

Vitor Amadeu Souza

Desenvolvendo uma aplicação PoE

Cliente

Na rede Ethernet com W5100 programado no Arduino

Servidor

Programado no Visual C#

© 2019 by Cerne Tecnologia e Treinamento Ltda.

© 2019 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.

Agosto de 2019

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

Como nos meus outros livros, dedico este livro a minha querida esposa Renata Leal.

“Até que o Sol não brilhe, acendamos uma vela na escuridão.”

Confúcio

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço cerne-tec.com.br para obter mais informações.



Sumário

Capítulo I – Web Server	6
1. Web server	6
2. Conector RJ45	7
3. Bibliotecas e funções utilizadas.....	8
4. Código fonte.....	9
Capítulo II – Conexão Cliente - Servidor	12
1. Esquema elétrico do cliente.....	12
2. Fluxograma do cliente.....	13
3. Código fonte do cliente.....	13
4. Esquema elétrico do servidor.....	16
5. Fluxograma do servidor.....	17
6. Código fonte do servidor.....	18
Capítulo III – Programação do servidor	18
1. Programação	18
Capítulo IV – PoE.....	23
1. Introdução.....	23
2. Pinagem.....	23
3. Regulador	24

Introdução

A proposta desta literatura é apresentar a conexão cliente-servidor via rede Ethernet sendo o cliente implementado no módulo W5100 programado no Arduino e o servidor em Visual C#.

O objetivo será ler o estado de um botão no cliente e enviar para o servidor o seu estado assim que pressionado, de modo que este indique a ocorrência de tal evento na tela da aplicação.

Finalmente é apresentado o hardware *PoE (Power Over Ethernet)* de modo a alimentar o projeto pelo próprio cabo de rede.

É fundamental que o leitor utilize como referência ou possua experiência dos assuntos abordados na literatura *Desenvolvendo um Web Server na rede Ethernet com W5100 para monitorar sensores e Comunicação TCP/IP com VB (2016)* do mesmo autor e editora.

Capítulo I

Web Server

1. Web server

A proposta deste experimento é inicialmente desenvolver um servidor Web (Webserver) no Arduino usando o módulo W5100. Desta forma, ao digitar o IP configurado no kit o mesmo irá apresentar a página HTML guardada em sua memória FLASH. Acompanhe a seguir a página proposta para este exemplo.



2. Conector RJ45

Apesar de disponibilizar 8 pinos no conector RJ45, apenas 4 são utilizados para realizar a comunicação com a rede Ethernet, conforme mostra a próxima tabela.

Número do pino	Função
1	TD+ Transmite dados
2	TD- Transmite dados
3	RD+ Recebe dados
6	RD- Recebe dados
4, 5, 7, 8	Reservados (não utilizados)

Obs: *Os pinos não utilizados podem ser utilizados para PoE (Power over Ethernet).*

De toda forma, o shield ethernet já disponibiliza este conector montado sobre o mesmo, como pode ser visualizado na próxima figura.



Fonte: banggood.com

3. Bibliotecas e funções utilizadas

Neste experimento foram utilizadas as bibliotecas Ethernet e SPI disponíveis no Arduino. A grande facilidade em usar este compilador é que ele fornece as rotinas de comunicação com o PHY W5100, facilitando a implementação deste recurso no Arduino. Para inseri-las ao programa, vá ao menu Sketch-> Include Library -> SPI e Ethernet.

Acompanhe a seguir os métodos utilizados ao longo deste capítulo.

Função	Descrição
Ethernet.begin	Inicializa a máquina de comunicação Ethernet, informando o endereço MAC e IP
client.stop	Encerra a comunicação com o cliente
client.available	Verifica se há caracteres disponíveis no buffer
client.connected	Verifica se há conexão com algum cliente
client.println	Envia caracteres para o cliente
client.read	Recebe caracteres do cliente

Obs: Cliente é o solicitante de serviço do Webserver.

4. Código fonte

A seguir está listado o código completo gravado no Arduino. Após tal apresentação, são feitos os comentários por partes do código que roda no mesmo. Após compilar e gravar o programa faça o teste acessando algum browser e digitando o IP default do programa, cujo endereço é 192.168.10.100, onde espera-se que seja apresentada a página HTML apresentada no início deste capítulo.

```
#include <SPI.h>
#include <Ethernet.h>

byte mac[]      = {0x01,0x02,0x03,0x04,0x05,0x06};
byte ip[]       = {192,168,10,100};
EthernetServer server(80);

void setup()
{
  Ethernet.begin(mac,ip);
  server.begin();
}

void loop()
{
  EthernetClient client = server.available();
  if (client)
  {
    while((client.available()) > 0)
      char c = client.read();

    client.println("HTTP/1.1 200 OK");
    client.println("Content-Type: text/html");
    client.println("<html>");
    client.println("<body>");
    client.println("<h1>Cerne Tecnologia e Treinamento</h1>");
    client.println("<h2>www.cerne-tec.com.br</h2>");
    client.println("</body>");
  }
}
```