```
// Kendali RPM menggunakan PID

#define hall_sensorA 2
#define encoder_kal 2 // kalibrasi

#define PWM 6   // pin PWM
#define in1 7
#define in2 8

int interval = 60;
int motorSpeed = 0;
long previousMillis = 0;
long currentMillis =0;

int rpm =0;
int motorPWM =0;
int encoderValue=0;
int data=0;

int sp = 140;
float kp = 1;
float ki = 0.95;
float kd = 0;

float p, i, d;

int error =0;
int last_error =0;
int sum_error =0;

unsigned long lastTime;
int SampleTime;
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(hall_sensorA, INPUT_PULLUP);
  pinMode(PWM,OUTPUT);
  pinMode(in1,OUTPUT);
  pinMode(in2,OUTPUT);
  digitalWrite(in1,LOW);
  digitalWrite(in2,HIGH);
  encoderValue =0;
attachInterrupt(digitalPinToInterrupt(hall_sensorA), updateEncoder,
RISING); // pembacaan sensor encoder interupsi eksternal
  previousMillis = millis();
}

void loop()
{
  while(data<=100)
  {
```

```
    currentMillis = millis();
    if(currentMillis - previousMillis > interval) //
    {
      rpm = (float)((encoderValue*encoder_kal)); // kalibrasi rpm
      previousMillis = currentMillis;
      Serial.print(sp);
      Serial.print(",");
      Serial.println(rpm);
      encoderValue = 0;

      error = sp-rpm;
      sum_error = sum_error + error;

      p = kp*error;
      i = ki*sum_error;
      d = kd*(error-last_error);

      motorSpeed = p + i + d;

      if(motorSpeed > 250) motorSpeed = 250; // windup map()

      else if(motorSpeed <0) motorSpeed = 0;

      analogWrite(PWM, motorSpeed);
      last_error = error;
      data++;

    }
    analogWrite(PWM, motorSpeed);
  }
  analogWrite(PWM, 0);
}

void updateEncoder() // counter sensor
{
  encoderValue++;
}
```