THE QUAD POD
THE TRANSFORMABLE VEHICLE

Team 6
Kee Woong Haan
Jiwon Park
Zenon Son

Department of Electrical and Computer Engineering

TA: Rajarshi Roy

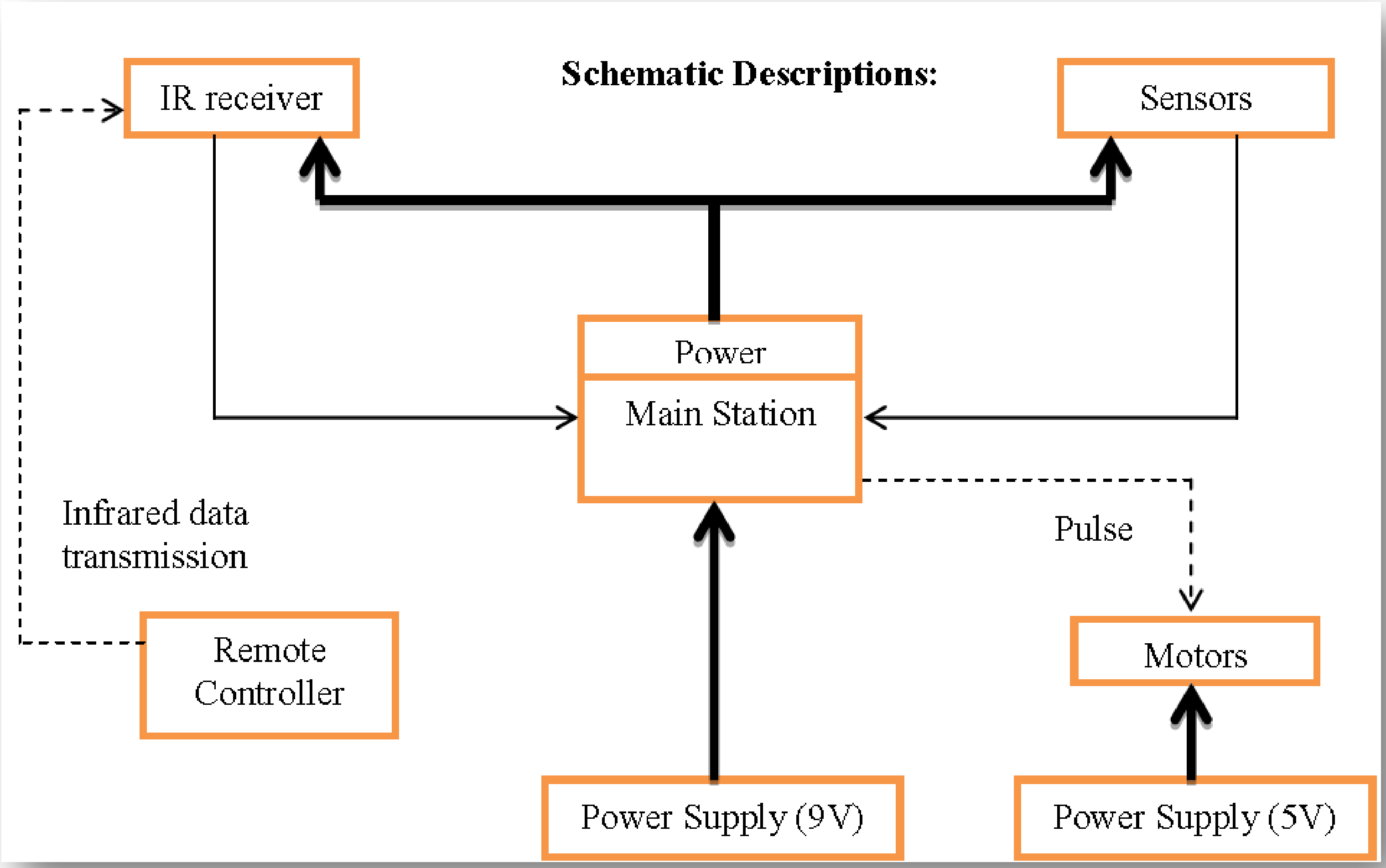
The Outline

- Introduction
- Block Diagram
- Mechanical and General Designs
- Electrical Parts
- The codes for the Quadpod
- Challenges and Future Works

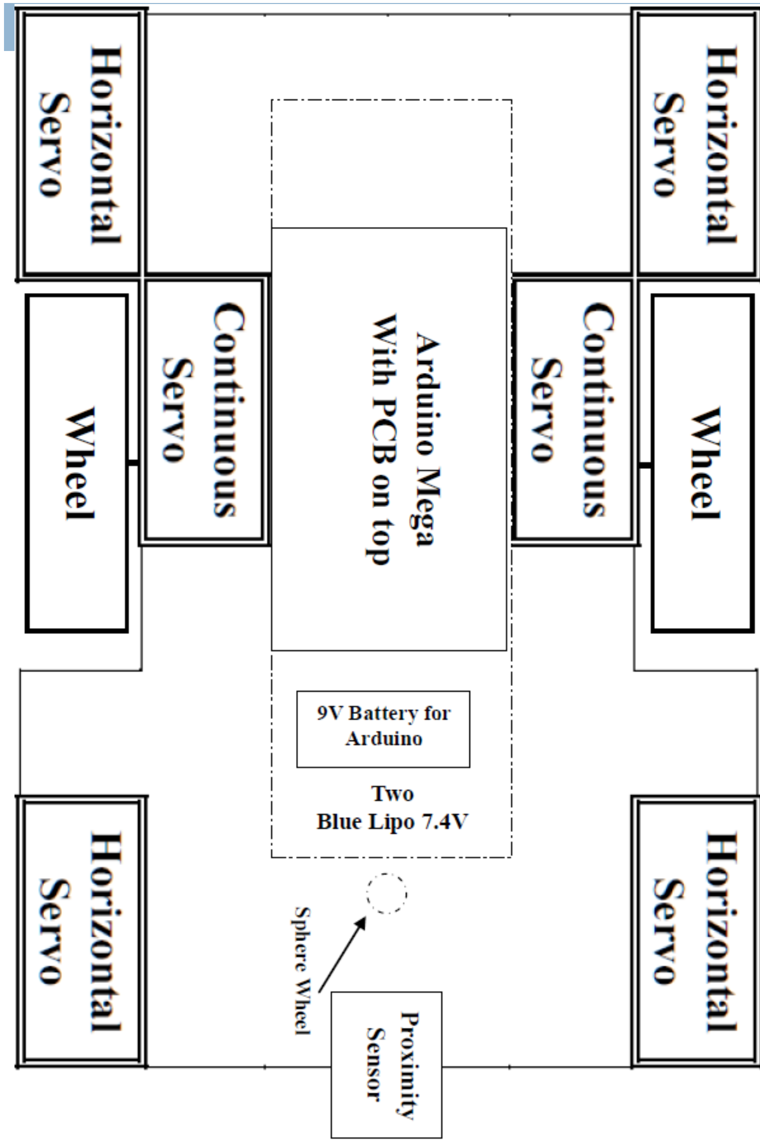
Introduction

- The transformable vehicle with two wheels and four legs
- The vehicle that can detect obstacles to transformed into the Quadpod mode
- The vehicle fully controlled by a remote controller
- The miniature model for the new type of vehicle for unpaved, bumpy roads

Block Diagram



Main Frame



6 X 8 inch frame size

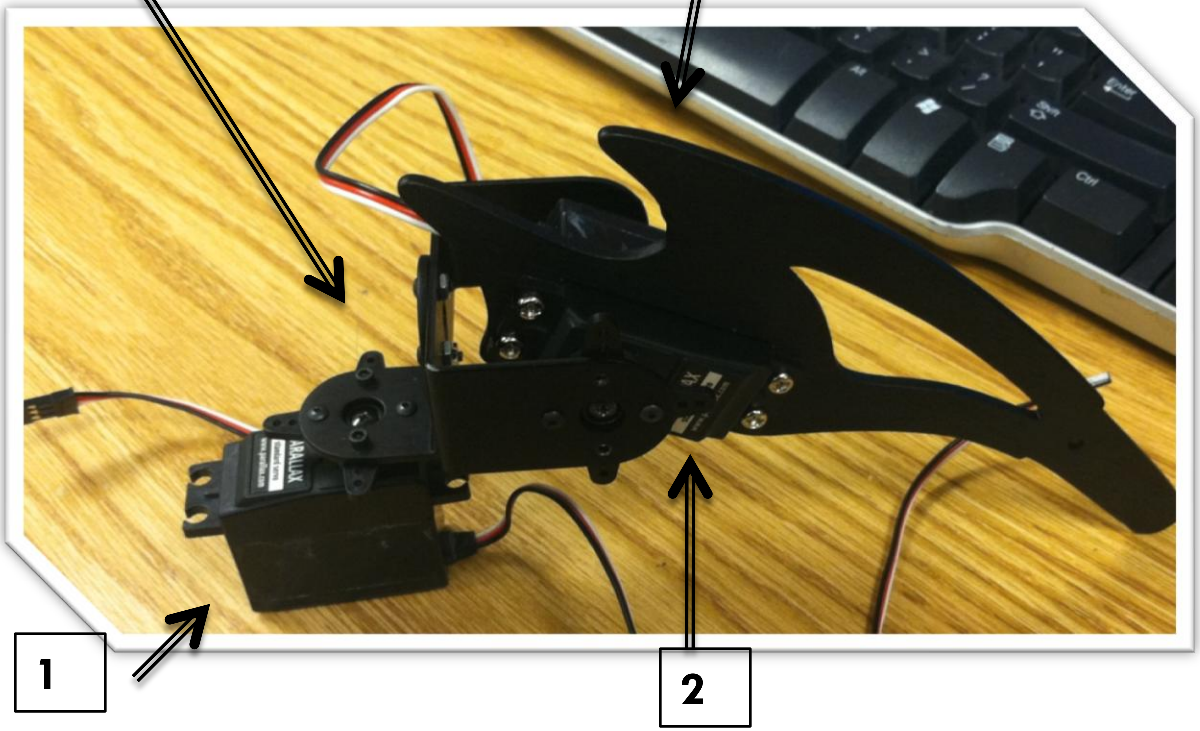
Top Side
- 4 Standard servo
- 2 Continuous servo
- Proximity Sensor
- 9V battery
- Arduino Mega with PCB

Bottom Side
- 2 Blue Lipo 7.4V
- Sphere Shape Wheel

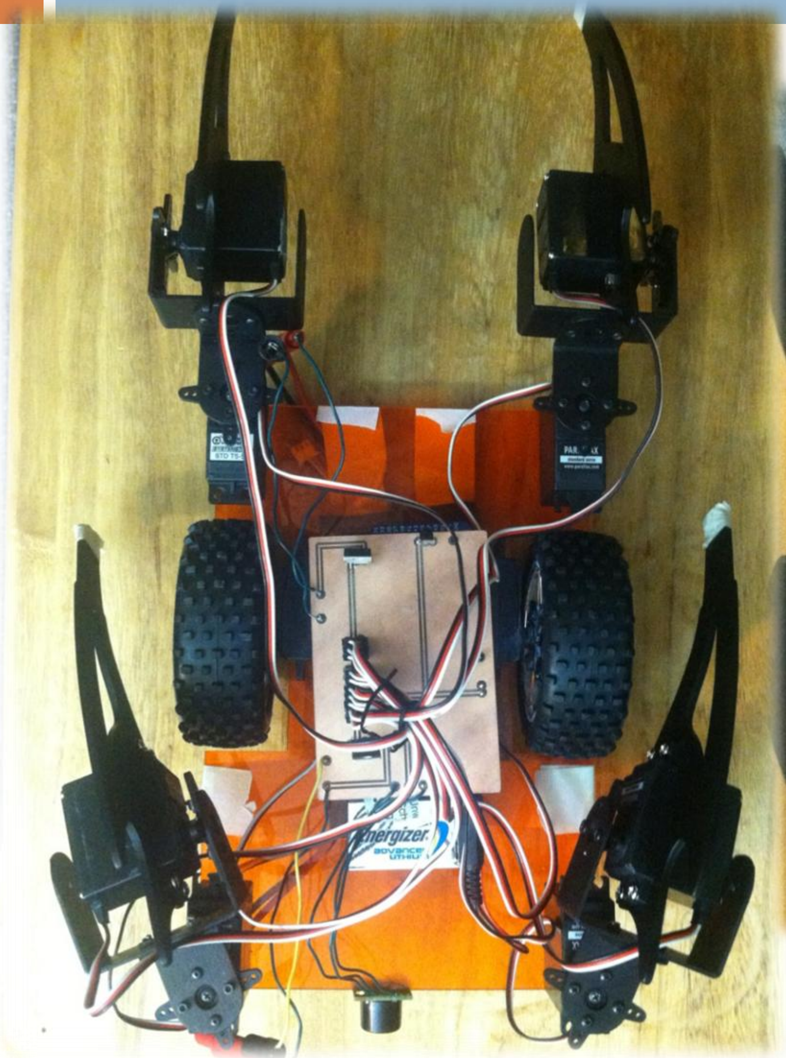
Leg Frame

- 2 Standard Servo
- Angle Converter

- Leg Part



Add Up



Blue Lipo Battery

Voltage: 7.4V

Max Capacity: 1.5Ah

	Theoretical	Actual
Voltage	7.4V	7.38V
Current Capacity	1500mAh	1470mAh
Running time for our Project	90min	80min

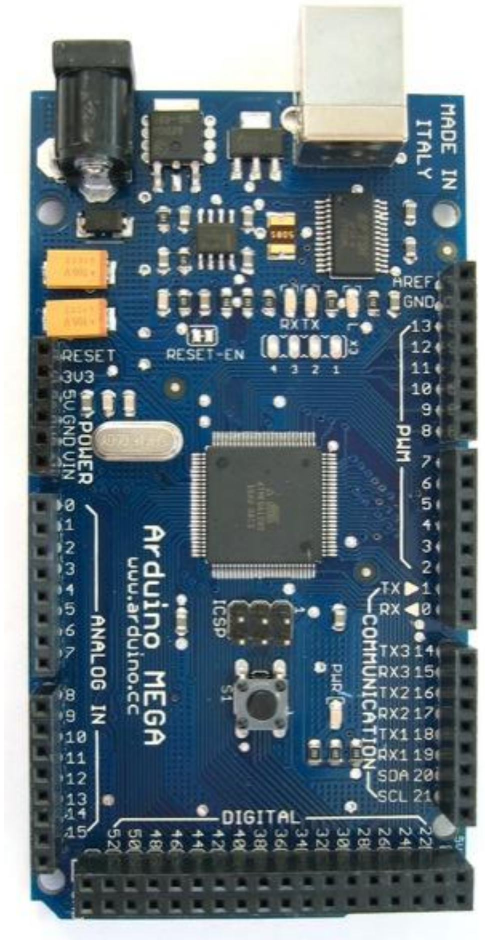
9V Battery

Voltage → 9V

Max Capacity → 550mAh

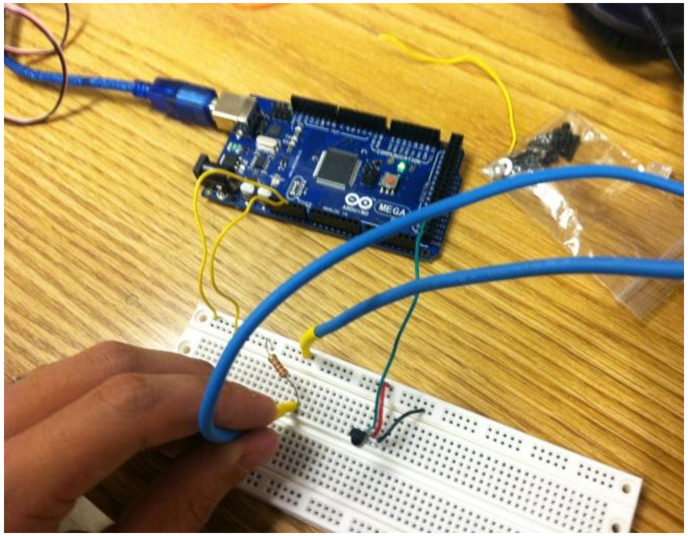
	Theoretical	Actual
Voltage	9V	9V
Current Capacity	550mAh	545mAh
Running time for our Project	5.5hours	5hours

Arduino Mega



Operating Voltage	5V
Input Voltage (recommended)	**7-12V**
Input Voltage (limits)	6-20V
Digital I/O Pins	**54 (of which 15 provide PWM output)**
Analog Input Pins	16
DC Current per I/O Pin	**40 mA**
DC Current for 3.3V Pin	50 mA
Flash Memory	128 KB of which 4 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

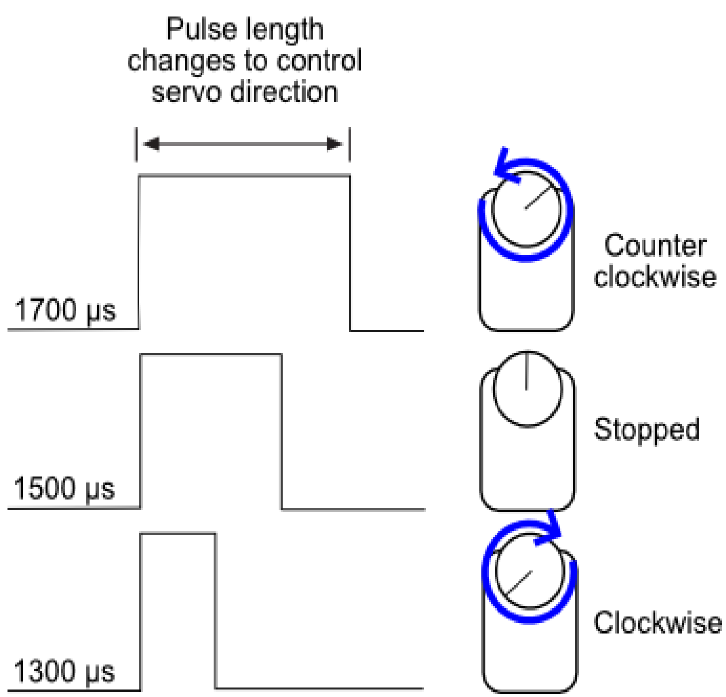
DC current I/O pin



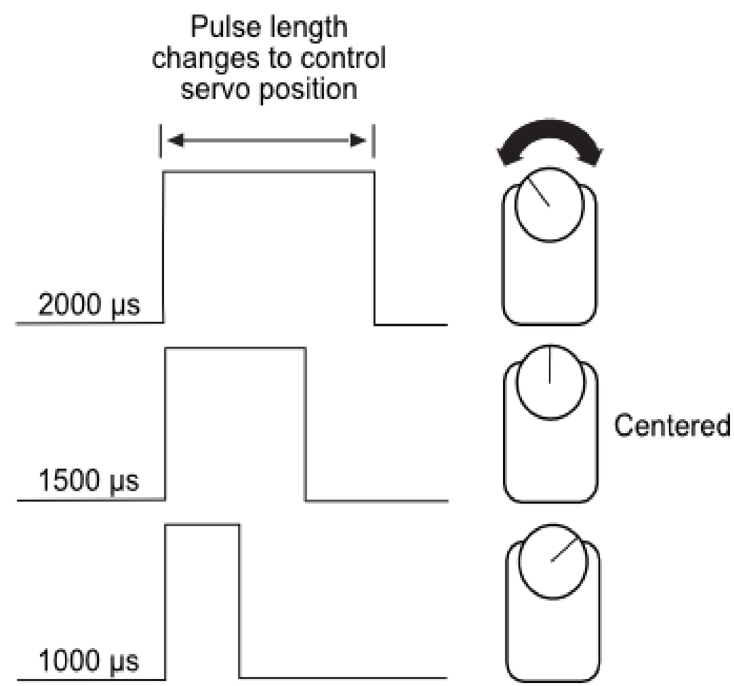
Theoretical	Measured	Error
40mA	41.288mA	3.12%

Servo Motor

- Operation of the Motors
  - The degree changes by pulse lengths



< **Continous Servo Motor** >



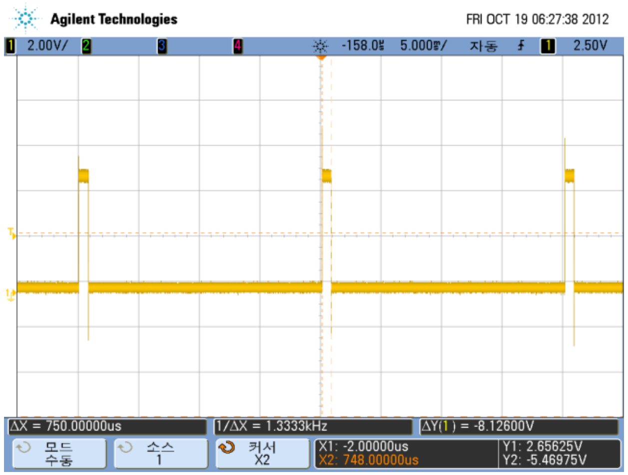
< **Standard Servo Motor** >

PWM and Pulse Length

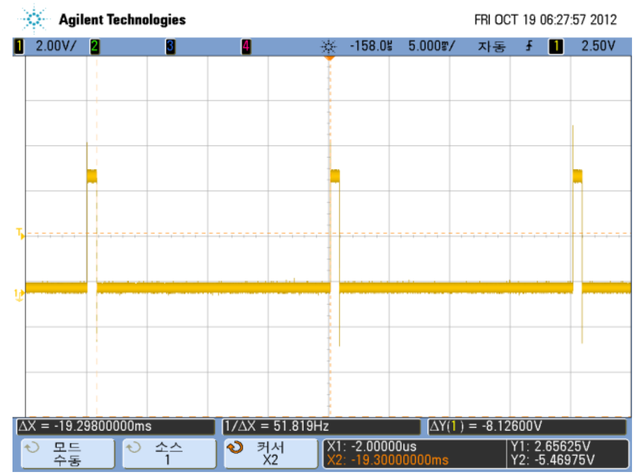
- What needs to be measured

Pulse Length	Operation
750μs	to the left-most position
1300μs	clockwise rotation
1500μs	to the center, stop at the center
1700μs	counter-clockwise rotation
2250μs	to the right-most position
< 20ms	to maintain a position

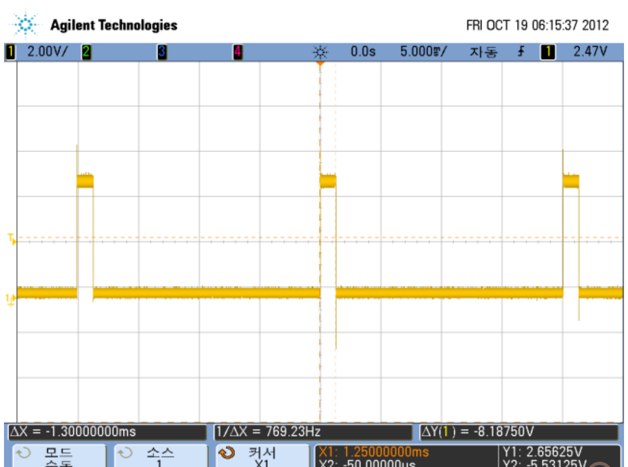
PWM signals



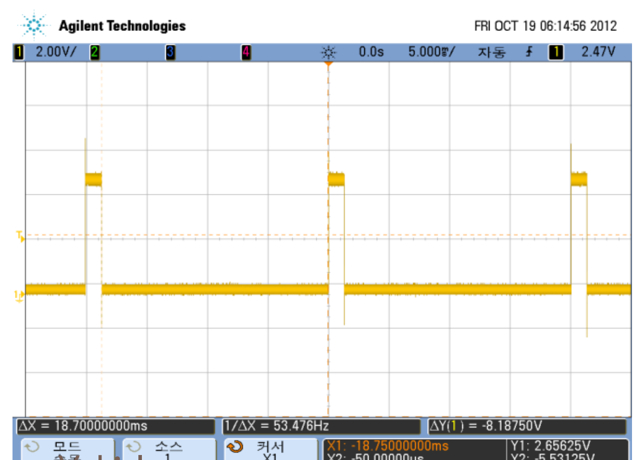
< 750us Pulse Length >



< 19.2ms for the gap >

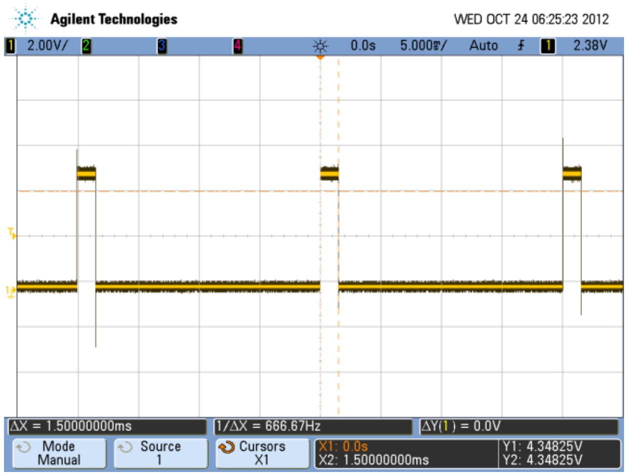


< 1300us Pulse Length >

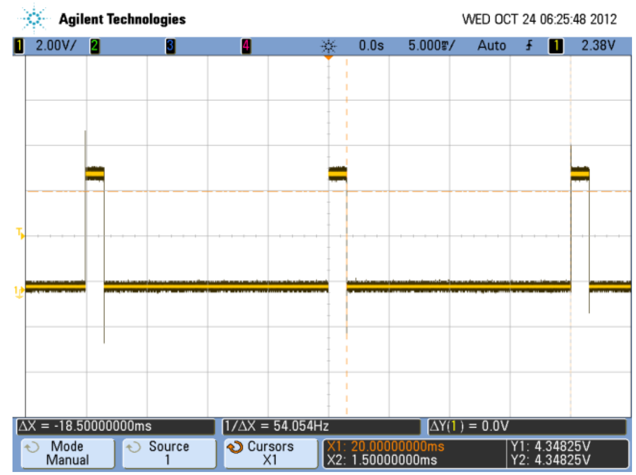


< 18.7ms for the gap >

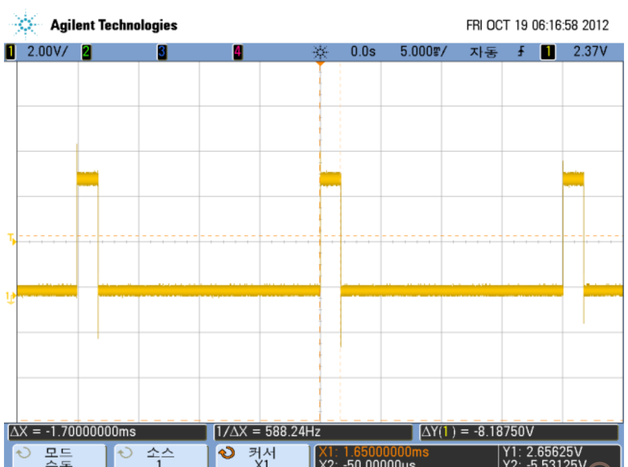
PWM signals



< 1500us Pulse Length >



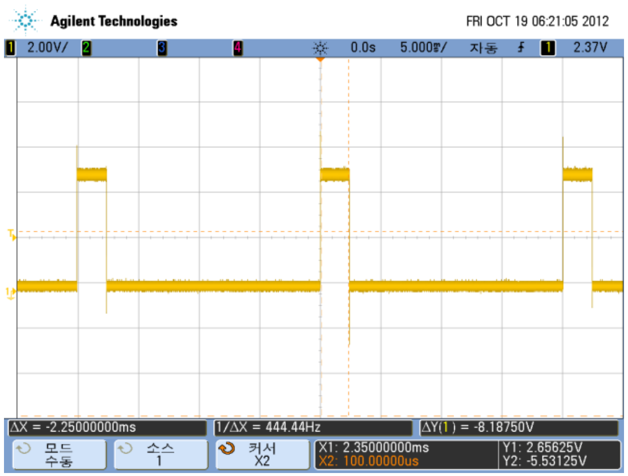
< 18.5ms for the gap >



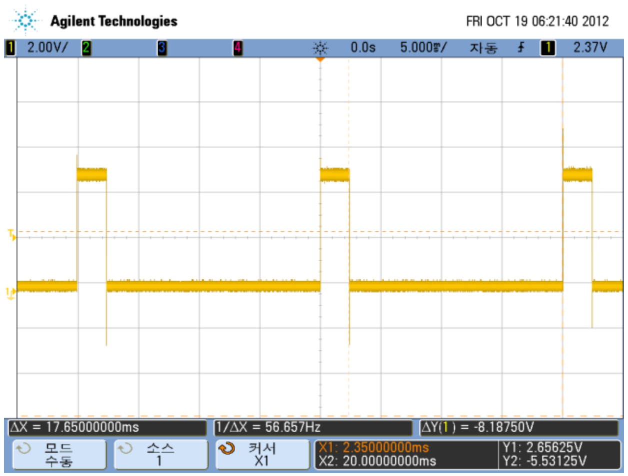
< 1700us Pulse Length >

< 18.2ms for the gap >

PWM signals



< 2250us Pulse Length >



< 17.6ms for the gap >

Rotation Checking

```
#include <Servo.h> // Use Servo library
Servo myServo; // Create Servo object

void setup() {
  myServo.attach(9); // Servo connected to pin 9
}

void loop() {
  myServo.writeMicroseconds(750); // 750us pulse
}
```

- **Issue at this stage**

  Every servo motor does not have the same default angle, so we needed to find the default angle to control the motors

  Code for angular input: Servo9.write(135); // rotate to 135 degree

Voltage Regulator

- To regulate voltage to 5V for servo motors(4~6V)
  - Maximum current capacity: 1.52A (8 motors × 190mA)
  - Each regulator minimum current capacity: 0.76A

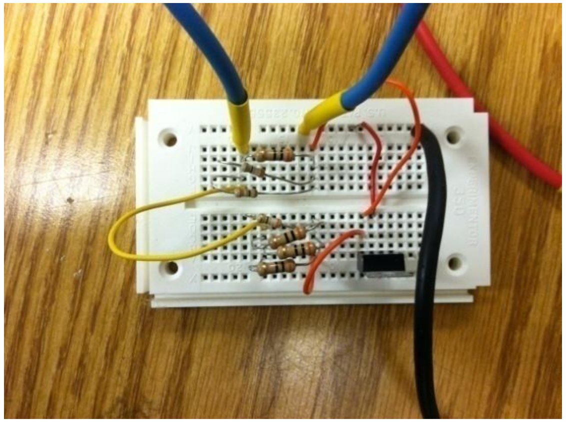
- Verification Model

$$\frac{1}{R_{tot}} = \frac{4}{30} + \frac{3}{39}, R_{tot} = 4.76\ \Omega$$

$$I = \frac{V}{R_{tot}} = \frac{5}{4.76} = 1.05\ A$$

Voltage Regulator

< Regulated Voltage : 4.95V >



< Wiring for the Verification>

< Measured Total Resistance: 4.8Ω>

< The current flowing through Regulator: 0.83A>

Proximity Sensor

< Ultrasonic Range Finder
- Maxbotix LV-EZ1 >

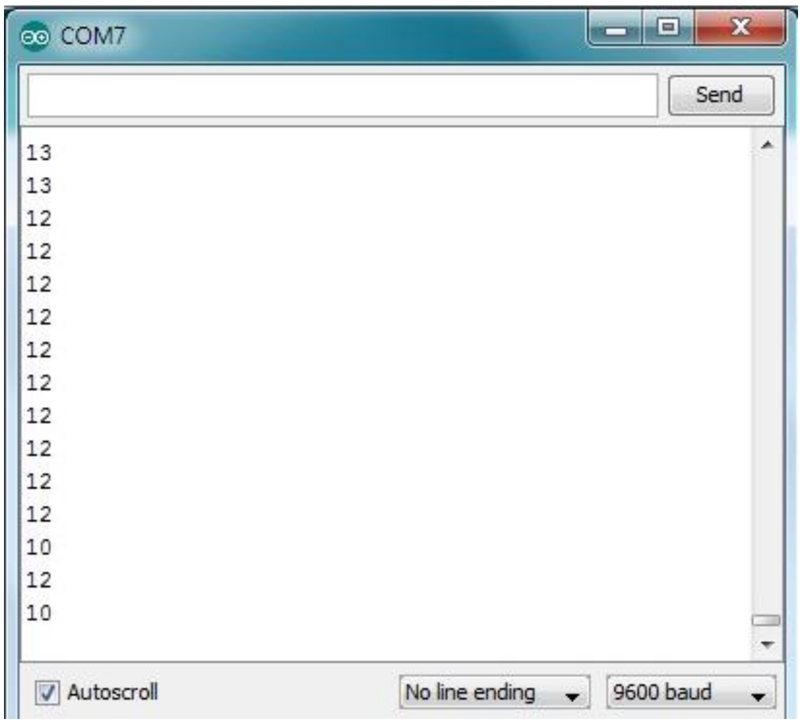
- 42kHz Ultrasonic sensor
- Operates from 2.5-5.5V
- Low 2mA supply current

Proximity Sensor

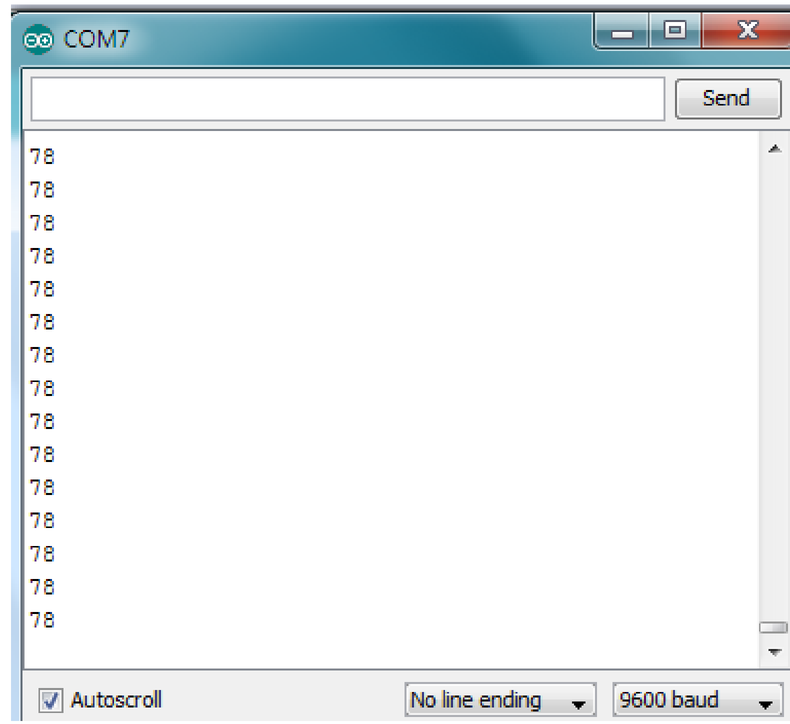
- Test Code

```
int sensorPin = 0; //analog pin 0

Void setup(){
        Serial.begin(9600);
}

void loop(){
        int val = analogRead(sensorpin);
        Serial.printIn(val);

        delay(100);
}
```

Proximity Sensor



< an object at the close distance>

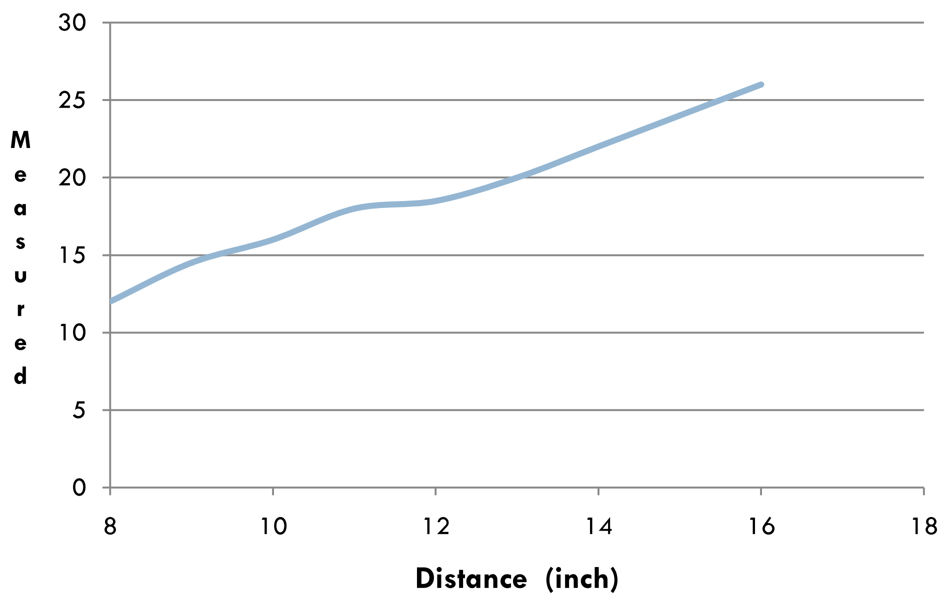


< an object at the further distance>

Proximity Sensor

Distance(Inch)	Measured
8	12
9	14.5
10	**16**
11	18
12	18.5
13	20
14	22
15	24
16	26

< a table for the distance measurement >

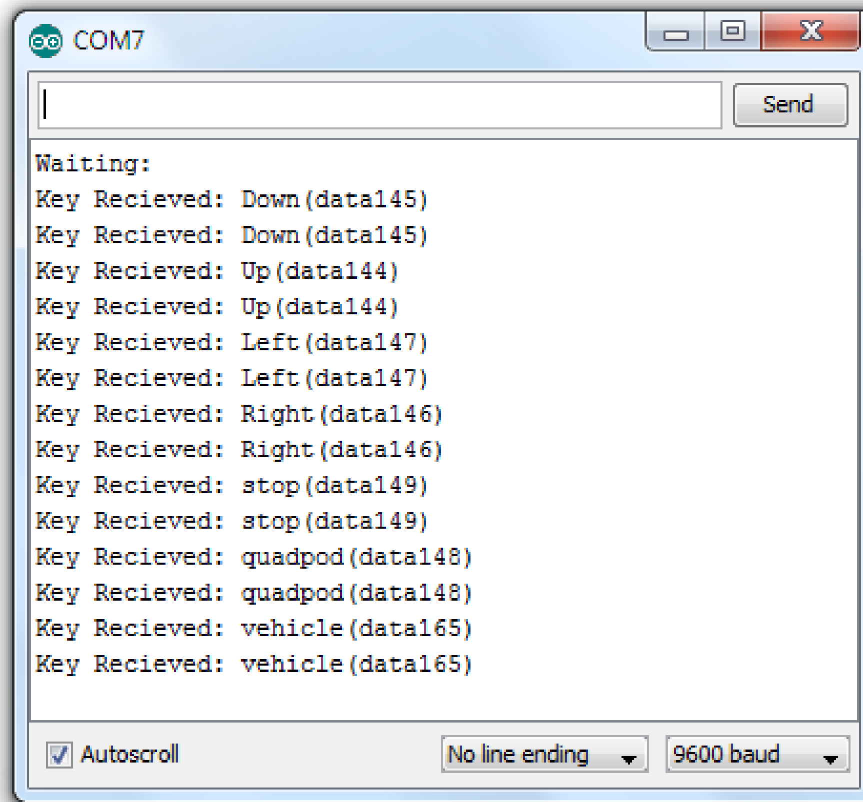


< A graph drawn from the table >

IR receiver and Remote Controller

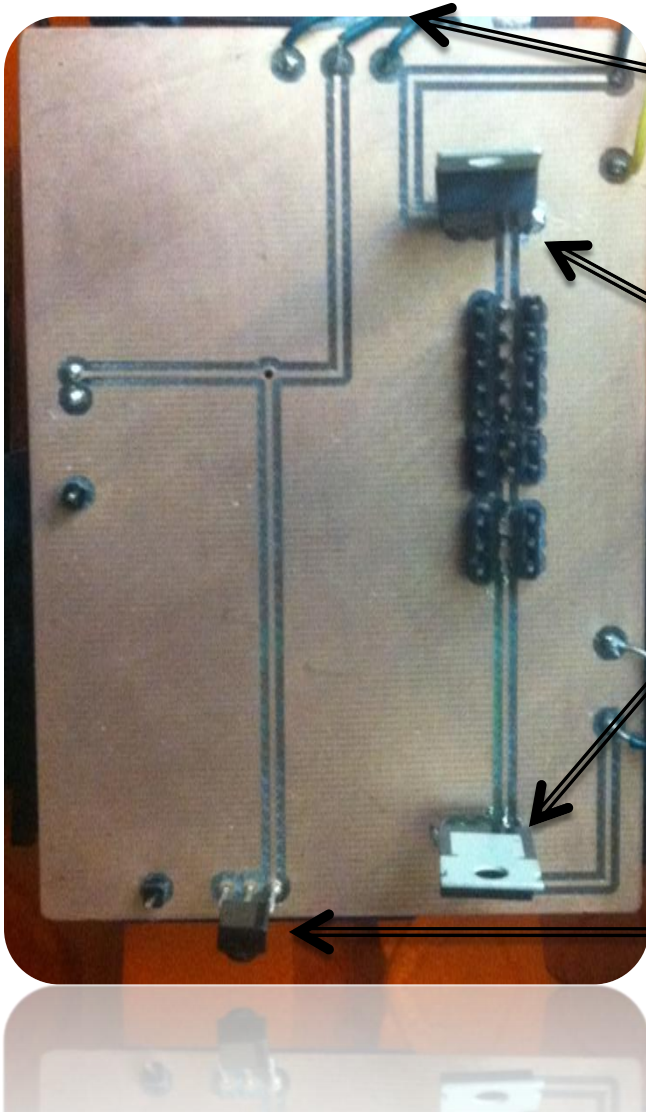
< IR receiver: TSOP38238 >

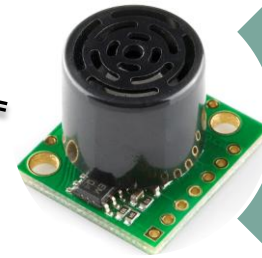
< Remote Controller >



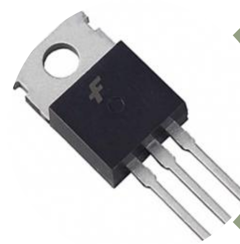
< Verification Results by the given code >

PCB

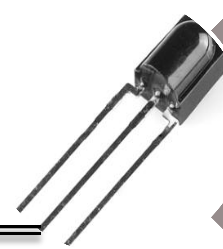




Proximity Sensor



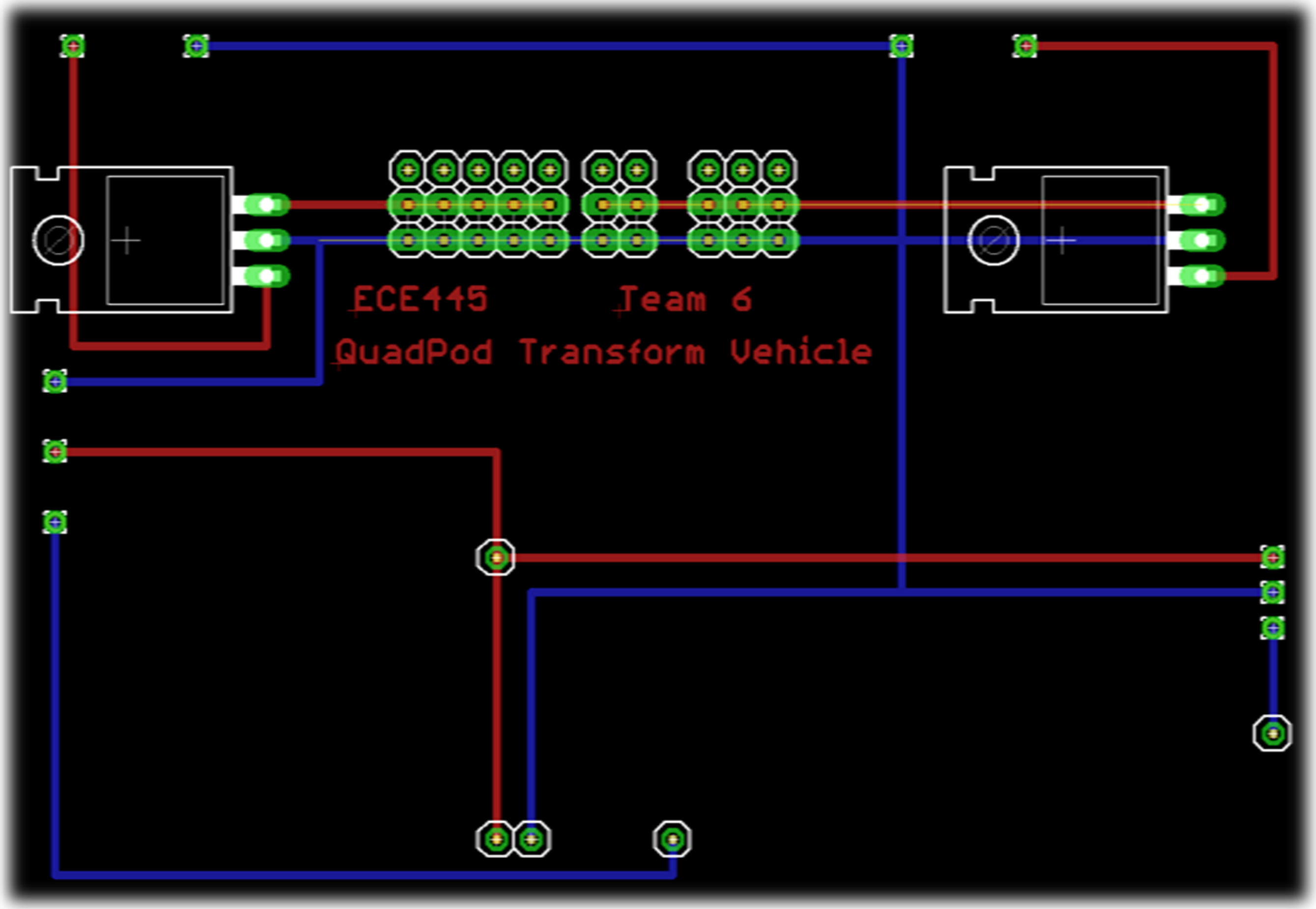
2 Voltage Regulator



IR Receiver

PCB

Proximity
Sensor



IR
Receiver

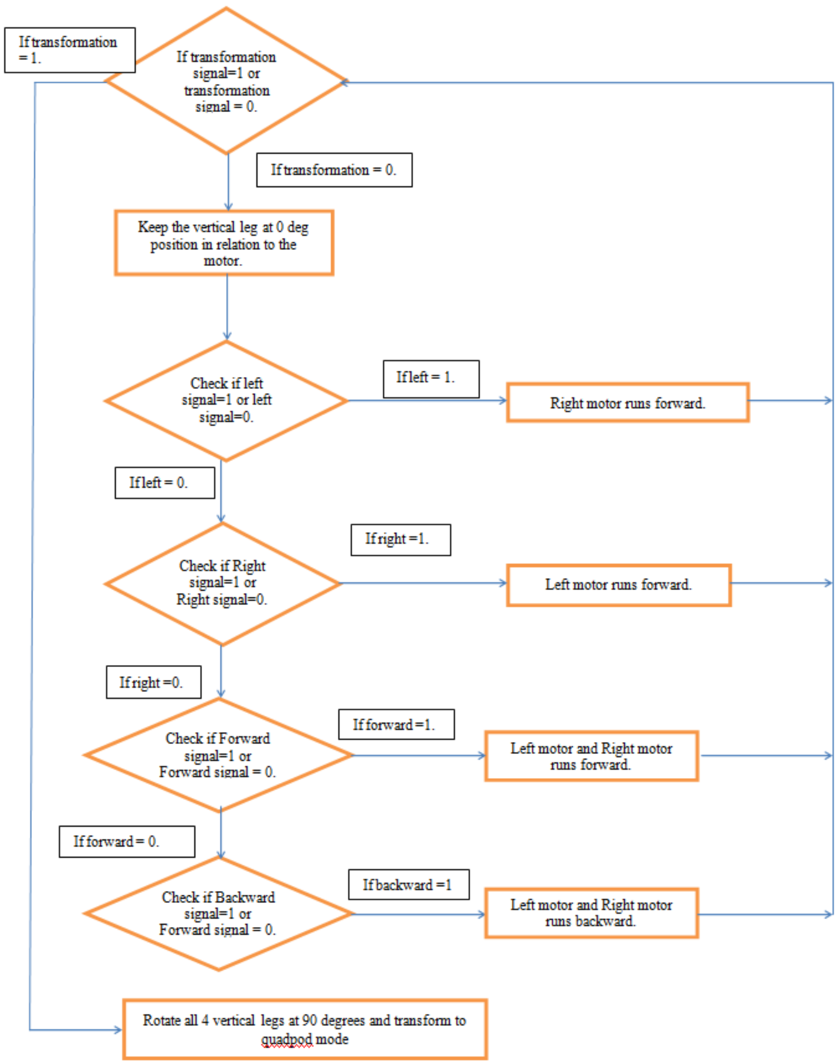
Code

- Contain two major codes
  - Vehicle mode
  - Quadpod mode
- Int “tr” was used to determine which code will be activa ted.

- Command “key = getIRKey()” is used to obtain signal fr om IR receiver.

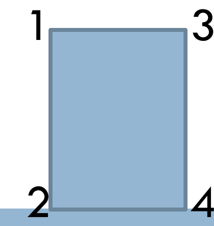
- Depends on which button is pressed from IR remote cont roller, different value of key will be provided.

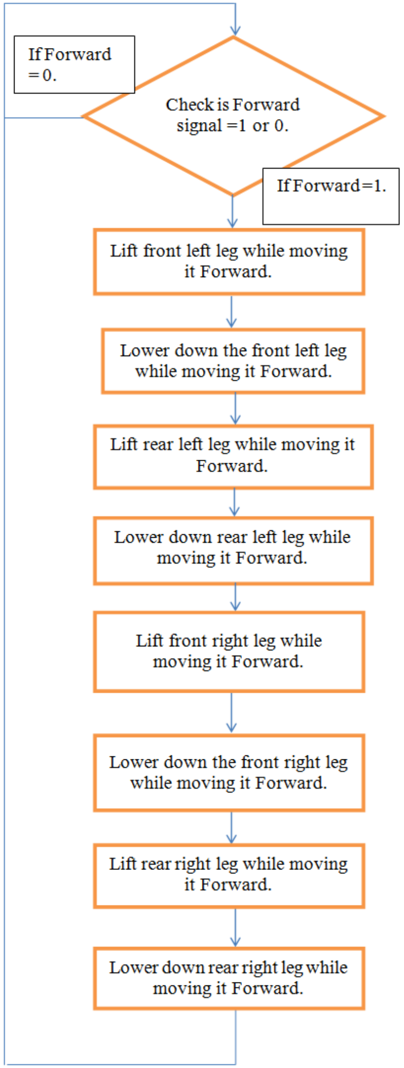
Codes for Vehicle mode




```
case 145: //("CH up") button is pressed from IR remote;
           Servo12.writeMicroseconds(1300);  // Right wheel motor rotate Cloc
          Servo11.writeMicroseconds(1700);  // Left wheel motor rotate Counte
          while(val > 15)  //do while distance is more than 10inch
          {
           key = getIRKey();  //Fetch the key from IR receiver
           if(key != 145)     //If key is not 145, break from the loop
           {
           break;}
           else{
             val = analogRead(sensorPin);  //read signal from bumper button
            if(val < 16)               //If the distance is less then 10inch,
            {Servo12.writeMicroseconds(1505);  //wheel motor should stop movi
            Servo11.writeMicroseconds(1515);   //quadpod mode
            Servo6.write(90);     // when quadpod mode start
            Servo7.write(90);     //all horizontal movement motors
            Servo9.write(85);     //are at 90 degree
            Servo10.write(90);
            delay(700);
            Servo2.write(165);
            Servo3.write(0);      //rotate legs to the lowest position
            Servo4.write(5);
            Servo8.write(180);
            delay(700);         //give 1 second delay so the vehicle will tran
            tr = 1;             //tr = 1 so the code for vehicle won't be acti
            }
           }
          }
          break;
```

Codes for Quadpod mode






```
case 145: //Serial.print("CH up");
        Servo3.write(25);//lift front left leg 30deg(25deg)
        Servo9.write(45);//rotate the front left leg forward(45deg)
        delay(100);  //give 0.1 second delay
        Servo6.write(132);//rotate rear left leg 15 deg backward(132deg)
        Servo10.write(90);//rotate front right leg 15 deg backward(90deg)
        Servo7.write(83);//rotate rear right leg 15 deg backward(deg)
        delay(100);  //give 0.1 second delay
        Servo3.write(0);
        delay(100);

        Servo8.write(155);//lift rear left leg 30 deg(30deg)
        Servo6.write(87);//rotate rear left leg forward(135deg)
        delay(100);  //give 0.1 second delay
        Servo9.write(60);//rotate front left leg 30 deg backward(105deg)
        Servo7.write(68);//rotate rear right leg 30 deg backward(105deg)
        Servo10.write(75);//rotate front right leg 30 deg backward(75deg)
        delay(100);  //give 0.1 second delay
        Servo8.write(180);
        delay(100);

        Servo2.write(140);//lift Front Right leg 30 deg(150deg)
        Servo10.write(120);//rotate the front right leg forward(45deg)
        delay(100);  //give 0.1 second delay
        Servo9.write(75);//rotate front left leg 30 deg backward(75deg)
        Servo7.write(53);//rotate rear right leg 30 deg backward(75deg)
        Servo6.write(102);//rotate rear left leg 30 deg backward(45deg)
        delay(100);  //give 0.1 second delay
        Servo2.write(165);
        delay(100);
```

Challenge

- Obtaining light and durable main body frame
- Keeping the balance of Quadpod while it is moving
- Weight issues

Demonstration Video

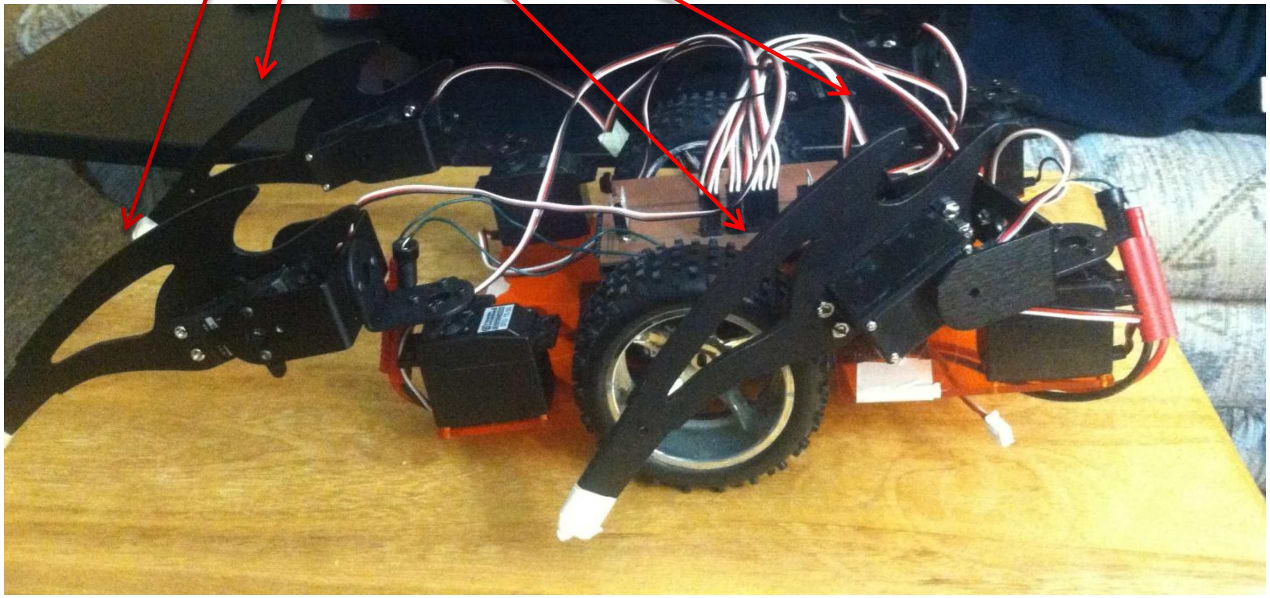
Quadpod Transform Vehicle

Recommendation

- Better Mechanical Design
- Use Stronger Servo Motor
- Multiple Proximity sensor to cover blind spot

Better Mechanical Design

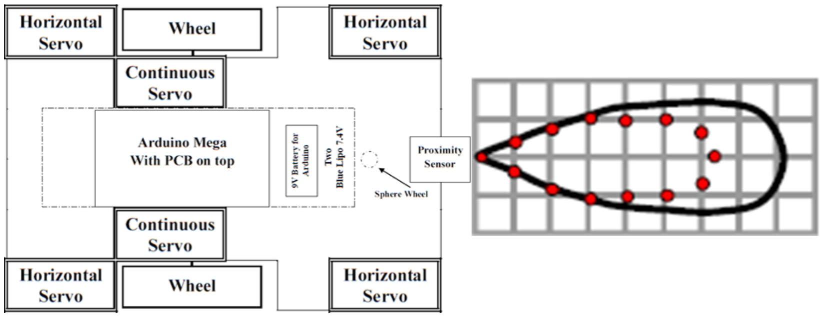
Pulse signal needs to be provided to hold the legs up

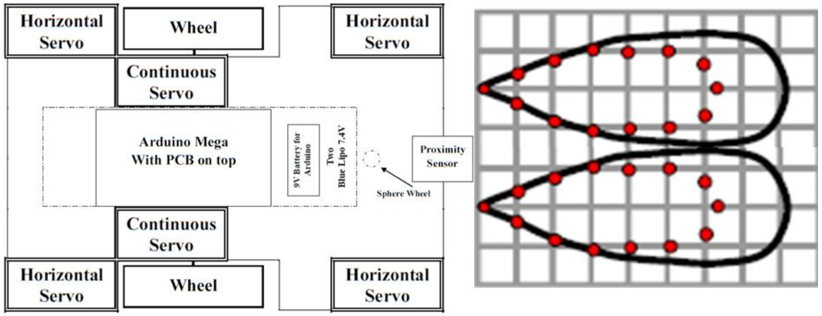


Using multiple proximity sensor

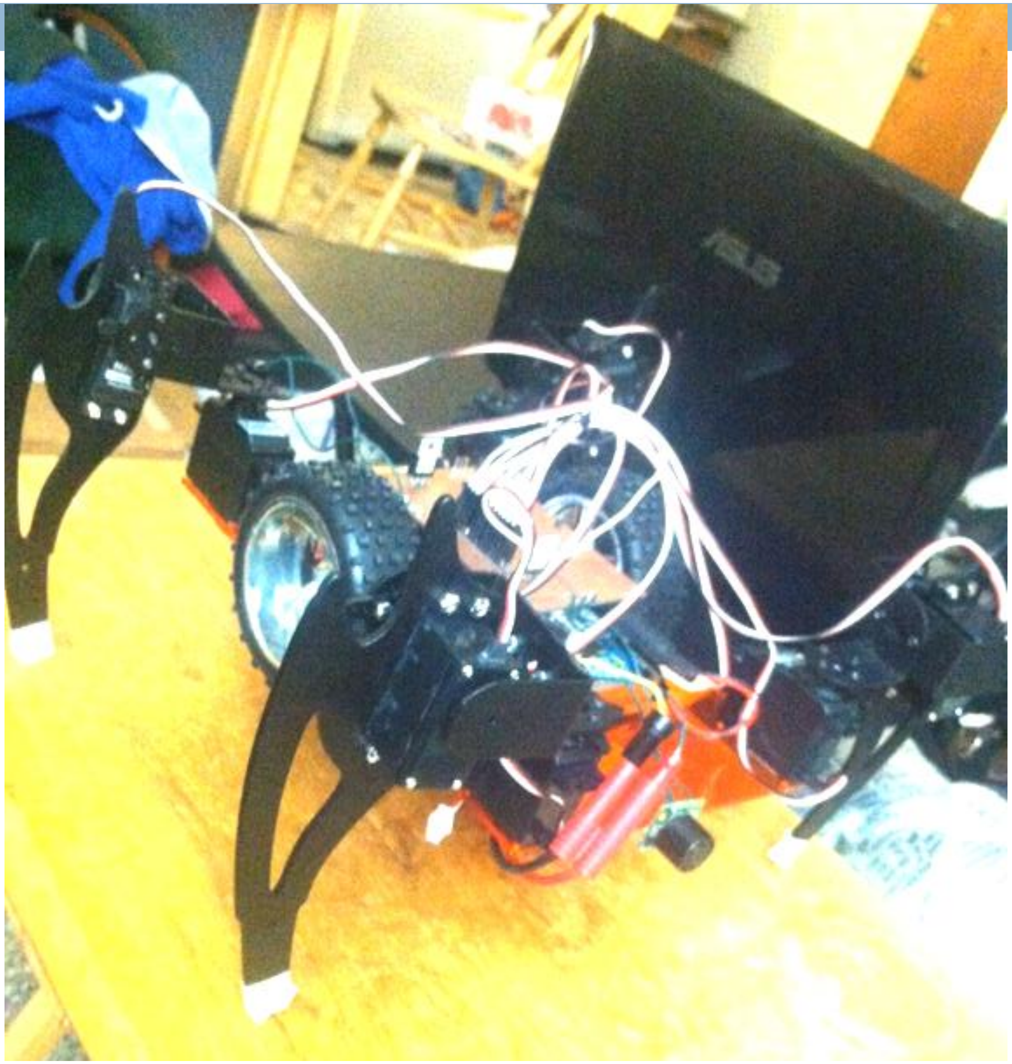
Current Design(1proximity sensor)

Recommendation(2 or more sensor)





Questions?



Quadpod Transform Vehicle