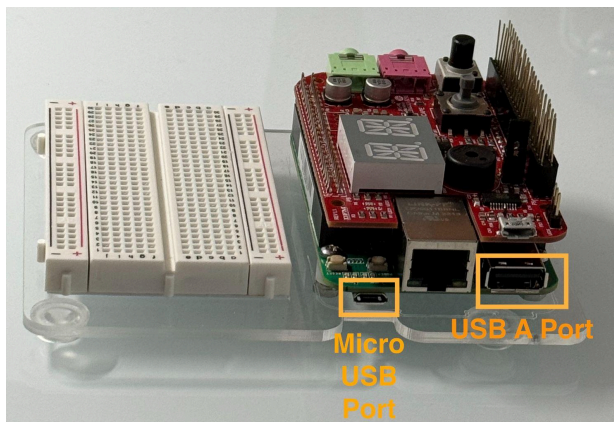# BeagleBone WiFi Guide

Hardware Installation:

1) Before plugging in your BeagleBone Green, identify the location of the USB A port and insert the WiFi adapter into it.

2) Connect your BeagleBone Green to your computer using the Micro USB port.

This should supply power to your BeagleBone and establish an 'Ethernet-over-USB' connection between your BeagleBone and your Debian virtual machine.

**'Ethernet-over-USB' must first be configured on your BeagleBone using Professor Brian Fraser's Networking Guide.

Software Configuration:

1) Connect to your BeagleBone using SSH using the following command:
    ssh debian@192.168.7.2

2) Once connected to your BeagleBone, you must edit two files using the following commands:
    (File 1) sudo nano **/etc/wpa_supplicant/wpa_supplicant.conf**

    **In this file you must write your WiFi credentials as follows:
        ctrl_interface=/var/run/wpa_supplicant

```
        ctrl_interface_group=0
        update_config=1
        network={
            ssid="wifi_name"
            psk="wifi_password"
        }
```
(File 2) sudo nano **/etc/network/interfaces**

**In this file you specify that you wish to use the WiFi interface 'wlan0' as follows:
auto wlan0
iface wlan0 inet dhcp
wpa-driver nl80211
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

3) Reboot your BeagleBone by issuing the command "**sudo reboot**". Once your BeagleBone restarts, it should connect to your specified WiFi automatically.

Verifying the connection:

WiFi Indicator Light

The light on the Wifi adapter should light up on successful connection.

ip a

This command should show a new interface with an ip address assigned.

```
2: wlp0s20f3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
   link/ether 04:6c:59:0f:81:1f brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.19/24 brd 192.168.1.255 scope global noprefixroute wlp0s20f3
      valid_lft forever preferred_lft forever
   inet6 2001:569:76f6:8d00:65ac:2c63:b120:9ae/64 scope global temporary dynamic
      valid_lft 14695sec preferred_lft 14395sec
   inet6 2001:569:76f6:8d00:8382:6a12:38e6:d687/64 scope global dynamic mngtmpaddr noprefixroute
      valid_lft 14695sec preferred_lft 14395sec
   inet6 fe80::554e:a8f8:9bca:9cb0/64 scope link noprefixroute
      valid_lft forever preferred_lft forever
```

ping 8.8.8.8

Try running the ping command without enabling the internet over the Ethernet-Over-USB script. This should be successful if the wifi router is able to provide connection to the internet.

```
[~] ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=60 time=10.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=60 time=26.1 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=60 time=9.63 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=60 time=10.1 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 9.631/13.987/26.081/6.985 ms
```

Troubleshooting:

1) After restarting, if your BeagleBone doesn't automatically connect to your specified WiFi do the following in order:

      i) Ensure that both the files edited in step 2 are free of any grammatical mistakes.

      ii) In the "**/etc/wpa_supplicant/**" directory, ensure that there is only one "wpa_supplicant.conf" file.

      iii) Kill the currently running "wpa_supplicant" process using the following commands:

            //Get the PID

            ps aux | grep wpa_supplicant

            //Kill the process

            kill -9 <PID>

      iv) Start a new "wpa_supplicant" process that uses the updated configuration files:

            sudo wpa_supplicant -B -c /etc/wpa_supplicant/wpa_supplicant.conf -i wlan0

```
debian@ysonik-beagle:~$ sudo wpa_supplicant -B -c /etc/wpa_supplicant/wpa_supplicant.conf -i wlan0
[sudo] password for debian:
Successfully initialized wpa_supplicant
```

            **If the new process was started successfully, you should see the above message

            .

Communication Using UDP Sockets:

Now that your BeagleBone is connected to your desired WiFi, you no longer need to use "Ethernet-over-USB". The BeagleBone simply needs to be connected to a power source to allow an SSH connection.

The following is a simple UDP server implementation in C that can be executed on your BeagleBone. When the server is running, you can use the "netcat" command line tool to send UDP messages to the server from any other device connected to the same WiFi. The server will then print the message to stdout.

```c
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

#define MAX_LEN 1024
#define SERVER_PORT 3000

int localSocket;
char receivedMessage[MAX_LEN];
struct sockaddr_in serverAddress;

void create_socket()
{
    //Create the UDP socket.
    localSocket = socket(AF_INET, SOCK_DGRAM, 0);
    if(localSocket < 0)
    {
        printf("Error: Could not create the local socket.\n");
        exit(1);
    }

    //Initialize the server address struct.
    memset(&serverAddress, '\0', sizeof(serverAddress));
    serverAddress.sin_family = AF_INET;
    serverAddress.sin_addr.s_addr = INADDR_ANY;
    serverAddress.sin_port = htons(SERVER_PORT);

    //Bind the local socket to the server address struct.
```

```
    int bind_result = bind(localSocket, (struct sockaddr*)&serverAddress,
sizeof(serverAddress));
    if(bind_result < 0)
    {
        printf("Error: Could not bind the local socket.\n");
        exit(2);
    }
    return;
}

int main()
{
    //(1) Create the UDP socket.
    create_socket();

    //(2) Listen for UDP messages using the local socket.
    while(true)
    {
        memset(receivedMessage, '\0', MAX_LEN);
        recvfrom(localSocket, receivedMessage, (MAX_LEN-1), MSG_WAITALL, NULL,
NULL);
        printf("Received: %s\n",receivedMessage);
    }

    return 0;
}
```

Testing the UDP server using netcat

Use the command below to test the server over UDP socket.

```
netcat -u <BeagleBone-IP-Address> <Port>
```

Type any text after executing the above command and press enter to send it to the server.

```
[~] netcat -u 127.0.0.1 3000        [/tmp] ./a.out
Testing                              Received: Testing
Here
                                     Received: Here
```

References:

https://deviceinbox.com/images/post/D-Link_DWA-127.png
https://beej.us/guide/bgnet/html/