

Sprawozdanie laboratorium 2, SKPS

Instalacja OpenWRT z wykorzystaniem systemu ratunkowego

Wpisujemy komendy zgodnie z instrukcją do tego podpunktu. Jedynie przy mountowaniu partycji pojawił się błąd, który nakazywał wykonanie fsck. Został zignorowany zgodnie z zaleceniem danym na lab1. Nie wystąpił błąd z DNS więc nie naprawialiśmy go.

Dostęp do internetu z rp4

Zgodnie z instrukcją zamieniliśmy plik network i sprawdziliśmy że Internet działa co widać na screenshot-cie

[illegible]

```
root@OpenWrt:/# nano
/bin/ash: nano: not found
root@OpenWrt:/# vi etc/config/network
root@OpenWrt:/# /etc/init
init.d/ inittab
root@OpenWrt:/# /etc/init.d/network reload
[ 365.243224] br-lan: port 1(eth0) entered disabled state
[ 365.254961] device eth0 left promiscuous mode
[ 365.259423] br-lan: port 1(eth0) entered disabled state
[ 365.297694] bcmgenet fd580000.ethernet eth0: Link is Down
[ 365.345155] bcmgenet: Skipping UMAC reset
[ 365.350667] bcmgenet fd580000.ethernet: configuring instance for external RGMII
[ 365.358300] bcmgenet fd580000.ethernet eth0: Link is Up - 100Mbps/Full - flow control rx/tx
[ 365.389703] bcmgenet fd580000.ethernet eth0: Link is Down
[ 365.399779] bcmgenet: Skipping UMAC reset
[ 365.405201] bcmgenet fd580000.ethernet: configuring instance for external RGMII
[ 365.412790] bcmgenet fd580000.ethernet eth0: Link is Up - 100Mbps/Full - flow control rx/tx
'radio0' is disabled
'radio0' is disabled
root@OpenWrt:/# ping 8.8.8.8 -c 1
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: seq=0 ttl=118 time=2.622 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 2.622/2.622/2.622 ms
root@OpenWrt:/#
```

Poprawnie załadował się interfejs sieciowy LuCI

Pakiety zostały zaktualizowane oraz dla wygody zostało zainstalowane nano. Pakiety gpod-tools i inne wymienione w instrukcji zostały zainstalowane z LuCI.

W ramach testów na podstawie instrukcji sprawdziliśmy czy działa obsługa GPIO - działało.

Zadanie 1

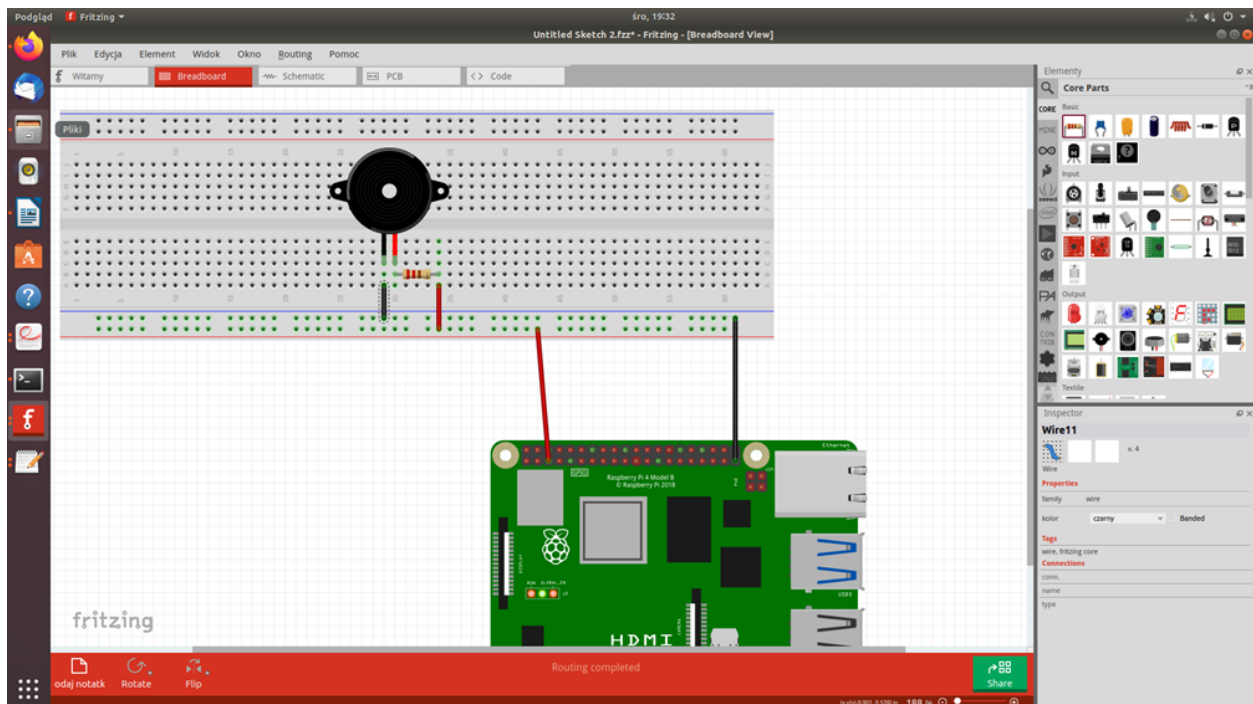
Pierwszy program (plik znajduje się na repo) napisaliśmy z użyciem biblioteki gpio4, wartość ustawiana na 0 wyłączała diodę, a ustawienie na 1 ją włączało.

Zadanie 2

Kod na repozytorium, do wytworzenia go użyliśmy pracy domowej w której zmieniliśmy aby zamiast wyprintowania czasu on i off dioda była ustawiana na on na czas on i wyłączana na czas off. Zmiana czasu była o równe wartości (nie używaliśmy funkcji sin, ponieważ uznaliśmy że to samo możemy osiągnąć prościej).

Zadanie 3

Wyszukaliśmy częstotliwości w internecie, korzystając z funkcji z pracy domowej ze skryptu do częstotliwości wypuszczamy na wyjście gpio zadane częstotliwości. Schemat poniżej.



Zadanie 4

Napisaliśmy program aby w momencie naciśnięcia przycisku zapaliła się dioda.

Zadanie 5

Wybraliśmy użycie dalmierza, stworzyliśmy schemat, który widać poniżej. Program załączony w repo. Z powodu braku czasu zainspirowaliśmy się kodem z Internetu i zmieniliśmy go do naszych potrzeb. W nieskończonej pętli, wywoływana jest funkcja do mierzenia dystansu i wyświetlany zmierzony przez nią dystans. Nadajemy krótki sygnał trigger. Bierzemy aktualny timestamp z sygnału echo oraz timestamp, gdy sygnał zostanie odebrany. Odległość to prędkość razy czas, więc mnożymy prędkość dźwięku razy różnicę czasu (obliczaną z timestamp-ów). Wynik skalujemy z ns na s (ponieważ używamy prędkość dźwięku 340 m/s) - rezultat dzielimy na $1e9$. I wtedy wynik jest w metrach.

