***PROJETO GARAGEM***

//DEFINIÇÕES DE PORTAS:

#define trig1 4

#define echo1 7

#define trig2 5

#define echo2 8

#define trig3 6

#define echo3 9

#define LED1 2

#define LED2 10

#define relay1 13

#define relay2 12

//DEFINIÇÕES DE TEMPO-DIREÇÕES:

#define M 280 //MUITO

#define d 30 //desconto de cargas de baterias com diferencial

#define I 150 //INTERMEDIARIO

#define E 5 //tempo de ESCAPE

//DEFINIÇÕES DE DISTANCIA LIMITE (K):

#define K1 40 //distancia limite para começar a ler os "ifs"

#define K2 K1+20

#define K3 K2+20

unsigned long trocaET;

unsigned long trocaDT;

unsigned long trocaDT1;

unsigned long trocaep;

unsigned long trocadp;

unsigned long trocadp1;

unsigned long trocadP1;

#define ET M+20

#define ETV M-55

#define DT M

#define DTV M+10

#define ep (M+15)/2

#define epv (M-50)/2

#define dp M/2

#define dpv (M)/2

#define eP (M+20)/3

#define ePv (M-50)/3

#define dP M/3

#define dPv M/3

void setup() {

/\*

Serial.begin(9600);

Serial.flush();

Serial.println("INICIANDO...");

\*/

pinMode(trig1, OUTPUT);

pinMode(echo1, INPUT);

pinMode(trig2, OUTPUT);

pinMode(echo2, INPUT);

pinMode(trig3, OUTPUT);

pinMode(echo3, INPUT);

pinMode(LED1, OUTPUT);

pinMode(LED2, OUTPUT);

digitalWrite(LED1, HIGH);

digitalWrite(LED2, HIGH);

pinMode(relay1, OUTPUT);

pinMode(relay2, OUTPUT);

digitalWrite(relay1, HIGH);

digitalWrite(relay2, HIGH);

delay(500);

}

void loop() {

long duration1, distance1;

digitalWrite(trig1, LOW);

delayMicroseconds(2);

digitalWrite(trig1, HIGH);

delayMicroseconds(10);

digitalWrite(trig1, LOW);

duration1 = pulseIn(echo1, HIGH);

distance1 = (duration1/2) / 29.1;

long duration2, distance2;

digitalWrite(trig2, LOW);

delayMicroseconds(2);

digitalWrite(trig2, HIGH);

delayMicroseconds(10);

digitalWrite(trig2, LOW);

duration2 = pulseIn(echo2, HIGH);

distance2 = (duration2/2) / 29.1;

long duration3, distance3;

digitalWrite(trig3, LOW);

delayMicroseconds(2);

digitalWrite(trig3, HIGH);

delayMicroseconds(10);

digitalWrite(trig3, LOW);

duration3 = pulseIn(echo3, HIGH);

distance3 = (duration3/2) / 29.1;

/\*

Serial.print("distance1:"); // indicador de distancias via serial

Serial.println(distance1);

delay(50);

Serial.print("distance2:");

Serial.println(distance2);

delay(50);

Serial.print("distance3:");

Serial.println(distance3);

delay(50);

\*/

if (distance1<K1 || distance2<K1 || distance3<K1) // SE ALGUMA SENSOR TIVER MENOS QUE K1(distancia limite)cm, faça a leitura:

{

if (distance1<distance2 && distance1<distance3) //COMANDOS PARA IR PRA "ESQUERDA TOTAL" se distance1 for a menor

{

for(int x=0; x<2; x++) //COMANDO DE INDICADOR LED

{

digitalWrite(LED2, LOW);

delay((M+d)/5);

digitalWrite(LED2, HIGH);

delay((M+d)/5);

}

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(ET);

digitalWrite(relay2, HIGH);

trocaET = millis();

}

if (distance1<distance2 && distance1<distance3 && trocaET>M)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(ETV);

digitalWrite(relay1, HIGH);

}

trocaET = millis();

delay(E);

}

if (distance3<distance1 && distance3<distance2) //COMANDOS PARA IR PRA "DIREITA TOTAL" se distance3 for a menor//perfeito!

{

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(DT);

digitalWrite(relay1, HIGH);

trocaDT = millis();

}

if (distance3<distance1 && distance3<distance2 && trocaDT>M)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(DTV);

digitalWrite(relay2, HIGH);

}

trocaDT = millis();

delay(E);

}

if (distance2<distance1 && distance2<distance3) //COMANDOS PRA DIREITA TOTAL se a distance2 for a menor

{

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(DT);

digitalWrite(relay1, HIGH);

trocaDT = millis();

}

if (distance2<distance1 && distance2<distance3 && trocaDT>M)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(DTV);

digitalWrite(relay2, HIGH);

}

trocaDT = millis();

delay(E);

}

}

else if (distance1<K2 || distance2<K2 || distance3<K2) //K2

{

if (distance1<distance2 && distance1<distance3){ // ir pra esquerda parcial

for(int x=0; x<2; x++)

{

digitalWrite(LED2, LOW);

delay((M+d)/5);

digitalWrite(LED2, HIGH);

delay((M+d)/5);

}

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(ep);

digitalWrite(relay2, HIGH);

trocaep = millis();

}

if (distance1<distance2 && distance1<distance3 && trocaep>I)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(epv);

digitalWrite(relay1, HIGH);

}

trocaep = millis();

delay(E);

}

if (distance3<distance1 && distance3<distance2){ // ir pra direita parcial

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(dp);

digitalWrite(relay1, HIGH);

trocadp = millis();

}

if (distance3<distance1 && distance3<distance2 && trocadp>I)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(dpv);

digitalWrite(relay2, HIGH);

}

trocadp = millis();

delay(E);

}

if (distance2<distance1 && distance2<distance3) //COMANDOS PRA direita parcial, se a distance2 for a menor

{

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(dp);

digitalWrite(relay1, HIGH);

trocadp1 = millis();

}

if (distance2<distance1 && distance2<distance3 && trocadp1>M)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(dpv);

digitalWrite(relay2, HIGH);

}

trocadp1 = millis();

delay(E);

}

}

else if (distance1<K3 || distance2<K3 || distance3<K3) //K3

{

if (distance1<distance2 && distance1<distance3){ // ir pra esquerda parcial/2

for(int x=0; x<2; x++)

{

digitalWrite(LED2, LOW);

delay((M+d)/5);

digitalWrite(LED2, HIGH);

delay((M+d)/5);

}

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(eP);

digitalWrite(relay2, HIGH);

trocaep = millis();

}

if (distance1<distance2 && distance1<distance3 && trocaep>I)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(ePv);

digitalWrite(relay1, HIGH);

}

trocaep = millis();

delay(E);

}

if (distance3<distance1 && distance3<distance2){ // ir pra direita parcial/2

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(dP);

digitalWrite(relay1, HIGH);

trocadp = millis();

}

if (distance3<distance1 && distance3<distance2 && trocadp>I)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(dPv);

digitalWrite(relay2, HIGH);

}

trocadp = millis();

delay(E);

}

if (distance2<distance1 && distance2<distance3) //COMANDOS PRA direita parcial, se a distance2 for a menor

{

for(int x=0; x<2; x++)

{

digitalWrite(LED1, LOW);

delay((M+d)/5);

digitalWrite(LED1, HIGH);

delay((M+d)/5);

}

digitalWrite(relay2, HIGH);

digitalWrite(relay1, LOW);

delay(dP);

digitalWrite(relay1, HIGH);

trocadP1 = millis();

}

if (distance2<distance1 && distance2<distance3 && trocadP1>M)

{

for(int x=0; x<1; x++)

{

digitalWrite(relay1, HIGH);

digitalWrite(relay2, LOW);

delay(dPv);

digitalWrite(relay2, HIGH);

}

trocadP1 = millis();

delay(E);

}

}

else {

digitalWrite(relay1, HIGH);

digitalWrite(relay2, HIGH);

digitalWrite(LED1, HIGH);

digitalWrite(LED2, HIGH);

}

}