**Vitor Amadeu Souza**

Monitorando através do Visual C# o

# Higrômetro

com o STM32F103C8 programado no Arduino

**© 2020 by Cerne Tecnologia e Treinamento Ltda.**

**© 2020 by Vitor Amadeu Souza**

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

**Outubro de 2020**

Direitos reservados por:
Cerne Tecnologia e Treinamento Ltda
*Produção: Cerne Tecnologia e Treinamento*
*E-mail da Empresa: cerne@cerne-tec.com.br*
*Home Page: www.cerne-tec.com.br.com.br*
*Atendimento ao Consumidor: sac@cerne-tec.com.br*
*Contato com o Autor: vitor@cerne-tec.com.br*



**FEITO NO BRASIL**

*“Se não se espera, não se encontra o inesperado.”*

**Heráclito**

**Kits Didáticos e Gravadores da Cerne Tecnologia**

A Cerne tecnologia têm uma linha completa de aprendizado para os microcontroladores da família PIC, 8051, Holtek, dsPIC, ARM e etc. Veja os detalhes de cada um nas figuras abaixo:




**Kit Arduino**

- Gravação On-Board
- Comunicação Serial RS232
- Entrada de 12 V

Uma linha completa de componentes para o desenvolvimento de seus projetos eletrônicos como displays, PICs, botões, leds, cristais e etc. Visite a nossa página na Internet, no endereço **www.cerne-tec.com.br** e conheça melhor nossos serviços e produtos.

Cerne
Conhecimento para o Desenvolvimento

# Sumário

**Capítulo I – Metodologia de desenvolvimento ..... 6**

1. Introdução ..... 6

**Capítulo II – Controle do display LCD ..... 7**

1. Introdução ..... 7
2. Montando o Hardware ..... 8
3. Programando o Arduino ..... 9

**Capítulo III – Higrômetro ..... 11**

1. Sensor de pressão HIH-4000 ..... 11
2. Esquema elétrico ..... 16
3. Código fonte ..... 17

**Capítulo IV – Transmissão serial ..... 19**

1. Introdução ..... 19
2. Montando o hardware ..... 19
3. Transmitindo dados pela serial ..... 20
4. Medindo a variável e enviando pela serial ..... 23

**Capítulo V – Monitoramento ..... 25**

1. Tela gráfica ..... 25
2. Programação ..... 26

# Capítulo I
# Metodologia de desenvolvimento

## 1. Introdução

Esta literatura é uma continuação da obra *Programando o ARM no Arduino – Com base no modelo STM32F103C8* (2019) do mesmo autor e editora, onde o objetivo é monitorar o sensor título da obra programado no Arduino e uma interface gráfica desenvolvida no Visual C# para monitorá-la.

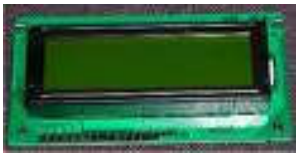
A placa didática utilizada foi a *Blue Pill* STM32F103C8, onde tal kit está à venda no site www.cerne-tec.com.br.

# Capítulo II

# Display LCD

## 1. Introdução

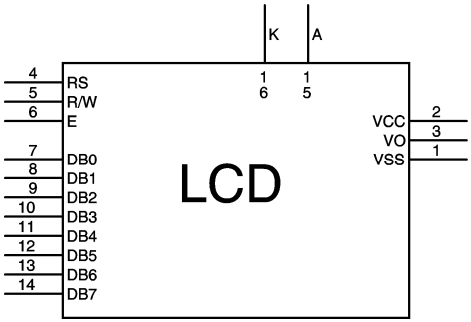
Os displays do tipo LCD são muito usados para mostrar informações do estado de uma máquina. Na figura a seguir, é apresentado aspecto típico desta interface.



Existem vários tipos de display, como os de 2x20, que significa 2 linhas por 20 colunas, 4x20, 4x16 e etc. Neste exemplo, será usado um display do tipo 2x16, onde haverá duas linhas e 16 colunas. Será apresentado um texto no LCD, de forma a entender como montar um programa para comandá-lo.

## 2. Montando o Hardware

Na figura a seguir, é apresentado a pinagem típica de um display LCD.




Siga a tabela apresentada a seguir para ligar o display LCD à placa STM32F103C8 Breakout Board.

| Display LCD | Placa STM8S103F3 Breakout Board |
|---|---|
| VCC | 5V |
| VSS | GND |
| VO | Em série com resistor de 1kΩ ao GND |
| RS | PA8 |
| EN | PA9 |
| D4 | PB5 |
| D5 | PB6 |
| D6 | PB7 |
| D7 | PB8 |

Os pinos não citados não são usados.

## 3. Programando o Arduino

Foi utilizada as funções disponíveis em uma biblioteca do Arduino para que o exemplo fique funcional. A biblioteca é a LiquidCrystal.h que contém as funções de inicialização e escrita no LCD. Além disso, será necessário utilizar a função lcd.begin para configurar o tipo de display conectado ao processador. A seguir é apresentada tal função.

```
lcd.begin(colunas,linhas);
```

Em colunas, define-se o número de colunas no qual o display possui e em linhas o número de linhas. Após este passo, utiliza-se a função lcd.setCursor para posicionar a impressão de determinada informação. A seguir, tal função é apresentada.

```
lcd.setCursor(coluna,linha);
```

Onde caso coluna esteja em 0, a coluna 1 estará selecionada e caso esteja em 1, a coluna 2 e assim sucessivamente. Mesma ideia é válida para a opção linha. Finalmente, para imprimir um valor no LCD, utiliza-se a função lcd.print como apresentado a seguir.

```
lcd.print(texto);
```

A seguir os passos para mostrar uma mensagem na 1º e 2º linha do display lcd.

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(PA8, PA9, PB5, PB6, PB7,
PB8);

void setup()
{
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print("Placa Arduino");
  lcd.setCursor(0,1);
  lcd.print("Exemplo LCD");
}


void loop()
{

}
```