

SERVIDOR WEB PARA ENTRADA DIGITAL

((Con período corto de refrescamiento))

Para reducir el periodo de refrescamiento podemos incorporar un fragmento de Javascript.

MATERIALES:

- 1 Wemos D1 R1
- 1 Protoboard mediano
- 1 Resistor 1 KOhm (CA-NE-RO)
- 2 alambres dupont M-M ROJO
- 1 alambre dupont M-M NARANJA
- 1 alambre dupont M-M NEGRO
- 1 Push button NA
- Compilador IDE Arduino

INSTRUCCIONES:

Compilar y grabar el siguiente código en el hardware:

String encabezado = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n";

```
String html_CadenaCruda = R"=====(
<!DOCTYPE html>
<html>
<head>
<meta name='viewport' content='width=device-width, initial-scale=1.0'/>
<meta charset='utf-8'>
<style>
  body {font-size:100%;}
  #main {display: table; margin: auto; padding: 0 10px 0 10px; }
  h2 {text-align:center; }
  p { text-align:center; }
</style>
<script>
  function refresca(PeriodoDeRefrescamiento)
  {
    setTimeout("location.reload(true);", PeriodoDeRefrescamiento);
  }
  window.onload = refresca(100);
</script>

  <title>Lectura de push button HTML-Javascript</title>
</head>

<body>
  <div id='main'>
    <h2>Servidor Web autoactualizable leyendo push button en HTML</h2>
```

```

    <div id='cont'>
        <p>Contador de actualizaciones = %cont%</p>
    </div>
</div>
</body>
</html>
)=====";

```

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

```

const char* ssid = "MEGACABLE-F2379A";
const char* password = "C233FtXd";

```

```

int pinEntrada4 = 4; // Se utilizará la entrada D4 de la Wemos D1
int valorEntrada4 = 0;

```

```
WiFiServer server(80);
```

```

String temporalString = "";
unsigned int cont = 0;

```

```

void setup()
{
    pinMode(pinEntrada4, INPUT); // se prepara como entrada al pin 4
    Serial.begin(115200);
    Serial.println();
    Serial.println("El puerto serie iniciado a 115200");
    Serial.println();

    // Conectarse a la red Wi Fi
    Serial.print(F("Conectando a ")); Serial.println(ssid);
    WiFi.begin(ssid, password);

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

    Serial.println("");
    Serial.println(F("[CONNECTED]"));
    Serial.print("[IP ");
    Serial.print(WiFi.localIP());
    Serial.println("]");
    // iniciar un servidor
    server.begin();
    Serial.println("Servidor iniciado");
}
// -----

```

```

void loop()
{
  // Verificar si un cliente se ha conectado
  WiFiClient client = server.available();
  if (!client) { return; }

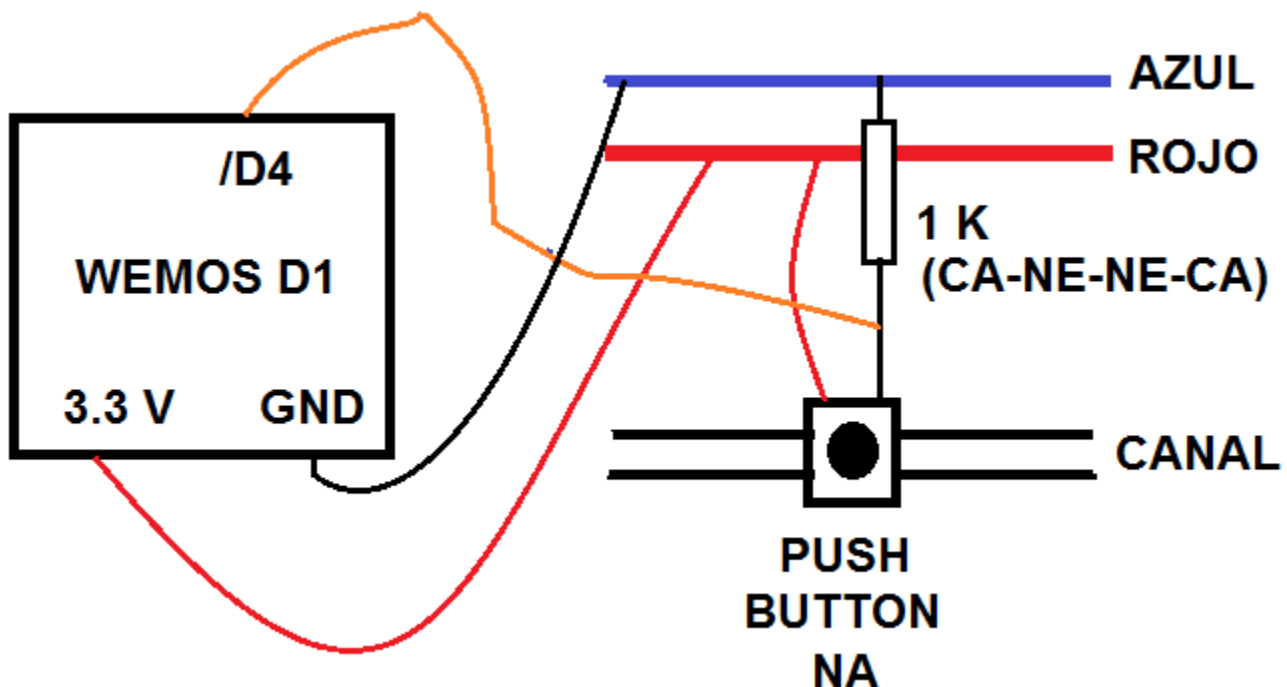
  cont ++;

  temporalString = html_CadenaCruda;
  temporalString.replace("%cont%", String(cont) );

  client.flush();
  client.print( encabezado );
  client.print( temporalString );
  valorEntrada4 = digitalRead(pinEntrada4); //lectura digital de pin
  client.print(" ");
  client.print("El valor digital de entrada es: ");
  client.print(valorEntrada4);
  Serial.print("contador = "); Serial.println(cont);
  delay(5);
  // El cliente será desconectado cuando la función retorna y el objeto 'cliente' es destruido
}

```

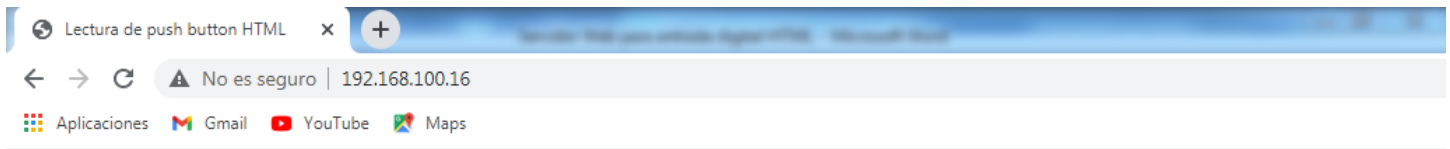
CONEXIONES:



PRUEBA:

Después de grabar en el hardware activar el monitor serie (en el menú, herramientas) y observar la IP que deberá teclearse en el navegador.

Después de teclear la IP debe aparecer lo siguiente:

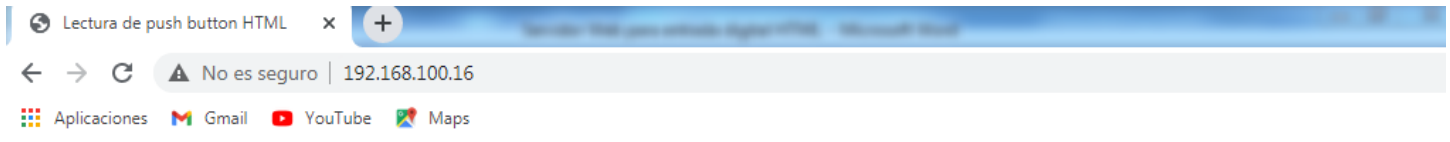


Servidor Web autoactualizable leyendo push button en HTML

Contador de actualizaciones = 7

El valor digital de entrada es: 0

Al presionar el push button debe aparecer lo siguiente:



Servidor Web autoactualizable leyendo push button en HTML

Contador de actualizaciones = 41

El valor digital de entrada es: 1