PROJECT

ON

ARDUINO & ITS APPLICATIONS AND IOT

SUBMITTED TO

DEPARTMENT OF PHYSICS

BY

D.REENI FRANCINA	1212-20-509-002
S.HARINI MEENAKSHI	1212-20-509-008
ASTHA SURYAPANI	1212-20-509-015
SADIYA SULTANA	1212-20-509-013

Project guide

MS.PALEENA THULLIMILLI

FACULTY, DEPARTMENT OF PHYSICS

ST.PIOUS X DEGREE AND PG COLLEGE FOR WOMEN,

NACHARAM, HYDERABAD-500076

ST.PIOUS X DEGREE AND PG COLLEGE FOR WOMEN

Accredited with A grade by NAAC, Affiliated to Osmania University,HYD

Snehapuri colony, Nacharam, Hyderabad

DEPARTMENT OF PHYSICS

BONAFIDE CERTIFICATE

This is to certify that the project on "Arduino Ultrasonic Sensor and its applications" being submitted by D.Reeni Francina (121220509002), S.Harini Meenakshi (121220509008), Astha Suryapani (121220509015), Sadiya Sultana (121220509013) in partial fulfilment of the requirements for PROJECT, during the academic year 2020-2022 and submitted to the Department of Physics, St. Pious X Degree & P.G College, affiliated to Osmania University.

MRS.PALEENA THULIMILLI
PROJECT SUPERVISOR

DR.R.KOMALA
H.O.D PHYSICS

SR.B.VELANGANI KUMARI
PRINCIPAL

Acknowledgements

It gives us an immense pleasure and pride to express our gratitude to REV.SR.B.Velangini Kumari. The Principal and Dr.R.Komala, Head Department of physics, St. Pious X Degree & PG College for Women, Nacharam for giving an opportunity to do this wonderful project and providing facilities to carry out the Project in the College.

We are delighted to thank with pleasure our project supervisor Mrs.PaleenaThlimilli, for her guidance, support, and encouragement throughout our project and also for her timely decisions, wise counsel, and unfailing attentions.

We acknowledge the cheerful assistance rendered by our classmates, who also helped us in doing a lot of Research and we learnt about so many new things, really thankful to them.

We would like to thank our parents who helped us a lot in finalizing this Project within the limited time frame.

Signature

INDEX

CONTENT	PAGE NO
1.1 Abstract	1
1.2 Introduction	2
1.3 Motivation of the project	3
1.4 Scientific basis	4
1.5 Methodology	4
1.6 Outcome of the project	5
2.1 Description of Arduino Ultrasonic Sensor	6
2.2 What is Arduino?	6
2.3 Hardware description of Arduino	7
2.4 Types of Sensors	9
3.1 Ultrasonic Sensor	13
3.2 Ultrasonic Ranging Module HC-SR04	13
3.3 Timing Diagram	15
3.4 ATmega32	16
3.5 Timer	19
3.6 LCD Display	20
3.7 HC-SR04 Ultrasonic Sensor Pinout	22
4.1 Working of Ultrasonic Sensor	23
4.2 Interfacing of Ultrasonic Sensor with Arduino	27
4.3 Code Explanation	29
4.4 Output	32

INDEX

CONTENT	PAGE NO
5. Innovations using Ultrasonic Sensor	33
6. Result	33
7. Applications	34
8. Limitations	34
9. References	36

ABSTRACT

When the electronic devices are communicating with each other without the human interference that is called an embedded system. In recent times, the embedded systems have been replaced with various other complex electronic circuits. Usually, the heart of the embedded system is the microcontroller. One of the good example of microcontroller is Arduino. Arduino is an open source electronic platform used to control the physical devices with the help of hardware and software, as it involves both. In short, Arduino is a tool for controlling electronics. As sensors have the capacity of detecting, they are used as inputs for every system.

There are three main sources to deal with Arduino. The first one is Arduino hardware, Arduino Integrated Development System(IDE) and Arduino code. So, to connect to the Arduino board we need a USB cable then, we have to write the source code. Arduino is used in automation systems, to design basic electronic circuits. There are different types of sensors used in Arduino like fire sensor, ultrasonic sensor, light sensor and many more. Here, we use ultrasonic sensor to detect the distance of an object. There are wide range of applications for ultrasonic sensor and many innovations are shaping the future with ultrasonic sensors as they can be used in car parking, robot sensing, liquid level control systems, trash level monitoring systems and a lot more.

In this project, we will be using ultrasonic sensors with Arduino. With the help of ultrasonic sensors we can measure the distance of an object. As these sensors are used in many real life applications, we need innovative solutions for various problems. The microcontroller in an Arduino plays a vital role in performing entire demonstration. So this gives us the information about Arduino boards combination with the other sensors as it needs both hardware and software components.

Keywords: Arduino, Embedded systems, Microcontroller, Ultrasonic sensor,

INTRODUCTION

In this project, we have covered the basic details regarding HC-SR04 Ultrasonic sensor, about how it works using Arduino board in applications as per the requirements.

Arduino UNO is an open source microcontroller board onmicrochip AT mega 328P microcontroller. It is equipped by sets of digital and analog input/output pins that can be interfaced to various expansion boardsand other circuits.It has 14 digital input/output pins, 6 analog input/output pins and it is programmable with Arduino IDE (Integrated Development environment. It is a USB based Arduino board.

Ultrasonic sensors are very beneficial to measure distant objects. Ultrasonic sensor is an electronic device used to determine the distance of an object by emitting soundwaves, and then converts the reflected sound wave into an electricalsignal. An ultrasonic wave is nothing but sound signal that can bemeasured and displayed at receiving end. An ultrasonic wave travels faster than an audible sound. It consists of two components, A transmitter and a receiver. The Transmitter transmits sound wave produced by electrical energy and the receiver encounters the echo and the returned sound wave to electrical energy.

In order to calculate the distance between the sensor and the object, the sensor measures the time it takes between the emissions of sound by transmitter to its contact with receiver.

1.3 MOTIVATION OF THE PROJECT

The motivation for this project is drawn from the desire to work on hardware as well as on software. The ease with which an Arduino can obtain sensor values is one of the features that makes it so useful. Sensors are devices that convert a physical quantity, such as light intensity or temperature, into an electrical quantity. Making projects with Arduino is a popular choice for many students due to its large-scale applications. Another reason student prefers to go for an Arduino project is because it is cost-effective and easy to program. Arduino IoT Cloud is an application that helps makers build connected objects in a quick, easy and secure way. You can connect multiple devices to each other and allow them to exchange real-time data. You can also monitor them from anywhere using a simple user interface. Arduino code is written in C++ with an addition of special methods and functions, which we'll mention later on. C++ is a human-readable programming language. When you create a 'sketch', it is processed and compiled into machine language. after learning Arduino you should learn automation and interfacing different sensors with Arduino. It helps in home automation projects and different kind of fun projects. Then you should learn UART, I2C, and Serial communication. That will help you to do wireless projects. Ultrasonic sensors can measure the distance to a wide range of objects regardless of shape, color or surface texture. They are also able to measure an approaching or receding object.

Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing. The transducer of the sensor acts as a microphone to receive and send the ultrasonic sound. Our ultrasonic sensors, like many others, use a single transducer to send a pulse and receive the echo. If the transmitter sends a small pulse of signal when the signal reaches any object it returns back to the receiver. We are calculating the distance using the traveling time of the signal in the air.

SCIENTIFIC BASIS

Ultrasonic sensors measure the distance to a wide range of objects irrespective of shape, color, or surface texture unless the material is soft like wool as it would absorb sound and a buzzer beeps when an obstacle is encountered. Technology is developed day by day in the world and ultrasonic sensors are used for Non-Destructive Testing and also for detecting emitted ultrasonic waves reflecting from the workspace.

THE METHODOLOGY

Ultrasonic sensing is one of the best ways to sense closeness and detect levels with high relativity.

The ultrasonic sensor works by sending sound waves at a frequency above the range of human hearing. The transducer present in this ultrasonic sensor acts as a microphone to receive and send sound.

In ultrasonic sensors, high-frequency sound waves will reflect from boundaries leading to distinct echo patterns.

The drawback of optical technologies to detect transparent and other items may be overcome by ultrasonic sensors.

Our ultrasonic sensor like many others uses a single transducer to send a pulse and to receive the echo.

Ultrasonic sensor sends an ultrasonic pulse out at certain frequency which travels through the air and if there is any obstacle or object it will bounce back to the sensor. By calculating the speed of sound the distance is calculated.

Hence, ultrasonic sensor gives the distance to a target by measuring the time lapses between the sending and receiving of the pulses.

OUTCOME OF THE PROJECT

The outcome of the project is that with the Ultrasonic sensor we can measure the distance of an object. When we take the object near to the sensor the values will be increasing and when the object is moved far away from the sensor the distance will be decreasing. So by this we can sense the distance of the object. Ultrasonic sensors are highly accurate and can be used to detect very small alterations in position. They can also measure the thickness of an object as well as the depth of the parallel surface.

DESCRIPTION OF ARDUINO ULTRASONIC SENSOR

This study aims at designing a Motion detector with a distance display at the liquid crystal display and producing a sound alarm. The use of Arduino Uno, LCD, LEDs, and Piezo Buzzer is cheap since the design of the circuit is not sophisticated. Arduino is a low-cost and effective microcontroller. It uses readily available and cheap appliances which can easily be found in electronics dealers.

WHAT IS ARDUINO?

The word "Uno" means "One" in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards; it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have now evolved to newer releases. The ATmega328 on the board comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.

While the Uno communicates using the original STK500 protocol, it differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments.

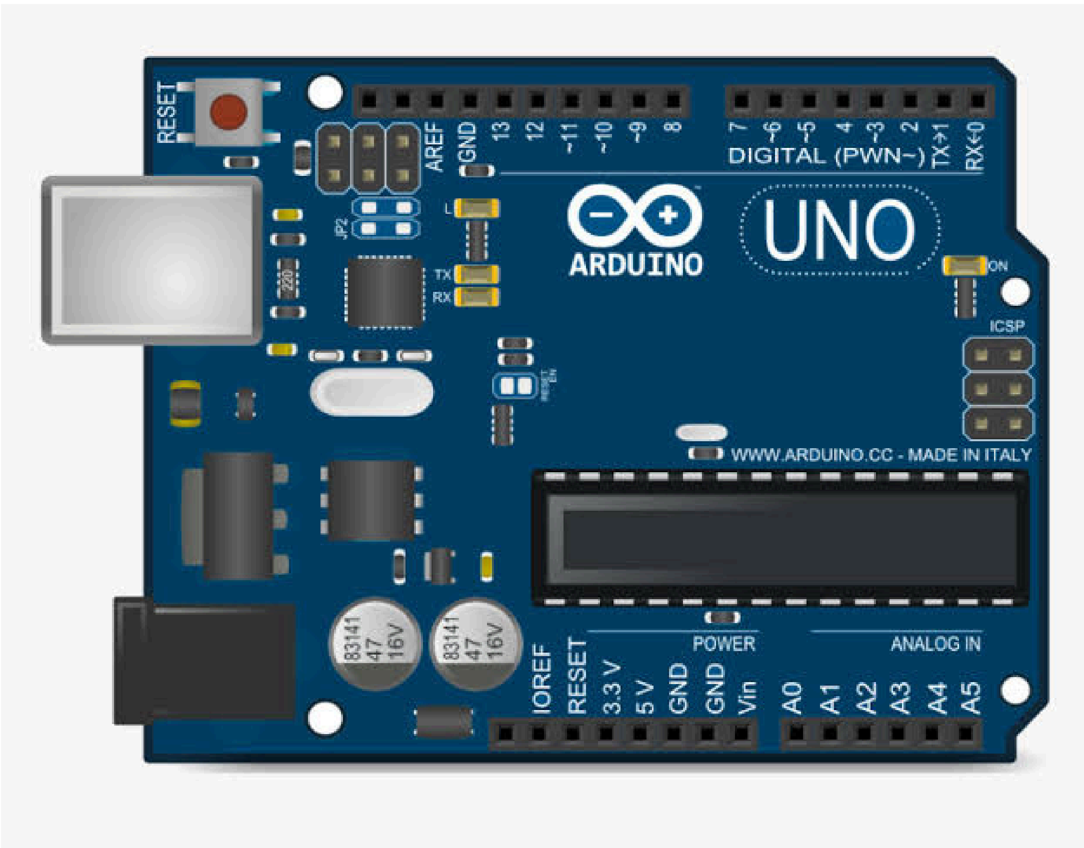
HARDWARE DESCRIPTION OF ARDUINO

The Arduino Uno is an open-source microcontroller Board based on the Microchip ATmega328P Microcontroller and developed by Arduino. The Board is equipped with sets of digital and analog Input/output (I/O) pins that may be interfaced with various expansion boards (shields) and other circuits. The board has 14 digital I/O pins (six capable of PW output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by a USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. A good example of the Arduino Uno microcontroller is shown in the figure.

Technical Specifications:

- Microcontroller: Microchip ATmega328P
- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 can provide PWM output)
- PWM Pins: 6 (Pin # 3, 5, 6, 9, 10 and 11)
- UART: 1
- I2C: 1
- SPI: 1
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA

- Flash Memory: 32 KB of which 0.5 KB used by boot loader.
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g
- ICSP Header: Yes
- Power Sources: DC Power Jack & USB Port

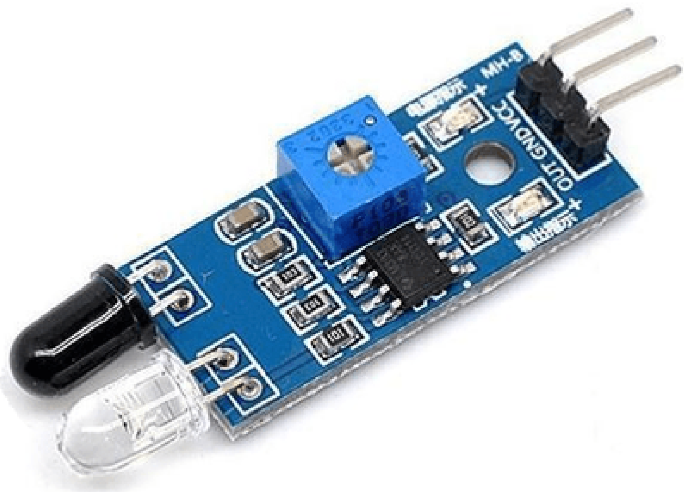


TYPES OF SENSORS

There is a wide range of sensors used in Arduino. Sensors are widely used in industrial applications and also they can be helpful in our day-to-day life like in food processing industries, restaurants and etc. The types of sensors are classified based on their functioning and depending upon the quantities like an infrared sensor is used to operate television remote, passive IR sensor is used for automatic opening of doors in shopping malls and there are many more applications with the use of sensors. Now we will look at the types of sensors in arduino:

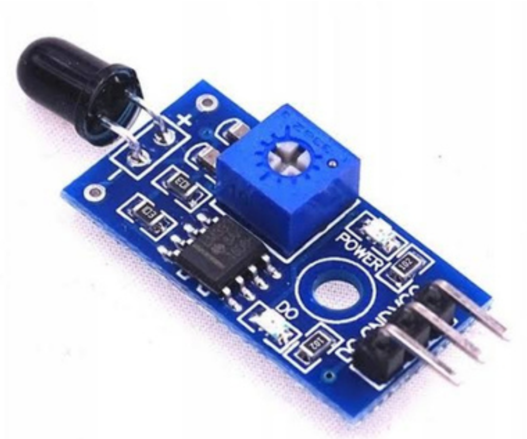
IR SENSOR:

An IR sensor is an electronic device that is used to sense some objects which are in our surroundings and IR sensor can detect the object and it can measure only infrared radiation. They are used in gas warning devices, gas analyzers medical gas measurement technology, flame detectors so these devices are used in the measurement of infrared radiation within the spectral range.



FLAME SENSOR

Flame sensor is also called as flame detector which is used to detect and respond to the presence of fire. It helps in flame detection. Fire sensors are used in industrial applications like hydrogen stations, industrial gas turbines, and domestic heating systems and also gas powered cooking devices the primary purpose of a flame detector is to minimise the risks associated with combustion.



ULTRASONIC SENSOR

An ultrasonic sensor is a device which is used to measure the distance of an object using the ultrasonic sound waves. Ultrasonic sensors are widely used in robot sensing, liquid level control in beverages, automatic doors, automatic hand washes in restaurants and there are many other applications of ultrasonic sensor.

GAS SENSOR

A gas sensor is used to detect toxic or explosive gases and it is also used to measure the gas concentration. They are employed in factories and manufacturing unit to identify the gas leakage and it also detect the smoke and carbon monoxide in homes. Gas sensors are used in detecting hazardous inert gases and ensuring safety air quality in many different industries. They are helpful in monitoring the indoor air quality and beneficial in medical and life science industries and many more.

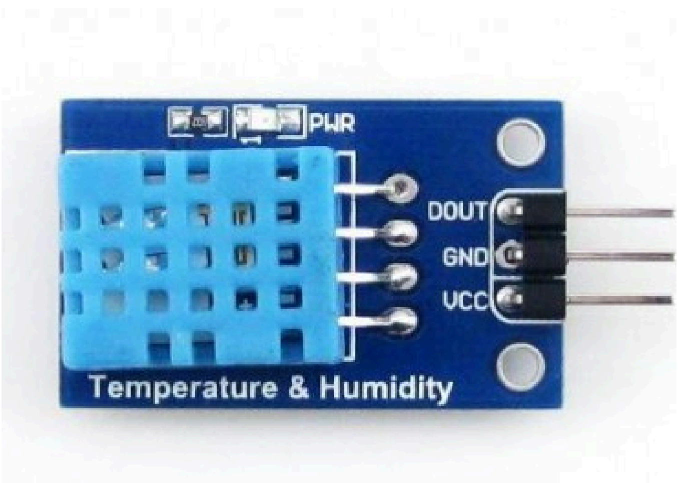
DC MOTOR

A DC motor is a type of electrical machine which converts the electrical energy into mechanical energy. It takes the electrical power through direct current and it will convert this energy into mechanical rotation. Small DC motors are very helpful in tools, toys and Home appliances and large DC motors can be used in propulsion of electric vehicles, elevators and hoists.

DC motor is beneficial where high starting torque is required and the speed variations are also possible. For instance, they can be used in the air compression, vacuum cleaners, sewing machines and further more.

DHT 11 SENSOR:

The sensor is used in detecting the temperature and humidity. It is used in measuring the humidity and temperature values in heating ventilation and air conditioning systems. It is also used in weather stations to predict the weather conditions. Offices, museums, cars, and industries use the sensor to measure the humidity values and it is also used as a safety measure.



ULTRASONIC SENSOR

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves and converting the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear). Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has traveled to and from the target).

COMPONENTS

1. Ultrasonic Sensor – HC-SR04
2. ATmega32
3. 16x2 LCD Display

ULTRASONIC RANGING MODULE HC-SR04

FEATURES

Ultrasonic ranging module HC-SR04 provides 2cm – 400cm non-contact measurement function. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- Using IO trigger for at least 10us high level signal.
- The module automatically sends eight 40kHz and detect whether there is a pulse signal back.

- If the signal back, through high level, time of high output IO duration is the time from sending ultrasonic to returning.
- Test distance = (high level time velocity of sound(340M/S)/2

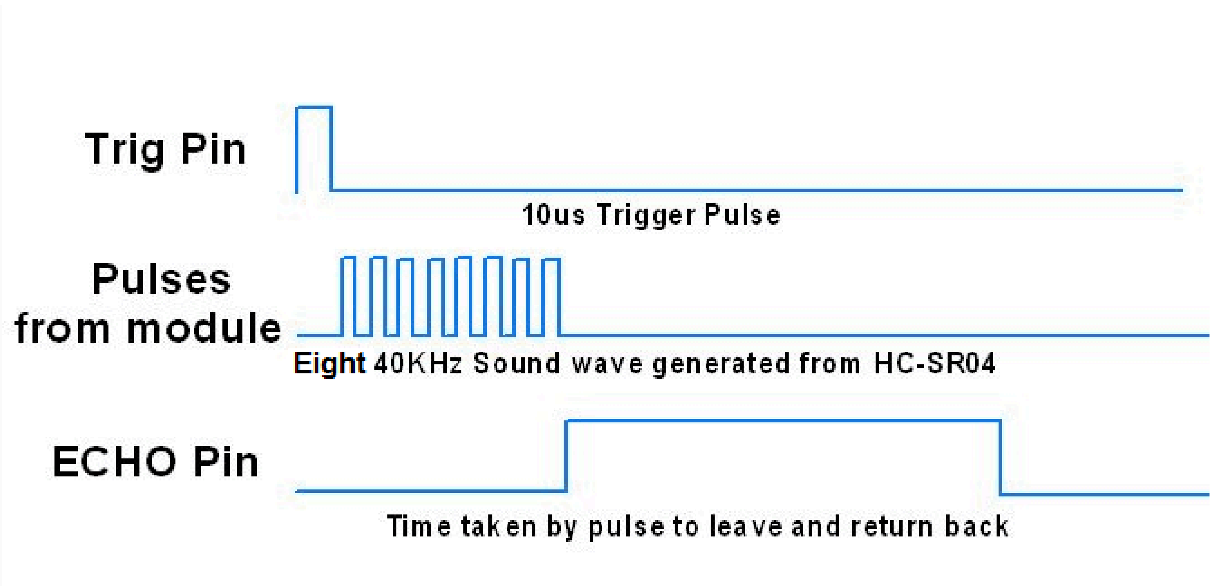
ELECTRICAL PARAMETERS

1	*Working Voltage*	*DC 5V*
2	*Working Current*	*15mA*
3	*Working Frequency*	*40Hz*
4	*Max Range*	*4m*
5	*Min Range*	*2cm*
6	*Measuring Angle*	*15 degree*
7	*Trigger Input Signal*	*10uS TTL Pulse*
8	*Echo Output Signal*	*Input TTL lever signal and the range in proportion*
9	*Dimension*	*45*20*15mm*

TIMING DIAGRAM

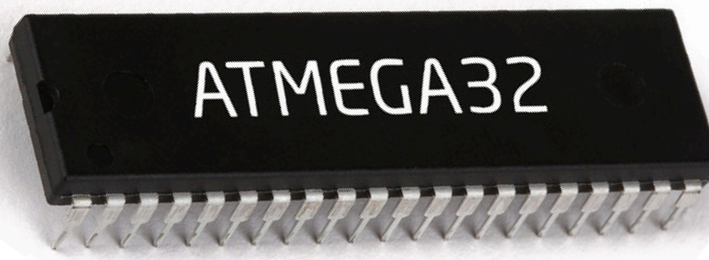
The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion. You can calculate the range through the time interval between sending the trigger signal and receiving the echo signal. Formula: uS/58 - centimeters or uS/148 -inch: or the range - high-level time*velocity (340M/S)/2; It is advised to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.

TIMING DIAGRAM OF HC-SR04



ATmega32

The ATmega32 is a low-power CMOS 8-bit microcontroller based on the AVR-enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32 achieves throughputs approaching I MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.



SPECIFICATION

1.High-performance, Low-power AVR 8-bit
Microcontroller

2. Advanced RISC Architecture

a)131 Powerful Instructions Most Single-clock Cycle Execution

b)32 x 8 General Purpose Working Registers

c)Fully Static Operation

d)Up to 16 MIPS Throughput at 16 MHz

e)On-chip 2-cycle Multiplier

3.Peripheral Features

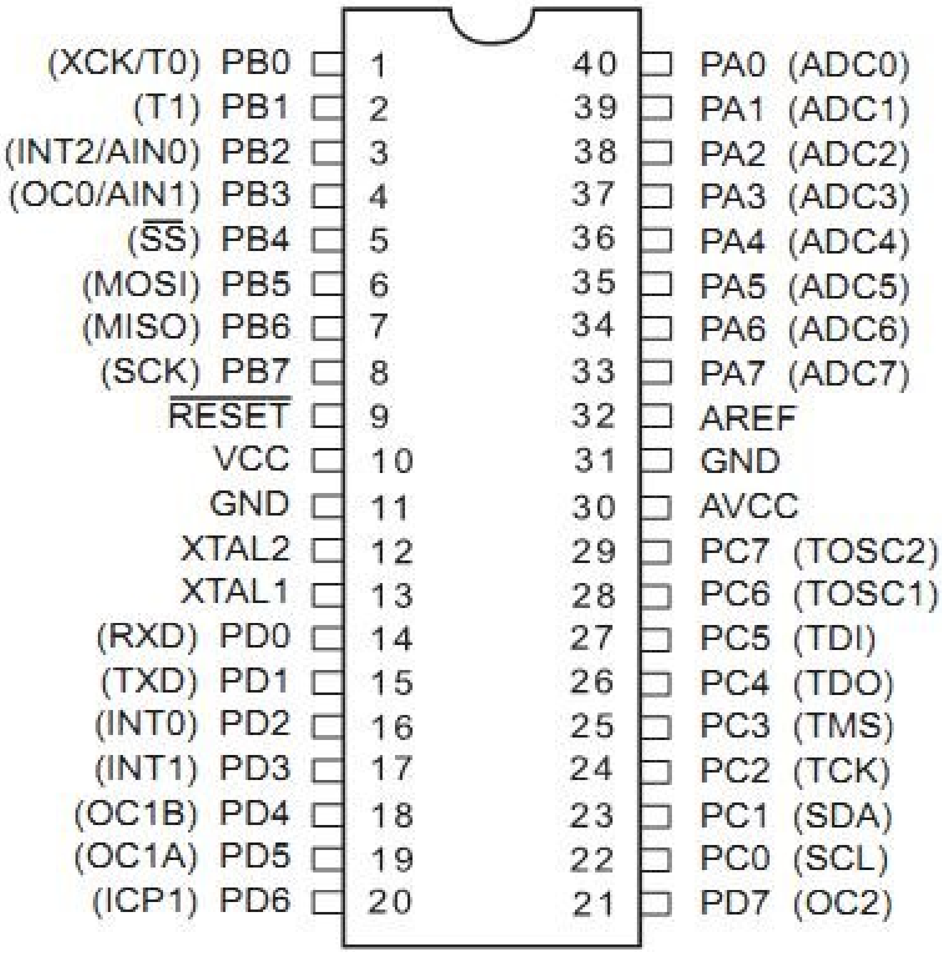
a)Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes

b)One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
c)Rear Time Counter with Separate Oscillator
d)Four PWM Channels
e)8-channel, 10-bit ADC
f)Master/Slave SPI Serial Interface
g) Programmable Watchdog Timer with Separate On-chip Oscillator
h) On-chip Analog Comparator
4.Special Microcontroller Features
a)Power-on Reset and Programmable Brown-out Detection
b)Internal calibrated RC Oscillator
c)External and Internal Interrupt Sources
I)Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby

GPIO

- **VCC:** Digital supply voltage.
- **GND:** Ground.
- **Port A (PA7-PA0):** Port A serves as the analog inputs to the A/D Converter. Port A is also used as an 8-bit bi-directional I/O port if the analog to digital converter is not used. The Port A output buffers have symmetrical drive characteristics. When pins PA0 to PA7 are used as inputs, they will source current if the internal pull-up resistors are activated. When a reset condition becomes active, Port A pins are tri-stated even if the clock is not running.

- **Port B (PB7-PB0):** Port B is an 8-bit bi-directional I/O port with internal pull-up resistors. The Port B output buffers also have symmetrical drive characteristics with both high sink and source capability. Port B pins which are externally pulled low will source current if the pull-up resistors are activated. When a reset condition becomes active and even if the clock is not running, the Port B pins become tri-stated.
- **Port C (PC7-PC0):** Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). If the pull-up resistors are activated Port C output buffers also have symmetrical drive characteristics with both high sink and source capability. Port C pins which are externally pulled low will source current. When a reset condition becomes active the Port C pins are tri-stated, even if the clock is not winning. The pull-up resistors on pins PC5 (TDI), PC3 (TMS) and PC2 (TCK) will be activated if the JTAG interface is enabled even if a reset occurs.
- **Port D (PD7-PD0):** Port D is an 8-bit bi-directional I/O port with internal pull-up resistors. The Port D output buffers also have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins which are externally pulled low will source current if the pull-up resistors are activated. When a reset condition becomes active the Port D pins becomes tri-stated, even if the clock is not running.



pin diagram of ATmega 32

TIMER

Timers are standard features of almost every microcontroller. So it is very important to learn their use. Since an AVR microcontroller has very powerful and multifunctional timers, the topic of timer is somewhat vast. Moreover there are many different timers on chip. So this section on timers will be multipart. I will be giving basic introduction first.

What is a timer?

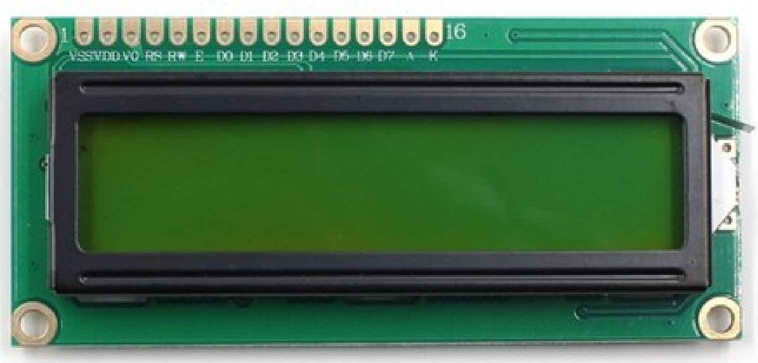
A timer in simplest term is a register. Timers generally have a resolution of 8 or 15 Bits. So an 8 bit timer is 8Bits wide so capable of holding value wishing 0-255. But this register has a magical property!

Its value increases/decreases automatically at a predefined rate (supplied by the user). This is the timer clock. And this operation does not need CPUs intervention. Since Timer works independently of the CPU it can be used to measure time accurately. The timer upon certain conditions takes some action automatically or informs the CPU. One of the basic conditions is the situation when the timer OVERFLOWS i.e. it's counted up to its maximum value (255 for 8 BIT timers) and rolled back to 0. In this situation, the timer can issue an interrupt and you must write an Interrupt Service Routine (ISR) to handle the event.

LCD Display

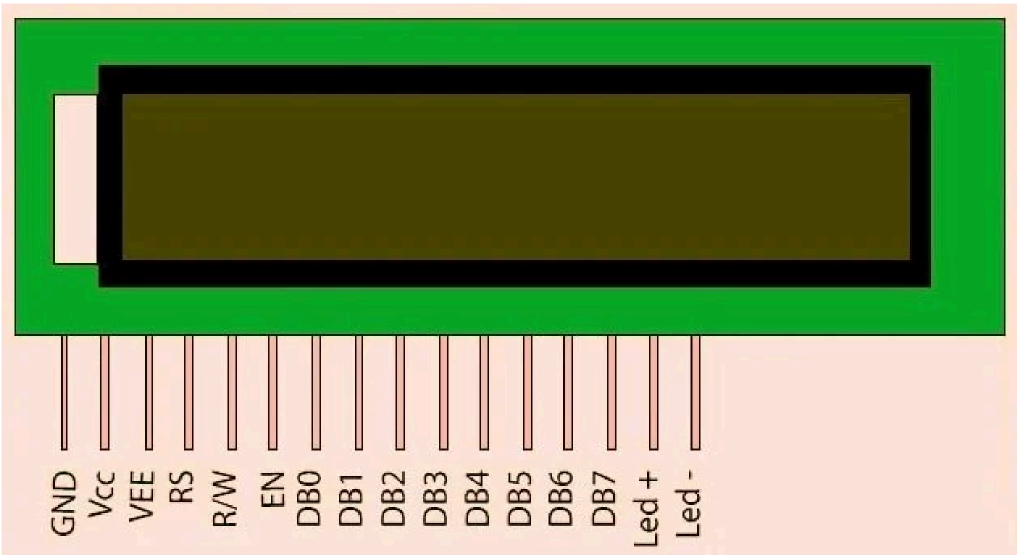
LCD (Liquid Crystal Display) is an electronic display system. An l6x2 LCD display is a very basic system and is commonly used in various devices and circuits. LCDs are preferred over seven segments and other multi-segment LEDs. The advantages of LCDs are as follows:

- LCDs are economical.
- They are easily programmable.
- A number of characters can be displayed.
- Very compact and light.
- Low power consumption.



An l6x2 LCD means it can display 16 characters per line and 2 such lines are there. In this LCD every character is displayed in a 5x7 pixel matrix. LCD possesses two registers: Data and Command registers. The command register stores the command instructions given to the LCD. A command can be defined as an instruction given to LCD to do a predefined task. For example, initializing the LCD, clearing the screen, controlling the cursor position, controlling the display etc. The data register stores the data which is displayed on the LCD screen. The data is the ASCII value of the character which is displayed on the LCD screen.

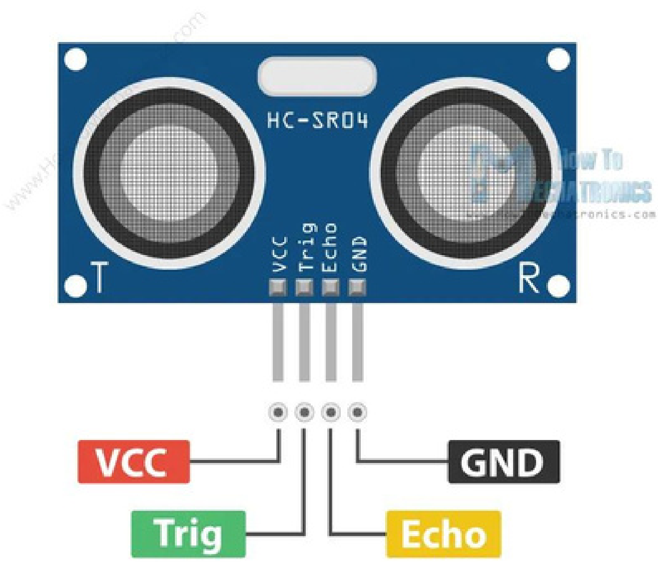
PIN DIAGRAM



1.Ground: Ground(0V)
2.Vcc: Supply Voltage 5V
3.VEE: Contrast adjustment through a variable resistor

4.Register Select(RS): Selects command register when low; and data register when high.
5.Read/write(RW): Low to write to the register; High to read from the register.
6.Enable(EN): Send data to data pins when a high to low pulse is given.
7.DB0-DB7: 8-bit data pins.

HC-SR04 ULTRASONIC SENSOR PINOUT



The sensor has 4 pins. VCC and GND go to 5V and GND pins on the Arduino, and the Trig and Echo go to any digital Arduino pin. Using the Trig pin we send the ultrasound wave from the transmitter, and with the Echo pin we listen for the reflected signal.

WORKING PRINCIPLE OF ULTRASONIC SENSOR

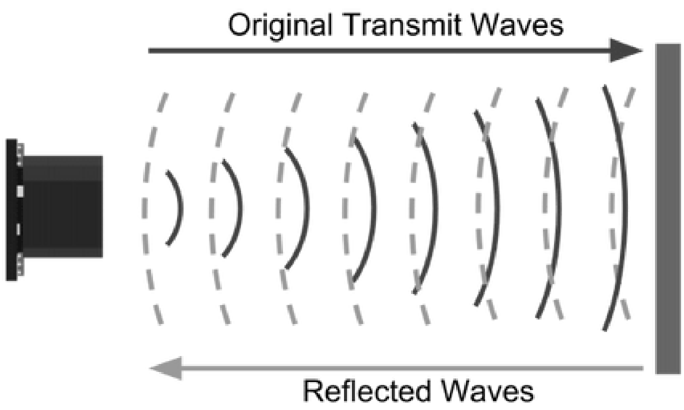
The ultrasonic sensor (or transducer) works on the same principles as a radar system. An ultrasonic sensor can convert electrical energy into acoustic waves and vice versa. The acoustic wave signal is an ultrasonic wave traveling at a frequency above 18kHz. The famous HC SR04 ultrasonic sensor generates ultrasonic waves at 40kHz frequency.

Typically, a microcontroller is used for communication with an ultrasonic sensor. To begin measuring the distance, the microcontroller sends a trigger signal to the ultrasonic sensor. The duty cycle of this trigger signal is 10µS for the HC-SR04 ultrasonic sensor. When triggered, the ultrasonic sensor generates eight acoustic (ultrasonic) wave bursts and initiates a time counter. As soon as the reflected (echo) signal is received, the timer stops. The output of the ultrasonic sensor is a high pulse with the same duration as the time difference between transmitted ultrasonic bursts and the received echo signal.

Theoretically, the distance can be calculated using the TRD (time/rate/distance) measurement formula. Since the calculated distance is the distance traveled from the ultrasonic transducer to the object and back to the transducer it is a two-way trip. By dividing this distance by 2, you can determine the actual distance from the transducer to the object.

The Ultrasonic waves travel at the speed of sound (343 m/s at 20°C). The distance between the object and the sensor is half of the distance traveled by the sound wave.

The Ultrasonic sensors work by emitting sound waves with a frequency that is too high for a human to hear. These sound. waves travel through the air with the speed of sound, roughly 343 m/s. If there is an object in front of the sensor, the sound waves get reflected and the receiver of the ultrasonic sensor detects them. By measuring how much time passed between sending and receiving the sound waves, the distance between the sensor and the object can be calculated.



At 20°C, the speed of sound is roughly 343 m/s or 0.034 cm/us. Let's say that the time between sending and receiving the sound waves is 2000 microseconds. If you multiply the speed. of sound by the time the sound waves traveled, you get the distance that the sound waves traveled.

DISTANCE = SPEED X TIME

But that is not the result we are looking for. The distance between the sensor and the object is actually only half this distance because the sound waves traveled from the sensor to the object and back from the object to the sensor. So you need to divide the result by two.

Distance (cm) = Speed of sound (cm/µs) × Time (µs)/2

For Example,

Distance (cm) = 0.0343 (cm/µs) × 2000 (µs)/2
= 34.3 cm

Temperature Dependence of the Speed of Sound

The speed of sound actually depends strongly on temperature and to a far lesser degree on the humidity of the air. Wikipedia states that the speed of sound increases with roughly 0.6 m/s per degree Celsius. For most cases at 20°C you can just use 343 m/s but if you want to get more accurate readings, you can calculate the speed of sound with the following formula:

The speed of sound actually depends strongly on temperature and to a far lesser degree on the humidity of the air. Wikipedia states that the speed of sound increases with roughly 0.6 m/s per degree Celsius. For most cases at 20°C you can just use 343 m/s but if you want to get more accurate readings, you can calculate the speed of sound with the following formula:

$$V\ (m/s) = 331.3 + (0.606 \times T)$$

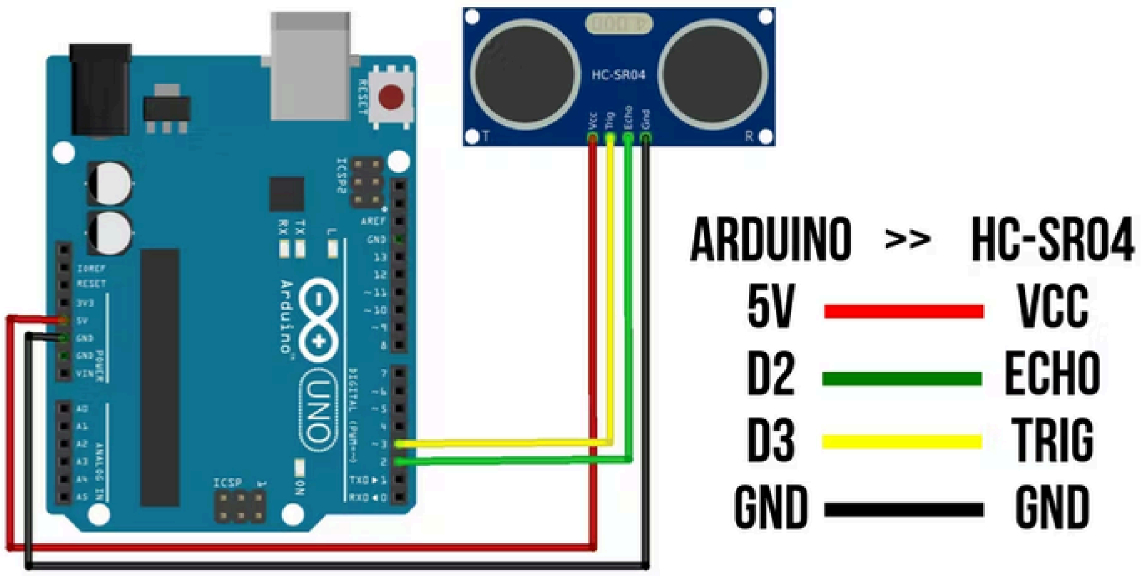
V = Speed of sound (m/s)

T = Air Temperature (°C)

This formula doesn't include the humidity since its effect on the speed of sound is only very small.

INTERFACING OF ULTRASONIC SENSOR WITH ARDUINO

- Below, the image shows the interfacing of the ultrasonic sensor with Arduino. The components required for the interfacing are,
- Arduino Uno
- HC-SR04-Ultrasonic Sensor
- Connecting wires



The Ground and the VCC pins of the module needs to be connected to the Ground and the 5 volts pins on the Arduino Board respectively and the trig and echo pins to any Digital I/O pin on the Arduino Board.

We have to download the Arduino IDE for our system and upload the below code in our Arduino Uno:

```
*/
// defines pins numbers
const int trigPin = 9;
const int echoPin = 10;
// defines variables
long duration;
int distance;
void setup() {
 pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as an Input
Serial.begin(9600); // Starts the serial communication
}
void loop() {
 // Clears the trigPin
digitalWrite(trigPin, LOW);
 delayMicroseconds(2);
// Sets the trigPin on HIGH state for 10 microseconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
// Reads the echoPin, returns the sound wave travel time in
microseconds
 duration = pulseIn(echoPin, HIGH);
// Calculating the distance
distance = duration * 0.034 / 2;
// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
 Serial.println(distance);
 }
```

CODE EXPLANATION:

First we have to define the Trig and Echo pins. In this case they are the pins number 9 and 10 on the Arduino Board and they are named trigPin and echoPin. Then we need a Long variable, named “duration” for the travel time that we will get from the sensor and an integer variable for the distance.

```
// defines pins numbers
const int trigPin = 9;
const int echoPin = 10;

// defines variables
long duration;
int distance;
```

In the setup, we have to define the trigPin as an output and the echoPin as an Input and also start the serial communication for showing the results on the serial monitor.

```
void setup() {
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
  Serial.begin(9600); // Starts the serial communication
}
```

In the loop first we have to make sure that the trigPin is clear so you have to set that pin on a LOW State for just 2 µs. Now for generating the Ultra sound wave we have to set the trigPin on HIGH State for 10 µs.

```
// Clears the trigPin
digitalWrite(trigPin, LOW);
delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
```

Using the pulseIn() function we read the travel time and put that value into the variable “duration”. This function has 2 parameters, the first one is the name of the Echo pin and for the second is the state of the pulse we are reading, either High or Low.

```
// Reads the echoPin, returns the sound wave travel time in
microseconds
duration = pulseIn(echoPin, HIGH);
```

In this case, we need this set to it HIGH, as the HC-SR04 sensors sets the Echo pin to High after sending the 8 cycle ultrasonic burst from the transmitter. This actually starts the timing and once we receive the reflected sound wave the Echo pin will go to Low which stops the timing. At the end the function will return the length of the pulse in microseconds.

For getting the distance we will multiply the duration by 0.034 and divide it by 2 as we explained this equation previously.

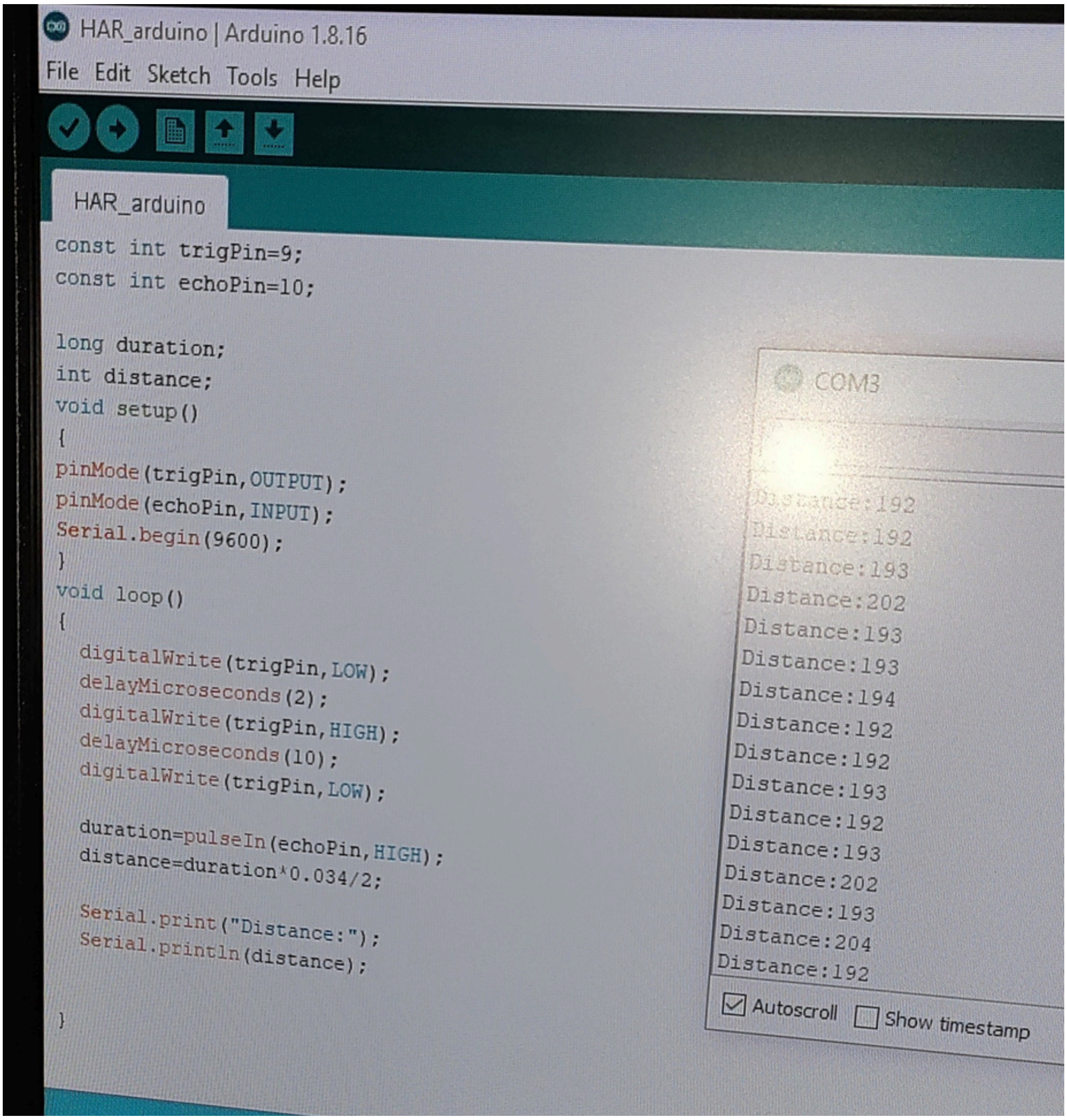
```
// Calculating the distance
distance= duration*0.034/2;

// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance);
```

And the end we will print the value of the distance on the Serial Monitor.

OUTPUT

After uploading the code, display the data with Serial Monitor. Now try to give an object in front of the sensor and see the measurement.



Innovations using Ultrasonic Sensors

- Transportation sector: Used in self- driving systems
- Used in smartphone displays
- Parking sensors in cars
- Trash-level monitoring.
- Used in drink filling machines.

RESULT

Ultrasonic sensor can measure the distance of an object. Here we require hardware and software component while working with aurdino. The connections in the aurdino board should be correct before we start the demonstration. The code should be written and uploaded. As the Ultrasonic sensor can measure distance and if there is any object on its way, and if we are moving the object back and forth, it can detect the distance of the object.

The reason we use an ultra sonic sensor is that is highly accurate. It gives us the accurate results, as its used to detect even small movement of the object. It can detect a wide range of materials irrespective of their size, shape and colour. The displacement of the object can be determined. Arduino has no moving parts so it has long life. They have an easy interface and it can be operated easily. Thus, the ultrasonic sensor have wide range of applications in the industry and this project explains about one of the ways in which the sensor can detect the distance of an object.

Applications of an Ultrasonic Sensor

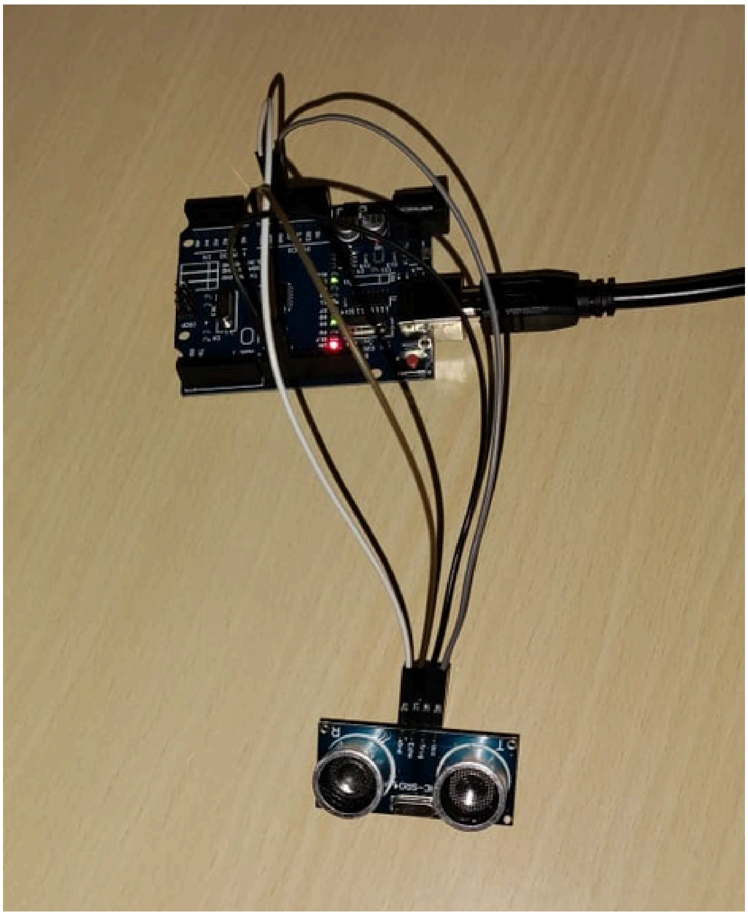
- It Uses to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, pathfinding robot etc.
- Used to measure the distance within a wide range of 2cm to 400cm.
- Used to map the objects surrounding the sensor by rotating it.
- Depth of certain places like wells, pits etc can be measured since the waves can penetrate through the water.
- Hope this article helps you to understand the

Limitations of Ultrasonic Sensor

Ultrasonic sensors such as the HC-SR04 can efficiently measure distances up to 400 cm with a slight tolerance of 3 mm.

However, if a target object is positioned such that the ultrasonic signal is deflected away rather than reflected back to the ultrasonic sensor, the calculated distance can be incorrect. In some cases, the target object is so small that the reflected ultrasonic signal is insufficient for detection, and the distance cannot be measured correctly.

Furthermore, objects like fabric and carpet can absorb acoustic signals. If the signal is absorbed in the target object’s end, it cannot reflect back to the sensor, and hence, the distance cannot be measured.



REFERENCES

https://www.javatpoint.com/arduino-sensors
https://learn.sparkfun.com/tutorials/what-is-an-arduino/all

https://opensource.com/resources/what-arduino

https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

https://www.tutorialspoint.com/arduino/arduino_ultrasonic_sensor.htm#:~:text=The%20HC%2DSR04%20ultrasonic%20sensor,or%201%E2%80%9D%20to%2013%20feet

https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6
https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/