A MINI-PROJECT REPORT ON

"ULTRASONIC RANGE FINDER USING ARDUINO"

Submitted in the partial fulfilment in requirement in

"MECHANICAL MEASUREMENTS AND CONTROLS"

SUBMITTED BY

1. BHAVESH ASHOK PATIL.	**(184)**
2. VIJAY SAMPATRAO SHINDE.	**(181)**
3. JAIMEEN HASMUKH PANCHAL.	**(178)**
4. NINAD GAIKWAD.	**(172)**
5. SHERYAS PAWAR.	**( )**

THIRD YEAR ENGINEERING (SEM V)

"MECHANICAL ENGINEERING"

UNIVERSITY OF MUMBAI

GUIDED BY

Prof. NITIN NANDESHWAR
(DEPARTMENT OF MECHANICAL ENGINEERING)

SARASWATI
EDUCATION SOCIETY
Saraswati Education Society's
Yadavrao Tasgaonkar College
of Engineering

YADAVRAO TASGAONKAR COLLEGE OF ENGINEERING AND MANAGEMENT

Academic Year 2015-2016

CERTIFICATE

This is to certify that this report entitled

"ULTRASONIC RANGE FINDER USING ARDUIO"

Submitted by the following students of

"Mechanical Engineering"

Department in the Academic Year 2015-2016 towards the partial fulfilment in the requirements in

"MECHANICAL MEASUREMENTS AND CONTROLS".

- ❖ **BHAVESH ASHOK PATIL**
- ❖ **VIJAY SAMPATRAO SHINDE.**
- ❖ **JAIMEEN HASMUKH PANCHAL.**
- ❖ **NINAD GAIKWAD.**
- ❖ **SHERYAS PAWAR.**

Throughout their tenure of completion of task they have been guided and assessed by me, I am satisfied that their contribution was proportionate, they were satisfactory progressive and their task is up to standard envisaged by University of Mumbai.

HOD MECHANICAL **Project Guide**
(Prof. R.K.AGRAWAL) (Prof. NITIN NANDESHWAR)

DATE: - ____________

TABLE OF CONTENT

SR. NO.	TITLE	PAGE NO.
1.	ACKNOWLEDGEMENT.	
2.	ABSTRACT.	
3.	CHAPTER 1-INTRODUCTION.	
4.	CHAPTER 2-LITERATURE REVIEW.	
5.	CHAPTER 3-PROPOSED SYSTEM.	
6.	CHAPTER 4-REQUIREMENT AND DESIGNS.	
7.	CHAPTER 5-CONCLUSION.	
8.	CHAPTER 6-EFERENCE.	

ACKNOWLEDGEMENT

"We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and web research. We would like to extend my sincere thanks to all of them.

We are highly indebted to **Prof. NITIN NANDESHWAR** for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in completing the project.

We would like to express our gratitude towards my parents & member of **YADAVRAO TASGONKAR COLLEGE OF ENGINEERING AND MANAGEMENT** for their kind co-operation and encouragement which help me in completion of this project.

Ours thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities."

ABSTRACT

The Arduino microcontroller is an easy to use yet powerful single board computer that has gained considerable traction in the hobby and professional market. The Arduino is open-source. which means hardware is reasonably priced and development software is free.

Monitoring Environmental factors such as temperature measurement, light intensity. relative humidity and air composition are not only important for industrial purposes but also a cause of growing concern in our day to day life.

“Arduino can sense the environment by receiving input from a variety 'of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language“.

Arduino acts as an interface between the sensors and LCD screen. This project is successfully implemented in hardware and works excellently. The temperature readings are precise and light intensity in lumens is accurate with a negligible error of 10%. The system is cost effective as it uses very cheap sensors, is easy to make, highly portable and compact. This is highly beneficial for low cost industrial applications, travelling situations, outdoor conditions. basic military purposes and household applications.

I Ultrasonic sensor is distance measurement sensor which use ultrasonic

sound waves to measure distance. An ultrasonic sensor use high frequency. sound wave of 40 KHz. Ultrasonic sensors consists of two basic modules transmitter and receiver. Transmitter acts as speaker and receiver acts as a microphone. Speaker emits ultrasonic waves and Microphone detects ultrasonic waves which are produced by speaker.

INTRODUCTION

CHAPTER NO.01

A Tmega48A/PA/88A/PA/l 68A/PA/328/P

ATMEL 841m MICROCONTROLLER with 4/8/Io/32KBYTES

IN-SYSTEM PROGRAMMABLE FLASH

DATASHEET SUMMARY

Features:

- High Performance, Low Power Atmelt®AVR® 8-Bit Microcontroller Family
- Advanced RISC Architecture
- 131 Powerful Instructions Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Fully Static Operation
- Up to 20 MIPS throughput at 20Mllz
- On-chip 2-eyele Multiplier
- High Endurance Non-volatile Memory Segments
- 4/8/16/32Kbytes In-System Sell Programmable Flash program memory
- 256/512/512/1KBytes EEPROM
- 512/1K/1K/2KBytes Internal SRAM
- Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
- Data retention: 20 years at 85 □C/IOO years at 25 □C(1)
- Optional Boot Code Section with Independent Lock Bits
- In-System Programming by On-chip Boot Program
- True Read-While-Write Operation
- Programming Lock for Software Security

- Atmel® Touch® library support
- Capacitive touch buttons, sliders and wheels
- Touch and Matrix® acquisition
- Up to 64 sense channels 3 ‘

Peripheral Features:

• Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
• One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture
Mode
• Real Time Counter with Separate Oscillator 'Six PWM Channels '
• 8channel 10bit ADC 1n TQFP and QFN/Ml F package
• Temperature Measur-ement-6 channel 10bit ADC In PDIP Package
.

V ATme1a48A/PA/88A/PA/l68A/PA/328/P
ATMEL 8-BIT MICROCONTROLLER WITH
4/8/16/32KBYTES
IN-SYSTEM PROGRAMMABLE FLASH

SUMMARY :-

- Tmega48A/PA/88A/PA/l 68A/PA/328/P [DATA SHEET] 2
- Atmel~827llS-AVRATmega-Datashcct__10/20l4 0

Special Microcontroller Features:

1. Power-on Reset and Programmable Brown-out Detection
2. Internal Calibrated Oscillator
3. External and Internal Interrupt Sources
4. Six Sleep Modes: Idle, ADC Noise Reduction, Power-save. Power' down, Standby, an Extended Standby.

- I/O and Packages:

  23 Programmable I/O Lines

  11. 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad

. • QFN/MLF

- Operating Voltage::1.8 -5.5V
- Temperature Range: -40DC to 85 DC
- Speed GradeiO4MHz@18 5.5V, 0 10MH2@2.-7 5-5v 0 20MHz @4 5 5.5V
- Power Consumption at 1MHZ, 1. 8V, 25 DC

  1. Active Mode: 0.2mA ,
  2. Power-down Mode: 0.1 uA
  3. Power-save Mode: 0.75 uA (Including 321<Hz RTC)

Figure 5-4. 32-pin MLF Top View

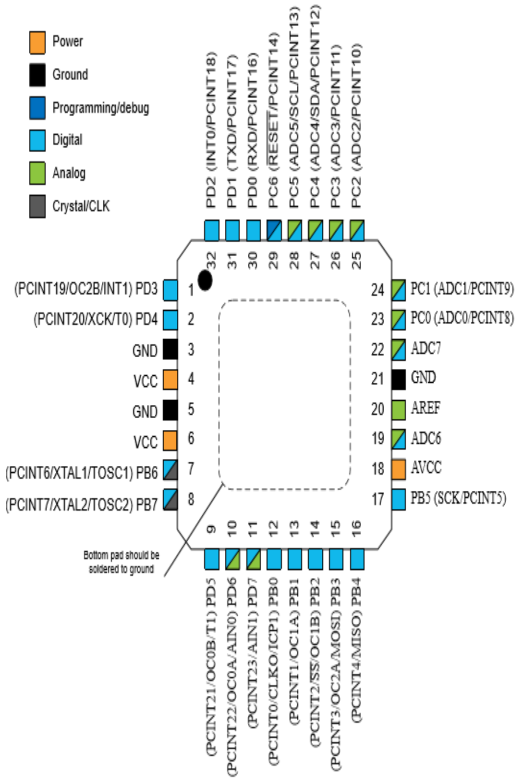
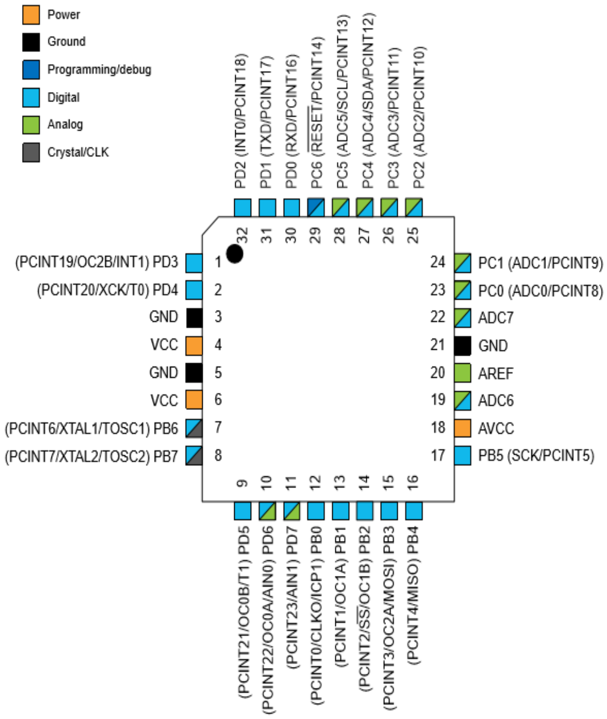


Figure 5-3. 32-pin TQFP Top View



1.1 Pin Descriptions:

1.1.1 VCC Digital supply voltage.

1.1.2 GND
Ground.

1.1.2 Port B (PB7:0) XTALI/XTALZ/TOSCI/TOSCZ

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pin that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tristated when a reset condition becomes active, even if the clock is not running Depending on the clock selection fuse settings, P86 can be used as input to the inverting Oscillator amplifier and input to the internal clock Operating circuit. Depending on the clock selection use settings, PB7 can be used as output from the inverting Oscillator Amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSCZ...1 input for the asynchronous Timer/Counter2 if the A32 bit in ASSR is set. The various special features of Port B are elaborated.

1.1.4 Port C(PC5:0)

Port C is a 7-bit bi-directional l/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull -up resistors are activated.

The Port C pins are tristated when a reset condition becomes active even if the clock IS not running.

1.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6differ from those of the other pins of Port C

If the RSTDISBL Fuse is programmed, l'C6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even it the clock is not running. The minimum pulse length is given in. Shorter pulses are not guaranteed to generate a Reset.

1.1.6 Port D (PD7:0)

Port D is an 8-bit iii-directional l/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs. P011 D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tristated when a reset condition becomes active, even if the clock is not running.

1.1.7 AVCC

AVCC is the supply voltage pin for the A/D Converter. PC320 and ADC7z6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter.

Note that PC6...4 use digital supply voltage. VCC.

1.1.8 AREF

AREF is the analog reference pin for the A/D Converter.

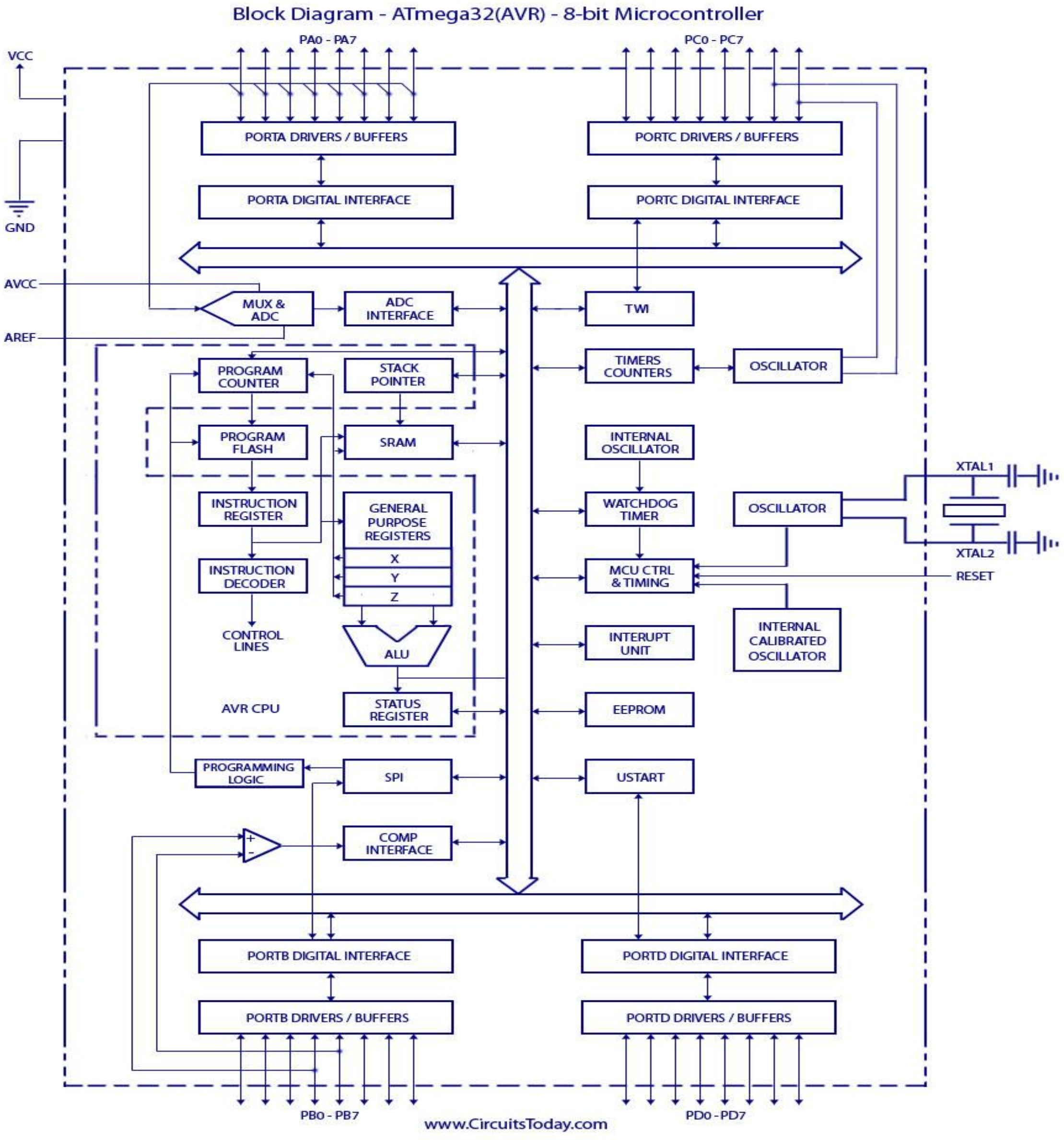
1.1.9 ADC7z6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 104m
ADC channels.

Overview :-

The atmcga48a/PA/88A/PA/l68A/PA/328/P is a low-power CMOS 8-bit microcontroller based on (he AVR enhanced RISC architecture By executing powerful instructions in a single clock cycle, the A atmcga48a/PA/88A/PA/l68A/PA/328/P achieves through approaching l MIPS per mhzallowing the system designer to optimize power consumption versus processing speed

Block Diagram :-

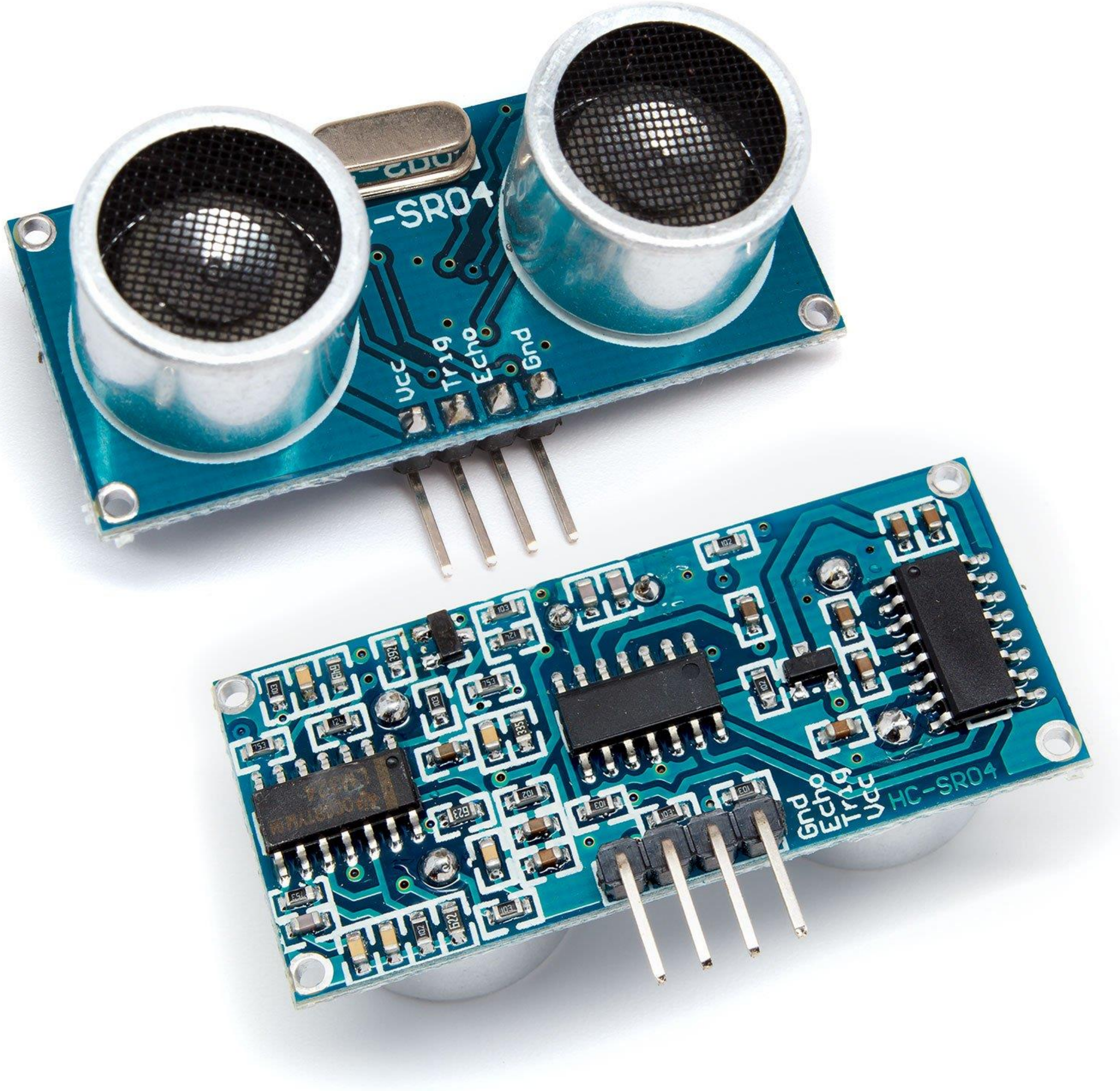


All the 32 registers are directly connected to the Arithmetic Logic Unit (AIJU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The atmcga48a/PA/88A/PA/l68A/PA/328/P provides the following features: 4K/8Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 256/5l2/5I2/kbytes EEPROM, SI2/IK/1K/2KbytcsSRAM, 23 general purpose [/0 lines, 32 general purpose working ‘registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channcl IO-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous ”timer continues to run, allowing the user to maintain a timer base while the, rest of the device is sleeping. The ADC Noise Reduction mode stops' the CPU ‘and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions: In Standby mode, the crystal/resonator Oscillator is running while the :rest of the device is sleeping. This allows very fast start-up combined with “low power consumption. Atmel® offers the qtouch® library for embedding capacitive touch buttons, sliders and wheels functionality into AVR® microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key

Suppression® (AKSTM) technology for unambiguous detection of key events. The easy-to-use touch Suite tool chain allows you to explore, develop and debug your own, touch applications.

ULTRASONIC RANGE FINDER



CHAPTER No.02

INTRODUCTION

Ultrasonic sensor is distance measurement sensor which use ultrasonic sound waves to measure distance. An ultrasonic sensor use high frequency sound wave of 40 KHz. Ultrasonic sensor consists of two basic modules transmitter and receiver. Transmitter acts as speaker and receiver acts as a microphone. Speaker emits ultrasonic waves and Microphone detects ultrasonic waves which are produced. by speaker. Basic functionality of ultrasonic sensor is shown in diagram below:

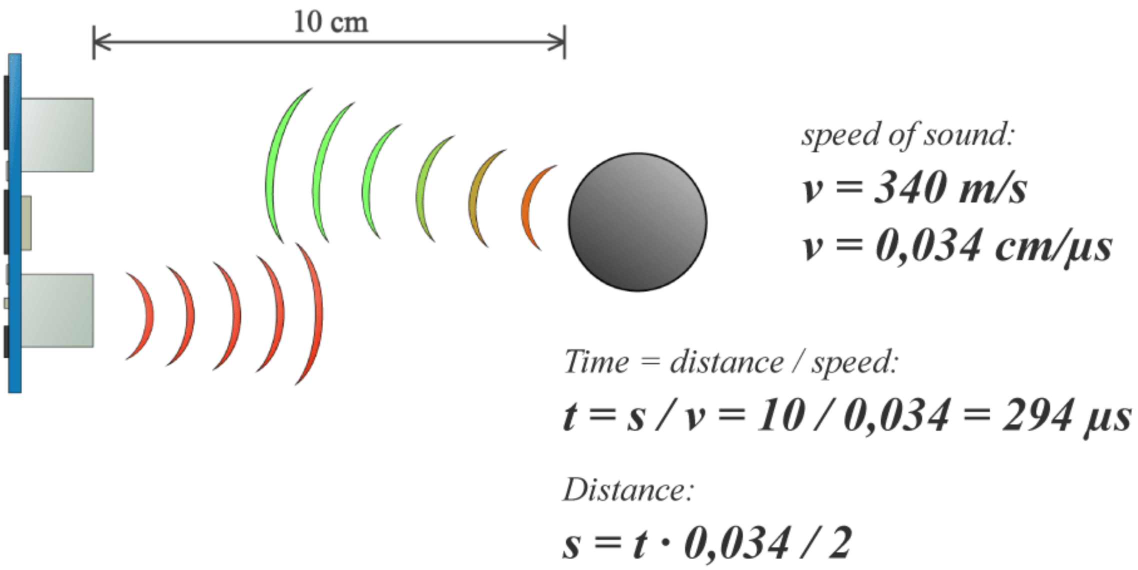


Fig shows ultrasonic sensor working

As shown in above figure one part is acting as speakers which emit ultrasonic waves which after collision with any object return ' back. These returning waves are detected by Microphone. The time taken by microphone to receive ultrasonic waves from transmitter after collision with any object is used to measure distance. I will' discuss it later how this time is used to measure distance from any object Ultrasonic sensor works on the same principle as radio based

Description

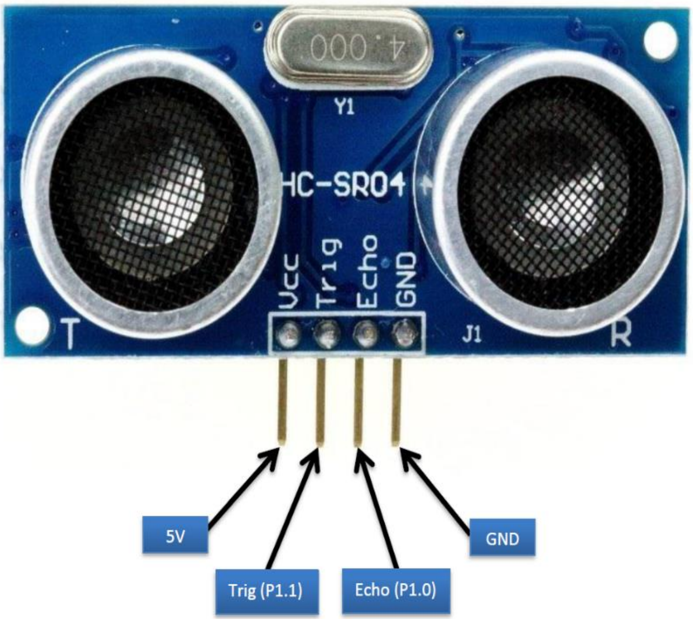
The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object like bats do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. From 2cm to 400 cm or 1” to 13 feet. It operation is not affected by sunlight or black material like Sharp rangefinders are (although acoustically soft materials like cloth can be difficult to detect). It comes complete with ultrasonic transmitter and receiver module.

Features

- Power Supply :+5V DC
- Quiescent Current : <2mA
- Working Current: 15mA
- Effectual Angle: <15°
- Ranging Distance : 2cm – 400 cm/1″ – 13ft
- Resolution : 0.3 cm
- Measuring Angle: 30 degree
- Trigger Input Pulse width: 10uS
- Dimension: 45mm x 20mm x 15mm

Pins

- VCC: +5VDC
- Trig : Trigger (INPUT)
- Echo: Echo (OUTPUT)
- GND: GND

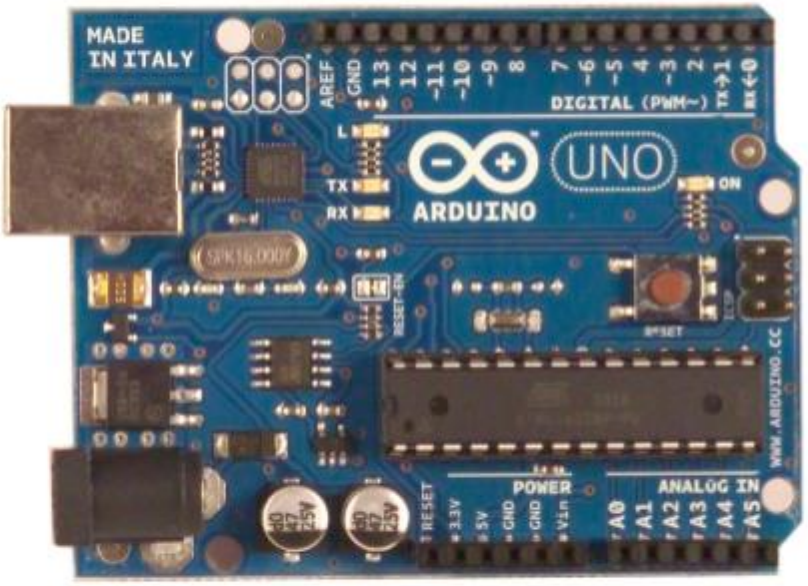


APPLICATIONS OF ULTRASONIC SENSOR :-

There are many application of ultrasonic sensor from domestic .use industrial use. But some of them are given below:

l. Obstacle avoidance robot
2. Robotics
3. object detection
4. distance measurement
5. liquid level monitoring system
6. Height measurement .
7. Agriculture
8. vehicle collision protection.

Arduino Uno



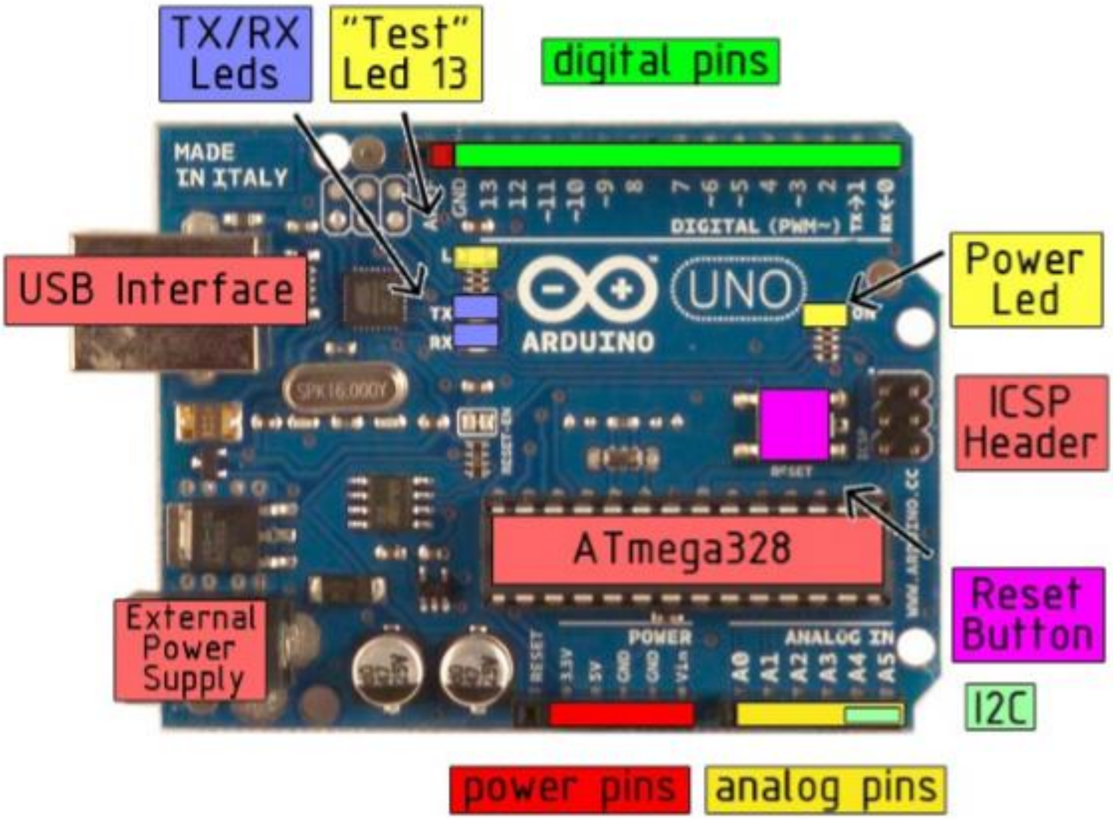
Product Overview :-

The Arduino Uno is a microcontroller board based on the ATmega328 . It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduno, moving forward. The Uno is the latest in a series of USB Arduino board

Technical specification

Microcontroller ATmega328
Operating Voltage 5V
Input Voltage (recommended) 7-12V
Input Voltage (limits) 6-20V
Digital I/O Pins 14 (of which 6 provide PWM output)
Analog Input Pins 6 DC Current per I/O Pin 40 mA
DC Current for 3.3V Pin 50 mA
Flash Memory 32 KB of which 0.5 KB used by bootloader
SRAM 2 KB
EEPROM 1 KB
Clock Speed 16 MHz



The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

• VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. • 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply. • 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA. • GND. Ground pins

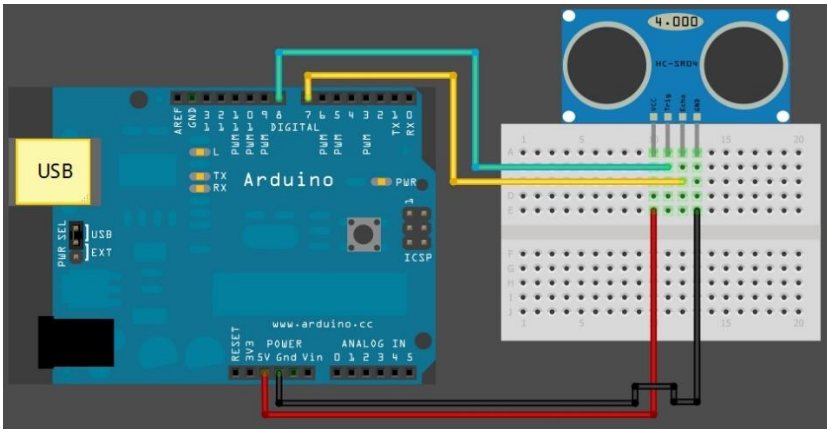
CHAPTER NO O3

PROPOSED SYSTEM

ULTRASONIC RAGE FINDER USING ULTRASONIC MODULE AND ARDUINO UNO:

As l have already discussed above, transmitter part of ultrasonic module emits ultrasonic high frequency waves in the form of pulses. After collision of these waves with any object, these waves detect by microphone. Time taken by these waves from transmitter to receiver is used to measure distance from any object. ’l‘here are many ultrasonic sensor modules are available in market but I will be using HC-SR04.

HC-SR04 ultrasonic range finder module is used as the sensor here. The display consists of a three digit multiplexed seven segment display. This range finder can measure up to 200 cm and has an accuracy of 1cm. There is an option for displaying the distance in inch also. Typical applications of this range finder are parking sensors, obstacle warning system, level controllers, terrain monitoring devices etc. Lets have a look at the HC-SR04 ultrasonic module first.



Working :-

1. Power up the sensor by SVDC using pins “VCC” and “GND”.

2.First of all a trigger input has to be given to the pin named “Trig” on the sensor

3. Starts one cycle of range conversion and sends 8 bursts of sound waves from the transmitter.

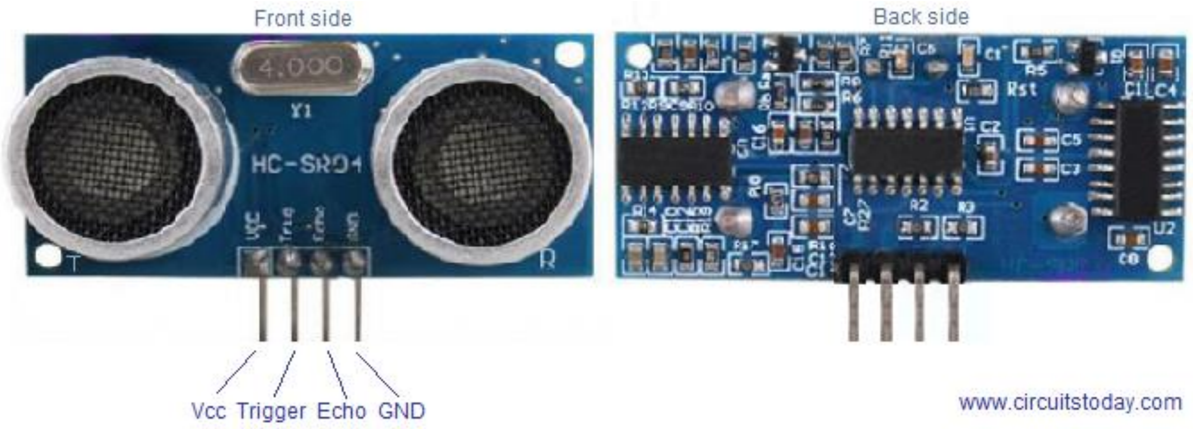
4. As soon as the signals are transmitted the “Echo” pin goes to high level and remains in high

5. Level until the same sound waves is received by the receiver. If the received sound waves are

6. Same as what the same sensor transmitted then the Echo pin goes to low level,

7.If no object is detected within 5M after 30ms the Echo signal will automatically go to low level.

8. Conversion is over and echo pin has been pulled to low by the sensor.



- **Vcc:** 5V supply voltage is given to this pin.
- **Trigger:** A 10uS long pulse is given to this pin for triggering the transmission. Upon receiving a valid trigger pulse, the HR-SR04 issues eight 40KHz pulses. Time taken by these pulses to reflect back is measured and the distance is calculated from it.
- **Echo:** At this pin the HC-SR04 outputs a signal whose high time is proportional to the range.
- **Ground :** Ground is connected to this pin.

Program :-

```
#include <NewPing.h>
#define trig 0
```

```
#define echo 13
#define maximum 200
int a;
int unit;
int usec;
int input=4;
int disp1=1;
int disp2=2;
int disp3=3;
int segA=5;
int segB=6;
int segC=7;
int segD=8;
int segE=9;
int segF=10;
int segG=11;
int segDP=12;
NewPing sonar(trig, echo, maximum);
void setup()
{
  pinMode(disp1, OUTPUT);
  pinMode(disp2, OUTPUT);
  pinMode(disp3, OUTPUT);
  pinMode(segA, OUTPUT);
  pinMode(segB, OUTPUT);
  pinMode(segC, OUTPUT);
  pinMode(segD, OUTPUT);
  pinMode(segE, OUTPUT);
  pinMode(segF, OUTPUT);
  pinMode(segG, OUTPUT);
  pinMode(segDP, OUTPUT);
  pinMode(input, INPUT);
}
void loop()
{
 delay(20);
```

```
usec=sonar.ping(); //sends ping and estimates the duration of echo in uS
unit= digitalRead(input); //reads the status of cm/inch selector switch
if(unit==1)
{
usec=usec/58; // distance in cm
}
else
{
  usec=usec/148; // distance in inch
}
 a=usec%10;
 digitalWrite(disp1,LOW);
 digitalWrite(disp2,LOW);
 digitalWrite(disp3, HIGH);
 digitalWrite(segDP,HIGH);
 display(a);
 delay(4);
 usec = usec/10;
 a = usec%10;
 digitalWrite(disp3,LOW);
 digitalWrite(disp2,HIGH);
 digitalWrite(segDP,HIGH);
 display(a);
 delay(4);
 usec=usec/10;
 a=usec;
 digitalWrite(disp2,LOW);
 digitalWrite(disp1,HIGH);
 digitalWrite(segDP,HIGH);
 display(a);
 delay(4);
 }
 int display (int a)
 {
 switch (a)
 {
```

```
case 0:
digitalWrite(segA, LOW);
digitalWrite(segB, LOW);
digitalWrite(segC, LOW);
digitalWrite(segD, LOW);
digitalWrite(segE, LOW);
digitalWrite(segF, LOW);
digitalWrite(segG, HIGH);
 break;

case 1:
digitalWrite(segA, HIGH);
digitalWrite(segB, LOW);
digitalWrite(segC, LOW);
digitalWrite(segD, HIGH);
digitalWrite(segE, HIGH);
digitalWrite(segF, HIGH);
digitalWrite(segG, HIGH);
break;

case 2:
digitalWrite(segA, LOW);
digitalWrite(segB, LOW);
digitalWrite(segC, HIGH);
digitalWrite(segD, LOW);
digitalWrite(segE, LOW);
digitalWrite(segF, HIGH);
digitalWrite(segG, LOW);
break;

case 3:
digitalWrite(segA, LOW);
digitalWrite(segB, LOW);
digitalWrite(segC, LOW);
digitalWrite(segD, LOW);
digitalWrite(segE, HIGH);
digitalWrite(segF, HIGH);
digitalWrite(segG, LOW);
break;

case 4:
digitalWrite(segA, HIGH);
digitalWrite(segB, LOW);
digitalWrite(segC, LOW);
digitalWrite(segD, HIGH);
digitalWrite(segE, HIGH);
digitalWrite(segF, LOW);
digitalWrite(segG, LOW);
break;

case 5:
digitalWrite(segA, LOW);
digitalWrite(segB, HIGH);
digitalWrite(segC, LOW);
digitalWrite(segD, LOW);
digitalWrite(segE, HIGH);
digitalWrite(segF, LOW);
digitalWrite(segG, LOW);
break;

case 6:
digitalWrite(segA, LOW);
digitalWrite(segB, HIGH);
digitalWrite(segC, LOW);
digitalWrite(segD, LOW);
digitalWrite(segE, LOW);
digitalWrite(segF, LOW);
digitalWrite(segG, LOW);
break;

case 7:
digitalWrite(segA, LOW);
digitalWrite(segB, LOW);
digitalWrite(segC, LOW);
```

```
  digitalWrite(segD, HIGH);
  digitalWrite(segE, HIGH);
  digitalWrite(segF, HIGH);
  digitalWrite(segG, HIGH);
  break;

  case 8:
  digitalWrite(segA, LOW);
  digitalWrite(segB, LOW);
  digitalWrite(segC, LOW);
  digitalWrite(segD, LOW);
  digitalWrite(segE, LOW);
  digitalWrite(segF, LOW);
  digitalWrite(segG, LOW);
   break;

  case 9:
  digitalWrite(segA, LOW);
  digitalWrite(segB, LOW);
  digitalWrite(segC, LOW);
  digitalWrite(segD, LOW);
  digitalWrite(segE, HIGH);
  digitalWrite(segF, LOW);
  digitalWrite(segG, LOW);
  break;
}}
```

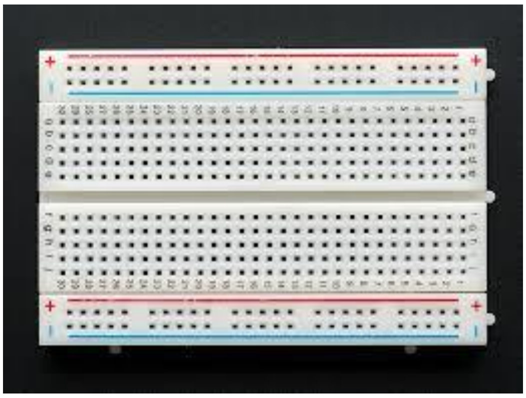
CHAPTER NO O4

Requirment & Design

Requirment :-

The required component for this project are listed below .

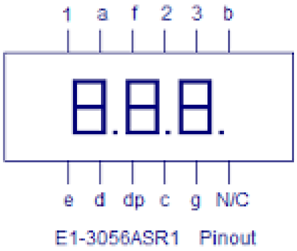
1)Bread board :



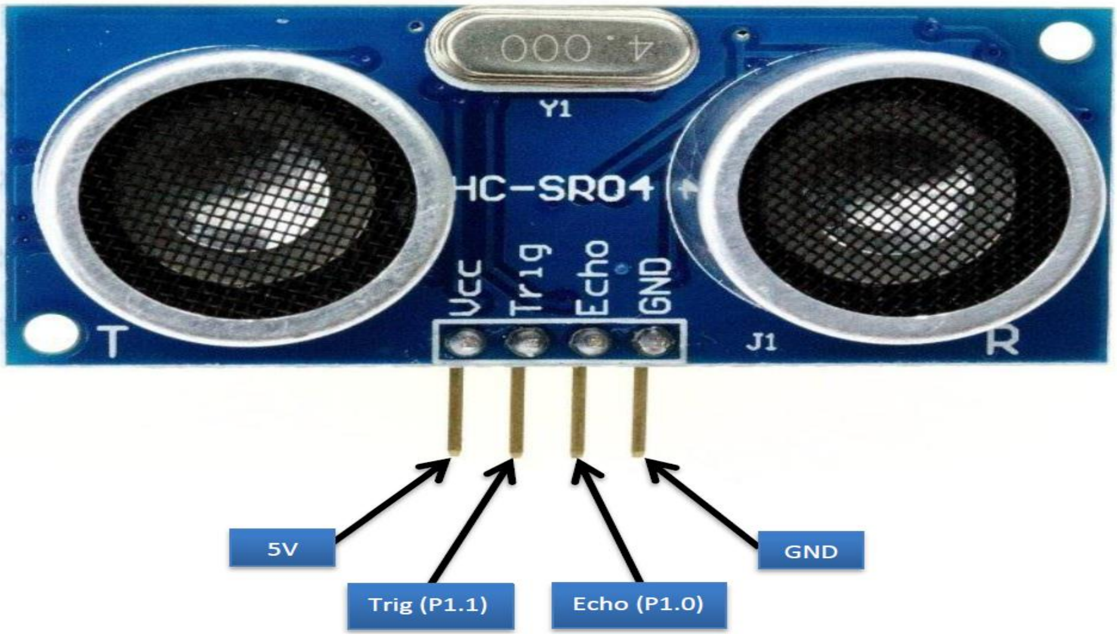
2)Arduio Uno R3 Development Board :

3. Jumper Wire :

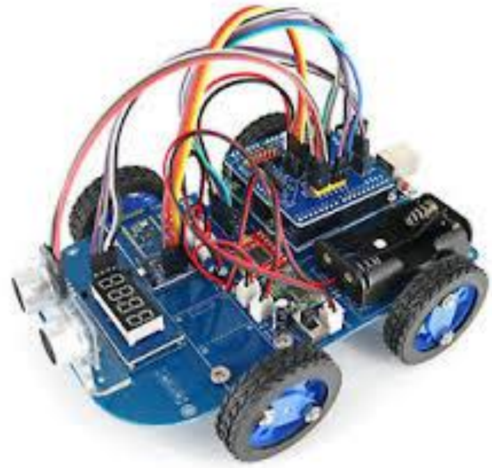
4. pinout E1-3056ASR1



5. Ultrasonic Sensor (HC SR04):

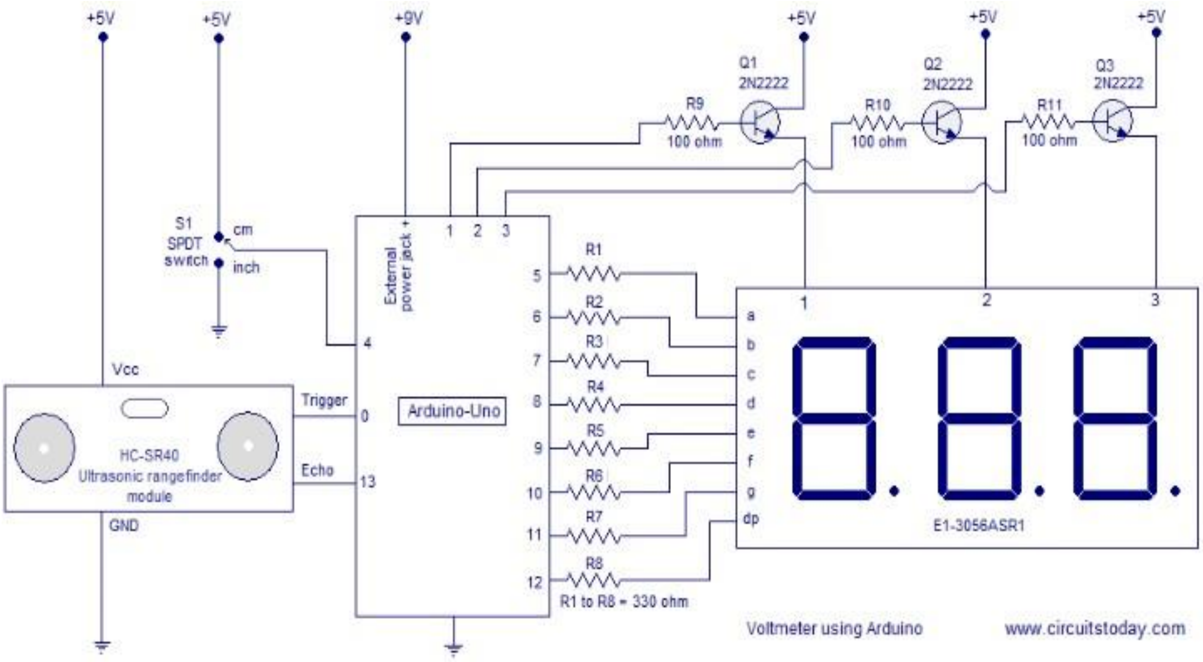


6. Gear Motor & Chassis :-



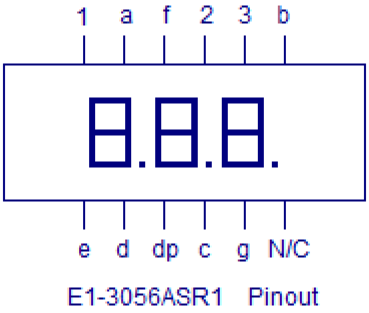
Connection :-

Circuit diagram

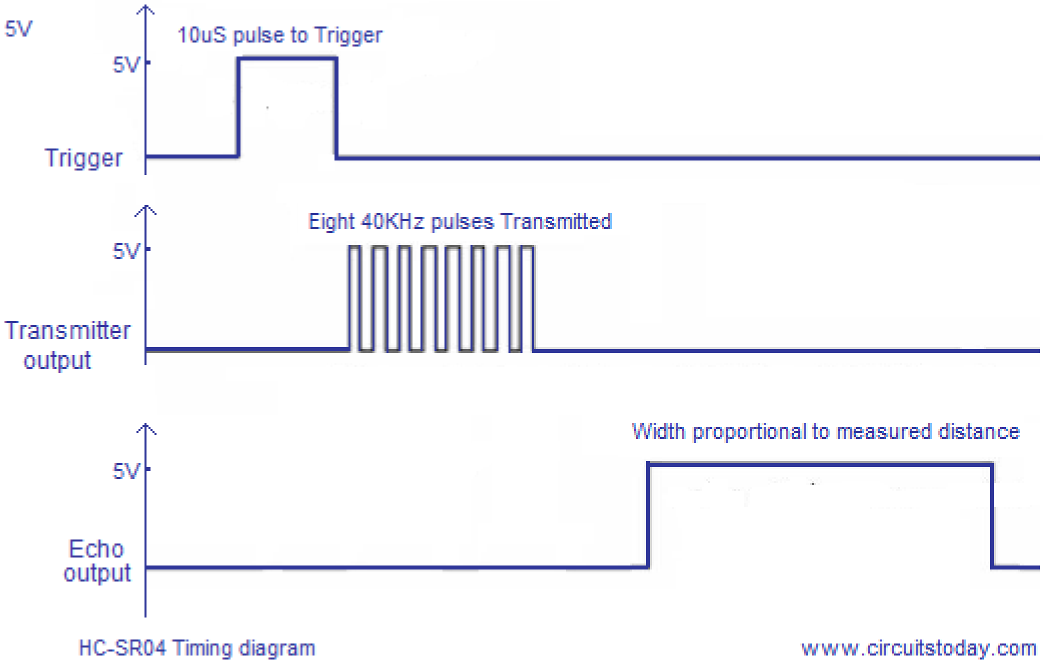


Voltmeter using Arduino www.circuitstoday.com

Trigger pin of the ultrasonic range finder module is connected to digital pin 0 of the arduino. Echo pin of the ultrasonic module is connected to the digital pin 13 of the arduino. SPDT switch S1 is used to select the unit of the measurement shown in the display. Pole of the SPDT switch S1 is connected to digital pin 4 of the arduino. If digital pin 4 is held high, the output will be in centimeters and if the digital pin 4 is held low, the output will be in inches. Digit driver transistor Q1, Q2 and Q3 of the arduino are interfaced to digital pins 1, 2 and 3 of the arduino. Multiplexed segments a to dp are interfaced to digital pins 5 to 12 of the arduino. The arduino board can be powered through the +9V jack given on the board. 5V supply needed in some other parts of the circuit can be obtained from the 5V source available in the arduino board. Resistors R9, R10 and R11 limits the base current of the corresponding transistors. 330 ohm resistors R1 to R8 limits the current through the corresponding segments. Pin out of an E1-3056ASR1 three digit MUX seven segment display in shown in the figure below.



The timing diagram of HC-SR04 is shown in the figure below.



CHAPTER No.05

CONCLUSION

A Simplified environmental monitoring system with least number of components, and less complexity has been constructed. This system a is compact and. highly cost effective When compared to prices of instruments used to measure environmental factors.
Arduino Uno has mostly all the capabilities inbuilt and requires less hardware for its operation. So, this makes it a perfect choice for our upgraded system .’ This project work is successfully implemented and the results obtained are accurate with a simple user interface.
Since, Ultrasonic sensor is ultrasonic wave finder sensor which use ultrasonic sound waves to measure distance. An ultrasonic sensor use high frequency sound wave of 40 kllz.

REFERENCE

1. www.wikipidia.org
2. http://www.circuitstoday.com