float Kp =0.7;

float Kd = 65;

float Ki =0.1;

int Rint = 8;

int Rext = 40;

unsigned long time = 0; // tiempo de ejecucion del ultimo ciclo

int period = 20; // Periodo de muestreo en ms

int sensorPin=0; //Pin Analogico donde esta conectada la señal del Sensor de distancia

int measure; // Lo que mide el sensor. Son ADCs.

int dcal [] = {-218, -143, -95, -47, 0, 35, 75, 115, 155};

int ADCcal [] = {281, 312, 339, 380, 433, 476, 578, 754, 973};

int lastDist; // Valor anterior de Distancia para calcular la Velocidad

int dist; // distancia en mm con el 0 en el centro de la barra

int nvel=5; // numero de valores de velocidad sobre los que calculamos la media

int v[5];

int vel; // valor medio de las nvel velocidades ultimas

float I; // Valor Integral

#include <Servo.h>

Servo myservo; // create servo object to control a servo

float pos;

float reposo = 1450; // valor que mantiene la barra horizontal

float salida;

void setup()

{

analogReference(EXTERNAL); // AREF conectado a 3.3V

myservo.attach(7); // attaches the servo on pin X to the servo object

Serial.begin(115200);

}

void loop()

{

if (millis()>time+period){ // ¿Ha transcurrido el periodo?

time = millis();

// Medimos DISTANCIA

measure = analogRead(sensorPin);

measure = constrain(measure, ADCcal[0], ADCcal[8]);

lastDist = dist; // Guardamos el valor anterior de dist para calcular la velocidad

for(int i =0; i<8; i++){ // Aplicamos curva de Calibracion de ADC a mm

if (measure >= ADCcal[i] && measure< ADCcal[i+1]){

dist = map(measure,ADCcal[i],ADCcal[i+1],dcal[i],dcal[i+1]);

}

}

// Calculo de la media de la VELOCIDAD

for (int i=0; i<nvel-1; i++){ // Movemos todas hacia la izq para dejar libre la ultima.

v[i] =v[i+1];

}

v[nvel-1]= (dist - lastDist); // Ponemos un dato nuevo

vel=0;

for (int i=0; i<nvel; i++){ // Calculamos la media

vel = vel+ v[i];

}

vel = vel/nvel;

// Integral

if(abs(dist)>Rint && abs(dist)<Rext){ // Solo si esta dentro de (-Rext,Rext) y fuera de (-Rint,Rint)

I=I+dist\*Ki;

}

else {

I=0;

}

// Calculamos posicion del servo

pos=Kp\*dist+Kd\*vel+I;

salida=reposo+pos;

if(salida<=1600 && salida>=1300){

myservo.writeMicroseconds(salida);

}

else{

if(salida>1600){

myservo.writeMicroseconds(1600);

}

else{

if(salida<1300){

myservo.writeMicroseconds(1300);

}

}

}

}

}