



Campus Quissamã

Curso Integrado Informática

Professor: Renato

Turma: 2º ano informática

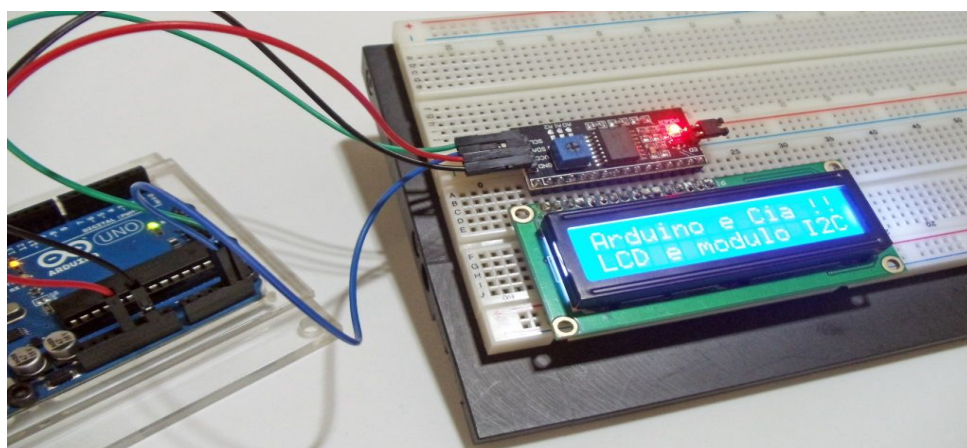
Aluno: Arthur França Freitas e Vitória Silva Nascimento Cabral

Assunto: Relatório I2C

- **O que é I2C?**

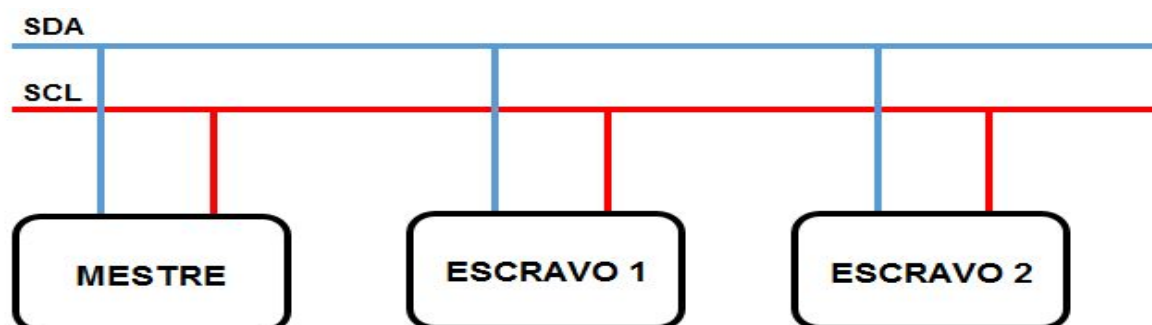
É um protocolo de comunicação.

O I2C é um barramento que foi criado por uma empresa da Holanda chamada (Philips) e pode ser como Inter IC ou I2C, responsável por realizar comunicações, podendo ser utilizado em Arduino ou outros Microcontroladores, serve de solução por disponibilizar várias portas, devidos a alguns projetos dependerem de mais entradas.



- **Para que serve?**

O I2C serve para se comunicar entre arduino e outros elementos microcontroladores, tendo como base a hierarquia chefe/escravo, por tanto, um deverá atuar como o principal e os outros devem seguir a premissa do mestre

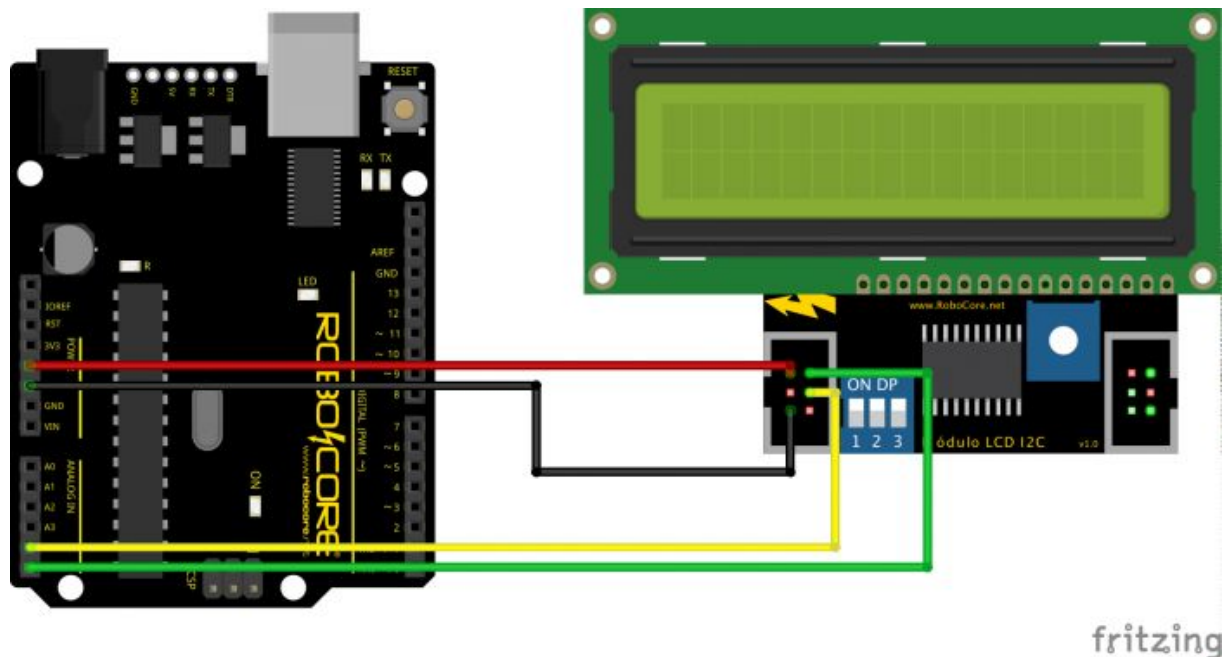


É notável nesta imagem dois barramentos, a qual fazem a comunicação, tendo em vista que, um deles é o serial data (SDA) e o outro é o Serial Clock (SCL).

O SDA é responsável pela troca de dados entre os dispositivos que estão sincronizados. Já o SCL é responsável possui a função de sincronizar os dispositivos.

- **Práticas:**

- 1) Este exemplo é uma simulação (Arduino) que é responsável por um código básico que mostra na tela uma informação, neste caso ilustrado, Hello World.



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

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

```
LiquidCrystal_I2C lcd(0x20,16,2); // Criando um LCD de 16x2 no endereço 0x20
```

```
void setup()
```

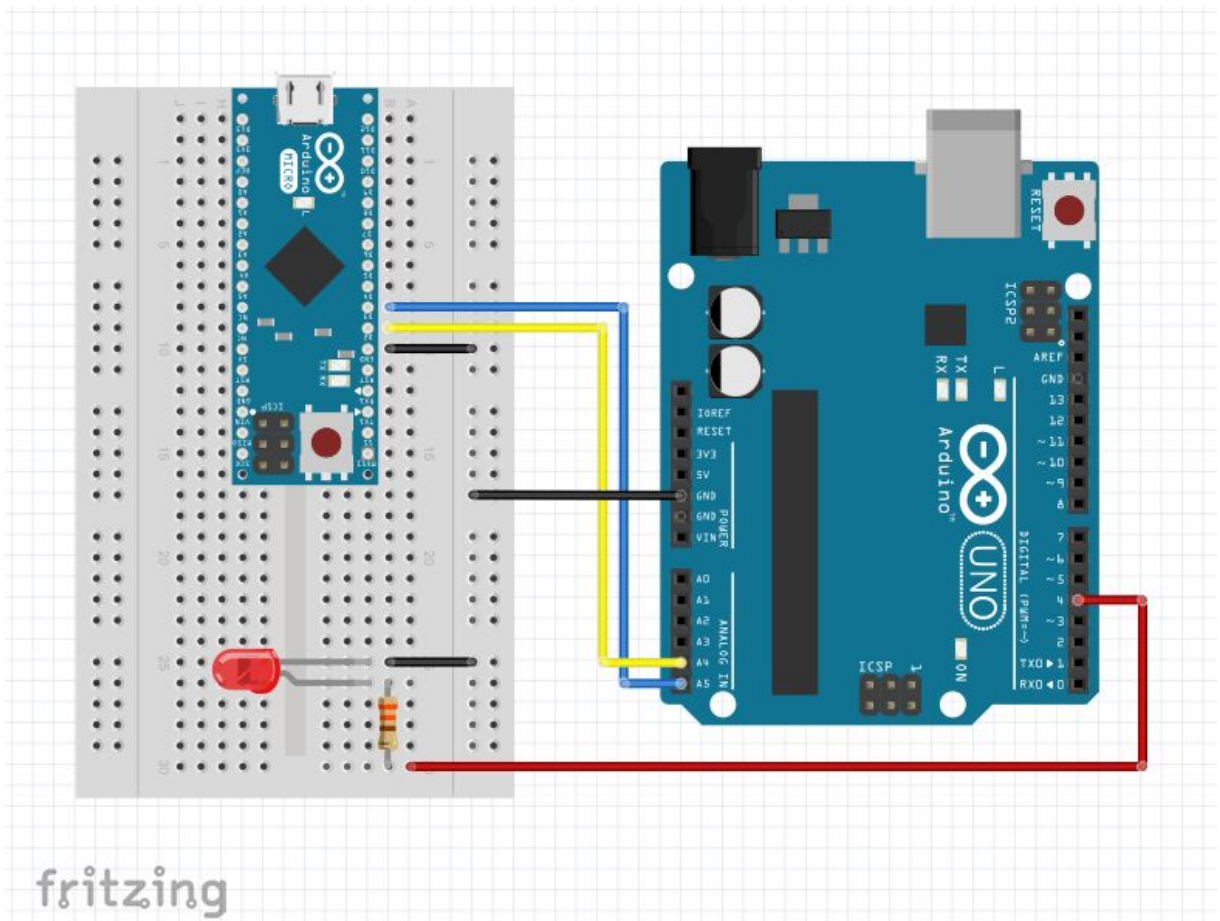
```
{  
  lcd.init();           // Inicializando o LCD  
  lcd.backlight();      // Ligando o BackLight do LCD  
  lcd.print("Hello, world!"); // Exibindo no LED Hello, world!  
}
```

```
void loop()
```

```
{  
}
```

<https://www.robocore.net/tutoriais/primeiros-passos-com-modulo-i2c.html>

2) Nesse exemplo iremos utilizar o Arduino Micro I2C e o Arduino Uno, portanto será utilizado dois códigos, que um será o mestre do barramento e outro que será o escravo.



- Neste código utilizando o Arduino Micro (I2C), será responsável por ser o mestre do barramento:

```
#include <Wire.h>

bool estado_LED;

void setup() {
  Wire.begin();
}

void loop() {
  Wire.beginTransmission(0x08);
  wire.write(estado_LED);
  Wire.endTransmission();

  estado_LED = !estadoLED;

  delay(1000);
}
```

- Já nesse utilizando o Arduino Uno, será o escravo:

```
#include <Wire.h>

void setup() {
  Wire.begin(0x08);
  Wire.onReceive(receiveEvent);
  pinMode(4,OUTPUT);
}

void loop() {
  delay(100);
}

void receiveEvent(int leitura) {

  bool estado = Wire.read();  // receive byte as an integer

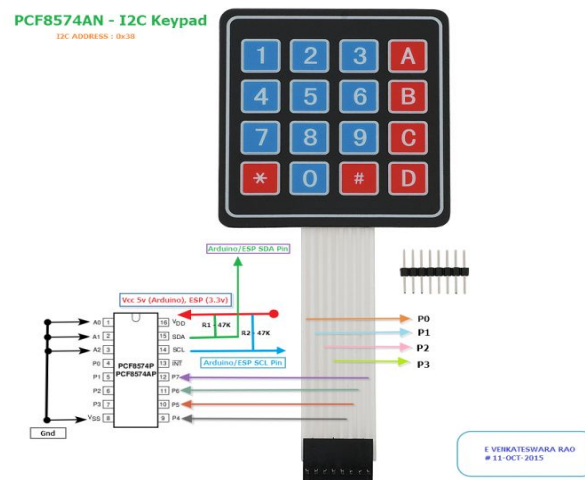
  if (estado == 1){
    digitalWrite(4,HIGH);
  }
  else{
    digitalWrite(4,LOW);
  }
}
```

<https://portal.vidadesilicio.com.br/i2c-comunicacao-entre-arduinos/>

- **Exemplos de aplicação:**

- 1) O método I2C no mundo do trabalho, serve para os desenvolvedores de projetos mais dinâmicos e que necessitam de recursos mais tecnológicos, utilizarem o mesmo conseguir com mais facilidade, mediante o mercado de trabalho utilizar esse método com projetos mais complicados.
- 2) É utilizado por pessoas que estão iniciando o ramo no universo HArdware, buscando uma facilidade, e também com mais conectivos. Por mais que quem esteja iniciando neste universo maker e de projetos pode ficar com algumas dúvidas, porém é a uma das opções mais viáveis. Tendo como base de uso, ainda mais que o código utilizado para o funcionamento de um Display com I2C é diferente do código convencional utilizado com outros Displays.
- 3) Esse exemplo é uma ilustração de uso onde professores desta área de hardware ensinam virtualmente o como utilizar, para as pessoas que buscam aprender essa área: <https://www.youtube.com/watch?v=fEUBQCUe1z8>

- 4) Este exemplo é um teclado numérico feita a partir do método I2C, assim para testar seus conhecimentos em seu método de aprendizagem. E podemos utilizar no dia a dia:



- Abaixo o código deste projeto:

```
// I2C Keypad for Arduino  
// Venkateswara Rao.E  
// 19-oct-2015  
// Credits to @author Alexander Brevig
```

```
#include  
#include  
#include  
#define I2CADDR 0x38  
  
const byte ROWS = 4; //four rows  
const byte COLS = 3; //three columns  
char keys[ROWS][COLS] = {  
  {"1", "2", "3"},  
  {"4", "5", "6"},  
  {"7", "8", "9"},  
  {"*", "0", "#"}  
};
```

```
// Digitran keypad, bit numbers of PCF8574 i/o port  
byte rowPins[ROWS] = {0, 1, 2, 3}; //connect to the row pinouts of the keypad  
byte colPins[COLS] = {4, 5, 6}; //connect to the column pinouts of the keypad
```

```
Keypad_I2C kpd( makeKeymap(keys), rowPins, colPins, ROWS, COLS, I2CADDR, PCF8574 );
```

```
void setup(){  
  Wire.begin( );  
  kpd.begin( makeKeymap(keys) );  
  Serial.begin(9600);  
  Serial.println( "start" );
```

```

}

void loop(){
    char key = kpd.getKey();

    if (key){
        Serial.println(key);
    }
}

```

<http://suadica.com/dica.php?d=412&t=teclado-com-protocolo-i2c-no-esp8266>

● Prós e Contras:

● Prós:

- 1) O método tem um baixo consumo, assim facilitando aos usuários a utilizarem e a gostarem mais do método;
- 2) Diversos sistemas tecnológicos podem ser conectados num único sistema

● Contras:

- 1) O sistema pode travar e assim não sair como combinado o seu projeto;
- 2) Sua velocidade de conexão vai de 100 bytes á 400 bytes.

Pesquisar também: (Watch Dogs) - Função; Utilização c/arduino
 Pesquisar o marco legal de internet das coisas.

● Referências Bibliográficas (junto com os links dos itens acima):

<https://www.robocore.net/tutoriais/primeiros-passos-com-modulo-i2c.html>
<https://portal.vidadesilicio.com.br/i2c-comunicacao-entre-arduinos/>
<https://www.arduinoecia.com.br/modulo-i2c-display-16x2-arduino/>
<https://portal.vidadesilicio.com.br/descobrimdo-o-endereco-i2c/>
<http://suadica.com/dica.php?d=412&t=teclado-com-protocolo-i2c-no-esp8266>
<https://www.youtube.com/watch?v=fEUBQCUE1z8>