

## **MODUL 2 SISTEM KENDALI PWM**



Mata Kuliah : Sistem Kendali  
Kelompok : TB-BERKAH JAYA  
Dosen : MHI

Nama Anggota :  
MUHAMMAD RAIHAN AKBAR 6702194044  
NANDA AZMIANSYAH KARDIMAN 6702190034

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER  
FAKULTAS ILMU TERAPAN  
TELKOM UNIVERSITY  
2021**

## 1. JUDUL PRAKTIKUM

Sistem Kendali PWM (*Pulse Width Modulation*)

## 2. MAKSUD DAN TUJUAN

Maksud dan tujuan dari praktikum ini adalah :

- a. Mahasiswa dapat memahami fungsi dan cara kerja PWM pada motor DC
- b. Mahasiswa dapat membuat program sistem kendali berbasis PWM

## 3. PERALATAN DAN BAHAN

### 1. Perangkat Lunak

- Tinker Cad
- Software IDE Arduino

### 2. Perangkat Keras

- Arduino UNO R3
- H-bridge Motor Driver
- 6 Photodiode
- 6 Resistor 33k Ohm

## 4. DASAR TEORI

PWM adalah kepanjangan dari *Pulse Width Modulation* atau dalam bahasa Indonesia dapat diterjemahkan menjadi Modulasi Lebar Pulsa. Jadi pada dasarnya, PWM adalah suatu teknik modulasi yang mengubah lebar pulsa (pulse width) dengan nilai frekuensi dan amplitudo yang tetap. PWM dapat dianggap sebagai kebalikan dari ADC (Analog to Digital Converter) yang mengkonversi sinyal Analog ke Digital, PWM atau Pulse Width Modulation ini digunakan menghasilkan sinyal analog dari perangkat Digital (contohnya dari Mikrokontroler).

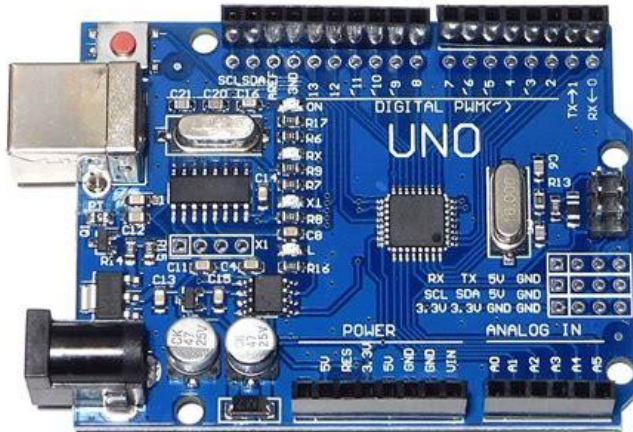
Rangkaian-rangkaian seperti Inverter, Konverter, Switch mode power supply (SMPS) dan Pengontrol kecepatan (Speed Controller) adalah rangkaian-rangkaian memiliki banyak sakelar elektronik di dalamnya. Sakelar-sakelar elektronik yang digunakan pada rangkaian tersebut umumnya adalah komponen elektronik daya seperti MOSFET, IGBT, TRIAC dan lain-lainnya. Untuk mengendalikan sakelar elektronik daya semacam ini, kita biasanya menggunakan sesuatu yang disebut sinyal PWM (Pulse Width Modulation). Selain itu, sinyal PWM juga sering digunakan untuk mengendarai motor Servo dan juga digunakan untuk melakukan tugas-tugas sederhana lainnya seperti mengendalikan kecerahan LED.

Untuk lebih memahami apa yang dimaksud dengan PWM atau Pulse Width Modulation ini. Kita coba melihat contoh dari sinyal yang dihasilkan oleh Mikrokontroler atau IC 555. Sinyal yang dihasilkan oleh Mikrokontroler atau IC555 ini adalah sinyal pulsa yang umumnya berbentuk gelombang segiempat. Gelombang yang dihasilkan ini akan tinggi atau rendah pada waktu tertentu. Misalnya gelombang tinggi di 5V dan paling rendah di 0V. Durasi atau lamanya waktu dimana sinyal tetap berada di posisi tinggi disebut dengan "ON Time" atau "Waktu ON" sedangkan sinyal tetap berada di posisi rendah atau 0V disebut dengan "OFF Time" atau "Waktu OFF". Untuk sinyal PWM, kita perlu melihat dua parameter penting yang terkait dengannya yaitu Siklus Kerja PWM (PWM Duty Cycle) dan Frekuensi PWM (PWM Frequency).

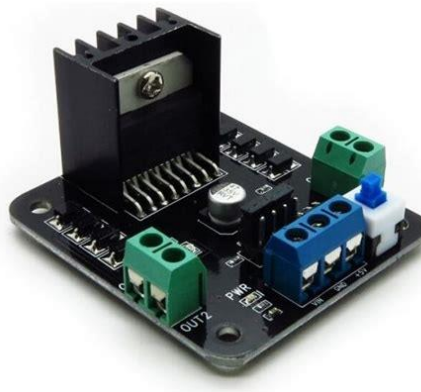
Referensi : <https://teknikelektronika.com/pengertian-pwm-pulse-width-modulation-atau-modulasi-lebar-pulsa/>

## 5. FOTO PERALATAN DAN BAHAN PRAKTIKUM

- Arduino UNO R3



- H-bridge Motor Driver



- Photodiode

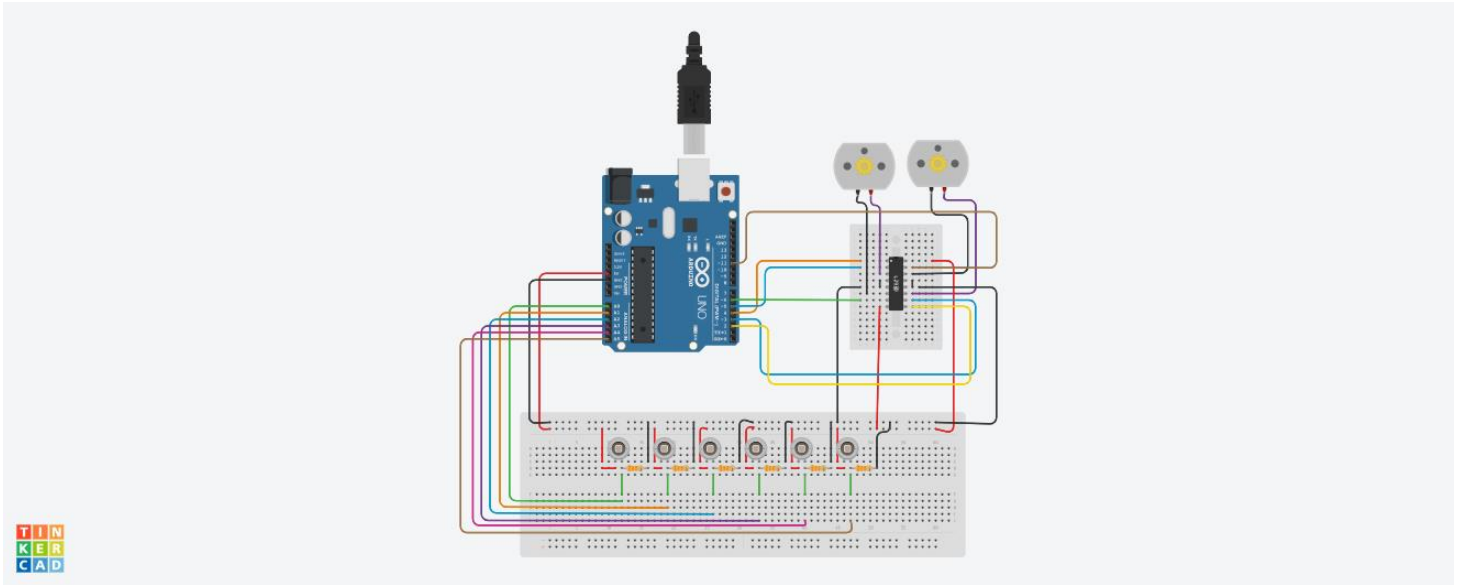


- Resistor 33k Ohm



## 6. HASIL PRAKTIKUM

### - FOTO RANGKAIAN



### - KODE PROGRAM

```
//inisialisasi sensor
int sensor1 = A0;
int sensor2 = A1;
int sensor3 = A2;
int sensor4 = A3;
int sensor5 = A4;
int sensor6 = A5;
int baca_sensor[6];

//inisialisasi pin output
int pinEnable = 4;
int pinEnable2 = 2;

//inisialisasi motor kiri
int motor_in1 = 5;
int motor_in2 = 6;

//inisialisasi motor kanan
int motor_on1 = 3;
int motor_on2 = 11;

//untuk mendeteksi error
int error = 0;
```

```

//bagian setup pin

void setup()

{

    pinMode(sensor1, INPUT);

    pinMode(sensor2, INPUT);

    pinMode(sensor3, INPUT);

    pinMode(sensor4, INPUT);

    pinMode(sensor5, INPUT);

    pinMode(sensor6, INPUT);


    pinMode(pinEnable, OUTPUT);

    pinMode(pinEnable2, OUTPUT);

    pinMode(motor_in1, OUTPUT);

    pinMode(motor_in2, OUTPUT);

    pinMode(motor_on1, OUTPUT);

    pinMode(motor_on2, OUTPUT);

    Serial.begin(9600);

}


//part untuk membaca sensor

void readsensor(){

    baca_sensor[0] = analogRead(sensor1);

    baca_sensor[1] = analogRead(sensor2);

    baca_sensor[2] = analogRead(sensor3);

    baca_sensor[3] = analogRead(sensor4);

    baca_sensor[4] = analogRead(sensor5);

    baca_sensor[5] = analogRead(sensor6);


    delay(500);


    for(int i=0; i<=5; i++){

        Serial.print("Sensor ");

        Serial.print(i+1);

        Serial.print(": ");

        Serial.print(baca_sensor[i]);

        Serial.print("\n");

    }

}

//part program

void loop(){

```

```
readsensor();
```

```
Serial.print("Nilai Deteksi error : ");
```

```
Serial.println(error);
```

```
//Apabila sensor 1 dan 2 mendeteksi gelap, 0% motor kiri, 50% motor kanan
```

```
if (baca_sensor[0] < 34 && baca_sensor[1] < 34 &&
```

```
    baca_sensor[2] > 34 && baca_sensor[3] > 34 &&
```

```
    baca_sensor[4] > 34 && baca_sensor[5] > 34){
```

```
digitalWrite (pinEnable, HIGH);
```

```
digitalWrite (pinEnable2, HIGH);
```

```
analogWrite (motor_in1, 0);
```

```
analogWrite (motor_in2, 0);
```

```
analogWrite (motor_on1, 0.5*255);
```

```
analogWrite (motor_on2, 0);
```

```
}
```

```
//Apabila sensor 2 dan 3 mendeteksi gelap, 20% motor kiri, 50% motor kanan
```

```
if (baca_sensor[0] > 34 && baca_sensor[1] < 34 &&
```

```
    baca_sensor[2] < 34 && baca_sensor[3] > 34 &&
```

```
    baca_sensor[4] > 34 && baca_sensor[5] > 34){
```

```
digitalWrite (pinEnable, HIGH);
```

```
digitalWrite (pinEnable2, HIGH);
```

```
analogWrite (motor_in1, 0.2*255);
```

```
analogWrite (motor_in2, 0);
```

```
analogWrite (motor_on1, 0.5*255);
```

```
analogWrite (motor_on2, 0);
```

```
}
```

```
//Apabila sensor 3 dan 4 mendeteksi gelap, 60% pada kedua motor
```

```
if (baca_sensor[0] > 34 && baca_sensor[1] > 34 &&
```

```
    baca_sensor[2] < 34 && baca_sensor[3] < 34 &&
```

```
    baca_sensor[4] > 34 && baca_sensor[5] > 34){
```

```
digitalWrite (pinEnable, HIGH);
```

```
digitalWrite (pinEnable2, HIGH);
```

```
analogWrite (motor_in1, 0.6*255);
```

```
analogWrite (motor_in2, 0);
```

```
analogWrite (motor_on1, 0.6*255);
```

```
analogWrite (motor_on2, 0);
```

```
}
```

```
//Apabila sensor 4 dan 5, 50% motor kiri, 20% motor kanan
```

```
if (baca_sensor[0] > 34 && baca_sensor[1] > 34 &&  
    baca_sensor[2] > 34 && baca_sensor[3] < 34 &&  
    baca_sensor[4] < 34 && baca_sensor[5] > 34){  
digitalWrite (pinEnable, HIGH);  
digitalWrite (pinEnable2, HIGH);  
analogWrite (motor_in1, 0.5*255);  
analogWrite (motor_in2, 0);  
analogWrite (motor_on1, 0.2*255);  
analogWrite (motor_on2, 0);  
}
```

```
//Apabila sensor 5 dan 6, 50% Motor kiri, 0% motor kanan
```

```
if (baca_sensor[0] > 34 && baca_sensor[1] > 34 &&  
    baca_sensor[2] > 34 && baca_sensor[3] > 34 &&  
    baca_sensor[4] < 34 && baca_sensor[5] < 34){  
digitalWrite (pinEnable, HIGH);  
digitalWrite (pinEnable2, HIGH);  
analogWrite (motor_in1, 0.5*255);  
analogWrite (motor_in2, 0);  
analogWrite (motor_on1, 0);  
analogWrite (motor_on2, 0);  
}
```

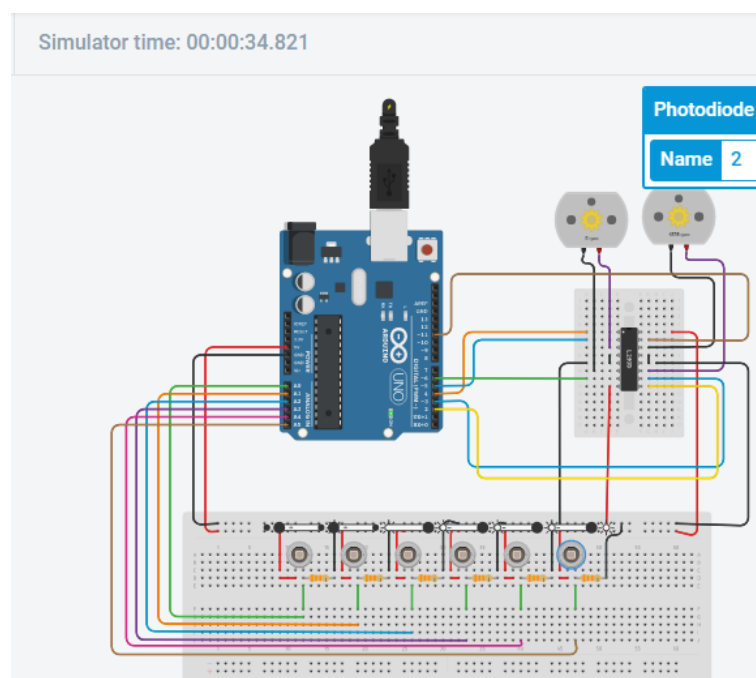
```
//Apabila semua sensor mendeteksi terang
```

```
if (baca_sensor[0] > 34 && baca_sensor[1] > 34 &&  
    baca_sensor[2] > 34 && baca_sensor[3] > 34 &&  
    baca_sensor[4] > 34 && baca_sensor[5] > 34){  
digitalWrite (pinEnable, HIGH);  
digitalWrite (pinEnable2, HIGH);  
analogWrite (motor_in1, 0);  
analogWrite (motor_in2, 0);  
analogWrite (motor_on1, 0);  
analogWrite (motor_on2, 0);  
}  
}
```

- TABEL

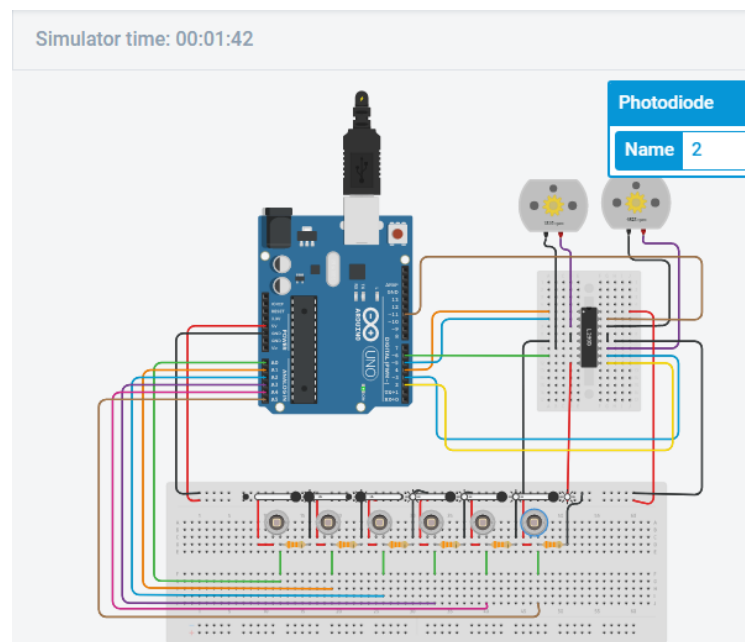
Nilai Sensor 1	Nilai Sensor 2	Nilai Sensor 3	Nilai Sensor 4	Nilai Sensor 5	Nilai Sensor 6	RPM Motor Kiri	RPM Motor Kanan
Sensor 1 dan 2 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 0% motor kiri, 50% motor kanan							
33	33	686	686	686	686	0	4873
Sensor 2 dan 3 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 20% motor kiri, 50% motor kanan							
686	33	33	686	686	686	1956	4873
Sensor 3 dan 4 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 60% pada kedua motor (kedua motor aktif)							
686	686	33	33	686	686	5873	5873
Sensor 4 dan 5 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 50% motor kiri, 20% motor kanan							
686	686	686	33	33	686	4873	1956
Sensor 5 dan 6 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 50% Motor kiri, 0% motor kanan							
686	686	686	686	33	33	4873	0
Semua sensor mendeteksi terang → <i>Duty cycle</i> kedua motor 0% (semua motor mati)							
686	686	686	686	686	686	0	0

- Sensor 1 dan 2 mendeteksi gelap, sisanya terang → *Duty cycle* 0% motor kiri, 50% motor kanan

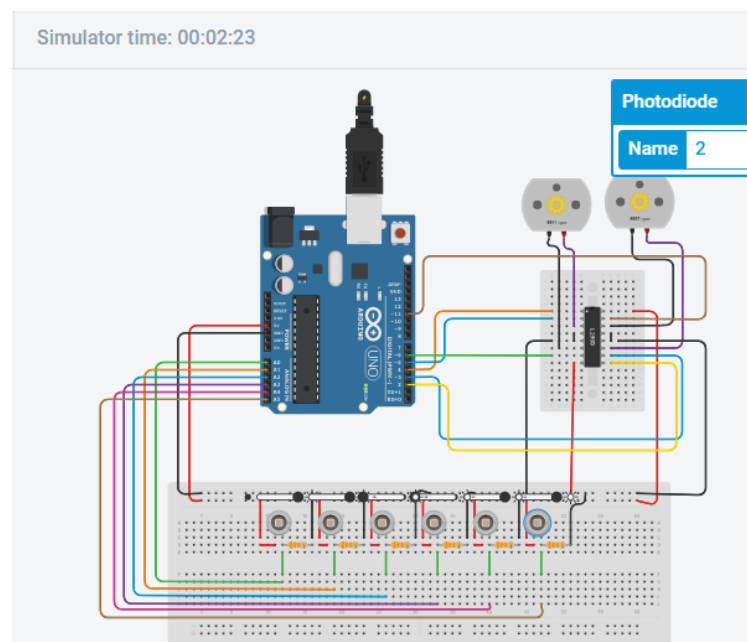




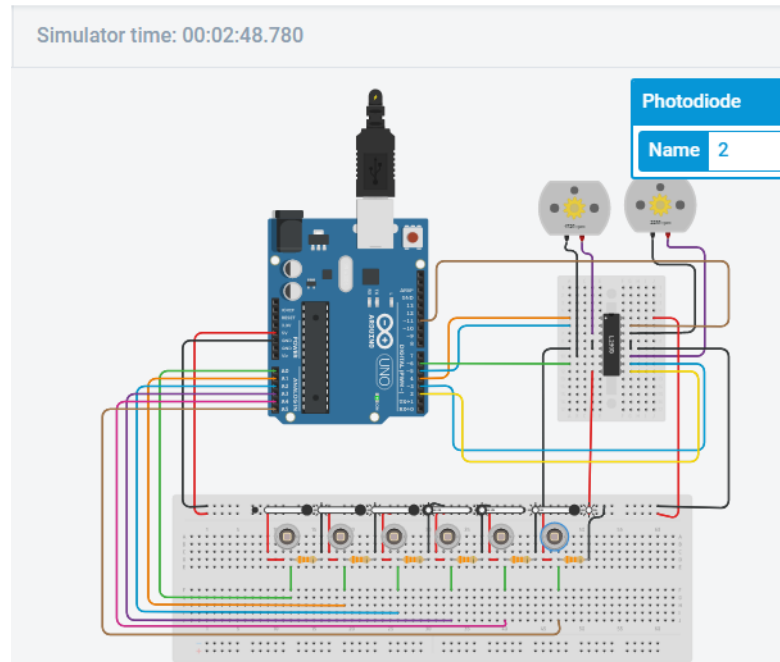
- Sensor 2 dan 3 mendeteksi gelap, sisanya terang → *Duty cycle* 20% motor kiri, 50% motor kanan



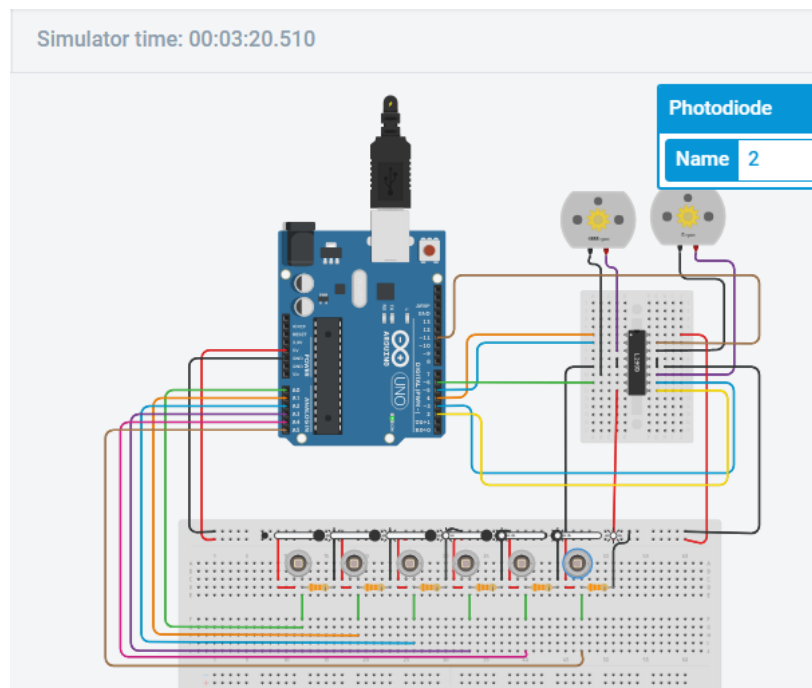
- Sensor 3 dan 4 mendeteksi gelap, sisanya terang → *Duty cycle* 60% pada kedua motor (kedua motor aktif)



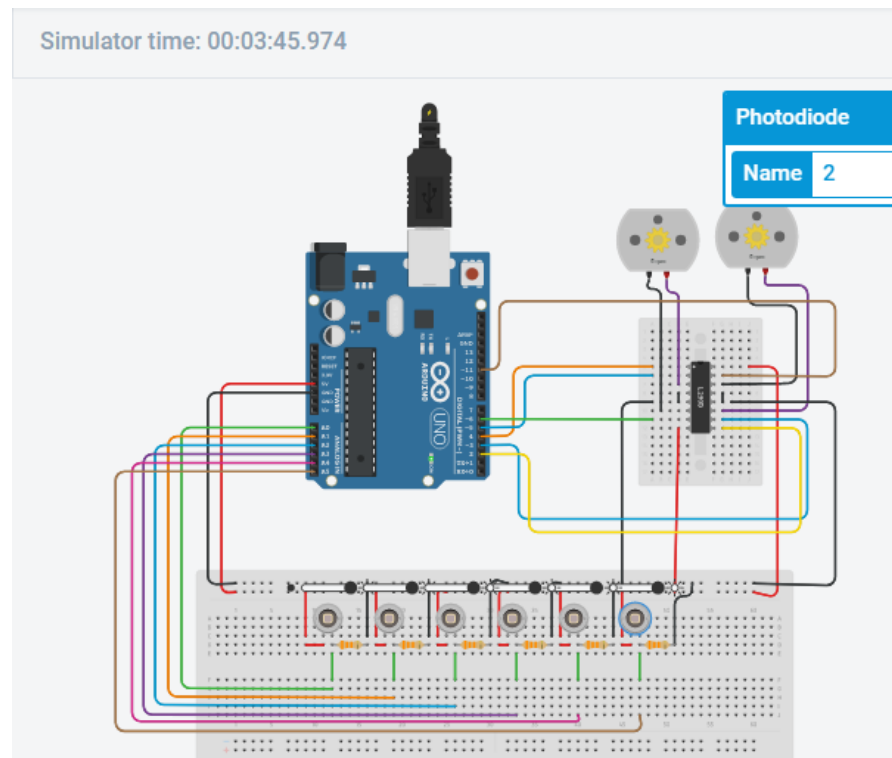
- Sensor 4 dan 5 mendeteksi gelap, sisanya terang → Duty cycle 50% motor kiri, 20% motor kanan



- Sensor 5 dan 6 mendeteksi gelap, sisanya terang → Duty cycle 50% Motor kiri, 0% motor kanan



- Semua sensor mendeteksi terang → Duty cycle kedua motor 0% (semua motor mati)



## 7. KESIMPULAN PRAKTIKUM

Dari hasil percobaan yang kita lakukan dan pembuatan rangkaian PWM kita bisa mengetahui ketika cahaa terang dan redup dan kecepatan motor yang bergerak dan bisa memudahkan user untuk mengetahuinya