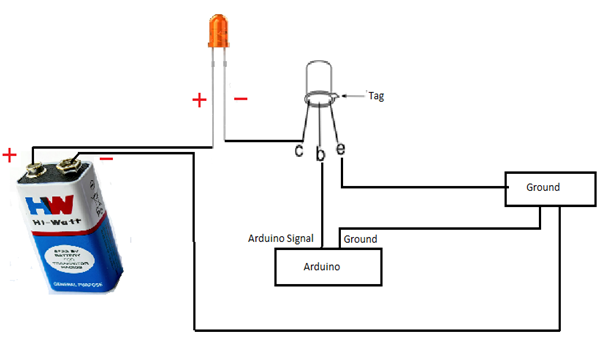
Then we can see that the transistor is a current operated device (Beta model) and that a large current ( Ic ) flows freely through the device between the collector and the emitter terminals when the transistor is switched “fully-ON”. However, this only happens when a small biasing current ( Ib ) is flowing into the base terminal of the transistor at the same time thus allowing the Base to act as a sort of current control input.

The transistor current in a bipolar NPN transistor is the ratio of these two currents ( Ic/Ib ), called the *DC Current Gain* of the device and is given the symbol of hfe or nowadays Beta, ( β ). The value of β can be large up to 200 for standard transistors, and it is this large ratio between Ic and Ib that makes the bipolar NPN transistor a useful amplifying device when used in its active region as Ib provides the input and Ic provides the output. Note that Beta has no units as it is a ratio.

Also, the current gain of the transistor from the Collector terminal to the Emitter terminal, Ic/Ie, is called Alpha, ( α ), and is a function of the transistor itself (electrons diffusing across the junction). As the emitter current Ie is the sum of a very small base current plus a very large collector current, the value of alpha α, is very close to unity, and for a typical low-power signal transistor this value ranges from about 0.950 to 0.999.

**3.8.5 Transistor Module**

The diagram below is called a schematic diagram. Like a breadboard layout, it is a way of showing how the parts of an electronic project are connected together.

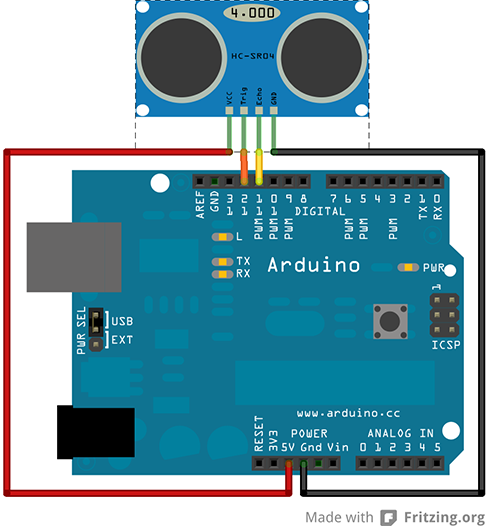
****

The pin D3 of the Arduino is connected to the resistor. Just like when using an LED, this limits the current flowing into the transistor through the base.

There is a diode connected across the connections of the motor. Diodes only allow electricity to flow in one direction (the direction of their arrow).

When you turn the power off to a motor, you get a negative spike of voltage, that can damage your Arduino or the transistor. The diode protects against this, by shorting out any such reverse current from the motor.

**3.7 INTEGRATION OF HARDWARE WITH SOFTWARE**



**fig:7**

**CHAPTER 4**

**IMPLEMENTATION**

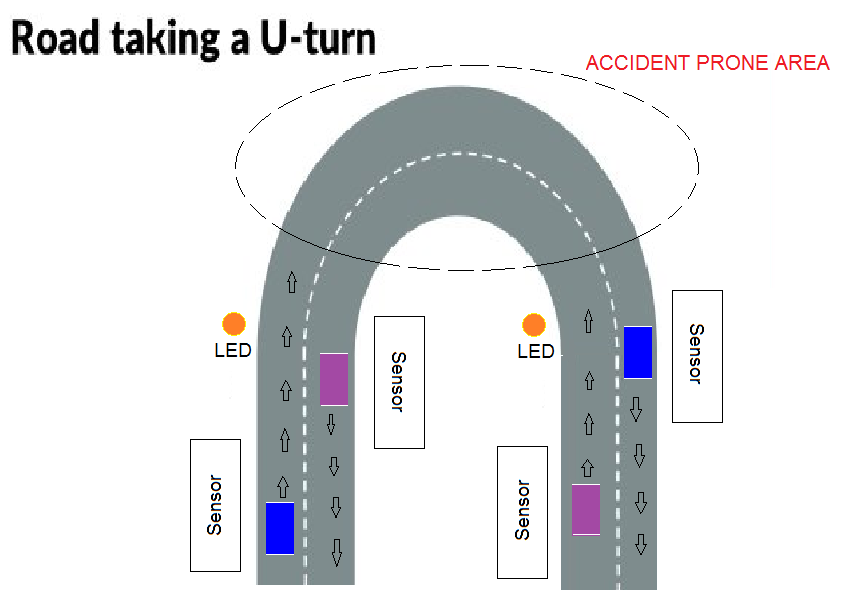
**4.1 INTRODUCTION**

In this project we are introducing sensing technology to the trafic system by using ultrasonic sensors, aurdino, LED's. Here we are using 2 pairs(4 in total) sensors, 2LED's, an Aurdino. Here in each pair one sensor acts as an starting point (in) and the other sensor acts as an ending point (out). Here we use two pairs such that to cover the two way road.

Here when any vehicle or any object will be in the boundary of the sensors then the led will glow. Whenever the object movies from the boundary then the led will turn off. Whenever any object will come in front of sensor the sensor will detect the object and send the signal to the aurdino, after that the output of the sensor is given as input to the LED then the Light will glow, here aurdino acts as a mediator between the sensor and LED. We write code in the Aurdino and the code is embedded in it.

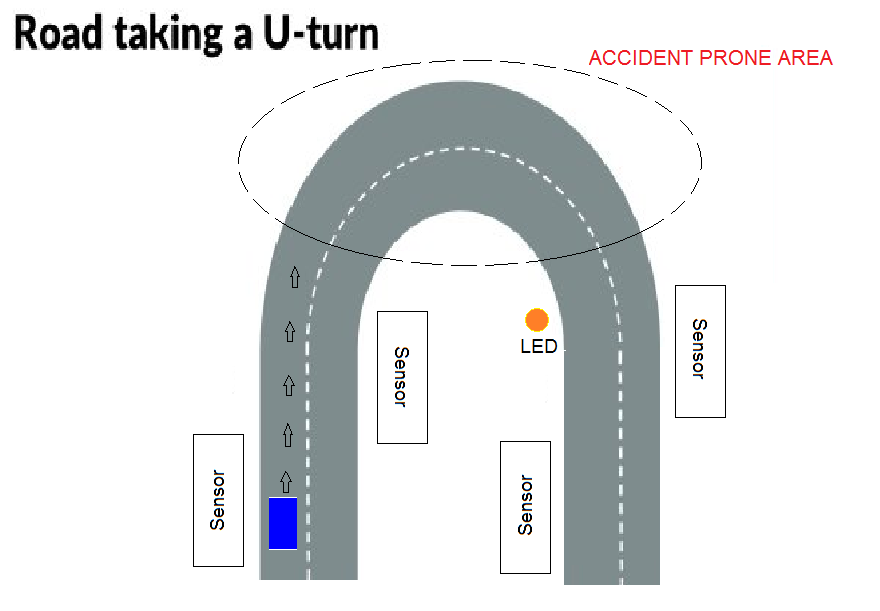
This project will work even if they are multiple objects which will come into the range of the sensors the LED will glow, this is due to we have used a count function such that it will count the no of objects which will pass through the sensor 1 and it increments the count function. When those number of objects will pass through the sensor 2(out) then only the count will be equal to zero then the LED will turn off. Here basic principle is whenever the count is greater than zero then the LED will glow otherwise it will turn off.

**4.2 PLAN**



**fig:**

**EXAMPLE 1: One Vehicle Taking a U-Turn (LED Glows).**

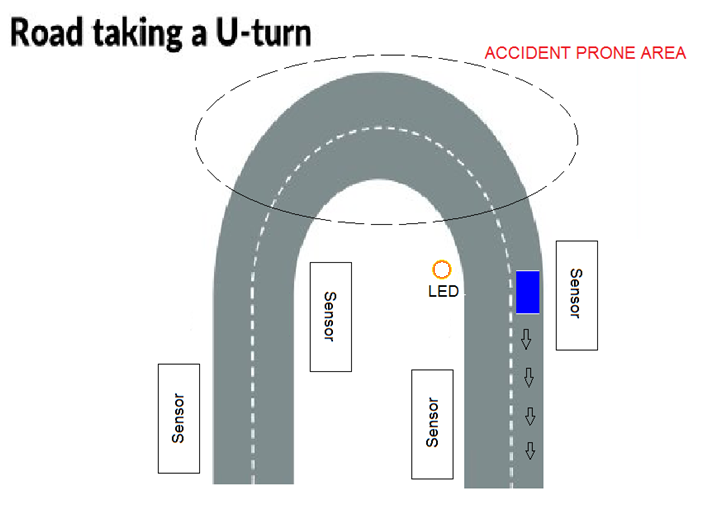


**Portability Requirements:** Some of the attributes of software that relate to the ease of porting the software to other host machines and/or operating systems. This may include:

Java is used to develop the product. So it is easiest to port the software in any

**fig:10**

**EXAMPLE 2: The Same Vehicle Leaving The U-Turn (LED OFF).**



]

**fig:11**

**fig:10**

**4.3 Building the System**

|  |  |
| --- | --- |
| **Parts** | **Quantity** |
| Breadboards | 4 |
| Sensors | 4 |
| LED lights | 8 |
| Connecting wires | Required Number |
| Aurdino | 1 |
| Batteries | 2 |
| Metal Plates | 2 |
| Power Supply | 5v |

**4.3.1 Arduino (Hardware)**

An Arduino board historically consists of an Atmel 8-, 16- or 32-bit AVR microcontroller (although since 2015 other makers' microcontrollers have been used) with complementary components that facilitate programming and incorporation into other circuits. An important aspect of the Arduino is its standard connectors, which let users connect the CPU board to a variety of interchangeable add-on modules termed *shields*. Some shields communicate with the Arduino board directly over various pins, but many shields are individually addressable via an I²C serial bus—so many shields can be stacked and used in parallel. Before 2015, Official Arduinos had used the AtmelmegaAVR series of chips, specifically the ATmega8, ATmega168, ATmega328, ATmega1280, and ATmega2560. In 2015, units by other producers were added. A handful of other processors have also been used by Arduino compatible devices. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator (or ceramic resonator in some variants), although some designs such as the Lily Pad run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions. An Arduino's microcontroller is also pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory, compared with other devices that typically need an external programmer. This makes using an Arduino more straightforward by allowing the use of an ordinary computer as the programmer. Currently, optiboot bootloader is the default bootloader installed on Arduino UNO.

[](https://en.wikipedia.org/wiki/File:UnoConnections.jpg)

**fig:12**

At a conceptual level, when using the Arduino integrated development environment, all boards are programmed over a serial connection. Its implementation varies with the hardware version. Some serial Arduino boards contain a level shifter circuit to convert between  RS-232 logic levels and transistor–transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods, when used with traditional microcontroller tools instead of the Arduino IDE, standard.AVR in-system.programming (ISP).programming.is.used.   
An official Arduino Uno Revision 2 with descriptions of the I/O locations

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The *Diecimila*, *Duemilanov*, and current *Uno* provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.

**4.4 Software Environment**

The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages Processing and *Wiring*. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and provides simple one-click mechanism to compile and load programs to an Arduino board. A program written with the IDE for Arduino is called a "sketch".

The Arduino IDE supports the languages C and C++ using special rules to organize code. The Arduino IDE supplies a software library called Wiring from the Wiring project, which provides many common input and output procedures. A typical Arduino C/C++ sketch consist of two functions that are compiled and linked with a program stub *main()* into an executable cyclic executive program:

* *setup()*: a function that runs once at the start of a program and that can initialize settings.
* *loop()*: a function called repeatedly until the board powers off.

After compiling and linking with the GNU tool chain, also included with the IDE distribution, the Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal coding that is loaded into the Arduino board by a loader program in the board's firmware.

**CHAPTER 5**

**TESTING AND VALIDATION**

**5.1 INTRODUCTION**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**5.2 TYPES OF TESTS**

**Unit Testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration Testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional Test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Testing**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**5.3 Design of Test Cases**

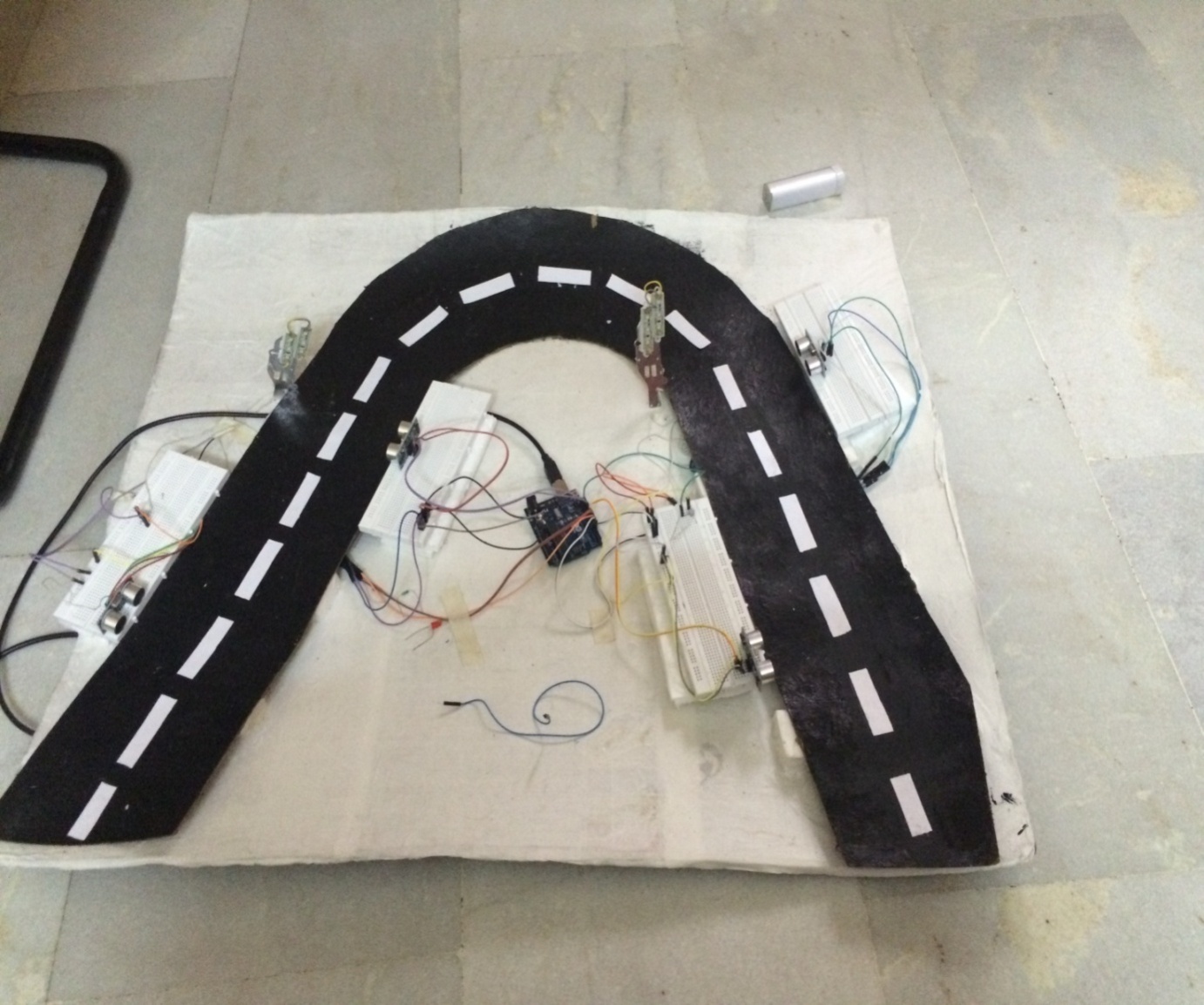
|  |  |  |
| --- | --- | --- |
| **Test Case** | **Expected Result** | **Pass/fail** |
| Passing an object in front of a sensor1 | The led should glow | pass |
| Passing an object in front of a sensor2 | The led should get off | Pass |
| Passing an object in front of a sensor1 | The count should increment | Pass |
| Passing an object in front of a sensor2 | The count should decrement | Pass |
| Place an object in front of a sensor1 | Count should increment only once | Pass |
| Place an object in front of a sensor2 | Count should increment only once | Pass |

**CHAPTER 6**

**RESULT**

**6.1 Output Screens**

**WOKING MODEL UNDER CONSTRUCTION**



**fig:13**

**WORKING MODEL AFTER CONSTRUCTION**



**fig:14**

**CHAPTER 7**

**ADVANTAGES**

**Advantages:**

|  |  |
| --- | --- |
|  | * An ultrasonic sensor’s response is not dependent upon the surface color or optical reflectivity of the object. For example, the sensing of a clear glass plate, a brown pottery plate, a white plastic plate, and a shiny aluminum plate is the same. |
|  | * Ultrasonic sensors with digital (ON/OFF) outputs have excellent repeat sensing accuracy. It is possible to ignore immediate background objects, even at long sensing distances because switching hysteresis is relatively low. |
|  | * The response of analog ultrasonic sensors is linear with distance. By interfacing the sensor to an LED display, it is possible to have a visual indication of target distance. This makes ultrasonic sensors ideal for level monitoring or linear motion monitoring applications. |

* The whole point of the "Arduino Platform" is to allow for easy and fast prototyping. Being able to just hook up an LCD and be able to display messages on it in a matter of minutes, instead of hours, is just amazingly powerful and convenient when you have an idea in your head and just want to see if it works.
* When you need more control and are actually thinking on converting your prototype into a real product, then yes, you need to get deep down into the microcontroller and get rid of all the excess fat, trim the circuit to just the bare bones,optimizethe code, etc.
* For prototyping, the Arduino platform gives you a lot of pre-wiring and free code libraries that will let you concentrate on testing your idea instead of spending your time building supporting circuitry or writing tons of low level code.
* With the help of our project we can reduce accidents.
* It is easy to install, maintain.
* Cost effective.
* Low power consumption
* Highly reliable

**CHAPTER 8**

**FUTURE SCOPE**

**Future Scope:**

* Ultrasonic is still in its infancy, it will most likely get faster and have more memory for storing data.
* The range can be considerably increased by using high power drive circuit.
* The ability of the sensor to sense objects correctly can be increased by using photoelectric sensors instead of ultrasonic sensors.
* This can also be used in the single roads which are used as two way roads.

**CHAPTER 9**

**CONCLUSION**

**Conclusion:**

* By this we conclude that by implementing our project , accidents near the steep curves especially in ghat roads, villages, forests etc can be reduced .
* This is more helpful at nights.
* This is cost effective, easy to install and maintain.

**APPENDICES**

**Appendices**

|  |  |
| --- | --- |
| **NALOG IN** | Analog input  Aurdino Uno rev 3 has 6 Analog pins (A0-A5).  Arduino power and analog pins  **fig:15**  The pins can be used as a 10 bit Analog converter (also known as 10 bit resolution).  2^10 = 1024 (0-1023) |
| 0 = 0V : 1023 = 5V |
| **breadboard** | A breadboard or protoboard is a solderless plugboard where electrical components are placed.  How the points are connected with each other see the image below:  Breadboard**fig:16** |
| **Digital pins** | Arduino Uno rev 3 has 14 digital pins (0 - 13).  Each digital pin supports 0V or 5V (max 40 mA). If the digital pin is set to HIGH, the digital pin has 5V. If the digital pin is set to LOW, the digital pin has 0V.  Pin 13 is special, it has a buildin resistor. If the led anode is connected to pin 13 and the cathode to GND, the buildin resistor protects the led (for too high current) by lowering the Voltage over the led.  Led connected to pin 13**fig:17** |
| **Drop down resistor or pull down resistor** | Usually a drop down resistor is used in combination with a switch. The drop down resistor makes sure the digital pin measures a HIGH (5V) or LOW (0V). If no drop down resistor is used the digital pin measures a noise which can be anything between 0V - 5V. The drop down resistor must have a high resistance value for example 100KΩ.  Drop down resistor**fig:18** |
| **FTDI** | Future Technology Devices International, commonly known by its abbreviation, FTDI, is a Scottish privately held semiconductor device company. The company's flagship product is its FTDI Chip.  The FTDI chip translates the serial data coming from a serial-enabled device into USB data and vice-versa. |
| **I2C** | Integrated - Integrated Circuit  I2C has 2 buses:   * SDA - Serial Data Access * SCL - Serial Clock   I2C has 1 master and one or more slaves. Master genertates clock signals shared by the slaves. |
| **I** | Current (A = Ampere)  I = U / R |
| I**/F** | Interface |
| **IMU** | Inertial Measurement Unit  Is an electronic device that measures and reports on a craft's velocity, orientation, and gravitational forces, using a combination of accelerometers and gyroscopes, sometimes also magnetometers. IMUs are typically used to maneuver aircraft, including unmanned aerial vehicles (UAVs). |
| **LDR** | Light Dependent Resistor  A photo resistor or Light Dependent Resistor (LDR) is a analog resistor whose resistance decreases with increasing incident light intensity. |
| **LED** | Light Emitting Diode  A light emitting diode (led) is a semiconductor light source and requires 2-3 V to operate. A led is always protected by a resistor in a series circuit, usually 220 Ohm. The led must be placed in the right direction (long lead connected to +)!  Led orientation**fig:19** |
| **NPN transistor** | An NPN transistor has 3 pins:   * (C)ollector * (B)ase * (E)mitter   If the base has a higher voltage than the emitter, current flows from collector to emitter. Small amount of current also flows from base to emitter. More current flows from (C) to (E) depending on the state of the (B)ase pin. If the voltage on the base increases more current will flow from (C) to (E).  NPN transistor  **fig:20** The direction of the arrow on the emitter distinguishes the NPN from the PNP transistor:   * If the arrow **P**oints i**N**, the transistor is a PNP. * If the arrow is **N**ot **P**ointing i**N**, the transistor is an NPN. * Follow the arrow to see the direction of current flow.   Always check the transistor datasheet what the collector, base and emitter leads are.  For example: The widely used transistor package TO-92 (**T**ransistor **O**utline package, Case Style **92**) has the following leads.  The BC547C (NPN) transistor is using the TO-92 package.  Transistor package TO-92  **fig:21** Usages:   * NPN transistor as electronic switch |
| **P** | Power (W = Watt)  P = U \* I |
| **parallel circuit** | Also know as current dividing circuit  The current is divided over its electrical components but the voltage is everywhere equal. |
| **PNP transistor** | A PNP transistor has 3 pins:   * (C)ollector * (B)ase * (E)mitter   If the base has a lower voltage than the emitter, current flows from emitter to collector. Small amount of current also flows from emitter to base. More current will flow from (E) to (C) depending on the state of the (B)ase pin. If the voltage on the base decreases more current will flow from (E) to (C).  PNP transistor  **fig:22** The direction of the arrow on the emitter distinguishes the NPN from the PNP transistor:   * If the arrow **P**oints i**N**, the transistor is a PNP. * If the arrow is **N**ot **P**ointing i**N**, the transistor is an NPN. * Follow the arrow to see the direction of current flow.   Always check the transistor datasheet what the collector, base and emitter leads are.  For example: The widely used transistor package TO-92 (**T**ransistor**O**utline package, Case Style **92**) has the following leads.  The B550D (PNP) transistor is using the TO-92 package.  Transistor package TO-92**fig:23** |
| **Potmeter** | A potentiometer  Is a three-terminal resistor with a sliding contact that forms an adjustable voltage divider.  Usage:   * Middle of potmeter connected to an Analog IN pin (for example A0) * One side to the 5V pin * Other side to the GND pin   By sliding the potmeter you will have 1024 (0-1023) different voltage steps.  2^10 = 1024 (0-1023)  0 = 0V : 1023 = 5V |
| **Power pins** | Arduino Uno rev 3 has 5 power pins (3.3V, 5V, GND, GND, Vin).  The power pin 5V can supply a current of max 500 mA. If an external power source is connected to the power jack the power pin Vin can supply a current of max 1000mA.  Arduino power and analog pins**fig:24** |
| **R** | Resistance (Ω = Ohm)  R = U / I |
| **RS-232** | An RS-232 serial port is used to connect modems, printers and other peripheral devices to computers. |
| **series circuit** | Also know as voltage dividing circuit  The voltage is divided over its electrical components but the current is everywhere equal. |
| **Servomotor** | A servomotor is a rotary actuator that allows for precise control of angular position, velocity and acceleration. It consists of a motor coupled to a sensor for position feedback.  Wiring:   * Orange - digital channel * Red - 5V+ * Black - GND |
| **transistor** | A transistor can be seen as a elecronic switch. See NPN transistor.  A transistor has 3 pins:   * (C)ollector * (B)ase * (E)mitter   In front of the base is always a resistor to protect it from high current. |
| **SPI** | Serial Peripheral Interface   * MOSI - Master Out Slave In * MISO - Master In Slave Out * CS - Chip Select Line |
| **U** | Voltage (V)  P = U \* I |
| **UART** | Universal Asynchronous Receiver/Transmitter  The Universal Asynchronous Receiver/Transmitter (UART) takes bytes of data and transmits the individual bits in a sequential fashion (serial communication). |
| **USB** | Universal Serial Bus  An USB has 4 pins:   * Rx - Receiver * Tx - Transaction * Vin - Voltage (0 - 5 V) * Gnd - Ground |
| **TTL** | Transister-Transistor Logic  Serial I/F (RS-232) works with (logic 1=)-15V (logic 0=) +15V Arduino uses 0-5V TTL |
| **Vin** | Power Pin max 1000mA |
| **Voltage divider** | Uout = R2 / (R1 + R2) \* Uin |
| **wiring color codes** | * VDD - red It is connected to VCC * GND - black Ground is black * Reset - blue It "freezes" the processor * (U)SCK - orange Think of the movie clockwork orange * MOSI - green Use green / yellow for communications (usually RS232) transmit / receive * MISO - yellow see above |

**REFERENCES**

**References:**

1. HC-SR04 (Ultra-Sonic Sensor) Datasheet: Link-(<https://docs.google.com/document/d/1Y-yZnNhMYy7rwhAgyL_pfa39RsB-x2qR4vP8saG73rE/edit?pref=2&pli=1> ).
2. Working Principle for HC-SR04 Link-(<http://www.micropik.com/PDF/HCSR04.pdf>).
3. Aurdino R3 Overview and Documentation Link-( Aurdino R3 Overview and Documentation Link-( <https://www.arduino.cc/en/main/arduinoBoardUno> ).
4. Brain W.Evans, “Aurdino Programming Notebook”, First Edition August 2007.