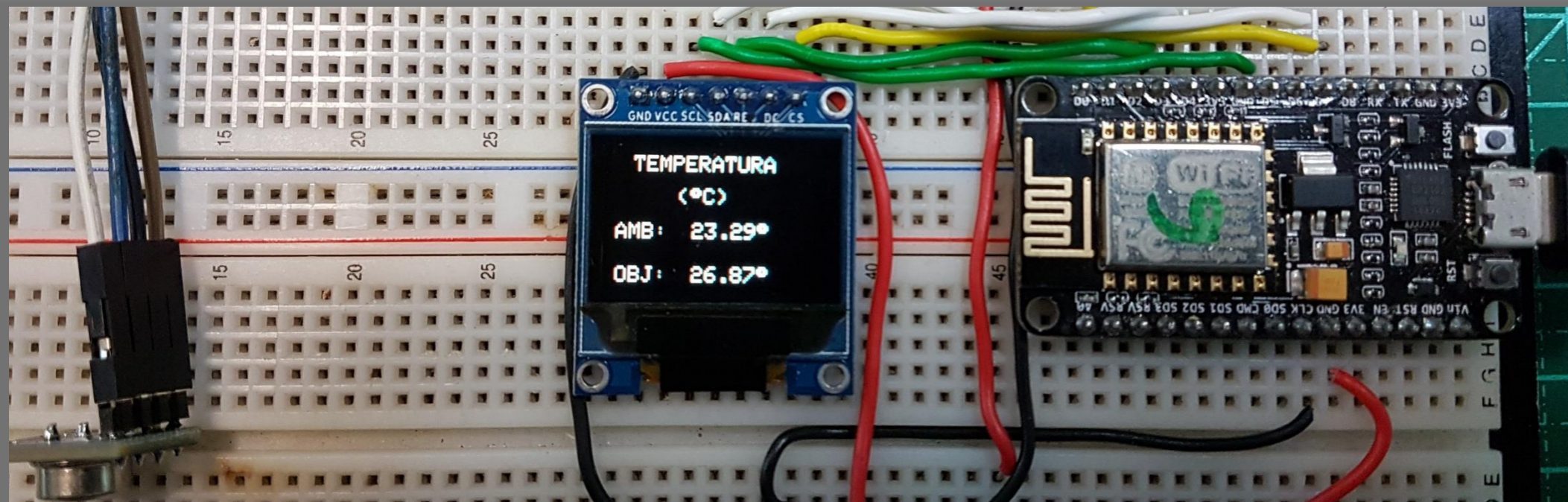


Sensor Infravermelho com ESP8266



Por Fernando Koyanagi



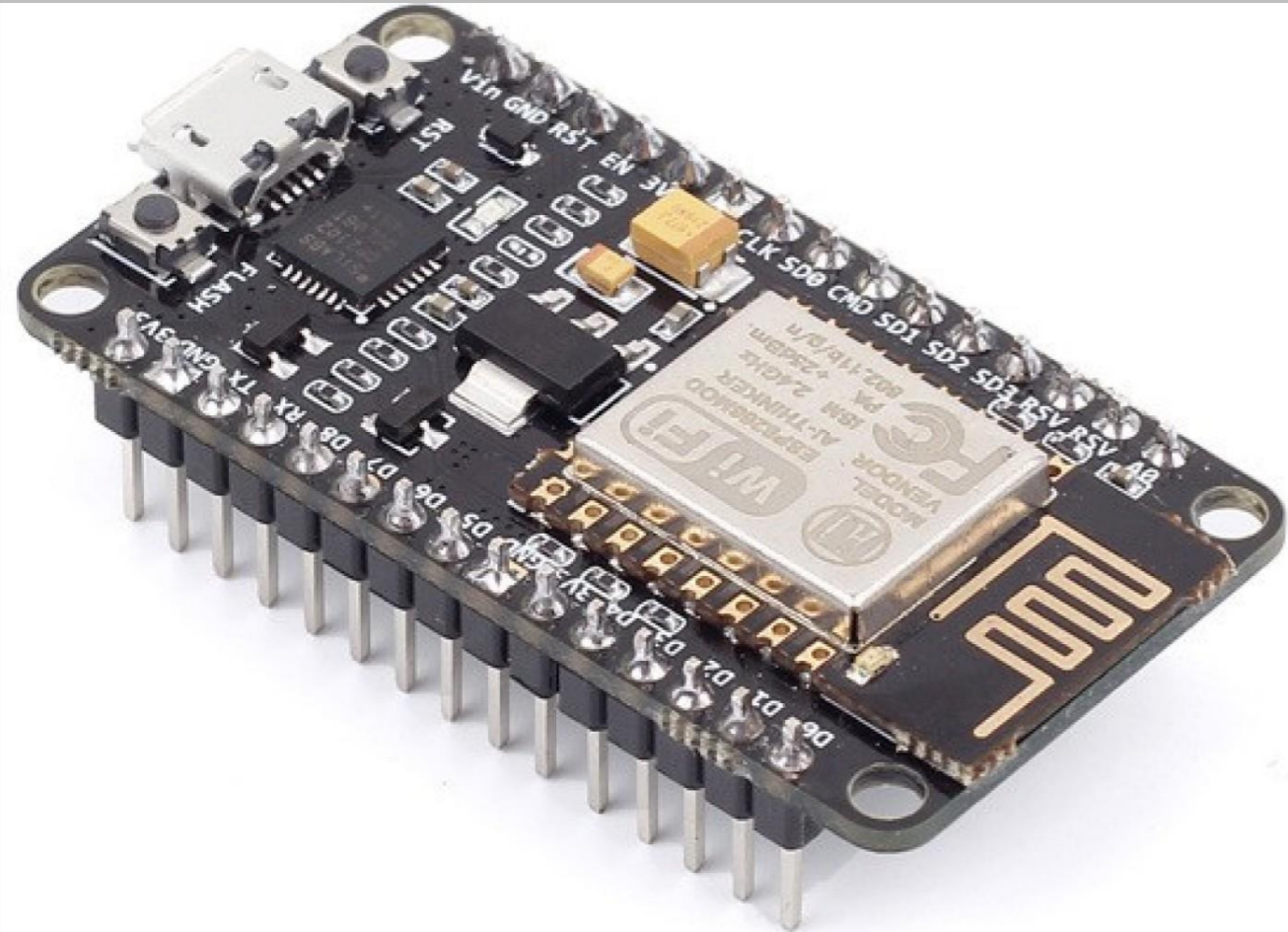
Objetivo

Nosso objetivo será criar um programa que fará a leitura da temperatura ambiente e de qualquer objeto que apontarmos nosso sensor. Os dados da temperatura serão mostrados em um display OLED.

Utilizaremos nesse projeto um nodeMCU ESP8266, um sensor infravermelho MLX90614 e um display OLED 96”.

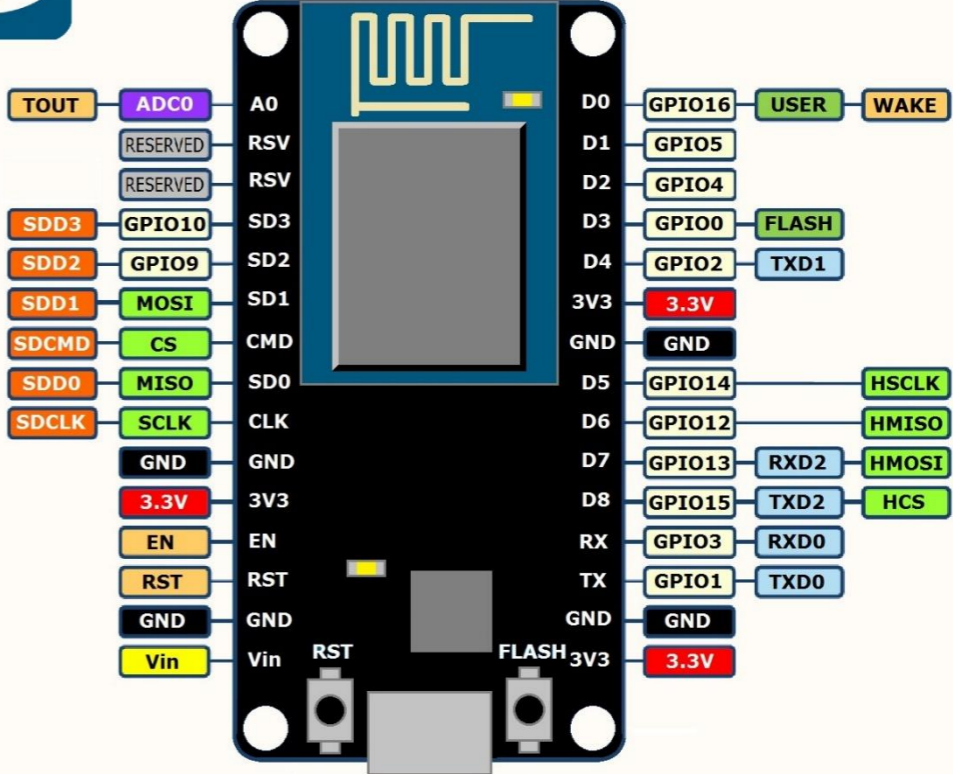


WiFi ESP8266 NodeMcu ESP-12E



NodeMCU ESP-12 development kit V1.0

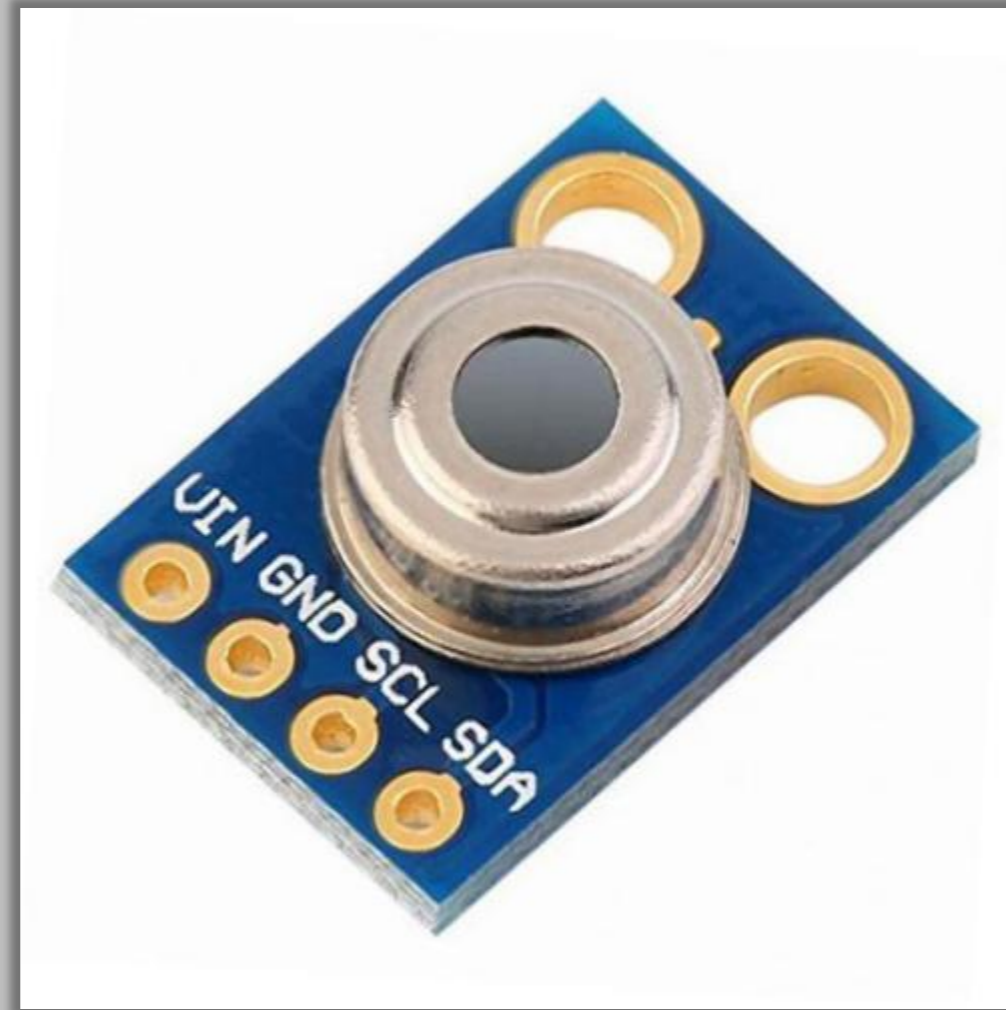
PIN DEFINITION



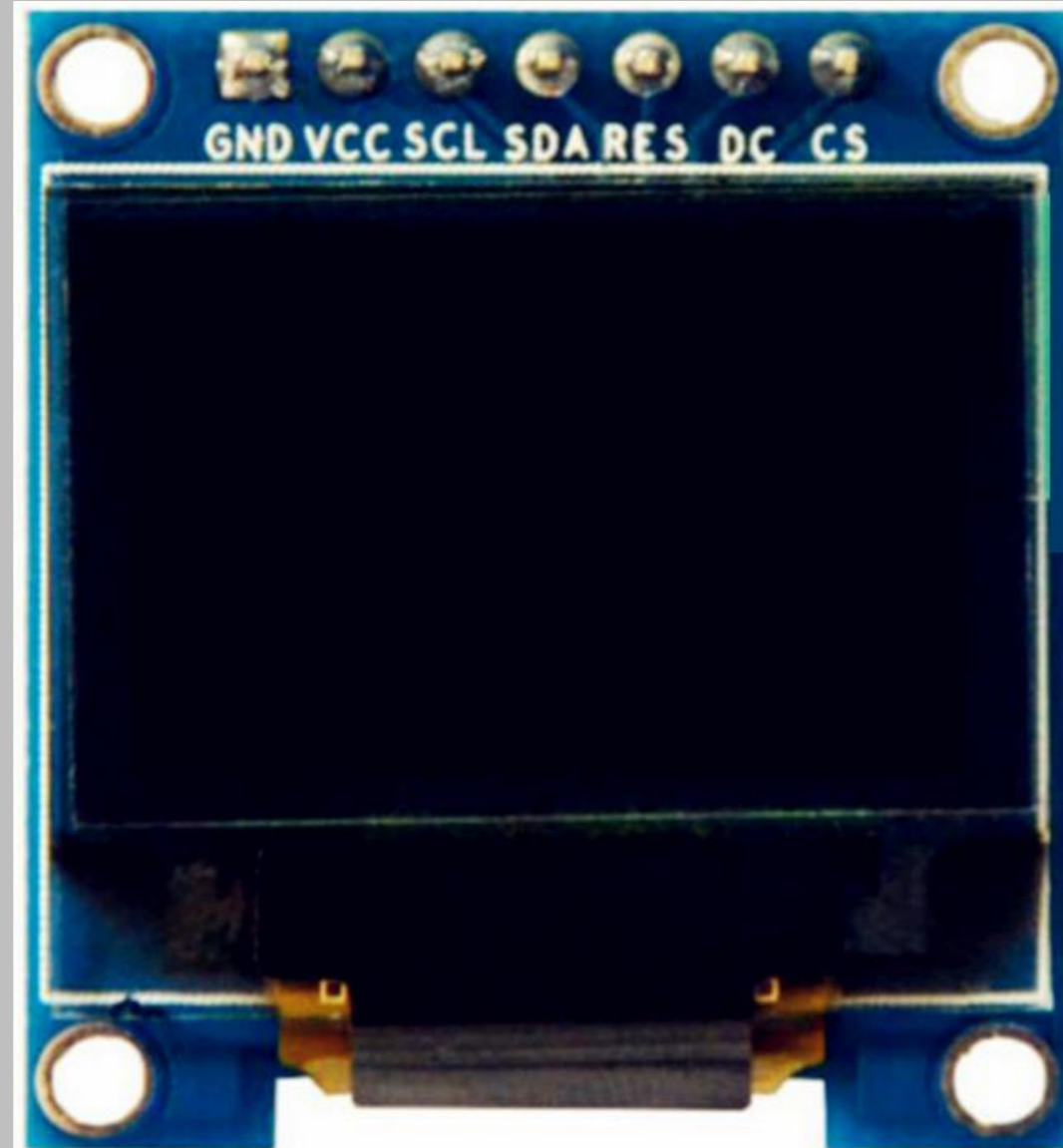
Arduining.com



Sensor Infravermelho MLX90614



Display OLED 96"

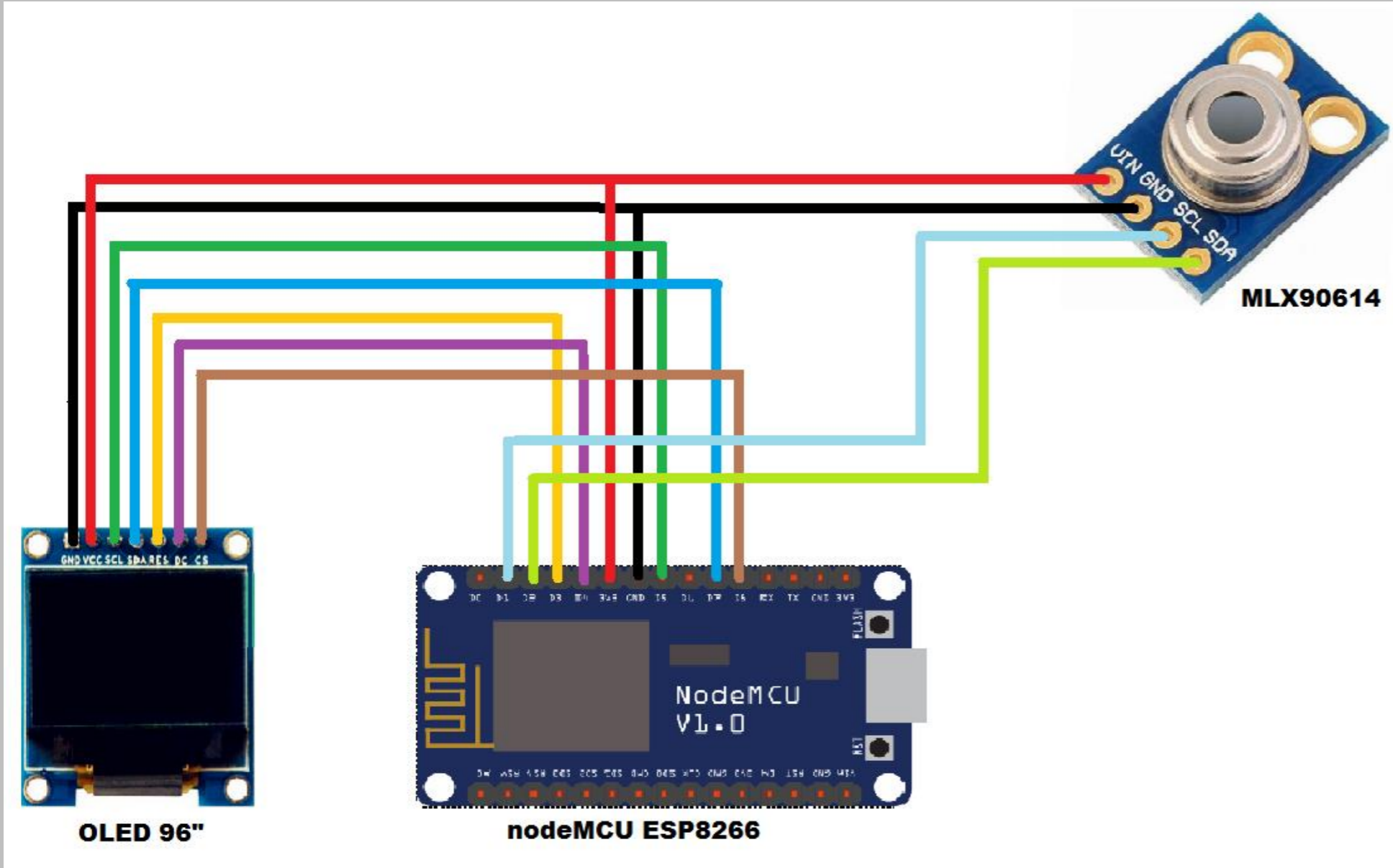


Em www.fernandok.com

Download arquivo **PDF** dos diagramas
Download arquivo **INO** do código fonte



Montagem



ESP8266

D5
D7
D3
D4
D8
3,3v
GND

OLED

SCL
SDA
RES
DC
CS
VCC
GND

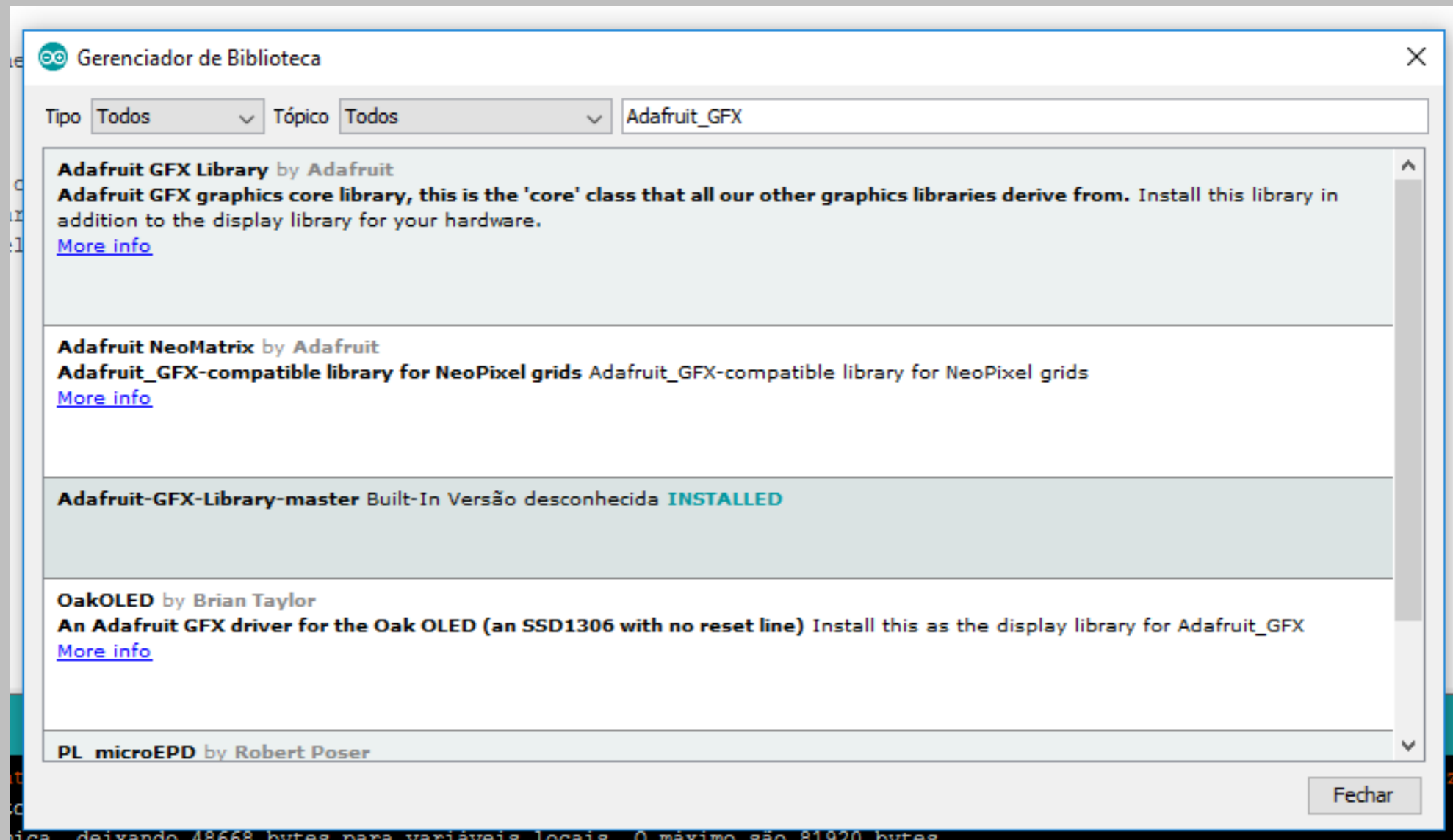
MLX90614

D1
D2
3,3v
GND

SCL
SDA
VCC
GND

Bibliotecas

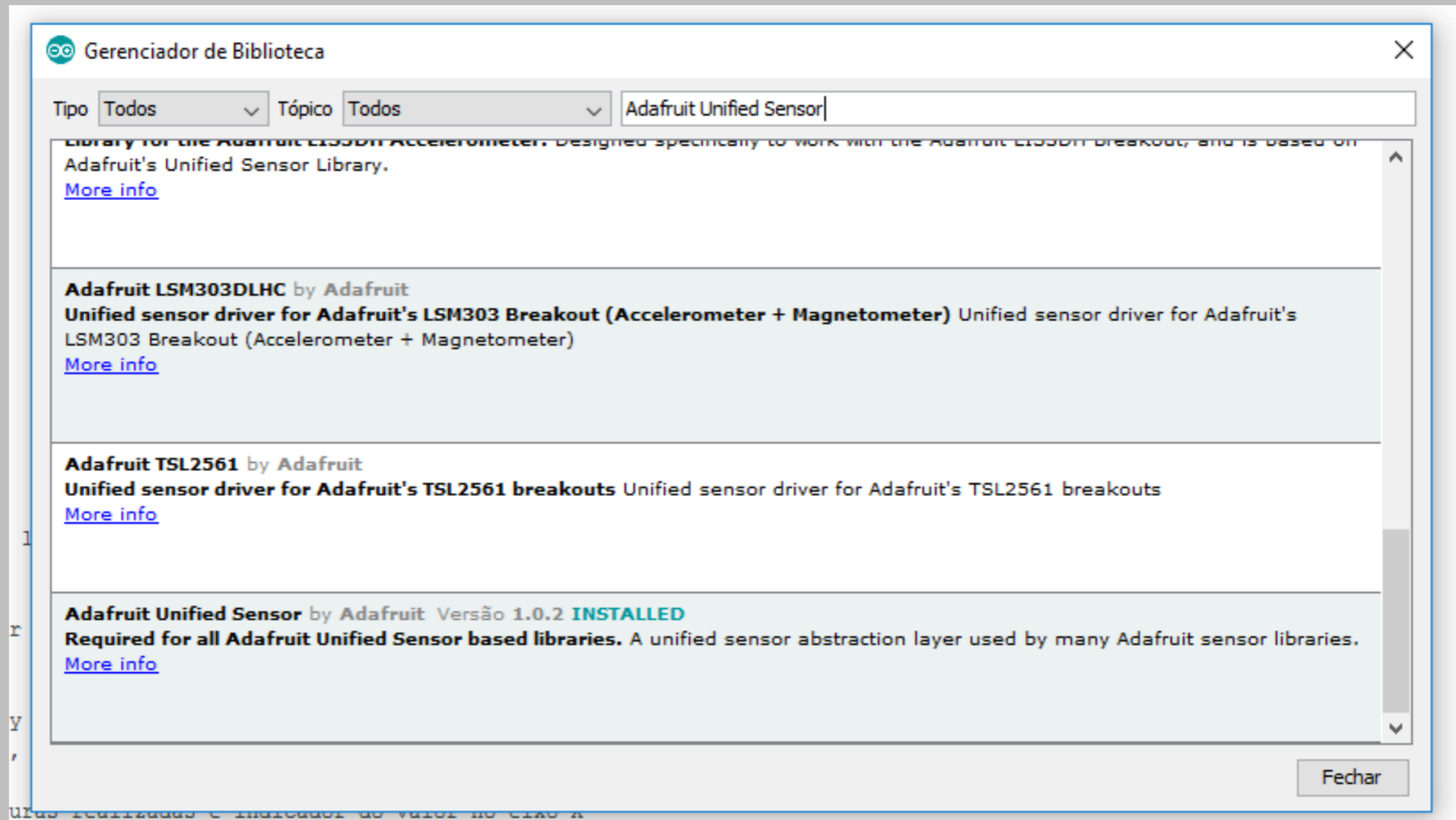
Adicione a seguinte biblioteca “Adafruit-GFX-Library-master”.
Basta acessar “Sketch >> Incluir Bibliotecas >> Gerenciar Bibliotecas...”



Bibliotecas

Adicione a seguinte biblioteca “Adafruit Unified Sensor”.

Basta acessar “Sketch >> Incluir Bibliotecas >> Gerenciar Bibliotecas...”



Bibliotecas

Adicione também a biblioteca “Adafruit-SSD1331-OLED” para comunicação com o display oled.

Acesse o [link](#) e faça download da biblioteca.

Descompacte o arquivo e cole na pasta de bibliotecas da IDE do arduino.

C:/Program Files (x86)/Arduino/libraries



Bibliotecas

Adicione também a biblioteca “SparkFunMLX90614” para comunicação com o sensor infravermelho.

Acesse o [link](#) e faça download da biblioteca.

Descompacte o arquivo e cole na pasta de bibliotecas da IDE do arduino.

C:/Program Files (x86)/Arduino/libraries



Vamos ao código

Começaremos definindo as bibliotecas e constantes que utilizaremos durante nosso código.

```
#include <Wire.h> //Biblioteca para I2C
#include <SparkFunMLX90614.h> //Biblioteca para comunicação com o sensor

#include <Adafruit_GFX.h> //Biblioteca para propriedades gráficas
#include <Adafruit_SSD1331.h> //Biblioteca para comunicação com display OLED

// pinagem para o NodeMCU ESP8266
#define sclk D5
#define mosi D7
#define cs D8
#define rst D3
#define dc D4

// definição das cores que serão utilizadas
#define BLACK 0x0000
#define WHITE 0xFFFF
```



Continuando nossas definições

```
//definição da coordenada onde escreveremos cada um dos dados
#define POS_X_AMBIENTE      2
#define POS_Y_AMBIENTE     35

#define POS_X_OBJETO        2
#define POS_Y_OBJETO        55

#define POS_X_TITULO        10
#define POS_Y_TITULO        4

// construtor do objeto para comunicar com o display OLED
Adafruit_SSD1331 display = Adafruit_SSD1331(cs, dc, mosi, sclk, rst);

//objeto responsável pela comunicação com o sensor infravermelho
IRTherm sensor;

//variáveis que armazenarão o valor das temperaturas lidas
float tempAmbiente;
float tempObjeto;
```



Setup

Na função *setup()*, vamos inicializar nosso objeto de comunicação com o sensor assim como o objeto de comunicação com o display. A seguir faremos algumas configurações de cada um deles.

```
void setup()
{
    //Inicializa sensor de temperatura infravermelho
    sensor.begin();
    //Seleciona temperatura em Celsius
    sensor.setUnit(TEMP_C); //podemos ainda utilizar TEMP_F para Fahrenheit
                           //ou TEMP_K para Kelvin

    //inicializa o objeto para comunicarmos com o display OLED
    display.begin();
    //pinta a tela toda de preto
    display.fillScreen(BLACK);

    //configura o tamanho do texto que escreveremos em tela
    display.setTextSize(0);
    //configura a cor branca para o texto
    display.setTextColor(WHITE);
    //...continua
}
```



Continuando Setup

```
void setup()
{
  //..continuando
  //os comandos abaixo posicionam o cursor no (x,y) desejado para a seguir
  escrevermos em tela
  display.setCursor(POS_X_TITULO, POS_Y_TITULO);
  display.print("TEMPERATURA");
  display.setCursor(POS_X_TITULO+20, POS_Y_TITULO+15);
  display.print("(");
  display.print((char)247); //símbolo de graus
  display.print("C");

  display.setCursor(POS_X_AMBIENTE, POS_Y_AMBIENTE);
  display.print("AMB:"); //AMBIENTE

  display.setCursor(POS_X_OBJETO, POS_Y_OBJETO);
  display.print("OBJ:"); //OBJETO
}
```



Loop

Na função *loop()*, vamos fazer a leitura dos dados do sensor e em seguida mostrar eles no display OLED.

```
//chamamos o método "read" do sensor para realizar a leitura da temperatura
//read retornará 1 caso consiga realizar a leitura, ou 0 caso contrário
if (sensor.read())
{
    //recupera a leitura da temperatura do ambiente
    tempAmbiente = sensor.ambient();
    //recupera a leitura da temperatura do objeto apontado pelo sensor
    tempObjeto = sensor.object();

    //limpa a área onde colocamos o valor da temperatura do ambiente e do objeto
    display.fillRect(POS_X_AMBIENTE+35, POS_Y_AMBIENTE, 35, 10, BLACK);
    display.fillRect(POS_X_OBJETO+35, POS_Y_OBJETO, 35, 10, BLACK);

    //posiciona o cursor e escreve a temperatura ambiente
    display.setCursor(POS_X_AMBIENTE+35, POS_Y_AMBIENTE);
    display.print(tempAmbiente);
    display.print((char)247); //símbolo de graus

    //posiciona o cursor e escreve a temperatura do objeto que o sensor está apontando
    display.setCursor(POS_X_OBJETO+35, POS_Y_OBJETO);
    display.print(tempObjeto);
    display.print((char)247); //símbolo de graus

}
delay(1000); //intervalo de 1 segundo para a próxima leitura
}
```



Em www.fernandok.com

Download arquivo **PDF** dos diagramas
Download arquivo **INO** do código fonte

