Leggy, the Legged Robot

by Natalia Perez

December 8, 2020

Description of the product:

Legged robots are used to reach out to places where usually 4-wheels cars cannot go. I am interested in working with robotics and this project is the combination between my curiosity for Internet of Things (IoT) and linkages design. The design was inspired by a product created by Wit Machine. My system is able to walk in different directions, its angular speed and direction can be controlled using the phone. In further applications, it is possible to install a camera in the front of the robot for exploration and it could also be a great toy for kids.

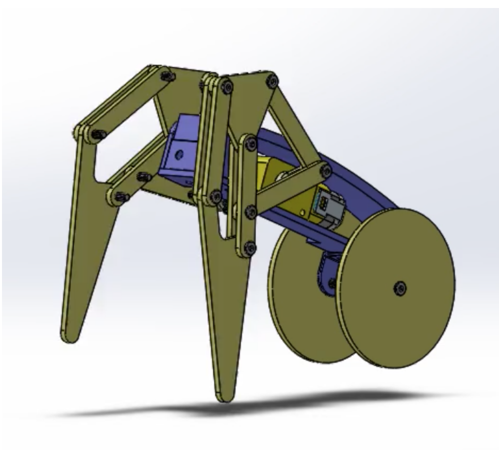

Figure 1: Legged robot CAD

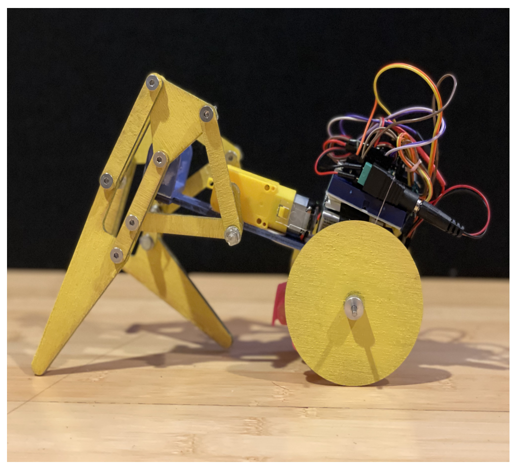

Figure 2: Legged robot final product

Electromechanical details:

Linkages:

All linkages were laser cut and the hinges where 3D printed as shown in Figure 3. I put everything together using M4 screws, nuts, and washers. The wheels are connected using a shaft and two collars as shown in Figure 6. All of the above constitutes the body of the robot as well as the housing for the electronics (Figure 2). Notice in Figure 5 that it was necessary to add hot glue to the end of the nuts to prevent them from falling off or tighten up the mechanism, due to the high vibrations of the robot.

DC motor - legs

The DC motor is stick to the center of mass of the vehicle and it is connected to the legs using the Motor Connectors (Figure 3) as shown in Figure 4 and 5. The Motor connectors are placed parallel to each other to allow the robot to create the walking movement when the shaft starts rotating.

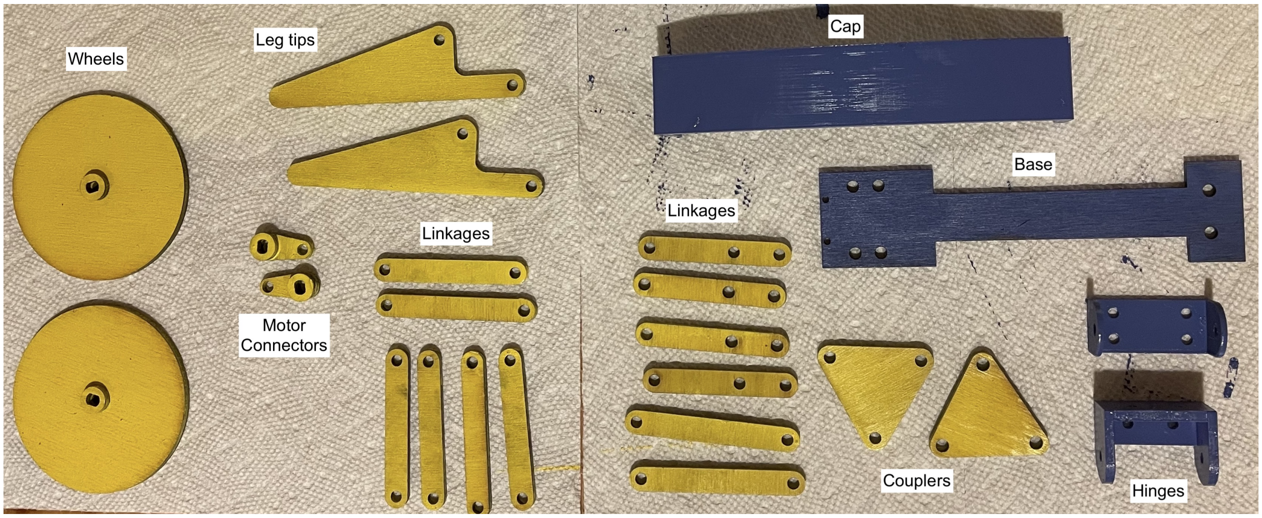

Figure 3: Legged robot parts

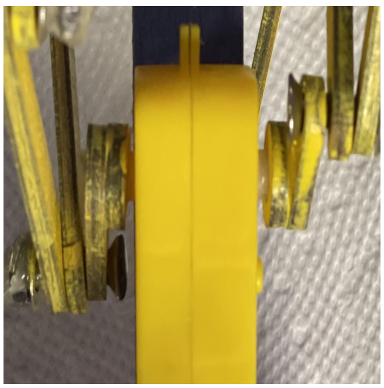

Figure 4: Top view of DC motor and Motor Connectors.

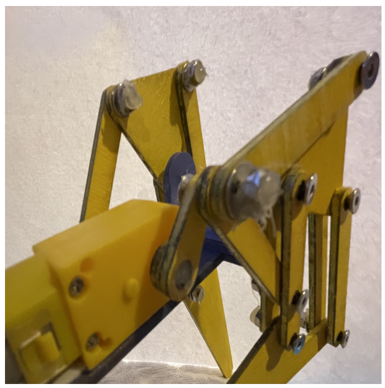

Figure 5: Side view of DC motor and Motor Connectors.

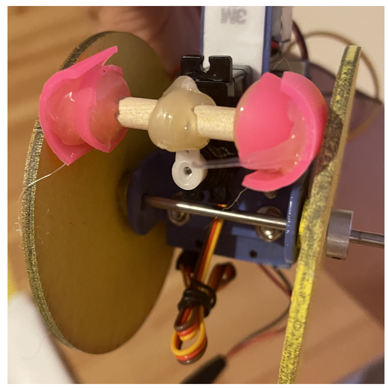

Figure 6: Directional system & wheels

Servo motor - direction

After I had the legged robot walking, I noticed that it was necessary to add a way to control the direction. As I was short in time to manufacture a shaft that could control direction using the wheels. I decided to use a servo motor connected to a wood stick and some silicon ends made of finger protectors (Figure 6). The way the direction control works is by moving the servo motor between 50° and 100° and this will allow the silicon end to push the wheel in the desired direction while the robot is walking.

App interface

Blynk is a IoT platform that allowed me to control the legged robot with my phone using WiFi and the created interface is shown in Figure 7. The first feature that I implemented was an on/off "Button." Additionally, the direction of the servo can be controlled by the second feature "DIRECTION LEFT/RIGHT" where 50° would allow the robot to move to the left and 100° to the right. The initial position of the servo motor in a straight path is 90°. Then, I mapped the maximum and minimum angular speed that the robot needs to walk using the third feature "MOTOR SPEED" as a potentiometer. The maximum and minimum pulse-width modulation (PWM) commands are 250 and 200, respectively.

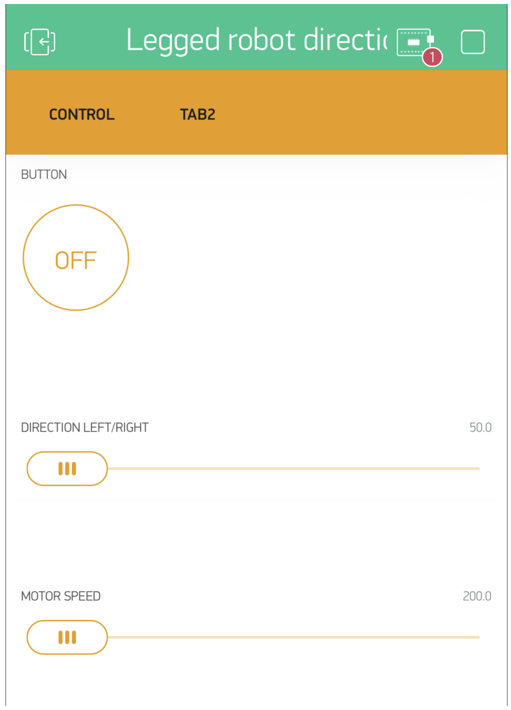

Figure 7: Blynk interface & controls for the legged robot

Circuit:

There are four main components for the circuit:

1. ESP32: The ESP32 (Assembled Adafruit HUZZAH32 – ESP32 Feather Board - with Stacking Headers, $21.95) came with the Microkit and it is the microcontroller for the system. It controls all the electronic components of the system and allows the connection to the phone.

2. Dual motor driver: The DRV8833 Dual Motor Driver Carrier (DRV8833 Dual Motor Driver Carrier, $4.95) allows to control the direction and angular velocity of the DC motor via PWM signal from the ESP32.

3. Servo motor: I had this old servo motor (Lewansoul 9G Servo Micro servo Metal Gear Motor Anti Blocking servo (1PCS), $8.99) to change the direction of the wheels so that the legged robot can go to different direction.

4. Double shaft DC motor: I purchased this double shaft DC motor (Antrader Gear Motor Dual Shaft 3-6V TT Motor for Smart Car Robot Pack of 6 (I Shape), $1.50/each) to be able to move the legs of the robot.

In addition to these parts, I also purchased some small breadboard (MCIGICM 6PCS 170 tie-Points Mini Breadboard for Arduino , $0.99/each) to have a more compact circuit. The circuit is powered by a 9V battery that is connected to the motor driver and a 3.7V Lithium Ion Polymer (LiPo) battery to power the microcontroller and the servo motor.

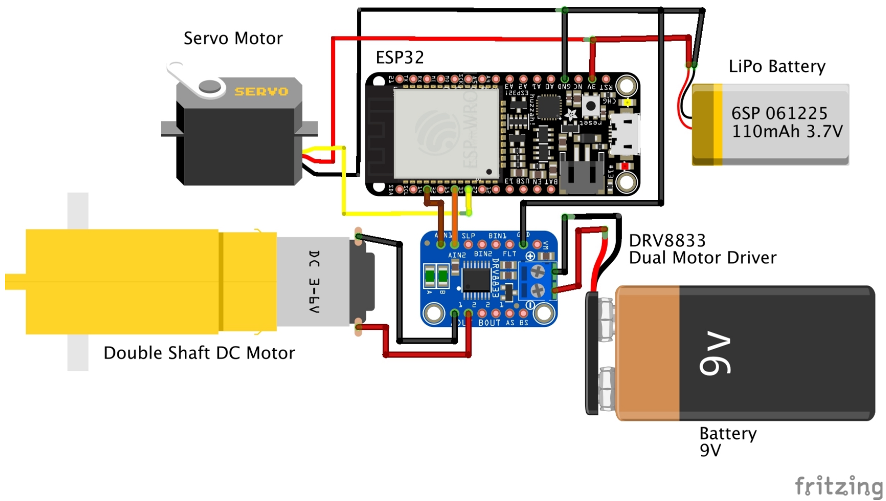

Figure 8: Circuit diagram for the system

Finite State Machine:

The system can be moving and changing direction at the same time. Therefore, "Legs moving" and "Turning" left/right are independent of each other. The directional mechanism depends on position (angle) input and the legs moving depend on the PWM input.

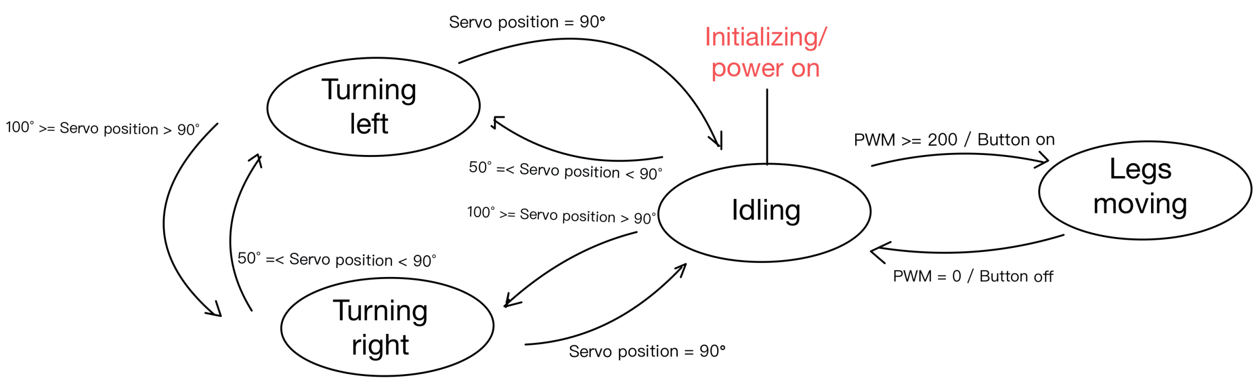

Figure 9: Finite State Machine

For the complete Arduino IDE code, please refer to Appendix I.

Appendix I: Arduino code

```
/* Leggy, the legged robot - Control code
   by Natalia Perez
   December 8, 2020
*/

//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LIBRARIES %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <WiFiClient.h>
#include <Wire.h>
#include <BlynkSimpleEsp32.h>
#include <ESP32Servo.h>

//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% WIFI/BLYNK VARIABLES %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
char auth[] = "CgU92_NQcTg1dp8jxiuJ0Rxltost8-N0";
char ssid[] = "Oxford WiFi-guest";
char pass[] = "SOMEPASSWORD";

BlynkTimer timer;

//%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CONSTANTS / VARIABLES %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
// OUTPUT
const int MOTOR_AIN1 = 32;
const int MOTOR_AIN2 = 33;

const int PWM_FREQUENCY = 30000;
const int PWM_RESOLUTION = 8;
const int LED_CHANNEL1 = 1;
const int LED_CHANNEL2 = 2;

// PWM - MOTOR VARIABLES
int motorSpeed_PWM = 0;

// SERVO VARIABLES:
Servo servo;
int servoPin = 27;
int servoVal = 0;
int minimum_speed = 200;

//%%%%%%%%%% DECLARING CONNECTIONS BETWEEN BLYNK APP AND MICROCONTROLLER %%%%%%%%%%
//ON/OFF BUTTON
BLYNK_WRITE(V1) {
  switch (param.asInt()) {
    case 1:
      ledcWrite(LED_CHANNEL1, minimum_speed);
      ledcWrite(LED_CHANNEL2, 0);
      break;
    default:
      Blynk.virtualWrite(V3, minimum_speed);
      ledcWrite(LED_CHANNEL1, 0);
      ledcWrite(LED_CHANNEL2, 0);
  }
}

//SERVO MOTOR
BLYNK_WRITE(V2) {
  servo.attach(servoPin);
  servoVal = param.asInt();
```

```
  servo.write(servoVal);
  delay(400);
  servo.detach();
}

//DC MOTOR
BLYNK_WRITE(V3) {
  motorSpeed_PWM = param.asInt();
  ledcWrite(LED_CHANNEL1, 0);
  ledcWrite(LED_CHANNEL2, motorSpeed_PWM);
}


//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% SETUP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
void setup() {
  //SERIAL
  Serial.begin(115200);
  Serial.println("Initializing..............");

  //PWM FUNCTIONALITIES
  // assign motor pin to channel
  ledcAttachPin(MOTOR_AIN1, LED_CHANNEL1);
  ledcAttachPin(MOTOR_AIN2, LED_CHANNEL2);
  // Initialize channel
  ledcSetup(LED_CHANNEL1, PWM_FREQUENCY, PWM_RESOLUTION);
  ledcSetup(LED_CHANNEL2, PWM_FREQUENCY, PWM_RESOLUTION);

  //WIFI FUNCTIONALITIES
  WiFi.disconnect();
  delay(10);
  Blynk.begin(auth, ssid, pass);
  Serial.print( "Connected " );
  Serial.println( WiFi.localIP() );
}


//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MAIN LOOP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  //BLYNK CONNECTIVITY COMMANDS
  if (Blynk.connected()) {
    Blynk.run();
    timer.run();
  }
  //PRINTING VALUES FOR DEBUGGING
  Serial.print("Connected ");
  Serial.println( WiFi.localIP());
  Serial.print("STATUS");

  Serial.print("PWM  ");
  Serial.print(motorSpeed_PWM);
  Serial.println("....");

  Serial.print("Servo  ");
  Serial.print(servoVal;
  Serial.println("....");
}
```