

ESCUDO ETHERNET COMPATIBLE CON W5100

ETHERNET-W5100-SHIELD



Descripción:

El Ethernet Shield W5100 es un módulo con 16 Kbytes de RAM de búfer y una interfaz serial SPI, el cuál permite que Arduino UNO/MEGA se conecte a una red, soporta hasta 4 conexiones simultáneas, incluye un slot para tarjetas micro-SD, el cual puede ser empleado para almacenar archivos que podrás poner disponibles a través de la red.

El Ethernet Shield W5100 está diseñado para facilitar la implementación de conectividad a Internet, lo que lo hace interesante para MCU y aplicaciones control y monitoreo remoto, domótica, automatización e IoT.

Características:

- Voltaje de Operación: 5V DC
- Controlador Microchip: Ethernet Wiznet W5100
- Velocidad Ethernet: 10/100 Mbps
- Interfaz con el microcontrolador: Host por SPI
- Protocolo: TCP y UDP.
- Compatible: Arduino Uno y Mega
- MicroSD Card: Si

Especificaciones:

Peso	27g
Dimensiones	70.3mm x 53.3mm
Compatibilidad	Arduino Uno y Mega

Parámetros:

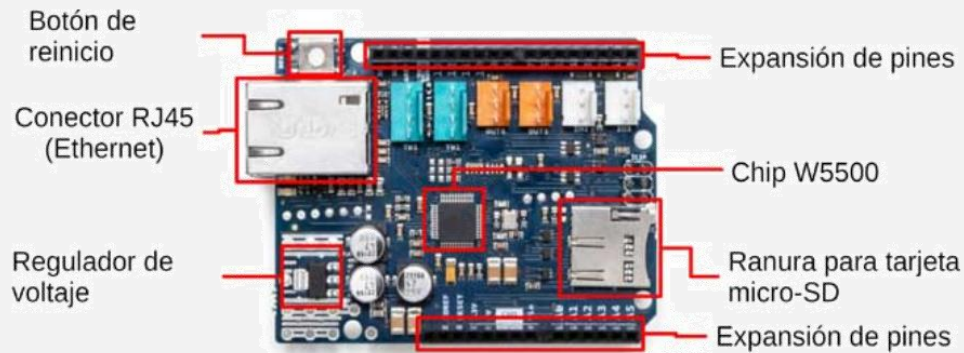
W5100	
Chip	Ethernet W5100
Protocolos de red	TCP Y UDP
Aplicaciones	Arduino UNO y MEGA 1280/2560
Otra característica	Ranura para tarjeta Micro-SD

Conexión del Ethernet Shield para Arduino Uno

El Ethernet Shield se conecta directamente sobre la tarjeta Arduino UNO en los pines digitales 11, 12 y 13 (SPI) para comunicarse con este shield.



Partes del modulo Ethernet:



- Conector Ethernet (RJ45): utilizado para conectar el sistema a la red.
- Regulador de voltaje: utilizado para la alimentación de la placa.
- Botón de reinicio: este botón reinicia tanto el controlador ethernet como la placa Arduino.
- Expansión de pines: conectores que expanden todos los pines del Arduino, por lo tanto, es posible conectar otro módulo u otro Shield encima de este.
- Chip W5100 o W5500: chip principal del módulo.
- Ranura para tarjeta micro-SD: esta ranura puede utilizarse para colocar una tarjeta que almacena archivos de texto, imágenes o cualquier otra información.

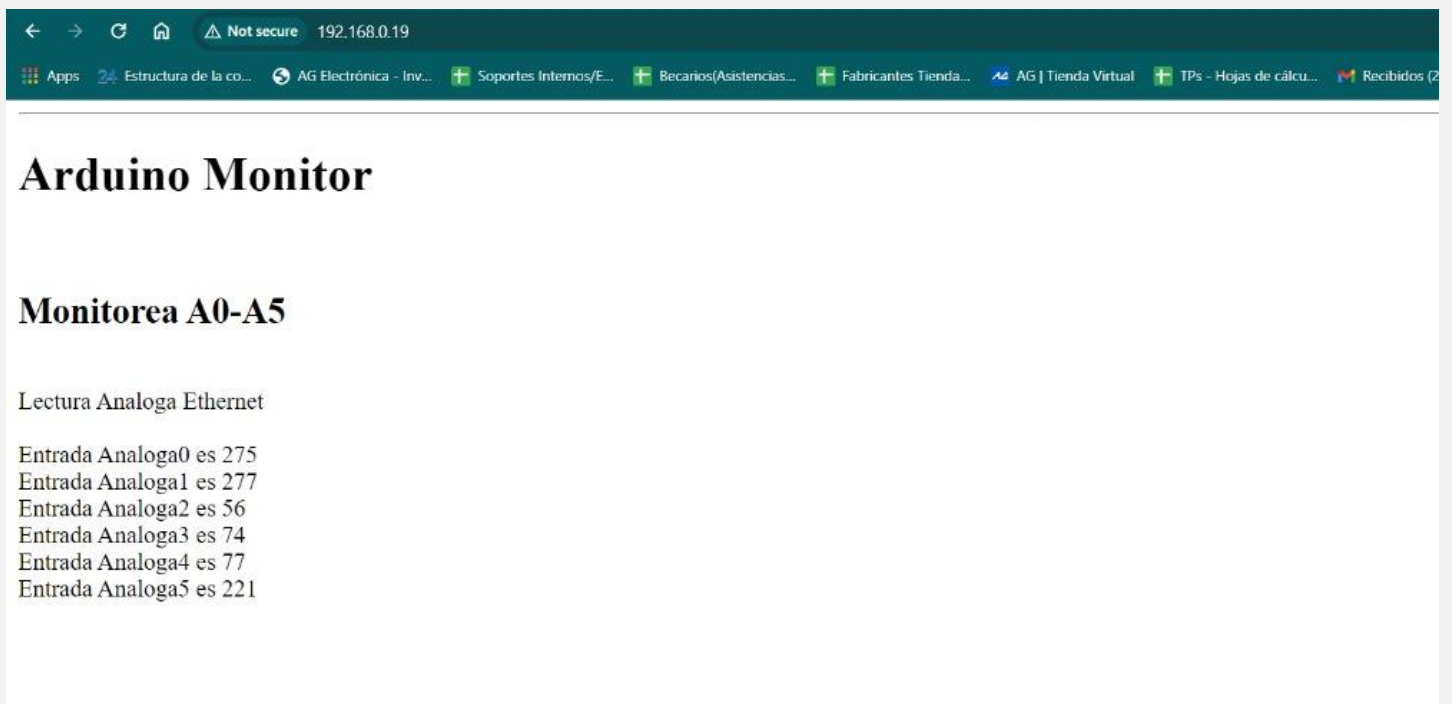
Monitoreo de entradas analógicas desde Internet

Con el siguiente código es posible realizar un monitoreo de las entradas analógicas del Arduino Uno

```
#include <SPI.h>
#include <Ethernet.h>
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192,168,0,19); // Esta dirección
IP debe ser cambiada
EthernetServer server(80);
void setup() {
  Ethernet.begin(mac, ip);
  server.begin();
}
void loop() {
  EthernetClient cliente = server.available();
  if (cliente) {
    boolean currentLineIsBlank = true;
    while (cliente.connected()) {
      if (cliente.available()) {
char c = cliente.read();
        if (c == '\n' && currentLineIsBlank) {
          cliente.println("HTTP/1.1 200 OK");
          cliente.println("Content-Type:
text/html");
          cliente.println("Connection: close");
          cliente.println("Refresh: 3");
          cliente.println();
          cliente.println("<!DOCTYPE HTML>");
          cliente.println("<html>");
          cliente.println("<HEAD>");
          cliente.println("<TITLE>Ethernet
Monitor</TITLE>");
          cliente.println("</HEAD>");
          cliente.println("<BODY>");
          cliente.println("<hr />");
          cliente.println("<H1>Arduino
Monitor</H1>");
          cliente.println("<br />");
          cliente.println("<H2>Monitorea A0-
A5</H2>");
          cliente.println("<br />");
          cliente.println("Lectura Analoga Ethernet");
          cliente.println("<br />");
          cliente.println("<br />");
          for (int puertoAnalogo = 0;
puertoAnalogo < 6; puertoAnalogo++) {
            int lecturaSensor =
analogRead(puertoAnalogo);
            cliente.print("Entrada Analoga");
            cliente.print(puertoAnalogo);
            cliente.print(" es ");
            cliente.print(lecturaSensor);
            cliente.println("<br />");
          }
          cliente.println("<br />");
          cliente.println("</html>");
          break;
        }
      }
    }
  }
}
```

```
if (c == '\n') {
    currentLineIsBlank = true;
}
else if (c != '\r') {
    currentLineIsBlank = false;
}
}
}
}
delay(15);
cliente.stop();
}
}
```

Entrando a cualquier navegador y escribiendo la dirección IP asignada se puede visualizar el monitoreo.



← → ↻ 🏠 Not secure 192.168.0.19

Apps Estructura de la co... AG Electrónica - Inv... Soportes Internos/E... Becarios(Asistencias... Fabricantes Tienda... AG | Tienda Virtual TPs - Hojas de cálcu... Recibidos (2

Arduino Monitor

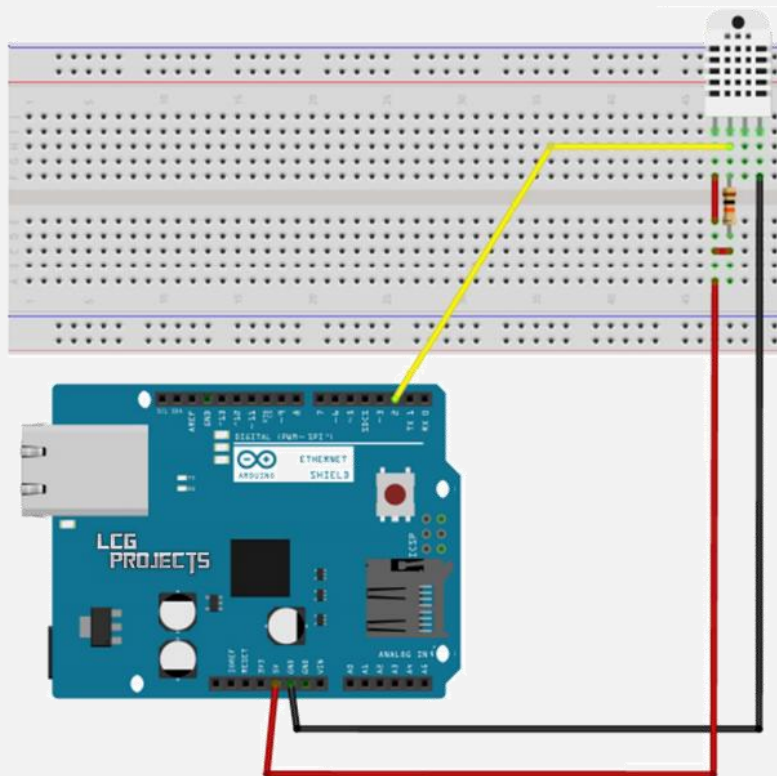
Monitorea A0-A5

Lectura Analoga Ethernet

Entrada Analoga0 es 275
Entrada Analoga1 es 277
Entrada Analoga2 es 56
Entrada Analoga3 es 74
Entrada Analoga4 es 77
Entrada Analoga5 es 221

Monitoreo de temperatura y humedad

Código y conexión para realizar un monitoreo de la temperatura y humedad desde un navegador de Internet utilizando el sensor DHT22.



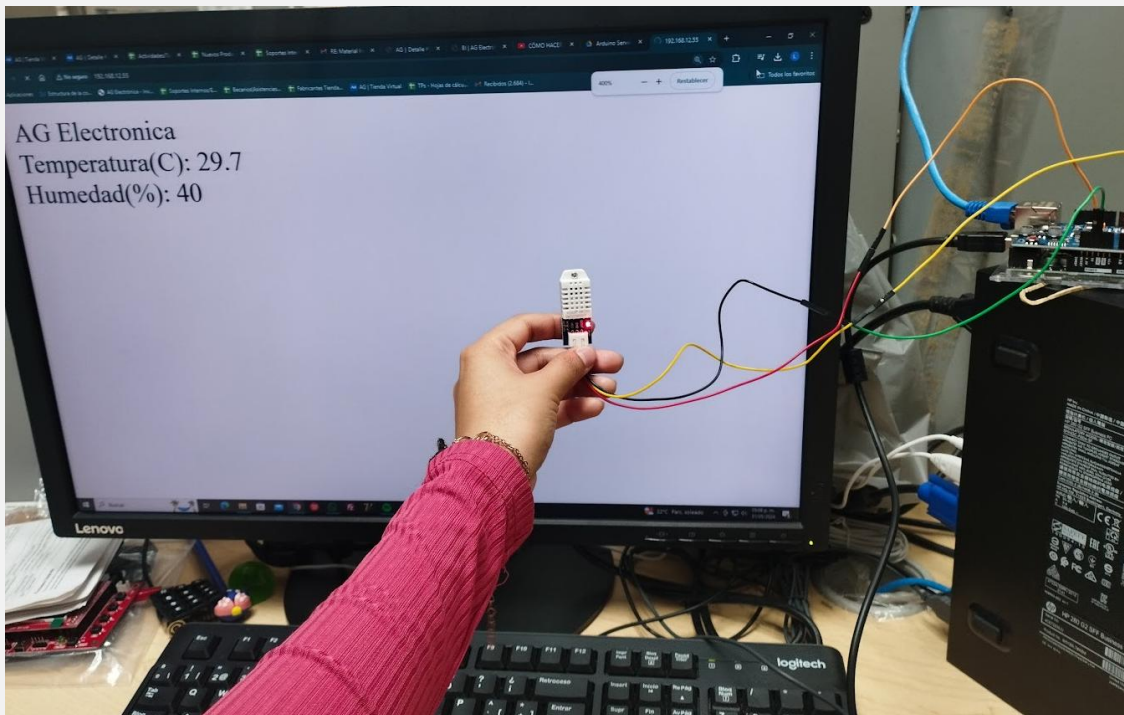
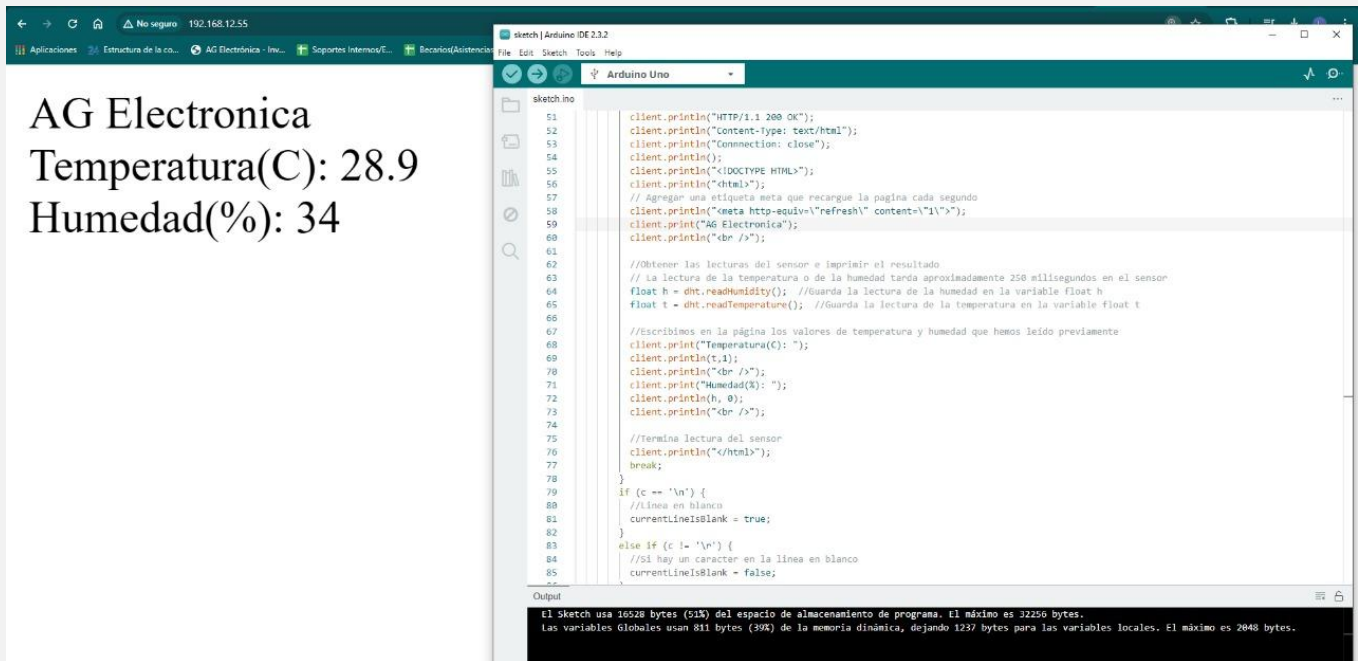
```
#include <SPI.h>
#include <Ethernet.h>
#include <DHT.h> //Introducir la libreria para el sensor DHT22
#include <Wire.h>
#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192,168,12,55);
EthernetServer server(80);
void setup()
{
```

```

Serial.begin(9600);
dht.begin();
while (!Serial) {
  ; }
Ethernet.begin(mac, ip);
server.begin();
Serial.print("server is at ");
Serial.println(Ethernet.localIP());
}
void loop()
{
  EthernetClient client = server.available();
  if (client) {
    Serial.println("new client");
    boolean currentLineIsBlank = true;
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
        Serial.write(c);
        if (c == '\n' && currentLineIsBlank) {
          client.println("HTTP/1.1 200 OK");
          client.println("Content-Type:
text/html");
          client.println("Connection: close");
          client.println();
          client.println("<!DOCTYPE HTML>");
          client.println("<html>");
          client.println("<meta http-
equiv=\"refresh\" content=\"1\">");
          client.print("AG Electronica");
          client.println("<br />");
          //Obtener las lecturas del sensor e imprimir
          el resultado
          float h =
          dht.readHumidity(); //Guarda la lectura de la
          humedad en la variable float h
          float t =
          dht.readTemperature(); //Guarda la lectura de
          la temperatura en la variable float t
          client.print("Temperatura(C): ");
          client.println(t,1);
          client.println("<br />");
          client.print("Humedad(%): ");
          client.println(h, 0);
          client.println("<br />");
          client.println("</html>");
          break;
        }
        if (c == '\n') {
          currentLineIsBlank = true;
        }
        else if (c != '\r') {
          currentLineIsBlank = false;
        }
      }
    }
    delay(1000);
    client.stop();
    Serial.println("client disonnected");}}

```

Visualización



AG Electrónica SAPI de CV
República de El Salvador 20 Piso 2,
Centro Histórico, Centro, 06000
Ciudad de México, CDMX
Teléfono: 55 5130 7210

Realizó Joel Martínez Vazquez

Revisó Ing. Luz Fernanda Domínguez Gómez

Fecha 8/07/2024

