



**Campus Quissamã**

**Curso Integrado Informática**

**Professor: Renato**

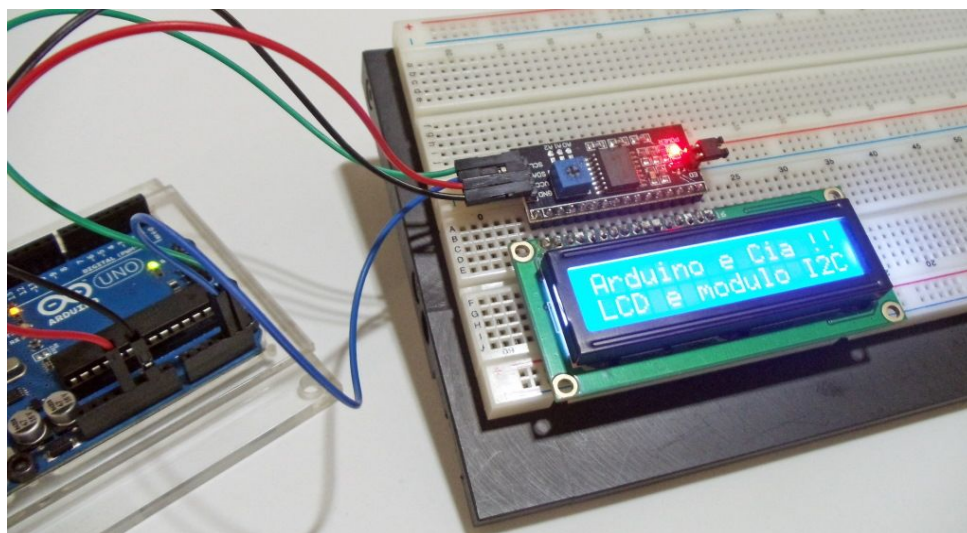
**Turma: 2º ano informática**

**Aluno: Arthur França Freitas e Vitória Silva Nascimento Cabral**

**Assunto: Relatório I2C**

- **O que é I2C?**

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

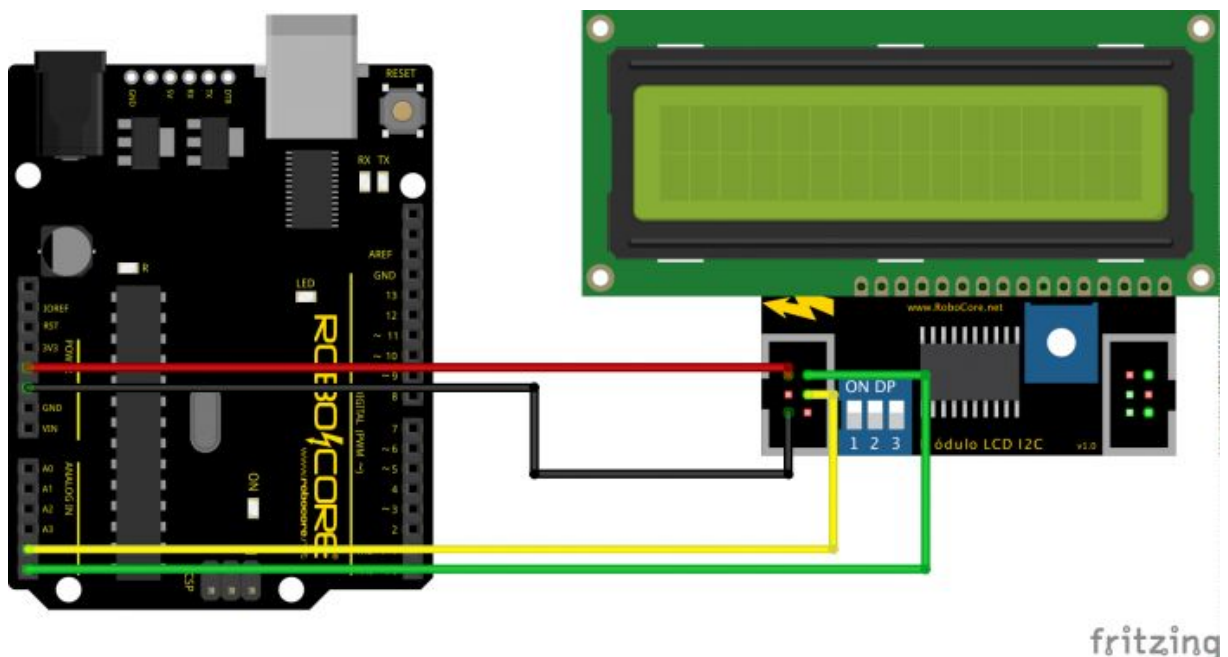


É notável nesta imagem dois barramentos, a qual fazem a comunicação, tendo em vista que, um deles é o serial data (SDA) e o outro é o Serial Clock (SCL).

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.

- **Exemplos:**

- 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!
```

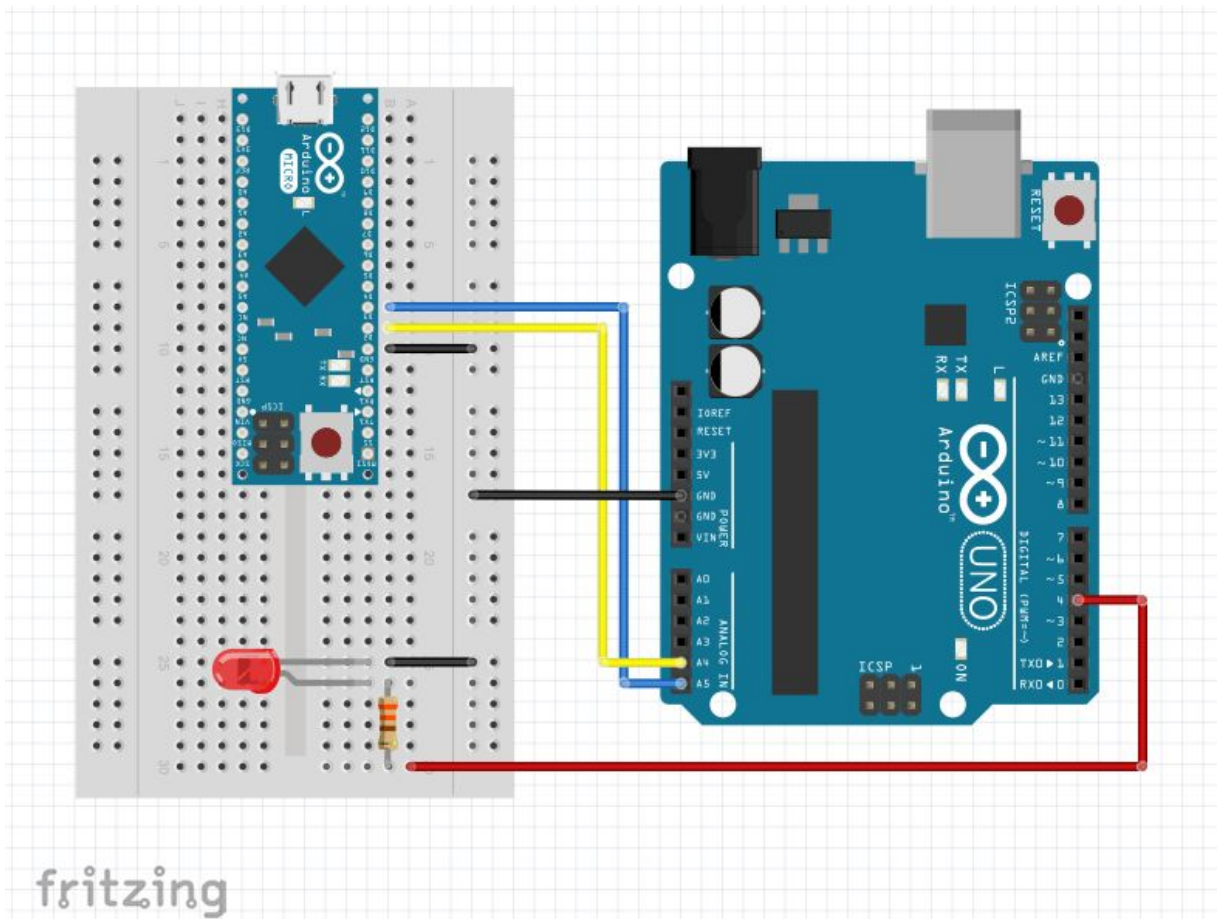
```
}
```

```
void loop()
```

```
{
```

```
}
```

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/>

<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/>