

(WNA) Wireless Networks Analyzer

Introducere

- Nume: Ploscaru Alexandru
- Grupa: 331CD

WNA este un mic dispozitiv de securitate cibernetica care ajuta la scanarea retelelor din proximitatea acestuia, avand mai multe optiuni de functionare. Acesta poate detecta atacuri cibernetice asupra retelelor WiFi, poate monitoriza traficul din apropierea acestuia, dar si sa genereze fisiere pcap care contin pachetele interceptate de dispozitiv, care vor fi stocate pe un card SD.

Dispozitivul este un tool de pentesting, pentru interceptarea traficului de retea si monitorizarea acestuia, pentru a detecta atacuri asupra retelelor, si pentru a invata atat retelistica, cat si punerea in practica a cunostintelor dobandite in cadrul laboratoarelor de PM.

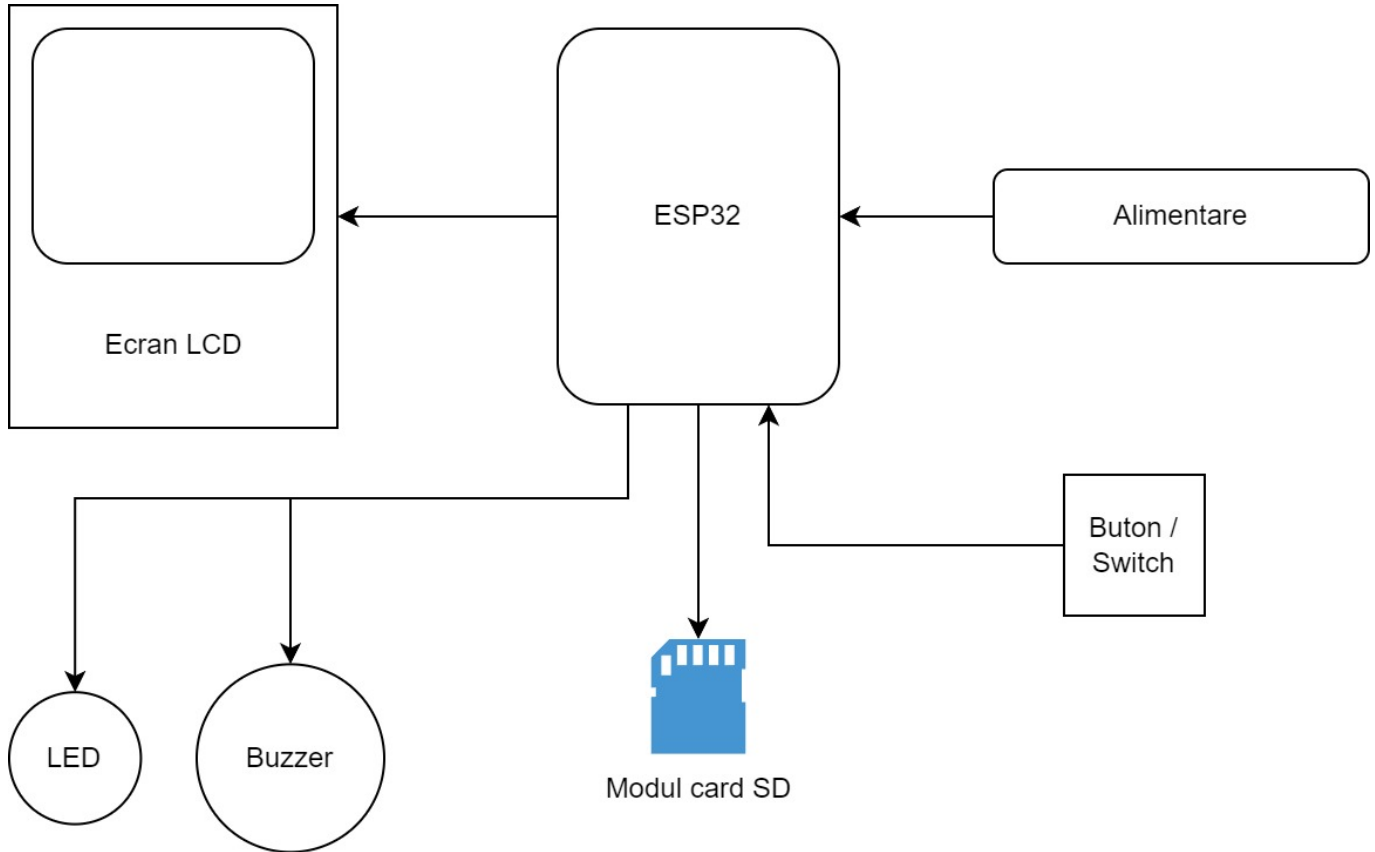
Ideea de baza a proiectului a fost imbinarea securitatii cibernetice cu un dispozitiv fizic, construind un tool util si portabil.

Un astfel de proiect ar fi util pentru monitorizarea traficului dintr-o retea WiFi, cu scopul de a alarma din timp initierea unui atac cibernetic asupra retelei, dar si interesant pentru pasionatii de securitate.

Descriere generală

WNA este un tool care la baza foloseste o placuta ESP32, in mod special modulul WiFi integrat din aceasta, pentru a scana retelele din proximitatea acestuia, cuprinzand si puterea semnalului si verificarea securitatii retelei; pentru a intercepta si monitoriza traficul din retea, stocandu-l pe un card SD pentru verificarea ulterioara a pachetelor, precum si pentru a detecta atacurile de deautentificare initializate asupra retelelor WiFi inconjuratoare. Datele sunt afisate pe un ecran LCD. La inceput va exista un meniu, pentru a alege modul de functionare al dispozitivului, cu ajutorul unui joystick. Atunci cand dispozitivul este in modul de detectare a atacurilor, daca un atac este interceptat, LED-ul se va aprinde si buzzer-ul va scoate un sunet. Placuta ESP32 va avea conectata o sursa de alimentare.

Schema bloc:

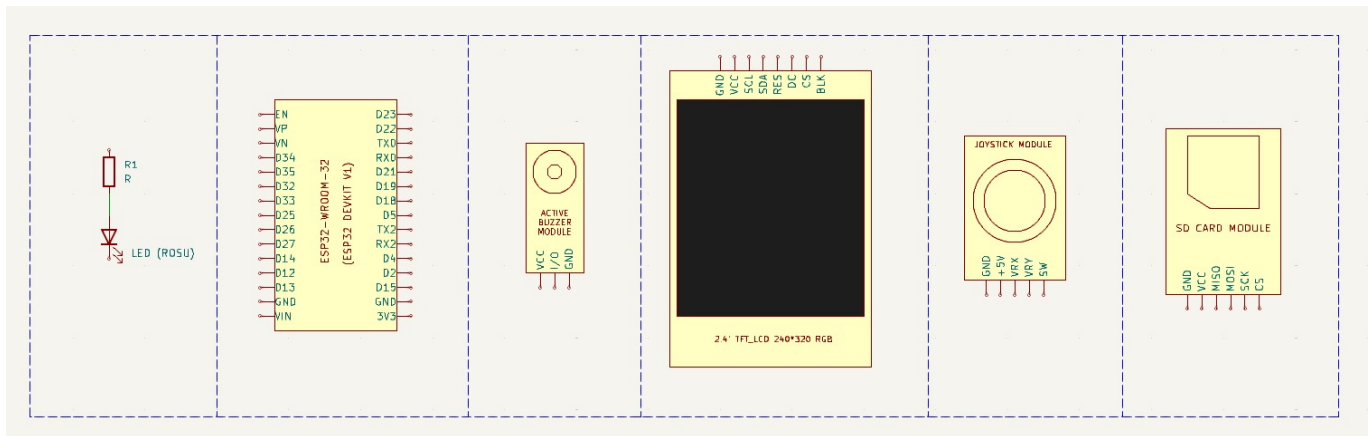


Hardware Design

Lista de componente:

- Placuta ESP32 cu modul WiFi integrat (ESP32 DEVKIT V1 - are la baza ESP-WROOM-32);
- LED rosu + rezistor 220 ohm;
- Modul cu Buzzer activ;
- Modul LCD de 2.4" cu SPI și Controller ILI9341 (240×320 px);
- Modul Joystick Biaxial cu 5 pini;
- Modul MicroSD;
- Card SD (Micro SDHC);
- Breadboard;
- Jump wires;
- Sursa alimentare.

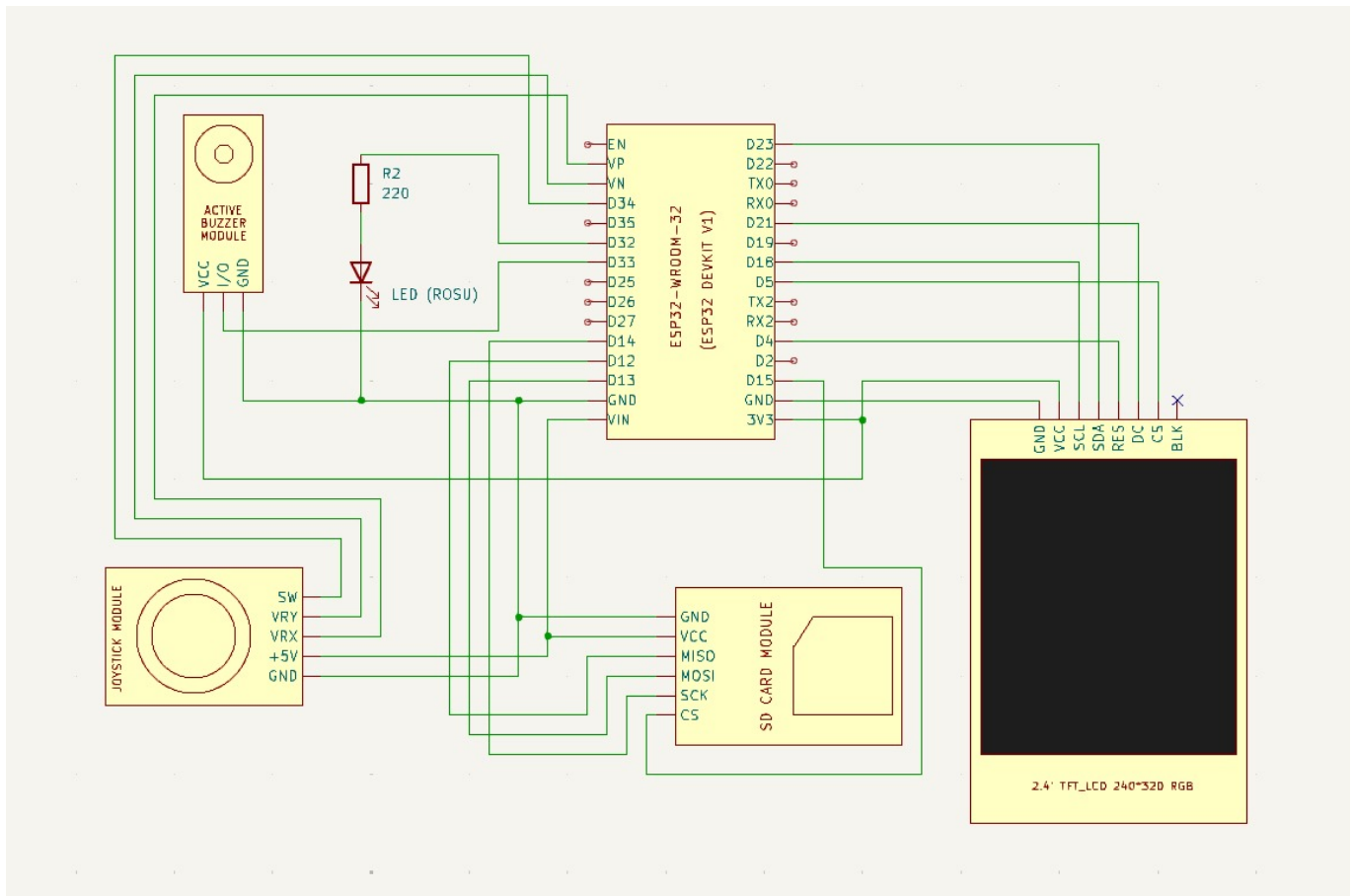
Schemele electrice pentru componente individuale:



Schemele din poza de mai sus, de la stanga la dreapta, reprezinta:

- LED rosu, care pentru a functiona in parametri optimi are nevoie de un rezistor de 220 ohm.
- Placa de dezvoltare ESP32 cu WiFi și Bluetooth 4.2 integrate (ESP32 DEVKIT V1) - microcontrollerul principal din acest proiect, cu 30 de pini
- Modul Buzzer Activ, cu 3 pini:
 - VCC - pentru alimenare (3V3)
 - I/O - folosit pentru trimiterea semnalelor catre buzzer
 - GND - pentru alimentare
- Display LCD TFT cu SPI, cu 8 pini:
 - GND - pentru alimentare
 - VCC - pentru alimentare (3V3)
 - SCL - pentru SPI_CLK (serial clock)
 - SDA - pinul SPI_MOSI (serial data from microcontroller)
 - RES - pinul de reset (TFT reset pin)
 - DC - data command pin
 - CS - chip select pin
 - BLK - input pentru backlight control (nefolosit)
- Modul Joystick Biaxial, cu 5 pini:
 - GND - pentru alimentare
 - +5V - pentru alimentare (5V)
 - VRX - pentru a trimite output analog catre microcontroller cu pozitia joystick-ului pe axa Ox
 - VRY - pentru a trimite output analog catre microcontroller cu pozitia joystick-ului pe axa Oy
 - SW - pentru a trimite un semnal catre microcontroller daca joystick-ul a fost apasat
- Modul MicroSD, cu SPI, cu 6 pini:
 - GND - pentru alimentare
 - VCC - pentru alimentare (5V)
 - MISO - serial data to microcontroller
 - MOSI - serial data from microcontroller
 - SCK - serial clock
 - CS - chip select pin

Schema electrica a proiectului:



Explicatie schema electrica:

- LED
 - Rezistorul de 220 ohm este conectat la pinul + al LED-ului.
 - Pinul - al LED-ului este conectat la GND.
 - Pinul + al LED-ului este conectat la pinul D32 al ESP32 (corespunde cu pinul GPIO32)
- Modul buzzer activ
 - GND - conectat la GND pe ESP32
 - VCC - 3V3 pe ESP32
 - I/O - D33 pe ESP32 (GPIO33)
- Display LCD TFT, cu SPI
 - GND - GND pe ESP32
 - VCC - 3V3 pe ESP32
 - SCL - D18 pe ESP32 (VSPI_CLK)
 - SDA - D23 pe ESP32 (VSPI_MOSI)
 - RES - D4 pe ESP32 (GPIO4)
 - DC - D21 pe ESP32 (GPIO21)
 - CS - D5 pe ESP32 (VSPI_CS)
 - BLK - neconectat
- Modul Joystick
 - GND - GND pe ESP32
 - +5V - VIN pe ESP32 (5V)
 - VRX - VP pe ESP32 (ADC1_0)
 - VRY - VN pe ESP32 (ADC1_3)
 - SW - D34 pe ESP32 (ADC1_6)
- Modul MicroSD, cu SPI
 - GND - GND pe ESP32

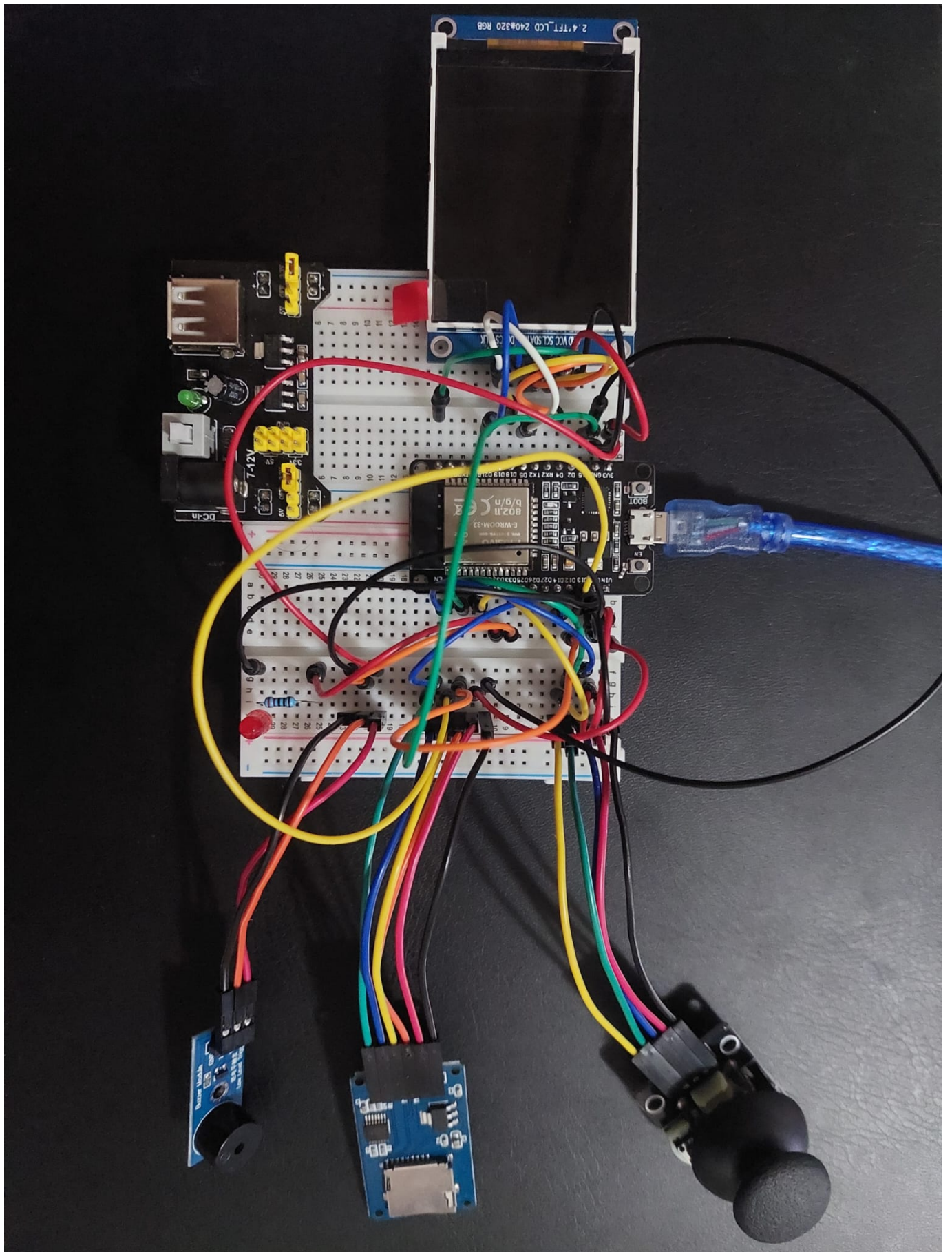
- VCC - VIN pe ESP32 (5V)
- MISO - D12 pe ESP32 (HSPI_MISO)
- MOSI - D13 pe ESP32 (HSPI_MOSI)
- SCK - D14 pe ESP32 (HSPI_CLK)
- CS - D15 pe ESP32 (HSPI_CS)

Pinout ESP32 DEVKIT V1:



Nota: Placuta ESP32 are 3 interfete SPI: SPI, VSPI, HSPI. Din cate am inteles, interfata SPI nu trebuie folosita, asa ca am utilizat VSPI pentru ecranul LCD TFT si HSPI pentru modulul microSD.

Proiect (15.05.2024):



Software Design

Codul sursa pentru acest proiect poate fi gasit pe GitHub:

https://github.com/PloscaruAlex/WNA_Wireless-Network-Analyzer

Mediu de dezvoltare

Am utilizat Arduino IDE pentru a putea programa placuta ESP32 si pentru a folosi diverse functii si tool-uri. Am folosit si Serial Monitor pentru testarea componentelor, precum joystickul.

Librarii folosite

- SPI.h - pentru a putea utiliza modulul de card SD si display-ul LCD;
- TFT_eSPI.h - pentru utilizarea display-ului LCD;
- ezButton.h - pentru a putea utiliza joystickul mai usor;
- WiFi.h - necesar folosirii modulului integrat de WiFi de pe placuta ESP32;
- PCAP.h - crearea fisierelor PCAP care vor fi stocate pe cardul sd (<https://github.com/spacehuhn/ArduinoPcap/tree/master?tab=readme-ov-file>);

Functii implementate

- drawMenu() - afiseaza meniul dispozitivului pe LCD;
- network_analyzer_setup() - initializeaza setarile pentru analizatorul de retea;
- network_analyzer_loop() - analizeaza retelele din jur si afiseaza pe LCD urmatoarele: SSID, RSSI, Channel, Encryption Type;
- wifi_promiscuous1() - functie care initializeaza setarile pentru detectarea pachetelor de deautentificare;
- setchannel() - schimba canalul wifi;
- detectorSetup() - functie de setup pentru detectarea atacurilor;
- detectorLoop() - functie care detecteaza atacurile asupra retelelor;

Rezultate Obținute

Concluzii

Download

Jurnal

Am schimbat putin pini joystick-ului, de la VIN l-am conectat la 3V3, iar SW l-am conectat la TX2 (pentru a functiona corespunzator).

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/avaduva/aploscaru2505>



Last update: **2024/05/27 13:01**