Beschreibung:

Das HTL-Car soll mit Hilfe der Ultraschallsensoren einen Parkour fahren. Es fährt, je nach dem welcher Sensor näher zur welchen seitlichen Wand ist, eine Links- oder Rechtskurve. Wenn keine Seitenwände gibt soll es geradeaus fahren.

Code:

#include <FahrzeugLib.h>

#include <SoftwareSerial.h>

// Bei den Trigger und Echo Anschlüssen der Ultraschallsensoren werden die Anschlüsse definiert.

const int trigPinmitte = 12;

const int echoPinmitte = 11;

const int trigPinrechts = 10;

const int echoPinrechts = 9;

const int trigPinlinks = 7;

const int echoPinlinks = 6;

// Anschlüsse der beiden Motoren:

CMotor MotorLinks(3, 2);

CMotor MotorRechts(5, 4);

CAntrieb Antrieb(&MotorLinks, &MotorRechts);

long durationmitte;

int distancemitte;

long durationrechts;

int distancerechts;

long durationlinks;

int distancelinks;

void setup() {

pinMode(trigPinmitte, OUTPUT);

pinMode(echoPinmitte, INPUT);

pinMode(trigPinrechts, OUTPUT);

pinMode(echoPinrechts, INPUT);

pinMode(trigPinlinks, OUTPUT);

pinMode(echoPinlinks, INPUT);

Serial.begin(9600);

}

void loop() {

// Distanz vom mittleren Sensor wird ausgegeben und in einer Variable gespeichert.

digitalWrite(trigPinmitte, LOW);

delayMicroseconds(2);

digitalWrite(trigPinmitte, HIGH);

delayMicroseconds(10);

digitalWrite(trigPinmitte, LOW);

durationmitte = pulseIn(echoPinmitte, HIGH);

distancemitte = durationmitte\*0.034/2;

Serial.print("Distance - Mitte: ");

Serial.println(distancemitte);

// Distanz vom rechten Sensor wird ausgegeben und in einer Variable gespeichert.

digitalWrite(trigPinrechts, LOW);

delayMicroseconds(2);

digitalWrite(trigPinrechts, HIGH);

delayMicroseconds(10);

digitalWrite(trigPinrechts, LOW);

durationrechts = pulseIn(echoPinrechts, HIGH);

distancerechts = durationrechts\*0.034/2;

Serial.print("Distance - Rechts: ");

Serial.println(distancerechts);

// Distanz vom linken Sensor wird ausgegeben und in einer Variable gespeichert.

digitalWrite(trigPinlinks, LOW);

delayMicroseconds(2);

digitalWrite(trigPinlinks, HIGH);

delayMicroseconds(10);

digitalWrite(trigPinlinks, LOW);

durationlinks = pulseIn(echoPinlinks, HIGH);

distancelinks = durationlinks\*0.034/2;

Serial.print("Distance - Links: ");

Serial.println(distancelinks);

// Wenn alle drei Sensoren eine Distanz von 15cm von einem Hindernis entfernt sind soll das Auto geradeaus mit der Geschwindigkeit 100 fahren

if (((distancemitte > 15) || (distancerechts > 15 )) || (distancelinks > 15)) {

Antrieb.Gerade(true,100);

}

// Wenn der rechte Sensor eine Distanz von kleiner gleich 15cm erkennt dann soll das Auto eine Kurve nach links mit der Schärfe 5 fahren.

if(distancerechts<=15)

{

Antrieb.Kurve(5, 100, true);

}

// Wenn der linke Sensor eine Distanz von kleiner gleich 15cm erkennt dann soll das Auto eine Kurve nach rechts mit der Schärfe 5 fahren.

if(distancelinks<=15)

{

Antrieb.Kurve(-5, 100, true);

}

}