

```
float KP = 5.5;
```

```
int setpoint = 820;
```

```
int position = 0;
```

```
int error = 0;
```

```
int saturation_min = -255;
```

```
int saturation_max = 255;
```

```
int output = 0;
```

```
// Process Serial
```

```
char HEADER_P ='P';
```

```
int MSG_LENGTH = 4;
```

```
int READ_P = setpoint;
```

```
int READ_P_OLD = 0;
```

```
// limitation
```

```
int uplimit = 880;
```

```
int downlimit = 720;
```

```
// timer
```

```
long present_time = 0;
```

```
long previous_time = 0;
```

```
int time_delay = 1000;
```

```
// PIN SETUP
```

```
int D1 = 10;
```

```
int D2 = 8;
```

```
int EN = 9;
```

```
void setup(){
```

```
  TCCR1B = TCCR1B & 0b11111000 | 0x01;
```

```
  pinMode(D1, OUTPUT);
```

```
  pinMode(D2, OUTPUT);
```

```
  pinMode(EN, OUTPUT);
```

```
  Serial.begin(9600);
```

```
}
```

```
void loop(){
```

```
  int val = processSerial();
```

```
  present_time = millis();
```

```
  if (present_time - previous_time > time_delay )
```

```
  {
```

```
    motor_stop();
```

```
  }
```

```
  if(present_time - previous_time < time_delay )
```

```
  {
```

```
    setpoint = READ_P;
```

```
    PID();
```

```
    motor_move();
```

```
}
```

```
if(val>0)
```

```
{
```

```
  if(READ_P != READ_P_OLD)
```

```
  {
```

```
    previous_time = present_time;
```

```
  }
```

```
}
```

```
  READ_P_OLD = READ_P;
```

```
}
```

```
void PID(){
```

```
  position = analogRead(0);
```

```
  error = setpoint - position;
```

```
  output = ( KP * error );
```

```
  // saturation
```

```
  if(output > saturation_max)
```

```
  {
```

```
    output = saturation_max;
```

```
  }
```

```
  if(output < saturation_min)
```

```
  {
```

```
    output = saturation_min;
```

```
  }
```

```
}
```

```
void motor_move(){
```

```
  if( output < 0 && position > downlimit) // DOWN
```

```
{
```

```
  digitalWrite(D1, HIGH);
```

```
  digitalWrite(D2, LOW);
```

```
  analogWrite(EN, output*(-1));
```

```
}
```

```
  else if( output > 0 && position < uplimit ) // UP
```

```
{
```

```
  digitalWrite(D1, LOW);
```

```
  digitalWrite(D2, HIGH);
```

```
  analogWrite(EN, output);
```

```
}
```

```
  else{
```

```
    motor_stop();
```

```
    //Serial.println("Motor Stop");
```

```
}
```

```
}
```

```
void motor_stop(){
```

```
  digitalWrite(D1, LOW);
```

```
  digitalWrite(D2, LOW);
```

```
  analogWrite(EN, LOW);
```

```

//Serial.println("MOTOR STOP");
}

int processSerial()
{
while(Serial.available() >= MSG_LENGTH ) // process messages when all characters are received
{
char check = Serial.read();
if( check == HEADER_P )
{
int val = 0;
for(int i =0; i < MSG_LENGTH-1; i++)
{
char ch = Serial.read();
if(ch >= '0' && ch <= '9'){ // is ch a number?
val = val * 10 + ch - '0'; // yes, accumulate the value
}
}
READ_P = val;
return 1;
}
}
return -1; // return -1 if nothing recieve
}

```