

Controle de Portão de Garagem

Sistemas Embarcados e Prototipação

Matheus Henrique de Arruda

Lógica de Implementação

- A lógica se baseia no controle de abertura e fechamento de um portão de Garagem
- A movimentação depende de um motor dc acionado por um único botão.
- Dados os feedbacks dos fins de curso, o portão abrirá ou fechará.



Bibliotecas e Variáveis

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <dht.h>

int trig = 11;
int echo = 10;
float distancia;
//Definicoes pinos Arduino ligados a entrada da Por
int IN1 = 8;
int IN2 = 9;

dht DHT;

//Crear el objeto lcd dirección 0x3F y 16 columnas
LiquidCrystal_I2C lcd(0x27,16,2); //
```

Bibliotecas e Variáveis

Declaração das variáveis e chamado das funções

```
void setup()
{
    //Define os pinos como saída
    pinMode(IN1, OUTPUT);
    pinMode(IN2, OUTPUT);
    pinMode(2, INPUT_PULLUP); //Fim de Curso Esq (Aber
    pinMode(3, INPUT_PULLUP); //Fim de Curso Dir (Fech
    pinMode(4, INPUT_PULLUP); //Botão de acionamento

    pinMode (trig, OUTPUT);
    pinMode (echo,INPUT);

    pinMode(A3, INPUT);

    // Inicializar el LCD
    lcd.init();
    //Encender la luz de fondo.
    lcd.backlight();

    Serial.begin(9600);
}

void loop()
{
    Serial.println(digitalRead(IN1));
    distancia = Ultra_soni();
    Controle_Portao(distancia);
    LCD();
}
```


Funções Implementadas

- Função de Controle do motor DC

```
void Controle_Portao(int distancia){

    //Abertura Portão
    if(digitalRead(3) == 0 && digitalRead(4) == 0){

        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, HIGH);
        delay(100);
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, LOW);
    }
    //Fechamento Portão
    else if(digitalRead(2) == 0 && digitalRead(4) == 0){
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, HIGH);
        delay(100);
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, HIGH);
    }
    //Comando parada Fim de Curso
    else if(digitalRead(2) == 0 || digitalRead(3) == 0){
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, HIGH);
    }
    //Detecção de carro entrando na garagem
    if(distancia <20 && digitalRead(IN1) == 0){
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, HIGH);
        delay(1000);
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, LOW);
        delay(1000);
    }
}
```

Funções Implementadas

- Função do Sensor Ultrassônico.

```
float Ultra_soni(){
    //Inicializando o sensor
    digitalWrite(trig, LOW);
    delay(0005);

    //enviando o sinal por 10 microseg
    digitalWrite(trig, HIGH);
    delay(0010);

    //Interrompendo o sinal
    digitalWrite(trig, LOW);

    //ler o tempo de viagem da onda em microseg
    distancia = pulseIn (echo, HIGH);

    distancia = distancia/58; // Convertendo de micros

    return distancia;
}
```

Funções Implementadas

- Controle do Display LCD para feedback do processo

```
void LCD(){
    DHT.read11(A3);

    lcd.print("Umidade ");
    lcd.print(DHT.humidity);

    lcd.setCursor(0,1);
    lcd.print("Temperatura ");
    lcd.print(DHT.temperature);
    delay(1000);
    lcd.clear();

    if(digitalRead(3) == 0){
        lcd.setCursor(0,0);
        lcd.print("Portao Fechado");
        delay(1000);
        lcd.clear();
    }
    else if(digitalRead(2) == 0){
        lcd.setCursor(0,0);
        lcd.print("Portao Aberto");
        delay(1000);
        lcd.clear();
    }
    else{
        lcd.setCursor(0,0);
        lcd.print("Portao em Movimento");
        delay(1000);
        lcd.clear();
    }
}
```

