

Sistem Kendali PID



Mata Kuliah : Interface, Peripheral, dan Komunikasi

Kode Dosen : AJR

Kelas : D3TK-43-02

Nama :

1. Topan Budiargo (6702190013)

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER
FAKULTAS ILMU TERAPAN
UNIVERSITAS TELKOM
BANDUNG
2021**

A. Tujuan

Maksud dan tujuan dari praktikum ini adalah :

1. Mahasiswa dapat memahami fungsi dan cara kerja PID pada motor DC
2. Mahasiswa dapat membuat program sistem kendali berbasis PID dengan error yang dihubungkan dengan konstanta proporsional

B. Alat dan Bahan

Alat dan Bahan :

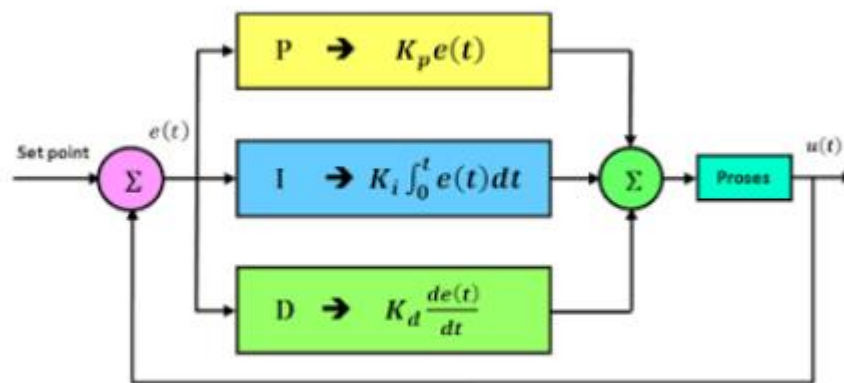
1. Robot Kit Line Follower
2. Baterai LiPo 2-Cell 1300 mAh
3. Kabel Mini-USB
4. Arduino Nano
5. Battery Checker
6. Battery Balancer

Perangkat Lunak :

1. Software IDE Arduino
2. Software Proteus/ Tinkercad (untuk simulasi)

C. Teori dasar

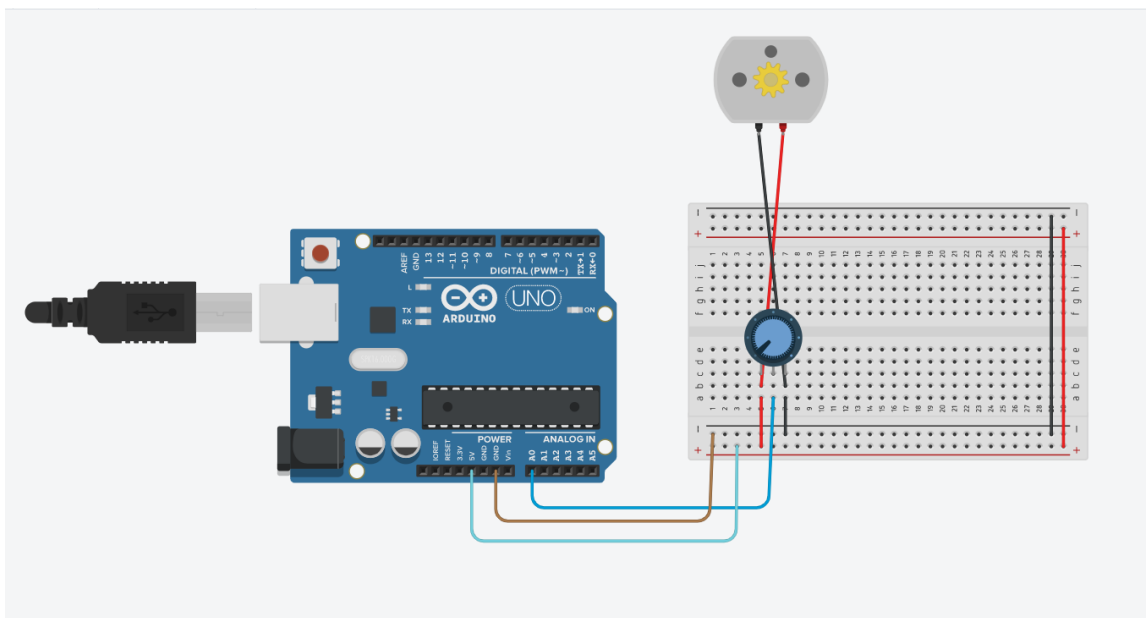
Teknik kendali PID adalah pengendali yang merupakan gabungan antara aksi kendali proporsional ditambah aksi kendali integral ditambah aksi kendali derivatif/turunan (Ogata, 1996). PID merupakan kependekan dari proportional integral derivative. Kombinasi ketiga jenis aksi kendali ini bertujuan untuk saling melengkapi kekurangan-kekurangan dari masing masing aksi kendali. Untuk memudahkan dalam memahami konsep teknik kendali PID silakan menyermati diagram blok pengendali PID pada gambar 1 di bawah ini.



Gambar 1. Diagram blok pengendali PID

Dalam aksi kendali PID, ada beberapa parameter variabel (dapat diubah/berubah) yang dapat dimanipulasi untuk tujuan menghasilkan aksi kendali terbaik dalam aplikasinya. Cara manipulasi parameter ini sering dinamakan dengan Manipulated Variable (MV). Dalam notasi matematikanya dapat ditulis dengan $MV(t)$ atau $u(t)$. Berikut persamaan matematik kendali PID.

D. Hasil Percobaan



Program :

```
float kp = 6, ki = 1, kd = 8; //Kostanta
int setpoint = 0; //Setpoint awal
int pv = 0; //PV awal = 0;
float integral = 0, derivative = 0;
int lastError = 0;
```

```

void setup()
{
    Serial.begin(9600);
    Serial.print(setpoint); Serial.print(", "); //Plot Setpoint Awal
    Serial.print(pv); Serial.print(", "); //Plot Present Value Awal
}

void loop()
{
    setpoint = analogRead(A0); // Atur Setpoint dengan Potensio
    int error = setpoint-pv; //Hitung Error
    derivative = error - lastError; //Hitung Rate / Derivative
    integral = integral + error; //Hitung Integral
    lastError = error; //Simpan Error Terakhir
    int pid = (kp * error) + (kd * derivative) + (ki * integral);
    //Hitung PID

    //-- Profile Actuator / Plant System --/
    int out = map(pid, -1023, 1023, -50, 50);
    if(out>50) out = 50;
    if(out<-50) out = -50;
    pv = pv + out;
    //-- End Profile Actuator // Plant System --/

    //Plot bentuk respon/
    Serial.print(setpoint); Serial.print(", "); //Plot Setpoint
    Serial.print(pv); Serial.print(", "); //Plot Present Value respon
    dari PID
    Serial.println();
}

```

Jadi pemberian nilai konstanta pada variable KP, KI dan KD akan mempengaruhi seberapa cepat program untuk mencari atau mencapai nilai setpoint yang dimana sudah ditentukan di program setpoint nya diatur menggunakan potensio.

Disini saya menggunakan konstanta KP=6, KI=1 dan KD=8
 Dengan potensio di set ke 962

Maka

nilai error = 962-0

derivative = 982-lastError

integral = integral + 962

lastError = error

int PID = (KP*error) + (KD*derivative) + (KI*integral)

= (6*962) + (8*962) + (1*962)

= 5772 + 7696 + 962

= 14430

E. Kesimpulan

Jadi penentuan nilai konstanta pada variable KP, KI dan KD akan sangat mempengaruhi seberapa cepat program untuk mencapai setpoint nya, dimana penentuan konstanta variable KP, KI dan KD adalah dengan cara try and error.

F. Link Video Kegiatan praktikum

<https://www.tinkercad.com/things/g6oT4w5np7m-swanky-wolt-duup/editel?sharecode=cIJ0j6cIINiK-yDN-zGBbmDlbg6GKErIXV7OKcwFUZc>