Department of Electronics and Electrical Engineering
Indian Institute of Technology Guwahati

Design Project Report (END-Semester Evaluation)

Title of the design project: HEXAPOD ROBOT

1. **Component/Equipment Used:-**

S.No.	Name of components with specifications	Quantity
1	Arduino microcontroller	1
2	servos	3
3	Soldering iron and solder	1
4	Hot glue gun	1
5	Wire stripper	1
6	Long-nose pliers	1
7	Diagonal cutter	1
8	Some wire for connecting the parts	
9	Board	1
10	Li- ion battery with charger	1
11	Sensor	1

2. Methodology Used:-

The aim of the project is to build an autonomous Hexapod Robot, that includes electronics, software & mechanical components integrated together to work in coordination among themselves.
But we have been focusing mainly on hardware implementation of the above design.

- **Principle behind movement of all the six legs in synchronism with coordination**
  Here we are using only 3 servos to move our hexapod robot instead of 6 or 18 servos which makes our design little bit complex but is more efficient than terrestrial old versions.
  - **To implement our movement we have use the following approach :-** One servo is being attached to first & third leg of left side of hexapod robot & second is being attached with first & third legs of right side of hexapod robot & third(last servo) will control the movement of both the middle legs. So for forward & backward movement of the legs, first of all the third servo controlling the right middle leg movement will be activated that will lift the left side of the bot above the ground, which help in the movement of left parts of the legs (first & third legs) in forward (or backward) direction with the help of first servo. Then left middle leg comes to ground & will lift right side of the bot which in the similar way help to move the right part of the bot in forward or backward direction. And when this process is done repeatedly it helps in movement of the whole bot either in forward or backward direction.

- **Principle behind sensor circuitry**
  Here we are using proximity sensor circuitry that can detect the incoming obstacle at a given predefined distance & help our hexapod robot not to hit the upcoming obstacle. We will be using one sensor at front end of our hexapod robot that is connected with main controller circuitry board which receives data from sensors that detect obstacles and then sends commands out to the central leg servo via Arduino to coordinate the movement of the hexapod. Diagram of sensor circuitry is given in **section 3.4.**

- **Principle behind host computer and handheld devices control with aurdino processor**
  - **Host computer program: -** The host computer program written for Arduino for using serial control, serial communication can be easily implemented. The program of handheld devices used to complete the serial communication.

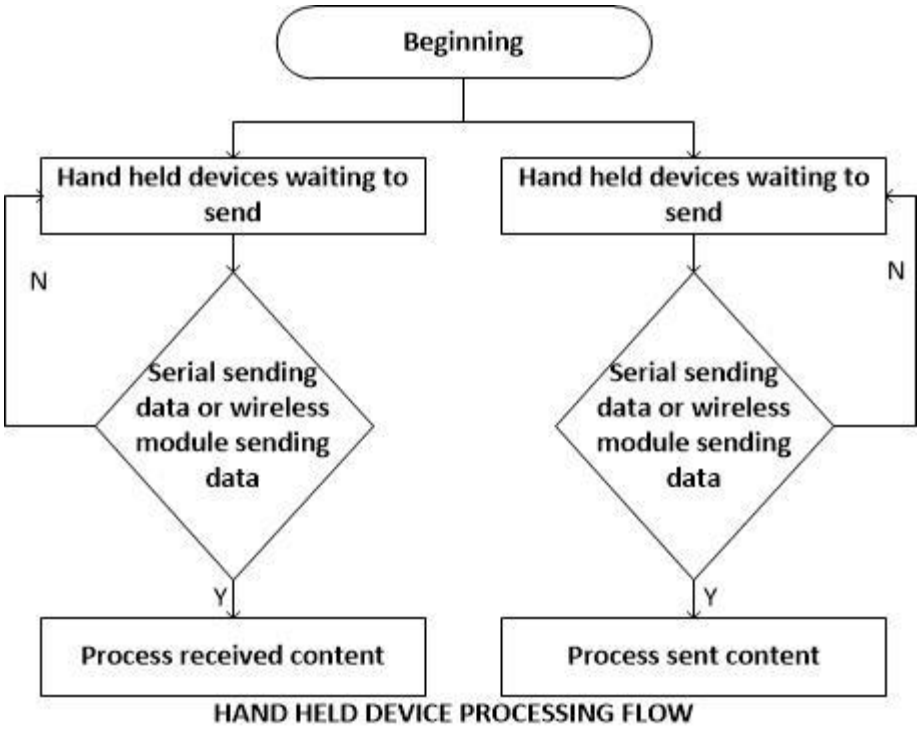


Figure 1(Host computer and handheld control flow diagram)
(http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6201877&tag=1)

The above flow diagram represents the logic behind the transfer of input & output messages from host computer to Arduino & helps us to run the program in Arduino that will indirectly controls the constraint movement of all 6 legs of hexapod robot in forward & backward direction.

  - **Bus control program: -** Bus is mainly used to transfer the gait control information as well as the legs movement of a variety of status information.

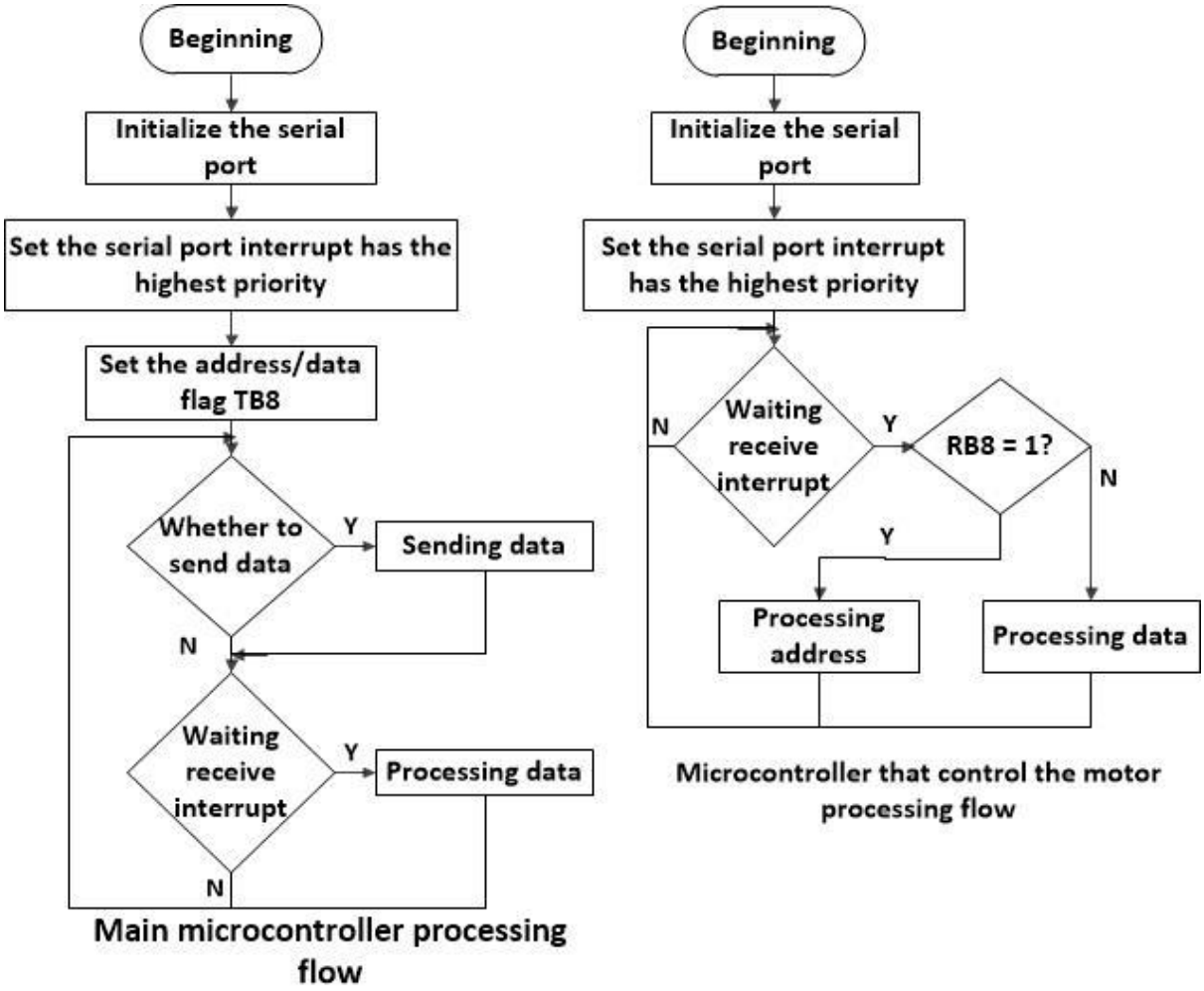


Figure 2(BUS CONTROL FLOW DIAGRAM)
(http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6201877&tag=1)

Arduino code for legs movement as well as sensor movement:-

```
#include <Servo.h>

// a maximum of three servo objects can be created

Servo myservo;                    // create servo object to control a servo

Servo myservo1;

Servo myservo2;

int pos = 0;                      // variable to store the left servo position

int pos1 = 0;                     // variable to store the right servo position

int middle = 0;                   // variable to store the middle servo position

void setup()

{

  myservo.attach(9);              // attaches the servo on pin 9 to the servo object

  myservo1.attach(11);            // attaches the servo on pin 11 to the servo object
```

```
 myservo2.attach(13);                    // attaches the servo on pin 13 to the servo object
}
void loop()
{
 for(pos = 75; pos < 105; pos += 1)          // goes from 0 degrees to 180 degrees
 {                                  // in steps of 1 degree
  myservo.write(pos);                    // tell servo to go to position in variable 'pos'
  delay(15);                             // waits 15ms for the servo to reach the position
 }
 for(pos = 105; pos>=76; pos-=1)             // goes from 180 degrees to 0 degrees
 {
  myservo.write(pos);                    // tell servo to go to position in variable 'pos'
  delay(15);                             // waits 15ms for the servo to reach the position
 }
 for(middle = 100; middle < 130; middle += 1)   // goes from 0 degrees to 180 degrees
 {                                  // in steps of 1 degree
  myservo2.write(middle);                // tell servo to go to position in variable 'pos'
  delay(15);                             // waits 15ms for the servo to reach the position
 }
 for(pos1 = 75; pos1 < 105; pos1 += 1)         // goes from 0 degrees to 180 degrees
 {                                  // in steps of 1 degree
  myservo1.write(pos1);                  // tell servo to go to position in variable 'pos'
  delay(15);                             // waits 15ms for the servo to reach the position
 }
 for(pos1 = 105; pos1>=76; pos1-=1)          // goes from 180 degrees to 0 degrees
 {
  myservo1.write(pos1);                  // tell servo to go to position in variable 'pos'
  delay(15);                             // waits 15ms for the servo to reach the position
 }
 for(middle = 130; middle>=101; middle-=1)     // goes from 180 degrees to 0 degrees
 {
  myservo2.write(middle);                // tell servo to go to position in variable 'pos'
```

```
  delay(15);                    // waits 15ms for the servo to reach the position
 }
}
```

3. **Circuit Diagram:-**
   - The overall circuit part and component of hexapod robot diagram are given below**:-**

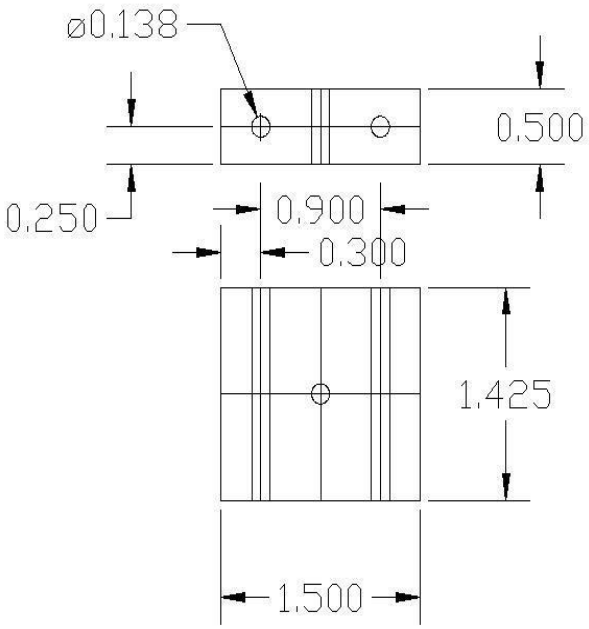


Figure 4(Dimensions of leg's)

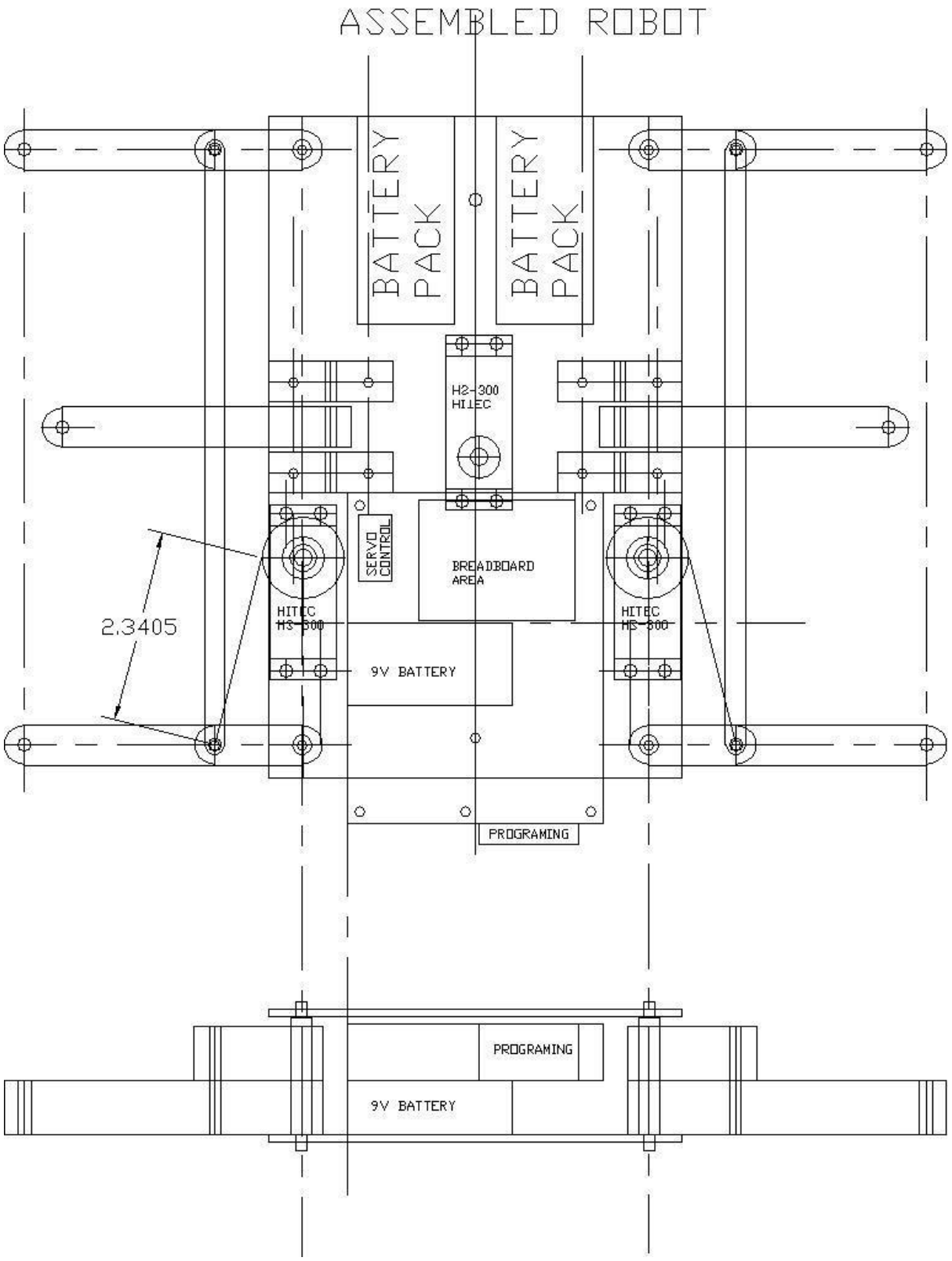


Figure 3 (Overall Front view of hexapod robot)

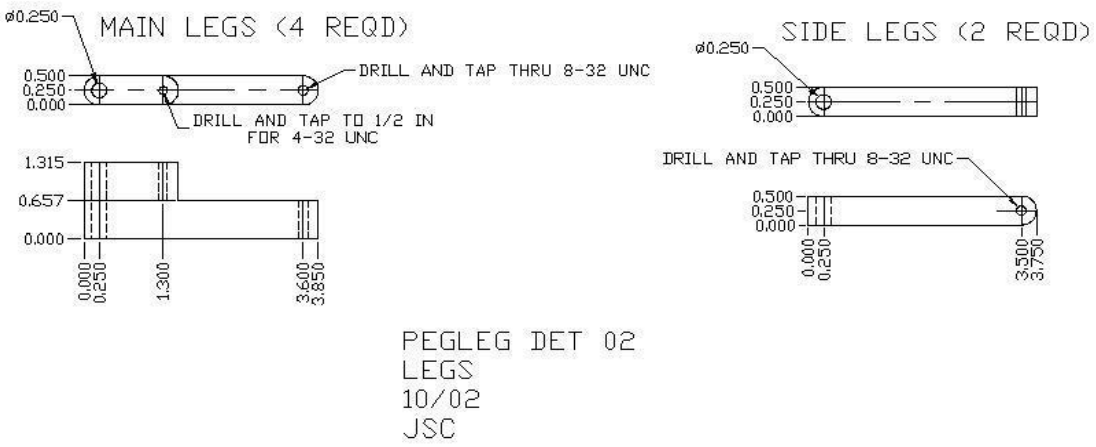
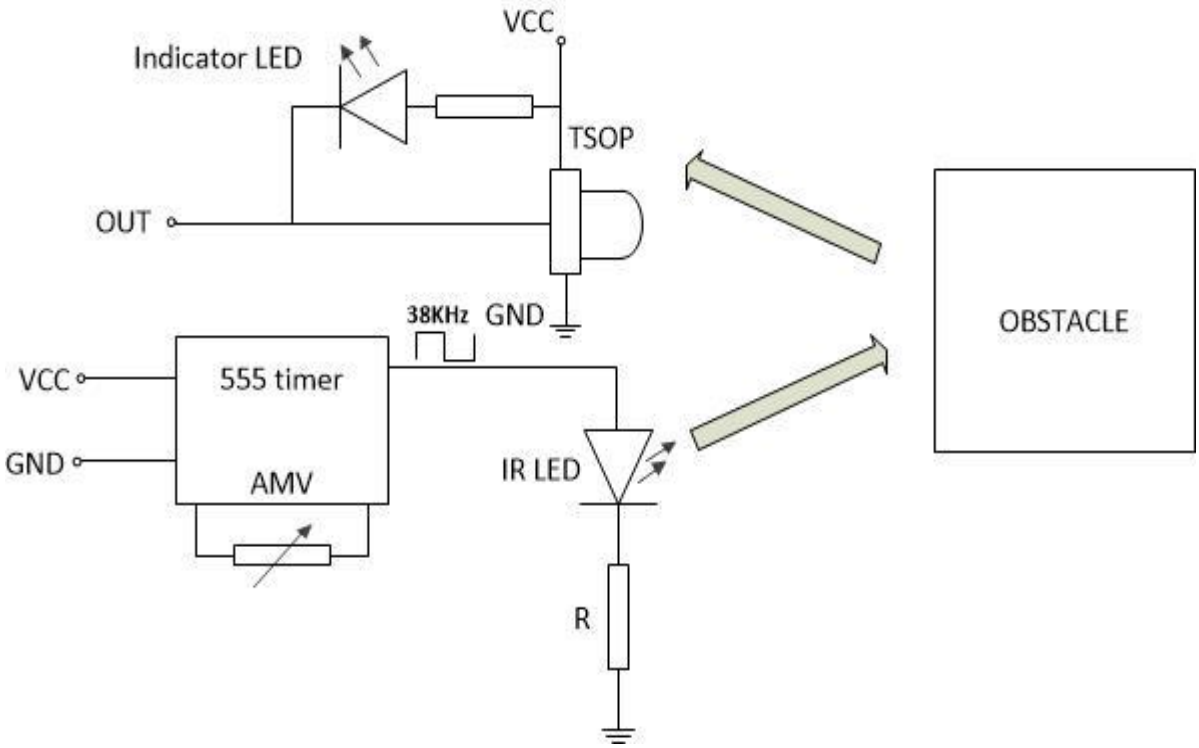


Figure 5(Front, Top and Bottom view of legs)

Circuit diagram of the sensor:-



4. Hardware Implementation:-

The overall look of our hexapod robot with circuitry being implemented will look somewhat like given below **figure 4**.

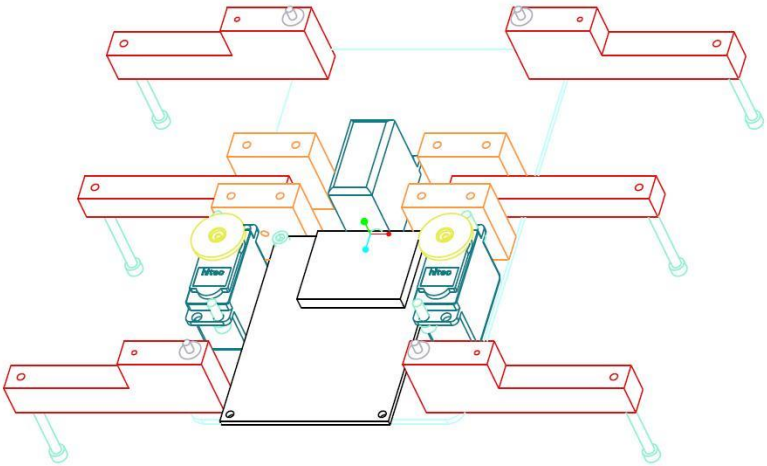


Figure 6(3D view of hexapod robot)

Here, our objective is to make 6 legged autonomous hexapod robots that can move forward & backward which is being controlled by the arduino. Arduino can control the robot walking forward & walking backward & Servo motors will provide necessary force for movement of the robot.

Weight, power & torque have been estimated that helps in deciding which type of servo motor is being used in our application.

The control circuitry helps in movement of all the six legs as being explained in 2.1 by "**Gait algorithm [3]**". Here we have used rod-shaped legs instead of eccentric type legs.

The sensor circuitry will help to tackle obstacle problem by indicating obstacles present in the front during movement to arduino that intern instructs the servos to stop movement till obstacle passes by.

COMPONENT DETAILS FOR HARDWARE IMPLEMENTATION:-

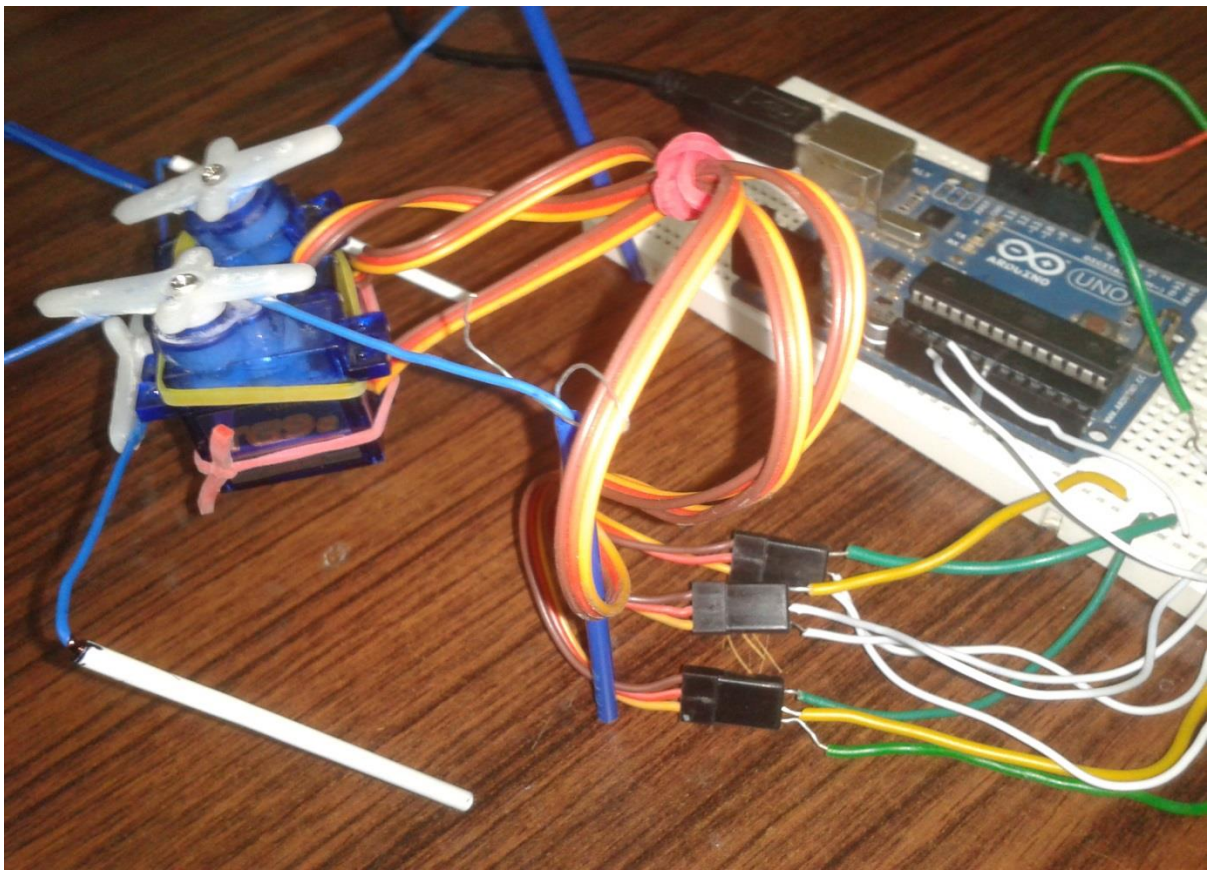
- **Legs**

  Here we are using 6 legs that are being implemented by paper clip & hence have only two degree of freedom.

- **Servo motor**

  Servos are DC motors with built in gearing and feedback control loop circuitry. And no motor drivers are required. Controlling a servo motors is done by sending a digital signal to the motor's control wire. The general idea is sending a square wave signal to the motor, where the wavelength of the wave should set the angle to which the motor will move. Here we are using 3 servo motors for forward & backward movement of the robot & its further functionality & principles is being explained in **section 2.1.**

- **Pin diagram of servo motor**:--



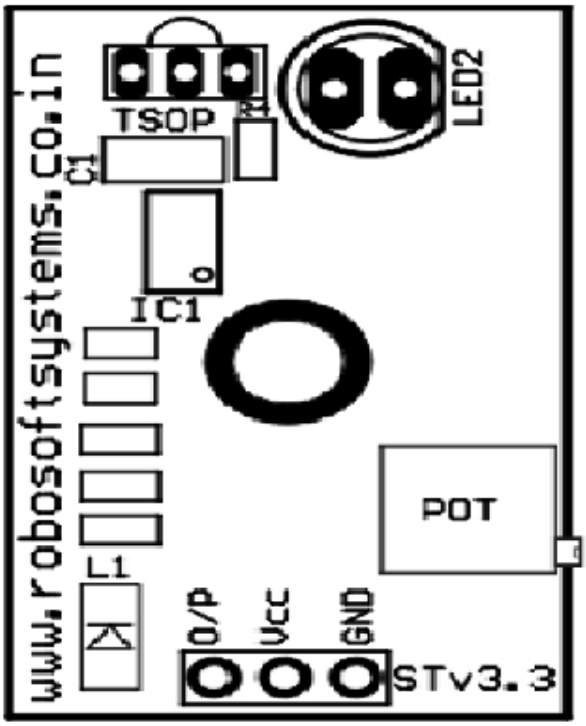
- **Pin configuration**:-

Pin No.	Connection	Description
1.	Power	Connect the red from servo to +5V on arduino
2.	ground	Connect the brown from servo to GND on arduino
3.	Signal	Connect orange from servo to digital on arduino

- **Sensor**

There are many sensors are available in market to detect any objects. Like SHARP IR sensor, NFRA RED and DIGITAL sensor etc. But here we have used proximity sensor that is a type of infrared sensor to detect the upcoming obstacle. Here we are using sensor circuitry that can detect the incoming obstacle at a given predefined distance & help our hexapod robot not to hit the upcoming obstacle. For further implementation is being explained in **section 2.2.**

➢ **Pin diagram of sensor**:-



(http://www.robosoftsystems.co.in/roboshop/media/catalog/product/pdf/dual-tsop)

➢ **Pin configuration**:-

Pin No.	Connection	Description
1.	Output	Digital Output(High or Low)
2.	VCC	Connected to circuit supply
3.	Ground	Connected to circuit ground

- **Arduino**

  The instructions that is to be feed to move the hexapod robot is being done in arduino processor in its own language. It is mainly used to transfer instruction for the movement of the leg in forward & backward direction with the help serial input output buses. The host computer program written for Arduino for using serial control, serial communication can be easily implemented. The program of handheld devices used to complete the serial communication. For further explanation have looks at **section 2.3 (2.3.1 & 2.3.2).**

- **The arduino that is being used in our project is given below**:-

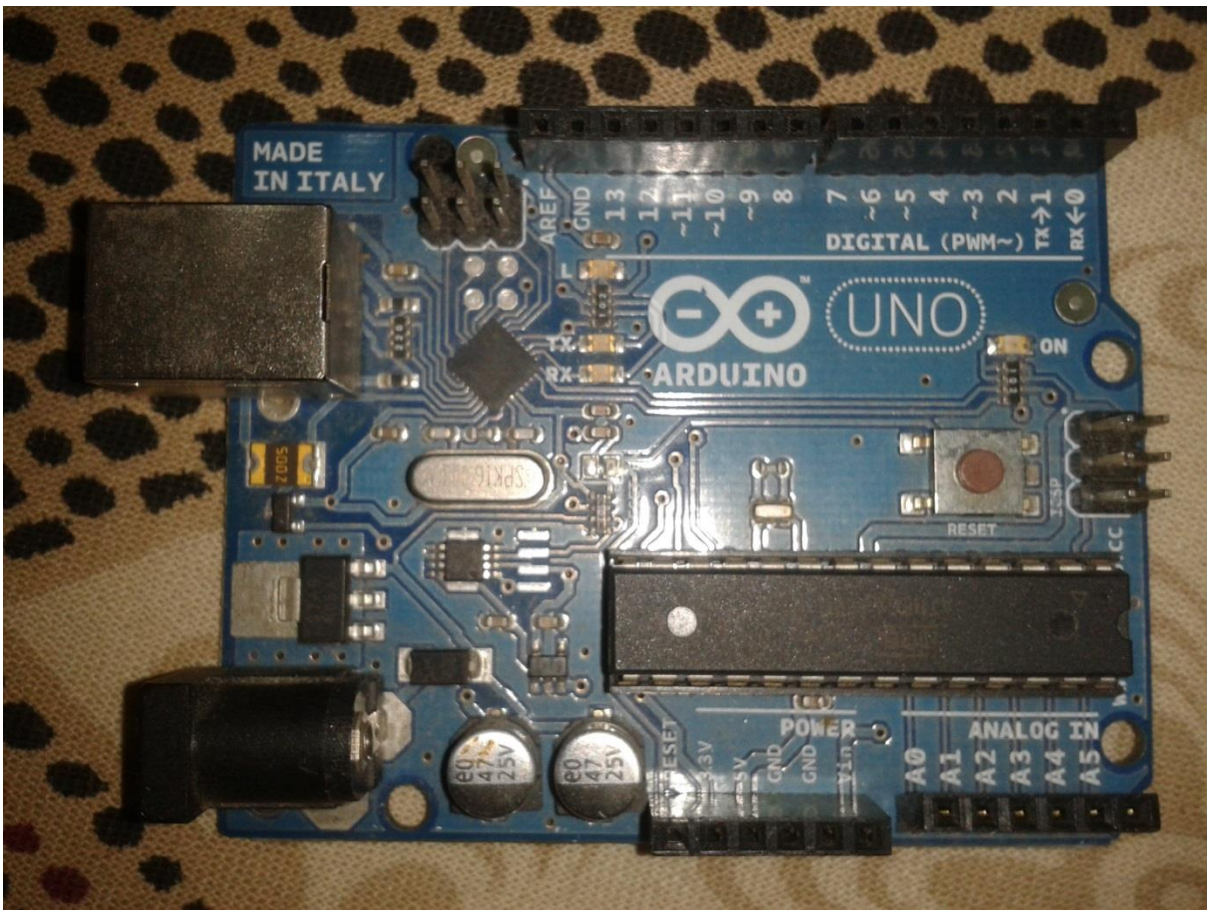
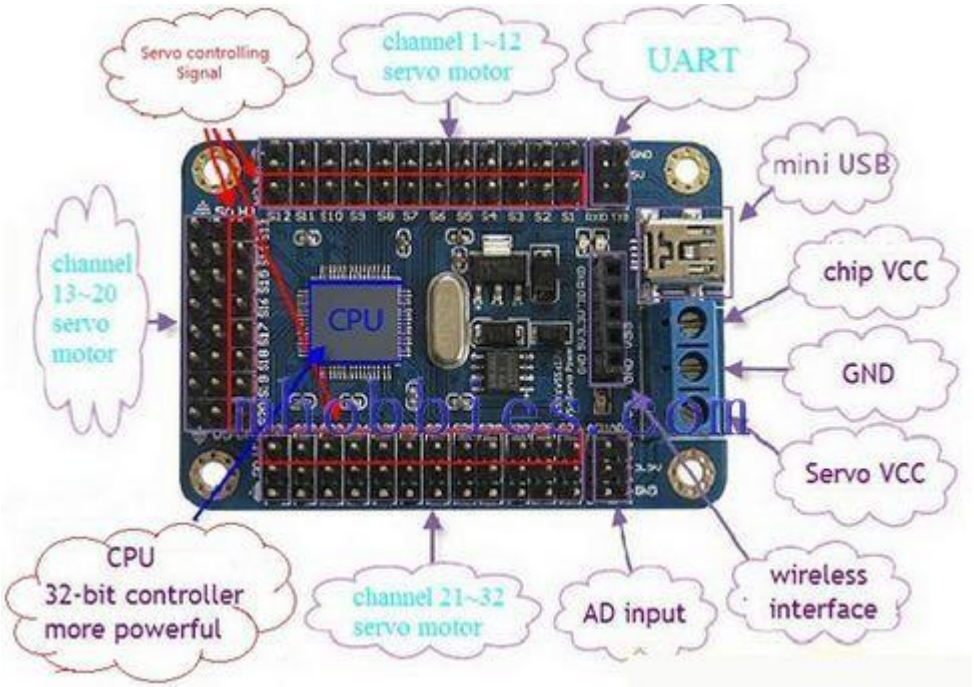


Figure7 (Arduino)

- **Pin digram of arduino**:--



- **Frame**
  The frame of the robot can be used as a plane board that are integrated together to connect all the six legs that are made of paper clips. We can take a larger plate that is used for mounting the electric equipment's such as the microcontroller boards, as well as there is extra space on the plate so additional electronic equipment and sensors could be added to the robot in the future, in accordance with the design requirements.

- **Li- ion battery with charger**
  We will also need Li-ion rechargeable battery because life of these batteries is very less & may be it gets exhausted even during the testing session.

5. **Results:-**

We are able to achieve following objectives of our project. They are:-

- ❖ We are able to implement the movement of all the six legs with the help of servos in forward & backward directions in synchronism with coordination.
- ❖ We also managed to improvise the movement of the servo attached to hexapod leg whose movement in the early stage looks very jerky.
- ❖ We have used paper clip to implement legs in our hexapod Robot as the cost of purchasing 6 legs kit was very high.
- ❖ We have implemented the circuitry for the sensors so that hexapod robot can detect obstacles within certain defined range.
- ❖ The complete algorithm for the implementation of movement of all six legs & sensors control is being implemented.

Conclusion:-

Following conclusion can be made from our development of the theory & circuit implementation:-

- ❖ While using a Li-ion rechargeable battery, handle it with very carefully as it might become risk to each component.
- ❖ We can reuse each and every broken component for something more relevant in the project.
- ❖ Practical implementation circuitry or any theoretical concept is much complex than its theoretical overview.
- ❖ The experimental results show that our design can achieve 6 legged robots to walk forward & backward & have a reliable performance. Through the experiment we can see its excellent performance in obstacle handling capacity.

Source or References from which we have taken help:-

1. IEEE journal on
   Hexapod robot. Mathematical support for modeling and control

2. IEEE journal on
   Fault-tolerant locomotion of the hexapod robot

3. **Dynamic crawl gait algorithm for quadruped robots**
   By Heeseon Hwang and Youngil Youm

4. Design of A Hexapod Robot

By Junke Li1, YujunWang2, TingWan3