

**MODUL 2**

**PRAKTIKUM INTERFACE, PERIPHERAL, DAN KOMUNIKASI**



Mata Kuliah : Sistem Kendali

Dosen : MHI

Kelas : D3TK-43-03

Anggota Kelompok :

- |                    |            |
|--------------------|------------|
| 1. Faidh Faziellah | 6702190059 |
| 2. M Ridzky Farhan | 6702190078 |

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER  
FAKULTAS ILMU TERAPAN  
UNIVERSITAS  
TELKOM BANDUNG  
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

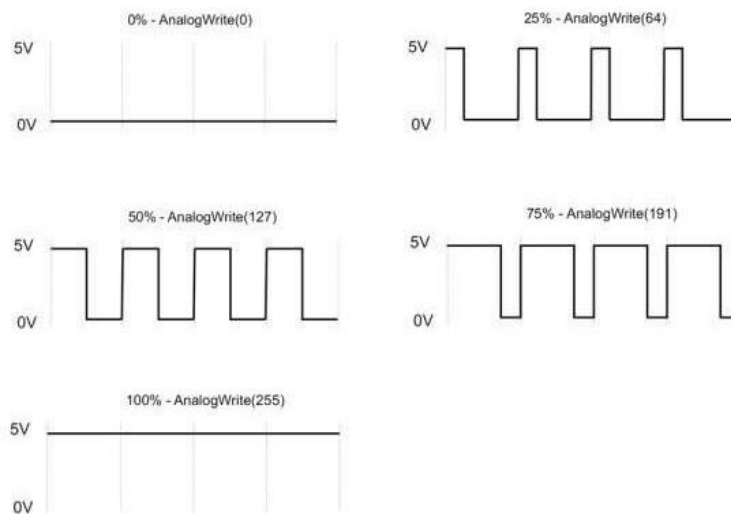
PWM adalah mekanisme untuk menghasilkan sinyal keluaran dengan siklus yang berulang antara level tinggi dan level rendah. Kita dapat mengontrol durasi sinyal level tinggi dan rendah sesuai kebutuhan. Siklus kerja adalah siklus sinyal level tinggi dan level rendah periode sinyal Persentase, produk dari siklus kerja akan sebanding dengan nilai rata-rata tegangan.

Pada robot linear follower, sistem kendali on / off tidak dapat digunakan untuk mengontrol kecepatan motor pada robot linear follower. Oleh karena itu, diperlukan PWM untuk mengatur kecepatan motor. Dengan menggunakan PWM, penyesuaian kecepatan motor dapat diubah dengan mengubah nilai siklus kerja pulsa. Denyut dari perubahan nilai duty cycle menentukan kecepatan motor. Ukuran dan frekuensi pulsa konstan, sedangkan ukuran duty cycle berubah sesuai dengan kecepatan yang dibutuhkan.

"Semakin besar duty cycle, semakin cepat kecepatan motor, dan  
Sebaliknya, semakin kecil duty cycle, semakin rendah kecepatan motor. "

Misal duty cycle dari pulse box adalah 50% Pada rangkaian elektronika digital setiap perubahan PWM akan dipengaruhi oleh resolusi PWM itu sendiri. Resolusi adalah besarnya perubahan nilai PWM. Misalnya resolusi PWM pada Arduino adalah 8 bits yang berarti 256 nilai PWM bervariasi pada range 0-255, dan variasi nilai pada range tersebut merepresentasikan tugas 0% -100%. siklus keluaran PWM. Beberapa kaki / pin Arduino mendukung fungsi PWM. Pin Arduino Nano yang mendukung PWM ditandai dengan tilde (~) di depan nomor pin, seperti 3, 5, 6, 9, 10, dan 11. Frekuensi yang digunakan Arduino untuk PWM adalah 500 Hz (500 siklus adalah 1 detik).

Untuk menggunakan PWM, kita bisa menggunakan fungsi `analogWrite()`. Nilai yang bisa dimasukkan dalam fungsi ini antara 0 dan 255. Nilai 0 berarti pulsa yang diberikan selalu 0 volt per siklus, dan nilai 255 berarti pulsa yang diberikan selalu 5 volt. Gambar berikut menunjukkan diagram fungsi `analogWrite()`.

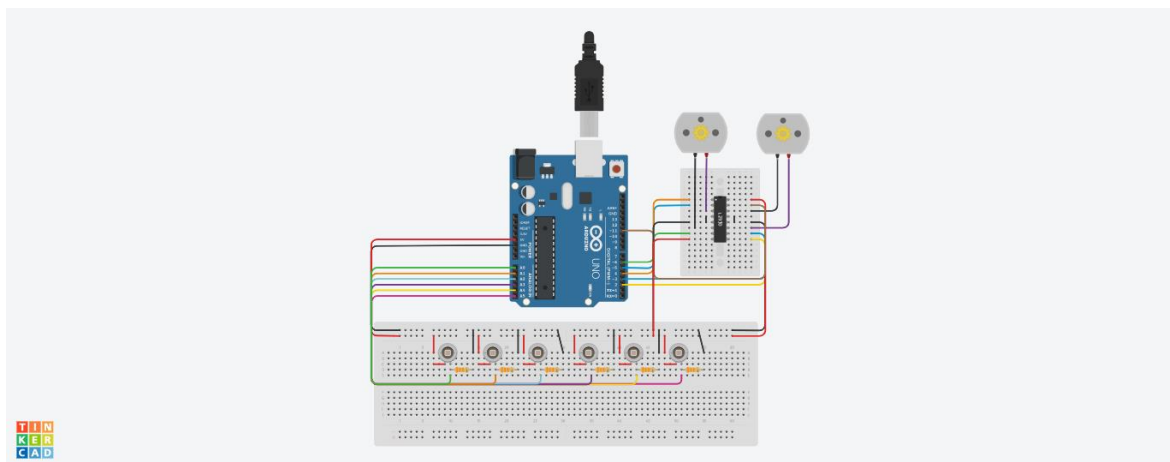


## 5. FOTO PERALATAN DAN BAHAN PRAKTIKUM



## 6. HASIL PRAKTIKUM

- FOTO RANGKAIAN



## - KODE PROGRAM

```
// PRATIUM MODUL 2 SISTEM PWM
// KELOMPOK 12 BINTANG JAYA
// Faidh Faziellah (6702190059)
// M Ridzky Farhan (6702190078)

//Deklarasi Pin yang digunakan untuk sensor
int sensor1 = A0;
int sensor2 = A1;
int sensor3 = A2;
int sensor4 = A3;
int sensor5 = A4;
int sensor6 = A5;
int baca_sensor[6];

//Variable bantuan untuk menyimpan memory a.k.a Error Detection Memory(EDM)
int LastError = 0;

//pin penggerak Motor
int pinEnable = 4; //Pin 1&2 harus HIGH
int pinEnable2 = 2; //Pin 3&4 harus HIGH

//pin Motorkiri
int motor_kiri1 = 5; //input motor driver
int motor_kiri2 = 6; //input motor driver

//pin Motorkanan
int motor_kanan1 = 3;
int motor_kanan2 = 11;

void setup(){
// Keenam Sensor Photodiode sebagai INPUT yaitu sensor cahaya
pinMode(sensor1, INPUT);
pinMode(sensor2, INPUT);
pinMode(sensor3, INPUT);
pinMode(sensor4, INPUT);
pinMode(sensor5, INPUT);
pinMode(sensor6, INPUT);

//Motor sebagai OOUTPUT penggerak
pinMode(pinEnable, OUTPUT);
pinMode(pinEnable2, OUTPUT);
pinMode(motor_kiri1, OUTPUT);
pinMode(motor_kiri2, OUTPUT);
pinMode(motor_kanan1, OUTPUT);
pinMode(motor_kanan2, OUTPUT);

Serial.begin(9600);
}

//Membaca sinyal analog dari 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(100);
    // Menampilkan data sensor ke Serial Monitor
    // Data sensor 1-6
    // Formatting tampilkan sensor

    for(int i=0; i<=5; i++){
        Serial.println(baca_sensor[i]);
        Serial.println(" ");
        Serial.println("\n");
        Serial.println("ERROR=");
        Serial.println(LastError);
    }
}

void loop(){
    readsensor();

    //Sensor 1 dan 2 mendeteksi gelap, sisanya terang □ Duty cycle 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);
        //analogWrite(pin,value 0-255)
        analogWrite(motor_kiri1,0);
```

```

analogWrite(motor_kiri2,0);

digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,127);
analogWrite(motor_kanan2,0);
}

//Sensor 2 dan 3 mendeteksi gelap, sisanya terang □ Duty cycle 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);
    //analogWrite(pin,value 0-255)
    analogWrite(motor_kiri1,0.2*255);
    analogWrite(motor_kiri2,0);

    digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,0.5*255);
    analogWrite(motor_kanan2,0);
}

//Sensor 3 dan 4 mendeteksi gelap, sisanya terang □ Duty cycle 60% pada kedua motor (kedua motor aktif)
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);
    //analogWrite(pin,value 0-255)
    analogWrite(motor_kiri1,0.6*255);
    analogWrite(motor_kiri2,0);

    digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,0.6*255);
    analogWrite(motor_kanan2,0);
}

//Sensor 4 dan 5 mendeteksi gelap, sisanya terang □ Duty cycle 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);
    //analogWrite(pin,value 0-255)
    analogWrite(motor_kiri1,0.5*255);
    analogWrite(motor_kiri2,0);

    digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,0.2*255);
    analogWrite(motor_kanan2,0);
}

//Sensor 5 dan 6 mendeteksi gelap, sisanya terang □ Duty cycle 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);
    //analogWrite(pin,value 0-255)
    analogWrite(motor_kiri1,0.5*255);
    analogWrite(motor_kiri2,0);

    digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,0);
    analogWrite(motor_kanan2,0);
}

//Semua sensor mendeteksi terang □ Duty cycle kedua motor 0% (semua motor mati)
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);
    //analogWrite(pin,value 0-255)
    analogWrite(motor_kiri1,0);
    analogWrite(motor_kiri2,0);

    digitalWrite(pinEnable2, HIGH);
    analogWrite(motor_kanan1,0);
    analogWrite(motor_kanan2,0);
}

//Studi Kasus Percobaan 2 : Error Condition --> Hanya 1 buah sensor terbaca
//Simpan kondisi terakhir pada suatu variabel

```

```

//Kondisi Sensor 1 saja yang membaca gelap
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);
  //analogWrite(pin,value 0-255)
  analogWrite(motor_kiri1,0);
  analogWrite(motor_kiri2,0);

  digitalWrite(pinEnable2, HIGH);
  analogWrite(motor_kanan1,127);
  analogWrite(motor_kanan2,0);
  Serial.print ("error = -2");
  LastError+=-2;

}

//Kondisi Sensor 2 saja yang membaca gelap
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);
  //analogWrite(pin,value 0-255)
  analogWrite(motor_kiri1,0.2*255);
  analogWrite(motor_kiri2,0);

  digitalWrite(pinEnable2, HIGH);
  analogWrite(motor_kanan1,0.5*255);
  analogWrite(motor_kanan2,0);
  Serial.print ("error = -1");
  LastError=- 1;

}

//Kondisi Sensor 3 saja yang membaca gelap
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)
{
  LastError+=1;

}

//Kondisi Sensor 4 saja yang membaca gelap
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)
{
  LastError+=1;

}

//Kondisi Sensor 5 saja yang membaca gelap
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)
{
  LastError+=1;

}

//Kondisi Sensor 6 saja yang membaca gelap
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)
{
  LastError+=1;

}

}

```

## - 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>Dutycycle</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% Makiri, 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% Makiri, 20% motor kanan							
686	686	686	33	33	686	4873	1956
Sensor 5 dan 6 mendeteksi gelap, sisanya terang → <i>Duty cycle</i> 50% Makiri, 0% motor kanan							
686	686	686	686	33	33	4873	0
Semua sensor mendeteksi terang → <i>Duty cycle</i> kedua motor 0% (semua mati)							
686	686	686	686	686	686	0	0

## 7. KESIMPULAN PRAKTIKUM

Sistem kendali PWM ini merupakan sistem kendali yang secara berkala mengulang sinyal repetitif tinggi dan rendah. Sangat cocok digunakan pada Robot Line Follower, karena jika hanya sistem kendali ON / OFF yang digunakan maka kecepatan motor tidak dapat dikendalikan. Melalui PWM ini anda dapat menyesuaikan ukuran dengan mengubah Nilai ukuran (siklus tugas pulsa) untuk mengubah kecepatan motor. Denyut dari perubahan nilai duty cycle menentukan kecepatan motor.

