**-------------FIRST TEST ------------------------------------------**

Wifi\_connect.lua

--- Connect to the wifi network ---

print("Connecting to WiFi access point...")

-- ------------------------------------------------

-- --this is where the captive portal

-- -------------------------------------------

--set table for wifi list------------

wifiTable = {}

-- table.setn(wifiTable, 5)

---set wifi mode

wifi.sta.disconnect()

wifi.setmode(wifi.STATIONAP)

-- -- -- --ESP SSID generated wiht its chipid

-- -- wifi.ap.config({ssid="Netop\_board-"..node.chipid(), auth=wifi.OPEN})

-- -- enduser\_setup.manual(true)

----start wifi manager-------------------

enduser\_setup.start(

function()

print("Connected" )

cfg={}

----Get the ssid and passord of AP connected to

ssid, password, bssid\_set, bssid = wifi.sta.getdefaultconfig()

print("\nCurrent Station configuration:\nSSID : "..ssid

.."\nPassword : "..password)

cfg.ssid=ssid

cfg.pwd=password

print(cfg)

table.insert(wifiTable,cfg)

-- print(wifiTable)

--------------------choose available network--------

counter = 0

wifi.sta.config(wifiTable[1][ssid], wifiTable[1][pwd] )

tmr.alarm(1, 1000, tmr.ALARM\_SEMI, function()

counter = counter + 1

if counter < 60 then

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. wifiTable[1][ssid])

tmr.start(1) -- restart

else

print("Connected to" .. wifiTable[1][ssid])

end

elseif counter < 120 then

wifi.sta.config(wifiTable[2][ssid], wifiTable[2][pwd])

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. wifiTable[2][ssid])

tmr.start(1) -- restart

else

print("Connected to ".. wifiTable[2][ssid])

end

elseif counter < 180 then

wifi.sta.config(wifiTable[3][ssid], wifiTable[3][pwd])

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. wifiTable[3][ssid])

tmr.start(1) -- restart

else

print("Connected to ".. wifiTable[3][ssid])

end

else

print("Out of options, giving up.")

end

end)

end,

function(err, str)

print("enduser\_setup: Err #" .. err .. ": " .. str)

end

);

---------------------------------------------

-- ---the end of the captive portal

-- -------------------------------------------

tmr.create():alarm(2000, tmr.ALARM\_AUTO, function(cb\_timer)

if wifi.sta.getip() == nil then

print("Waiting for IP address...")

else

cb\_timer:unregister()

print("WiFi connection established " )

tmr.create():alarm(1, tmr.ALARM\_SINGLE, function()

local clock = require("ntp-clock")

clock.sync()

end)

end

end)

tmr.create():alarm(10000, tmr.ALARM\_AUTO, function(cb\_timer)

if mykey == nil and wifi.sta.getip() ~= nil and rtctime.get() > 1000000 then

print("Acquiring auth keys.")

local cred = require("client\_credentials")

cred.acquire()

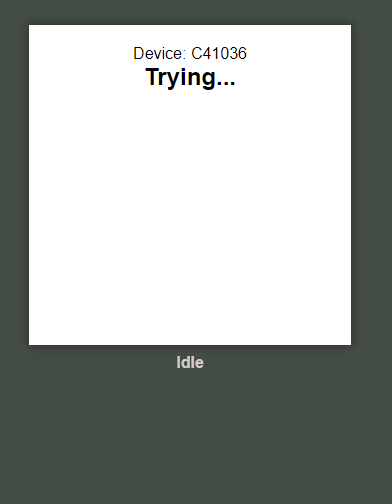
elseif mykey ~= nil then

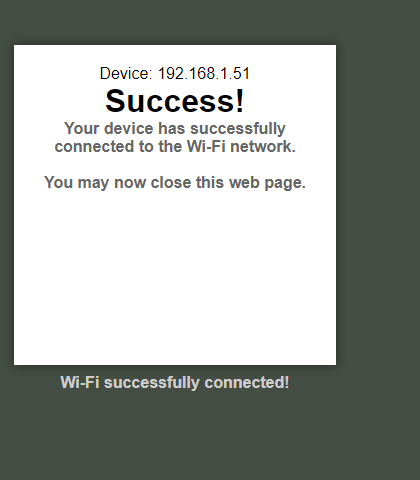
cb\_timer:unregister()

end

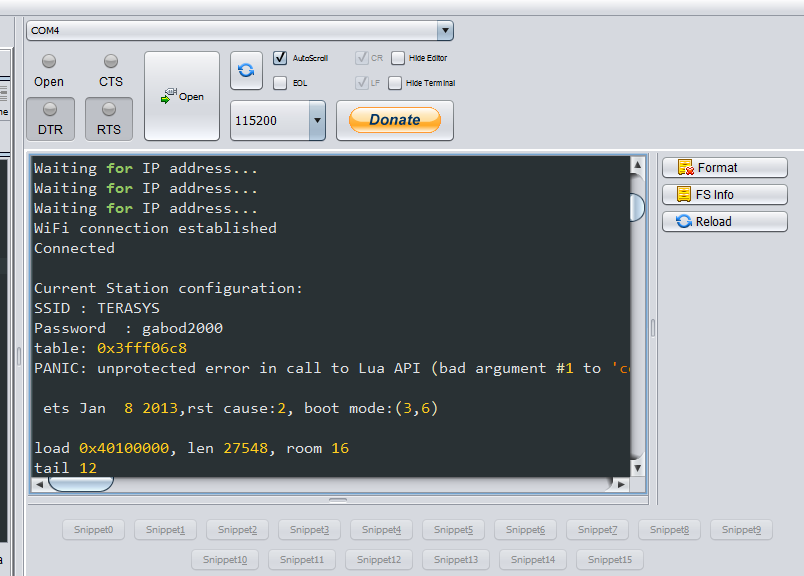
end)

* AFTER COMPILING THE CODE, I INPUTED THE WIFI CREDENTIAL IN THE WIFI MANAGER. HERE IS THE SCREENSHOT OF THE WIFI MANAGER AFTER THAT.





* BELOW IS THE SCREENSHOT OF THE ESPLORER ENVIRONMENT



* Error

PANIC: unprotected error in call to Lua API (bad argument #1 to 'config' (config table not found!))

ets Jan 8 2013,rst cause:2, boot mode:(3,6)

load 0x40100000, len 27548, room 16

tail 12

chksum 0xdb

ho 0 tail 12 room 4

load 0x3ffe8000, len 2408, room 12

tail 12

chksum 0xcb

ho 0 tail 12 room 4

load 0x3ffe8968, len 136, room 12

tail 12

chksum 0x43

csum 0x43===========================================================

**----SECOND TEST ---------**

Wifi\_config.lua

--- Connect to the wifi network ---

print("Connecting to WiFi access point...")

---set wifi mode

wifi.sta.disconnect()

wifi.setmode(wifi.STATIONAP)

----start wifi manager-------------------

function startPortal( )

enduser\_setup.start(

function()

print("Connected to wifi as:" .. wifi.sta.getip())

end,

function(err, str)

print("enduser\_setup: Err #" .. err .. ": " .. str)

end,

print -- Lua print function can serve as the debug callback

);

local ip = wifi.sta.getip()

if ip ~= nil then

----Get the ssid and passord of AP connected to

ssid, password, bssid\_set, bssid = wifi.sta.getdefaultconfig()

print("\nCurrent Station configuration:\nSSID : "..ssid

.."\nPassword : "..password)

end

if table.getn(wifi1) == nil then

wifi1.ssid=ssid

wifi1.pwd=password

elseif table.getn(wifi2) == nil then

wifi2.ssid=ssid

wifi2.pwd=password

else

terasys = nil

terasys.ssid=ssid

terasys.pwd=password

end

end

--------------------------------------End wifi manager------------------

--------------------choose available network--------

counter = 0

wifi.sta.config(terasys.ssid, terasys.pwd )

tmr.alarm(1, 1000, tmr.ALARM\_SEMI, function()

counter = counter + 1

if counter < 20 then

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. terasys.ssid)

tmr.start(1) -- restart

else

print("Connected to" .. terasys.ssid )

end

elseif counter < 40 then

wifi.sta.config(dapo.ssid, dapo.pwd)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. dapo.ssid)

tmr.start(1) -- restart

else

print("Connected to ".. dapo.ssid)

end

elseif counter < 60 then

wifi.sta.config(deji.ssid, deji.pwd)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. deji.ssid)

tmr.start(1) -- restart

else

print("Connected to ".. deji.ssid)

end

elseif counter < 80 then

if table.getn(wifi1) ~= nil then

wifi.sta.config(wifi1.ssid, wifi1.pwd)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. wifi1.ssid)

tmr.start(1) -- restart

else

print("Connected to ".. wifi1.ssid)

end

end

elseif counter < 100 then

if table.getn(wifi2) ~= nil then

wifi.sta.config(wifi2.ssid, wifi2.pwd)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. wifi2.ssid)

tmr.start(1) -- restart

else

print("Connected to ".. wifi2.ssid)

end

end

else

print("Out of options, giving up. Please input WiFi credentials in the WiFi Manager")

startPortal()

end

end)

--------------------------------------------

-- ---the Wifi switching ends here

-- -------------------------------------------

tmr.create():alarm(2000, tmr.ALARM\_AUTO, function(cb\_timer)

if wifi.sta.getip() == nil then

print("Waiting for IP address...")

else

cb\_timer:unregister()

print("WiFi connection established " )

tmr.create():alarm(1, tmr.ALARM\_SINGLE, function()

local clock = require("ntp-clock")

clock.sync()

end)

end

end)

tmr.create():alarm(10000, tmr.ALARM\_AUTO, function(cb\_timer)

if mykey == nil and wifi.sta.getip() ~= nil and rtctime.get() > 1000000 then

print("Acquiring auth keys.")

local cred = require("client\_credentials")

cred.acquire()

elseif mykey ~= nil then

cb\_timer:unregister()

end

end)

config.lua

--- SERVER AND API DEFINITIONS ---

server = "www.terasyshub.io"

port = 443

url\_t = "/api/v1/data"

url\_h = "/api/v1/data"

url\_cred = "/api/v1/keys"

ntpserver = "pool.ntp.org"

--- LOCATION ---

lat = 6.497492

lon = 3.382360

-- --- WIFI CONFIGURATION ---

--set table for wifi list------------

wifi1 = {}

wifi2= {}

-- table.setn(wifiTable, 5)

terasys = {}

terasys.ssid = Terasys

terasys.pwd = gabod2000

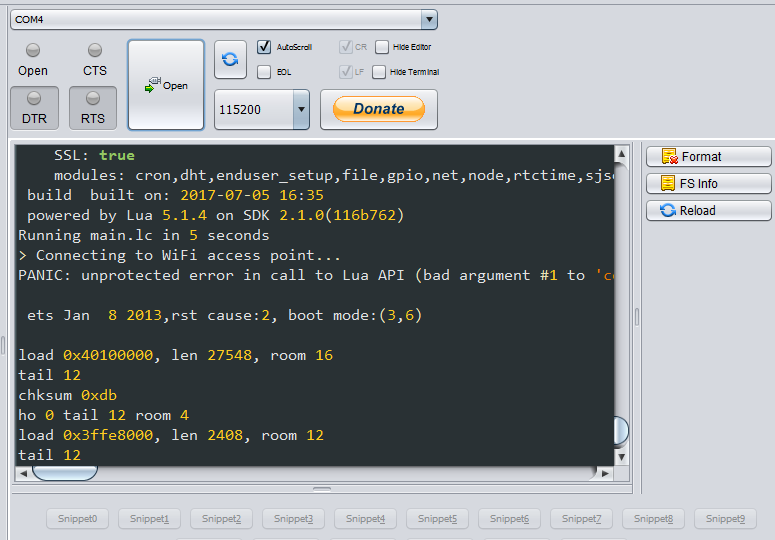
dapo= {ssid = "pama", pwd= "adedapoaderemi"}

deji = {ssid= "deji", pwd= "password"}

--- DEVICE KEY ---

mykey = "98f4f6409ae547223e4c"

* BELOW IS THE SCREENSHOT AFTER COMILING THE CODE



* Error

PANIC: unprotected error in call to Lua API (bad argument #1 to 'config' (config table not found!))

ets Jan 8 2013,rst cause:2, boot mode:(3,6)

load 0x40100000, len 27548, room 16

tail 12

chksum 0xdb

ho 0 tail 12 room 4

load 0x3ffe8000, len 2408, room 12

tail 12

chksum 0xcb

ho 0 tail 12 room 4

load 0x3ffe8968, len 136, room 12

tail 12

chksum 0x43

csum 0x43===========================================================

* After going through some doc. I was able to debug the error and here is the code snippet below.
* **NEW CODE TO IMPLEMENT SAVING THE WIFI CREDENTIALS FROM THE WIFI MANAGER**

**Wifi\_config.lua**

**--- Connect to the wifi network ---**

**print("Connecting to WiFi access point...")**

**wifi.sta.disconnect()**

**wifi.setmode(wifi.STATIONAP)**

**counter = 0**

**if wifi.sta.getip() ~= nil then**

**---Get the ssid and passord of AP connected to**

**ssid, pwd, bssid\_set, bssid = wifi.sta.getdefaultconfig()**

**print("\nCurrent Station configuration:\nSSID : "..ssid .."\nPassword : "..password) ---------------this is for debuging purpose**

**tmr.alarm(1, 1000, tmr.ALARM\_SEMI, function()**

**counter = counter + 1**

**if counter < 20 then**

**if terasys.ssid == nil or terasys.ssid== ssid then ------------test if ssid is empty or if it exist**

**if terasys.pwd ~= pwd then ------------test if the ssid has the same password**

**terasys.ssid= ssid**

**terasys.pwd = pwd**

**end**

**print(terasys.ssid)**

**print(terasys.pwd)**

**elseif dapo.ssid == nil or dapo.ssid== ssid then**

**if dapo.pwd ~=pwd then**

**dapo.ssid= ssid**

**dapo.pwd= pwd**

**end**

**print(dapo.ssid)**

**print(dapo.pwd)**

**elseif deji.ssid == nil or deji.ssid == ssid then**

**if deji.pwd ~= pwd then**

**deji.ssid=ssid**

**deji.pwd=pwd**

**end**

**print(deji.ssid)**

**print(deji.pwd)**

**elseif wifi1.ssid== nil or wifi1.ssid == ssid then**

**if wifi1.pwd~= pwd then**

**wifi1.ssid= ssid**

**wifi1.pwd=pwd**

**end**

**print(wifi1.ssid)**

**print(wifi1.pwd)**

**elseif wifi2.ssid== nil then**

**if wifi2.pwd ~= pwd then**

**wifi2.ssid= ssid**

**wifi2.pwd= pwd**

**end**

**print(wifi2.ssid)**

**print(wifi2.pwd)**

**else**

**print ("All the table is populated")**

**end**

**end**

**end)**

**else**

**enduser\_setup.start(**

**function()**

**print("End Module Connected")**

**do ------------------------------------ this is for debugging purpose**

**x=wifi.sta.getapinfo()**

**y=wifi.sta.getapindex()**

**print("\n Number of APs stored in flash:", x.qty)**

**print(string.format(" %-6s %-32s %-64s %-18s", "index:", "SSID:", "Password:", "BSSID:"))**

**for i=1, (x.qty), 1 do**

**print(string.format(" %s%-6d %-32s %-64s %-18s",(i==y and ">" or " "), i, x[i].ssid, x[i].pwd and x[i].pwd or type(nil), x[i].bssid and x[i].bssid or type(nil)))**

**end**

**end**

**end,**

**function(err, str)**

**print("enduser\_setup: Err #" .. err .. ": " .. str)**

**end,**

**print -- Lua print function can serve as the debug callback**

**);**

**wifi.sta.connect()**

**end**

**------------------------------------------------------------------------------------------------------------------------------**

**---this block is for debugging purpose. i want to see if the tables has been updated**

**tmr.alarm(2, 1000, tmr.ALARM\_SEMI, function()**

**if counter < 40 then**

**counter = counter + 1**

**print("wifi2 credential")**

**print(wifi2.ssid)**

**print(wifi2.pwd)**

**print("wifi1 credential")**

**print(wifi1.ssid)**

**print(wifi1.pwd)**

**print("Deji credential")**

**print(deji.ssid)**

**print(deji.pwd)**

**print("Pama credential")**

**print(dapo.ssid)**

**print(dapo.pwd)**

**print("Terasys credential")**

**print(terasys.ssid)**

**print(terasys.pwd)**

**end**

**end)**

**-- --------------------------------------------------------------Y**

**tmr.create():alarm(2000, tmr.ALARM\_AUTO, function(cb\_timer)**

**if wifi.sta.getip() == nil then**

**print("Waiting for IP address...")**

**else**

**cb\_timer:unregister()**

**print("WiFi connection established " )**

**tmr.create():alarm(1, tmr.ALARM\_SINGLE, function()**

**local clock = require("ntp-clock")**

**clock.sync()**

**end)**

**end**

**end)**

**tmr.create():alarm(10000, tmr.ALARM\_AUTO, function(cb\_timer)**

**if mykey == nil and wifi.sta.getip() ~= nil and rtctime.get() > 1000000 then**

**print("Acquiring auth keys.")**

**local cred = require("client\_credentials")**

**cred.acquire()**

**elseif mykey ~= nil then**

**cb\_timer:unregister()**

**end**

**end)**

**config.lua**

**--- SERVER AND API DEFINITIONS ---**

**server = "www.terasyshub.io"**

**port = 443**

**url\_t = "/api/v1/data"**

**url\_h = "/api/v1/data"**

**url\_cred = "/api/v1/keys"**

**ntpserver = "pool.ntp.org"**

**--- LOCATION ---**

**lat = 6.497492**

**lon = 3.382360**

**-- --- WIFI CONFIGURATION ---**

**terasys={}**

**terasys.ssid=nil**

**terasys.pwd=nil**

**terasys.save=true**

**terasys.auto=false**

**dapo={}**

**dapo.ssid=nil**

**dapo.pwd=nil**

**dapo.save=true**

**dapo.auto=false**

**deji={}**

**deji.ssid= nil**

**deji.pwd= nil**

**deji.save= true**

**deji.auto=false**

**wifi1= {}**

**wifi1.ssid = nil**

**wifi1.pwd = nil**

**wifi1.save = true**

**wifi1.auto=false**

**wifi2={}**

**wifi2.ssid= nil**

**wifi2.pwd= nil**

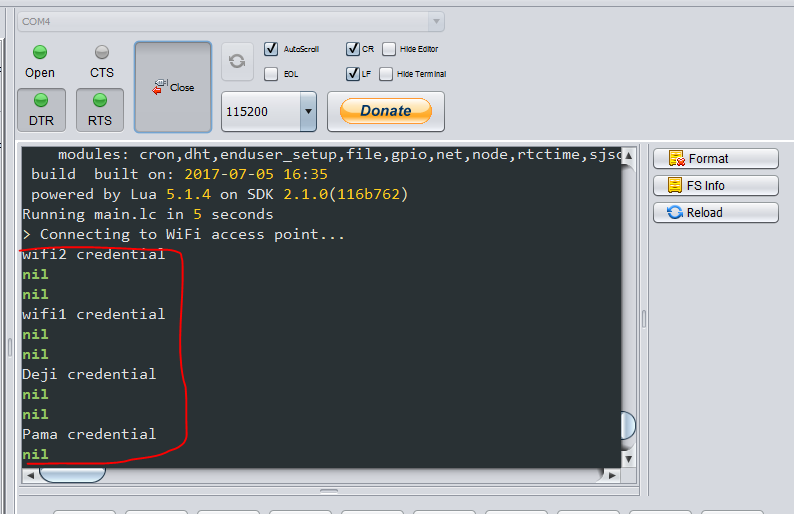
**wifi2.save= true**

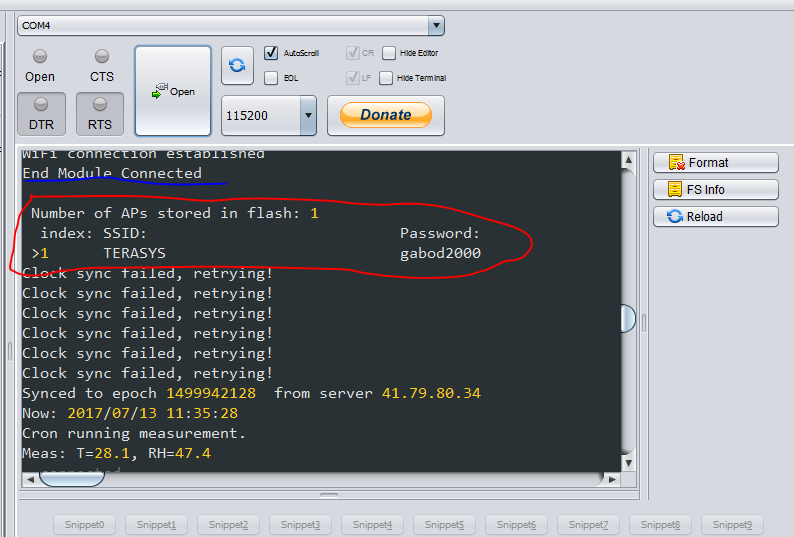
**wifi2.auto=false**

**--- DEVICE KEY ---**

**mykey = "98f4f6409ae547223e4c"**

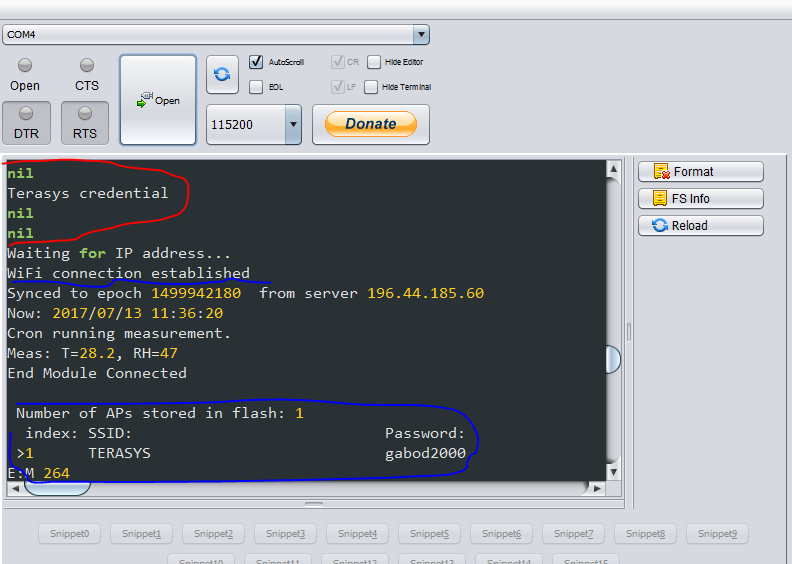
* **After compiling the code and starting the device, the wifi manager start and I was able to input scan for Ap and input credentials. It then showed a message of successful connection as usual.**
* **Below is the screenshot from the Esplorer.**



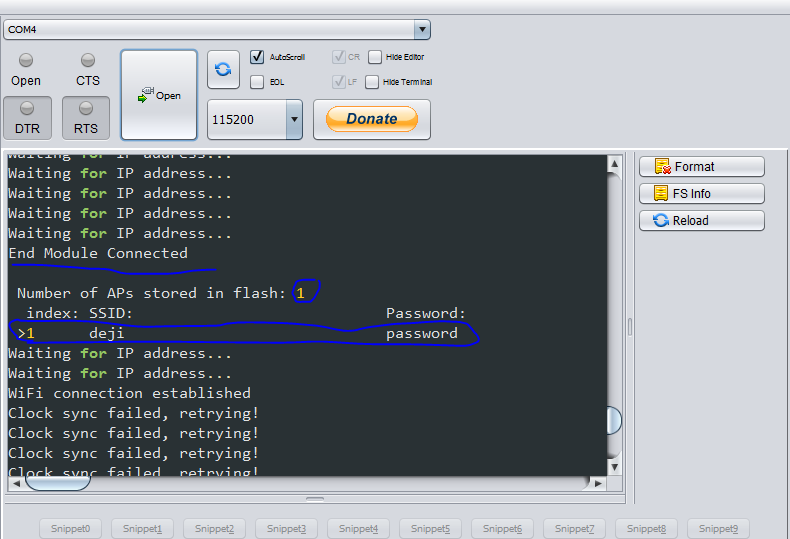


**NB:**  THE RED MARKING SHOWS IT SAVED THE CRENDENTIALS IN FLASH.

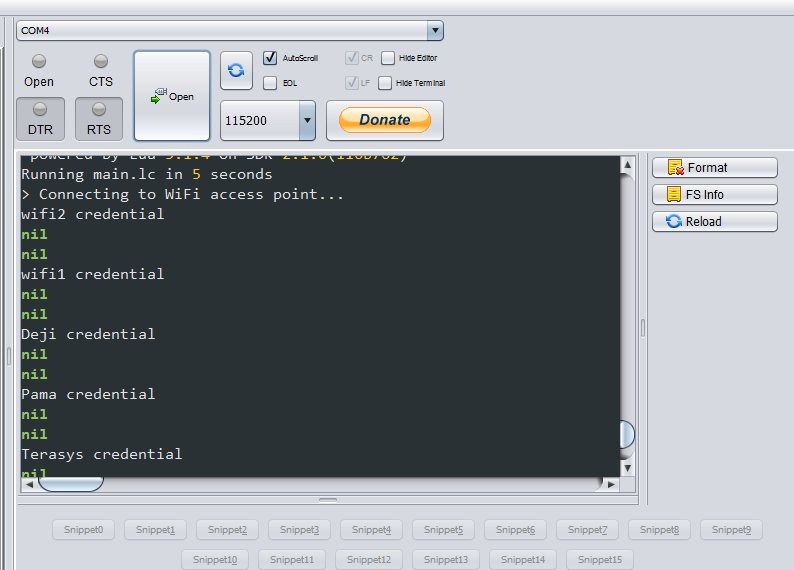
* I THEN UNPLUGED AND AFTER A MINUTE PLUGED IT AGAIN TO POWER SOURCE AND IT CONNECTED ITSELF TO ROUTER. IT ALSO RETAINED THE LAST WIFI CREDENTIAL IN MEORY.

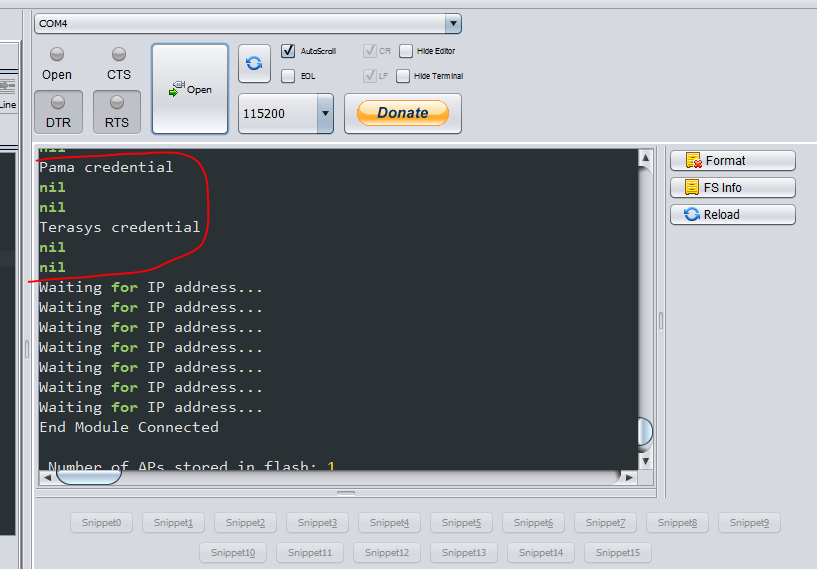


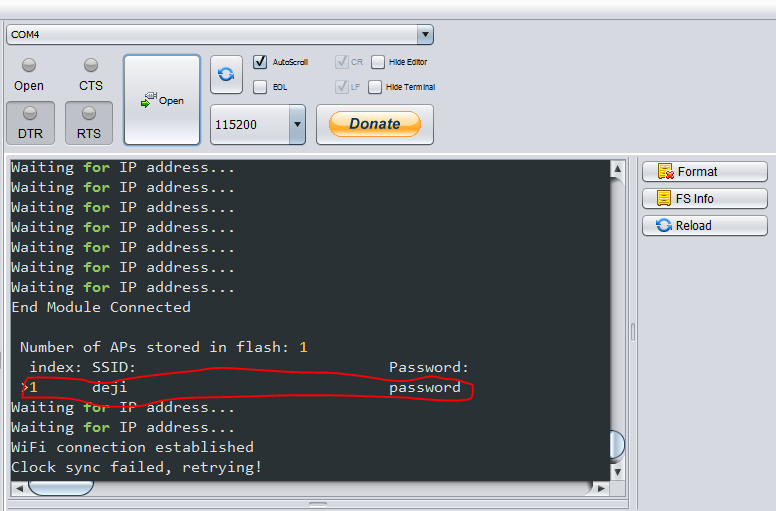
* **I THEN SWITED OFF THE ROUTER AND THE DEVICE STARTED THE WIFI MANAGER WHEN IT COULDN’T CONNECT TO THE ROUTER AND I IMPUTTED ANOTHER WIFI CREDENTIALS VIA THE WIFI MANAGER. IT THEN OVERWRITE THE EXISTING CREDENTIALS.**



* **I THEN UNPLUGGED THE DEVICE TO POWER IT AGAIN AND IT RECONNECTED ITSELF TO THE LAST WIFI SAVED.**







**NB**: I HAVE NOT IMPLEMENTED THE LOGIC WHERE IT WILL RECONNECT TO ANOTHER ROUTER IF THE ONE IT IS PRESENTLY CONNECTED TO FAILS. I CAN’T IMPLEMENT THAT WITHOUT HAVING CREDENTIALS SAVED IN MY EXISTING EMPTY TABLES.

**I WAS ABLE TO IMPLEMENT THE NETWORK SWITCHING IN THE SECOND UPDATED 2ND TEST CODE BELOW**

**WIFI\_CONNECT.LUA**

**---------------------------------------------------------------**

--THE BUG IN THIS CODE IS THAT WHEN IT CONNECT TO ANY OF THE PRE-DEFINED AP

--IT WILL CONNECT, MEASURE BUT NOT POST DATA BUT IF IT STARTS THE WIFI MANAGER

-- AND WE INPUT THE WIFI CREDENTIALS, IT WILL CONNECT AND START POSTING DATA------------------

---I THINK I’M HAVING ISSUE WITH THE TIMER PREFERENCE, I WILL LOOK INTO THAT---

--------------------------

--- Connect to the wifi network ---

print("Connecting to WiFi access point...")

---set wifi mode

wifi.setmode(wifi.STATIONAP)

--------------------choose available network--------

counter = 0

wifi.sta.config(terasys)

tmr.alarm(1, 1000, tmr.ALARM\_SEMI, function()

counter = counter + 1

if counter < 20 then

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. terasys.ssid)

tmr.start(1) -- restart

else

print("Connected to" .. terasys.ssid )

end

elseif counter < 40 then

wifi.sta.config(dapo)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. dapo.ssid)

tmr.start(1) -- restart

else

print("Connected to " .. dapo.ssid)

end

elseif counter < 60 then

wifi.sta.config(deji)

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect to " .. deji.ssid)

tmr.start(1) -- restart

else

print("Connected to " .. deji.ssid)

end

elseif counter < 80 then

if wifi1.ssid ~= nil then

wifi.sta.config(wifi1)

end

if wifi.sta.getip() == nil then

print("NO IP yet! Keep trying to connect " )

tmr.start(1) -- restart

else

print("Connected " )

end

-- elseif counter < 100 then

-- wifi.sta.config(wifi2)

-- if wifi.sta.getip() == nil then

-- print("NO IP yet! Keep trying to connect to " .. wifi2.ssid)

-- tmr.start(1) -- restart

-- else

-- print("Connected to " .. wifi2.ssid)

-- end

else

print("Out of options, giving up. Please input WiFi credentials in the WiFi Manager")

enduser\_setup.start(

function()

print("Connected")

end,

function(err, str)

print("enduser\_setup: Err #" .. err .. ": " .. str)

end,

print -- Lua print function can serve as the debug callback

);

local ip = wifi.sta.getip()

if ip ~= nil then

----Get the ssid and passord of AP connected to

ssid, password, bssid\_set, bssid = wifi.sta.getdefaultconfig()

print("\nCurrent Station configuration:\nSSID : "..ssid

.."\nPassword : "..password)

end

if wifi1.ssid == nil then

wifi1.ssid=ssid

wifi1.pwd=password

elseif wifi2.ssid == nil then

wifi2.ssid=ssid

wifi2.pwd=password

else

terasys = nil

terasys.ssid=ssid

terasys.pwd=password

end

end

end)

tmr.create():alarm(2000, tmr.ALARM\_AUTO, function(cb\_timer)

if wifi.sta.getip() == nil then

print("Waiting for IP address...")

else

cb\_timer:unregister()

print("WiFi connection established " )

tmr.create():alarm(1, tmr.ALARM\_SINGLE, function()

local clock = require("ntp-clock")

clock.sync()

end)

end

end)

tmr.create():alarm(10000, tmr.ALARM\_AUTO, function(cb\_timer)

if mykey == nil and wifi.sta.getip() ~= nil and rtctime.get() > 1000000 then

print("Acquiring auth keys.")

local cred = require("client\_credentials")

cred.acquire()

elseif mykey ~= nil then

cb\_timer:unregister()

end

end)

**CONFIG.LUA**

--- SERVER AND API DEFINITIONS ---

server = "www.terasyshub.io"

port = 443

url\_t = "/api/v1/data"

url\_h = "/api/v1/data"

url\_cred = "/api/v1/keys"

ntpserver = "pool.ntp.org"

--- LOCATION ---

lat = 6.497492

lon = 3.382360

-- -- --- WIFI CONFIGURATION ---

terasys={}

terasys.ssid="TERASYS"

terasys.pwd="gabod2000"

terasys.save=true

dapo={}

dapo.ssid="pama"

dapo.pwd="adedapoaderemi"

dapo.save=true

deji={}

deji.ssid= "deji"

deji.pwd= "password"

deji.save= true

wifi1= {}

wifi1.ssid = nil

wifi1.pwd = nil

wifi1.save = true

wifi2={}

wifi2.ssid= nil

wifi2.pwd= nil

wifi2.save= true

--- DEVICE KEY ---

mykey = "98f4f6409ae547223e4c"