Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Gregor Kerševan

Sobivanje protokolov BLE in Wi-Fi

diplomsko delo

univerzitetni študijski program
prve stopnje
računalništvo in informatika

Mentor: prof. dr. Nikolaj Zimic

Ljubljana, 2019

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Protokol Bluetooth različica 4 in višje različice (Bluetooth Low Energy - BLE) omogočajo prenos manjših količin podatkov in so optimizirani za majhno porabo energije. Poenostavljeno je tudi iskanje novih naprav in priključevanje.

Ker BLE uporablja isto frekvenčno območje kot Wi-Fi, se poraja vprašanje, koliko se protokola med seboj motita? Wi-Fi lahko oddaja z višjo močjo, zato lahko pričakujemo težave predvsem pri BLE.

V diplomski nalogi preučite vpliv protokola Wi-Fi na prenos podatkov preko protokola BLE. Preverite delovanje izogibanja zasedenih kanalov, ki ga omogoča protokol BLE.

Zahvaljujem se mentorju prof. dr. Nikolaju Zimicu, ki mi je omogočil izvedbo diplomske naloge s posojo strojne opreme ter z usmerjanjem z napotki in nasveti. Zahvala gre tudi družini in prijateljem, ki so me podpirali v času študija.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
PAN	Personal Area Network	Osebno omrežje
IoT	Internet of Things	Internet stvari
BLE	Bluetooth Low Energy	Bluetooth z majhno porabo energije
BR	Basic Rate	Osnovna hitrost
EDR	Enhanced Data Rate	Izboljšana podatkovna hitrost
ISM	Industrial, Scientific and Medical	Industrijski, znanstveni in zdravstveni
SIG	Special Interest Group	Posebna namenska skupina
GFSK	Gaussian Frequency Shift Keying	Modulacija s frekvenčnim pomikom z Gaussovim filtriranjem
FHSS	Frequency Hopping Spread Spectrum	Razpršeni spekter s frekvenčnim skakanjem
AFHSS	Advanced FHSS	Napredni FHSS
IRK	Identity Resolving Key	Ključ za razreševanje identitete
CRC	Cyclic Redundancy Check	Ciklično preverjanje redundanc
PDU	Packet Data Unit	Paketna podatkovna enota
RFU	Reserved for Future Use	Rezervirano za uporabo v prihodnosti
MIC	Message Integrity Check	Preverjanje neokrnjenosti sporočila
LLID	Link Layer ID	Identifikacijska številka povezovalne plasti
SN	Sequence Nimber	Sekvenčna številka
NESN	Next Expected Sequence Number	Naslednja pričakovana sekvenčna številka

MD	More Data	Več podatkov
STK	Short Term Key	Kratkoročni ključ
MITM	Man In The Middle (Attack)	(Napad) s posrednikom
LTK	Long Term Key	Dolgoročni ljuč
RSSI	Received Signal Strength Indicator	Indikator moči sprejetega signala
LLCP	Link Layer Control Protocol	Nadzorni protokol povezovalne plasti

Povzetek

Naslov: Sobivanje protokolov BLE in Wi-Fi

Avtor: Gregor Kerševan

Bluetooth Low Energy je ena izmed bolj popularnih tehnologij, ki se uporablja za implementacijo komunikacije manjših elektronskih naprav. Zaradi delovanja v frekvenčno zelo zasedenem območju pa je izpostavljen mnogim motnjam. Diplomsko delo se osredotoča na vpliv motenj Wi-Fi na komunikacijo BLE naprav.

S praktičnimi poskusi želimo prikazati načine izmikanja motnjam ter analizirati uspešnost komunikacije ob prisotnosti motenj. Ugotovili smo, da motnje vplivajo na oglaševalske kanale le v neposredni bližini BLE naprav. Ob vzpostavljeni BLE komunikaciji je vpliv motenj na enem Wi-Fi kanalu majhen, saj napravi uporabljata mehanizem skakanja med kanali, ki skoraj izniči njihov vpliv. Ob motnjah na celotnem frekvenčnem spektru ostaneta dva kanala, ki sta vedno zadnja dva podatkovna kanala.

Ključne besede: BLE, Wi-Fi, motnje.

Abstract

Title: Wi-Fi and BLE coexistence

Author: Gregor Kerševan

Bluetooth Low Energy is one of the most popular technologies used by small electronic devices for wireless communication. Because it operates in an overcrowded frequency band it is exposed to a lot of interferences. This thesis focuses on the influence of Wi-Fi interferences on the communication of BLE devices.

With practical experiments, we want to show different ways of countering interferences and analyze how successful BLE communication is. We found out, that interferences have an effect on advertising channels only in close proximity. When a BLE communication is established, Wi-Fi traffic on one Wi-Fi channel has little effect on the success rate of packets, as BLE uses a frequency hopping method, which almost completely negates any effect. When there is Wi-Fi traffic on the whole BLE frequency band only two channels remain, which are always the last two data channels.

Keywords: BLE, Wi-Fi, interferences.

Poglavje 1

Uvod

V vsakdanjem življenju uporabljamo vedno več elektronskih naprav, ki nas povezujejo v brezžična omrežja. Vedno bolj popularna postajajo osebna omrežja PAN (*Personal Area Network*) ter naprave, povezane v internet stvari IoT (*Internet of Things*). Ena najbolj uporabljanih tehnologij na teh področjih pa je *Bluetooth Low Energy* (BLE), ki se hitro širi v vedno več naprav. Predvideva se, da bo do leta 2023 kar 90% vseh Bluetooth naprav podpiralo BLE [1]. Samo v letu 2019 naj bi bilo odposlanih kar 2.7 milijard naprav, ki bi podpirale tako BLE kot Bluetooth BR/EDR (*Basic Rate/Enhanced Data Rate*). S prihodom Bluetooth 5, ki je uradno izšel leta 2016, pa je bilo izboljšanih še več vidikov varčevanja energije ter performanc komunikacije.

En izmed problemov, ki se je pojavil ob povečanju števila brezžičnih naprav, pa je zasedenost frekvenčnega območja ISM (*Industrial, Scientific and Medical*). Veliko tehnologij namreč uporablja iste frekvence in morajo zato zagotoviti sobivanje z drugimi protokoli. Na frekvenčnem območju 2.4 GHz, kjer deluje tudi BLE, tako najdemo Wi-Fi, ZigBee, brezžične telefone, mikrovalovne pečice, računalniške miške ... Protokol Bluetooth BR/EDR se je problema motenj lotil s skakanjem med frekvencami ter beleženjem zasedenih frekvenc, ki se jim v prihodnje izogiba. BLE je prevzel oba načina izogibanja motnjam od Bluetooth BR/EDR, le da je vpeljal nekaj svojih sprememb.

V tej diplomski nalogi bomo poskušali predstaviti problem motenj pri protokolu BLE ter kakšen vpliv ima na njegove performance. Kot motnje smo si izbrali protokol Wi-Fi, ki je prisoten že skoraj povsod in zato predstavlja eno izmed večjih ovir BLE-ju. Z različnimi poskusi bomo preizkusili, na kakšnih razdaljah začne Wi-FI motiti BLE ter kako učinkovito deluje skakanje med frekvencami pri BLE-ju.

Poglavje 2

Opis BLE

Od začetka nastajanja protokola BLE so njegovi avtorji želeli novo tehnologijo, ki bi bila namenjena manjšim napravam, senzorjem in podobnim aparaturam. Namenjena bi bila trgu IoT, ki se je takrat šele razvijal in ni imel prevladujoče tehnologije komuniciranja med napravami. Pogoji so bili predvsem zelo majhna poraba energije in enostavna implementacija. Pomembno je razlikovati med Bluetooth BR/EDR in BLE, saj kljub temu da sta oba Bluetooth, se razlikujeta tako po namembnosti kot po implementaciji. BLE se uporablja za pametne ure, pedometre, senzorje v trgovinah, ki zaznavajo, kje se kupci nahajajo, merilce glukoze in mnoge druge naprave. Za naprave, kot so brezžični zvočniki, telefoni in tablice, ki pošiljajo večjo količino podatkov, so bolj primerne druge verzije Bluetooth oz. drugi protokoli. Skupna vsem tem napravam je predvsem majhna poraba energije, saj nekatere od teh naprav delujejo po več let z isto baterijo.

2.1 Zgodovina

Bluetooth je izšel leta 1999 kot protokol, namenjen brezžičnemu prenosu podatkov in zvoka na kratke razdalje. Osnovala so ga podjetja Ericsson, Nokia, IBM, Intel in Toshiba, nadaljni razvoj pa je prevzelo združenje Bluetooth SIG (*Special Interest Group*). Glavni izdelki, ki so implementirali Bluetooth, so

bili predvsem zvočniki, računalniške miške, tipkovnice ... Torej izdelki, kjer je Bluetooth nadomestil potrebo kablov. Po verziji 1.0 je leta 2004 prišla verzija 2.0, ki je prinesla EDR, 2009 pa je prišla še verzija 3.0, ki je z načinom prenosa, podobnim Wi-Fi-ju, omogočala še hitrejše prenose.

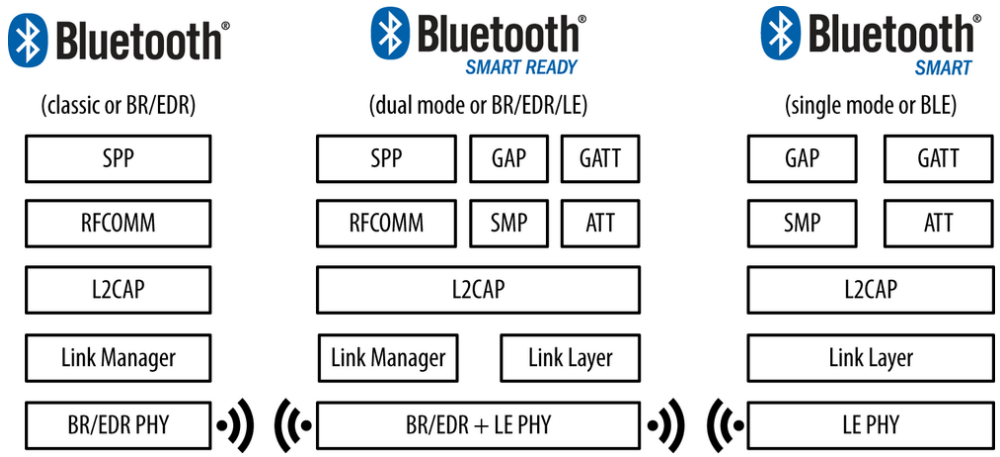
Leta 2001 so pri Nokiji začeli razvijati novo tehnologijo, ki bi temeljila na majhni porabi energije. Svoje ugotovitve so predstavili leta 2004, ko so objavili raziskavo *Bluetooth Low End Extension*. Po nadaljnem razvoju z Logitechom in znotraj evropskega projekta MIMOSA so oktobra 2006 izdali Wibree. Junija naslednje leto pa so se z Bluetooth SIG dogovorili, da bodo Wibree vključili v eno izmed prihodnjih verzij Bluetooth, ki bo temeljila na majhni porabi energije.

Leta 2009 je Bluetooth SIG oznanil Bluetooth verzijo 4.0 kot BLE, uradna verzija pa je izšla junija 2010. Prvi telefoni z implementirano tehnologijo BLE so prišli na tržišče oktobra 2011, in sicer Apple iPhone 4S. To je bila prva verzija Bluetootha, katere cilj je bila čim manjša poraba energije, saj sta do takrat obstajali le 2 verziji: klasični Bluetooth (Bluetooth BR) in Bluetooth za visoke hitrosti (Bluetooth EDR). S tem se je odcepila nova veja v razvoju. Ker se BLE od Bluetooth BR/EDR razlikuje že v fizičnem nivoju, so nastale tri vrste implementacij teh dveh tehnologij: klasični Bluetooth, Bluetooth Smart in Bluetooth Smart Ready. Razlika med njimi je, da slednji podpira obe tehnologiji, prvi dve pa le eno. Razlike so lepo vidne na sliki 2.1.

Decembra 2013 je prišla prva večja posodobitev, Bluetooth 4.1, ki je bila povratno združljiva z verzijo 4.0. Spremembe so se nanašale le na programsko opremo in ne na strojno opremo. Vključevala je tri večje izboljšave:

- izboljšano sobivanje z LTE
- boljši nadzor nad porabo energije z avtomatskim vklapljanjem in izklapljanjem naprave
- dodana podpora za več hkratnih vlog v omrežju

Decembra prihodnje leto pa je prišla še verzija 4.2, ki je prinesla nekaj izboljšav, namenjenih IoT-ju:



Slika 2.1: Primerjava med arhitekturami Bluetooth, Bluetooth Smart Ready in Bluetooth Smart [5]

- povečana hitrost pošiljanja paketov za 250% in povečana velikost paketov
- izboljšana zasebnost (LE Privacy in LE Secure Connections)
- podpora za IPv6

Junija 2016 je Bluetooth SIG predstavil Bluetooth 5, ki je ravno tako namenjen IoT-ju in še izboljša nekatere vidike BLE-ja, saj se lahko odločimo, ali bi imeli raje daljši doseg in manjšo hitrost prenosa podatkov ali ravno obratno. S to verzijo lahko pošiljamo podatke dvakrat hitreje na manjše razdalje ali na štirikrat večje razdalje ob manjši hitrosti. S povečanjem velikosti oglaševalskih paketov lahko v njih pošiljamo skoraj osemkrat več podatkov kot pri Bluetooth 4.0.

2.2 Fizična in povezovalna plast komunikacije

V tem poglavju bodo opisane lastnosti nižjih plasti protokola BLE, ki skrbijo za uspešno iskanje in povezovanje naprav ter za njihovo komunikacijo. Na začetku bodo predstavljeni frekvenčni kanali, njihova logična razdelitev ter

način skakanja med kanali, ki ga uporablja BLE za bolj učinkovito izmenjavo paketov [12]. Sledil bo opis vrst naslovov, ki jih lahko uporabljajo naprave za svojo identifikacijo, ter sam način vzpostavitve komunikacije. Tu bodo predstavljena tudi vsa logična stanja, ki jih lahko zavzema naprava v svojem delovanju. Na koncu bodo opisane še vrste paketov, ki jih lahko naprave oddajajo [13].

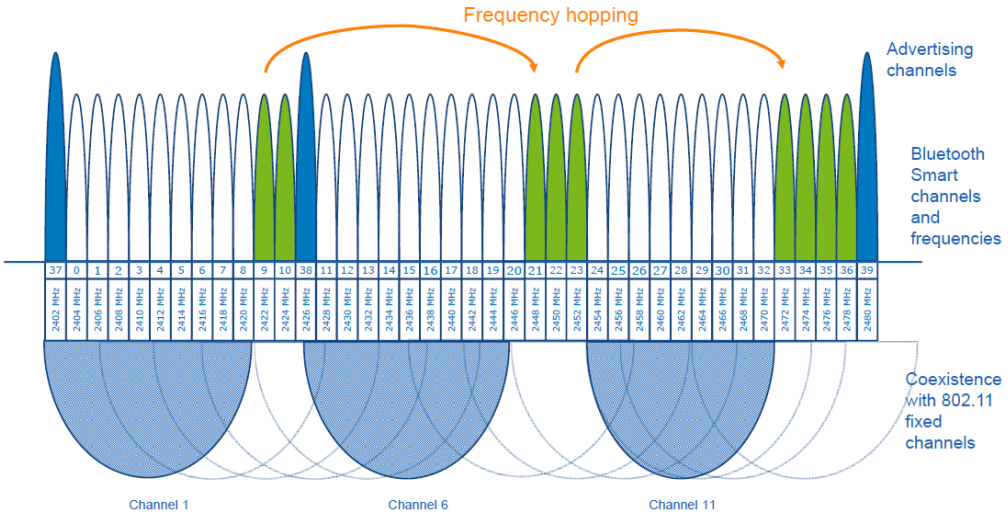
2.2.1 Kanali

BLE, ravno tako kot Bluetooth BR/EDR, uporablja frekvence ISM, ki se nahajajo na območju 2.4000 GHz do 2.4835 GHz. To območje je nadaljno razdeljeno na 40 kanalov, ki so široki po 2 MHz. Za primerjavo, Bluetooth BR/EDR je bil razdeljen na 79 kanalov širine 1 MHz. Kanali se delijo v dve skupini: kanali za oglaševanje in kanali za izmenjavo podatkov. Prvi se uporabljajo za odkrivanje naprav, vzpostavljanje povezav med napravami in razpršeno oddajanje, medtem ko se podatkovni kanali uporabljajo za obojestransko komunikacijo povezanih naprav. Oglaševalski kanali so le trije (37, 38, 39), ostali so podatkovni. Ker je oglaševalskih kanalov precej manj in predstavljajo nujen korak v povezavi, so porazdeljeni tako, da se čim manj prekrivajo z najbolj uporabljanimi kanali protokola 802.11 (Wi-Fi). Zato je kanal 37 prvi izmed kanalov na frekvenci 2.402 GHz, kanal 38 se nahaja med 802.11 kanaloma 1 in 6 na 2.426 GHz, kanal 39 pa je zadnji kanal na 2.480 GHz. Razpored oglaševalskih kanalov se lepo vidi na sliki 2.2.

Paketi se prenašajo z modulacijo GFSK (*Gaussian Frequency Shift Keying*), ki jo uporabljajo tudi Bluetooth BR/EDR ter mnogi drugi protokoli.

BLE naprave pošiljajo pakete z maksimalno hitrostjo 1 Mbit/s. Oddajna moč je odvisna od potreb oz. zahtev naprave, saj ob višjem dometu povzročimo višjo porabo baterije. Maksimalna oddajna moč znaša 10 mW, vendar se v praksi zaradi krajše razdalje med napravami ta številka zniža pod 1 mW. Domet naj bi znašal tudi nad 50 metrov na terenu brez ovir in frekvenčnih motenj.

V Bluetooth 5 je možnost prenosa podatkov z 2 Mbit/s, maksimalna moč



Slika 2.2: Skakanje med kanali [3]

oddajanja pa je omejena s 100 mW, s čimer se poveča tudi domet v primerjavi s 4.x verzijami.

2.2.2 Skakanje med kanali

Frekvenčno območje 2.4 GHz zaseda precej naprav, ki uporabljajo protokole, kot so Wi-Fi, Zigbee in Bluetooth BR/EDR. Za boljši izkoristek frekvenčnega območja in porazdelitev frekvenčne energije med kanali uporablja BLE funkcionalnost FHSS (*Frequency Hopping Spread Spectrum*), ki je način preklapljanja oz. skakanja med kanali. Vsaka naprava ob vzpostavitvi komunikacije prejme vrednost skoka (*hop*), ki kot parameter določa izbor naslednjega kanala. Formula za izračun naslednjega kanala:

$$kanal = (trenutni\_kanal + skok) * mod37$$

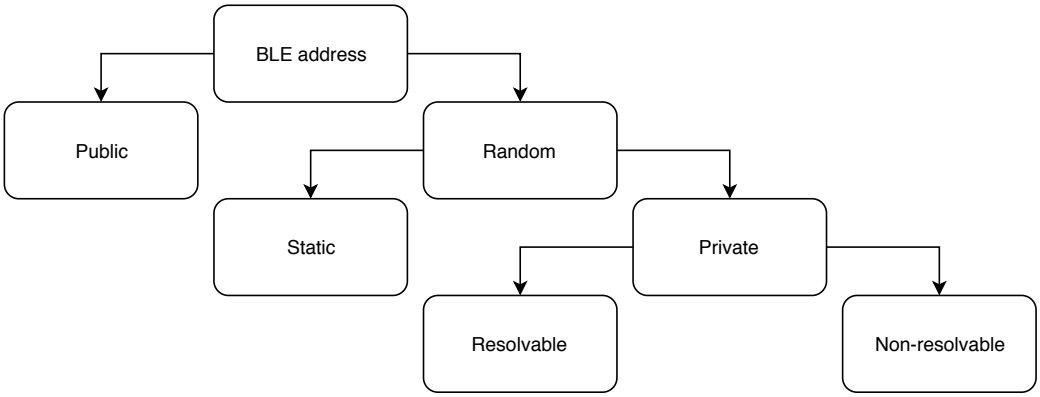
Glavne prednosti skakanje med kanali so:

- težko prestrezanje podatkov (če prisluškovalec ne pozna vrednosti skoka, ne pozna sekvence kanalov in zato težko sledi komunikaciji)
- učinkovitejša komunikacija več naprav z manj interferencami

- robustnejša povezava

Poleg standardnega FHSS, BLE uporablje tudi AFHSS (*Adaptive FHSS*), ki lahko zazna, da so nekateri kanali bolj zasedeni, in se jim v prihodnje izogiba.

2.2.3 Naslovi



Slika 2.3: Vrste naslovov

Vsaka BLE naprava ima svoj edinstven 48-bitni naslov (*Bluetooth Device Address*), s katerim se identificira med drugimi napravami. Naslovi se delijo na javne (*Public Device Address*) in naključne (*Random Device Address*). Prvi se pridobijo pri organizaciji IEEE Registration Authority, kjer le-ta določi 24 najpomembnejših bitov, ostalih 24 bitov določi proizvajalec sam. Ta naslov se nikoli ne spremeni. BLE omogoča tudi naključne naslove, ki so namenjeni predvsem anonimnosti naprav, saj jim tako težje sledimo. Ti naslovi se delijo na statične (*Static Address*) in zasebne (*Private Address*). Statični naslovi so naključno število, ki se lahko ustvari po vsakem vklopu naprave ali pa ostanejo isti celotno življensko dobo naprave. Ne morejo pa se spremeniti med delovanjem naprave. Zasebni naslovi so podobni statičnim, s to razliko, da se lahko periodično spreminjajo tudi med delovanjem naprave. Te nadaljno delimo še na razrešljive (*Resolvable Private Address*) in nerazrešljive (*Non-resolvable Private Address*). Nerazrešljivi se ne uporabljajo veliko, saj so le

naključni naslovi, ki se spremenijo po določenem času, medtem ko razrešljivi predstavljajo osnovo zasebnega komuniciranja. Izpeljemo jih lahko iz ključa IRK (*Identity Resolving Key*), ki se izmenja pri vzpostavitvi povezave med napravama. Naprava lahko svoj razrešljiv naslov zamenja tudi med že vzpostavljeno povezavo, saj jo bo druga naprava v komunikaciji prepoznala. Vse ostale naprave, ki ne poznajo ključa IRK, pa je ne morejo identificirati in slediti.

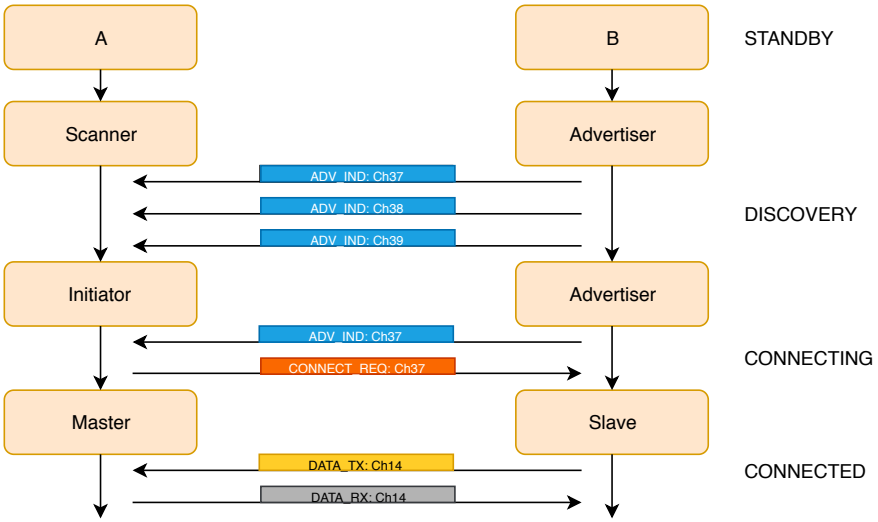
2.2.4 Vzpostavitev komunikacije

Komunikacijo delimo v enovrstno oddajanje (*Unicast*) in razpršeno oddajanje (*Broadcast*). Naprave lahko zasedejo eno izmed naslednjih parov vlog:

- oglaševalec/preiskovalec (*Advertiser/Scanner*)
- podložnik/gospodar (*Slave/Master*)

Prvi par deluje pred vzpostavitvijo povezave, drugi pa nastane po njej. Oglaševalec oznanja svojo navzočnost s pošiljenjem oglaševalskih paketov, s katerimi želi oznaniti vsem preiskovalcem svojo prisotnost in možnost povezovanja. Preiskovalci pregledujejo oglaševalske kanale in se s tem seznanjajo z vsemi napravami v bližini. Po uspešno vzpostavljeni povezavi postane gospodar tista naprava, ki je bila pobudnik povezovanja, podložnik pa tista, ki je sprejela zahtevo po povezovanju. Od tu naprej gospodar nadzoruje povezavo.

Enovrstno oddajanje ali oddajanje enemu sprejemniku se začne s fazo odkrivanja (*Discovery*), kjer ima ena izmed naprav vlogo oglaševalca, druga pa preiskovalca. V primeru na sliki 2.4 je naprava A preiskovalec in naprava B oglaševalec. Oglaševalec oddaja oglaševalske pakete, na katere se lahko preiskovalec odzove. Ko se to zgodi, se začne faza povezovanja (*Connecting*), kjer preiskovalec postane pobudnik (*Initiator*) in pošlje paket CONNECT_REQ (2.1) namenjen oglaševalcu. V zadnjem koraku oglaševalec sprejme vzpostavitev povezave in s tem napravi preideta v stanje vzpostavljene povezave (*Connection*). Oglaševalec postane podložnik, pobudnik pa gospodar.

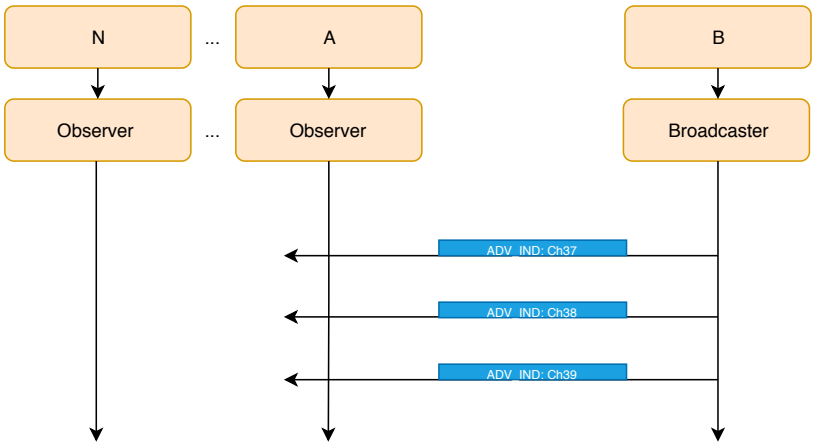


Slika 2.4: Enovrstna povezava

Pri razpršenem oddajanju (slika 2.5) obstaja še en par vlog, ki je podoben paru oglaševalec/preiskovalec:

- oddajnik/opazovalec (*Broadcaster/Observer*)

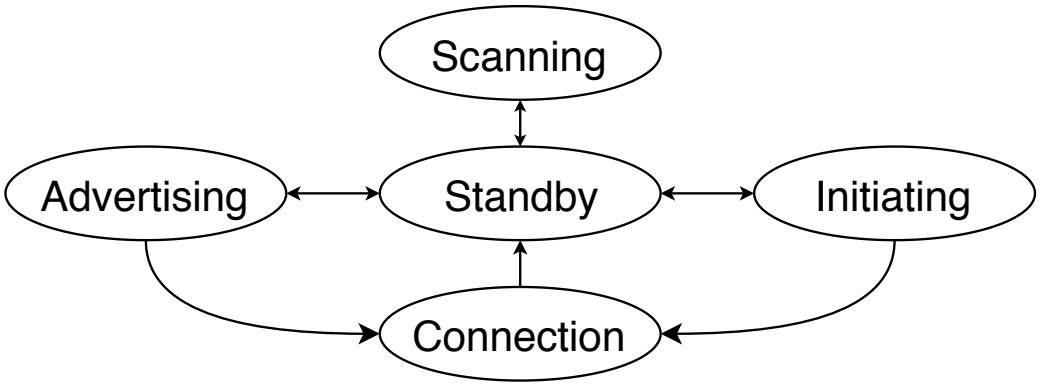
Oddajnik periodično oddaja oglaševalske pakete z določenimi podatki, ki jih želi deliti z opazovalci. Opazovalcev je lahko več in na sprejete pakete ne odgovarjajo.



Slika 2.5: Razpršena povezava

Delovanje BLE naprav lahko predstavimo z avtomatom stanj, kjer naprava zavzema eno izmed sledečih 5 stanj:

- stanje pripravljenosti (*Standby*) - naprava ne sprejema ali pošilja paketov
- oglaševanje (*Advertising*) - naprava oddaja oglaševalske pakete in posluša njihove morebitne odgovore
- preiskovanje (*Scanning*) - naprava posluša, če katera izmed naprav oddaja oglaševalske pakete
- vzpostavljanje povezave (*Initiating*) - naprava posluša in odgovarja na oglaševalske pakete določene naprave, s katero hoče vzpostaviti povezavo
- stanje vzpostavljene povezave (*Connection*) - če je naprava prešla v to stanje iz stanja oglaševanja, prevzame vlogo podložnika, če pa je prišla v to stanje iz stanja vzpostavljanja povezave, postane gospodar



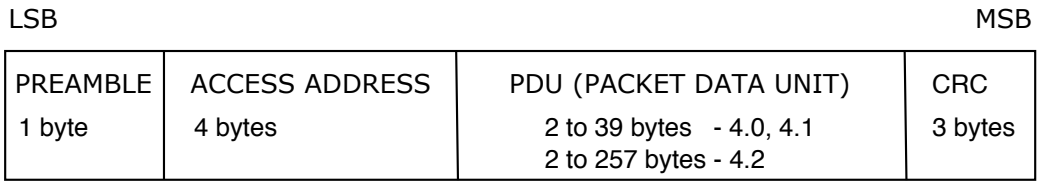
Slika 2.6: Avtomat stanj

2.2.5 Paketi

BLE za oglaševalske in podatkovne pakete uporablja isti format, s čimer stremi k enostavnosti. BR/EDR verziji sta imeli več vrst paketov. Vsak

paket je sestavljen iz štirih polj: uvodne sinhronizacije, dostopnega naslova, podatkovne enote in kontrolne vsote. Z verzijo Bluetooth 4.2 se je povečala velikost podatkovnega okna paketov na podatkovnih kanalih iz 39 bajtov na 257 bajtov, s čimer se je povečala prepustnost in zmanjšal transakcijski čas.

Uvodna sinhronizacija (*Preamble*) je 1 bajt dolga izmenjujoča sekvenca enk in ničel: 10101010 ali 01010101. Sprejemnik s to sekvenco sinhronizira svoj radijski sprejemnik za boljši sprejem signala (*Automatic Gain Control*) ter s pošiljateljem uskladi trajanje simbolov.

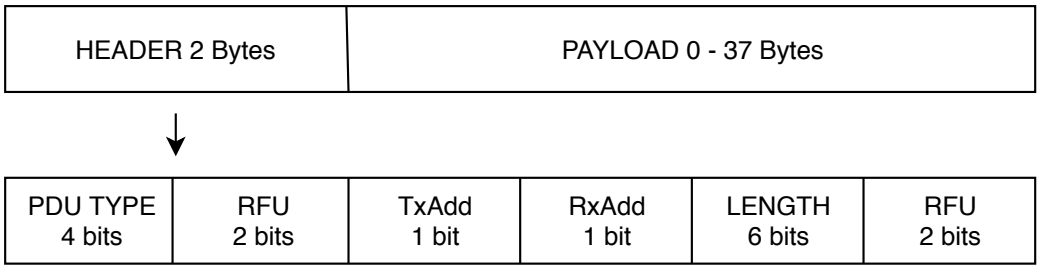


Slika 2.7: Format BLE paketa

Dostopni naslov (*Access Address*) je 4 bajte dolg naslov, s katerim naprave vedo, v kateri povezavi sodelujejo. Vsaka vzpostavljena povezava med dvema napravama ima svoj edinstven naslov, le paketi oddajnikov (pri razpršenih povezavah) imajo fiksno vrednost 0x8E89BED6.

Kontrolna vsota CRC (*Cyclic Redundancy Check*) je dolga 3 bajte in je izračunana nad podatkovno enoto. Služi preprečevanju napak nad podatki, s čimer dosežemo večjo robustnost povezave.

Podatkovna enota PDU (*Packet Data Unit*) je dolga med 2 in 39 bajtov, pri verziji 4.2 za pakete na podatkovnih kanalih do 257 bajtov. Ločimo dve vrsti pokatkovnih enot, in sicer glede na to, na katerem kanalu pošiljamo paket – PDU na oglaševalskih oz. podatkovnih kanalih.



Slika 2.8: PDU paketa na oglaševalskem kanalu

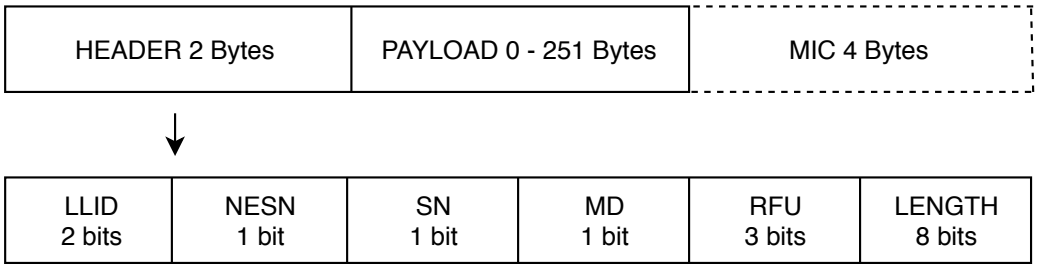
PDU na oglaševalskih kanalih je sestavljen iz glave in podatkov. Podatki zavzemajo največ 37 bajtov, odvisno od vrednosti dolžine v glavi paketa. Glava je dolga 2 bajta in vsebuje:

- Tip PDU-ja – velikosti 4 bite in pove vrsto paketa. Ločimo oglaševalske, preiskovalne in vzpostavljalske pakete (v tabeli 2.1 si sledijo s predponami „ADV“, „SCAN“ in „CONNECT“).
  - Oglaševalske pakete oddaja naprava v stanju oglaševanja, sprejemajo jih naprave v stanju preiskovanja in vzpostavljanja povezave.
  - Preiskovalne pakete uporabljata preiskovalec in oglaševalec; prvi, da zaprosi oglaševalca za podatke, drugi pa za odgovor preiskovalcu na zahtevo.
  - Tretja vrsta so vzpostavljalski PDU-ji, ki jih uporabljajo naprave, ki želijo vzpostaviti povezavo z oglaševalcem.
- TxAdd in RxAdd – vsak je dolg po 1 bit; povesta, ali je naslov v podatkih javen ali naključen.
- Dolžina – velikosti 6 bitov; pove nam dolžino podatkov v bajtih.
- RFU (*Reserved for Future Use*) – dve polji po 2 bita sta rezervirani za možno uporabo v prihodnosti.

PDU Name	Event Type	Sender Link Layer State	Receiver Link Layer State
ADV_IND	Connectable Undirected	Advertising	Scanning/Initiating
ADV_DIRECT_IND	Connectable Directed	Advertising	Scanning/Initiating
ADV_NONCONN_IND	Non-Connectable Directed	Advertising	Scanning/Initiating
ADV_SCAN_IND	Scannable Undirected	Advertising	Scanning/Initiating
SCAN_REQ	Scanner requesting data from Advertiser	Scanning	Advertising
SCAN_RES	Advertiser responding to the request from Scanner	Advertising	Scanning
CONNECT_REQ	Initiator requesting to connect to Advertiser	Initiating	Advertising

Tabela 2.1: PDU-ji na oglaševalskih kanalih

PDU za podatkovne kanale je, podobno kot PDU za oglaševalske kanale, sestavljen iz glave in podatkov, ima pa še možnost dodatnega polja MIC (*Message Integrity Check*) ob uporabi kriptirane povezave. MIC je dolg 4 bajte in služi avtentikaciji podatkov, ki so dolgi med 0 in 251 bajtov.



Slika 2.9: PDU paketa na podatkovnem kanalu

Glava PDU-ja za podatkovne kanale je dolga 2 bajta in vsebuje:

- LLID (*Link Layer ID*) – 2 bita dolgo polje označuje, za katero vrsto PDU-ja gre:
  - Podatkovni se uporablja za pošiljanje L2CAP podatkov.
  - Nadzorni za izmenjavo kontrolnih informacij med napravami.
- SN (*Sequence Number*) – 1 bit, ki skrbi za enostaven mehanizem potrjevanja paketov.

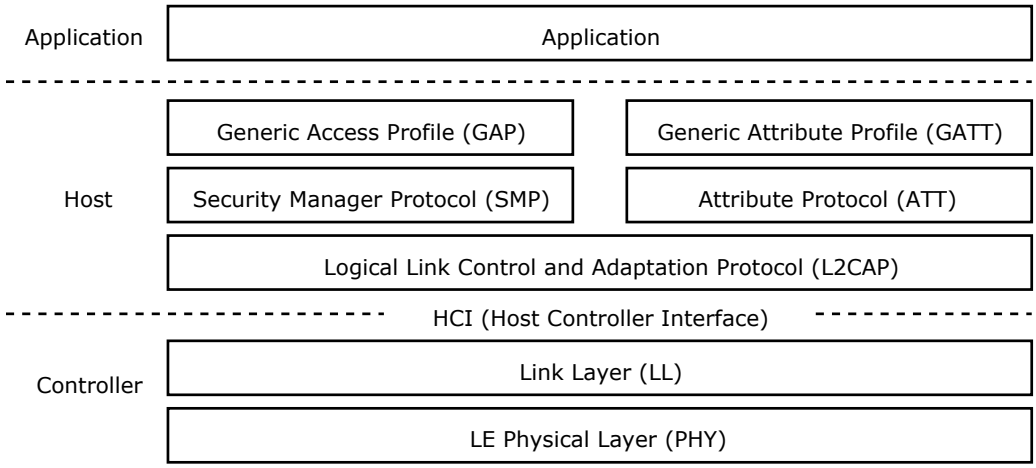
- NESN (*Next Expected Sequence Number*) – 1 bit, ki skupaj s SN skrbi za urejen tok paketov.
- MD (*More Data*) – 1 bit, ki pove, da ima naprava še podatke za poslati in mora zato povezava ostati odprta.
- RFU (*Reserved for Future Use*) – 3 biti, rezervirano za možno uporabo v prihodnosti.
- dolžina – 8 bitov sporoča dolžino podatkov in MIC-a v bajtih.

2.3 Višje plasti komunikacije

Vsaka Bluetooth naprava je sestavljena iz treh sestavnih delov:

- Aplikacija (*Application*) - program, s katerim upravlja uporabnik
- Gostitelj (*Host*) - zgornje plasti (programski del) sklada Bluetooth
- Kontroler (*Controller*) - spodnje plasti (strojni del) sklada Bluetooth

V tem poglavju bomo bolj podrobno spoznali protokole, module in profile, ki se po hiererhiji nahajajo nad kontrolerjem.



Slika 2.10: BLE protokolni sklad

2.3.1 Vmesnik Host Controller Interface (HCI)

HCI je univerzalni vmesnik, ki loči strojni del (kontroler) od programskega dela (gostitelj) sklada Bluetooth. Vsi ukazi in funkcijski klici, ki jih izvedejo višje ležeče plasti, se izvedejo v modulu HCI, ki edini komunicira s kontrolerjem. Ravno tako se vsi prejeti podatki in dogodki sporočajo skozi HCI naprej v L2CAP.

2.3.2 Modul Logical Link Control and Adaptation Protocol (L2CAP)

L2CAP skrbi za multipleksiranje protokolov višje ležečih plasti in enkapsulacijo njihovih podatkov v standardne BLE pakete. Večje bloke podatkov razdeli na manjše kose ter poskrbi, da so fragmentirani paketi pravilno dostavljeni. Ravno tako skrbi za tok podatkov v drugo smer z defragmentacijo in demultipleksiranjem. L2CAP deluje s pomočjo kanalov, ki so lahko fiksni ali dinamični. Ločijo se s pomočjo CID-ov (*Channel ID*).

2.3.3 Modul Attribute Protocol (ATT)

ATT je protokol, ki definira, kako se bodo podatki prenašali med napravami. Skrbi za odkrivanje, branje in pisanje atributov, ki so sestavni bloki protokola. Ti vsebujejo:

- Tip atributa (*Attribute Type*) – določa kakšen tip podatkov vsebuje atribut.
- Identifikator atributa (*Attribute Handle*) – enolična številka atributa med 0x0000 in 0xFFFF.
- Pravice atributa (*Attribute Permissions*) – možne pravice: pisanje, branje, pisanje in branje.

2.3.4 Modul Security Manager (SM)

SM skrbi za upravljanje z varnostnimi mehanizmi. Zadolžen je za varno komunikacijo naprav čez kriptirano povezavo, identifikacijo naprav in upravljanje z naključnimi naslovi. Za vzpostavitev varne povezave pozna SM tri procedure: uparjanje (*pairing*), povezovanje (*bonding*) in ponovno vzpostavljanje povezave (*encryption re-establishment*).

Uparjanje se začne z avtentikacijo obeh naprav, nato pa se vzpostavi kriptirana povezava z uporabo ključa STK (*Short Term Key*). Ključ STK se lahko generira na tri načine:

- *Passkey Display* – Na eni ali obeh napravah je potrebno vpisati geslo preko uporabniškega vmesnika. Ta način je zaščiten pred napadom s posrednikom MITM (*Man In The Middle*).
- *Just Works* – Če vsaj ena naprava ne omogoča izmenjave gesel prek uporabniškega vmesnika, si napravi izmenjata potrebne informacije za generiranje ključa STK v nekriptiranih paketih. Ta način omogoča napad MITM.
- *Out of Band* je metoda izmenjave gesel prek drugih načinov komunikacije, npr. NFC. Ta način je zaščiten pred napadom MITM.

Pri povezovanju se ustvari in izmenja LTK (*Long Term Key*), ki se shrani v trajni spomin naprave. Z njim se lahko napravi v prihodnosti hitreje povežeta brez potrebe po novem povezovanju.

Ponovno vzpostavljanje povezave je postopek, ki definira kako vzpostaviti povezavo med napravama brez ponovne potrebe po uparjanju ali povezovanju, če sta si napravi že predhodno izmenjali enkripcijske ključe.

2.3.5 Profil Generic Attribute Profile (GATT)

Profil GATT definira, kako so podatki organizirani oz. kakšna hierarhija obstaja med atributi. GATT je tesno povezan z ATT-om *(2.3.3)*, saj skupaj

skrbita za pravilno izmenjavo podatkov med napravami. Prvi definira format podatkov, drugi pa skrbi za njihovo zanesljivo dostavo.

GATT definira dve vlogi za svoje delovanje: strežnik in odjemalec. Strežnik ima podatke, do katerih želi odjemalec dostopati. Ti podatki so atributi, ki so združeni v servise, ki vsebujejo karakteristike. Servisi so definirani v profilih, ki so predefinirane zbirke servisov, ki so jih določile Bluetooth SIG ali proizvajalci naprav. Karakteristike so logične enote, ki vsebujejo določene podatke uporabnika ter metapodatke. Metapodatki služijo opisu podatkov v karakteristikah, npr. lastnosti podatkov, enote ...

2.3.6 Profil Generic Access Profile (GAP)

Profil GAP je zadolžen za upravljenje s splošnim dostopom oz. določa, kako se naprave med seboj sporazumevajo. Nahaja se na vrhu protokolnega sklada BLE, kjer definira celotno topologijo delovanja. Njegove glavne naloge so iskanje naprav, povezovanje naprav in vzpostavljanje varnosti. Poleg tega GAP definira tudi nekatere koncepte, ki smo jih že opisali v poglavju *Vzpostavitev komunikacije (2.2.4)*, npr. možne vloge v komunikaciji in interakcije med njimi, prehodi med stanji naprav, varnost ...

Poglavje 3

Izogibanje motnjam

Kot je bilo že opisano v poglavjih *Kanali (2.2.1)* in *Skakanje med kanali (2.2.2)*, BLE za svojo komunikacijo uporablja 40 kanalov na frekvenčnem območju 2.4 GHz. Oglaševanju so namenjeni trije kanali, ostalih 37 pa skrbi za izmenjavo podatkovnih paketov. Oglaševalski kanali imajo dodeljene kanale 37, 38 in 39, vendar so porazdeljeni na takšne frekvence, da se čim bolj izogibajo najbolj popularnim kanalom protokola Wi-Fi. Zato imajo fizične pozicije 0, 12 in 39.

Od Bluetooth verzije 1.2, ki je izšla novembra 2003, Bluetooth naprave uporabljajo funkcionalnost AFH, ki lahko nekatere kanale označi kot zasedene in se jim v prihodnje izogiba. To lahko počne vse, dokler ne ostaneta vsaj dva prosta kanala. V Bluetooth verziji 5 je bil dodan nov algoritem za skakanje med kanali, ki vnaša naključnost v skoke. Vsak nov kanal se izračuna iz trenutnega števca dogodkov ter iz dostopnega naslova. Kateri algoritem bosta napravi uporabljali, se določi ob vzpostavitvi povezave.

3.1 Moč signala

RSSI (*Received Signal Strength Indicator*) nam pove moč prejetega signala. S tem si lahko predstavljamo, kako dobro lahko napravi komunicirata. Moč signala se meri v dBm (decibel-milivat), kjer višje število pomeni močnejši

signal. BLE naprave lahko oddajajo z največjo močjo 10 mW, iz česar lahko s formulo izračunamo moč signala v dBm:

$$P_{(dBm)} = 10 \log_{10} \frac{P_{(mW)}}{1mW}$$

Največja oddajna moč signala je torej 10 dBm. Za primerjavo, pri protokolu Wi-Fi lahko naprave oddajajo z močjo 100 mW, s čimer je največja moč signala 20 dBm. Pri analizi moči signala pa moramo upoštevati tudi manjšanje moči z oddaljenostjo:

$$P_r = P_t G_t G_r (\frac{\lambda}{4\pi d})^2$$

Pr = sprejemna moč
Pt = moč oddajnika
Gt = ojačanje oddajne antene
Gr = ojačanje sprejemne antene
d = oddaljenost med sprejemnikom in oddajnikom

Moč signala se manjša s kvadratom razdalje, kjer nismo niti upoštevali zunanjih motenj, kot so odboji, lomljenja, drugi signali ... V realnih situacijah je sprejeti signal veliko manjši od oddanega, zato mnogi komunikacijski protokoli uporabljajo dodatne načine za zagotavljanje boljšega prenosa podatkov: boljše kodiranje, kontrolne vsote, več anten ...

3.2 Časi med paketi

Prenos na fizičnem nivoju je razdeljen na časovne enote, ki jim pravimo tudi dogodki. Ločimo oglaševalske in povezavne dogodke, kjer se prvi uporabljajo na oglaševalskih kanalih, drugi pa na podatkovnih.

Oglaševalski dogodki se vedno začnejo z oglaševalskim paketom oglaševalca, na katerega lahko preiskovalec odgovori. Če se to zgodi, oglaševalec odgovori preiskovalcu, vse to pa se zgodi znotraj istega oglaševalskega dogodka. Nato oglaševalec nadaljuje s svojim delom na naslednjem kanalu. Oglaševalec

pošilja pakete v skupinah po tri – na vsakem kanalu enega, nato pa počaka nekaj več časa pred naslednjim ciklom paketov. To lahko opazimo na sliki 3.1, kjer so časi med paketi na kanalih 37 veliko večji kot med kanaloma 38 in 39. Tak način delovanja je uporaben za varčevanje z baterijo, saj je oddajnik veliko časa nedejaven, ko pa je, deluje le krajše časovno obdobje. Čas med začetkom dveh zaporednih oglaševalskih dogodkov je definiran kot:

$$advEvent = advInterval + advDelay$$

advInterval = med 20 ms in 10.24 s
advDelay = naključen čas med 0 in 10 ms

Povezavni dogodki se uporabljajo med povezanima napravama, ko sta vlogi gospodarja in podložnika že določeni. Ob začetku povezavnega dogodka (*Anchor Point*), gospodar prvi pošlje paket podložniku, za tem pa si izmenjujoče pošiljata pakete. Podložnik vedno odgovori na gospodarjev paket, medtem ko gospodarju ni vedno potrebno odgovoriti podložniku. Vsi paketi znotraj povezavnega dogodka se izmenjajo na istem kanalu, skakanje med kanali pa se zgodi na začetku takega povezavnega dogodka. Parametri, ki določajo povezavne dogodke:

- connInterval – interval med zaporednima povezavnima dogodkoma, ki je večkratnik 1.25 ms ter omejen med 7.5 ms in 4s
- connSlaveLatency – število zaporednih povezavnih dogodkov, ki jih lahko podložnik preskoči, preden mora poslušati gospodarja
- connSupervisionTimeout – če v tem času ne pride noben paket do naprave, se predpostavlja, da je bila povezava izgubljena
- transmitWindowOffset – časovni zamik med paketom CONNECT_REQ in prenosnim oknom
- transmitWindowSize – velikost prenosnega okna

No.	Time	Source	Destination	Protocol	Length	Channel	Time delta from previous captured frame	Info
36	25.638262	Raspberr_04:48:55	Broadcast	LE LL	62	37	1.306673000	ADV_IND
37	25.639266	Raspberr_04:48:55	Broadcast	LE LL	62	38	0.001004000	ADV_IND
38	25.639992	Raspberr_04:48:55	Broadcast	LE LL	62	39	0.000726000	ADV_IND
39	26.946301	Raspberr_04:48:55	Broadcast	LE LL	62	37	1.306309000	ADV_IND
40	26.947326	Raspberr_04:48:55	Broadcast	LE LL	62	38	0.001025000	ADV_IND
41	26.948061	Raspberr_04:48:55	Broadcast	LE LL	62	39	0.000735000	ADV_IND
42	28.153634	Raspberr_04:48:55	Broadcast	LE LL	62	37	1.205573000	ADV_IND
43	28.154528	Raspberr_04:48:55	Broadcast	LE LL	62	38	0.000894000	ADV_IND
44	28.155361	Raspberr_04:48:55	Broadcast	LE LL	62	39	0.000833000	ADV_IND
45	29.461330	Raspberr_04:48:55	Broadcast	LE LL	62	37	1.305969000	ADV_IND
46	29.462214	Raspberr_04:48:55	Broadcast	LE LL	62	38	0.000884000	ADV_IND
47	29.462987	Raspberr_04:48:55	Broadcast	LE LL	62	39	0.000773000	ADV_IND
48	30.768902	Raspberr_04:48:55	Broadcast	LE LL	62	37	1.305915000	ADV_IND
49	30.769830	Raspberr_04:48:55	Broadcast	LE LL	62	38	0.000928000	ADV_IND
50	30.770645	Raspberr_04:48:55	Broadcast	LE LL	62	39	0.000815000	ADV_IND

Slika 3.1: Primer časov med oglaševalskimi paketi (predzadnji stolpec)

3.3 Paketi namenjeni izogibanju motnjam

Za sinhrono menjavo kanalov se morata napravi najprej dogovoriti o nekaterih parametrih, preko katerih časovno uskladita kdaj in kje pričakovati nov paket. Te parametre si izmenjata s paketom CONNECT_REQ, za vse nadaljne spremembe in popravke povezave pa ima BLE namenjenih nekaj posebnih paketov, ki so del nadzornega protokola povezovalne plasti LLCP (*Link Layer Control Protocol*).

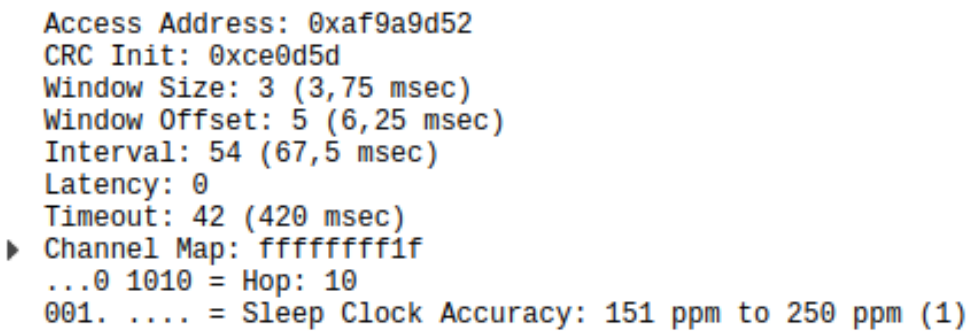
LLCP nadzira in upravlja s povezavo med dvema napravama. Definiranih ima več postopkov, preko katerih lahko napravi posodabljata parametre povezave, začneta z enkripcijo podatkov, podaljšata dolžino PDU-jev ... Postopki se izvajajo sekvenčno, kar pomeni, da se lahko izvaja le en naenkrat. Izjema je postopek za prekinitev povezave.

V nadaljevanju bodo predstavljeni nekateri paketi, ki so namenjeni sporazumevanju med napravama o stanju povezave in spreminjanju parametrov povezave. Ti paketi so: CONNECT_REQ, LL_CHANNEL_MAP_REQ, LL_CONNECTION_UPDATE_REQ ter LL_CONNECTION_PARAM_REQ.

Ob vzpostavljanju povezave pošlje preiskovalec oglaševalcu paket CONNECT_REQ, ki poleg obeh naslovov naprav vsebuje tudi podatkovno polje povezovalne plasti (*Link Layer Data*), velikosti 22 bajtov. V tem polju se nahajajo parametri o povezavi:

- *Access Address* – dostopni naslov
- *CRC Init* – inicializacijska vrednost za računanje CRC

- *Window Size* – velikost oddajnega okna
- *Window Offset* – zamik oddajnega okna
- *Interval* – interval med zaporednima povezavnima dogodkoma
- *Latency* – latenca podložnika
- *Timeout* – koliko časa zdrži povezava brez paketov
- *Channel Map* – bitna slika prostih in uporabljenih kanalov
- *Hop* – skok za AFH
- *Sleep Clock Accuracy* – spodnja meja natančnosti gospodarjeve ure



Slika 3.2: Primer polja Link Layer Data v paketu CONNECT_REQ

Za posodabljanje parametrov povezave ima BLE namenjena dva paketa: LL_CONNECTION_UPDATE_REQ in LL_CONNECTION_PARAM_REQ. Prvega lahko pošlje samo gospodar ter posodobi interval, latenco in timeout. LL_CONNECTION_PARAM_REQ lahko pošljeta gospodar ali podložnik in s tem zaprosita za posodobitev istih treh parametrov povezave. Če se sprejemna naprava ne strinja s predlaganimi spremembami parametrov, lahko pošlje isti paket s svojimi vrednostmi parametrov ali pa zavrne spremembe s paketom LL_REJECT_IND_EXT.

```
Control Opcode: LL_CHANNEL_MAP_REQ (0x01)
▾ Channel Map: ffff3f0018
    .... ...1 = RF Channel 1 (2404 MHz - Data - 0): True
    .... ..1. = RF Channel 2 (2406 MHz - Data - 1): True
    .... .1.. = RF Channel 3 (2408 MHz - Data - 2): True
    .... 1... = RF Channel 4 (2410 MHz - Data - 3): True
    ...1 .... = RF Channel 5 (2412 MHz - Data - 4): True
    ..1. .... = RF Channel 6 (2414 MHz - Data - 5): True
    .1.. .... = RF Channel 7 (2416 MHz - Data - 6): True
    1... .... = RF Channel 8 (2418 MHz - Data - 7): True
    .... ...1 = RF Channel 9 (2420 MHz - Data - 8): True
    .... ..1. = RF Channel 10 (2422 MHz - Data - 9): True
    .... .1.. = RF Channel 11 (2424 MHz - Data - 10): True
    .... 1... = RF Channel 13 (2428 MHz - Data - 11): True
    ...1 .... = RF Channel 14 (2430 MHz - Data - 12): True
    ..1. .... = RF Channel 15 (2432 MHz - Data - 13): True
    .1.. .... = RF Channel 16 (2434 MHz - Data - 14): True
    1... .... = RF Channel 17 (2436 MHz - Data - 15): True
    .... ...1 = RF Channel 18 (2438 MHz - Data - 16): True
    .... ..1. = RF Channel 19 (2440 MHz - Data - 17): True
    .... .1.. = RF Channel 20 (2442 MHz - Data - 18): True
    .... 1... = RF Channel 21 (2444 MHz - Data - 19): True
    ...1 .... = RF Channel 22 (2446 MHz - Data - 20): True
    ..1. .... = RF Channel 23 (2448 MHz - Data - 21): True
    .0.. .... = RF Channel 24 (2450 MHz - Data - 22): False
    0... .... = RF Channel 25 (2452 MHz - Data - 23): False
    .... ...0 = RF Channel 26 (2454 MHz - Data - 24): False
    .... ..0. = RF Channel 27 (2456 MHz - Data - 25): False
    .... .0.. = RF Channel 28 (2458 MHz - Data - 26): False
    .... 0... = RF Channel 29 (2460 MHz - Data - 27): False
    ...0 .... = RF Channel 30 (2462 MHz - Data - 28): False
    ..0. .... = RF Channel 31 (2464 MHz - Data - 29): False
    .0.. .... = RF Channel 32 (2466 MHz - Data - 30): False
    0... .... = RF Channel 33 (2468 MHz - Data - 31): False
    .... ...0 = RF Channel 34 (2470 MHz - Data - 32): False
    .... ..0. = RF Channel 35 (2472 MHz - Data - 33): False
    .... .0.. = RF Channel 36 (2474 MHz - Data - 34): False
    .... 1... = RF Channel 37 (2476 MHz - Data - 35): True
    ...1 .... = RF Channel 38 (2478 MHz - Data - 36): True
    ..0. .... = RF Channel 0 (2402 MHz - Reserved for Advertising - 37): False
    .0.. .... = RF Channel 12 (2426 MHz - Reserved for Advertising - 38): False
    0... .... = RF Channel 39 (2480 MHz - Reserved for Advertising - 39): False
```

Slika 3.3: Primer bitne slike prostih in zasedenih kanalov

Paket LL_CHANNEL_MAP_REQ uporablja LLCP postopek za posodobitev načrta kanalov. Vsebuje seznam vseh 40 kanalov ter pri vsakem pripadajočo vrednost, ki določa, ali je kanal zaseden ali prost. Ta postopek lahko začne samo gospodar. Na sliki 3.3 je primer paketa, kjer so kanali od 24 do 36 označeni kot zasedeni. Za zajem tega paketa smo na Wi-Fi kanalu 11 vzpostavili dostopno točko, kar je povzročilo motnje v komunikaciji med povezanima napravama. Kmalu zatem je gospodar poslal ta paket podložniku, po čemer sta se začela izogibati zasedenim kanalom.

Poglavje 4

Uporabljena orodja

V tem poglavju bodo predstavljena orodja, ki smo jih uporabili pri poskusih. Glede strojne opreme smo uporabili dva Raspberry Pi-ja, dva prenosna računalnika, nekaj Wi-Fi USB adapterjev ter Bluefruit LE Sniffer. Od programske opreme smo uporabili orodja iz paketa BlueZ, Nping za generiranje UDP prometa ter analizator omrežnih paketov Wireshark.

4.1 Bluetooth sklad BlueZ

BlueZ je uradni Bluetooth sklad za operacijski sistem Linux. S svojo modularno sestavo ponuja podporo glavnim protokolom in plastem Bluetootha. Uporabili smo verzijo 5.43, kjer so nekatere funkcije in ukazi za delovanje z BLE še v fazi razvoja, zato smo jih morali najprej aktivirati v konfiguracijski datoteki. Izmed orodij smo uporabili hcitool za iskanje naprav ter gatttool za komunikacijo med napravama. Več informacij o BlueZ je dostopnih na njihovi uradni spletni strani [4].

4.2 Mikroračunalnik Raspberry Pi

Raspberry Pi je majhen računalnik, ki je bil razvit za namene poučevanja osnov računalništva v šolah in v državah v razvoju. Ker je mikroračunalnik

	Model 3B	Model 3B+
Processor	Broadcom BCM2837 SoC @ 1.2Ghz	Broadcom BCM2837 SoC @ 1.4GHz
Ethernet	100Base	1000Base
Wi-Fi	802.11b/g/n	Dual-Band 802.11ac
PoE	No	Yes
RAM	1GB LPDDR2	1GB LPDDR2
Ports	DSI x1, RCA x1, HDMI x1. USB x4	DSI x1, RCA x1, HDMI x1, USB x4
Bluetooth	4.1	4.2

Tabela 4.1: Primerjava Raspberry Pi model 3B in Raspberry Pi model 3B+

poceni, je postal zelo popularen med tehnološkimi navdušenci za manjše projekte, ki jih lahko vsak ustvari doma. Prvi modeli so izšli v letu 2012, od takrat pa se je razvilo že več novih generacij mikroračunalnikov, ki so doprinesle bolj zmogljivo ter razširjeno strojno opremo.

Pri tem projektu smo uporabili dva Raspberry Pija, Raspberry Pi 3 model B in Raspberry Pi 3 model B+. Razlike med njima so navedene v tabeli 4.1. Več informacij o Raspberry Pi-jih je dostopnih na njihovi uradni spletni strani [9].

4.3 Vohljač Adafruit BLE Sniffer

Bluefruit LE Sniffer je orodje, ki lahko v skoraj realnem času prikaže BLE pakete, ki jih zajame v zraku. Uporaben je za odpravljanje programskih napak in vpogled v vsebnost BLE paketov. Temelji na izdelku Bluefruit LE Friend, ima pa namensko sistemsko programsko opremo, ki ga spremeni v enostaven BLE vohljač. Proizvaja ga podjetje Adafruit, ki se ukvarja s proizvodnjo elektronskih izdelkov, elektronskih komponent in različnih učnih pripomočkov. Izdelek dobro sodeluje z Wiresharkom, odprtokodnim programom za prikaz paketov, ki olajša vizualizacijo polj v paketih.

Vohljač temelji na ploščici nRF51822, ki jo je oblikovalo norveško podjetje Nordic Semiconductor. Nameščeno ima sistemsko programsko opremo verzije v2, ki se od verzije v1 razlikuje v boljši podpori za Wireshark in različne

operacijske sisteme. Strojna oprema je verzije v3, ki je bolj cenovno ugodna od prejšnjih dveh verzij. Zasluga za to gre predvsem USB čipu CP2104 in odstranjenemu SWD konektorju, ki je bil namenjen reprogramiranju ploščice. Sprejemnik deluje za frekvenčno območje 2.4 GHz z občutljivostjo do -93 dBm. Procesor je 32-bitni ARM Cortex-M0 s 256/128 KB bliskovnega pomnilnika in 32/16 KB delovnega pomnilnika. Oddajna moč se giblje od +4 dBm do -20 dBm v korakih po 4 dBm. Ploščica omogoča tudi nadgradnjo sistemsko programske opreme preko zraka OTA-DFU (*Over The Air Device Firmware Upgrade*).

Na ploščici se nahajajo 4 LED diode:

- modra za status povezave (*connection status*)
- rdeča za vrsto povezave (*mode status*)
- rumena za sprejemanje podatkov (RX)
- rumena za oddajanje podatkov (TX)

Ploščica ima nameščeno tudi stikalo, s katerim določimo, v katerem načinu naj deluje. Prvi način ima na ploščici oznako „DAT“ in je namenjen navadni komunikaciji. Drugi način ima oznako „CMD“ in pretvori vohljač v ukazni način, v katerem lahko ustvarimo naše lastne GATT servise in karakteristike, spremenimo ime naprave ... Na ploščici se nahaja tudi gumb z oznako „DFU“, ki služi nadgradnji sistemsko programske opreme.

Vohljač lahko BLE pakete zajema na dva načina. Vedno se najprej začne prvi način, kjer vohljač sledi vsem oglaševalskim kanalom ter poskuša sprejeti čim več paketov iz čim več naprav. Ko ima podatke naprav, lahko aktivira drugi način delovanja, kjer sledi vsem paketom, ki jih je poslala oz. so namenjeni neki napravi. Z drugim načinom lahko sledimo povezavi med dvema napravama in zaznamo vse pakete, ki si jih napravi izmenjujeta. Več informacij o Bluefruit LE Snifferju je dostopnih na spletni strani [2].

4.4 Analizator omrežnih paketov Wireshark

Wireshark je en izmed najbolj razširjenih in uporabljanih analizatorjev omrežnih protokolov. Omogoča zajemanje in pregled veliko različnih protokolov, nad katerimi omogoča uporabo filtrov in grafičnih prikazov. Podatke lahko zajema iz več naprav hkrati ter v živo prikazuje zajete pakete.

Z namestitvijo potrebne programske opreme, lahko Wireshark zajema podatke tudi iz Adafruit BLE Snifferja. Zajema lahko vse oglaševalske pakete, ki jih ujame, lahko pa določimo tudi filtriranje oz. zajemanje iz točno določene naprave. Ob uporabi druge možnosti, lahko sledimo tudi povezavi med dvema napravama, ki sta se uspešno povezali. Več informacij o Wiresharku je dostopnih na uradni spletni strani [10].

4.5 Generiranje Wi-Fi motenj

Za generiranje Wi-Fi prometa smo uporabili več orodij, s katerimi smo poskušali vpeljati čim večje motnje v testno okolje. Oprema, ki smo jo uporabili, je sledeča: več Wi-Fi USB adapterjev D-Link DWL-G122 [6] ter orodje Nping, nameščeno na operacijskem sistemu Linux Mint. Wi-Fi USB adapterji podpirajo protokol 802.11g, ki omogoča hitrosti do 54 Mbps. Za generiranje motenj na enem Wi-Fi kanalu smo uporabili USB Wi-Fi adapter, na katerem smo odprli dostopno točko, nato pa se z drugim USB Wi-Fi adapterjem povezali nanjo in generirali UDP promet s pomočjo programa Nping. Nping [8] je odprtokodno orodje, ki omogoča generiranje omrežnih paketov za veliko različnih protokolov ter njihovo časovno analizo. Poskušali smo ustvariti čim več UDP paketov čim hitreje, zato smo generirali 10.000 paketov na sekundo. Paketi so bili velikosti 42 bajtov. Wi-Fi USB adapterji niso mogli doseči tako visoke številke, so pa delovali po svojih maksimalnih zmožnostih. Za generiranje motenj na treh Wi-Fi kanalih smo imeli enako postavitev, le da smo vse potrojili.

Najprej smo želeli generirati UDP pakete velikosti 1500 bajtov, saj smo mislili, da bodo povzročili večje motnje kot krajši paketi. Ob upoštevanju, da

bi bili vmesni časi med paketi podobni tistim pri paketih, velikih 42 bajtov, bi bilo namreč manj nezasedenega prostora. Po nadaljni analizi smo ugotovili, da smo lahko ustvarili večje motnje s krajšimi paketi.

Ob odprtju ene dostopne točke in generiranju motenj (10.000 UDP paketov na sekundo) iz drugega Wi-Fi USB adapterja smo zajeli UDP promet in analizirali zajete pakete. Ustvarili smo približno 1092 UDP paketov na sekundo velikosti 42 bajtov. Hitrost zajemanja paketov je bila ves čas precej konstantna brez nenadnih poskokov ali padcev. Ob generiranju paketov velikosti 1500 bajtov je hitrost sprejemanja paketov zelo nihala – med 15 in 95 paketov na sekundo. S tem so nastajala večja območje nezasedenega prostora, ko bi se lahko BLE paketi prosto pošiljali brez motenj. Zato smo se odločili za motnje s paketi, velikimi 42 bajtov.

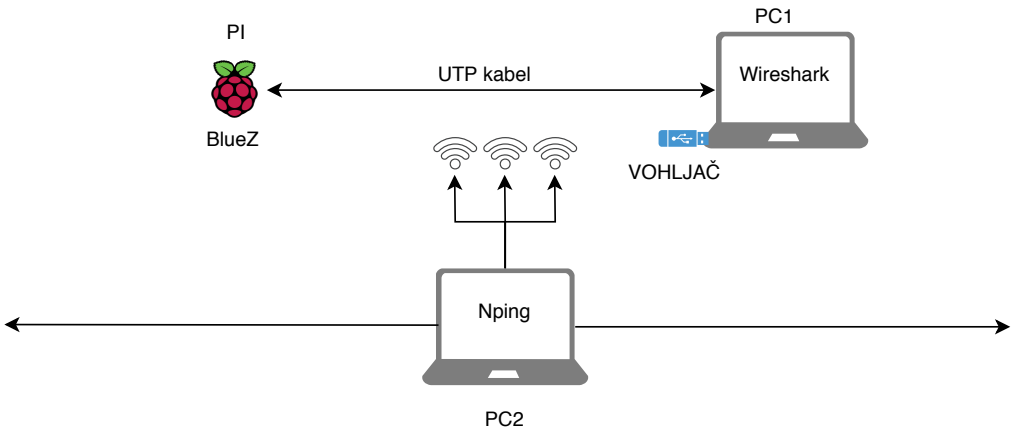
Poglavje 5

Poskusi

S tremi različnimi poskusi želimo testirati uspešnost komunikacije pri protokolu BLE. Meritve bodo usmerjene predvsem v analizo števila zaznanih paketov ter v delovanje skakanja med kanali. Prvi poskus bo testiral delovanje oglaševalskih kanalov, kjer želimo izmeriti vpliv motenj Wi-Fi na zaznavo oglaševalskih paketov. Pri tem poskusu ne bomo vzpostavili komunikacije med dvema napravama, temveč bomo z eno BLE napravo oglaševali svojo prisotnost ter z BLE vohljačem poskušali zaznati čim več paketov te naprave. Pri drugem in tretjem poskusu pa bomo med dvema napravama vzpostavili BLE povezavo ter vpeljali motnje na enem Wi-Fi kanalu oz. na treh Wi-Fi kanalih. Pri tem nas bo najbolj zanimal čas reagiranja naprav na motnje ter način izmikanja motnjam.

5.1 Oglaševalski kanali

Na testiranju oglaševalskih kanalov smo se odločili meriti razdaljo od BLE oglaševalca, pri kateri motnje Wi-Fi povzročajo preveč napak oz. napačno sprejetih paketov, da bi se napravi lahko povezali. Vse eksperimente smo opravili na domačem dvorišču, saj smo želili odprt prostor, kjer bi bilo čim manj odbojev signala. Kot oglaševalca smo uporabili napravo Raspberry Pi, na kateri smo pognali Python skripto za začetek oglaševanja, ki je del pro-

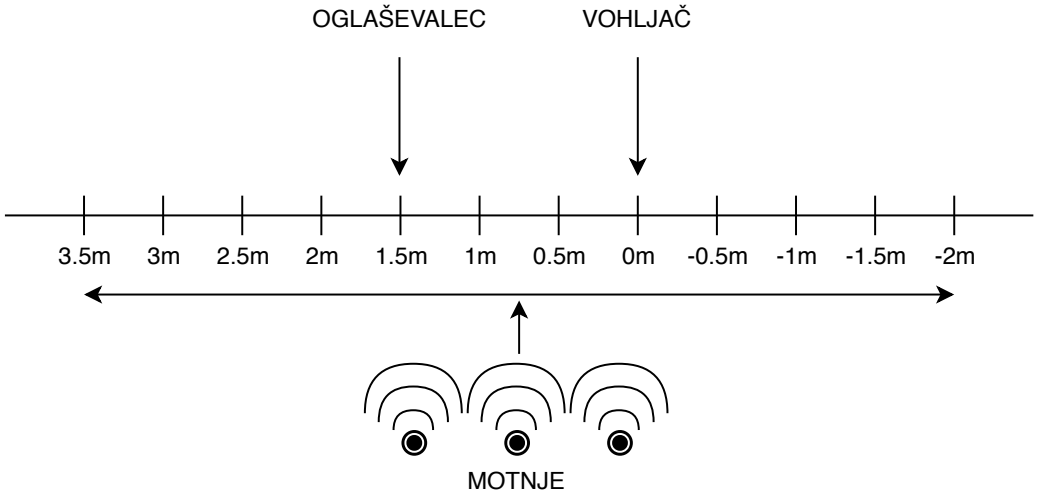


Slika 5.1: Fizična postavitev testnega okolja

grama BlueZ. Pri tem nismo spreminjali nobenih parametrov oglaševanja, saj smo želeli preizkusiti delovanje naprave po svojih samodejnih nastavitvah. BLE vohljača smo postavili 1.5 m od oglaševalca ter postavili sistem premikajočih Wi-Fi motenj. Na računalnik z nameščenim operacijskim sistemom Linux Mint smo povezali 5 Wi-Fi USB adapterjev ter jih razmaknili na približno 30-40 cm narazen. Na treh izmed njih smo nato odprli dostopno točko in se z dvema Wi-Fi USB adapterjema ter z omrežno kartico prenosnega računalnika povezali na ustvarjene dostopne točke. Nato smo še pognali program Nping za generiranje UDP prometa ter z BLE vohljačem merili 5 minut na vsaki izmed razdalj.

Motnje so bili UDP paketi dolžine 42 bajtov, ki smo jih generirali čim več v čim hitrejšem času. Omejitev je bila samo zmožnost naše opreme. Sistem motenj smo premikali na razmakih po 0.5 m, kjer smo kot ničelno točko merjenja izbrali BLE vohljač. Merili smo od 3.5 m na eno stran ter do 2 m na drugo stran vohljača. Postavitev je bolj nazorna na slikah 5.1 ter 5.2.

Dostopne točke smo odprli na Wi-Fi kanalih 1, 3 in 11, saj smo želeli čim bolj ovirati oglaševalske kanale 37, 38 in 39, ki jih ti kanali najbolj prekrivajo. Največ motenj smo pričakovali na BLE kanalu 38, saj je Wi-Fi kanal 3 postavljen ravno čezenj, medtem ko sta BLE kanala 37 in 39 postavljena bolj na obrobje Wi-Fi kanalov 1 in 11.



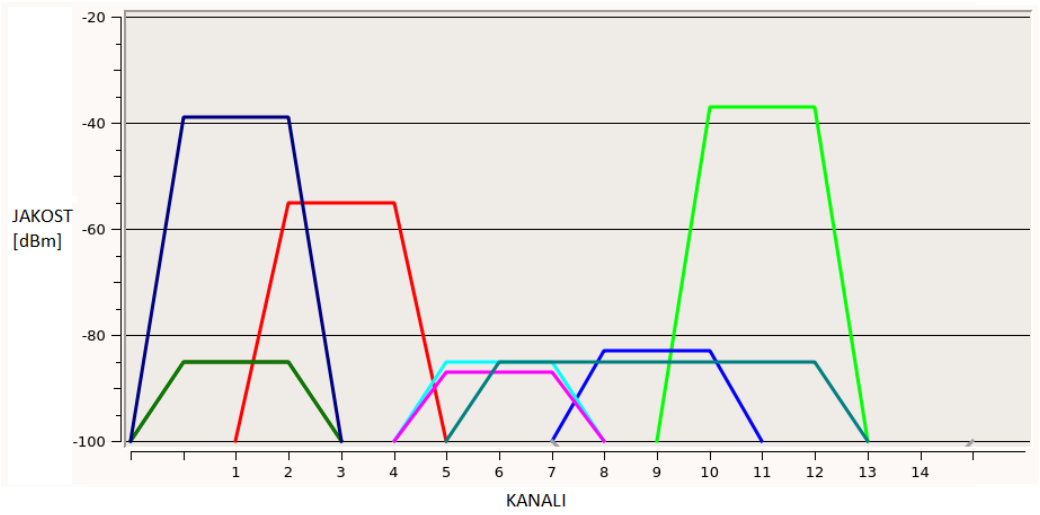
Slika 5.2: Logična postavitev testnega okolja

Z grafičnim analizatorjem omrežij LinSSID [7] smo zajeli sliko motenj (5.3), kjer najvišji modri, rdeči ter zeleni signal predstavljajo naše dostopne točke. Na sliki lahko vidimo tudi druge dostopne točke, ki imajo veliko šibkejše signale od naših. Privzeli smo, da so ti signali preveč šibki, da bi vplivali na naše rezultate oz. da so to motnje, ki se jim ne moremo izogniti, saj ne moremo zagotoviti laboratorijskih pogojev testiranja.

Pred začetkom meritev smo postavili tri hipoteze:

- skupno število paketov v petih minutah bi bilo brez motenj okrog 700
- največ motenj pričakujemo na kanalu 38
- število paketov bo približno simetrično s središčem pri vohljaču

Prva hipoteza temelji na že prej izmerjenih časih, ko smo testirali različne postavitve naprav. BLE naprava oddaja oglaševalske pakete periodično na kanalih 37, 38, 39 ter nato počaka nekaj več časa. Nato spet v hitrem intervalu odda 3 pakete. Pričakujemo, da bodo časi med paketi na kanalih 37 in 38 ter med kanaloma 38 in 39 nekaj milisekund, med sprejemom na kanalu 39 in 37 pa približno 1.2 do 1.3 sekunde. Ti časi so spremenljivi, saj



Slika 5.3: Primer motenj na Wi-Fi kanalih 1, 3 in 11

se osnovnemu intervalu vsakič doda naključen čas med 0 in 10 ms. Največji osnovni čas znaša 10.24 s. Več o teh časih je pojasnjeno v poglavju *Čas med paketi 3.2*.

Največ napak oz. najmanj sprejetih paketov pričakujemo na kanalu 38, saj se najbolj prekriva s katero izmed vklopljenih dostopnih točk - Wi-Fi kanalom 3. Kanala 37 in 39 sta bolj na obrobju najbolj izpostavljenih Wi-Fi kanalov in zato se lahko signal bolje širi.

Pri 0 m pričakujemo najmanj sprejetih paketov, ki se nato simetrično večajo z oddaljevanjem motenj.

Na tabeli 5.1 lahko vidimo zastopanost posameznih kanalov pri različnih oddaljenostih motenj. Po pričakovanjih lahko razberemo, da je največ sprejetih paketov nekaj pod 700, ko motenj ni oz. ko so motnje precej oddaljene. Motnje začnejo postajati opaznejše pri razdalji 1.5 m, ko so v neposredni bližini oglaševalca. Z bližanjem vohljaču smo sprejeli vedno manj paketov, nato pa se med -0.5 m in -1 m začne število sprejetih paketov spet večati. Opazimo lahko tudi, da imamo na mestih, kjer motnje niso zelo močne, približno enako porazdeljenost po kanalih. Veliko manjšo zastopanost ima le kanal 38 v bližini motenj.

	Vsi paketi v 5 minutah	Zastopanost 37	Zastopanost 38	Zastopanost 39
Brez motenj	679	227	225	227
3.5m	689	230	229	230
3m	673	224	224	225
2.5m	686	230	226	230
2m	669	224	221	224
1.5m	70	29	12	29
1m	12	5	1	6
0.5m	19	8	3	8
0m	14	6	2	6
-0.5m	12	4	4	4
-1m	469	168	159	169
-1.5m	644	220	203	221
-2m	680	226	227	227

Tabela 5.1: Primerjava zastopanosti posameznih kanalov

	Vsi paketi v 5 minutah	Napake kanal 37	Napake kanal 38	Napake kanal 39	Napake vsi kanali
Brez motenj	679	0.00%	0.89%	0.00%	0.29%
3.5m	689	0.00%	1.75%	0.00%	0.58%
3m	673	0.00%	3.12%	0.00%	1.04%
2.5m	686	0.00%	0.44%	0.00%	0.15%
2m	669	0.00%	1.36%	0.00%	0.45%
1.5m	70	0.00%	66.67%	0.00%	11.43%
1m	12	0.00%	0% (samo 1 paket)	0.00%	0.00%
0.5m	19	0.00%	66.67%	0.00%	10.53%
0m	14	0.00%	100.00%	0.00%	85.71%
-0.5m	12	0.00%	50.00%	0.00%	83.33%
-1m	469	0.00%	10.06%	0.00%	3.23%
-1.5m	644	0.00%	4.93%	0.00%	1.55%
-2m	680	0.00%	0.00%	0.00%	0.00%

Tabela 5.2: Primerjava napačno sprejetih paketov po posameznih kanalih

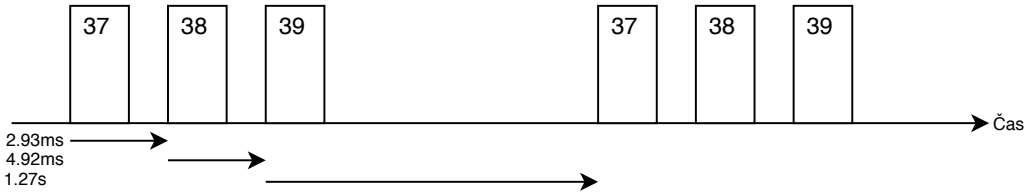
Na tabeli 5.2 imamo navedene napačno sprejete pakete po posameznih kanalih ter skupne napake vseh treh kanalov. Med napake so všteti paketi z napačno vrednostjo CRC, z neznanimi podatki ali slabo oblikovani paketi -

taki, ki jih Wireshark ni znal povsem razbrati. Največ napak je bilo zaznih v bližini vohljača, zanimivo pa je, da so bili vsi sprejeti paketi na kanalih 37 ter 39 brez napak. Naše motnje so torej povzročale napake samo na kanalu 38, na ostalih dveh kanalih pa so povzročile samo zmanjšanje števila sprejetih paketov. Podatke med 1 m ter -0.5 m je potrebno gledati v širšem kontekstu, saj je bilo število sprejetih paketov izjemno majhno in je le nekaj paketov oblikovalo statistiko. Tako imamo pri 1 m vse pakete pravilno sprejete, hkrati pa smo pri tej razdalji zajeli najmanj paketov izmed vseh meritev.

	Paketi/minuto	Primerjava brez motenj	BLE RSSI (dBm)
Brez motenj	135.8	100.00%	-47.17
3.5m	137.8	101.47%	-49.93
3m	134.6	99.12%	-50.38
2.5m	137.2	101.03%	-50.90
2m	133.8	98.53%	-52.71
1.5m	14	10.31%	-57.47
1m	2.4	1.77%	-55.50
0.5m	3.8	2.80%	-53.37
0m	2.8	2.06%	-57.93
-0.5m	2.4	1.77%	-50.83
-1m	99.2	69.07%	-49.41
-1.5m	128.8	94.85%	-49.94
-2m	136	100.15%	-49.56

Tabela 5.3: Primerjava prejete moči signala ter razmerja med prejetimi paketi v primerjavi z zajemom brez motenj

Tabela 5.3 vsebuje preračunano število sprejetih paketov na minuto, primerjavo sprejetih paketov z meritvijo brez motenj ter povprečno vrednost RSSI vseh paketov pri določeni meritvi. Pri primerjavi števila prejetih paketov z meritvijo brez motenj imamo v kar nekaj primerih rezultate tudi nad 100%, kar lahko pripišemo napaki vohljača. Vohljač je vedno ob začetku sprejemanja paketov v Wiresharku prikazal še nekaj paketov, ki jih je zajel



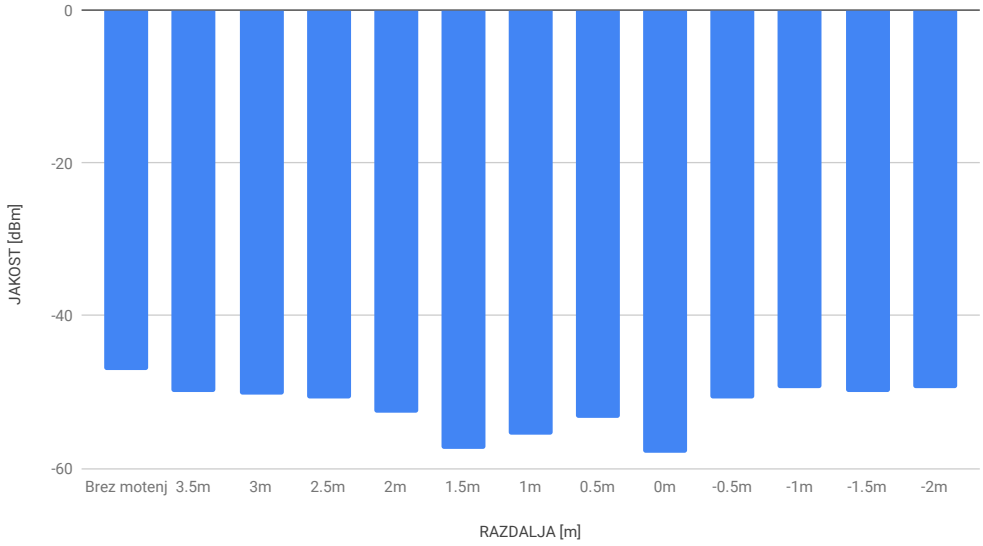
Slika 5.4: Časi med sprejemom paketov iz oglaševalskih kanalov

še pred našim ukazom zajemanja. To je Wireshark prikazal kot sprejete pakete v prvi sekundi. V nekaterih primerih je zajel tudi do nekaj deset takih paketov. To smo odpravili tako, da smo nekaj sekund po začetku zajemanja začeli ponovno zajemati pakete, kar je izbrisalo vse do tedaj sprejete pakete, in vohljač je začel sprejemati od začetka.

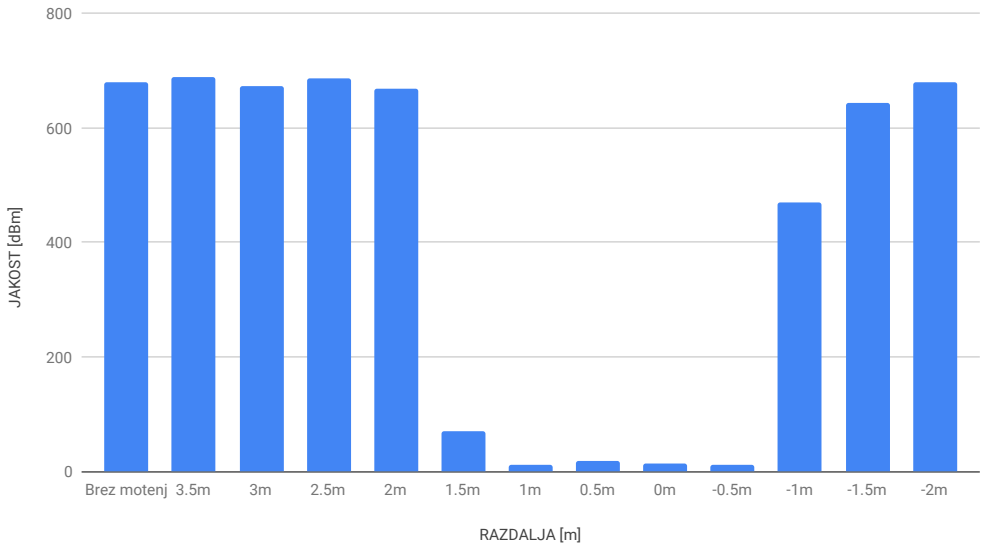
Kot napako pri primerjavi brez motenj moramo upoštevati tudi začetni čas zajemanja, saj dobimo različne čase, če sprejmemo prvi paket iz kanala 37 ali iz kanala 39. Časi med paketi iz zaporednih kanalov so namreč različni. To se vidi tudi na sliki 5.4, kjer imamo izračunane povprečne čase med sprejemom paketov iz zaporednih kanalov. Še en razlog za to anomalijo podatkov lahko pripišemo kvaliteti vohljača, saj le-ta spada med bolj cenene izdelke v svojem segmentu.

Zaznana moč signala RSSI se lepo prikaže pri primerjavi grafov 5.5 ter 5.6. Vidi se namreč, da je bilo največ sprejetih paketov takrat, ko je bila tudi najvišja zaznana moč signala. Oglaševalec je oddajal pakete ves čas z enako močjo, vendar je zaradi motenj vohljač ta signal včasih zaznal kot šibkejšega.

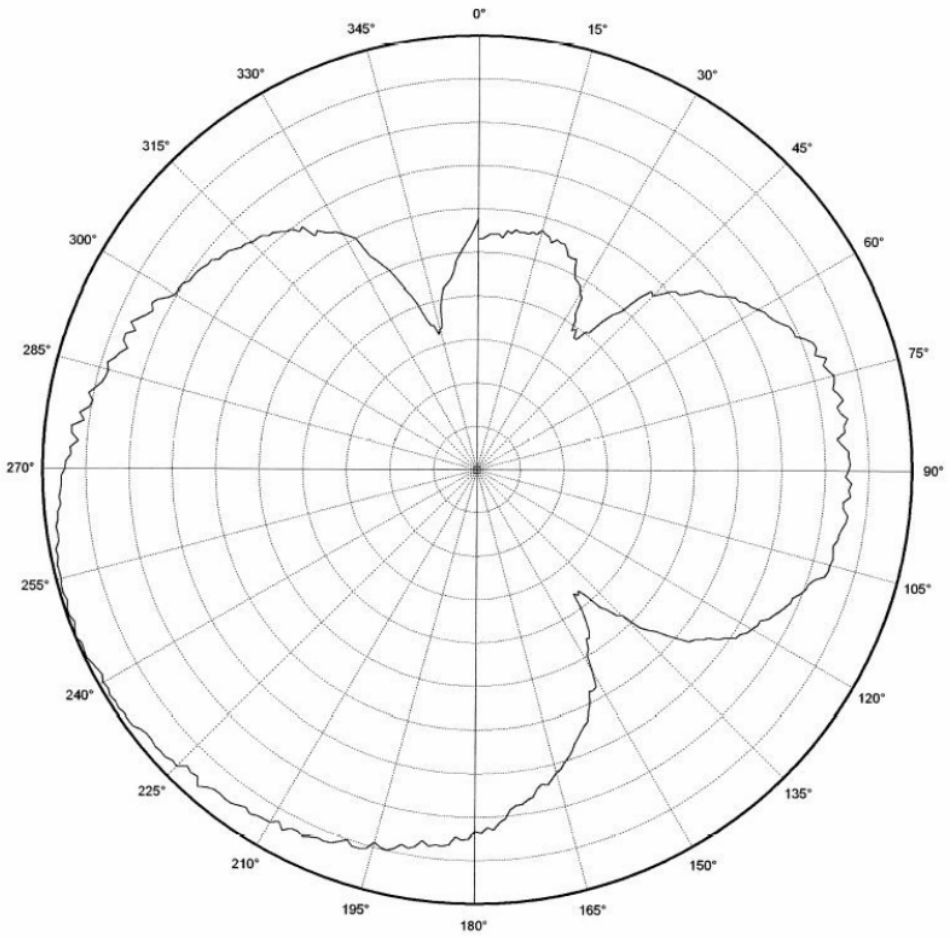
Problem, ki nam je oteževal delo pri poskusih, je bila usmerjenost antene. Wi-Fi USB adapterji, ki smo jih uporabili, ter vohljač ne oddajajo oz. sprejemajo signala v vse smeri enako močno. Primer tega lahko vidimo na sliki 5.7, ki pa ni slika oddajne moči katere izmed naših naprav, temveč je samo primer delovanja PCB antene. Na obodu slike je merilo središčnega kota v stopinjah, notranji koncentrični krogi pa predstavljajo moč signala. V središču je oddajna moč 0 dBm, vsak koncentrični krog pa predstavlja zmanjšanje za 4 dBm. Na sliki vidimo, da je moč signala v eni smeri veliko večja kot v



Slika 5.5: Primerjava zaznane moči signala



Slika 5.6: Primerjava št. prejetih paketov v petih minutah



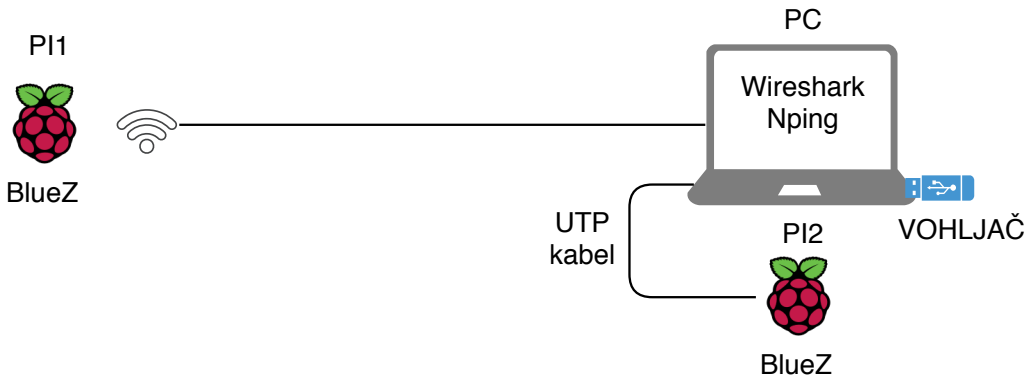
Slika 5.7: Karakteristika antene [11]

drugi oz. v vmesnih območjih. Naši Wi-Fi USB adapterji ter vohljač niso bili fiksno pritrjeni, saj smo jih morali večkrat premikati. Usmerjenost antene bi lahko pojasnila nesimetričnost števila prejetih paketov glede na razdaljo od vohljača.

5.2 Motnje na enem Wi-Fi kanalu

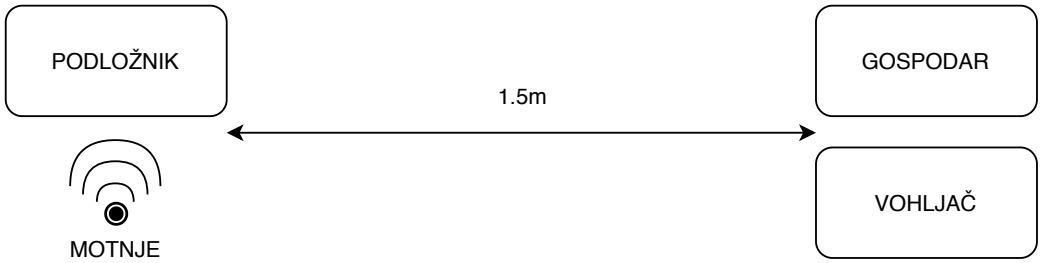
Pri drugem poskusu smo želeli ugotoviti, koliko časa potrebujeta napravi, ki imata vzpostavljeno BLE komunikacijo, da zaznata motnje in se začneta

izogibati nekaterim BLE kanalom. Za ta poskus smo uporabili dva Raspberry Pi mikroračunalnika, kjer smo enemu določili vlogo gospodarja, drugemu pa podložnika. Nato smo postavili vohljača h gospodarju, saj smo želeli zaznati vse njegove pakete. Gospodar namreč lahko pošilja nekatere pakete, ki jih podložnik ne more.



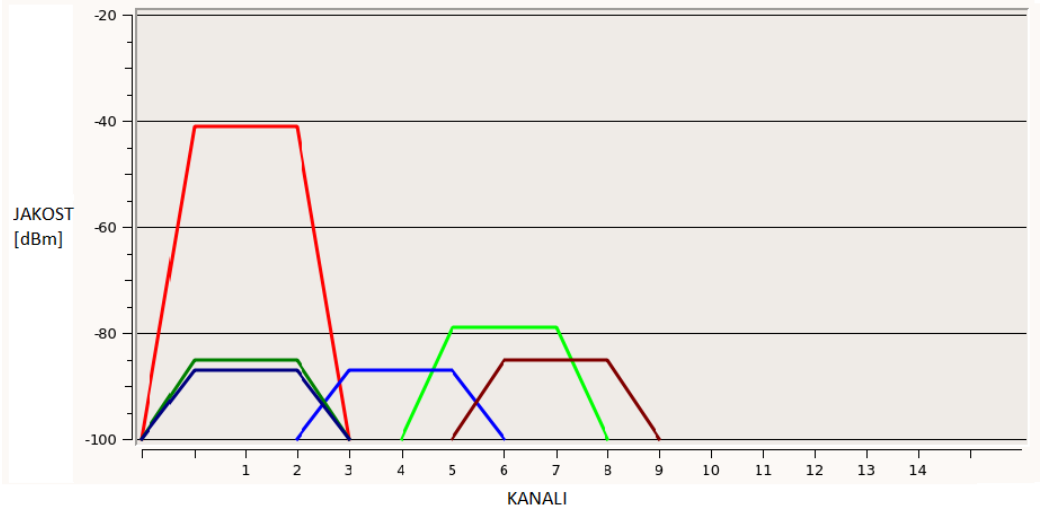
Slika 5.8: Fizična postavitev testnega okolja

Postavitev testnega okolja lahko vidimo na slikah 5.8 ter 5.9. Na prenosniku smo na USB podaljšek povezali dva Wi-Fi USB adapterja ter ju postavili k podložniku. Na enem izmed adapterjev smo odprli dostopno točko na Wi-Fi kanalu 1, z drugim adapterjem pa smo se povezali nanjo. Na prenosniku smo nato pognali motnje s programom Nping, kjer smo poskušali generirati 10.000 UDP paketov na sekundo, vendar naši Wi-FI USB adapterji oz. procesor in pomnilnik prenosnika tega niso mogli doseči, zato so generirali čimveč paketov na sekundo po svojim zmožnostih. UDP paketi so bili velikosti 42 bajtov, meritve pa so trajale 2 minuti.



Slika 5.9: Logična postavitev testnega okolja

Na sliki 5.10 lahko vidimo našo dostopno točko na Wi-Fi kanalu 1. Prepoznamo je lahko kot najvišji stolpec rdeče barve.



Slika 5.10: Primer motenj na Wi-Fi kanalu 1

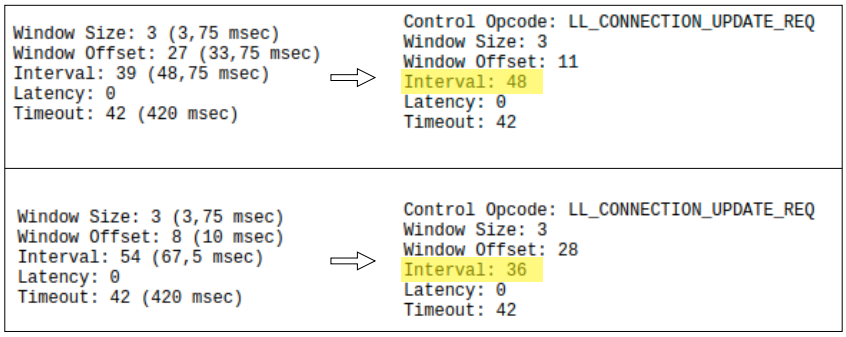
V tabeli 5.4 lahko vidimo statistiko prejetih paketov. Prva vrstica predstavlja test brez motenj, nato imamo 5 testov (Test_A*), kjer smo kot gospodarja imeli napravo Raspberry Pi 3 model B, nato pa še 2 testa (Test_B*), kjer smo kot gospodarja imeli napravo Raspberry Pi 3 model B+. Na dnu imamo še povprečji vseh testov Test_A* in Test_B*. Opazimo lahko, da smo vedno prejeli več paketov gospodarja, uspešnost pa je bila precej podobna, skoraj brez napak. Manj paketov podložnika smo prejeli zato, ker smo postavili motnje bliže njemu, zaradi BLE mehanizma izmikanja zasede-

	Št. paketov/minuto	Poslani paketi gospodar	Poslani paketi podložnik	Uspešnost gospodar	Uspešnost podložnik
Brez motenj	2,002.80	50.07%	49.93%	100.00%	99.95%
Test_A1	1992.5	50.14%	49.86%	99.90%	99.95%
Test_A2	1994	50.03%	49.97%	99.90%	100.00%
Test_A3	1991	50.30%	49.70%	100.00%	99.95%
Test_A4	2000	50.28%	49.73%	100.00%	99.90%
Test_A5	2003.5	50.24%	49.76%	100.00%	100.00%
Test_B1	2620	50.48%	49.52%	100.00%	99.81%
Test_B2	2628	50.27%	49.73%	99.92%	99.92%
Povprečje A	1996.2	50.20%	49.80%	99.96%	99.96%
Povprečje B	2624	50.37%	49.63%	99.96%	99.87%

Tabela 5.4: Primerjava prejetih paketov

nim kanalom pa so bili skoraj vsi paketi brez napak CRC oz. so bili pravilno oblikovani.

Zanimiva je tudi primerjava števila paketov na minuto med testi A ter testi B, saj lahko opazimo velik skok v hitrosti. To lahko pojasnimo z drugačnimi začetnimi nastavitvami naprave Raspberry Pi 3 model B+, ki je določila manjši interval med paketi. Na sliki 5.11 imamo prikazana paketa CONNECT_REQ in LL_CONNECTION_UPDATE_REQ pri poskusih Test_A5 ter Test_B1. Pomemben je predvsem drugi paket, ki določi dokončne čase, ki se uporabljajo med komunikacijo. Paket je bil izmenjan že v prvi sekundi komunikacije in je določil interval med paketi. Vidimo lahko, da je interval pri Test_A5 določen na 48, pri Test_B1 pa na 36. Razmerje 48/36 znaša 1.34, ki je podobno razmerju 2624/1996.2, ki znaša 1.31. Ta razlika je torej nastala izključno zaradi parametrov komunikacije, ki jih je določil gospodar. Več o paketih, namenjenih vzdrževanju komunikacije, lahko preberete v poglavju *Izogibanje motnjam 3*.



Slika 5.11: Primerjava med paketoma CONNECT_REQ in LL_CONNECTION_UPDATE_REQ pri poskusih Test_A5 (zgoraj) ter Test_B1 (spodaj)

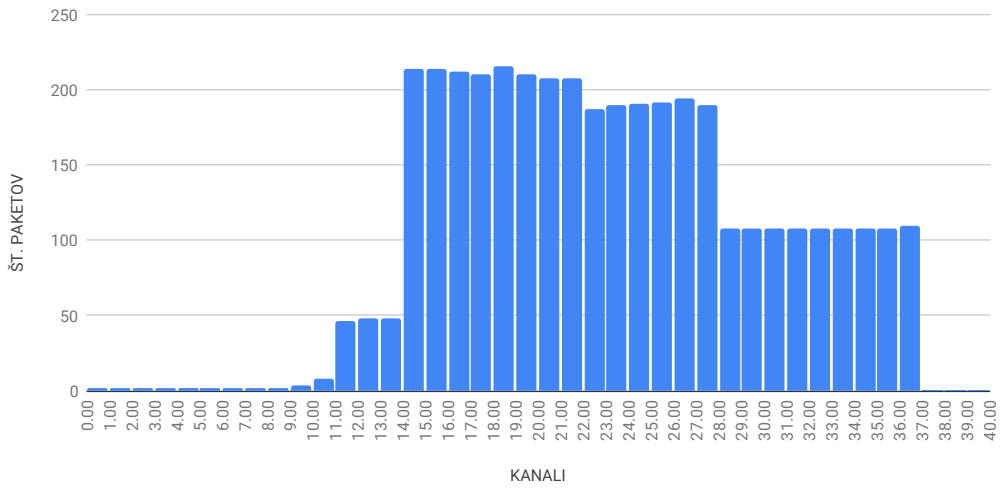
Tabela 5.5 vsebuje podatke iz paketov LL_CHANNEL_MAP_REQ. Prvi stolpec nam pove zaporedno število prvega takega prejetega paketa, drugi stolpec pa čas v sekundah, kdaj smo prejeli ta paket. Vidimo lahko, da so ti rezultati precej konstantni. Napravi sta potrebovali približno 3.9 sekunde, da sta si prvič izmenjali paket LL_CHANNEL_MAP_REQ. Nekaj začetnih paketov je bilo namenjenih izmenjavi parametrov komunikacije ter prilagoditvi časov med paketi. Predzadnji stolpec vsebuje število največ blokiranih kanalov. V vseh primerih so bili blokirani kanali 0–13, v treh testih pa je bil blokiran še vsaj en kanal, ki pa ni bil na območju naših motenj. Zadnji stolpec vsebuje podatek o številu paketov LL_CHANNEL_MAP_REQ, preračunanih na minuto.

Na grafu 5.12 lahko vidimo zastopanost posameznih kanalov pri Test_A3. Pri tem testu smo v dveh minutah prejeli 3 pakete LL_CHANNEL_MAP_REQ: prvi je blokiral kanale 0-9, drugi 0-10, tretji 0-13. Skok je bil določen kot 9. Opazimo lahko, da je zastopanost kanalov 28–36 precej manjša od 14–27, kljub temu da ti kanali niso bili blokirani. Predvidevamo, da je to nastalo zaradi implementacije BLE-ja, ki ob preverjanju blokiranih kanalov včasih ne sledi sekvenci skoka, temveč vstavi nek drug kanal namesto predvidenega.

	1.paket C_MAP	Čas (s) 1.paket C_MAP	Največ blokiranih kanalov	Število paketov C_MAP/minuto
Test_A1	124	3.84177	14 + 1	2.00
Test_A2	124	3.851431	14 + 2	2.50
Test_A3	127	3.889524	14	1.50
Test_A4	137	3.890979	14	2.00
Test_A5	141	3.894224	14	1.50
Test_B1	136	3.938762	14	1.5
Test_B2	132	4.039994	14 + 1	2
Povprečje A	130.6	3.8735856	14.6	1.90
Povprečje B	134	3.989378	14.5	1.75

Tabela 5.5: Primerjava paketov LL_CHANNEL_MAP_REQ

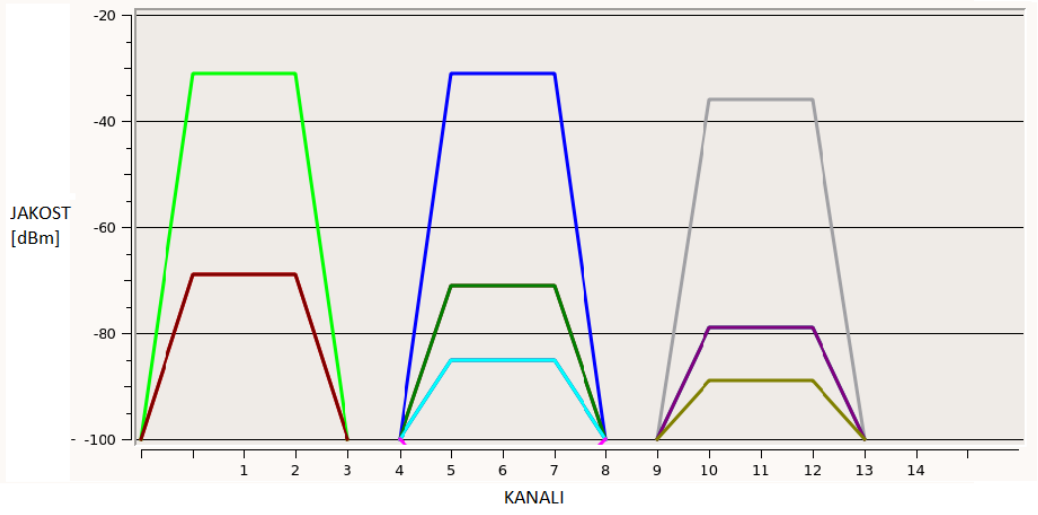
To se verjetno zgodi zato, ker bi včasih lahko bilo veliko kanalov blokiranih in BLE raje nadaljuje s pošiljanjem paketov v predvidenih intervalih, kljub temu da ne sledi sekvenci, predvideni s skokom. V našem primeru se je zgodilo, da je bil približno vsak tretji skok na blokiran kanal in je BLE začel vstavljati veliko nepredvidenih skokov na take kanale, ki so bili večinoma med 14 in 27.



Slika 5.12: Porazdelitev paketov po kanalih

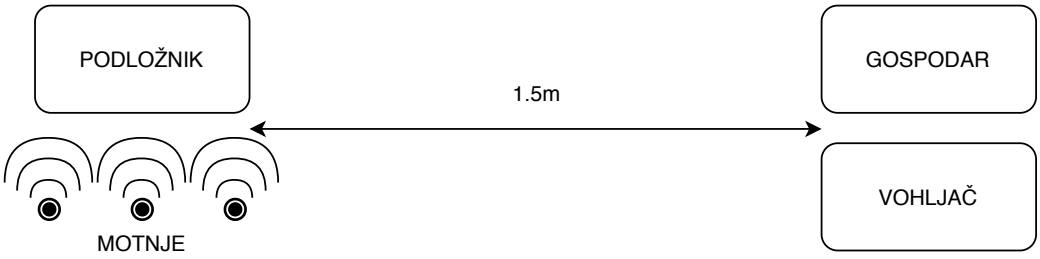
5.3 Motnje na treh Wi-Fi kanalih

S tretjim poskusom smo želeli z motnjami povzročiti čim več blokiranih kanalov pri vzpostavljeni BLE komunikaciji ter opazovati, kako bosta napravi reagirali. Predvsem nas je zanimalo kateri kanali bodo ostali zadnji in koliko časa bosta napravi potrebovali, da blokirata večino kanalov. Fizična postavitev je bila podobna kot pri motnjah na enem Wi-Fi kanalu (slika 5.8), tokrat smo le še dodali 3 Wi-Fi USB adapterje. Na treh adapterjih smo odprli dostopne točke na kanalih 1, 6 in 11 (slika 5.13) ter se povezali nanje z dvema adapterjema in omrežno kartico prenosnika. USB podaljšek smo tokrat uporabili za vohljača, ki smo ga premaknili čim bližje gospodarju. Logična postavitev testnega okolja je vidna na sliki 5.14. Motnje smo vklopili po že vzpostavljeni povezavi BLE, saj so UDP paketi zelo obremenili Wireshark, ki je generiral zelo velike datoteke. Opravili smo le tri testne meritve po približno minuto merjenja, saj smo opazili, da sta napravi dokaj hitro reagirali na motnje. Analizirali smo samo tisti del zajetih Wireshark datotek, ki so vsebovale UDP motnje.



Slika 5.13: Primer motenj na Wi-Fi kanalih 1, 6 in 11

Tabela 5.6 predstavlja analizo števila prejetih paketov. Vidimo lahko, da



Slika 5.14: Logična postavitev testnega okolja

se je število prejetih paketov zmanjšalo, bolj viden pa je tudi vpliv motenj na podložnika. Veliko paketov podložnika namreč sploh nismo prejeli, od prejetih pa je bilo več takih z napakami CRC kode in napačno oblikovanimi polji. Vpliv na gospodarja je po pričakovanjih veliko manjši, saj smo ga imeli postavljenega veliko bliže vohljaču.

	Št. paketov/minuto	Poslani paketi gospodar	Poslani paketi podložnik	Uspešnost gospodar	Uspešnost podložnik
Povprečje motenj na enem kanalu	1,996.20	50.20%	49.80%	99.96%	99.96%
Test_C	1,800.72	55.40%	44.60%	99.44%	97.51%
Test_D	1,743.75	57.16%	42.84%	99.77%	94.50%
Test_E	1,880.68	53.07%	46.93%	99.90%	98.22%

Tabela 5.6: Primerjava prejetih paketov

Tabela 5.7 vsebuje analizo paketov LL_CHANNEL_MAP_REQ. Pri teh vrednostih moramo biti pozorni, da so bile izmerjene po že vzpostavljeni komunikaciji med napravama, torej sta napravi reagirali veliko hitreje kot pri motnjah na enem Wi-Fi kanalu, ko sta si napravi na začetku morali še izmenjati nekatere parametre povezave. Pri testu Test_D smo prvi paket LL_CHANNEL_MAP_REQ zaznali že zelo zgodaj, zato nismo bili prepričani, ali je bil povzročen zaradi naših motenj ali je bil povzročen zaradi kakšnega drugega vpliva. Zato smo dopisali še čas sprejetja drugega takega paketa.

Pri vseh treh testih je bilo blokiranih največ možnih kanalov - 35. Ko ostaneta še zadnja dva kanala, BLE preneha blokirati kanale, saj bi tako ostal samo še zadnji in ne bi mogel več skakati med njimi. V vseh primerih

	1.paket C_MAP	Čas (s) 1.paket C_MAP	Največ blokiranih kanalov	Št. paketov C_MAP/minuto
Povprečje motenj na enem kanalu	130.6	3.8735856s	14.6	1.90
Test_C	89	3.540028s	35	7.78
Test_D	19 oz. 119	0.709311s oz. 4.753252s	35	5.70
Test_E	63	2.832990s	35	8.82

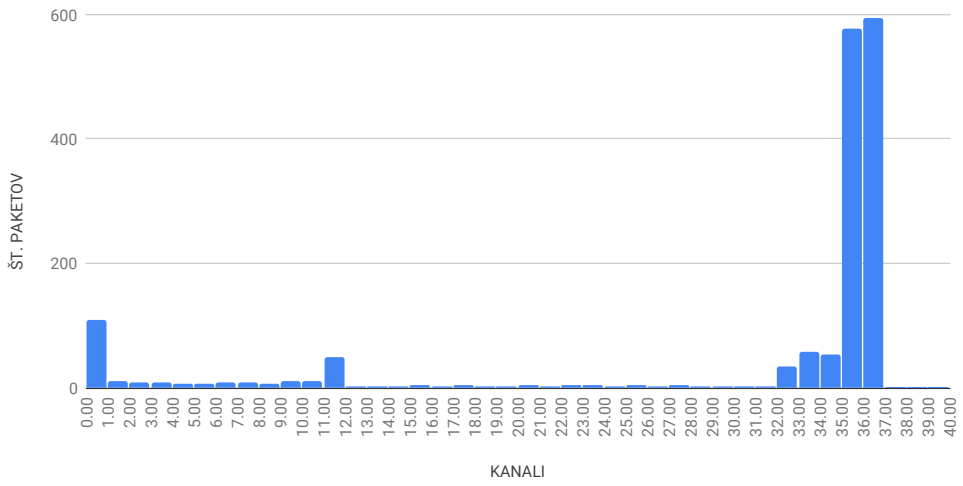
Tabela 5.7: Primerjava paketov LL_CHANNEL_MAP_REQ

	Čas (s) >= 31 blokiranih kanalov	Paket >= 31 blokiranih kanalov	Čas (s) 35 blokiranih kanalov	Paket 35 blokiranih kanalov
Test_C	11.884011	369	27.882816	842
Test_D	8.902676	227	20.856794	559
Test_E	6.984205	175	6.984205	175

Tabela 5.8: Primerjava časov do blokiranja kanalov

sta kot zadnja ostala kanala 35 in 36 - zadnja dva možna kanala. Če jih je ostalo še manj kot sedem, je ostalo večinoma zadnjih pet ter prvi kanal. Porazdelitev po kanalih lahko vidimo na grafu 5.15.

V tabeli 5.8 imamo navedene še čase in zaporedna števila tistih paketov LL_CHANNEL_MAP_REQ, ki so blokirali 31 ali več kanalov ter 35 kanalov.



Slika 5.15: Porazdelitev paketov po kanalih

Poglavje 6

Zaključek

V sklopu diplomske naloge smo testirali uspešnost komunikacije protokola BLE ob motnjah Wi-Fi. Testiranje je bilo razdeljeno v tri sklope, kjer smo v prvem sklopu ciljali oglaševanje BLE naprav, v drugih dveh pa je bil cilj motenje že vzpostavljene komunikacije med BLE napravama.

Ugotovili smo, da je vpliv na oglaševalske kanale močan le v neposredni bližini naprav, medtem ko se že nekaj metrov od vohljača ne pozna več nobenega vpliva. Opazili smo tudi, da je motnjam najbolj podvržen BLE kanal 38, ki je zaradi svojega frekvenčnega območja najbolj izpostavljen Wi-Fi motnjam, saj ga lahko moti več Wi-Fi kanalov.

Na podatkovnih kanalih smo merili, koliko časa potrebujeta BLE napravi, da zaznata motnje in reagirata nanje. Ugotovili smo, da zelo hitro reagirata ter da se ob motnjah na enem Wi-Fi kanalu skoraj ne pozna padec v hitrosti prenosa podatkov.

Pri tretjem testiranju smo želeli preveriti, kako bosta povezani BLE napravi reagirali na motnje po celotnem razpoložljivem frekvenčnem območju. Pri vseh testih sta napravi blokirali vse kanale razen dveh in od tam naprej skakali na zadnjih dveh razpoložljivih kanalih. Opazili smo tudi, da sta kot zadnja dva kanala vedno ostala kanala 35 in 36.

Literatura

[1] 2019 Bluetooth Market Update. Dosegljivo: https://3pl46c46ctx02p7rzdsvsg21-wpengine.netdna-ssl.com/wp-content/uploads/2018/04/2019-Bluetooth-Market-Update.pdf. [Dostopano: 21. 06. 2019].

[2] Adafruit LE Sniffer. Dosegljivo: https://www.adafruit.com/product/2269. [Dostopano: 21. 06. 2019].

[3] Adaptive Frequency Hopping. Dosegljivo: https://microchip.wdfiles.com/local--files/wireless:ble-link-layer-channels/adaptive-frequency-hopping.png. [Dostopano: 21. 06. 2019].

[4] BlueZ. Dosegljivo: http://www.bluez.org/. [Dostopano: 21. 06. 2019].

[5] Comparison between Bluetooth, Bluetooth Smart Ready and Bluetooth Smart. Dosegljivo: https://developex.com/blog/wp-content/uploads/2016/12/image03.png. [Dostopano: 21. 06. 2019].

[6] D-Link-DWL-G122. Dosegljivo: https://www.amazon.com/D-Link-DWL-G122-Compact-Wireless-Adapter/dp/B0002DQUHC. [Dostopano: 21. 06. 2019].

[7] LinSSID. Dosegljivo: https://sourceforge.net/projects/linssid/. [Dostopano: 21. 06. 2019].

[8] Nping. Dosegljivo: https://nmap.org/nping/. [Dostopano: 21. 06. 2019].

[9] Raspberry Pi. Dosegljivo: `https://www.raspberrypi.org/`. [Dostopano: 21. 06. 2019].

[10] Wireshark. Dosegljivo: `https://www.wireshark.org/`. [Dostopano: 21. 06. 2019].

[11] Audun Anderson. Application Note ANO43. Technical report. [Dostopano: 21. 06. 2019].

[12] N. Gupta. *Inside Bluetooth Low Energy.* Artech House Remote Sensing Library. Artech House, 2013.

[13] K. Townsend, R. Davidson, and C. Cufí. *Getting Started with Bluetooth Low Energy.* O'Reilly, 2014.