Vitor Amadeu Souza

Lendo umidade e

Temperatura

com o sensor

AM2320

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.

Maio 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

Dedicatória

A minha querida esposa Renata Leal Souza.

"Companheiro sou de todos os que te temem, e dos que guardam os teus preceitos."

Sl 119:63

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 – Display LCD em modo paralelo 7

1. Introdução 7

2. Código fonte 8

Capítulo III – Comunicação I^2C 9

1. Introdução 9

2. Pinagem da memória 10

3. Acesso a memória 12

4. Esquema elétrico 23

5. Código fonte 25

Capítulo IV –AM2320 27

1. Introdução 27

2. Esquema elétrico 27

3. Fluxograma 29

4. Biblioteca 29

5. Código fonte 30

Capítulo I
Metodologia de desenvolvimento

1. Introdução

Esta literatura é uma continuação da obra *Arduino – Prático e Objetivo* (2011) e *Programação para Arduino – Avançado* (2014) do mesmo autor e editora, onde o objetivo é conectar o dispositivo título desta obra ao Arduino.

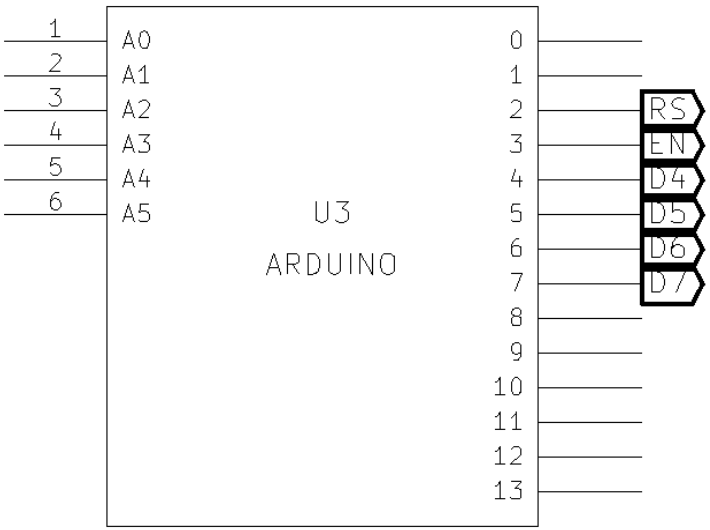
A placa didática utilizada foi a Arduino UNO, onde tal kit está à venda no site www.cerne-tec.com.br.

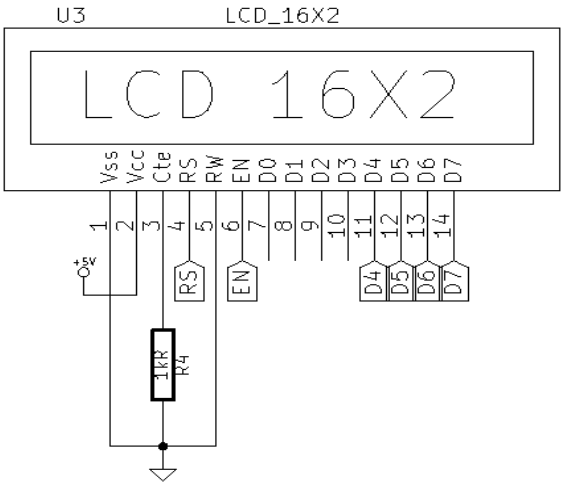
Capítulo II

Display LCD em modo paralelo

1. Introdução

Este exemplo demonstra o controle de um display LCD em modo paralelo. Observe que para isso, mais pinos do Arduino são utilizados para este fim. A IHM (Interface Homem-Máquina) utilizada no projeto foi um display do tipo LCD, no qual o controlador utilizado é o HT44780. Observe na próxima figura os pinos digitais utilizados para comunicação com o display LCD:





2. Código fonte

A seguir o código que configura os pinos digitais citados para conectar o LCD. Após a inclusão e definição do objeto lcd, dentro do bloco setup o mesmo é inicializado e uma mensagem impressa no LCD.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(2,3,4,5,6,7);
void setup()
{
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print("Placa Arduino");
  lcd.setCursor(0,1);
  lcd.print("Exemplo LCD");
}
void loop()
{}
```

Capítulo III
Comunicação I^2C

1. Introdução

Este experimento demonstra a utilização do protocolo I^2C conectando o Arduino a uma memória 24C04. Para isso, está implementado um experimento que escreve um valor na memória e logo em seguida faz a sua leitura, apresentando no display lcd o valor lido.

A memória EEPROM utilizada é a 24LC04 que tem capacidade de armazenar até 512 bytes de dados. Existem diversas memórias da família 24. Para saber a capacidade de armazenamento destas memórias, basta dividir os últimos números de identificação por oito. Por exemplo, a memória utilizada tem 4 kbits e 4kbits dividido por 8 é igual a 512 bytes. Este mesmo conceito vale, por exemplo, para uma memória do tipo 24LC256 que suporta até 32kB de informação.

O protocolo I^2C foi desenvolvido pela Philips e utiliza dois fios para fazer a comunicação com outros periféricos. É um protocolo síncrono composto de uma linha de clock (SCL) e outra de dados

(SDA). Através destas duas linhas, conseguiremos escrever e ler os dados internos nesta memória.

As linhas de clock (SCL) e dados (SDA) não conseguem impor o nível lógico 1, apenas o nível baixo. Para garantir este nível, é necessário a utilização de dois resistores de pull-up conectados nestas linhas, de modo a garantir o nível lógico alto. Estes resistores variam de 1kΩ até 10 kΩ e quanto menor o valor, maior pode ser a taxa de comunicação.

2. Pinagem da memória

Abaixo a figura da memória utilizada:

