ASSIGNMENT NO: 03

TITLE:

Study of application of Open source prototype platform

AIM:

Open source prototype platform- Raspberry-Pi/Beagle board/Arduino

-Simple program digital read/write using LED and Switch

-Analog read/write using sensor and actuators

Theory:

Arduino is popular microcontroller. It's cheap. There's lots of sensors that can be used with Arduino, as well as actuators.

Introducing the Arduino

The Arduino is a small development board with a brain (also known as a microcontroller) that you can program. It interacts with the real world through leds, sensors, motors, LCDs, buzzers, etc...

What's an Arduino?

Arduino is essentially a tiny computer that can connect to electrical circuits. The Arduino Uno is powered by an ATmega328P chip, it is the biggest chip on the board as you can see on the picture above. That's where you store your programs.

The top row of the Arduino has 14 digital pins, labelled 0-13. These pins can act as either inputs or outputs. You can connect them to your Circuits to turn them on or off. You can also read buttons – see if a button is either pressed or not.

On the bottom left row, you can see the power pins. The Arduino has 3.3V or 5V supply. This is really useful since most components require 3.3V or 5V. You will also find some pins labelled "GND" on the Arduino, these are ground pins.

Arduino UNO R3 board with ATmega328P

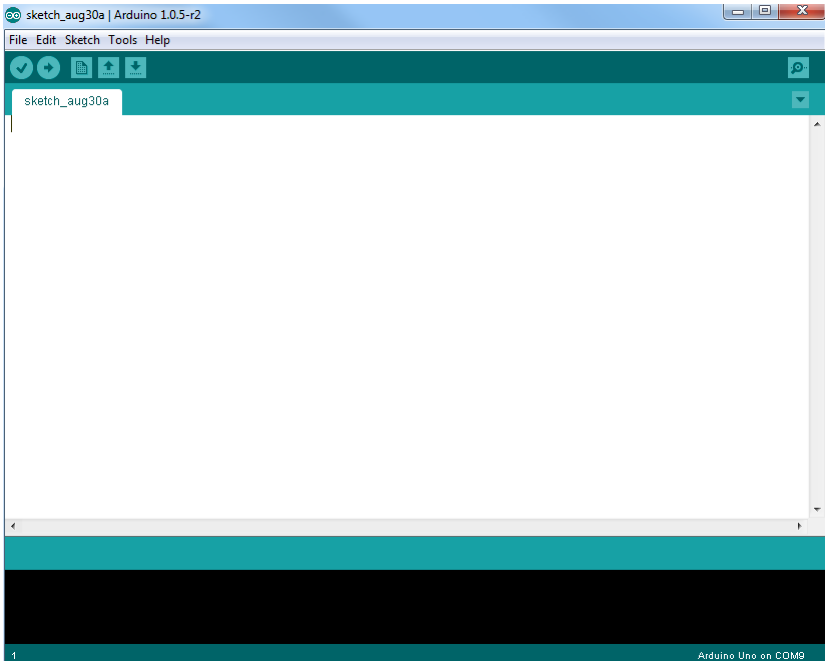
On the bottom right row, you can see the analog input pins, labelled A0-A5. These pins are used to make analog measurements of sensors or other components. Analog inputs are especially good for measuring things with a range of possible values. For example measuring temperature sensors or potentiometers.

Downloading the Arduino IDE

You can load new programs onto the main chip - ATmega328p – via USB using the Arduino IDE. Visit the link below to download the latest Arduino IDE:

- http://arduino.cc/en/Main/Software

Official Arduino website does a great job explaining how to do it any of the three operating systems (Windows, Mac and Linux).



In the end, you should see a similar window on your computer.

Uploading an Arduino Sketch

Connect your Arduino UNO to your computer via USB.

For this example you will be uploading the most basic example that the Arduino has. Which is blinking an on-board LED or digital pin 13.

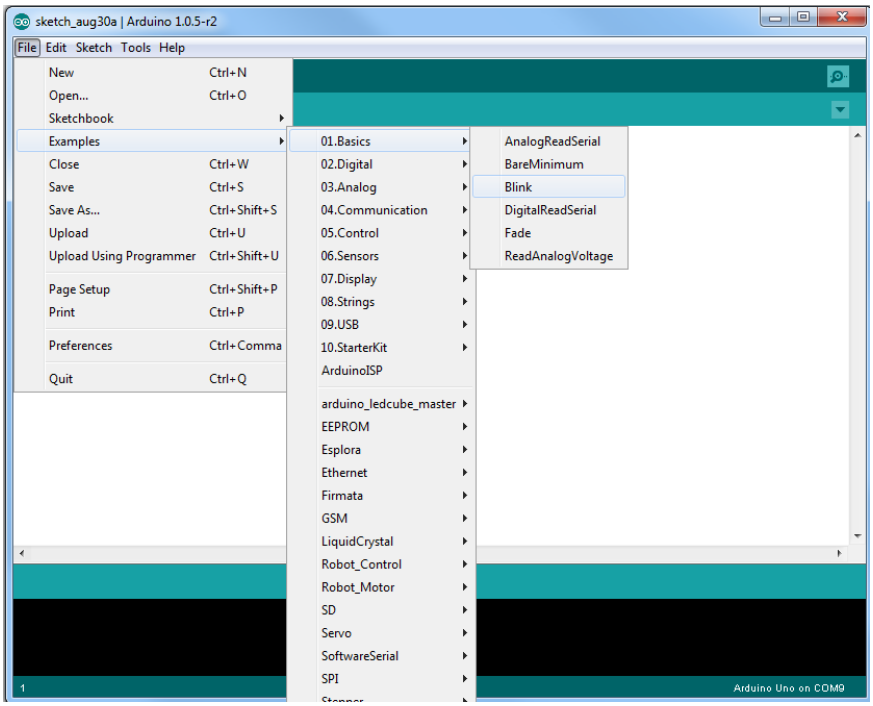
Open your Arduino IDE.

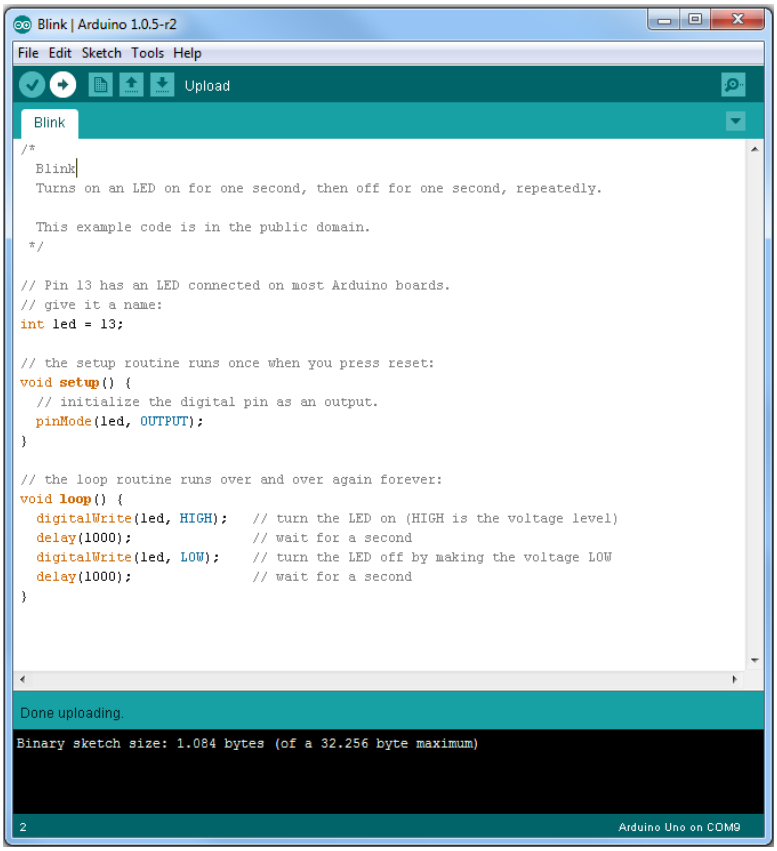
- Go to File > Examples > 01.Basics > Blink

By default your Arduino IDE comes pre-configured for the Arduino UNO. Left-click the "Upload" button and wait a few seconds until a

- "Done uploading." message appears.

This code simply blinks the on-board LED on your Arduino UNO (highlighted with red color). If you look closely you should see the LED staying on for one second and off for another second repeatedly.





Program:

```
void setup()
{
 pinMode(13, OUTPUT);
}
void loop()
{
 digitalWrite(13, HIGH);
 delay(1000); // Wait for 1000 millisecond(s)
 digitalWrite(13, LOW);
 delay(1000); // Wait for 1000 millisecond(s)
}
```

What is a sensor?

In simple words, a sensor is a device that "listens" to the physical environment and tells you what happens. Each sensor is able to listen to a specific input that can be light, heat, motion, moisture, pressure, or any one of a great variety of other environmental phenomena.

What is a actuator?

In the strict meaning, actuator is a device that converts energy in movement. It could be a valve or a motor. But I prefer not to be so strict, I consider an actuator anything that can convert electric energy in an output, for example a display, a led, a loudspeaker.

Digital and Analog

Information may be represented and transmitted both in analogic or digital format. The difference between them is that in analogic technology information is represented by amplitude variations, while in digital it's represented in binary format.

Arduino can do analog and digital inputs, and only digital outputs. The sensors themselves usually are analog, but most of times they have a small circuit that translates values into digital format. In some cases you'll find both connectors on the same device. So, analog pins on Arduino are always input. Digital pins can be input or output, your code will decide.

Libraries

In programming, library is a piece of code that someone else previously wrote that can help to use. Most of, if not all, the peripherals that can be attached to an Arduino have a

library. The library controls the hardware, hiding complexity behind some simple routines that someone else wrote and we can use.

Ultrasonic Sensor interfacing with Servo Motor

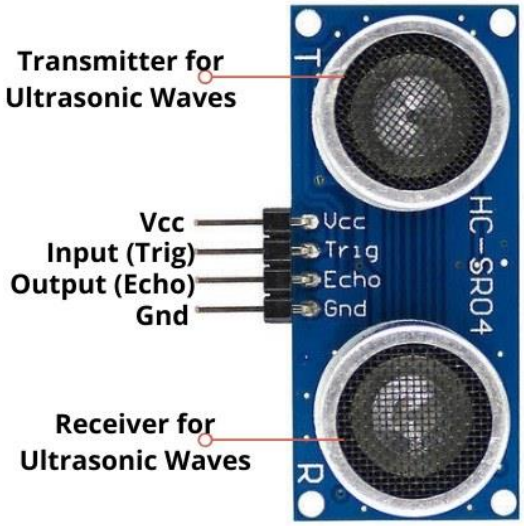
Step 1: Parts Needed

A HC-SR04 ultrasonic ranging sensor is used to measure distances. This sensor is very economical and provides 2cm to 400cm of non-contact measurement functionality with a good ranging accuracy.

The sensor has only four pins: VCC (Power), Trig (Trigger), Echo (Receive), and GND (Ground).

Connect the sensor:

VCC to proto board + line GND to proto board - line Trig to Arduino pin 9 Echo to Arduino pin 10



What is a Servo?

Servo motors are great devices that can turn to a specified angle or called position. Usually, they have a servo arm that can turn 180 degrees. Using the **Ardiuno**, we can control a servo to go to a specified position. As simple as that! Here we will see how to connect a servo motor and then how to turn it to different positions.

Connection to Servo

The next job is to connect your servo motor.

There are two common types of servo:

1. **White - Red - Black wired servo**
2. **Orange - Red - Brown wired servo**

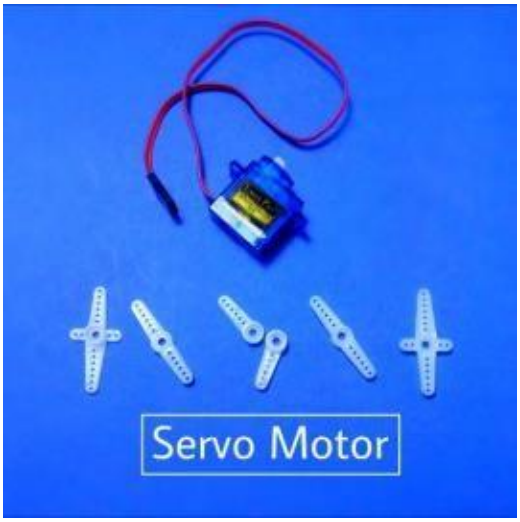
✓ If your servo has White - Red - Black wires, then connect it as follows

- White wire connects to Digital pin D4
- Black wire connects to GND pin
- Red wire connects to 3V3 pin

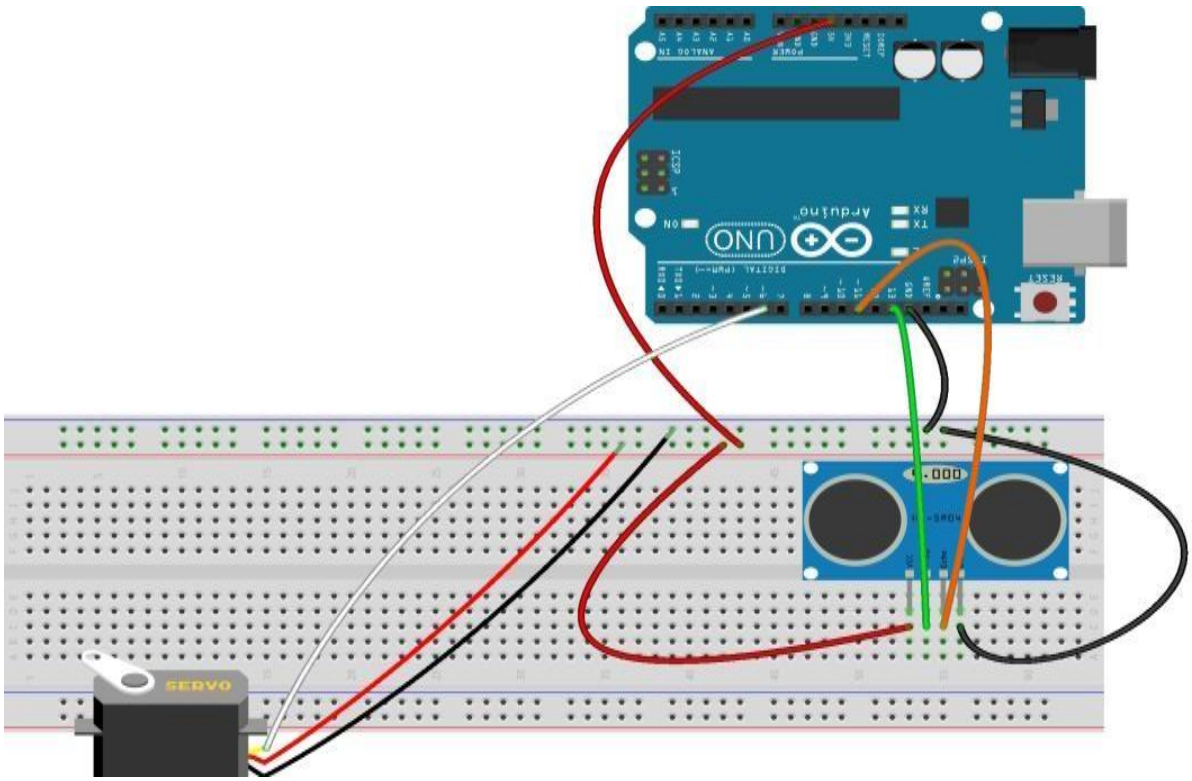
✓ If your servo has Orange - Red - Brown wires, then connect it as follows

- Orange wire connects to Digital pin D4.
- Brown wire connects to GND pin

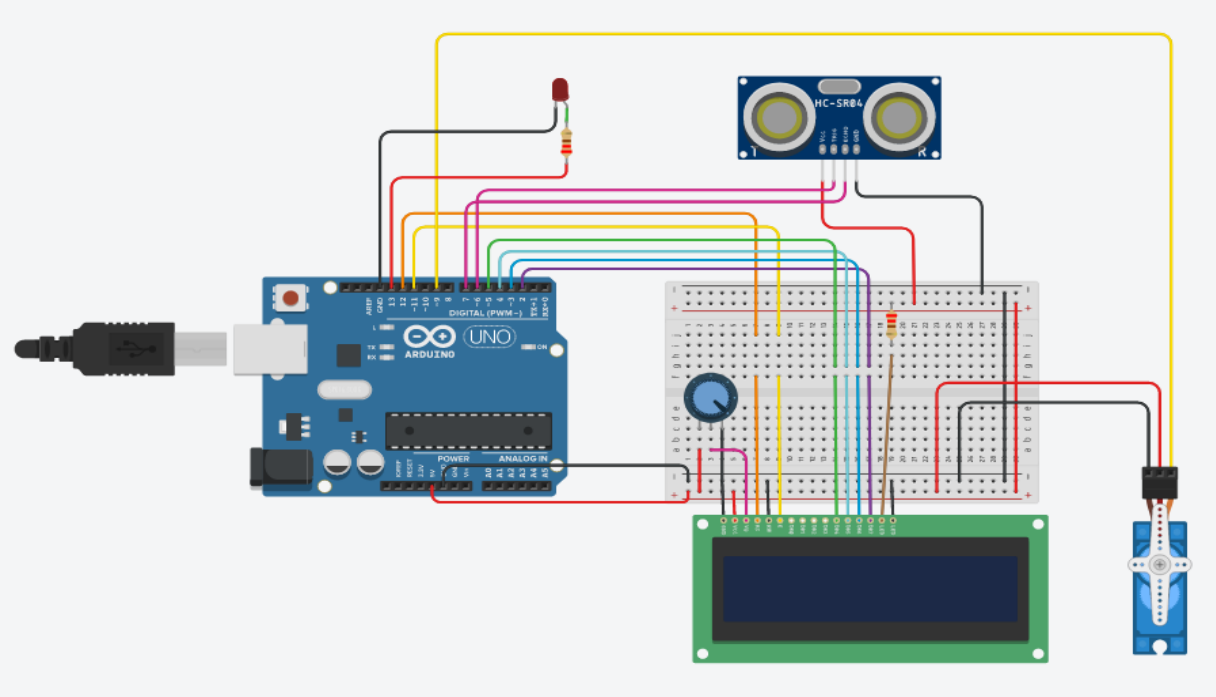
Red wire connects to 3V3 pin



Interfacing of Servo motor and Ultrasonic sensor with Ardiuno:



Simulation on TINKERCAD:



Program:

```
#include <LiquidCrystal.h>
// includes the LiquidCrystal Library
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
// Creates an LCD object. Parameters: (RS,EN,D4,D5,D6,D7 )
#include <Servo.h>
Servo myservo;                  // create servo object to control a servo
const int trigPin = 6;
const int echoPin = 7;

void setup() {
Serial.begin(9600); // initialize serial communication
myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}
void loop() {
long duration, cm;
pinMode(13, OUTPUT);
pinMode(trigPin, OUTPUT);
digitalWrite(trigPin, LOW);
```

```
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(20);
digitalWrite(trigPin, LOW);
pinMode(echoPin, INPUT);
duration = pulseIn(echoPin, HIGH);        // read echo
cm = microsecondsToCentimeters(duration); // convert time to distance
if ( cm > 100 && cm < 150)
{
digitalWrite(13, HIGH);
myservo.write(90);  // sets the final servo position
delay(10);
}
else
{
digitalWrite(13, LOW);
myservo.write(0);    // sets the initial servo position
delay(10);
}
Serial.print(cm);                //print distance on serial window
Serial.print("cm");
Serial.println();
delay(10);
lcd.setCursor(0,1);
lcd.print("Dist: ");
lcd.print(cm);
lcd.print(" cm");
delay(50);
}
long microsecondsToCentimeters(long microseconds) {
// The speed of sound is 340 m/s or 29 microseconds per centimeter.
return microseconds / 29 / 2; }
```

Conclusion: We have study application of Open source prototype platform board.