

Controle de braços robóticos auxiliados por visão computacional.

Marco Reis (orientador)

SENAI CIMATEC

Salvador, Brasil

marco.reis@aln.senaicimatec.edu.br

Tiago Barretto Sant'Anna

SENAI CIMATEC

Salvador, Brasil

tiago.sant'anna@ba.estudante.senai.br

Abstract—O objetivo deste artigo é expor o desafio de comunicação entre dois Arduinos. Portanto para cumprir com o desafio é necessário receber os dados de distancia através de um sensor ultrassônico de acordo com distancia acende determinado LED e envia o sinal para outro Arduino que exibe em um display LCD. Esse desafio tem como objetivo o aprendizado sobre programação, uso de sensores e comunicação entre dispositivos. Por fim foi possível realizar essa tarefa e adquirir os conhecimentos necessários.

Index Terms—Arduino, comunicação, sensores, simulação.

I. INTRODUÇÃO

Arduino é uma placa de prototipagem com um microprocessador que permite a realização de projetos de eletrônica com alta facilidade [1]. Dessa forma, busca-se realizar uma atividade para aumentar o conhecimento nessas tecnologias. Assim foi realizado uma atividade que consiste em utilizar dois Arduino para se comunicar, um deles sendo responsável por enviar os dados de um sensor ultrassônico e o outro para receber e mostrar em um display lcd os valores de distancia. Com isso busca-se adquirir conhecimentos de programação, uso de sensores e comunicação.

II. DESENVOLVIMENTO

A. Materiais utilizados

- 2 Arduinos UNO;
- 2 protoboards;
- display 16x2;
- jumpers;
- potenciômetro;
- resistores;
- LEDs de cor vermelha amarela e branca.

B. Métodos

Para a realização do desafio foi necessário realizar um estudo acerca do uso do display no Arduino [2]. Logo em seguida foi necessário fazer um estudo sobre como se comunicar com as portas seriais para comunicar com 2 arduinos [3]. Por fim, foi feito um estudo do funcionamento do sensor ultrassônico e como obter seus dados [3].

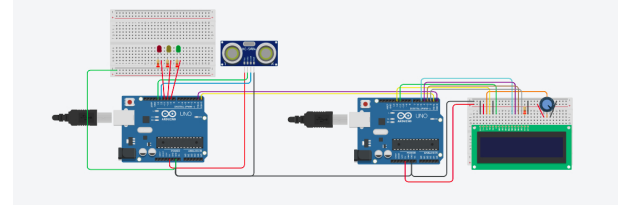


Fig. 1. Exemplos de tags [4]

C. Funcionamento do circuito

Como mostrado na figura 1 o primeiro Arduino é conectado no sensor ultrassônico e nos LEDs. A função deste Arduino é de receber os dados do sensor, acender os LEDs de acordo com a distancia e enviar os dados da distancia para o segundo Arduino.

Desse modo, o segundo Arduino, esta conectado no primeiro e também no display. Sua função é de receber os dados do primeiro e mostra-los no display. O potenciômetro conectado no display tem uma função de controlar o contraste de fundo do display.

D. Código do primeiro Arduino

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 // distance sensor
6 int trigger = 13, echo = 12;
7 int cm = 0;
8 char num[10];
9
10 //leds
11 int red = 11, yllw = 10, grn = 9;
```

Essa primeira parte do código declara as bibliotecas e variáveis utilizadas no código.

```
1 long readUltrasonicDistance(int triggerPin, int echoPin)
2 {
3   pinMode(triggerPin, OUTPUT); // Clear the trigger
4   digitalWrite(triggerPin, LOW);
5   delayMicroseconds(2);
6   // Sets the trigger pin to HIGH state for 10 microseconds
7   digitalWrite(triggerPin, HIGH);
8   delayMicroseconds(10);
9   digitalWrite(triggerPin, LOW);
10  pinMode(echoPin, INPUT);
```

```

11 // Reads the echo pin, and returns the sound wave travel time in ↔
    microseconds
12 return pulseIn(echoPin, HIGH);
13 }

```

Essa função foi inspirada nos estudos feitos [5] e responde ao sensor ultrassônico. Ela funciona calculando o tempo da viagem da onda do sensor e multiplicando por uma constante.

```

1 void setup()
2 {
3   Serial.begin(9600);
4   pinMode(red, OUTPUT);
5   pinMode(yllw, OUTPUT);
6   pinMode(grn, OUTPUT);
7 }

```

Esse código define o *baud rate*, que é a taxa de transmissão de dados e os pinos de saída dos LEDs.

```

1 void loop()
2 {
3   // measure the ping time in cm
4   cm = 0.0175 * readUltrasonicDistance(trigger, echo);
5   itoa(cm,num,10);

```

Essa parte do código recebe o dado da distancia em valor *long* e transforma em dado do tipo *String* para que possa ser enviado pela porta serial.

```

1 if(cm < 100){
2   Serial.write("Regiao1:");
3   Serial.write(num);
4   Serial.write("");
5   digitalWrite(red, HIGH);
6
7 } else if(100 <= cm && cm < 200){
8   Serial.write("Regiao2:");
9   Serial.write(num);
10  digitalWrite(yllw, HIGH);
11
12 } else{
13   Serial.write("Regiao3:");
14   Serial.write(num);
15   digitalWrite(grn, HIGH);
16 }
17 Serial.write("\n");
18 delay(300);
19 digitalWrite(red, LOW);
20 digitalWrite(yllw, LOW);
21 digitalWrite(grn, LOW);
22 }

```

Essa parte é responsável por acender os LEDs e enviar os dados pela porta serial.

E. Código do segundo arduino

```

1 #include <LiquidCrystal.h>
2 // Reciever
3 char mystr[20]; //Initialized variable to store recieved data
4 char resp[20];
5 // Define os pinos que serao utilizados para ligacao ao display
6 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

```

Essa parte define a variável que recebe a resposta e os pinos do display.

```

1 void setup()
2 {

```

```

3 // Define o numero de colunas e linhas do LCD
4 lcd.begin(16, 2);
5 Serial.begin(9600);
6 }

```

Nessa parte é definido o *baud rate* e inicializa o display.

```

1 void loop()
2 {
3   if(Serial.available() > 0){
4     Serial.readBytes(mystr,11);
5     lcd.setCursor(0, 0);
6     if(mystr[0] == 'R'){
7       lcd.print(mystr);
8       delay(1000);
9     }
10  }
11
12  delay(300);
13
14  lcd.clear();
15 }

```

Esta parte consiste em verificar se a porta serial esta disponivel e logo depois verificar se o sinal esta sendo enviado da maneira correta.

III. CONCLUSÃO

Com a finalização do desafio foi possível compreender e obter conhecimentos acerca da programação em Arduino, uso do sensor ultrassônico e da comunicação serial.

REFERENCES

- [1] Arduino, "Arduino," <https://www.arduino.cc/>, (Accessed on 05/25/2022).
- [2] A. Thomsen, "Controlando um lcd 16x2 com arduino - filipeflopp," <https://www.filipeflopp.com/blog/controlando-um-lcd-16x2-com-arduino/>, 2011, (Accessed on 05/25/2022).
- [3] Iotguider, "Serial communication between two arduino boards," <https://iot-guider.com/arduino/serial-communication-between-two-arduino-boards/>, 2017, (Accessed on 05/25/2022).
- [4] "Opencv: Detection of aruco markers," https://docs.opencv.org/3.4/d5/dae/tutorial_aruco_detection.html, (Accessed on 03/06/2022).
- [5] E. Oliveira, "Como usar com arduino - sensor ultrasonico hc-sr04 - blog masterwalker shop," <https://blogmasterwalkershop.com.br/arduino/como-usar-com-arduino-sensor-ultrasonico-hc-sr04>, 2018, (Accessed on 05/25/2022).