



# TinySine WIFI Shield

## User Manual



Tinysine Electronics @ 2013

Version 1.0

## INTRODUCTION

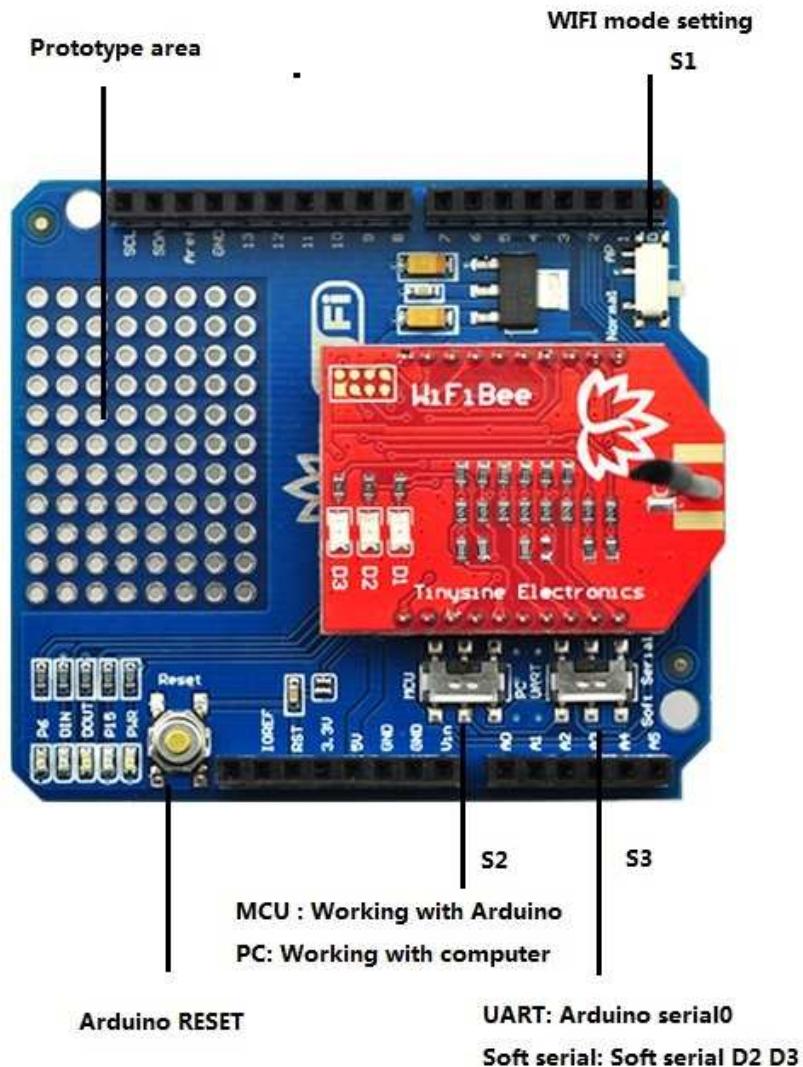
Tinysine WIFI shield based on Roving Networks RN-XV module. It can be separate two parts – Tinysine Bee shield and WiFiBee module. It provides the bridging from TTL serial port communication to IEEE802.11b/g/n wireless communication. So any device with TTL serial ports can easily be connected with this WIFI shield and controlled and managed remotely through a wireless network. Different kinds of communication protocols and encryption algorithms are integrated with the module. The Arduino architecture enables you to easily integrate this module into any Arduino based project.

This shield can connect your router and join your local network. It can also be set to AP mode. That means you can control your arduino project point to point by WIFI even without a router!

## SPECIFICATIONS

- Based on common 802.15.4
- Ultra low power: 4uA sleep mode, 38mA active
- Onboard TCP/IP stack includes DHCP, UDP, DNS, ARP, ICMP, HTTP client, FTP client and TCP
- Firmware configurable transmit power: 0dBm to 12dBm
- Hardware interfaces:TTL UART, soft serial
- Host data rate up to 464Kbps over UART
- Supports soft AP and infrastructure networking
- Real-time clock for time-stamping, auto-sleep, and auto-wakeup modes
- 3dbi antenna

## Overview



### Switch setting

Function	S1	S2	S3
AP mode	AP	-	-
Work with PC	Normal	PC	UART
Work with Arduino Serial	Normal	MCU	UART
Work with Arduino Soft Serial	Normal	MCU	Soft Serial

## LED States

Condition	Red LED-D3	Yellow LED-D2	Green LED-D1
On solid	-	-	Connected over TCP
Fast blink	Not associated	Rx/Tx data transfer	No IP address
Slow blink	Associated, no Internet	-	IP address OK
Off	Associated, Internet OK	-	-

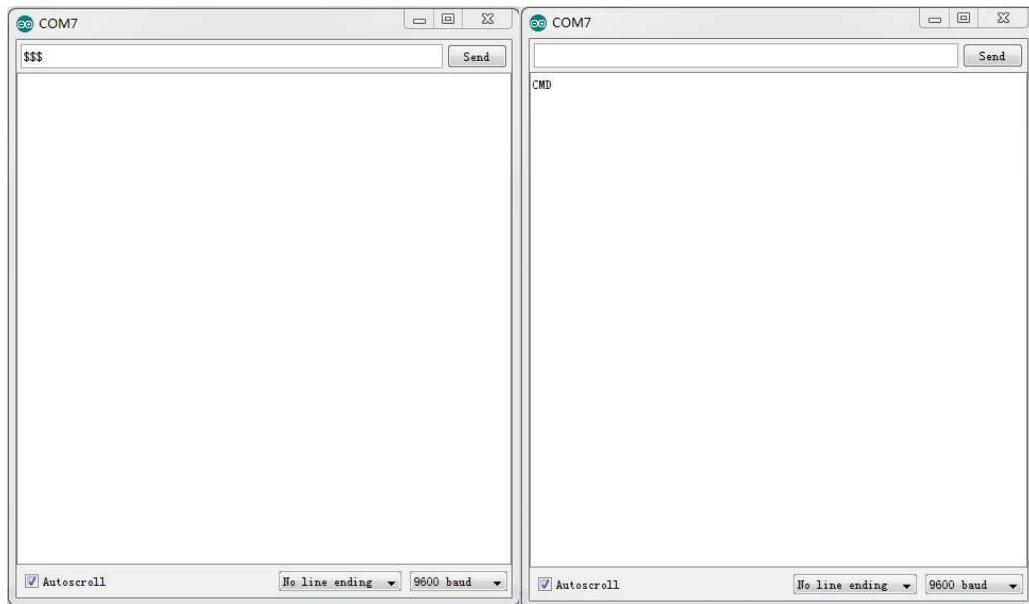
# How to use

## Configure

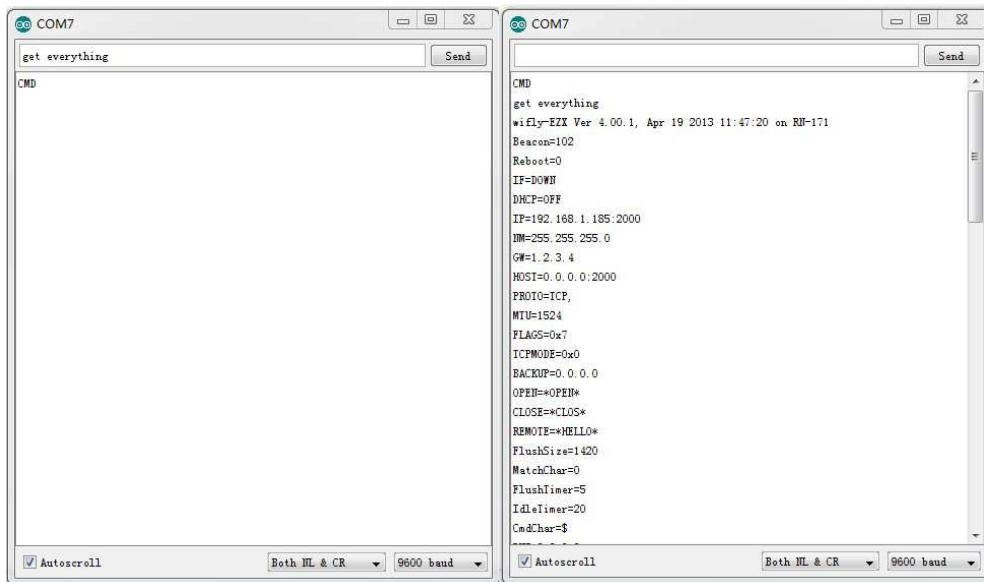
We need configure this WIFI shield before use it. There are two ways to configure it. You can configure it through arduino serial port or wirelessly.

### Configure by serial port

1. Plug this shield on your Arduino main board. Set mode switch to **PC** and **Soft Serial** position. WIFI mode switch to **Normal** position. Then connect your Arduino board to your computer USB port.
2. Uploading example sketch 'Blink' to Arduino board.
3. Set mode switch to **PC** and **UART** position. WIFI mode switch to **Normal** position
4. Open Arduino IDE and serial monitor (No line ending, 9600 ), send command **\$\$\$** to entering command mode. You will get a reply **CMD**.

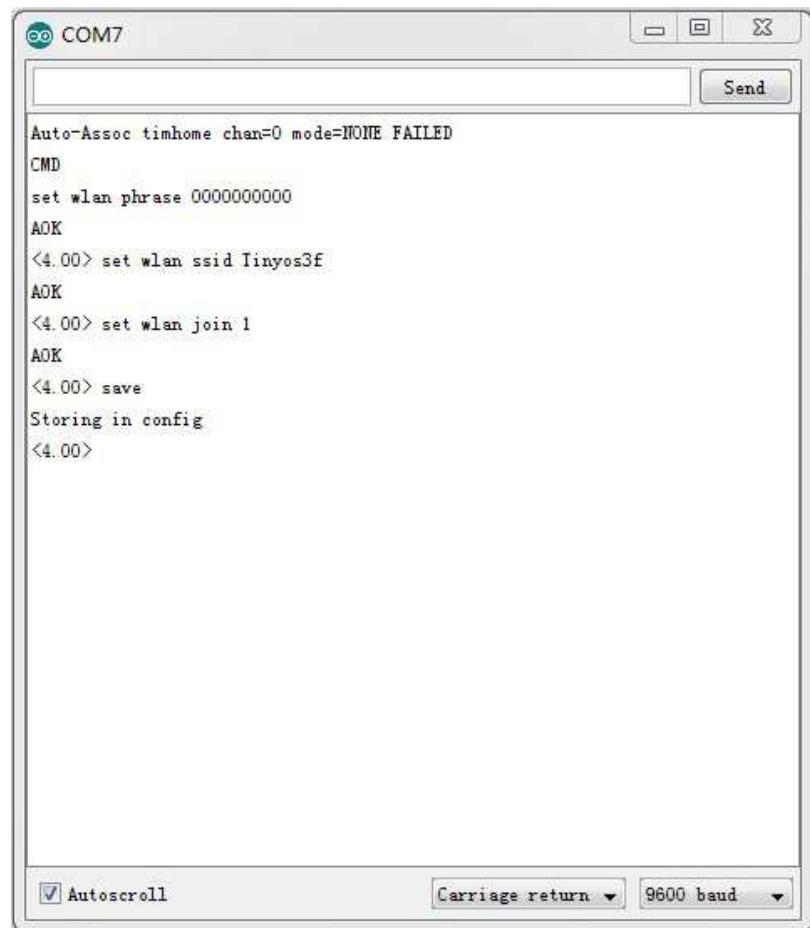


5. Change serial setting to 'Carriage return' send AT command [get everything](#) you will get all the module information.



6. Now you can setup the module's WIFI settings

```
set wlan phrase <your wpa password>
set wlan ssid <your ssid>
set wlan join 1
save
```



\*For example, our test network ssid is tinysine, wlan phrase is 0000000000.

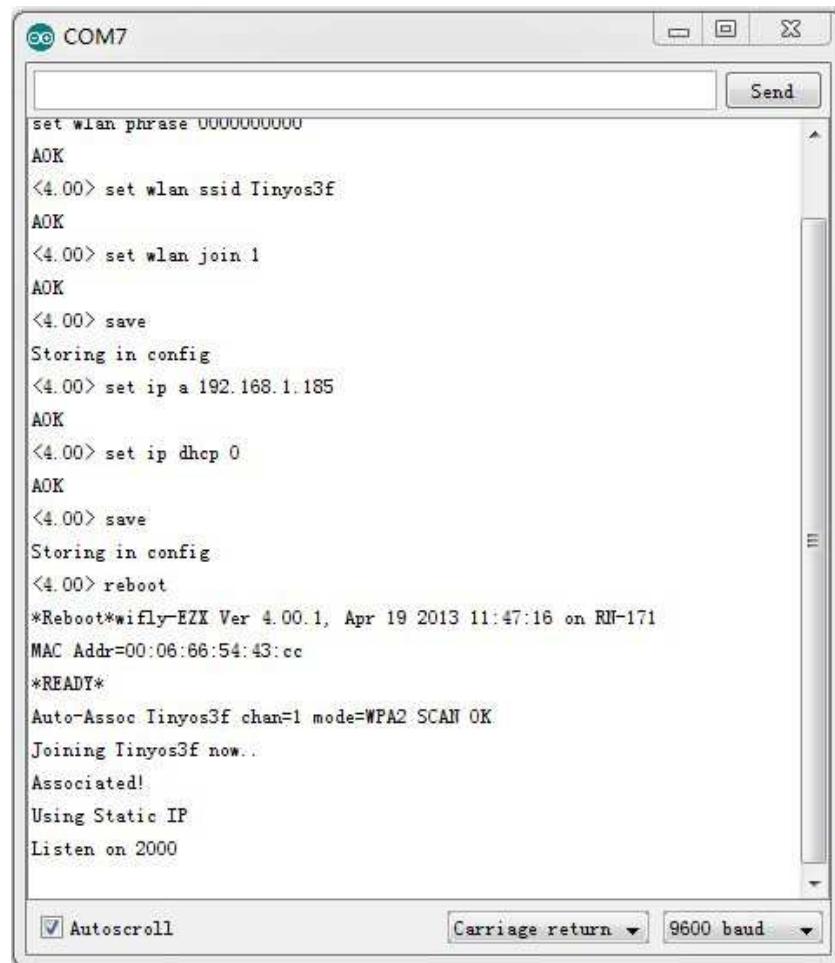
If you want a fixed IP address, here are the commands:

set ip a <your IP>

set ip dhcp 0

save

reboot



The screenshot shows a Windows-style serial communication window titled "COM7". The text area contains the following log output:

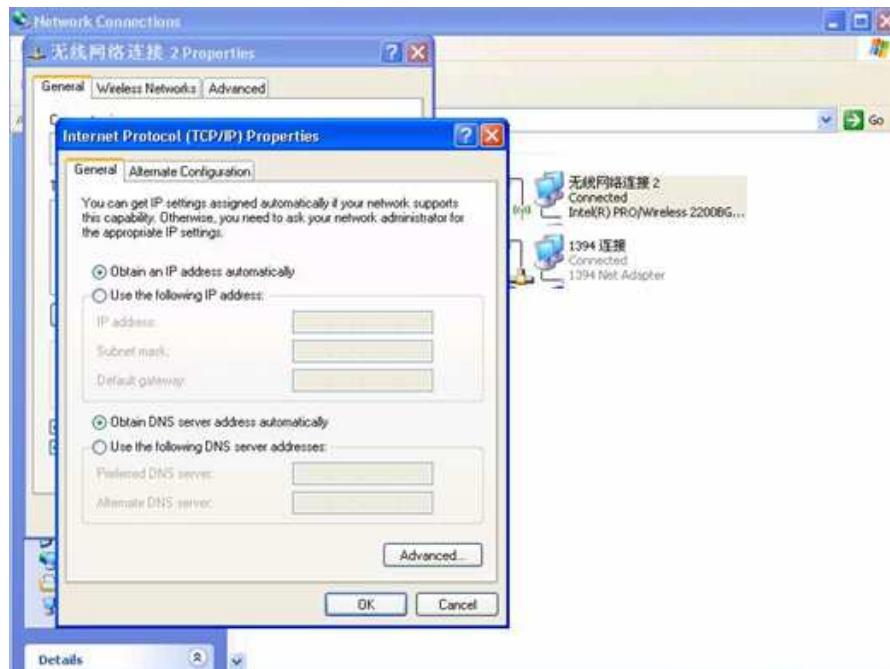
```
set wlan phrase 0000000000
AOK
<4.00> set wlan ssid Tinyos3f
AOK
<4.00> set wlan join 1
AOK
<4.00> save
Storing in config
<4.00> set ip a 192.168.1.185
AOK
<4.00> set ip dhcp 0
AOK
<4.00> save
Storing in config
<4.00> reboot
*Reboot*wifly-EZX Ver 4.00.1, Apr 19 2013 11:47:16 on RH-171
MAC Addr=00:06:66:54:43:cc
*READY*
Auto-Assoc Tinyos3f chan=1 mode=WPA2 SCAN OK
Joining Tinyos3f now..
Associated!
Using Static IP
Listen on 2000
```

At the bottom of the window, there are three buttons: "Autoscroll" (checked), "Carriage return", and "9600 baud".

Here we set our WIFI module IP address: 192.168.1.185

## Configure by WIFI

1. Plug this shield on your Arduino main board. Set WIFI mode switch S1 to **AP** position
2. Connect your Arduino board to your computer USB port. This shield will works as a AP, it create a new network. The LEDs on the board will blink in sequence.
3. Set your computer Obtain an IP address automatically.



4. From your computer, You will find a new wireless network named **Wifly-EZX-XX**



Connect to the WiFly-EZX-XX network. This is an open network that does not require a pass phrase or pass key.

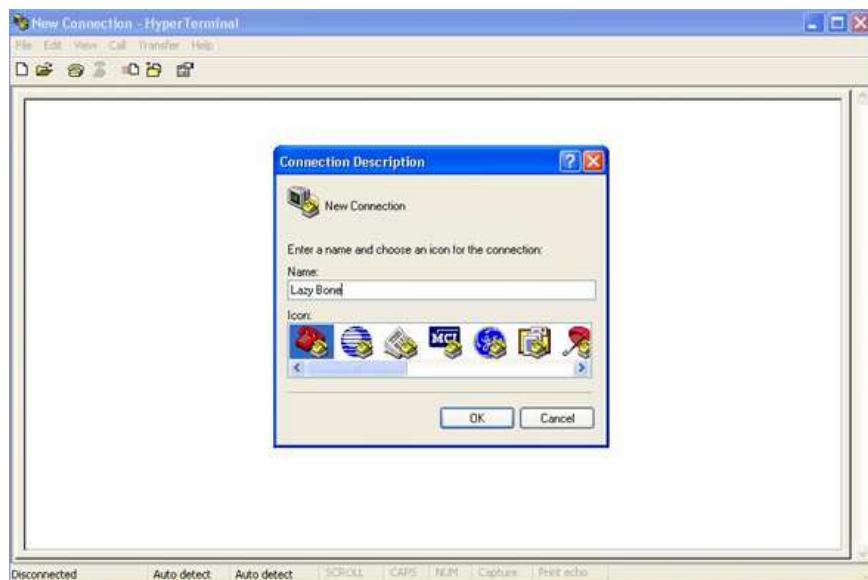
**NOTE:** It may take a couple of minutes for Auto IP in Windows to assign an IP address and connect to the network. You can check IP address of your Windows computer by running the ipconfig command in the command window. If connected, this command will show you the IP address and net mask for your computer.

The IP address assigned by Auto IP must be on the subnet 1.2.3.x subnet otherwise the

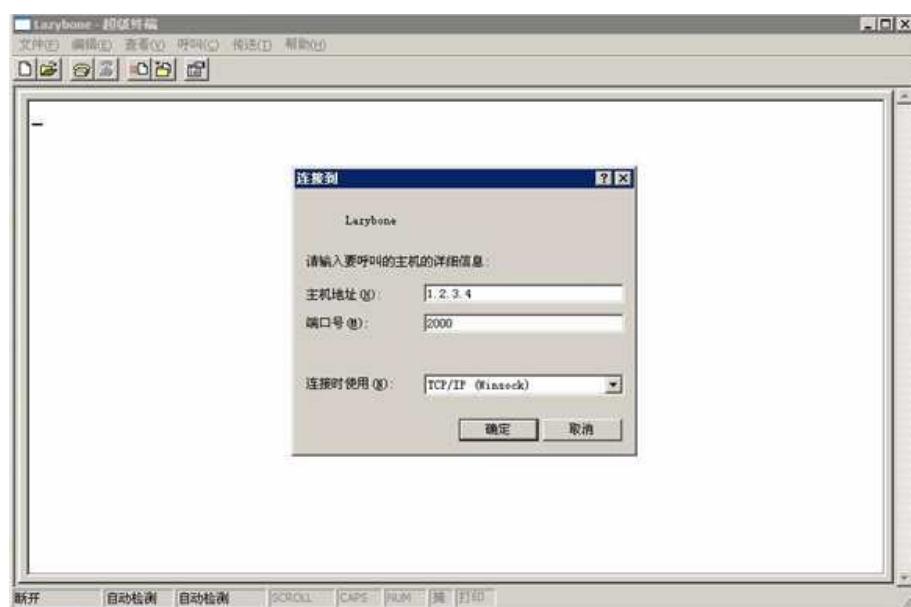
Wifly module will not be accessible.

**NOTE:** If your machine has both wireless and wired interface hardware you may need to disable the wired LAN interface hardware before connecting to the adhoc network. If the wired LAN is enabled, the computer may assign an IP address that is not on the same subnet as the Wifly module.

5. Run HyperTerminal and create a new connection.

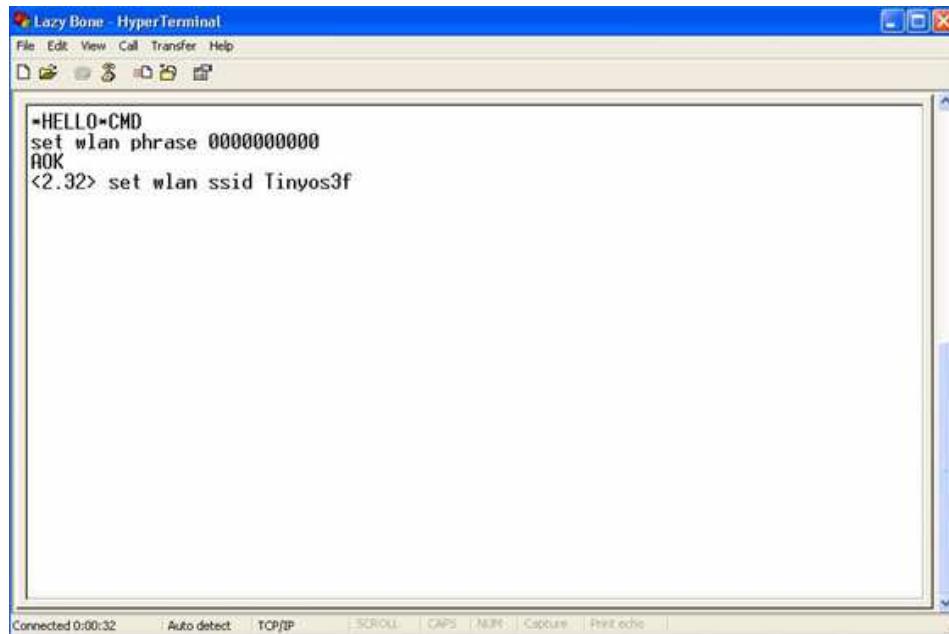


Telnet into the Wifly WiFi module on port 2000 telnet 1.2.3.4 2000



Once connected and you have a good IP address. You should see the response \*HELLO\*. Type \$\$\$ (without hitting return) to enter command mode. Now you can setup the module's wifi settings (hit return after each command):

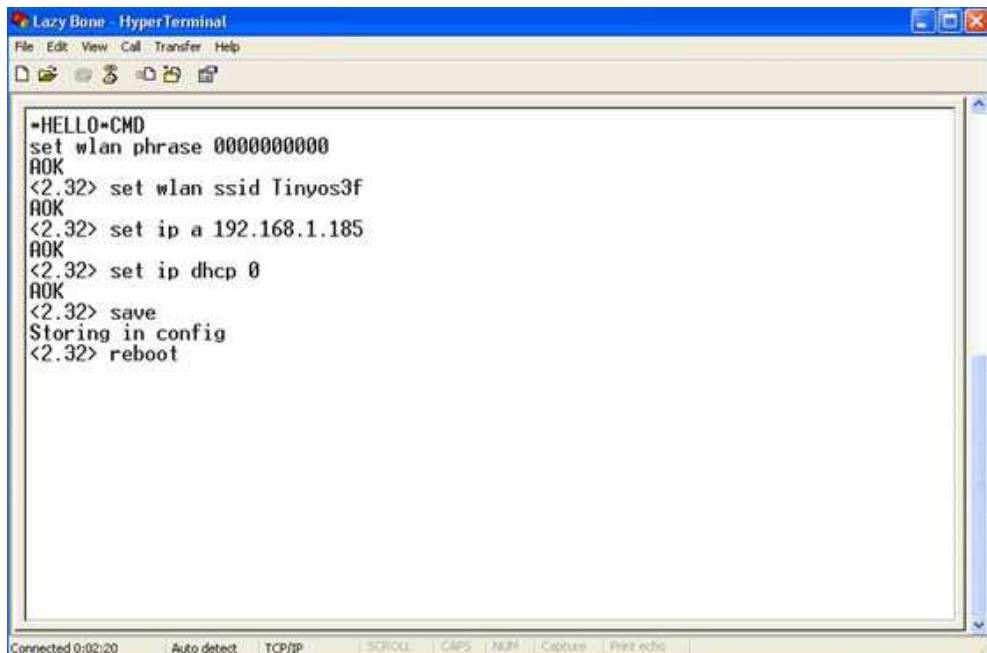
```
set wlan phrase <your wpa password>
set wlan ssid <your ssid>
set wlan join 1
save
```



\*For example, our test network ssid is Tinyos3f, wlan phrase is 0000000000.

If you want a fixed IP address, here are the commands:

```
set ip a <your IP>
set ip dhcp 0
save
reboot
```



A screenshot of the Windows HyperTerminal application window titled "Lazy Bone - HyperTerminal". The window shows a command-line interface with the following text:

```
-HELLO=CMD
set wlan phrase 0000000000
AOK
<2.32> set wlan ssid Tinyos3f
AOK
<2.32> set ip a 192.168.1.185
AOK
<2.32> set ip dhcp 0
AOK
<2.32> save
Storing in config
<2.32> reboot
```

The status bar at the bottom of the terminal window displays "Connected 0:02:20", "Auto detect", "TCP/IP", and several status indicators: SCROLL, CAPS, NUM, Capture, Print echo.

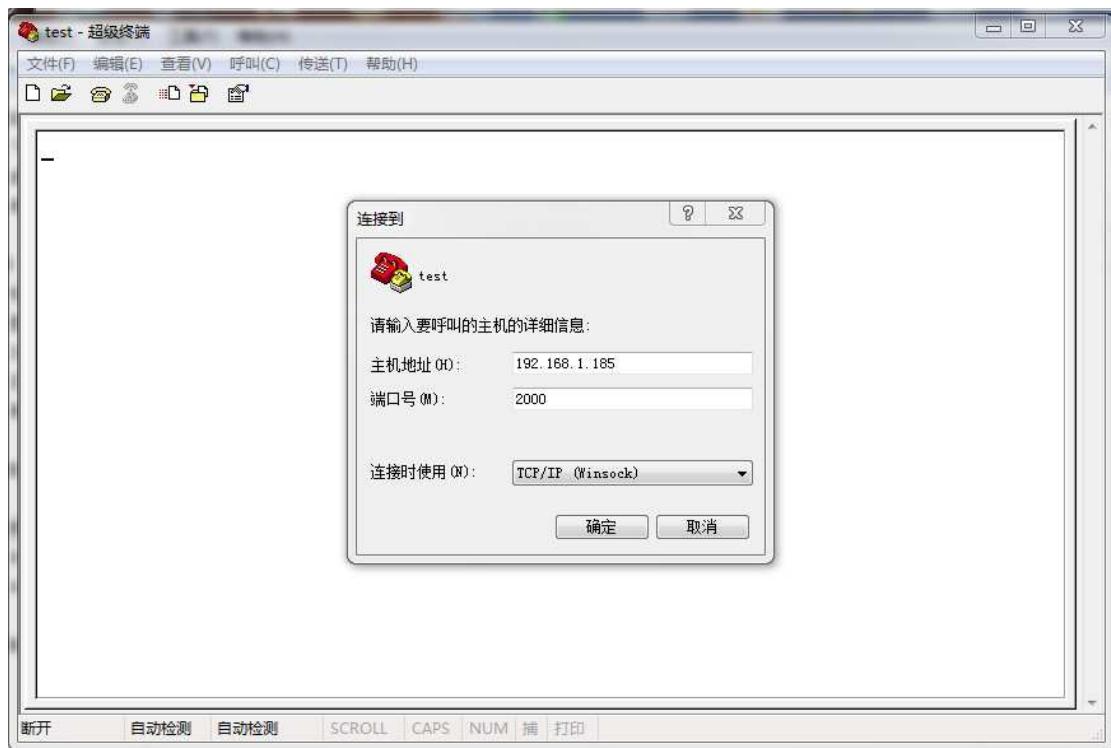
Here we set our WIFI module IP address: 192.168.1.185

More details commands please download the WiFiBee user manual [here](#)

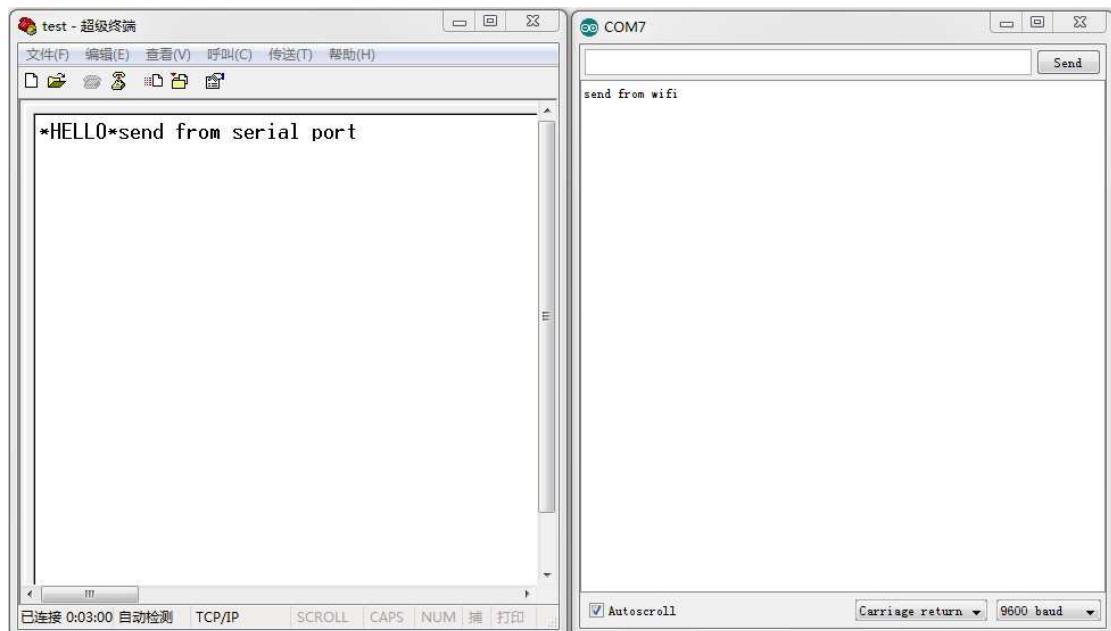
## Communicate

Disconnect the USB connection. Switch back to Normal mode and power it. It will auto connect to your local WIFI network. Green LED D1 slow blink.

Now open a terminal, You can use Putty, or if you have a Windows XP machine you can use the Hyperterminal included. and connect to the WiFi shield' s IP address. Don't forget to indicate the server port. In our example we are using port "2000".



When it connected with your WIFI shield. You should see the response \*HELLO\*. Green LED D1 state on solid. Settings to allow you to see what you are typing in Hyperterminal once connected to the WiFi shield you should be able to send data to Arduino through WiFi. Just input data into Hyperterminal input blank, and the serial monitor of arduino which connected with arduino will show the data arduino receives.



## Schematic

