LAMPIRAN

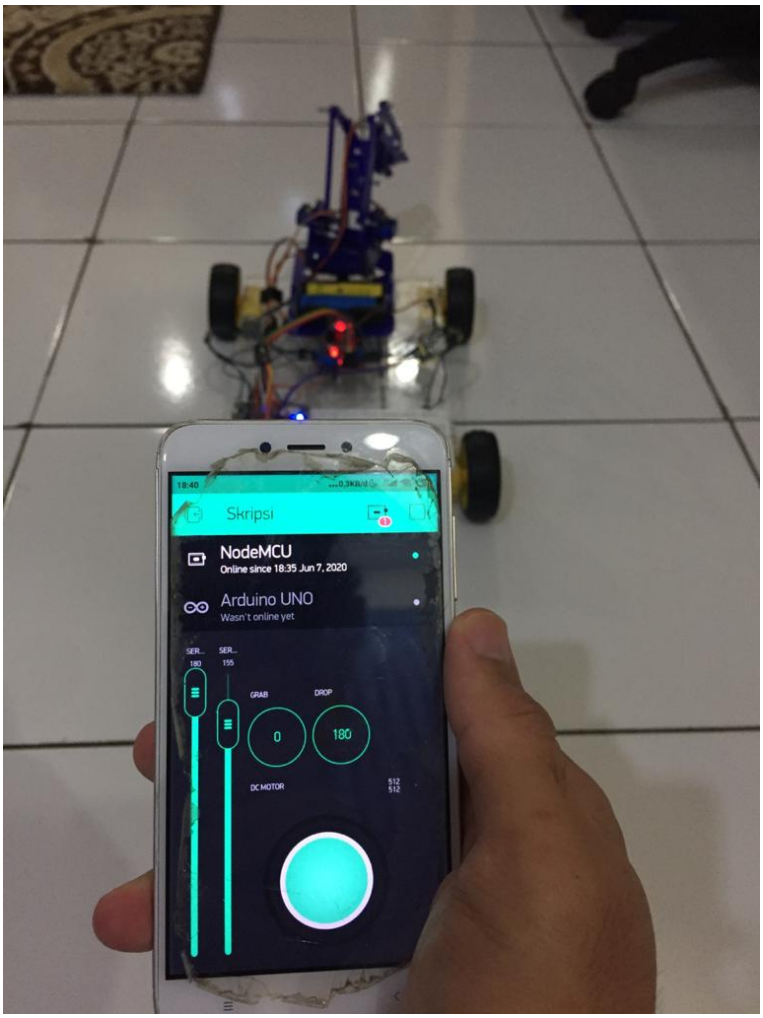
LAMPIRAN 1: Listing Program *Blynk*.

```
//#include <Blynk.h>
//#include <WiFiClient.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>


#define IN_1  15          // L298N in1 motors Right          GPIO15(D8)
#define IN_2  13          // L298N in2 motors Right          GPIO13(D7)
#define IN_3  2           // L298N in3 motors Left           GPIO2(D4)
#define IN_4  0           // L298N in4 motors Left           GPIO0(D3)


char auth[] = "7d786b75eb914657ba029a78faff50d1";
char ssid[] = "qwerty";
char pass[] = "qwerty123";

int minRange = 312;
int maxRange = 712;



Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;


void moveControl(int x, int y)
{

  if(y >= maxRange && x >= minRange && x <= maxRange) //move forward
  {
      digitalWrite(IN_1, LOW);
      digitalWrite(IN_2, HIGH);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, HIGH);
      digitalWrite(IN_4, LOW);


  }

  // turn right
  else if(x >= maxRange && y >= maxRange) {

      digitalWrite(IN_1, LOW);
      digitalWrite(IN_2, HIGH);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, LOW);
      digitalWrite(IN_4, HIGH);
//      analogWrite(ENB, speedCar);   }
  }

    // turn left
  else if(x <= minRange && y >= maxRange){

      digitalWrite(IN_1, HIGH);
```

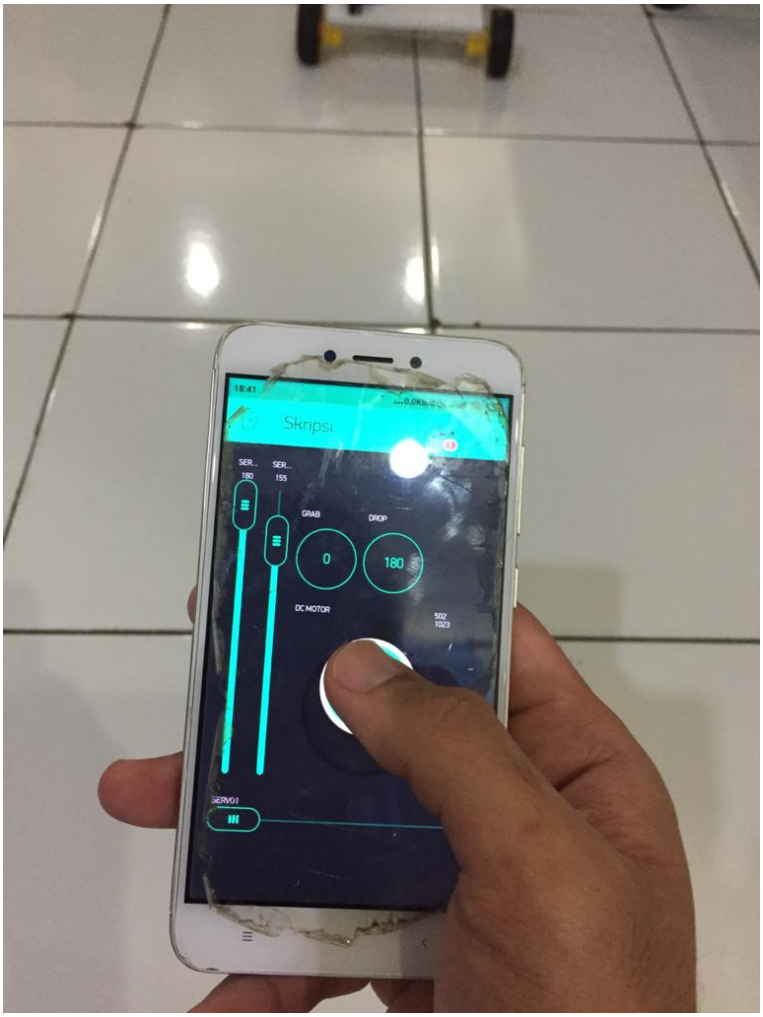
```
      digitalWrite(IN_2, LOW);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, HIGH);
      digitalWrite(IN_4, LOW);
//      analogWrite(ENB, speedCar);
  }

  // STOP
  else if(y < maxRange && y > minRange && x < maxRange && x > minRange){

      digitalWrite(IN_1, LOW);
      digitalWrite(IN_2, LOW);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, LOW);
      digitalWrite(IN_4, LOW);
//      analogWrite(ENB, speedCar);
 }

 // move back
  else if(y <= minRange && x >= minRange && x <= maxRange){

      digitalWrite(IN_1, HIGH);
      digitalWrite(IN_2, LOW);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, LOW);
      digitalWrite(IN_4, HIGH);


  }


  // move back and right
 else if(y <= minRange && x <= minRange){

      digitalWrite(IN_1, LOW);
      digitalWrite(IN_2, HIGH);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, LOW);
      digitalWrite(IN_4, HIGH);
//      analogWrite(ENB, speedCar);    }
  }

  // move back and left
  else if(y <= minRange && x >= maxRange){

      digitalWrite(IN_1, HIGH);
      digitalWrite(IN_2, LOW);
//      analogWrite(ENA, speedCar);

      digitalWrite(IN_3, HIGH);
      digitalWrite(IN_4, LOW);
//      analogWrite(ENB, speedCar);
  }
}
BLYNK_WRITE(V1)

{
   servo1.write(param.asInt());
}
BLYNK_WRITE(V2)
{  int a = param.asInt();
servo2.write(a);

}
BLYNK_WRITE(V3)
{
  int a = param.asInt();
servo3.write(a);

}

BLYNK_WRITE(V4){
 int state = param.asInt();
  if (state == 1) {
    servo4.write(0);
  }
}

BLYNK_WRITE(V5) {
```

```
  int state = param.asInt();
  if (state == 1) {
    servo4.write(180);
  }
}


void setup() {


  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);

  servo1.attach(5);
  servo2.attach(4);
  servo3.attach(14);
  servo4.attach(12);


 pinMode(IN_1, OUTPUT);
 pinMode(IN_2, OUTPUT);
 pinMode(IN_3, OUTPUT);
 pinMode(IN_4, OUTPUT);

}
BLYNK_WRITE(V6)
{
  int x = param[0].asInt();
  int y = param[1].asInt();

  Serial.print("x value is: ");
  Serial.println(x);
  Serial.print("y value is: ");
  Serial.println(y);

  moveControl(x,y);
}


void loop()
{

  Blynk.run();
}
```

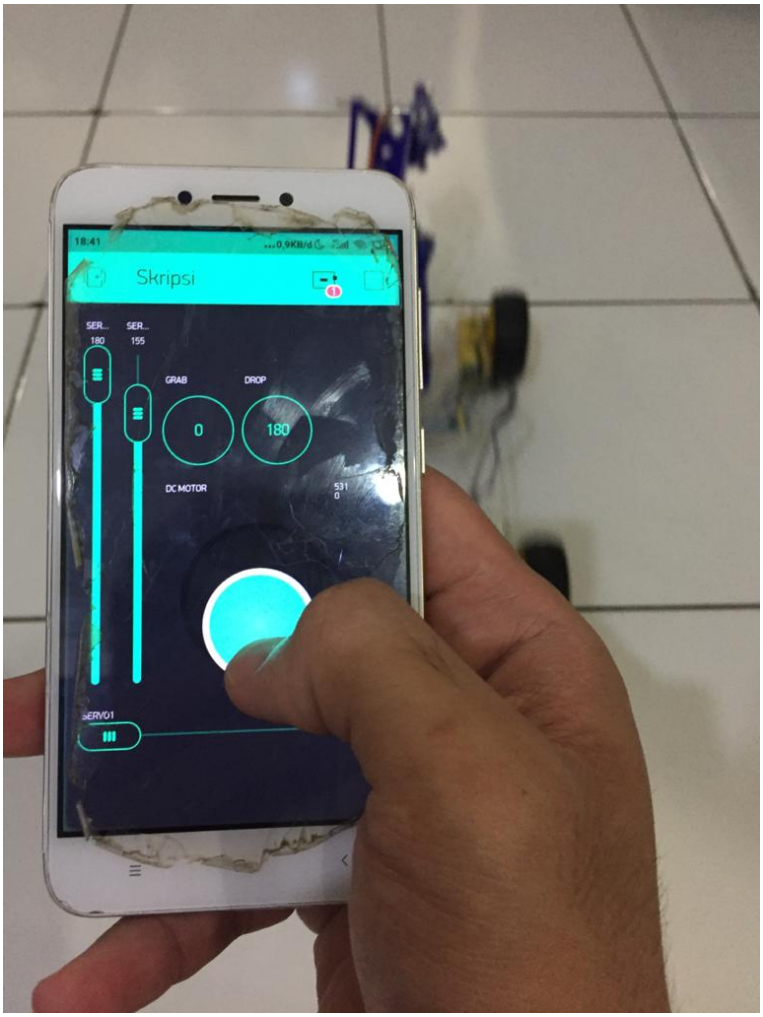
LAMPIRAN 2: Gambar bentuk fisik rangkaian.



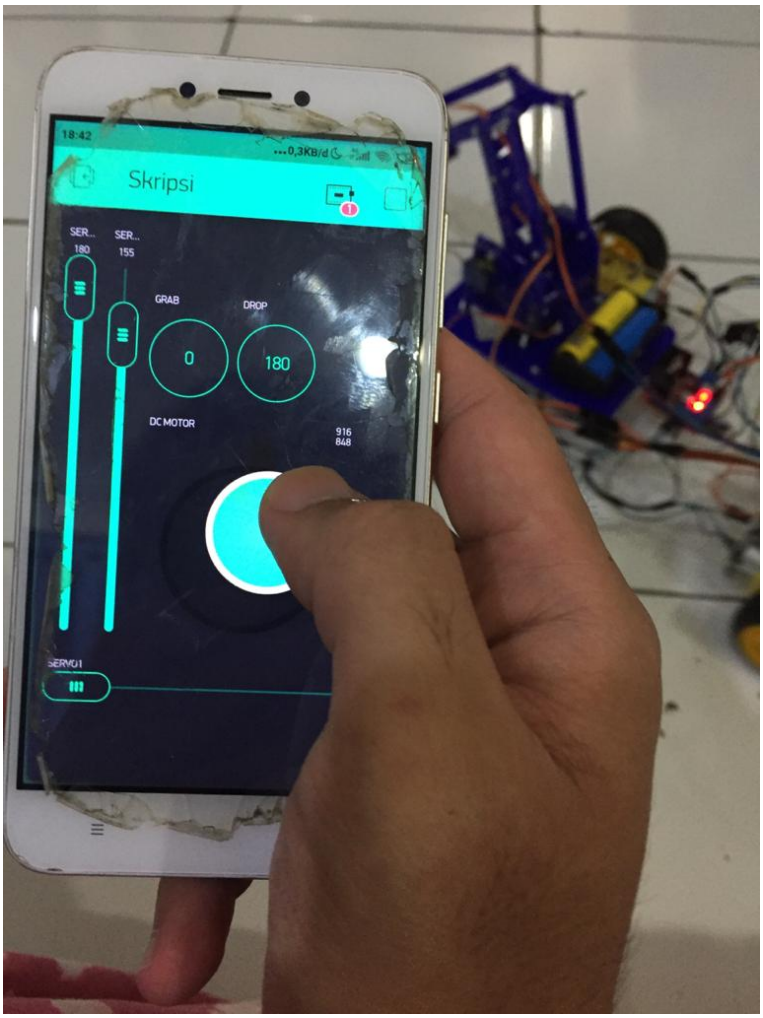
Gambar 1: Robot telah terkoneksi dengan blynk server



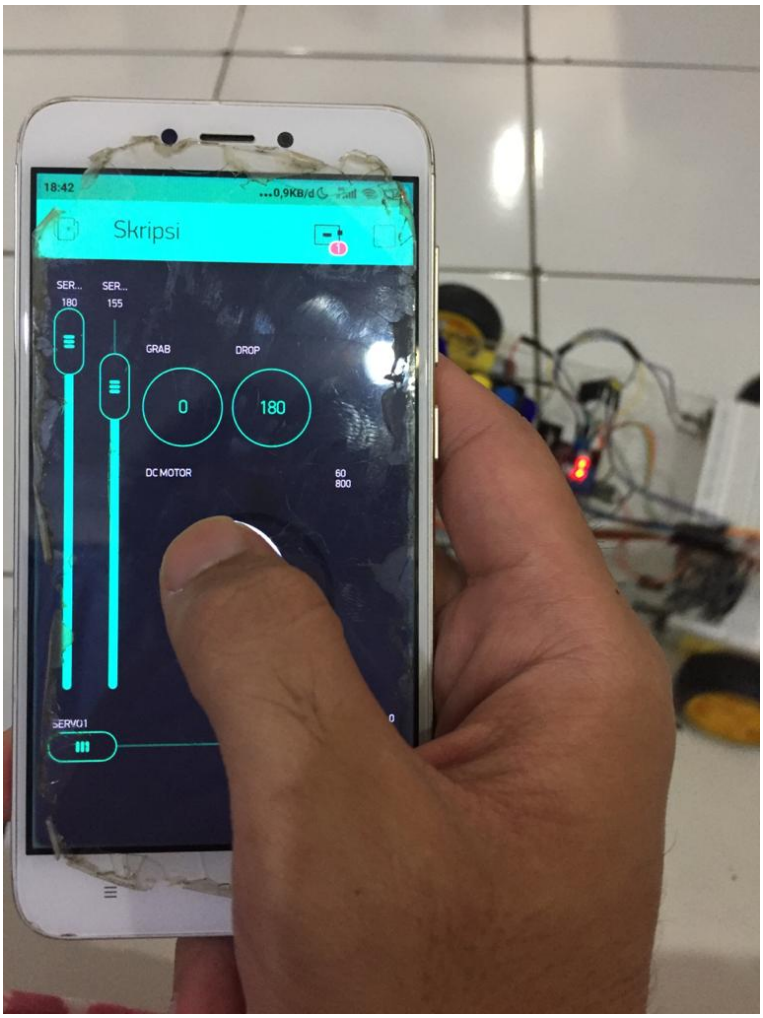
Gambar 2: Robot bergerak mundur



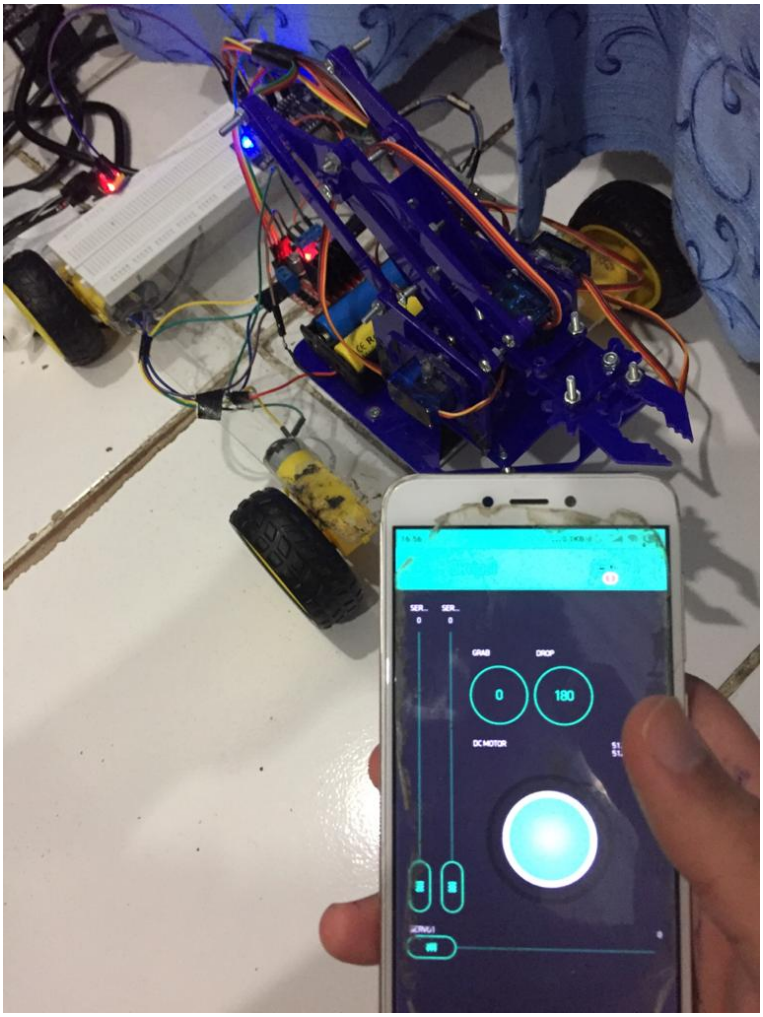
Gambar 3: Robot bergerak mundur



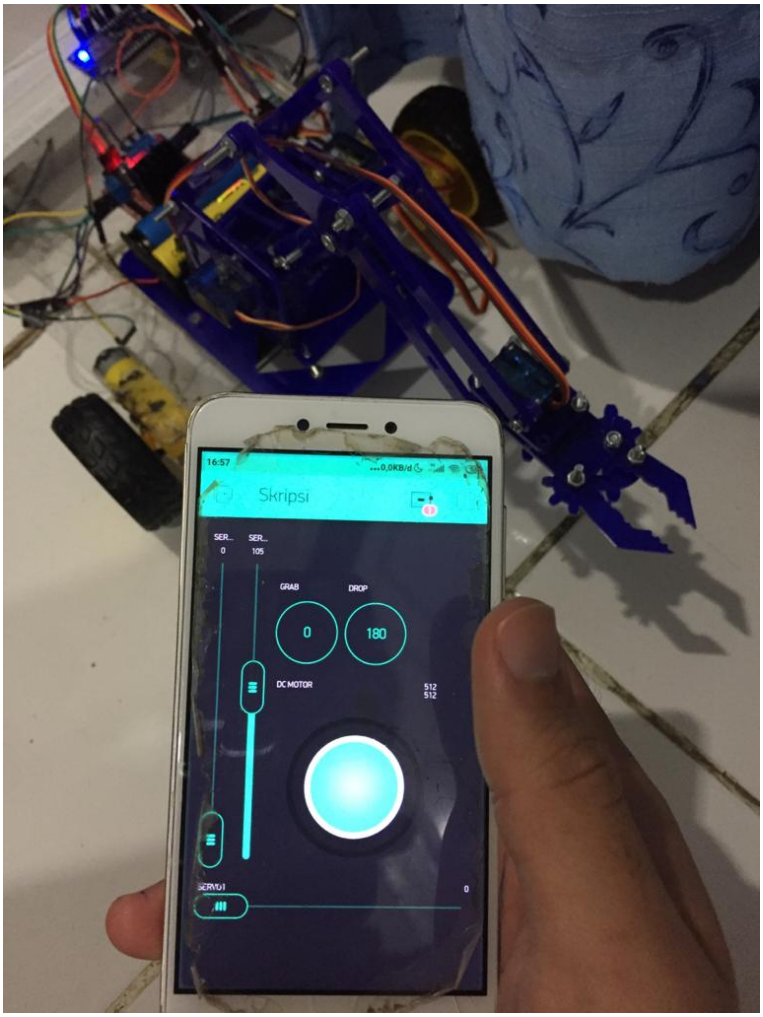
Gambar 4: Robot belok kanan



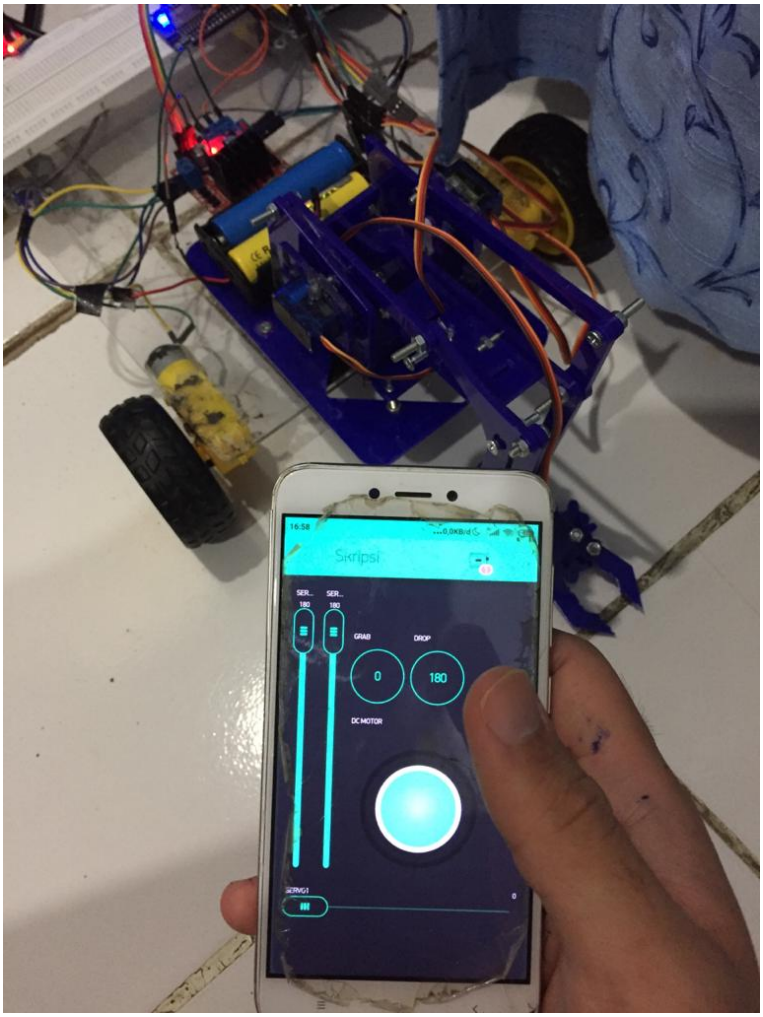
Gambar 5: Robot belok kiri



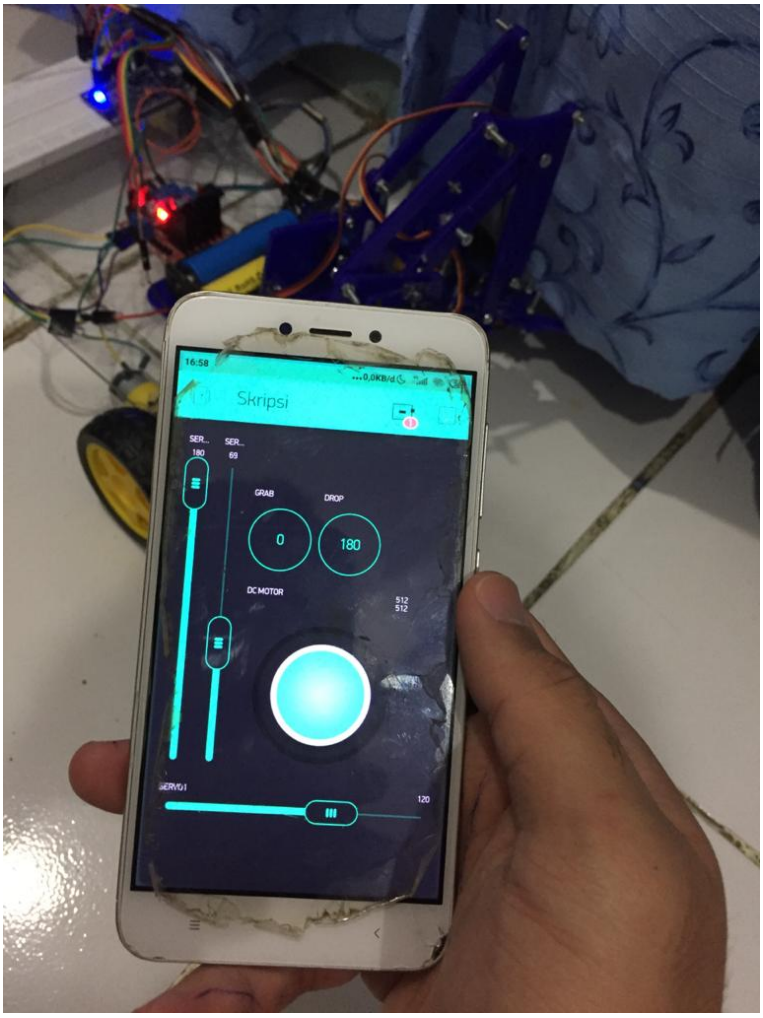
Gambar 6: Servo dalam posisi belum digerakan



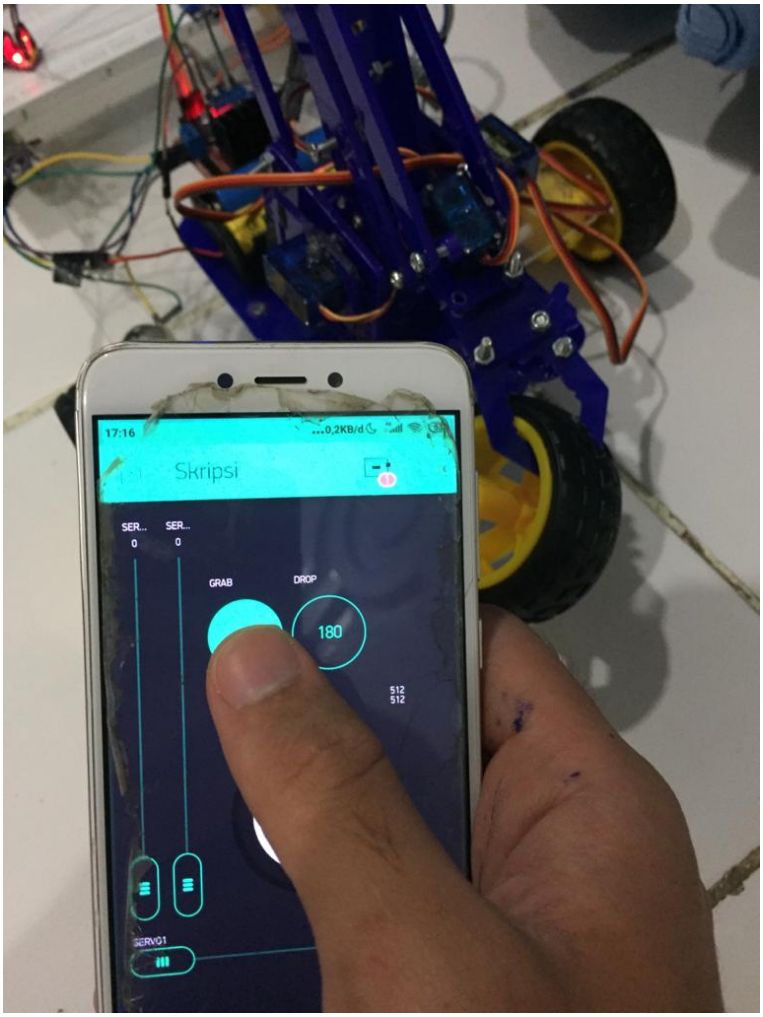
Gambar 7: *Sholder servo* berputar 105 derajat



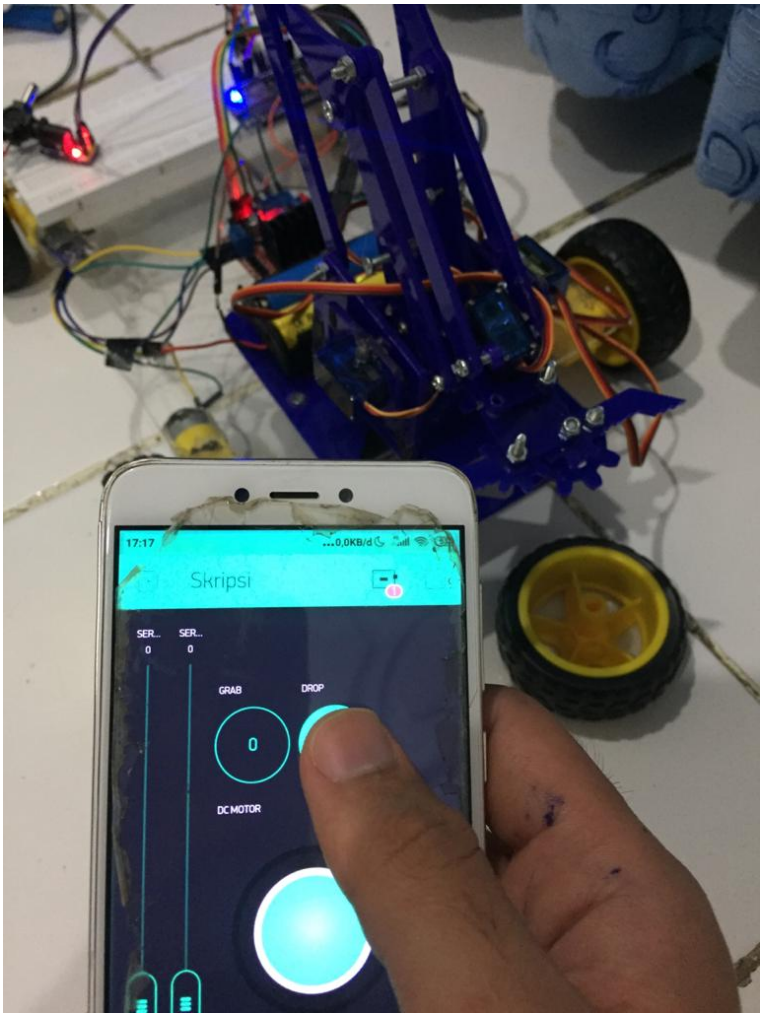
Gambar 8: *Elbow servo* berputar 180 derajat



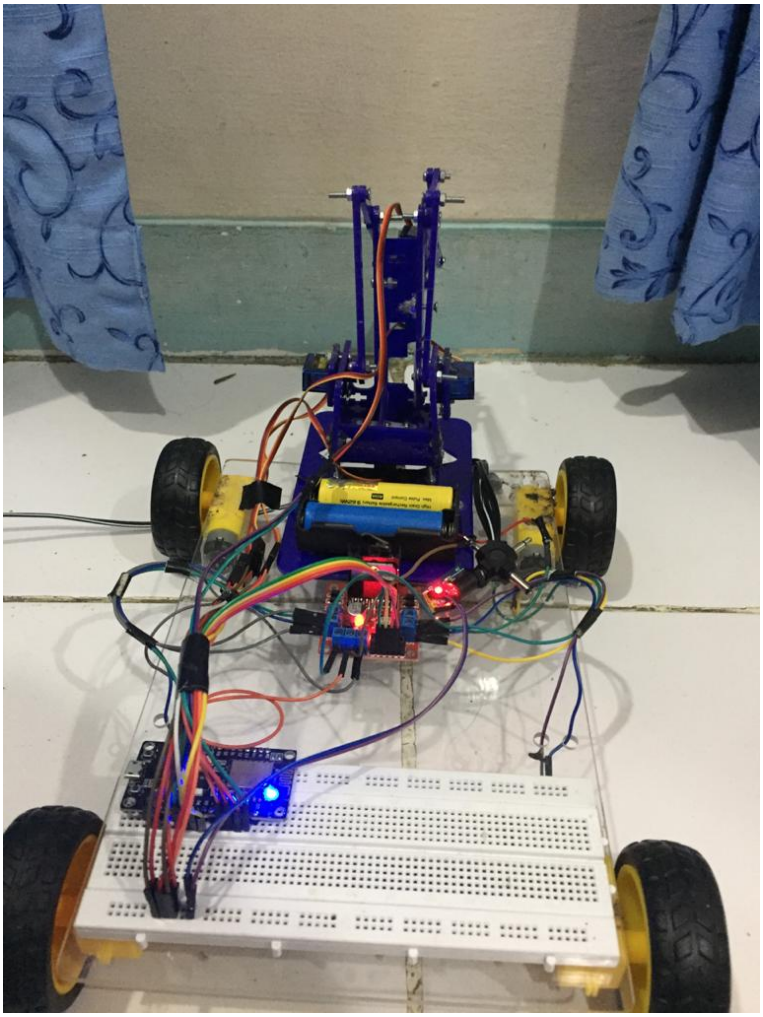
Gambar 9: *Base servo* berputar 180 derajat



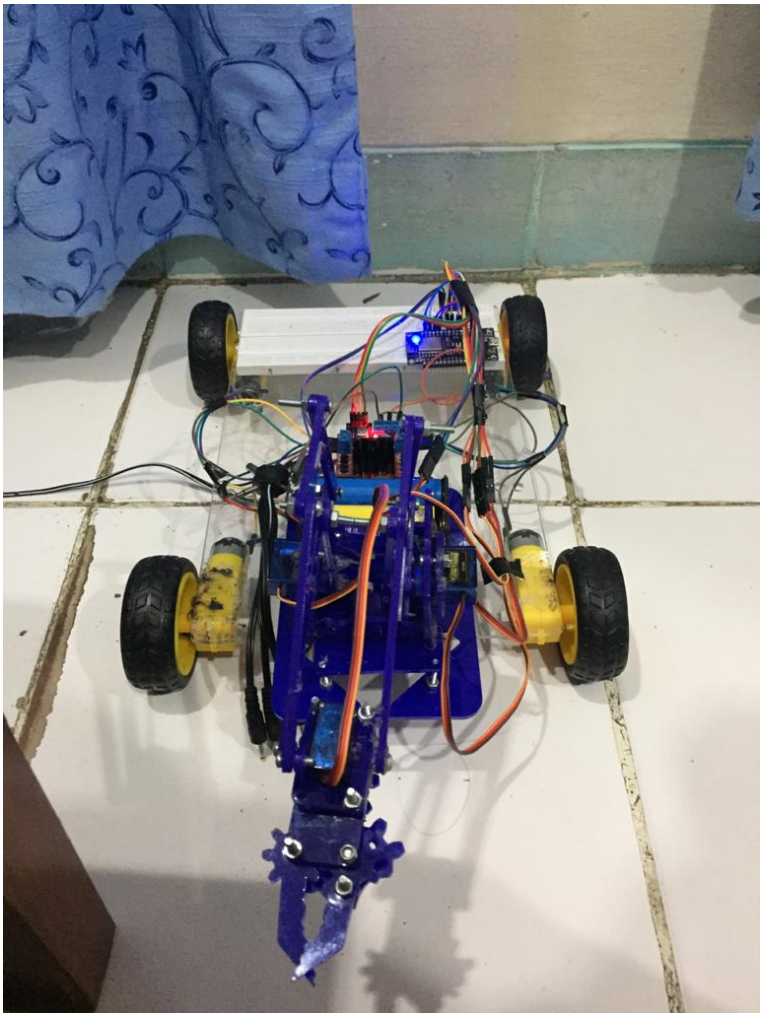
Gambar 10: *Gripper servo* berputar 0 derajat



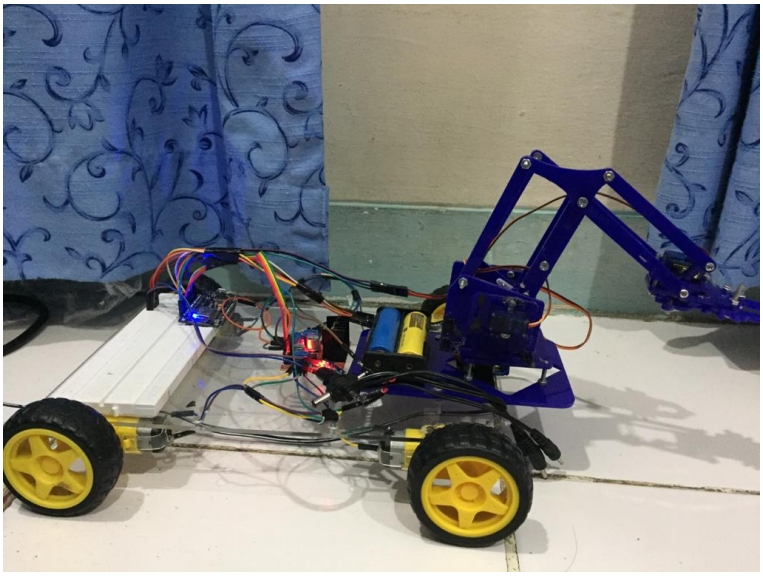
Gambar 11: *Gripper servo* berputar 180 derajat



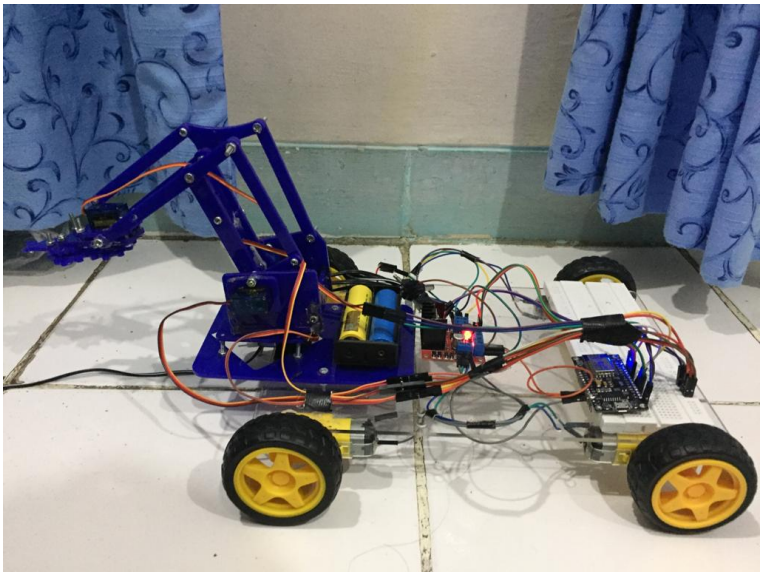
Gambar 12: Robot Tampak dari Belakang



Gambar 13: Robot Tampak dari Depan



Gambar 14: Robot Tampak dari Kanan



Gambar 15: Robot Tampak dari Kiri