

Introdução

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

A plataforma NodeMCU é uma placa open source baseada nas funcionalidades providas pelo chip de baixo custo com suporte a redes sem fio 802.11 ESP8266 que utiliza o microprocessador Xtensa.

Esta plataforma oferece um ambiente adequado para o desenvolvimento de dispositivos que implementam funcionalidades para atuarem em um conceito de IoT de forma simples. Pode ser visto como uma evolução da estratégia de desenvolvimento utilizando Arduino



Introdução

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

A plataforma NodeMCU é um kit de desenvolvimento open source baseada nas funcionalidade providas pelo chip de baixo custo com suporte a redes sem fio 802.11 ESP8266 que utiliza o microprocessador Xtensa.

Esta plataforma oferece um ambiente adequado para o desenvolvimento de dispositivos que implementam funcionalidades para atuarem em um conceito de IoT de forma simples. Pode ser visto como uma evolução da estratégia de desenvolvimento utilizando Arduino



Apresentando a Placa

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem

Monitor Serial

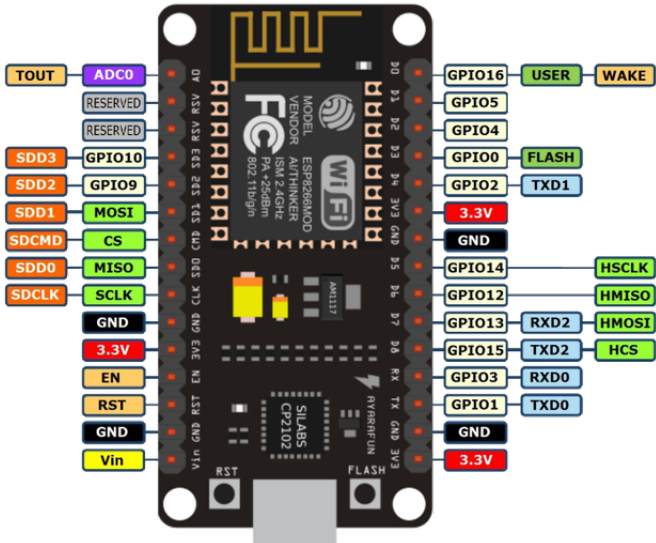
Utilizando
exemplos

Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays



ESP8266

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

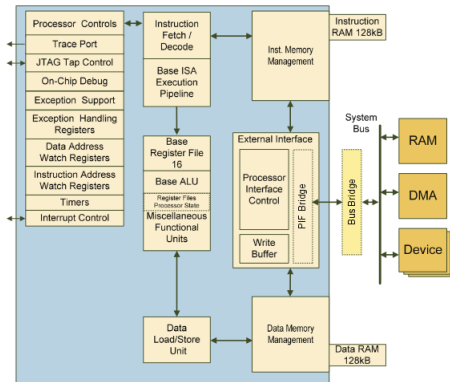
Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

- Microcontrolador: Xtensa L106 (32-bit) 80Mhz
- Memória Interna: 128K para instruções; 128K para dados
- Memória Flash: 4Mb.
- I/O: 16 Pinos GPIO
- Tensão: 3.3 VDC
- Wi-Fi: 802.11 b/g/n



The Diamond Standard 106Micro

Instalando o Arduino IDE

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

Download da IDE em

<https://www.arduino.cc/en/Main/Software>

Extrair o arquivo com

```
tar xvf arduino-1.8.X-linux64.tar.xz
```

Ir para a pasta e iniciar a IDE

```
cd arduino-1.8.X  
./arduino
```

Para adicionar suporte a placa ESP12 e NodeMCU é necessário carregar o compilador e configurações para compilação, geração da flash e copiar o arquivo gerado para o processador.

Instalando o Arduino IDE

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

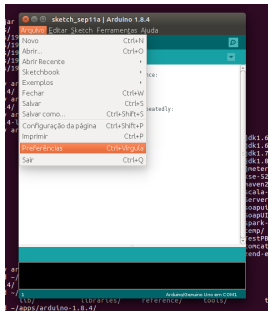
Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

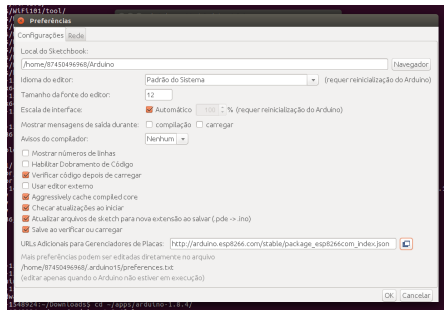
Funcionalidades
Wi-Fi

Motores

Displays



Selecionar o submenu
Preferências pelo menu
Arquivo, será aberta uma
janela com várias opções de
configuração



No campo URLs
Adicionais para
Gerenciador de Placas
deve ser inserido o
seguinte valor

http://arduino.esp8266.com/stable/package_esp8266com_index.json

Configurando suporte para ESP12 e NodeMCU

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

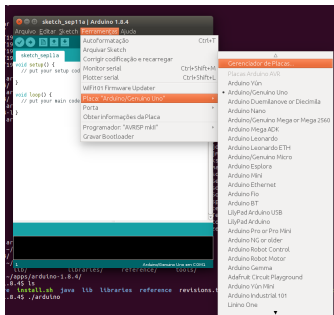
Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

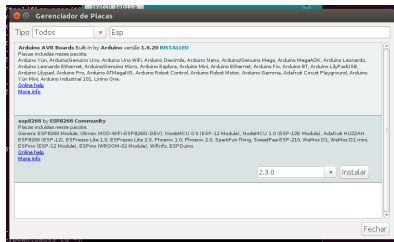
Motores

Displays



Selecionar o submenu
Gerenciador de Placas a partir
pelo menu Ferramentas,
submenu Placa

Para facilitar a busca é possível
aplicar filtros no capo Refine sua
busca. O valor Esp é suficiente
para retornar poucos resultados.
Basta selecionar ESP8266 by
ESP8266 Communit e instalar



Configurações Finais

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o Arduino IDE

Baixando e Instalando o Arduino IDE

Configurando o Arduino IDE

Programando com a IDE Arduino

Estrutura da linguagem

Monitor Serial

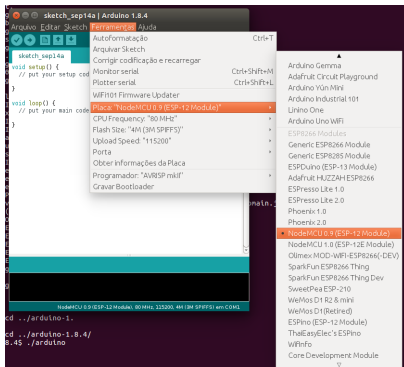
Utilizando exemplos

Incluindo bibliotecas

Funcionalidades Wi-Fi

Motores

Displays



Após a seleção da placa NoceMCU 0.9 retorne ao menu ferramentas e selecione a porta USB correta. No Linux as portas de comunicação serial com o Módulo é mapeada como `/dev/ttyUSBX`, onde X é um número inteiro com valor inicial 0. Você pode verificar as portas ativas utilizando o comando `ls /dev/tty*`

Compilação e cópia

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua

Monitor Serial

Utilizando
exemplos

Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

Para a compilação a IDE utiliza uma coleção de programas para execução do GCC, gerando código para o microprocessador Xensa. Quando é feita a instalação de suporte para NodeMCU este conjunto de programas é baixado localmente.

A cópia do código binário gerado para a placa tem que ser feita passando um conjunto de instruções concebidas especificamente para o processador. No caso da ESP12 o programa é na realidade um firmware.

Para apresentar detalhadamente o processo de compilação e cópia do programa é necessário configurar a IDE habilitando os campos compilação e carregar em Mostrar Mensagens de Saída na tela de preferências (Arquivo, Preferências)

A programação é feita utilizando a sintaxe da linguagem C++, podendo ser utilizadas as funções e tipos primitivos da mesma. No entanto muitas das boas práticas utilizadas para programação em computadores tradicionais devem ser revistas para desenvolvimento de sistemas embarcados.

- Duas funções que tem que existir em qualquer programa;
 - `setup()`
 - `loop()`
- `setup()` é executado na inicialização do programa, é equivalente a `main()`
- `loop()`, é chamado após a conclusão da função `setup()` e é um loop infinito

Algumas funções e constantes

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

Constantes

- OUTPUT, INPUT
- LOW, HIGH

Programação para Portas

- pinMode(PIN, [INPUT—OUTPUT])
- digitalWrite(PIN, [LOW—HIGH])
- digitalRead(PIN)
- delay(TIME_Ms)

Comunicação Serial

- Serial.begin(SPEED)
- Serial.println(DATA)
- Serial.print(DATA)
- Serial.available()
- Serial.read()

Blink um Hello Word em IoT

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando com a IDE Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades Wi-Fi

Motores

Displays

```
/*  
  ESP8266 Blink by Simon Peter  
  Blink the blue LED on the ESP-01 module  
  This example code is in the public domain  
*/  
  
void setup() {  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(LED_BUILTIN, LOW);  
  delay(1000);  
  digitalWrite(LED_BUILTIN, HIGH);  
  delay(2000);  
}
```

Blick com 2 Leds

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
int EXTLED = 5; //Aqui devemos usar o GPIO  
int count = 0;
```

```
void setup() {  
    pinMode(LED_BUILTIN, OUTPUT);  
    pinMode(EXTLED, OUTPUT);  
}
```

```
void loop() {  
  
    if(count %2 == 0){  
        digitalWrite(EXTLED, HIGH);  
        digitalWrite(LED_BUILTIN, LOW);  
    } else {  
        digitalWrite(EXTLED, LOW);  
        digitalWrite(LED_BUILTIN, HIGH);  
    }  
    count++;  
    delay(1000);  
}
```

Monitor Serial

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

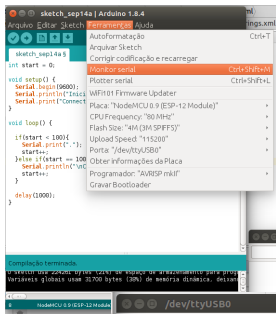
Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

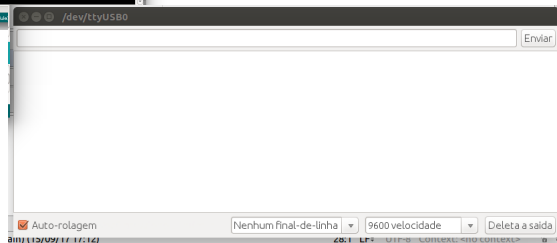
Funcionalidades
Wi-Fi

Motores

Displays



A IDE Arduino oferece uma interface que permite ler e enviar dados para a saída serial. Quando está é habilitada passa a comunicar-se com a interface na porta selecionada.



Comunicação serial

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua

Monitor Serial

Utilizando
exemplos

Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
int start = 0;

void setup() {
  Serial.begin(9600);
  Serial.println("Iniciando");
  Serial.print("Connectando com");
}

void loop() {

  if(start < 100){
    Serial.print(".");
    start++;
  } else if(start == 100){
    Serial.println("\nConnected");
    start++;
  }

  delay(1000);
}
```

Comunicação serial

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
String data;  
int porta = 0;  
  
void setup() {  
  Serial.begin(9600);  
  delay(500);  
  Serial.write(" Digite algo:\n");  
}  
  
void loop() {  
  if (Serial.available() > 0) {  
  
    data = Serial.readString();  
    Serial.print(" Digitado: ");  
    Serial.println(data);  
  
    Serial.println(" Digite um número: ");  
    while(Serial.available() <= 0){}  
    porta = Serial.readString().toInt();  
    Serial.print(porta, HEX);  
    Serial.write("\n Digite algo:\n");  
  }  
  delay(10);  
}
```


NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem

Monitor Serial

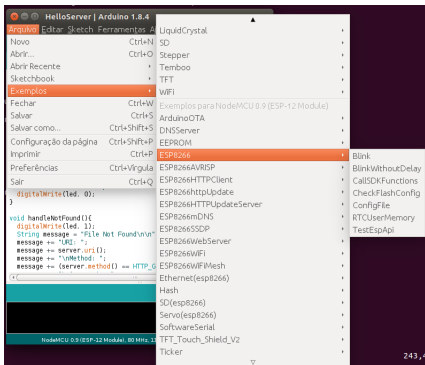
Utilizando
exemplos

Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays



Quando é baixado o suporte para ESP na IDE Arduino são configurados vários exemplos de programas que ajudam não apenas iniciantes mas podem ser uma boa ferramenta para funcionalidades necessárias durante o desenvolvimento de aplicações.

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE

Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial

Utilizando
exemplos

Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

Para facilitar a implementação de alguns módulos é possível utilizar bibliotecas especialmente desenvolvidas para estes produtos. Estes programas podem ser obtidos utilizando o Sketch acessando o menu Sketch, Incluir Biblioteca, Gerenciar Biblioteca.

Alternativamente é possível baixar os arquivos do módulo que deseja instalar e extrair o mesmo na pasta libraries no diretório de instalação da IDE.

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

Classes

- `pinMode(PIN, [INPUT—OUTPUT])`
- `digitalWrite(PIN, [LOW—HIGH])`
- `digitalRead(PIN)`
- `delay(TIME_Ms)`

WI-FI (definidos em ESP8266WiFi.h)

- `WiFi.begin(SSID, SENHA);`
- `WiFi.status()`
- `WiFi.localIP()`

Conectar ao AP

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

```
#include <ESP8266WiFi.h>
```

```
const char* ssid      = "ssid";  
const char* password = "senha_do_ssid";
```

```
void setup() {  
    Serial.begin(9600);  
    delay(10);  
    Serial.print("Conectando_com: ");  
    Serial.println(ssid);  
  
    WiFi.begin(ssid, password);  
  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(500);  
        Serial.print(".");  
    }  
  
    Serial.println("");  
    Serial.println("WiFi_Conectado");  
    Serial.println("IP_address:");  
    Serial.println(WiFi.localIP());  
}
```

```
void loop() {  
  
}
```

Configurar como AP

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando com a IDE Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades Wi-Fi

Motores

Displays

```
#include <ESP8266WiFi.h>

const char* ssid      = "MeuAP";
const char* password = "minhasenha";
IPAddress IP(192,168,200,1);
IPAddress net(255,255,255,0);

void setup() {
  Serial.begin(9600);
  delay(10);
  Serial.print(" Iniciando _çservio:_");

  WiFi.mode(WIFI_AP);
  WiFi.softAPConfig(IP, IP, net)
  WiFi.softAP(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println(" IP _address:_");
  Serial.println(WiFi.localIP());
}

void loop() {
  Serial.println(WiFi.localIP());
}
```

Atendendo requisições (setup)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
#include <ESP8266WiFi.h>

const char* ssid      = "MeuAP";
const char* password  = "minhasenha";
IPAddress IP(192,168,200,1);
IPAddress net(255,255,255,0);
int EXTLED = 5;

WiFiServer server(80);

void setup() {
  Serial.begin(9600);
  delay(10);

  WiFi.mode(WIFI_AP);
  WiFi.softAPConfig(IP, IP, net);
  WiFi.softAP(ssid, password);
  delay(1000);
  server.begin();

  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(EXTLED, OUTPUT);
}
```

Atendendo requisições (loop)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
void loop() {  
  WiFiClient client = server.available();  
  if(!client)  
    return;  
  
  while(!client.available()){  
  
    if (Serial.available() > 0) {  
      data = Serial.readString();  
      Serial.print(" Digitado: ");  
      Serial.println(data);  
      continue;  
    }  
  }  
  
  response = "<html><meta charset='utf-8'/><h1>áOlá cliente " + data + "</h1></html>";  
  Serial.println("ãConexão recebida");  
  client.println("HTTP/1.1 200 OK");  
  client.println("Content-Type: text/html");  
  client.println(""); //Fim do cabeçalho http \n\r  
  client.println(response);  
  
  delay(10);  
}
```

Atendendo requisições (loop)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
void loop() {  
  WiFiClient client = server.available();  
  if (!client)  
    return;  
  
  while (!client.available())  
    delay(1);  
  Serial.println("ãConexão recebida");  
  client.println("HTTP/1.1 200 OK");  
  client.println("Content-Type: text/html");  
  client.println(""); //Fim do cabeçalho http \n\r  
  client.println("<html><meta charset='utf-8'><h1>óOl cliente </h1></html>");  
  
  digitalWrite(EXTLED, HIGH);  
  digitalWrite(LED_BUILTIN, HIGH);  
  delay(1000);  
  digitalWrite(EXTLED, LOW);  
  digitalWrite(LED_BUILTIN, LOW);  
  
  delay(10);  
}
```


Passando parâmetros (loop)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
void loop() {  
  WiFiClient client = server.available();  
  if(!client)  
    return;  
  
  while(!client.available())  
    delay(1);  
  
  String request = client.readStringUntil("\n");  
  String response = "<html><meta_charset='utf-8'/><h1>Nada_a_fazer </h1></html>";  
  client.flush();  
  
  if(request.indexOf("?l=on") > 0){  
    digitalWrite(EXTLED, HIGH);  
    response = "<html><meta_charset='utf-8'/><h1>Luz_Acesa </h1></html>";  
  }else if(request.indexOf("?l=off") > 0)  
    response = "<html><meta_charset='utf-8'/><h1>Luz_Acesa </h1></html>";  
  }  
  
  Serial.println("ãConexo_recebida");  
  client.println("HTTP/1.1_200_OK");  
  client.println("Content-Type:_text/html");  
  client.println("");  
  client.println(response);  
  
  delay(10);  
}
```

Melhorando a interface (loop)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

```
void loop() {  
  WiFiClient client = server.available();  
  if(! client)  
    return;  
  
  while(! client.available())  
    delay(1);  
  
  String request = client.readStringUntil('\r');  
  String response = "<html><style>input{width:500px;height:180px;margin-bottom:10px;}</style><body><div><input type='button' value='On' onclick='location.href=?l=on'></div><div><input type='button' value='Off' onclick='location.href=?l=off'></div></body></html>";  
  response += "<input type='button' value='On' onclick='location.href=?l=on'>";  
  response += "<input type='button' value='Off' onclick='location.href=?l=off'>";  
  client.flush();  
  
  Serial.println(request);  
  
  if(request.indexOf("?l=on") >= 0){  
    digitalWrite(EXTLED, HIGH);  
  }else if(request.indexOf("?l=off") >= 0){  
    digitalWrite(EXTLED, LOW);  
  }  
  
  Serial.println("ãConexo_recebida");  
  client.println("HTTP/1.1 200 OK");  
  client.println("Content-Type: text/html");  
  client.println("");  
  client.println("//Fim do cabecalho http \n\r");  
  client.println(response);  
  
  delay(10);  
}
```

Sobre Motores

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
línguaem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

Ponte H

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
línguaem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

PWM (Pulse-Width Modulation)

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
linguagem
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
Wi-Fi

Motores

Displays

É uma técnica que permite a simulação de sinais analógicos utilizando sistemas digitais. Consiste no envio de sinais em intervalos de tempos fazendo que o dispositivo receba ou não sinais de forma que seja possível a representação de curvas.

- Intensidade de luminosidade;
- Simular controle de velocidade de motores;
- Amplificação de ondas;

Servo Motores

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

Dispositivo que movimenta a partir de informação do angulo passado como parâmetro. Os servomotores utilizam comprimento de onda de sinais digitais para estabelecer quanto tempo o motor fica energizado. Aliado com uma construção mecânica os ângulos de rotação costumam ser bastante precisos.

Motor de passo

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays

Escrevendo no display

NodeMCU

Fabiano
Sardenberg
Kuss

Introdução

Usando o
Arduino IDE

Baixando e
Instalando o
Arduino IDE
Configurando o
Arduino IDE

Programando
com a IDE
Arduino

Estrutura da
língua
Monitor Serial
Utilizando
exemplos
Incluindo
bibliotecas

Funcionalidades
WI-FI

Motores

Displays