

Q

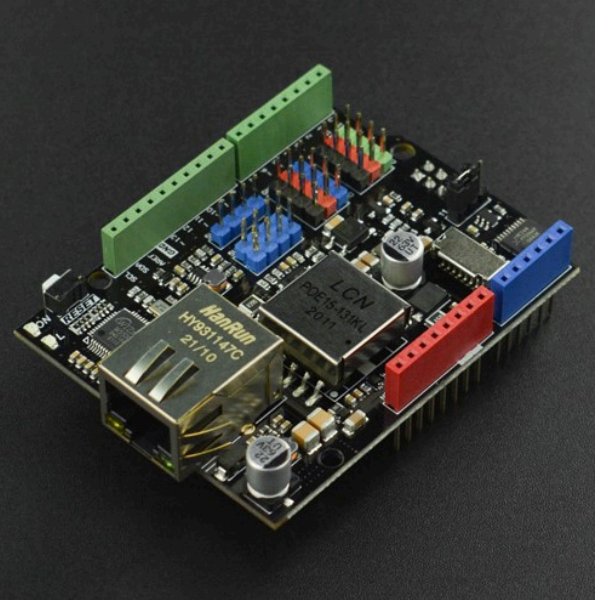
Search

- DFRobot Products (DFR) Index >
- Sensor (SEN) Index >
- Telecom Module (TEL) Index >
- Motor Driver (DRI) Index >
- Fitting (FIT) Index >
- Robot (ROB) Index >
- micro:bit (MBT) Index >
- Kit (KIT) Index >
- Toy (TOY) Index >
- Servos (SER) Index > <
- What is XX series >
- How to select series >
- How it works series >
- PinPong Library >

- Introduction
- Features
- Application
- Specification
- Interface Function
- Ethernet Usage Tutorial
- Requirements
- Connection
- Burning Program
- POE Function Wiring Diagram
- Non-POE Function Wiring Diagram
- Result
- Solutions to Abnormal Situations
- Tutorial of using MicroSD card
- Requirements
- Connection Diagram
- Program Burning
- FAQ
- More Documents

# Ethernet & PoE Shield for Arduino-W5500 Wiki - DFRobot

**SKU:DFR0850** (<https://www.dfrobot.com/product-2370.html>)



(<https://www.dfrobot.com/product-2370.html>)

## Introduction

The W5500 Ethernet expansion board is a new member of the DFRobot Ethernet family. It integrates PoE power supply and W5500 Ethernet chip, which can meet the requirements of general IoT projects.

The W5500 chip is a Hardwired TCP/IP embedded Ethernet controller that provides easier Internet connection to embedded systems. It enables users to have the Internet connectivity in their applications just by using the single chip in which TCP/IP stack, 10/100 Ethernet MAC and PHY embedded. The Hardwired TCP/IP is the market-proven technology that supports TCP, UDP, IPv4, ICMP, ARP, IGMP, and PPPoE protocols. W5500 embeds the 32Kbyte internal memory buffer for the Ethernet packet processing. It's faster and easier way rather than using any other Embedded Ethernet solutions. Besides, the W5500's SPI supports 80MHz speed and new efficient SPI protocol for high-speed network communication. In order to reduce power consumption of the system, W5500 provides WOL (Wake on LAN) and power down mode.

With the same size as Arduino UNO, this W5500 Ethernet expansion board adopts IEEE 802.3at standard-compliant POE Class 0 power classification and offers MicroSD card socket and 2 standard Gravity-4Pin I2C ports, 1 UART, 4 Gravity-3Pin digital ports and 4 analog ports, etc.



### NOTE

- This board has a POE power supply unit whose voltage can reach up to 50V. Although it is not harmful for humman beings, please do not touch the circuit borad with your hands or other conductors since there are some low-voltage circuits on the

<div><div>Q</div><div></div></div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

Requirements

Connection Diagram

Program Burning

FAQ

More Documents

board that cannot bear such high voltage. Otherwise, product damage may occur.

- The transformer and power chip used in the board will generate a lot of heat during high-load use. Therefore, please maintain a good heat dissipation environment during using to avoid damage to the product due to overheating and be careful of high temperature burns.

## Features

- Support hardware TCP/IP protocol: TCP, UDP, ICMP, IPv4, ARP, IGMP, PPPoE
- Support 8-channel independent Sockets
- Support WOL Wake-on-LAN
- Internal 32KB transceiver buffer
- On-board level conversion circuit
- Gravity sensor extension interface

## Application

- Home Network Equipment: set-top box, PVR, digital media center
- Serial Ethernet: access control system, LED display, wireless AP relay, etc.
- Parallel Ethernet: POS machine or handheld printer, duplicator
- USB Ethernet: storage devices, network printers
- Security System: network camera
- Monitoring equipment
- Embedded server

# Specification

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

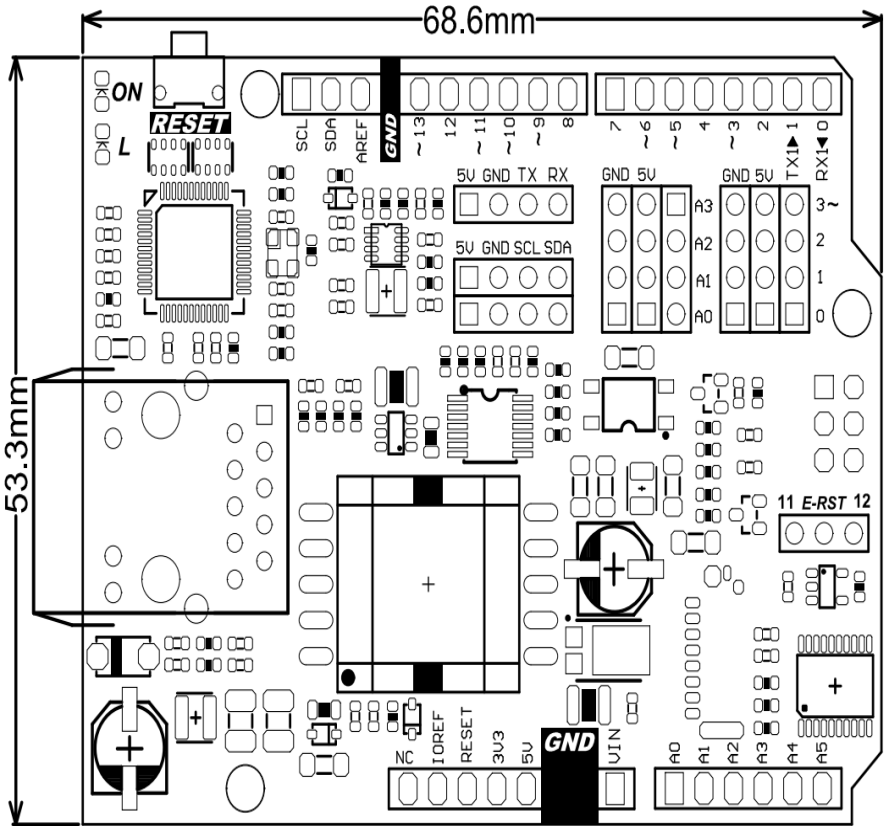
Requirements

Connection Diagram

Program Burning

FAQ

More Documents



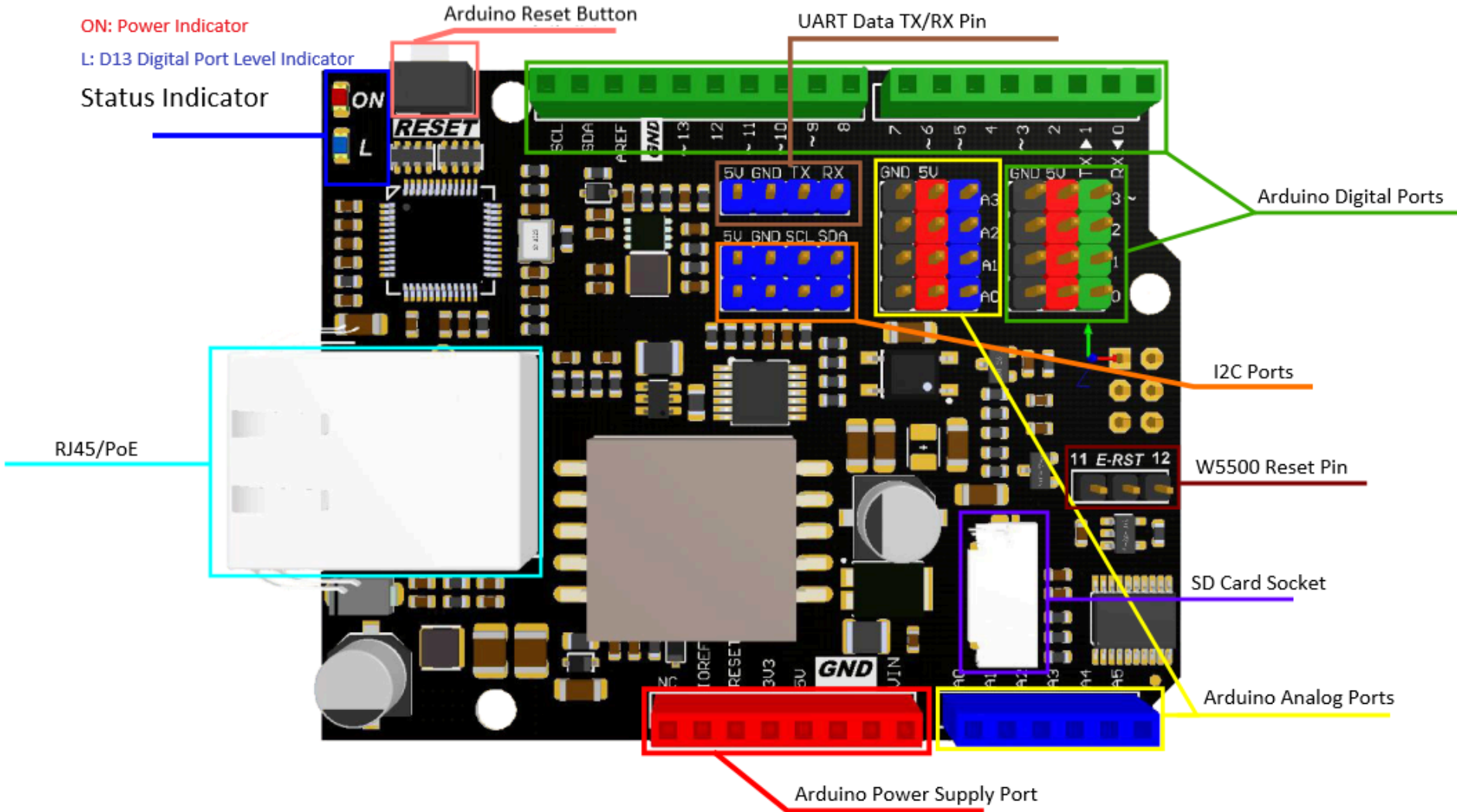
- PoE Input: 48V AC/DC (the product is a Class 0 PD device that meets the 802.3at standard)
- PoE Power: up to 12W (the VIN port can output 12V/1A)
- SPI clock: 80MHz (Max)
- PHY: WIZnet W5500
- PCB Size: 68.6mmx53.3mm / 2.68 x 2.1 inch
- Overall Size: 72.5mmx53.3mm / 2.85 x 2.1 inch

## Interface Function

The board has extra pin 11 and 12 for easy connection to the w5500 reset pin. The board comes with a jumper cap. If you do not use the w5500 reset function, please remove the jumper cap to avoid affecting the normal use of the pins.

<div><div>Q</div><div></div></div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction
Features
Application
Specification
Interface Function
Ethernet Usage Tutorial
Requirements
Connection
Burning Program
POE Function Wiring Diagram
Non-POE Function Wiring Diagram
Result
Solutions to Abnormal Situations
Tutorial of using MicroSD card
Requirements
Connection Diagram
Program Burning
FAQ
More Documents



## Ethernet Usage Tutorial

### Requirements

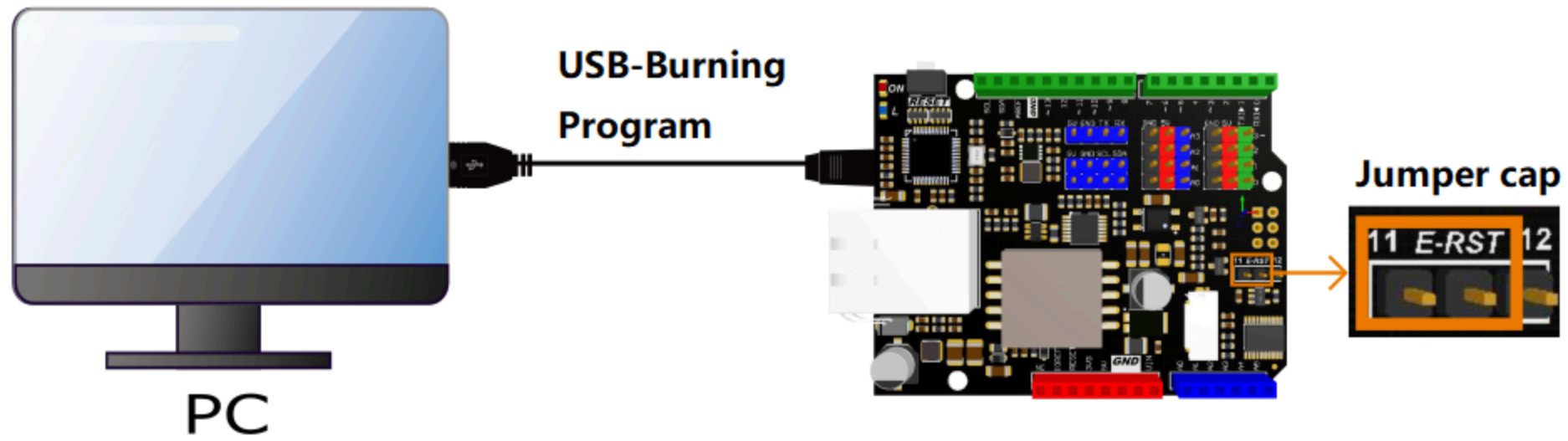
- Hardware
  - POE Ethernet Expansion Board-W5500 (Arduino compatible) X1
  - Leonardo x1
  - Micro USB cable X1
  - Network cable X1
- Software
  - Arduino IDE (<https://www.arduino.cc/en/Main/Software>)



Q	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>
Introduction	
Features	
Application	
Specification	
Interface Function	
Ethernet Usage Tutorial	
Requirements	
Connection	
Burning Program	
POE Function Wiring Diagram	
Non-POE Function Wiring Diagram	
Result	
Solutions to Abnormal Situations	
Tutorial of using MicroSD card	
Requirements	
Connection Diagram	
Program Burning	
FAQ	
More Documents	

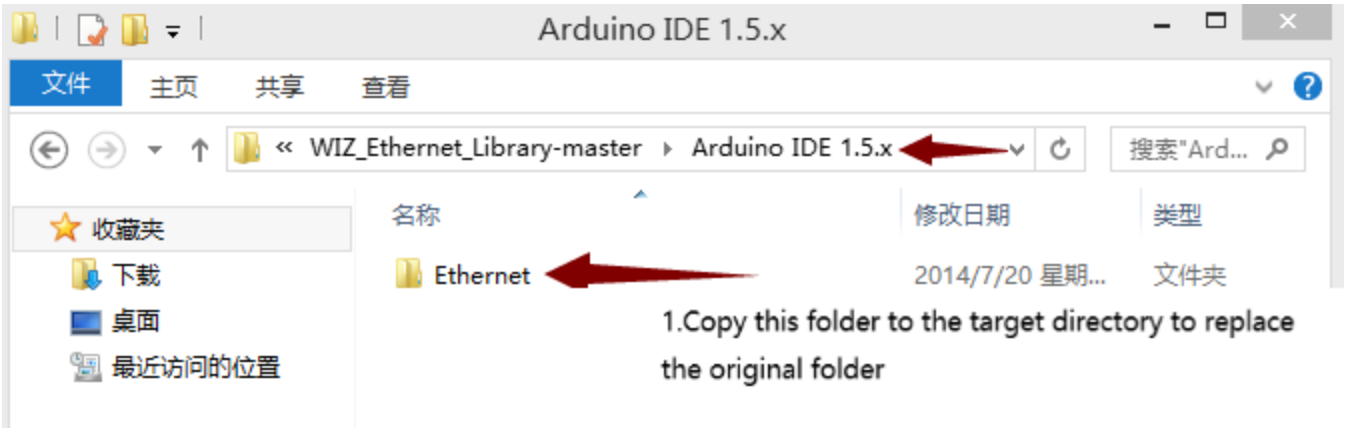
Connection

- First install the POE Ethernet expansion board on the Leonardo board, then connect the E-RST pin on the POE Ethernet expansion board to the D11 pin through the jumper cap, then connect your Leonardo to a computer with the micro USB cable.



Burning Program

- Download the official Arduino library of WIZnet ([https://github.com/Wiznet/WIZ\\_Ethernet\\_Library](https://github.com/Wiznet/WIZ_Ethernet_Library)), unzip the compressed package after downloading, according to the IDE version you are using, put the Ethernet folder in one of the two folders of Arduino IDE 1.0.x and Arduino IDE 1.5.x into the library file of the Arduino IED and overwrite it (overwrite the original Ethernet library), as shown below. Open the Arduino IDE compiler, find the WebServer sample in File -> Examples -> Ethernet and open it.



<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

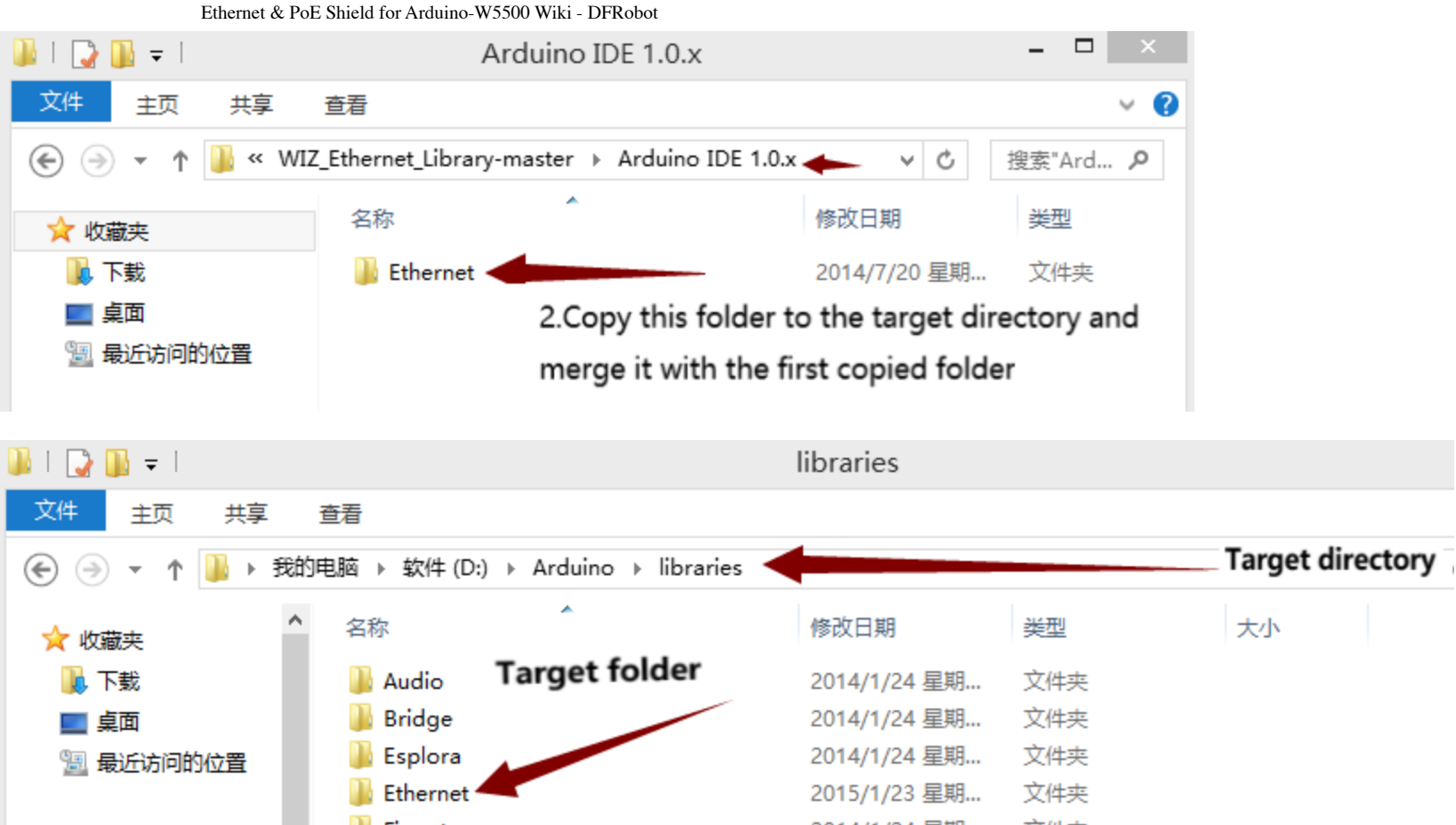
Requirements

Connection Diagram

Program Burning

FAQ

More Documents



- Add the reset pin initialization code at the beginning of the WebServer sample code and burn it.

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction
Features
Application
Specification
Interface Function
Ethernet Usage Tutorial
Requirements
Connection
Burning Program
POE Function Wiring Diagram
Non-POE Function Wiring Diagram
Result
Solutions to Abnormal Situations
Tutorial of using MicroSD card
Requirements
Connection Diagram
Program Burning
FAQ
More Documents

```
/*
  Web Server

  A simple web server that shows the value of the analog input pins.
  using an Arduino Wiznet Ethernet shield.

  Circuit:
  * Ethernet shield attached to pins 10, 11, 12, 13
  * Analog inputs attached to pins A0 through A5 (optional)

  created 18 Dec 2009
  by David A. Mellis
  modified 9 Apr 2012
  by Tom Igoe

  */

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

#define SS      10U    //W5500 CS
#define RST     11U    //W5500 RST

// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network:
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};
IPAddress ip(192, 168, 1, 177);

// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
EthernetServer server(80);

void setup() {

  pinMode(SS, OUTPUT);
  pinMode(RST, OUTPUT);

  digitalWrite(SS, LOW);

  digitalWrite(RST,HIGH); //Reset this module
  delay(200);
  digitalWrite(RST,LOW);
  delay(200);
  digitalWrite(RST,HIGH);
  delay(200);
}
```

<div>Q</div>		
DFRobot Products (DFR) Index	>	Introduction
		Features
		Application
Sensor (SEN) Index	>	Specification
		Interface Function
Telecom Module (TEL) Index	>	Ethernet Usage Tutorial
		Requirements
Motor Driver (DRI) Index	>	Connection
		Burning Program
		POE Function Wiring Diagram
Robot (ROB) Index	>	Non-POE Function Wiring Diagram
micro:bit (MBT) Index	>	Result
		Solutions to Abnormal Situations
Kit (KIT) Index	>	
Toy (TOY) Index	>	Tutorial of using MicroSD card
Servos (SER) Index	> <	Requirements
		Connection Diagram
What is XX series	>	Program Burning
		FAQ
How to select series	>	More Documents
How it works series	>	
PinPong Library	>	

```
// Open serial communications and wait for port to open:
Serial.begin(9600);
//while (!Serial) {
//  ; // wait for serial port to connect. Needed for Leonardo only
//}

// start the Ethernet connection and the server:
Ethernet.begin(mac, ip);
server.begin();
Serial.print("server is at ");
Serial.println(Ethernet.localIP());
}

void loop() {
  // listen for incoming clients
  EthernetClient client = server.available();
  if (client) {
    //Serial.println("new client");
    // an http request ends with a blank line
    boolean currentLineIsBlank = true;
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
        Serial.write(c);
        // if you've gotten to the end of the line (received a newline
        // character) and the line is blank, the http request has ended,
        // so you can send a reply
        if (c == '\n' && currentLineIsBlank) {
          // send a standard http response header
          client.println("HTTP/1.1 200 OK");
          client.println("Content-Type: text/html");
          client.println("Connection: close");// the connection will be closed after completion of the response
          client.println("Refresh: 5"); // refresh the page automatically every 5 sec
          client.println();
          client.println("<!DOCTYPE HTML>");
          client.println("<html>");
          // output the value of each analog input pin
          for (int analogChannel = 0; analogChannel < 6; analogChannel++) {
            int sensorReading = analogRead(analogChannel);
            client.print("analog input ");
            client.print(analogChannel);
            client.print(" is ");
            client.print(sensorReading);
            client.println("<br />");
          }
          client.println("</html>");
          break;
        }
      }
    }
  }
}
```



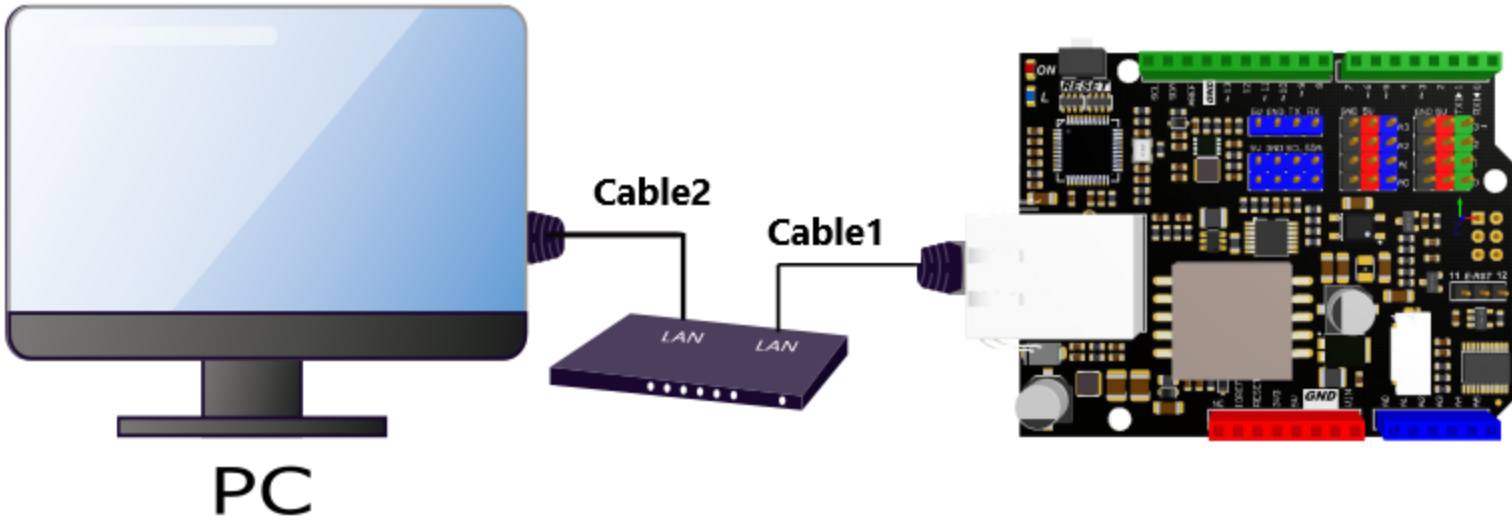
Q	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction
Features
Application
Specification
Interface Function
Ethernet Usage Tutorial
Requirements
Connection
Burning Program
POE Function Wiring Diagram
Non-POE Function Wiring Diagram
Result
Solutions to Abnormal Situations
Tutorial of using MicroSD card
Requirements
Connection Diagram
Program Burning
FAQ
More Documents

```
}
if (c == '\n') {
    // you're starting a new line
    currentLineIsBlank = true;
}
else if (c != '\r') {
    // you've gotten a character on the current line
    currentLineIsBlank = false;
}
}
}
// give the web browser time to receive the data
delay(1);
// close the connection:
client.stop();
Serial.println("client disconnected");
}
```

POE Function Wiring Diagram

- Plug the POE Ethernet expansion board and the computer cable into any LAN port of the POE switch (common switches and routing equipment can also be used. When using common switches and routing equipment, the module needs to be powered).



Non-POE Function Wiring Diagram

- Connect the POE Ethernet expansion board to the computer through a network cable (when directly connected to the PC network port, the module needs to be powered).

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

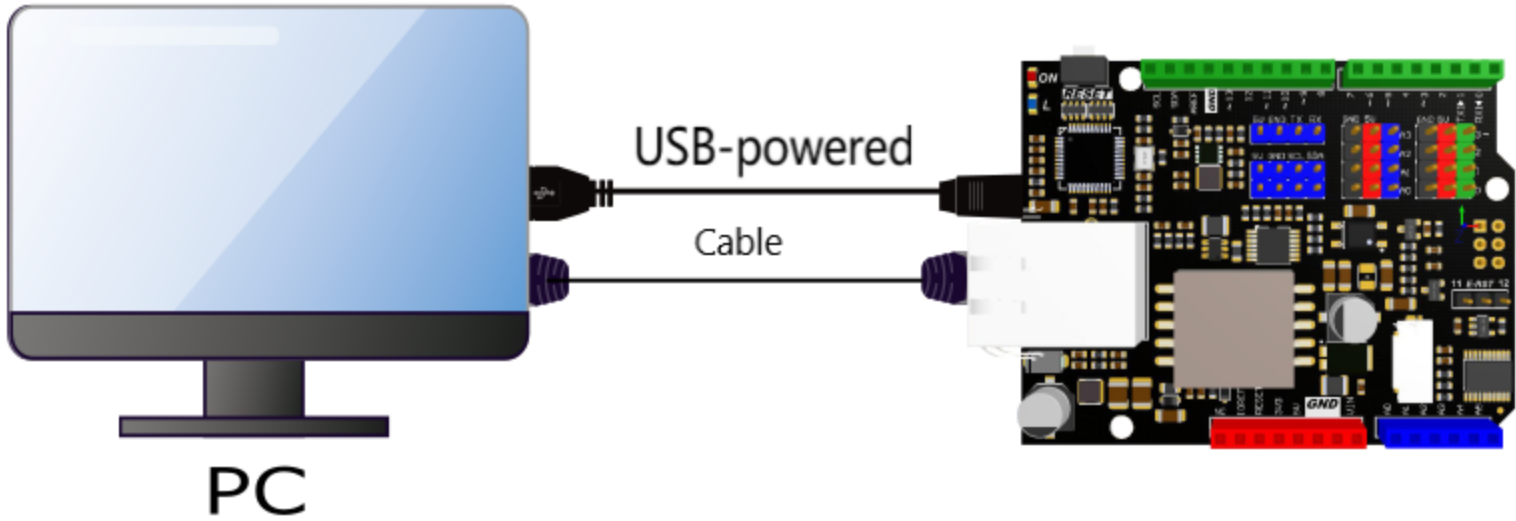
Requirements

Connection Diagram

Program Burning

FAQ

More Documents



Result

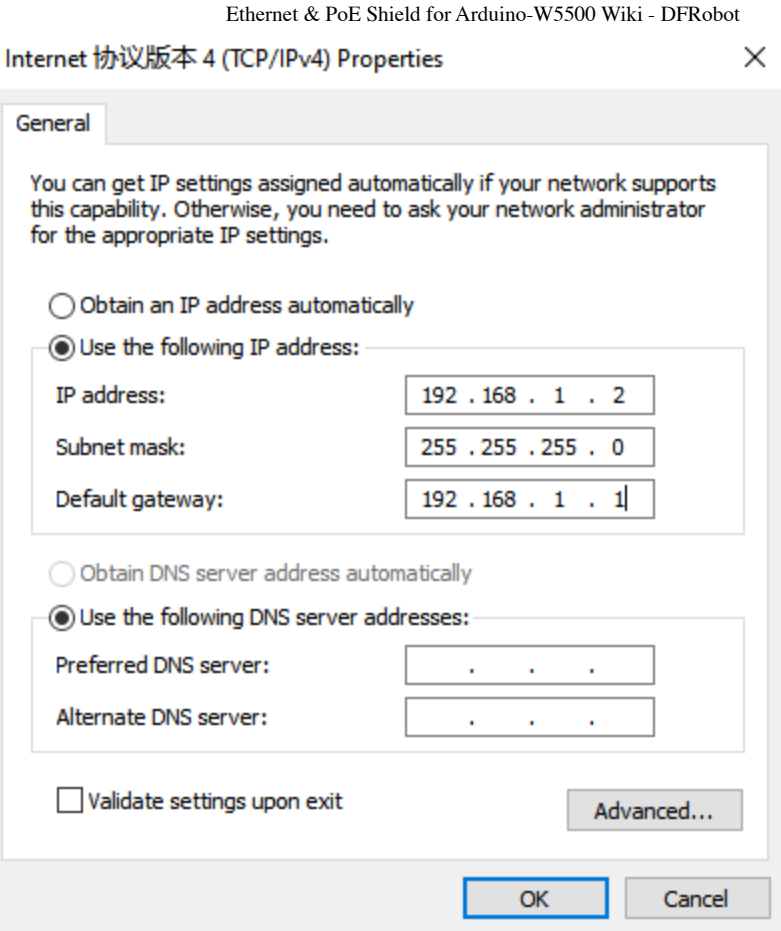
Open the browser -> enter the IP address -> press Enter to open the web page and you can see that the input values of the 6 analog ports of the board are automatically refreshed and displayed on the web page at a frequency of 5S/time, which means that a small Web Server we created has been working normally.

Solutions to Abnormal Situations

For those who have the problem of "Webpage cannot be loaded" in the computer browser, don't worry, please follow the steps below to configure your network adapter:

1. Open the Control Panel-->Network and Internet-->Network and Sharing Center-->Change adapter settings
2. Find the network you connect to the module-->right-click-->Properties-->Find "Internet Protocol Version 4 (TCP/IPv4)" in the network tab and double-click to enter the property settings
3. Enter the parameters shown in the figure below and click OK. After this, open the browser and enter the IP address, then press Enter to open the webpage.

<div>Q</div>		
DFRobot Products (DFR) Index	>	Introduction
Sensor (SEN) Index	>	Features
Telecom Module (TEL) Index	>	Application
Motor Driver (DRI) Index	>	Specification
Fitting (FIT) Index	>	Interface Function
Robot (ROB) Index	>	Ethernet Usage Tutorial
micro:bit (MBT) Index	>	Requirements
Kit (KIT) Index	>	Connection
Toy (TOY) Index	>	Burning Program
Servos (SER) Index	>	POE Function Wiring Diagram
What is XX series	>	Non-POE Function Wiring Diagram
How to select series	>	Result
How it works series	>	Solutions to Abnormal Situations
PinPong Library	>	Tutorial of using MicroSD card
	<	Requirements
		Connection Diagram
		Program Burning
		FAQ
		More Documents



## Tutorial of using MicroSD card

### Requirements

- **Hardware**
  - POE Ethernet Expansion Board-W5500 (Arduino compatible) X1
  - Leonardo x1
  - MicroSD card X1
- **Software**
  - Arduino IDE (<https://www.arduino.cc/en/Main/Software>)

### Connection Diagram

- First install the POE Ethernet expansion board on the Leonardo board and insert the MicroSD card into the MicroSD card slot of the POE Ethernet expansion board, then plug the Micro USB cable into the Leonardo, and connect the other end to the computer USB port.

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

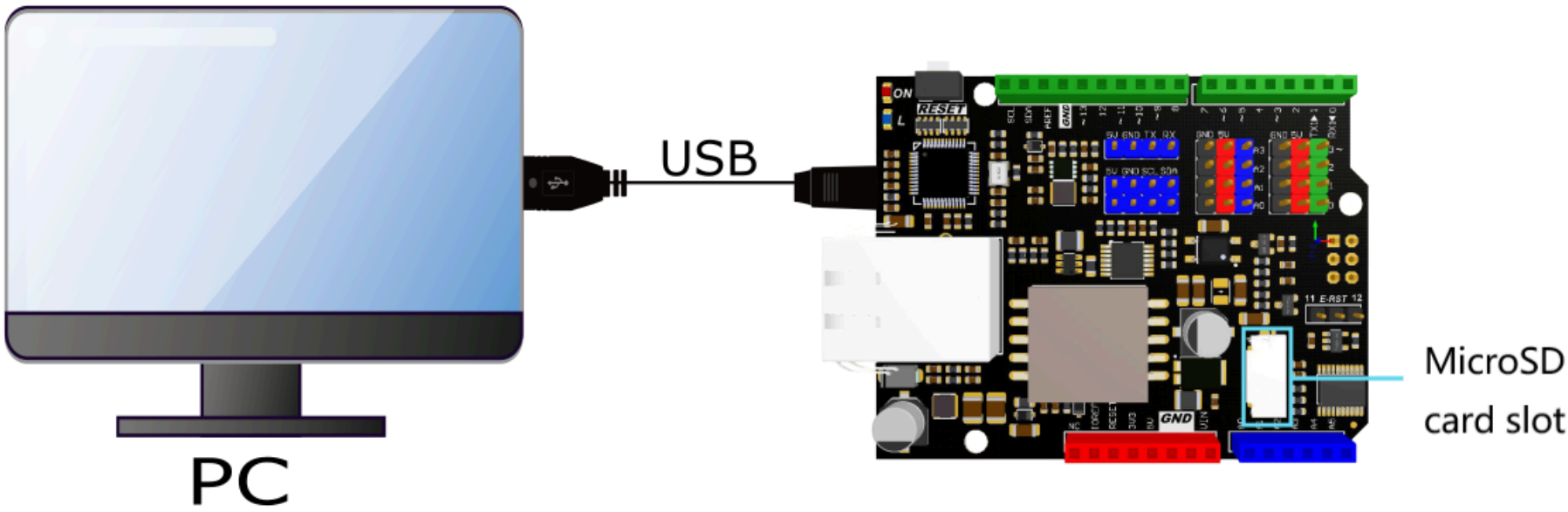
Requirements

Connection Diagram

Program Burning

FAQ

More Documents



### Program Burning

Open the sample program "examples\SD\CardInfo" in the Arduino SD library, and click upload.

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

Requirements

Connection Diagram

Program Burning

FAQ

More Documents

```
/*
  SD card test

  This example shows how use the utility libraries on which the'
  SD library is based in order to get info about your SD card.
  Very useful for testing a card when you're not sure whether its working or not.

  The circuit:
    * SD card attached to SPI bus as follows:
    ** MOSI - pin 11 on Arduino Uno/Duemilanove/Diecimila
    ** MISO - pin 12 on Arduino Uno/Duemilanove/Diecimila
    ** CLK - pin 13 on Arduino Uno/Duemilanove/Diecimila
    ** CS - depends on your SD card shield or module.
            Pin 4 used here for consistency with other Arduino examples

  created  28 Mar 2011
  by Limor Fried
  modified 9 Apr 2012
  by Tom Igoe
  */
// include the SD library:
#include <SD.h>

// set up variables using the SD utility library functions:
Sd2Card card;
SdVolume volume;
SdFile root;

// change this to match your SD shield or module;
// Arduino Ethernet shield: pin 4
// Adafruit SD shields and modules: pin 10
// Sparkfun SD shield: pin 8
const int chipSelect = 4;

void setup()
{
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }

  Serial.print("\nInitializing SD card...");
  // On the Ethernet Shield, CS is pin 4. It's set as an output by default.
  // Note that even if it's not used as the CS pin, the hardware SS pin
  // (10 on most Arduino boards, 53 on the Mega) must be left as an output
```



<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction
Features
Application
Specification
Interface Function
Ethernet Usage Tutorial
Requirements
Connection
Burning Program
POE Function Wiring Diagram
Non-POE Function Wiring Diagram
Result
Solutions to Abnormal Situations
Tutorial of using MicroSD card
Requirements
Connection Diagram
Program Burning
FAQ
More Documents

```
// or the SD library functions will not work.
pinMode(10, OUTPUT);    // change this to 53 on a mega

// we'll use the initialization code from the utility libraries
// since we're just testing if the card is working!
if (!card.init(SPI_HALF_SPEED, chipSelect)) {
    Serial.println("initialization failed. Things to check:");
    Serial.println("* is a card is inserted?");
    Serial.println("* Is your wiring correct?");
    Serial.println("* did you change the chipSelect pin to match your shield or module?");
    return;
} else {
    Serial.println("Wiring is correct and a card is present.");
}

// print the type of card
Serial.print("\nCard type: ");
switch(card.type()) {
    case SD_CARD_TYPE_SD1:
        Serial.println("SD1");
        break;
    case SD_CARD_TYPE_SD2:
        Serial.println("SD2");
        break;
    case SD_CARD_TYPE_SDHC:
        Serial.println("SDHC");
        break;
    default:
        Serial.println("Unknown");
}

// Now we will try to open the 'volume'/'partition' - it should be FAT16 or FAT32
if (!volume.init(card)) {
    Serial.println("Could not find FAT16/FAT32 partition.\nMake sure you've formatted the card");
    return;
}

// print the type and size of the first FAT-type volume
uint32_t volumesize;
Serial.print("\nVolume type is FAT");
Serial.println(volume.fatType(), DEC);
Serial.println();

volumesize = volume.blocksPerCluster();    // clusters are collections of blocks
volumesize *= volume.clusterCount();       // we'll have a lot of clusters
volumesize *= 512;                         // SD card blocks are always 512 bytes
Serial.print("Volume size (bytes): ");
Serial.println(volumesize);
Serial.print("Volume size (Kbytes): ");
```

<div>Q</div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction
Features
Application
Specification
Interface Function
Ethernet Usage Tutorial
Requirements
Connection
Burning Program
POE Function Wiring Diagram
Non-POE Function Wiring Diagram
Result
Solutions to Abnormal Situations
Tutorial of using MicroSD card
Requirements
Connection Diagram
Program Burning
FAQ
More Documents

```
volumesize /= 1024;
Serial.println(volumesize);
Serial.print("Volume size (Mbytes): ");
volumesize /= 1024;
Serial.println(volumesize);

Serial.println("\nFiles found on the card (name, date and size in bytes): ");
root.openRoot(volume);

// list all files in the card with date and size
root.ls(LS_R | LS_DATE | LS_SIZE);
}

void loop(void) {

}
```

Open the serial port monitor window:

<div><div></div><div>Q</div></div>	
DFRobot Products (DFR) Index	>
Sensor (SEN) Index	>
Telecom Module (TEL) Index	>
Motor Driver (DRI) Index	>
Fitting (FIT) Index	>
Robot (ROB) Index	>
micro:bit (MBT) Index	>
Kit (KIT) Index	>
Toy (TOY) Index	>
Servos (SER) Index	> <
What is XX series	>
How to select series	>
How it works series	>
PinPong Library	>

Introduction

Features

Application

Specification

Interface Function

Ethernet Usage Tutorial

Requirements

Connection

Burning Program

POE Function Wiring Diagram

Non-POE Function Wiring Diagram

Result

Solutions to Abnormal Situations

Tutorial of using MicroSD card

Requirements

Connection Diagram

Program Burning

FAQ

More Documents

Ethernet & PoE Shield for Arduino-W5500 Wiki - DFRobot

COM8

Send

Initializing SD card...Wiring is correct and a card is present.

Card type: SDHC

Clusters: 516190

Blocks x Cluster: 1

Total Blocks: 516190

Volume type is: FAT32

Volume size (Kb): 258095

Volume size (Mb): 252

Volume size (Gb): 0.25

Files found on the card (name, date and size in bytes):

SYSTEM~1/ 2021-04-01 17:57:42

WPSETT~1.DAT 2021-04-01 17:57:42 12

INDEXE~1 2021-04-01 17:57:42 76

☒ Autoscroll

☐ Show timestamp

You've reached the end of the file

No line ending

9600 baud

Clear output


## FAQ

- Is it compatible with UNO? -Yes, it is.  
For any other questions, advice or cool ideas to share, please visit the **DFRobot Forum** (<https://www.dfrobot.com/forum/>).

## More Documents

- Schematic (<https://dfimg.dfrobot.com/62b2fb5caa613609f271523c/wiki/d28633d4c8d32fd619b76f9da9aea385.pdf>)

Back to Top 

<div></div>		
DFRobot Products (DFR) Index	>	Introduction
		Features
		Application
Sensor (SEN) Index	>	Specification
		Interface Function
Telecom Module (TEL) Index	>	Ethernet Usage Tutorial
Motor Driver (DRI) Index	>	Requirements
		Connection
Fitting (FIT) Index	>	Burning Program
		POE Function Wiring Diagram
Robot (ROB) Index	>	Non-POE Function Wiring Diagram
micro:bit (MBT) Index	>	Result
Kit (KIT) Index	>	Solutions to Abnormal Situations
Toy (TOY) Index	>	Tutorial of using MicroSD card
Servos (SER) Index	> <	Requirements
		Connection Diagram
What is XX series	>	Program Burning
How to select series	>	FAQ
How it works series	>	More Documents
PinPong Library	>	