Examples > WiFi Library

WiFi Chat Server

A simple server that distributes any incoming messages to all connected clients. To use, open a terminal window, telnet to your WiFi shield's IP address, and type away. Any incoming text will be sent to all connected clients (including the one typing). Additionally, you will be able to see the client's input in your serial monitor as well.

Hardware Required

- Arduino WiFi Shield
- Shield-compatible Arduino board

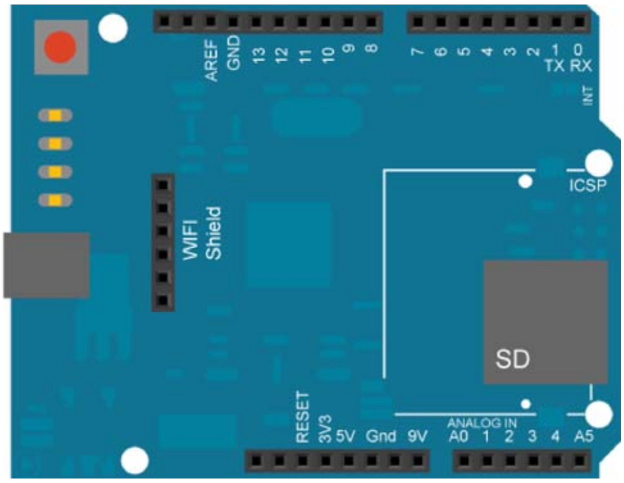
Circuit

The WiFi shield uses pins 10, 11, 12, and 13 for the SPI connection to the HDG104 module. Digital pin 4 is used to control the slave select pin on the SD card.

You should have access to a 802.11b/g wireless network that connects to the internet for this example. You will need to change the network settings in the sketch to correspond to your particular networks SSID.

For networks using WPA/WPA2 Personal encryption, you need the SSID and password. The shield will not connect to networks using WPA2 Enterprise encryption.

WEP network passwords are hexadecimal strings known as keys. A WEP network can have 4 different keys; each key is assigned a "Key Index" value. For WEP encrypted networks, you need the SSID, the key, and key number.



(//www.arduino.cc/en/uploads/Tutorial/WiFiShield_bb.png)

image developed using Fritzing (http://www.fritzing.org). For more circuit examples, see the Fritzing project page (http://fritzing.org/projects/)

In the above image, the Arduino would be stacked below the WiFi shield.

Code:

```
/*
 Chat  Server

 A simple server that distributes any incoming messages to all
 connected clients.  To use telnet to  your device's IP address and type.
 You can see the client's input in the serial monitor as well.

 This example is written for a network using WPA encryption. For
 WEP or WPA, change the Wifi.begin() call accordingly.


 Circuit:
 * WiFi shield attached

 created 18 Dec 2009
 by David A. Mellis
 modified 31 May 2012
 by Tom Igoe

 */

#include <SPI.h>
#include <WiFi.h>

char ssid[] = "yourNetwork"; //  your network SSID (name)
char pass[] = "secretPassword";    // your network password (use for WPA, or use as key for WEP)

int keyIndex = 0;            // your network key Index number (needed only for WEP)

int status = WL_IDLE_STATUS;

WiFiServer server(23);

boolean alreadyConnected = false; // whether or not the client was connected previously

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

  // check for the presence of the shield:
  if (WiFi.status() == WL_NO_SHIELD) {
    Serial.println("WiFi shield not present");
    // don't continue:
    while(true);
  }

  // attempt to connect to Wifi network:
  while ( status != WL_CONNECTED) {
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(ssid);
    // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
    status = WiFi.begin(ssid, pass);

    // wait 10 seconds for connection:
    delay(10000);
  }
  // start the server:
  server.begin();
  // you're connected now, so print out the status:
  printWifiStatus();
 }


void loop() {
  // wait for a new client:
  WiFiClient client = server.available();


  // when the client sends the first byte, say hello:
  if (client) {
    if (!alreadyConnected) {
      // clead out the input buffer:
      client.flush();
      Serial.println("We have a new client");
      client.println("Hello, client!");
      alreadyConnected = true;
    }

    if (client.available() > 0) {
      // read the bytes incoming from the client:
      char thisChar = client.read();
      // echo the bytes back to the client:
      server.write(thisChar);
      // echo the bytes to the server as well:
      Serial.write(thisChar);
    }
  }
}


void printWifiStatus() {
  // print the SSID of the network you're attached to:
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your WiFi shield's IP address:
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);

  // print the received signal strength:
  long rssi = WiFi.RSSI();
  Serial.print("signal strength (RSSI):");
  Serial.print(rssi);
  Serial.println(" dBm");
}
```

[Get Code] (//www.arduino.cc/en/Tutorial/WiFiChatServer?action=sourceblock&num=1)

See Also:

- ConnectNoEncryption (//www.arduino.cc/en/Tutorial/ConnectNoEncryption) : Demonstrates how to connect to an open network
- ConnectWithWEP (//www.arduino.cc/en/Tutorial/ConnectWithWEP) : Demonstrates how to connect to a network that is encrypted with WEP
- ConnectWithWPA (//www.arduino.cc/en/Tutorial/ConnectWithWPA) : Demonstrates how to connect to a network that is encrypted with WPA2 Personal
- ScanNetworks (//www.arduino.cc/en/Tutorial/ScanNetworks) : Displays all WiFi networks in range
- WiFiChatServer (//www.arduino.cc/en/Tutorial/WiFiChatServer) : Set up a simple chat server
- WiFiWebClient (//www.arduino.cc/en/Tutorial/WiFiWebClient) : Connect to a remote webserver
- WiFiWebClientRepeating (//www.arduino.cc/en/Tutorial/WiFiWebClientRepeating): Repeatedly make HTTP calls to a server
- WiFiWebServer (//www.arduino.cc/en/Tutorial/WiFiWebServer) : Serve a webpage from the WiFi shield

Share