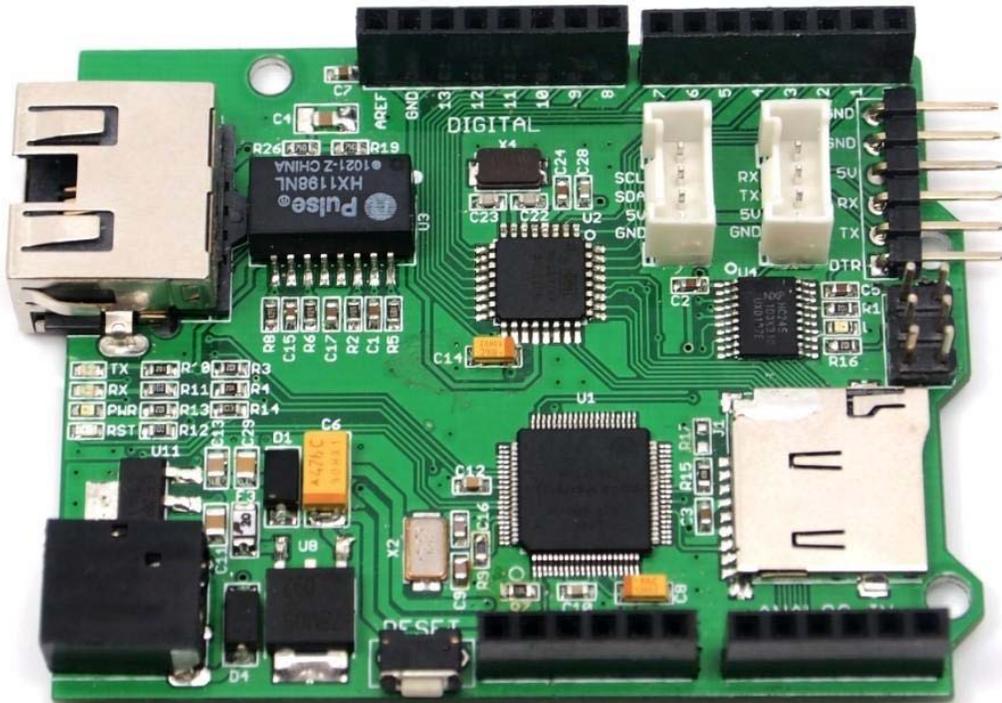


## Seeeduino Ethernet



Seeeduino Ethernet is a compact and multifunctional development platform, which merges data logging and processing, device control and Ethernet communication together into one. It's armed with a MEGA328P chip and a Wiz5100: the former provides an Arduino controlling style and the latter provides TCP and UDP Ethernet communication ability. Plus integrated SD card module, it's convenient and neat for remote data logging, processing or transferring via network.

In this version, we lowered the height of RJ45 to balance it with the headers. Besides, with an I2C and a UART Grove port populated, it's convenient to connect corresponding Grove modules to this board.

## Specification

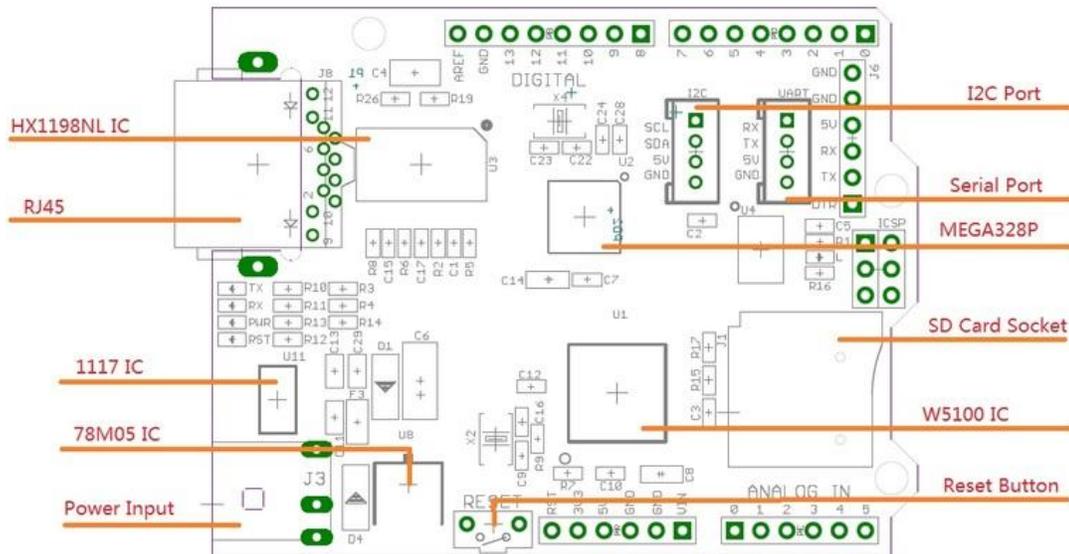
---

- Voltage: 6.2-13V
- Current: 140-550mA
- Supported Card Type: Micro SD card(**must be SanDisk**), FAT/FAT32 (More than 2G is not guaranteed)
- Supported Connection: TCP/UDP
- Net Weight: 23±2g \*
- Ethernet jack: Standard RJ45

## Demonstration

---

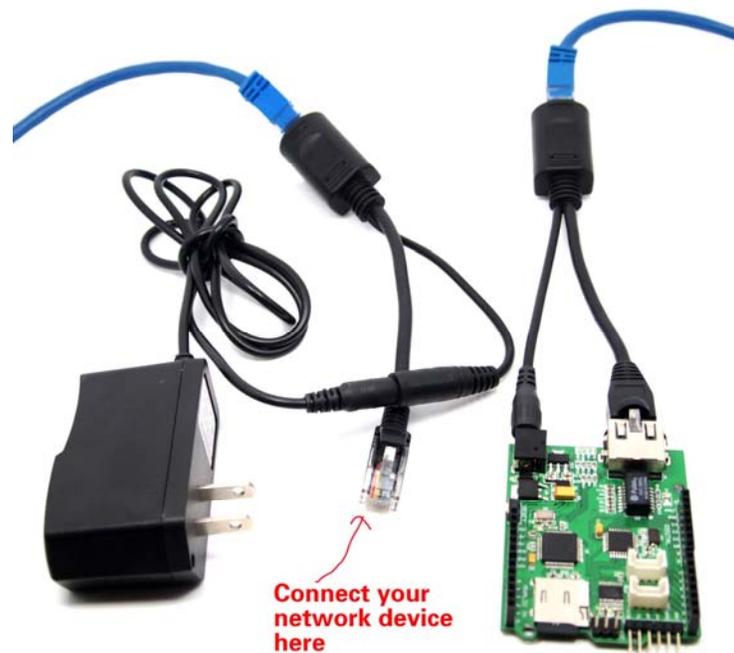
We will test the Seeeduino Ethernet functions of writing or reading information from the SD Card and sending out A/D data to network.



## Step 1: Install the Hardware

Firstly, install the hardware. We have two alternatives to plug in the power.

**Method 1:** You can use one Passive PoE Cable Set to provide power and Ethernet connection at the same time like the picture below.



**Method 2:** Use separate power cable and Ethernet cable.



## Step 2: Programming

After hardware installation, here we continue to try the test code. This program is written to test the functions of writing or reading data from the SD Card and sending out A/D data to network. This demo code can be used as a test program as well as reference if you wanna explore more functions of the board.

### NOTICE:

1. All the ".h" file needed have been pre-installed in Arduino IDE(1.0).
2. Install a Micro SD card. Make sure the Micro SD card is not full and the format is FAT or FAT32.

```
3. /*
4.   SD card read/write
5.
6.   This example shows how to read and write data to and from an SD card file
7.   The circuit:
8.   * SD card attached to SPI bus as follows:
9.   ** MOSI - pin 11
10.    ** MISO - pin 12
11.    ** CLK - pin 13
12.    ** CS - pin 4
13.
14.    Web Server
15.
16.    A simple web server that shows the value of the analog input pins.
17.    using an Arduino Wiznet Ethernet shield.
18.
19.    Circuit:
20.    * Ethernet shield attached to pins 10, 11, 12, 13
21.    * Analog inputs attached to pins A0 through A5 (optional)
22.
23.    created 18 Dec 2009
24.    by David A. Mellis
25.    modified 4 Sep 2010
26.    by Tom Igoe
27.
28.    */
29.
30.    #include <SD.h>
31.    #include <SPI.h>
32.    #include <Ethernet.h>
33.
34.    // Enter a MAC address and IP address for your controller below.
35.    // The IP address will be dependent on your local network:
36.    byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
37.    IPAddress ip(192,168,1,177);
38.
39.    // Initialize the Ethernet server library
40.    // with the IP address and port you want to use
41.    // (port 80 is default for HTTP):
42.    EthernetServer server(80);
43.
```

```

44.   File myFile;
45.
46.   void setup()
47.   {
48.       Serial.begin(9600);
49.       Serial.print("Initializing SD card...");
50.       // On the Ethernet Shield, CS is pin 4. It's set as an output by
default.
51.       // Note that even if it's not used as the CS pin, the hardware SS
pin
52.       // (10 on most Arduino boards, 53 on the Mega) must be left as an
output
53.       // or the SD library functions will not work.
54.       pinMode(10, OUTPUT);
55.       // pinMode(4,OUTPUT);
56.       if (!SD.begin(4)) {
57.           Serial.println("initialization failed!");
58.           return;
59.       }
60.       Serial.println("initialization done.");
61.
62.       // open the file. note that only one file can be open at a time,
63.       // so you have to close this one before opening another.
64.
65.       myFile = SD.open("test.txt",FILE_WRITE);
66.       // if the file opened or created okay, write to it:
67.       if (myFile) {
68.           Serial.print("Writing to test.txt...");
69.           myFile.println("testing 1, 2, 3.");
70.           // close the file:
71.           myFile.close();
72.           Serial.println("done.");
73.       } else {
74.           // if the file didn't open, print an error:
75.           Serial.println("error opening test.txt");
76.       }
77.
78.       // re-open the file for reading:
79.       myFile = SD.open("test.txt");
80.       if (myFile) {
81.           Serial.println("test.txt:");
82.
83.           // read from the file until there's nothing else in it:
84.           while (myFile.available()) {
85.               Serial.write(myFile.read());
86.           }
87.           // close the file:
88.           myFile.close();
89.       } else {
90.           // if the file didn't open, print an error:
91.           Serial.println("error opening test.txt");
92.       }
93.       // start the Ethernet connection and the server:
94.       Ethernet.begin(mac, ip);
95.       server.begin();
96.   }
97.   unsigned char buff[6];

```

```

98. void loop()
99. {
100.     EthernetClient client = server.available();
101.     if (client) {
102.         // an http request ends with a blank line
103.         boolean currentLineIsBlank = true;
104.         while (client.connected()) {
105.             if (client.available()) {
106.                 char c = client.read();
107.                 // if you've gotten to the end of the line (received a
newline
108.                 // character) and the line is blank, the http request
has ended,
109.                 // so you can send a reply
110.                 if (c == '\n' && currentLineIsBlank) {
111.                     // send a standard http response header
112.                     client.println("HTTP/1.1 200 OK");
113.                     client.println("Content-Type: text/html");
114.                     client.println();
115.
116.                     // output the value of each analog input pin
117.                     for (int analogChannel = 0; analogChannel < 6;
analogChannel++) {
118.                         client.print("analog input ");
119.                         client.print(analogChannel);
120.                         client.print(" is ");
121.                         client.print(analogRead(analogChannel));
122.                         client.println("<br />");
123.                         buff[analogChannel] =
analogRead(analogChannel);
124.                     }
125.                     break;
126.                 }
127.                 if (c == '\n') {
128.                     // you're starting a new line
129.                     currentLineIsBlank = true;
130.                 }
131.                 else if (c != '\r') {
132.                     // you've gotten a character on the current line
133.                     currentLineIsBlank = false;
134.                 }
135.             }
136.         }
137.         // give the web browser time to receive the data
138.         delay(1);
139.         // close the connection:
140.         client.stop();
141.         myFile = SD.open("test.txt", FILE_WRITE);
142.         if (myFile) {
143.             Serial.println("test.txt:");
144.             for (int analogChannel = 0; analogChannel < 6;
analogChannel++) {
145.                 myFile.print("analog input ");
146.                 myFile.print(analogChannel);
147.                 myFile.print(" is ");
148.                 myFile.println(analogRead(analogChannel));
149.             }

```

```

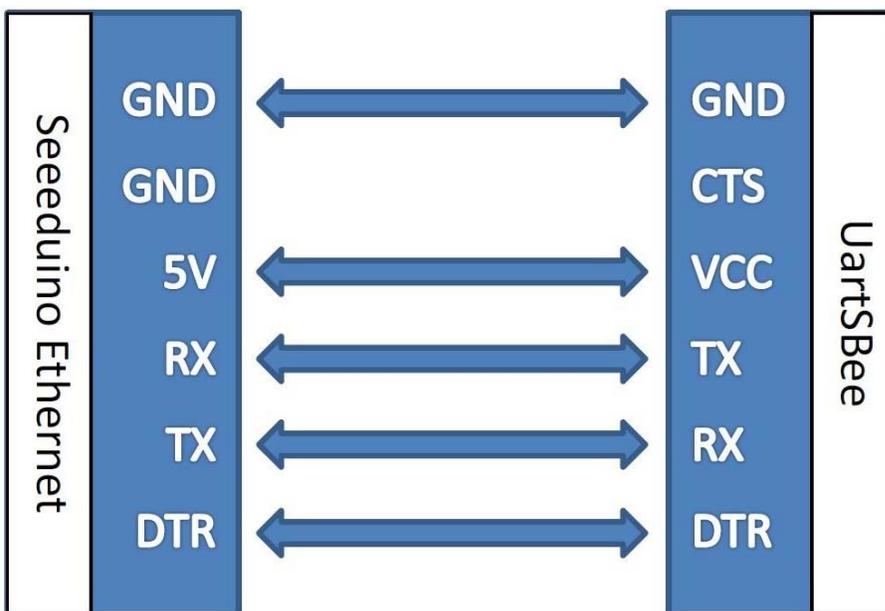
150.         // read from the file until there's nothing else in it:
151.         myFile.close();
152.     }
153.     else {
154.         // if the file didn't open, print an error:
155.         Serial.println("error opening test.txt");
156.     }
157.
158.     myFile = SD.open("test.txt");
159.     if (myFile) {
160.         Serial.println("test.txt:");
161.
162.         // read from the file until there's nothing else in it:
163.         while (myFile.available()) {
164.             Serial.write(myFile.read());
165.         }
166.         // close the file:
167.         myFile.close();
168.     }
169.     else {
170.         // if the file didn't open, print an error:
171.         Serial.println("error opening test.txt");
172.     }
173. }
174. }

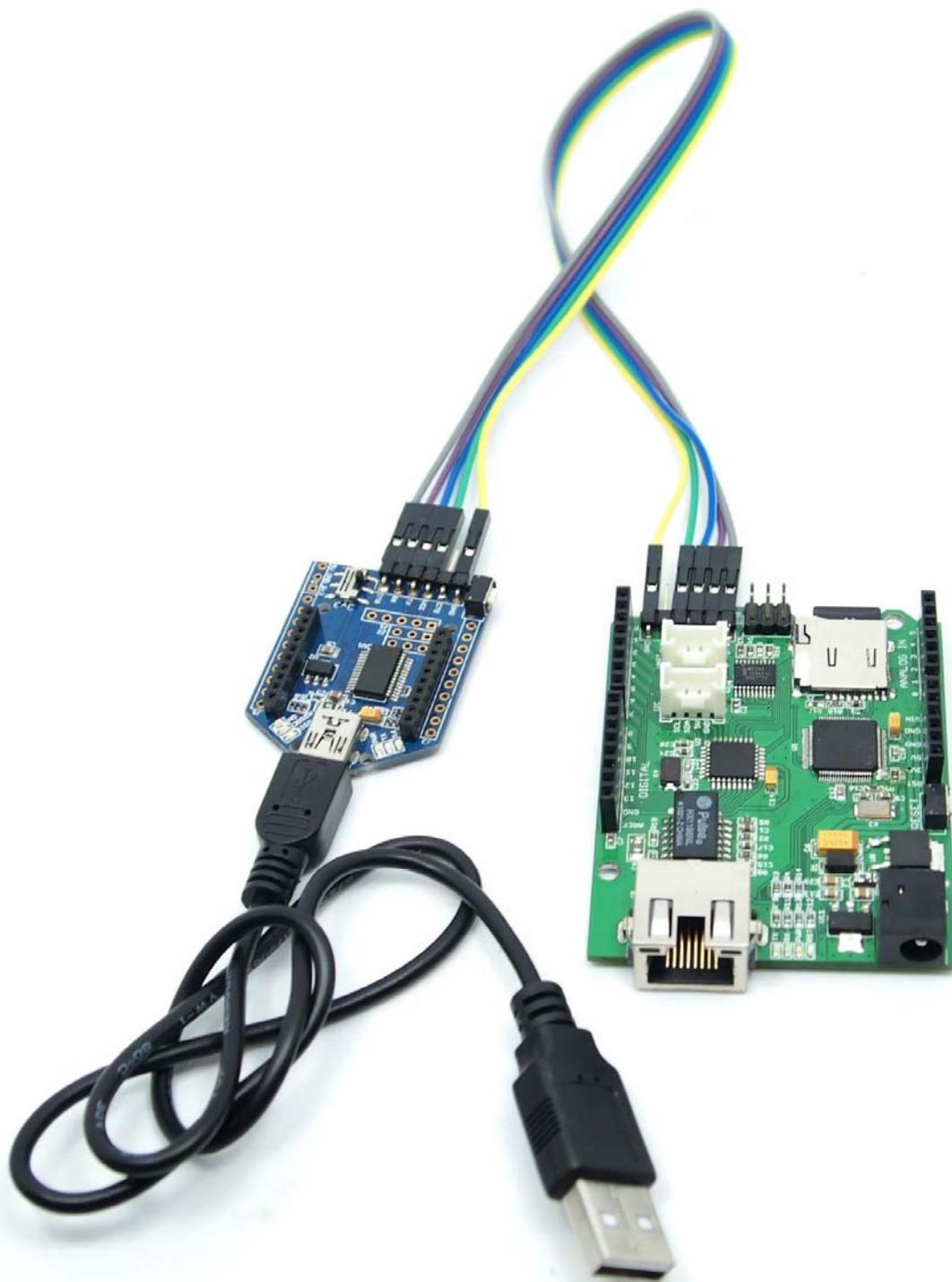
```

### Step 3: Download the program

Because there is no USB connector on the board. Seeeduino Ethernet requires a UartSBee to download the program.

Connect the UartSBee to Seeeduino Ethernet as below:



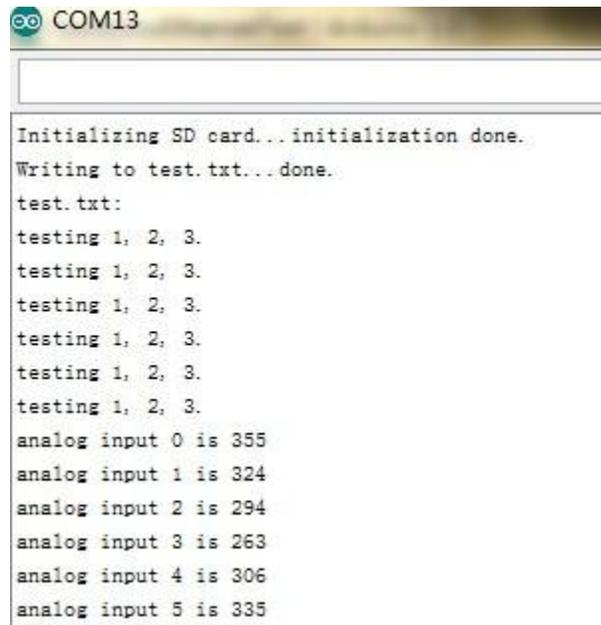


**Warning**

Please select Seeeduino V3.0 as board in tool menu to download the program.

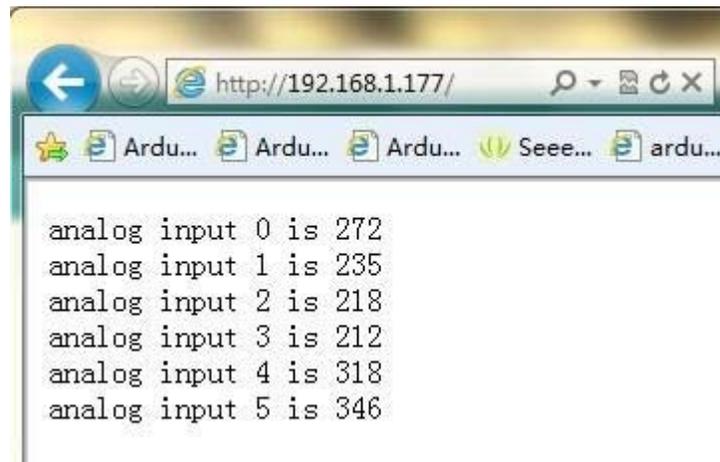
## Step 4: Result testing

**NOTICE:** Make sure the Seeeduino Ethernet and your computer are in the same local area network. After the program was downloaded, open the Serial Monitor. You will check the status of the board.



```
COM13  
Initializing SD card... initialization done.  
Writing to test.txt... done.  
test.txt:  
testing 1, 2, 3.  
analog input 0 is 355  
analog input 1 is 324  
analog input 2 is 294  
analog input 3 is 263  
analog input 4 is 306  
analog input 5 is 335
```

Open a web browser and enter the web address:192.168.1.177, then you can check the data sent out by Seeeduino Ethernet:



```
http://192.168.1.177/  
Ardu... Ardu... Ardu... See... ardu...  
analog input 0 is 272  
analog input 1 is 235  
analog input 2 is 218  
analog input 3 is 212  
analog input 4 is 318  
analog input 5 is 346
```

## Resources

- **Seeedduino Ethernet Eagle Files**
- **Seeedduino Ethernet Schematic.pdf**
- **This documentation is licensed under the Creative Commons Attribution-Share Alike License 3.0. Source code and libraries are licensed under GPL/LGPL, see source code files for details.**
- **Please reformat SD card to FAT/FTA32 .If it still can't work ,suggest to download the SD Formatter and reformat SD card by this software.**

## Tech Support

**Please do not hesitate to contact [techsupport@seeed.cc](mailto:techsupport@seeed.cc) if you have any technical issue. Or submit the issue into our forum.**