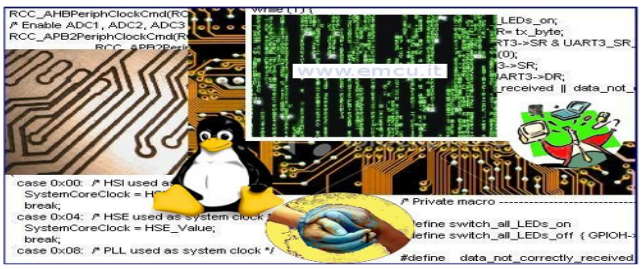


ATTENTION

To see the latest DOC, AN, Examples, etc, regarding the:
STM32, WireLess, MEMS, IoT, etc.
visit the new site:
www.emcu.eu
In the future emcu.it and emcu.eu will become a unique site.

WiFi
Quick start guide
SPWF01SA.11 and SPWF01SC.11

Home Page
WiFi page

Introductions

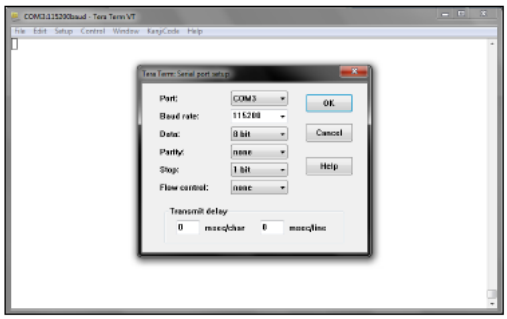
- We suppose that you already know how to use the **Silica BRANCA board** that is an eva board for the STM WiFi module named: **SPWF01Sxxx**
  If for you is the first time that using the Silica BRANCA board, please read **here** and also the

Hands on Session that is **here**.

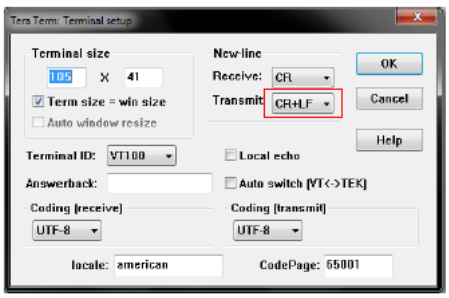
- If you use a corporate PC, for use **SOCKET** in conjunction with **server.exe** (Socket Server), you must **disable the FIREWALL**.
  On some corporate PC this is not possible, in this case I suggest to use your home PC for do the Socket test.

- If you connect the **Branca-Board** to the **PC**, we suggest to use Tera Term and the configuration is shown below.

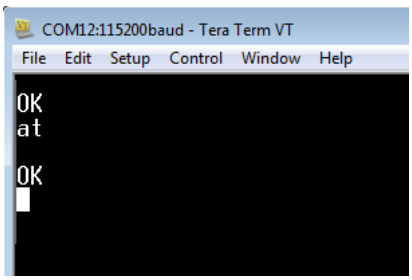
- Open Tera Term
  - Run Tera Term (open Tera term folder and then run **ttermpro.exe**)
  - Open the assigned COM port
  - Tera Term: Setup → Serial port
    - Baud rate: 115200
    - Data: 8 bit
    - Parity: none
    - Stop: 1 bit
    - Flow control: none



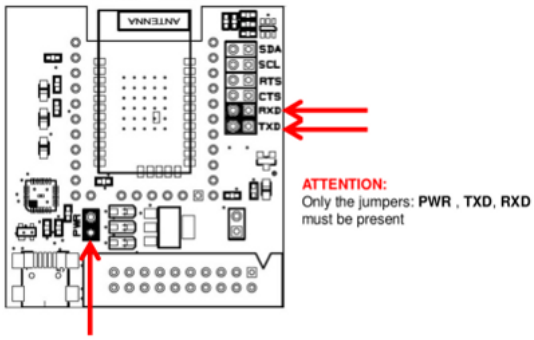
- Set CR+LF in the Terminal setup (to properly paste text in the terminal):
  - Tera Term: Setup → Terminal → Transmit: CR+LF



For test the connection between PC/TeraTerm and the Branca-Board type, in Tera Term, the command: **AT**
You must see the response: **OK**
See below.



- See below, the jumpers on the Silica Branca Board.



UP

Reset the WiFi module

AT&F

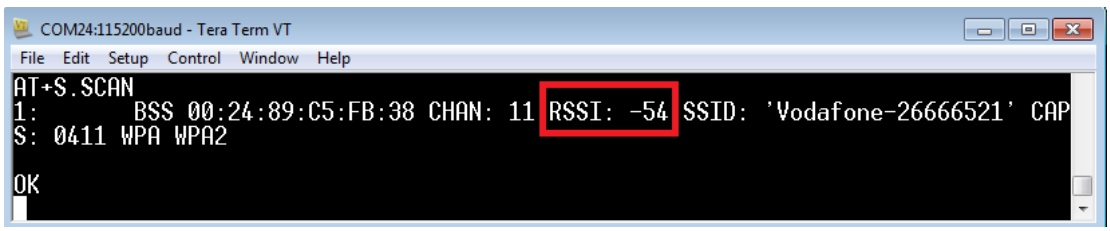
restores the factory default values of the configuration variables and writes them to non-volatile storage.

UP

Radio signal strength received

AT+S.SCAN

See below the red box.



UP

SOCKET

Open a network socket

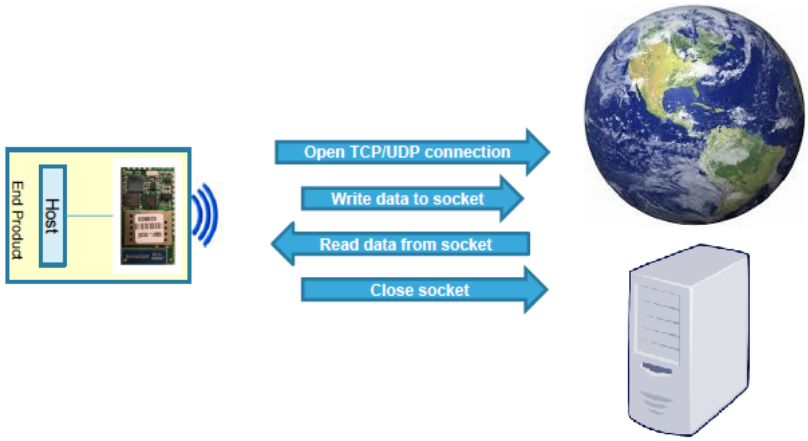
AT+S.SOCKON = ID, PORT, PROTOCOL, ind

ID == Host Name - IP or Internet name (www.google.com)
PORT == The IP of the Computer or Host that you need connect (TCP/UDP)
PROTOCOL == **t** for TCP socket, **u** for UDP socket
ind == indicate when data has arrived (optional)

Up to 8 socket connections at same time.

Attention: the WiFi module at the moment (January 2014) **is CLIENT** this means that only the WiFi module have the possibility to open a socket for connect a remote SERVER (PC, HOST, www...).
In other words, is not possible for a PC or a HOST open a socket to connect the WiFi module.

The possibility of connection are shown below.



Below there is an example that open a socket to send a string of data to the PC.

For PC with WINDOWS

Open the TCP socket server (**disable the firewall to properly run it**).
Run the program:
server.exe

Server.exe is a: TCP server listens for incoming connections on the port 32000, it sends back all data received.
Please read also this **note**.

For PC with LINUX

Use my PYTHON server, see below:

```
#!/user/bin/python

import socket, select

data = 'a'

#Function to broadcast chat messages to all connected clients
def broadcast_data (sock, message):
    #Do not send the message to master socket and the client who has send us the message
    for socket in CONNECTION_LIST:
        if socket != server_socket and socket != sock :
            try :
                socket.send(message)
            except :
                # broken socket connection may be, chat client pressed ctrl+c for example
                socket.close()
                CONNECTION_LIST.remove(socket)

if __name__ == "__main__":

    # List to keep track of socket descriptors
    CONNECTION_LIST = []
    RECV_BUFFER = 4096 # Advisable to keep it as an exponent of 2
    PORT = 32000

    server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    # this has no effect, why ?
    server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    server_socket.bind(("0.0.0.0", PORT))
    server_socket.listen(10)

    # Add server socket to the list of readable connections
    CONNECTION_LIST.append(server_socket)

    print " "
```

```
        print "************************************ by www.emcu.it"
        print "Chat server started on port " + str(PORT)
        print "For end the connection and end this program send me: @"
        print "For end this program type: ctrl + c"
        print " "

        while data != '@':
            # Get the list sockets which are ready to be read through select
            read_sockets,write_sockets,error_sockets = select.select(CONNECTION_LIST,[],[])

            for sock in read_sockets:
                #New connection
                if sock == server_socket:
                    # Handle the case in which there is a new connection recieved through
server_socket
                    sockfd, addr = server_socket.accept()
                    CONNECTION_LIST.append(sockfd)
                    print "Client (%s, %s) connected" % addr

                    broadcast_data(sockfd, "[%s:%s] entered room\n" % addr)

                #Some incoming message from a client
                else:
                    # Data recieved from client, process it
                    try:
                        #In Windows, sometimes when a TCP program closes abruptly,
                        # a "Connection reset by peer" exception will be thrown
                        data = sock.recv(RECV_BUFFER)
                print "Received: " + data
                if data == '@':
                            sock.close()
                            #CONNECTION_LIST.remove(sock)
                        if data:
                            broadcast_data(sock, "\r" + '<' + str(sock.getpeername()) + '> ' +
data)

                    except:
                        broadcast_data(sock, "Client (%s, %s) is offline" % addr)
                        print "Client (%s, %s) is offline" % addr
                        sock.close()
                        CONNECTION_LIST.remove(sock)
                print " "
                print " *** END program ***"
                print " "
                        continue

        server_socket.close()
```

This Python server is tested on PC with Xubuntu and on RaspBerry Pi with RaspBian (Linux)

Open TeraTerm and follow the step below.

NOTE:

- **BOLD** == command that you must write on TeraTerm
- **BLUE** == answer from the WiFi module
- *ITALIC* == character that must be type on Teraterm but that you don't see on TeraTerm but you must see on the server.exe window.

AT+S.SOCKON=192.168.1.6,32000,t,ind<*CR*>

ID: 00

OK

AT+S.SOCKW=00,13<*CR*>

Hello World<CR>

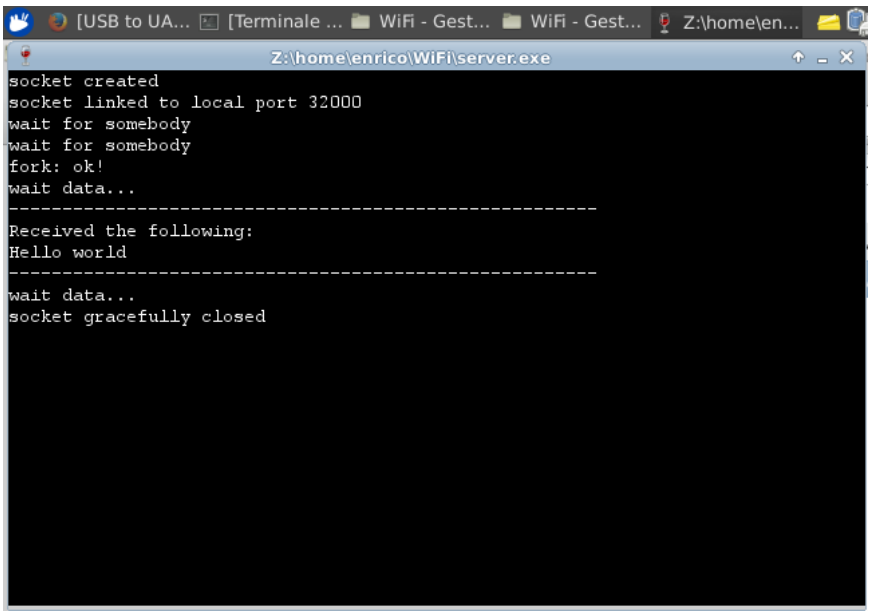
OK

+WIND:55:Pending Data:0:13

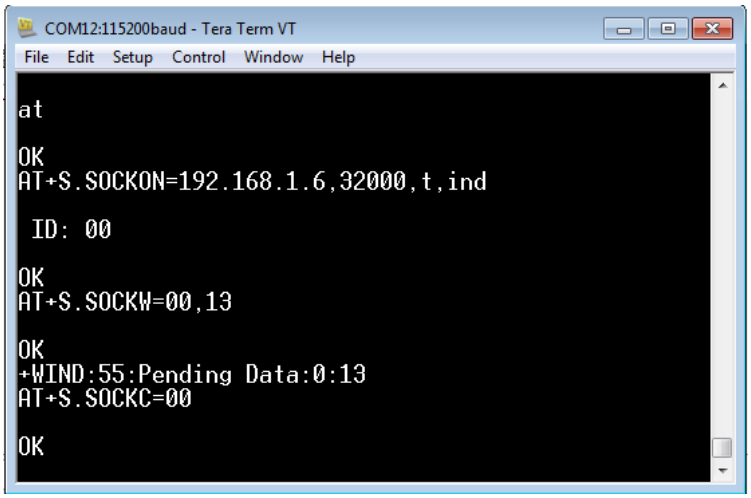
AT+S.SOCKC=00<*CR*>

The above command Close the socket 00

Below is what you must see in **server.exe** screen.



Below is what you must see in **TeraTerm**



NOTE:

server.exe and **cygwin1.dll** must be in the same directory.
Server.exe is tested on **Windows7** and on **Xubuntu using Vine**.
Click **here** for download server.exe and cygwin1.dll

UP

Mini AP

MINI AP without password

AT+S.SCFG=wifi_mode,3
AT+S.SCFG=wifi_priv_mode,0
at+s.ssidtxt=**EM**
at&w
at+cfun=1

NOTE:
EM is the name of the mini AP.

MINI AP with password

at+s.scfg=wifi_mode,3
at+s.ssidtxt=**EM**

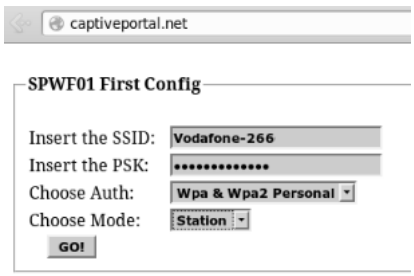
at+s.scfg=wifi_priv_mode,1
AT+S.SCFG=wifi_wep_keys[0],**7465737431**

AT+S.SCFG=wifi_wep_key_lens,05
AT+S.SCFG=wifi_wep_default_key,0

at&w
at+cfun=1

NOTE:
EM is the name of the mini AP.
7465737431 is the password that is equivalent to **test1**

Also is available a web page, for the configuration of your WiFi module.
http://captiveportal.net



UP

Some connection examples to an AP

ANDROID - the configuration si:

HotSpot
WPA2 PSK

Below the commands to use on WiFi module.

at+s.ssidtxt=AndroidEM
at+s.scfg=wifi_wpa_psk_text,cb41c492e2a9
at+s.scfg=wifi_priv_mode,2
at+s.scfg=wifi_mode,1
at+s.scfg=ip_use_dhcp,1
at&w
at+cfun=1

ROUTER VodafonStation 2

```
at+s.ssidtxt=Vodafone-26666521
at+s.scfg=wifi_wpa_psk_text,enrico
at+s.scfg=wifi_priv_mode,2
at+s.scfg=wifi_mode,1
at+s.scfg=ip_use_dhcp,1
at&w
at+cfun=1
```

ROUTER-Belkin N150 and **BlackBerry (HotSpot)** used during STDay 2013 in Italy

```
at+s.ssidtxt=BBMHem
at+s.scfg=wifi_wpa_psk_text,enrico321
at+s.scfg=wifi_priv_mode,2
at+s.scfg=wifi_mode,1
at+s.scfg=ip_use_dhcp,1
at&w
at+cfun=1
```

ROUTER-FRITZ Box 7390

```
at+s.ssidtxt=EMCU7390
at+s.scfg=wifi_wpa_psk_text,enrico
at+s.scfg=wifi_priv_mode,2
at+s.scfg=wifi_mode,1
at+s.scfg=ip_use_dhcp,1
at&w
at+cfun=1
```

UP

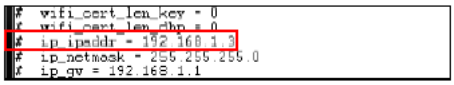
Mode Point to Point / AD-HOC or IBSS network

Associate the iOS device with the ADHOC network created using the WiFi module


```
AT+S.SSIDTXT=emADHOC
AT+S.SCFG=wifi_priv_mode,0
AT+S.SCFG=wifi_mode,2
AT+S.SCFG=ip_ipaddr,192.161.0.3
AT+S.SCFG=ip_gw,192.161.0.255
AT+S.SCFG=ip_dns,192.161.0.255
AT+S.SCFG=ip_netmask,255.255.255.0
AT+S.SCFG=ip_use_dhcp,0
AT&W
AT+CFUN=1
```

After the restart of the WiFi module, find your IP address, type:

AT+S.STS



Open the SAFARI web browser (on iPAD) and insert your IP address + /index.html.

192.168.1.3/index.html

UP

General purpose commands

*** Reset - restores the factory default values of the configuration variables and writes them to non-volatile storage.
AT&F

*** Lists the types, sizes, and names of all the existing files present on the WiFi module.
AT+s.fsl

*** Deletes an existing file by name. Static files may not be deleted, only overridden.
at+s.fsd=file_name

*** Displays the name and value of all configuration variables in the WiFi module
AT&V

*** Displays the current values of all the status variables.
AT+S.STS

*** enable/disable Wi-Fi device
0 disable
1 enable

AT+S.WIFI=0
AT+S.WIFI=1

*** trigger Wi-Fi reassociation sequence
AT+S.ROAM

Example:

```
AT+S.ROAM
OK
+WIND:41:WiFi Disassociation
+WIND:21:WiFi Scanning
+WIND:35:WiFi Scan Complete (0x0)
+WIND:39:FOUND: 80:60:07:56:17:48 FREQ: 2437 RSSI: -45 SSID: 'BBMHem' CAPS: 0431
WPA: 0 WPA2: 20
+WIND:19:WiFi Join: 80:60:07:56:17:48
+WIND:25:WiFi Association with 'BBMHem' successful
```

```
+WIND:51:WPA Handshake Complete
+WIND:24:WiFi Up: 10.0.0.100
```

*** Set the IP addres
AT+S.SCFG=ip_ipaddr,192.168.178.36

*** Return the IP addres
AT+S.GCFG=ip_ipaddr

*** ReStart the WiFi
AT+CFUN=1

For a complete list of command see the User Manual of the WiFi module, see link here (**SPWF01SA.11** and **SPWF01SC.11**).

UP
