BIPED

ROBOTICS CLUB OF CEG

Anna University, Guindy, Chennai-600025.

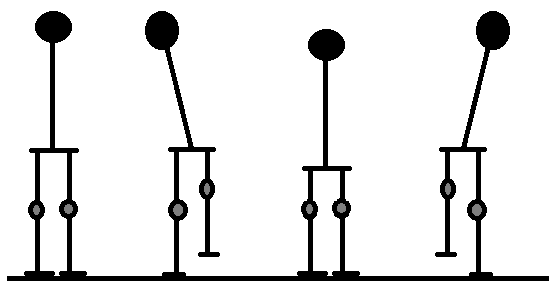
**Team members:**

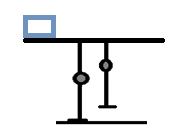
Vijay (Mech), Ramya P (Mech), Ashok Kumar S (Mech), Arul Muthusamy E (ECE).

***ABOUT***:

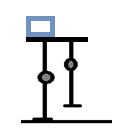
* Bipedalism is a form of locomotion where an organism moves by means of its two rear limbs or legs.
* Biped is an animal or ***machine*** that usually moves in the above (bipedal) manner.
* Our biped is intended to be autonomous robot. And designed to be interacting with its environment.
* Its functional purpose is to serve the experimental studies in various locomotion related to humans.
* Now the major task lies in understanding the human balance .It is nothing but controlling the body posture without falling in various actions made to move the same.
* The advantages of legs over wheels in a robot are that legged robots could navigate on any kind of surfaces. Whereas wheels are designed to work on prepared surfaces like rails, roads, terrain, etc.
* Legged robots can step over the objects that needed to be avoided in its path. But the wheeled one can’t.
* Wheeled bots could not make it over when the path has discontinuity, but the other one could cross it.
* And most of all the humanoid body structure allows it to act in human environments.
* The focus of this paper is to create a pair of Biped Robotic legs which can move autonomously, detecting obstacles and avoiding them or can be controlled by your own laptop or mobile through Bluetooth pairing.

***Design:***

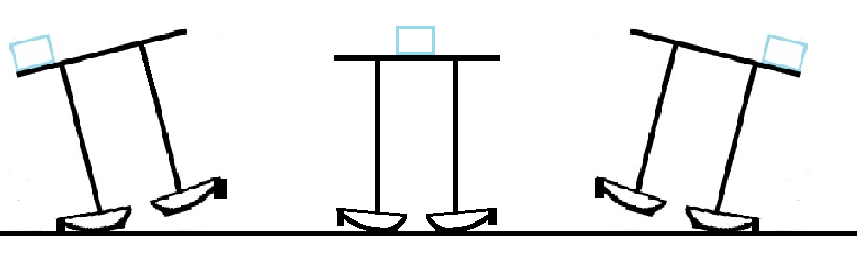
* **Design of biped includes equal amount of mechanical and electronics considerations.** Each leg of the robot has Hip and Ankle. The actuators used in the biped to control the joints are servos. They are small electric motors that provide position feedback control. The stability is achieved by moving these joints. The task is to maintain the center of mass within its area of support, and achieving it by using minimal number of actuators.
* Here in this biped we use simple U shaped brackets specially designed for this kind of bipeds.
* Both the hip and ankle joints undergo ROLLING actions. And anything doesn’t perform steering actions.
* This biped has 4 Degrees of freedom (D.O.F). The Degree of freedom of a body is defined as number of independent movement it has.
* So the biped has 4 joints in which it could make its movement.
* Next part is to make the biped stable when it lifts a leg off the ground.
* There are two methods to achieve this when it is in one leg phase.
* The first method is to tilt one leg at its ankle so that whole body tilts to one side and maintains the mass while the other leg moves front. And then it comes to rest to attain stability.
* The other method is more likely to the human. In humans half of the body above the pelvis region (hip) is moved above one leg when the other one swings to and fro. So a suitable mass should be moved above the two legs (on the hip) and change the centre of the gravity.
* The constraint in the first one is the tilt angle should be so accurate enough to lift one leg. If the angle is more it falls on the other side. And the foot pad should be larger.
* The second idea is like an inverted pendulum kept above the legs. So it needs more space above the hip. Since the weight is at far height from the ground, it is very hard to maintain the centre of mass.
* We decided to go with inverted pendulum idea and defy its constraint. 
* The main constraint is in moving the weight at that height makes the biped fall front or back during motions.
* So we concluded to make the distance between the weight and the legs zero and roll the weight above the legs on a platform (hip).



* In this case we need to move the weight far from the leg to maintain the balance.
* We need this:



* The weight should not move far from projection view of foot. But this is not possible with this kind of flat foot.
* So we decided to make changes in ankle joints and foot.
* Like this one:
* In this model the foot is curved like it is present in humans.



* Whenever the weight is displaced to one side the biped tilts automatically.

**Determining the Mechanical Constraints:**

There are various design considerations when designing a Bipedal robot. Among them, the major factors that have to be considered are:-

**1) Robot Size Selection:**

Robot size plays a major role. Based on this the Cost of the Project, Materials required for fabrication and the no of Actuators required can be determined. In this project miniature size of the robot is preferred so a height of 48cm is decided which includes mounting of the control circuits on top, but the actual size of the robot is 30cm without controlling circuits.

**2) Degrees of Freedom (D.O.F):**

Human leg has got Six Degrees of freedom (Hip – 3 D.O.F, Knee – 1 D.O.F, Ankle – 2 D.O.F), but implementing all the Six D.O.F is difficult due to increase in cost of the project and controlling of the actuators which become complex, so in this project reduced degrees of freedom is aimed so 4 D.O.F has been finalized (Hip – 2 D.O.F, Knee – 2 D.O.F).

**3) Link Design:**

In this project U-shaped bracket like arrangement called servo frames and flat brackets called servo clamps are used for various joints and connecting servos to the leg parts wherever needed. Servo frames used are of various lengths according to the various lengths of the different parts of the leg whereas the servo flat brackets will join the servo motors to the different joints.

**4) Stability:**

With Biped mechanism, only two points will be in contact with the ground surface. In order to achieve effective balance, actuator will be made to rotate in sequence and the robot structure will try to balance. If the balancing is not proper, in order to maintain the Centre of Mass, dead weight would be placed in inverted pendulum configuration with 1 D.O.F. This dead weight will be shifted from one side to the other according to the balance requirement. In this project such configuration is not used since the controlling units are mounted on the top and it acts as the dead weight. So the dead weight is actually the biped’s main control unit (Arduino+servo shield+ultrasonic sensor+Bluetooth module).

**5) Foot Pad Design:**

The stability of the robot is determined by the foot pad. Generally there is a concept that over sized and heavy foot pad will have more stability due to more contact area. But there is a disadvantage in using the oversized and heavy foot pad, because more material will be required leading to apply more torque for lifting the various leg parts and no significant contribution to the stability of the system. By considering this disadvantage an optimal sized foot pad was used.

**WALKING GAIT:**

Generally walking cycle consists of two steps namely Initialization and Walking.

NOTE: Every time the cycle is initiated only after reading the values from ultrasonic sensor.

The sensor indicates and stops the cycle if any obstacle is found in its way.

**1) Initialization:**

In the Initialization step the robot will be in balanced condition and in this step the servomotors are made to return to home position. This will certainly help the robot to advance into the next step.

**2) Walking:**

Walking step is further classified into **six** phases.

**Phase 1** – Double Support:

In this phase both the legs are in same line and the centre of mass is maintained between the two legs.

**Phase 2** – Single Support (Pre-Swing):

In this phase both the ankle joints are in actuated in roll orientation which shifts the centre of mass towards the left leg and the right leg will be lifted up from the ground.

**Phase 3** – Single Support (Swing):

In this phase, the right leg is lifted further and made to swing in the air. Hip and knee joints are actuated in pitch orientation so that right leg is moved forward.

**Phase 4** – Post Swing:

In this phase the lifted leg is placed down with the actuation of ankle joints. **Phase 5 and 6** are the mirror image of **Phase 2 and Phase 3**.

After Phase 6, motion continues with a transition to Phase 1 and the walking continues. It takes approximately 30 seconds to complete one walking cycle (all 6 phases). Bipedal robot has a step length of approximately 70mm. The Robot has the capability of carrying a dead weight of approximately 150gms.

**CONTROLLING OF BIPEDAL ROBOT**

Generally any robot has a combination of motors and sensors, which are controlled by microcontrollers. There are wide varieties of motors, sensors and microcontrollers available. In this project Arduino UNO is used as a microcontroller and for 4 D.O.F, each D.O.F has one servomotor. The Bluetooth module is used to make transmission of signal from phone to the arduino wireless.

**TASKS IT CAN PERFORM**

* WALK IN FORWARD, BACKWARD.
* KICK ANY OBJECT IN FRONT OF IT.
* HEAD BUTT ANY OBSTACLE.
* DETECT OBSTACLES SUCH AS WALL AND AVOID THEM WHEN IT OPERATES AUTONOMOUSLY.

**TASKS IT CAN BE AIMED FOR**

* IF FALLEN IT CAN GET UP ON ITS OWN.
* WALK VERY FAST.
* CLIMB STAIRS
* A FULL HUMANOID ROBOT WITH HANDS AND A HEAD CAPABLE OF IMAGE PROCESSING, PICKING UP THINGS, ETC.

***Components:***

***SERVO MOTOR:***



**TOWER PRO mg – 945**

Specifications:

• Weight: 55 g

• Dimension: 40.7 x 19.7 x 42.9 mm approx.

• Stall torque: 8.5 kgf·cm (4.8 V ), 10 kgf·cm (6 V)

• Operating speed: 0.2 s/60º (4.8 V), 0.16 s/60º (6 V)

• Operating voltage: 4.8 V a 7.2 V

• Dead band width: 5 µs

• Stable and shock proof double ball bearing design

• Temperature range: 0 ºC – 55 ºC

**ARDUINO:**



**Overview**

* 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 (ATmega328) of which 0.5 KB used by bootloader
* SRAM 2 KB (ATmega328)
* EEPROM 1 KB (ATmega328)
* Clock Speed 16 MHz

***Ultrasonic Ranging Module HC - SR04***



**Product features:**

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work: (1) Using IO trigger for at least 10us high level signal, (2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back. (3) 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,

Wire connecting direct as following:

\*5V Supply

\*Trigger Pulse Input

\*Echo Pulse Output

\* 0V Ground

**Electric Parameter:**

* Working Voltage DC 5 V
* Working Current 15mA
* Working Frequency 40Hz
* Max Range 4m
* Min Range 2cm
* Measuring Angle 15 degree

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 trigger signal and receiving echo signal.

**Formula:** uS / 58 = centimeters or uS / 148 =inch; or: the range = high level time \* velocity (340M/S) / 2.

***Bluetooth module HC-05***



HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature).

**CODE:**

|  |  |  |
| --- | --- | --- |
| #include <Servo.h>  Servo a1,a2,b1,b2,c1;  int x[20];  String z;  char y;  int d=90;  int h=90;  int s=0,i=0,j=90,t=90;  int k=0,a=0,l=0,p=0,f=1,q=0,o=0;  long duration, distance;  void setup()  {  Serial.begin(9600);  a1.attach(11);  a2.attach(10);  b1.attach(9);  b2.attach(5);  c1.attach(3);  a1.write(90);  a2.write(90);  b1.write(90);  b2.write(90);  c1.write(95);  pinMode(12, OUTPUT);  pinMode(8, INPUT);  pinMode(2,OUTPUT);  digitalWrite(2,HIGH);  } | void loop()  {  if(Serial.available()>0)  {  z=Serial.readStringUntil('#');  y= z[z.length()-1];  }  digitalWrite(12, LOW);  delayMicroseconds(2);  digitalWrite(12, HIGH);  delayMicroseconds(10);    digitalWrite(12, LOW);  duration = pulseIn(8, HIGH);  distance = duration/58.2;  if(distance>=25||distance==0)  {  y=y;  }  else  {  y='s';  }  if(y=='f')//FORWARD  { | if(s==0)  {  // delay(5000);  c1.write(105);  delay(600);  while(k==0)  {  c1.write(105);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);  k=0;  }  for(i=90;i<=105;i+=1)  {  a1.write(i);  delay(30);  a2.write(j);  j=j-1;  delay(30);  }  delay(100); |
| for(i=90;i<=105;i+=1)  {  b1.write(i);  delay(30);  b2.write(t);  t=t-1;  delay(30);  }  delay(100);  c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(200);  k=0;  c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7); | if (a==0)  {  k=1;  }  }  c1.write (95);  delay(100);  k=0;  x[0]=a1.read();  x[1]=a2.read();  x[2]=b1.read();  x[3]=b2.read();  for(i=x[0];i>=90;i-=1)  {  a1.write(i);  delay(30);  a2.write(x[1]);  x[1]=x[1]+1;  delay(30);  }  delay(100);  for(i=x[2];i>=90;i-=1)  {  b1.write(i);  delay(30);  b2.write(x[3]);  x[3]=x[3]+1;  delay(30);    }  delay(100); | for(i=90;i>=75;i-=1)  {  b1.write(i);  delay(30);  b2.write(d);  d=d+1;  delay(30);  }  delay(100);  for(i=90;i>=75;i-=1)  {  a1.write(i);  delay(30);  a2.write(h);  h=h+1;  delay(30);  }  c1.write(105);  delay(400);  while(k==0)  {  c1.write(105);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  k=0; |
| c1.write(95);  delay(200);  c1.write(105);  delay(400);  while(k==0)  {  c1.write(105);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  k=0;  c1.write(95);  delay(100);  x[4]=a1.read();  x[5]=a2.read();  x[6]=b1.read();  x[7]=b2.read();  for(i=x[6];i<=90;i++)  {  b1.write(i);  delay(30);  b2.write(x[7]);  x[7]=x[7]-1;  delay(30);  } | delay(100);  for(i=x[4];i<=90;i++)  {  a1.write(i);  delay(30);  a2.write(x[5]);  x[5]=x[5]-1;  delay(30);  }  delay(100);  /\*c1.write(83);  delay(400);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);\*/  k=0;  d=90;  h=90;  t=90;  j=90;  s=1;  l=1;  f=0;  q=1;  } | //delay(50);  else if(y=='s')//STOP  {  if(l==1)  {  delay(100);  c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);  k=0;  }  f=1;  p=0;  s=0;  l=0;  o=0;  q=0;  } |
| else if(y=='k')//KICK  {  if(f==1)  {  k=0;  c1.write(105);  delay(400);  while(k==0)  {  c1.write(105);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  k=0;  c1.write(95);  delay(200);  }  f=0;  if(p==0)  {  /\*c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);  k=0;\*/  x[0]=a1.read();  x[1]=a2.read(); | for(i=x[0];i<=145;i++)  {  a1.write(i);  delay(30);  a2.write(x[1]);  x[1]=x[1]-1;  delay(30);  }  delay(100);  //  x[0]=a1.read();  x[1]=a2.read();  for(i=x[1];i>=30;i--)  {  a2.write(i);  delay(30);  }  delay(100);  x[0]=a1.read();  x[1]=a2.read();  for(i=x[0];i>=100;i--)  {  a1.write(i);  delay(30);  }  delay(100);  x[2]=b1.read();  x[3]=b2.read();  for(i=x[2];i<=105;i++)  {  b1.write(i);  delay(60);  b2.write(x[3]);  x[3]=x[3]-1;  delay(60);  }  delay(100); | x[2]=b1.read();  x[3]=b2.read();  x[0]=a1.read();  x[1]=a2.read();  for(i=x[0];i>=55;i--)  {  a1.write(i);  delay(30);  }  delay(100);  x[0]=a1.read();  x[1]=a2.read();  for(i=x[0];i<=100;i++)  {  a1.write(i);  delay(30);  }  delay(100);  for(i=x[2];i>=90;i--)  {  b1.write(i);  delay(60);  b2.write(x[3]);  x[3]=x[3]+1;  delay(60);  }  x[0]=a1.read();  x[1]=a2.read();  delay(200);  for(i=x[0];i<=145;i++)  {  a1.write(i);  delay(30);  }  delay(100);  x[0]=a1.read();  x[1]=a2.read(); |

|  |  |  |
| --- | --- | --- |
| for(i=x[1];i<=35;i++)  {  a2.write(i);  delay(30);  }  delay(100);  x[0]=a1.read();  x[1]=a2.read();  for(i=x[0];i>=90;i-=1)  {  a1.write(i);  delay(30);  a2.write(x[1]);  x[1]=x[1]+1;  delay(30);  }  delay(100);  //  c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);  k=0;  p=1;  l=0;  delay(50);  }  } | else if(y=='l')  {  if(q==1)  {  c1.write(83);  delay(800);  while(k==0)  {  c1.write(83);  a=digitalRead(7);  if(a==0)  {  k=1;  }  }  c1.write(95);  delay(100);  k=0;  }  q=0;  delay(100);  if(o==0)  {  x[2]=b1.read();  x[3]=b2.read();  for(i=x[2];i<=105;i+=1)  {  b1.write(i);  delay(30);  b2.write(x[3]);  x[3]=x[3]-1;  delay(30);  }  delay(3000);  x[2]=b1.read();  x[3]=b2.read(); | for(i=x[2];i>=90;i--)  {  b1.write(i);  delay(30);  b2.write(x[3]);  delay(30);  x[3]=x[3]+1;  }  delay(100);  }  o=1;  }  } |