



**UNIVERSITATEA
TEHNICĂ
DIN CLUJ-NAPOCA**

**Realizarea unei conexiuni la rețea în funcție de SSID
utilizând placa de dezvoltare PYNQ Z1**

Structura Sistemelor de Calcul

Autori: Pescaru Silviu și Vlasă Rafaella

Grupa: 30235

Profesor îndrumător: SL.Dr.Ing. Dragos Florin Lisman

FACULTATEA DE AUTOMATICA
ȘI CALCULATOARE

17 Ianuarie 2024

Cuprins

1	Rezumat	2
2	Introducere	2
3	Fundamentare teoretica	2
4	Proiectare si implementare	3
4.1	Setup	3
4.2	Conectare la retea	3
5	Rezultate experimentale	5
6	Concluzii	7
7	Referinte si bibliografie	7

1 Rezumat

Proiectul constă în conectarea plăcii Pynq Z1 la o rețea Wi-Fi folosind un adaptor USB Wi-Fi RALINK RT5370 [1]. Prin intermediul plăcii de dezvoltare Pynq Z1, care integrează un procesor ARM și un FPGA, se poate realiza configurarea adaptorului USB Wi-Fi pentru a stabili conexiunea la rețeaua Wi-Fi existentă.

2 Introducere

Pentru a putea realiza conectarea la o rețea Wi-Fi avem nevoie de o placa de dezvoltare Pynq Z1, un adaptor de Wi-Fi USB cu chipset RT5370, un cablu microUSB și un card de memorie microSD.

Se va face flash la o imagine de Linux de pe site-ul pynq.io/board.html după care placa se va conecta la calculator, se va porni din switch-ul de pe placa, iar când boot-ul este complet, se va introduce adaptorul de Wi-Fi USB care trebuie neapărat să aibă chipset RT5370, altfel realizarea conexiunii nu va fi posibilă.

3 Fundamentare teoretică

Conectarea la o rețea Wi-Fi implică utilizarea tehnologiei de rețea wireless pentru a permite dispozitivelor să comunice și să acceseze internetul sau să se conecteze la alte dispozitive fără a fi nevoie de cabluri fizice. Această tehnologie se bazează pe standardele IEEE 802.11 și utilizează unde radio pentru transmiterea datelor între dispozitive.

Iată principalele componente și concepte ale unei rețele Wi-Fi:

1. Punct de acces (Access Point - AP)

Un punct de acces este un dispozitiv care acționează ca interfață între dispozitivele wireless și rețeaua cablată (de exemplu, routerul). El emite semnale radio pentru a permite dispozitivelor să se conecteze la rețea. Un router tipic de acasă integrează funcționalitatea de punct de acces Wi-Fi.

2. SSID (Service Set Identifier)

Este numele rețelei Wi-Fi la care dispozitivele se pot conecta. Alegerea și difuzarea unui SSID este esențială pentru identificarea rețelei și conectarea la aceasta.

3. Securitate Wi-Fi

Protocolul de securitate Wi-Fi (de exemplu, WEP, WPA, WPA2, WPA3) ajută la protejarea rețelei împotriva accesului neautorizat. Este esențial să fie utilizate parole puternice și metode de criptare robuste pentru a asigura securitatea rețelei.

4. Autentificare și asociere

Procesul de conectare a unui dispozitiv la o rețea Wi-Fi implică autentificarea și asocierea. Autentificarea verifică identitatea dispozitivului, în timp ce asocierea permite dispozitivului să obțină acces la rețeaua Wi-Fi.

5. DHCP (Dynamic Host Configuration Protocol)

Protocolul DHCP atribuie automat adrese IP dispozitivelor conectate la rețea Wi-Fi, permițându-le să comunice între ele și să acceseze internetul.

6. Calitatea semnalului și interferența

Calitatea semnalului Wi-Fi poate fi afectată de distanța față de punctul de acces, obstacole fizice, interferențe din alte dispozitive sau surse radio. Interferența din alte rețele Wi-Fi, dispo-

zitive electronice sau alte surse radio poate reduce performanța rețelei Wi-Fi.

Conectarea la o rețea Wi-Fi implică configurarea corectă a dispozitivului pentru a găsi și a se conecta la SSID-ul dorit, introducerea parolei (dacă este necesar) și asigurarea unei conexiuni stabile pentru accesul la internet sau pentru comunicarea între dispozitive.

4 Proiectare si implementare

Cerinte preliminare: placa de dezvoltare Pynq Z1, calculator cu aplicatie serial terminal(de exemplu PuTTY), card microSD (minim 8GB recomandat) si un cablu microUSB.

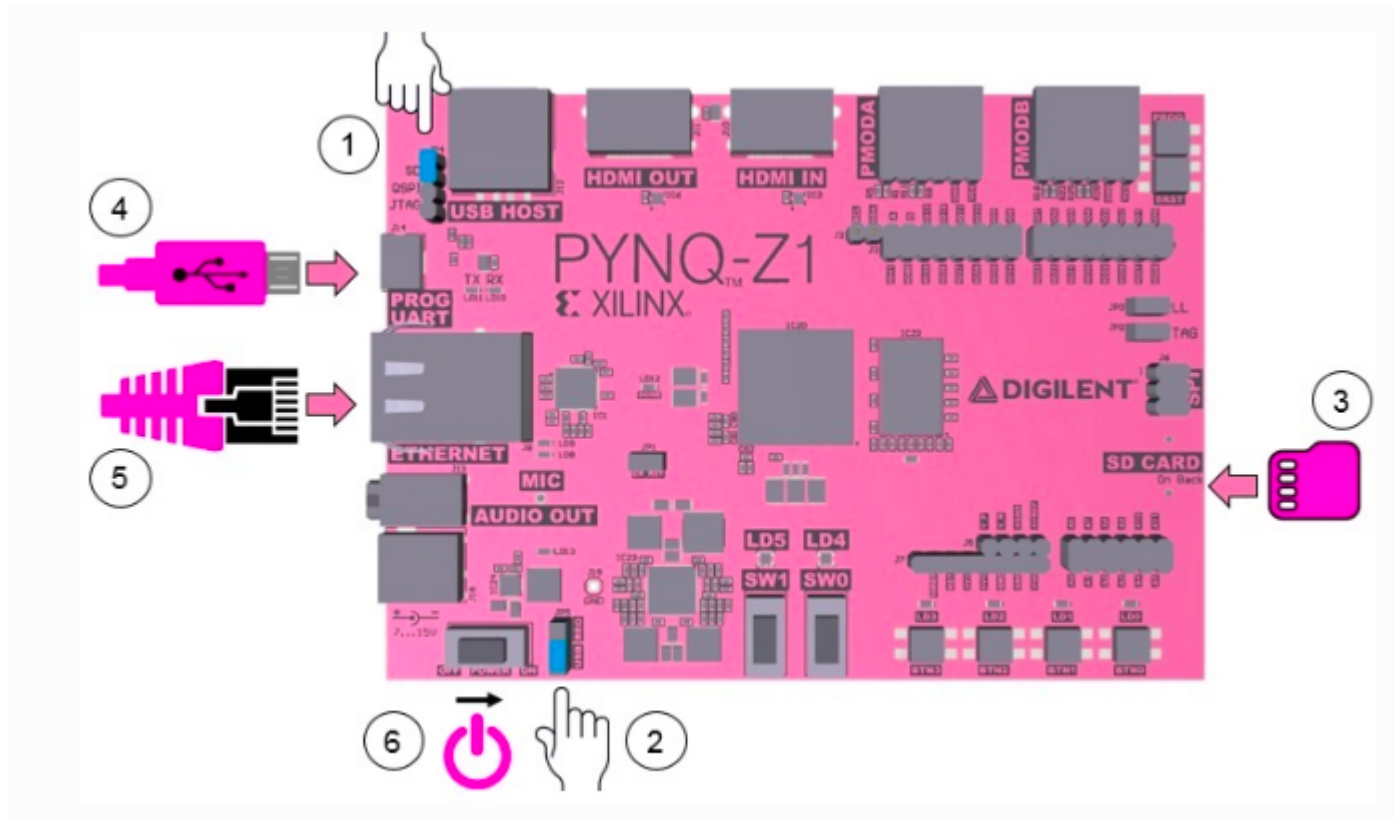


Figura 1: Pynq Board Setup

4.1 Setup

1. Setam jumperul JP4/Boot pe pozitia de SD ca in imaginea de mai sus [2]
2. Setam jumperul JP5/Power pe pozitia USB ca in imaginea de mai sus
3. Introducem cardul microSD ce contine imaginea bootabila pentru placa noastra de dezvoltare Pynq Z1
4. Conectam placuta prin microUSB la calculator si pornim placuta din switchul dedicat

4.2 Conectare la retea

Pentru a putea vedea ce se intampla pe placuta, dupa ce am realizat pasii de mai sus va trebui sa deschidem un serial terminal. In exemplu ce urmeaza voi folosi comanda “screen” prezenta in Linux.

```
screen 115200 /dev/ttyUSB1
```

- screen: comanda utilizata pentru a deschide un serial terminal
- 115200: baud rate
- /dev/ttyUSB1: portul USB la care placuta este conectata la calculator

ATENTIE! portul USB difera de la un calculator la altul.

Dupa rularea acestei comenzi vom vedea un terminal Ubuntu care ruleaza pe placuta noastra. O sa fim intampinati de un login unde vom folosi credentialele universal valabile (user: *xilinx*, password: *xilinx*) pentru a ajunge in directorul /home.

Pentru a putea sa ne conectam la o retea de Wi-Fi, va trebui sa introducem adaptorul Wi-Fi RaLink RT5370 iar apoi sa rulam comanda `ip a`, care ne va afisa toate interfetele de retea disponibile. Trebuie sa vedem printre toate acele interfete *wlan0* care va avea initial state DOWN.

Ca sa activam interfata ce ne va permite ulterior sa ne conectam la o retea de tip Wi-Fi, vom rula urmatoarele comenzi:

- `sudo ip link set wlan0 up`

Aceasta comanda schimba state-ul din DOWN in UP

- `sudo ifconfig wlan0 up`

Aceasta comanda va asigna o adresa IP interfetei noastre.

Acum rulam din nou comanda `ip a`, iar daca state-ul este UP inseamna ca putem folosi interfata de retea pentru a ne conecta. Vom face asta ruland urmatoarea comanda:

- `sudo nmcli dev wifi connect „SSID” password „pass”`

In loc de „SSID” vom scrie numele retelei la care dorim sa ne conectam, iar in loc de „pass” vom pune parola retelei respective.

Daca totul este in regula o sa ne apara un mesaj care ne anunta ca placuta s-a conectat la retea cu succes. Pentru a testa daca conexiunea este intr-adevar corespunzatoare rulam comanda:

- `ping ac.utcluj.ro -c 3`

Aceasta comanda va trimite 3 pachete catre `ac.utcluj.ro`, iar daca transmiterea lor este de 100% atunci conexiunea este realizata si stabila. Tot aici putem observa si despachetarea pachetelor.

Pentru a putea vedea toate retelele disponibile din raza de actiune a adaptorului wireless vom folosi urmatoarea comanda:

- `sudo iw dev wlan0 scan | grep SSID`

Aceasta comanda va scana folosind interfata wlan0 si apoi cu `| grep SSID` vom filtra toate datele astfel incat sa putem vedea doar numele retelelor.

Toate comenzile prezentate mai sus pot fi scrise intr-un script, rulat simplu prin comanda `wificonnect`.

```
xilinx@pynq:~$ cat /usr/bin/wificonnect
#!/bin/bash

# Verificam daca interfata wlan0 este pornita
if ip a | grep wlan0 > /dev/null; then
    if ip a | grep wlan0 | grep DOWN > /dev/null; then
        sudo ip link set wlan0 up
        sudo ifconfig wlan0 up
    fi
fi

# Introducem ca si argumente ssid si parola
read -p "Enter SSID: " ssid
read -s -p "Enter password: " pswd
echo

# Incercam conectarea
sudo nmcli dev wifi connect "$ssid" password "$pswd"
xilinx@pynq:~$
```

Figura 2: Connection Script

Conexiunea la retea se poate realiza si folosind un cablu ethernet conectat la router. Se va accesa portalul Jupyter, prin adresa IP a placii : 192.168.2.99:9090 sau prin adresa `http://pynq:9090`. Se vor scrie in notebook urmatoarele comenzi [3]:

```
1  #1. Create WiFi instance
2  from pynq.lib import Wifi
3
4  port = Wifi()
5
6  #2. Connect to a WIFI link
7  ssid = input("Type in the SSID:")
8  pwd = input("Type in the password:")
9  port.connect(ssid, pwd)
10
11 #3. Test connection
12 ! ping -I wlan0 www.yahoo.com -c 10
```

5 Rezultate experimentale

In urma rularii comenzilor/scriptului se pot observa urmatoarele:

- Scanarea retelelor este facuta in mod corespunzator; sunt listate toate SSID-urile din aria de acoperire

```
xilinx@pynq:~$ sudo iw dev wlan0 scan | grep SSID
SSID: TP-Link_8570
SSID: Apartamentul11
SSID: TP-LINK_39FE
SSID: DIGI-2gJV
SSID: Apartament10
SSID: Apartament21
SSID: Apartament 2
SSID: Apartament17
SSID: Apartament05
SSID: apartament12
SSID: Clanul Balog
SSID: MERCUSYS_18C0
SSID: TP-LINK_7526
```

Figura 3: SSID Listing

- Conexiunea la un hotspot mobil a fost realizata cu succes

```
xilinx@pynq:~$ wificonnect
Enter SSID: Silviu
Enter password:
Device 'wlan0' successfully activated with 'b33d68f7-53be-4ae2-b5b5-672bd7fb6a84'.
xilinx@pynq:~$
```

Figura 4: Successful Connection

- Pachetele au fost transmise si despachetate cu succes

```
xilinx@pynq:~$ ping ac.utcluj.ro -c 3
PING ac.utcluj.ro (193.226.5.8) 56(84) bytes of data.
64 bytes from hosting.utcluj.ro (193.226.5.8): icmp_seq=1 ttl=54 time=35.2 ms
64 bytes from hosting.utcluj.ro (193.226.5.8): icmp_seq=2 ttl=54 time=36.2 ms
64 bytes from hosting.utcluj.ro (193.226.5.8): icmp_seq=3 ttl=54 time=44.4 ms

--- ac.utcluj.ro ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 35.225/38.630/44.436/4.125 ms
xilinx@pynq:~$
```

Figura 5: Successful Ping

Observatie. Din cauza ca adaptorul USB Wi-Fi RALINK RT5370 este limitat din punct de vedere al protocolului de securitate, nu va putea sa se conecteze la retele ce folosesc un protocol de securitate mai avansat de WPA2 si WPA2-PSK. De exemplu:

```
xilinx@pynq:~$ wificonnect
Enter SSID: Apartament6
Enter password:
Error: Connection activation failed: (7) Secrets were required, but not provided.
xilinx@pynq:~$
```

Figura 6: Failed Connection

Nota. Incercarea de a conecta placuta Pynq Z1 folosind alt tip de adaptor decat cel prezentat mai sus poate duce la incapacitatea placii sa "vada" retelele Wi-Fi.

6 Concluzii

In urma realizarii acestui proiect am acumulat mai multe cunostinte legate de programarea unei placute de dezvoltare. Am devenit mai familiari cu folosirea terminalului Linux si cu notiuni de baza despre retele si conectivitate Wi-Fi. Am interactionat pentru prima data cu mediul de dezvoltare Jupyter si cu o placuta programabila in Python.

7 Referinte si bibliografie

Referințe

[1] <https://discuss.pynq.io/t/ralink-wifi-dongle-with-pynq-pynq-notebooks-common-wifi-ipynb/>

- [2] https://pynq.readthedocs.io/en/latest/getting_started/pynq_z1_setup.html
- [3] <https://github.com/Xilinx/PYNQ/blob/master/pynq/notebooks/common/wifi.ipynb>

Bibliografie

<https://www.malwarebytes.com/what-is-an-ssid#:~:text=The%20term%20SSID%20may%20sound,is%20a%20unique%20network%20name>
<https://digilent.com/reference/pmod/pmodwifi/start>
<https://digilent.com/reference/programmable-logic/pynq-z1/start>
<https://discuss.pynq.io/t/pynq-z1-wifi-dongle-support/4879>
https://www.youtube.com/watch?v=SuXkbcK3w9E&ab_channel=CathalMcCabe
<https://stackoverflow.com/questions/24518522/run-python-script-at-startup-in-ubuntu>
<https://pynq.readthedocs.io/en/v2.1/faqs.html>
https://www.reddit.com/r/FPGA/comments/ai4lok/pynqz1_wifi_help/