#include <QTRSensors.h>

#define SolMotorileri 9

#define SolMotorGeri 10

#define SagMotorileri 11

#define SagMotorGeri 12

#define ena 3

#define enb 5

#define KP .2

#define KD 5

#define M1\_minumum\_hiz 80

#define M2\_minumum\_hiz 80

#define M1\_maksimum\_hiz 85

#define M2\_maksimum\_hiz 85

#define MIDDLE\_SENSOR 3

#define NUM\_SENSORS 5

#define TIMEOUT 2500

#define EMITTER\_PIN 2

#define DEBUG 0

int lastError = 0;

int last\_proportional = 0;

int integral = 0;

QTRSensorsRC qtrrc((unsigned char[]) { A2, A3, A4, A5} ,NUM\_SENSORS, TIMEOUT, EMITTER\_PIN);

unsigned int sensorValues[NUM\_SENSORS];

void setup() {

pinMode(SolMotorileri,OUTPUT);

pinMode(SolMotorGeri,OUTPUT);

pinMode(SagMotorileri,OUTPUT);

pinMode(SagMotorGeri,OUTPUT);

pinMode(ena,OUTPUT);

pinMode(enb,OUTPUT);

delay(1500);

manual\_calibration();

set\_motors(0,0);

}

void loop() {

unsigned int sensors[5];

int position = qtrrc.readLine(sensors);

int error = position - 3000;

int motorSpeed = KP \* error + KD \* (error - lastError);

lastError = error;

int leftMotorSpeed = M1\_minumum\_hiz + motorSpeed;

int rightMotorSpeed = M2\_minumum\_hiz - motorSpeed;

set\_motors(leftMotorSpeed, rightMotorSpeed);

}

void set\_motors(int motor1speed, int motor2speed)

{

if (motor1speed > M1\_maksimum\_hiz ) motor1speed = M1\_maksimum\_hiz; //MAKSİMUM MOTOR 1 HIZ LİMİTİ

if (motor2speed > M2\_maksimum\_hiz ) motor2speed = M2\_maksimum\_hiz; // MAKSİMUM MOTOR 2 HIZ LİMİTİ

if (motor1speed < 0) motor1speed = 30; // MİNIMUMMOTOER 1 HIZ LİMİTİ

if (motor2speed < 0) motor2speed = 30; // MİNİMUM MOTOR 2 HIZ LİMİTİ

analogWrite(ena,motor1speed);

analogWrite(enb,motor2speed);

digitalWrite(SolMotorileri,HIGH);

digitalWrite(SagMotorileri,HIGH);

digitalWrite(SolMotorGeri,LOW);

digitalWrite(SagMotorGeri,LOW);

}

void manual\_calibration() {

int i;

for (i = 0; i < 250; i++)

{

qtrrc.calibrate(QTR\_EMITTERS\_ON);

delay(20);

}

if (DEBUG) {

Serial.begin(9600);

for (int i = 0; i < NUM\_SENSORS; i++)

{

Serial.print(qtrrc.calibratedMinimumOn[i]);

Serial.print(' ');

}

Serial.println();

for (int i = 0; i < NUM\_SENSORS; i++)

{

Serial.print(qtrrc.calibratedMaximumOn[i]);

Serial.print(' ');

}

Serial.println();

Serial.println();

}

}