

Langaton tiedonsiirto UHF-taajuudella

Kotiautomaatio



Ammattikorkeakoulututkinnon opinnäytetyö

Tietojenkäsittelyn koulutusohjelma

Hämeenlinna, syksy 2017

Antti Juntunen

Tietojenkäsittelyn koulutusohjelma
Visamäki

Tekijä	Antti Juntunen	Vuosi 2017
Työn nimi	Langaton tiedonsiirto UHF-taajuudella	
Työn ohjaaja/t	Erkki Laine	

TIIVISTELMÄ

Opinnäytetyön aiheena oli langaton tiedonsiirto UHF-taajuudella, ja työn toimeksiantajana toimi Älykkäät palvelut -tutkimusyksikkö. Opinnäytetyön tarkoituksena on tuottaa uutta tietoa ja koota olemassa olevaa tietoa UHF-taajuuksista lukijalle ja Älykkäät palvelut -tutkimusyksikölle.

Tämän työn tavoitteena on selkeyttää lukijalle, mitä yleisempiä tiedonsiirtomenetelmiä toimii UHF-taajuudella ja kuinka nämä eroavat toisistaan. Tämän opinnäytetyön käytännön osuudessa lukijaa opastetaan luomaan oma helposti laajennettavissa oleva kotiautomaatiojärjestelmä, jolla voi ohjata asunnon sähkölaitteita Android-puhelinsovelluksella ja lukea tietoja sensoreilta. Tämä järjestelmä luodaan Raspberry Pi v3 -tietokoneella, lähetin- ja vastaanotinpiirillä sekä 433 MHz taajuudella toimivilla etähallittavilla sähköpistorasioilla. Lopuksi lukija osaa tehdä kotiautomaatiojärjestelmän, joka on edullinen rakentaa sekä ylläpitää ja jota on helppo laajentaa omiin tarpeisiin.

Tässä opinnäytetyössä odotetaan lukijalta perustietämystä Raspberry Pi -tietokoneesta ja PHP, C# ja C++ -ohjelmointikielestä. Jos lukijalla ei ole yleistä tietämystä ohjelmoinnista, voi lukijan olla hankala toteuttaa toimivaa kotiautomaatiojärjestelmää ja muokata järjestelmää omiin tarpeisiinsa.

Avainsanat Raspberry Pi
UHF -taajuus
Kotiautomaatio

Sivut 36 sivua, joista liitteitä 1 sivu

Degree Programme in Business Information Technology
Visamäki, Hämeenlinna

Author	Antti Juntunen	Year 2017
Subject	Wireless Data Communication with UHF	
Supervisors	Erkki Laine	

ABSTRACT

The subject of this Bachelor's thesis is wireless data communication with UHF and the thesis was commissioned by the HAMK Smart services research unit. The purpose of this thesis is to provide new and gather existing information about ultra-high frequencies for reader and the Smart services research unit.

The goal of this thesis is to clarify reader what wireless communication protocols work on UHF range and what is the difference between those protocols. In the practical section, thesis will guide reader to build a home automation system that allows the user to control home lights and read data from the sensors. This automation system is built with raspberry pi v3 computer and 433 MHz receiver- transmitter circuit and power sockets that work 433 MHz range. In the end, reader knows how to build home automation system and modify that.

In this thesis, the writer expects reader to know the basics of a Raspberry Pi computer and PHP, C# and C++ code languages. If the reader does not know basics of programming, it may be difficult for the reader to modify home automation system, to their own needs.

Keywords Raspberry Pi
receiver- transmitter circuit
Ultra-High Frequency
Home automation system

Pages 36 pages including appendices 1 page

SISÄLLYS

TERMIT JA SANASTO.....	
1 JOHDANTO.....	1
2 UHF-TAAJUUS	2
2.1 UHF-taajuuksien vertailu.....	4
2.1.1 UHF-taajuuden kantomatka	5
2.1.2 UHF-taajuuksien datasiirtonopeudet	7
2.1.3 UHF-taajuuksien esteiden läpäisemiskyky ja käyttökohteet.....	9
2.2 Eri tiedonsiirtomenetelmien tietoturva	10
3 KOTIAUTOMAATIO 433 MHZ TAAJUUTTA HYÖDYNTÄEN.....	12
3.1 Raspberry Pi 3 Model B	13
3.2 Langattomat etäohjattavat virtakytkimet.....	14
3.3 433 MHz lähetin ja vastaanotin	14
3.4 Sääasema.....	14
3.5 Raspberry Pi käyttöjärjestelmän ja palvelimen asennus	15
3.6 Lähettimen ja vastaanottimen kytkeminen	17
3.7 Ohjelmakoodit.....	19
3.7.1 C++ piHomeEasy ohjelma	19
3.7.2 PHP REST -rajapinta	20
3.7.3 PHP tietokantayhteys	22
3.7.4 C++ bittiviran tarkkailu	23
3.7.5 Android-sovellus.....	28
4 TULOKSET JA POHDINTA.....	31
4.1 Markkinoilta löytyvien kotiautomaatiojärjestelmien vertailu	33
4.2 Pohdinta	34
5 YHTEENVETO	35
LÄHTEET	36

Liitteet

Liite 1 UHF-taajuusalueen käyttö Suomessa

TERMIT JA SANASTO

UHF	Ultra High Frequency (suom. Ultrakorkea taajuus) tarkoittaa mikroaaltojen taajuusalueita, joka sijoittuu välille 0,3 – 3 GHz.
Raspberry Pi	Raspberry Pi on yhden piirilevyn tietokone, jolla on ohjelmoitavia sisään- ja ulostuloja.
Raspbian	Debian pohjainen käyttöjärjestelmä Raspberry Pi -tietokoneelle.
REST- rajapinta	REpresentational State Transfer on lyhyesti selitettynä arkkitehtuurimalli ohjelmointirajapintojen toteuttamiseen. Se käyttää HTTP-protokollaa, jossa operaatioina ovat GET, POST, PUT ja DELETE (Paikkatietoikkuna 2012).
Git	Versionhallintaohjelmisto, hajautettuun työskentelyyn suunniteltu ohjelmisto, jota voidaan myös käyttää ohjelmien jakamiseen.
Github	Verkkosivusto, jonne voidaan tallentaa Git-versionhallintaohjelmalla projekteja.
PHP	Hypertext Preprocessor on ohjelmointikieli, jolla pääsääntöisesti ohjelmoidaan web-sivuja ja näiden toiminnallisuuksia.
C++	Kehittyneempi ohjelmointikieli C-kielestä, perustuu Olio-ohjelmointiin.
C#	Olio-ohjelmointikieli, johon on yhdistetty C++:n tehokkuus ja Java-kielen helppokäytettävyys.
GPIO	General Purpose I/O on ohjelmoitava portti, joka voidaan asettaa joko signaalin vastaanottajaksi tai lähettäjäksi.

WiringPi

Rajapintakirjasto Raspberry Pi GPIO:lle.

Bittivirta

Bittivirta (engl. Bitrate) -termiä käytetään kuvaamaan, kuinka paljon dataa voidaan siirtää tietyssä aikayksikössä.

1 JOHDANTO

Valitsin aiheen, koska olen jo pitkään miettinyt kotiautomaation rakentamisesta omaan asuntooni, mutta valmiiden kotiautomaatiojärjestelmien hinnat ovat todella korkeat ja näiden järjestelmien muokkaaminen omiin tarpeisiin melkein mahdotonta. Näin syntyi ajatus, miksi en tekisi itse järjestelmää, jolla voin ohjata asunnon valoja, tarkkailla lämpötilaa ja hallita muita sähkölaitteita. Asunnot ovat kovaa vauhtia muuttumassa älytaloiksi, joten aiheeni on ajankohtainen ja varmasti kiinnostava. Suomenkielistä tietoa UHF-taajuuksilla toimivista tiedonsiirtomenetelmistä on jonkin verran, tämä aihe avaisi lukijalle eri taajuuksien hyviä ja huonoja puolia ja kuinka toteuttaa edullinen kotiautomaatiojärjestelmä.

Tämä opinnäytetyö sisältää tietoa siitä, kuinka eri UHF-taajuudet kykenevät siirtämään dataa ja mitkä ovat näiden tiedonsiirtomenetelmien heikkoudet ja vahvuudet. Opinnäytetyö keskittyy yleisimpiin markkinoilla käytettäviin tiedonsiirtomenetelmiin, joita käytetään tiedonsiirtämiseen langattomasti. Näitä kyseisiä UHF-taajuuksia ovat mm. 433 MHz, 868 MHz, 900 MHz, 2.4 GHz ja 5 GHz.

Opinnäytetyössä käytännönosuus keskittyy tiedonsiirtämiseen 433.92 MHz taajuudella ja tällä taajuudella ohjataan myös asunnon valaistusta puhelin sovelluksen avulla. Kotiautomaatiojärjestelmä luodaan Raspberry Pi v3 -tietokoneella, etähallittavilla pistorasioilla sekä lähetin- ja vastaanotinpiirillä. Raspberry Pi:lle luodaan palvelin, joka käsittelee Android -sovellukselta tulevat käskyt. Käyttöliittymällä voidaan ohjata etähallittavia pistorasioita ja näyttää myös lämpötilaa sääasemalta. Raspberryn palvelin toimii kotiverkossa, mutta se on mahdollista siirtää myös pilvipalveluun, jos tarve vaatii. Kotiverkossa toimiva palvelin tarkoittaa sitä, ettei palvelimelle pääse käsiksi kuin saman reitittimen kautta, joten asunnon valoja ei voi hallita kodin reitittimen WiFi kantaman ulottumattomissa.

Opinnäytetyössä pyritään vastaamaan kysymyksiin: Mihin eri UHF-taajuuksilla toimivat tiedonsiirtomenetelmät soveltuvat ja mitkä ovat näille soveltuvia käyttötarkoituksia.

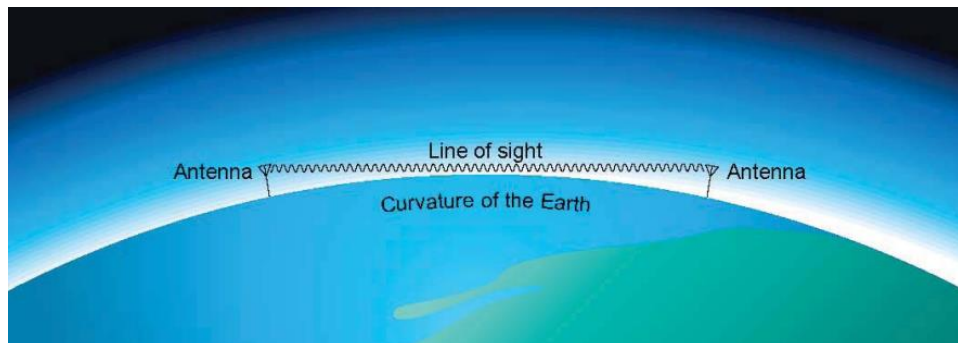
Opinnäytetyössä odotetaan lukijan perustietämystä Raspberry Pi:stä ja PHP, C# ja C++ -ohjelmointikielestä, siksi opinnäytetyö ei perehdy syvemmin siihen, kuinka Raspberry Pi ja etähallittavat laitteet toimivat.

2 UHF-TAAJUUS

UHF tulee englannin kielen sanoista Ultra High Frequency ja tarkoittaa suomeksi ultrakorkea taajuus. UHF -taajuuksia käytetään laajasti arkipäivän laitteissa vastaanottamaan ja lähettämään dataa.

UHF -taajuus sijoittuu mikroaalto välille 0,3 – 3GHz. Ultrakorkea on nykyään hieman harhaanjohtava nimitys tälle taajuusvälille, koska se ei ole läheskään korkeimpia taajuuksia, mitä nykyään hyödynnetään (WikiBooks 2017).

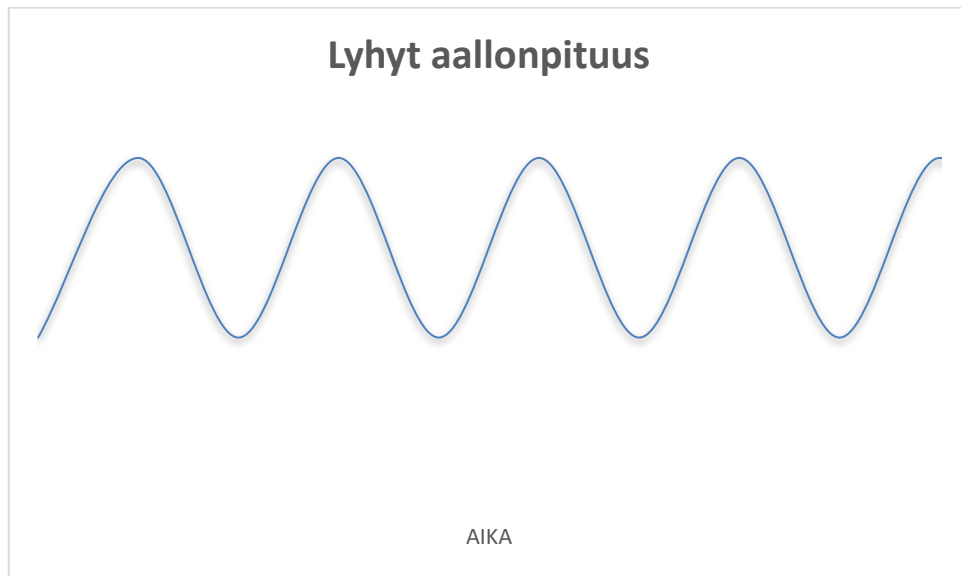
UHF -alue lienee tärkein taajuusalue, koska sillä toimii suurin osa arkipäivän laitteista, esimerkiksi matkapuhelinverkot, pääosa televisiolähetysistä, langaton lähiverkko (WiFi), kotiautomaatio sekä paljon muita laitteita. UHF -taajuudella lähetettäviä taajuusaaltoja rajoittavat muun muassa fyysiset esteet kuten rakennukset, mäet ja metsät. Tämä johtuu siitä, että UHF -aallot kulkevat suoraviivaisesti ilmassa (Kuva 1), eivätkä kimpoa takaisin ionisfääristä vastaanottimeen. Matalimmilla taajuuksilla ionisfääri voi heijastaa radioaaltoja takaisin. Esteiden takia esimerkiksi gsm -tukiasemat rakennetaan korkeisiin radiomastoihin, jotta esteitä olisi mahdollisimman vähän lähettimen ja vastaanottimen välissä. UHF:n kantomatka vaihtelee muutamasta senttimetristä muutamaan kymmeneen kilometriin, lähetystehosta riippuen (WikiBooks 2017, Rose, M 2007).



Kuva 1 suoraviivainen yhteys kahden antennin välillä (Pacific crest 2011).

Taajuuden yksikkö on hertsi (Hz). Taajuudella kuvataan, kuinka monta kertaa ilmiö tapahtuu aikayksikössä. SI-järjestelmässä taajuutta mitataan kertoina sekunnissa.

Aallonpituus määräytyy, kuinka monta kertaa aallonpituus toistuu yhden sekunnin aikana. Eli mitä korkeampi taajuus sitä lyhyempi aallonpituus.



Kaavio 1 Lyhyt aallonpituus.



Kaavio 2 Pitkä aallonpituus.

Radioaallot ovat sähkömagneettista säteilyä ja ne sijoittuvat 3 Hz-300 GHz taajuusalueelle. Radioaaltojen välityksellä voidaan esimerkiksi ohjata etänä laitteita, viestittää matkapuhelimilla sekä lähettää radio- ja tv-lähettyksiä.

Radioaallot etenevät väliaineiden kautta ilmassa ja valonnopeudella tyhjiössä. Radioaaltojen etenemisnopeus riippuu väliaineesta (Livescience 2015).

Radioaallot synnytetään johtamalla vaihtovirtaa johtimeen. Näin johtimeen syntyy värähtelevä magneettikenttä, joka tuottaa radioaaltoja. Esimerkiksi, jos vaihtovirta muuttaa suuntaansa 60 kertaa sekunnissa, tarkoittaa tämä, että radiotaajuus on 60Hz (Tekstenuitleg 2012).

Yleisesti ottaen mitä pienempi taajuus sitä alttiimpi se on häiriötekijöille. Häiriötekijöitä ovat esimerkiksi sähkölaitteet, luonnonilmiöt ja aurinko. Metallikotelo häiriöalttiiden osien ympärillä suojaa häiriöiltä. Nykyaikaiset laitteet osaavat myös suodattaa häiriöitä paremmin kuin vanhat laitteet.

Tässä opinnäytetyössä käytettävät laitteet käyttävät taajuuksiväliä 432 – 438 MHz, joka on suunnattu Suomessa radioamatööri liikenteelle. Markkinoilla kuitenkin on 433 MHz taajuudella toimivia laitteita, joita voi ostaa kotikäyttöön, kuten etähallittavia pistorasioita ja langattomia sääsasemia. Tälle taajuuksivälille on annettu tietyt rajoitteet, jotka koskevat lähettimen tehoa. Nämä rajoitteet ovat sitä varten, etteivät laitteet häiritsisi muita alueella toimivia laitteita. Euroopassa 433 MHz taajuudella toimivia laitteita on jo useita, mutta esimerkiksi Yhdysvalloissa 433 MHz radioliikenne on luvanvaraista ja siellä käytetäänkin kotitalouksissa pääsääntöisesti 900 MHz taajuudella toimivia laitteita (WikiBooks 2017, Viestintävirasto 2016). Taajuusaluejakoa Suomessa (Liite 1).

Tahallista häirintää tai väärinkäytöksiä tapahtuu Suomessa harvoin. Usein kyseessä on rikkoutunut tai väärinkytetty laite, joka häiritsee samalla taajuudella toimivia laitteita. Rikosnimike väärinkäytöksille Suomessa on tietoliikenteen häirintä ja väärinkäytöksiä tarkkailee Suomessa viestintävirasto peilaustauton avulla. Vankeutta tietoliikenteen häirinnästä voi saada enintään kaksi vuotta ja törkeästä tietoliikenteen häirinnästä enintään neljä vuotta vankeutta. Pienimmistä väärinkäytöksistä tuomitaan sakkoihin. Vuonna 2017 on todettu vain yksi luvaton merirosvo- tai piraattiradioksi sanottu radioasema. Piraattiradioasemat ovat vähentyneet huomattavasti, koska nykyisin kuka vain voi perustaa Internettiin radiokanavan. (Rikoslaki 578/1995, Peilausauto 2017).

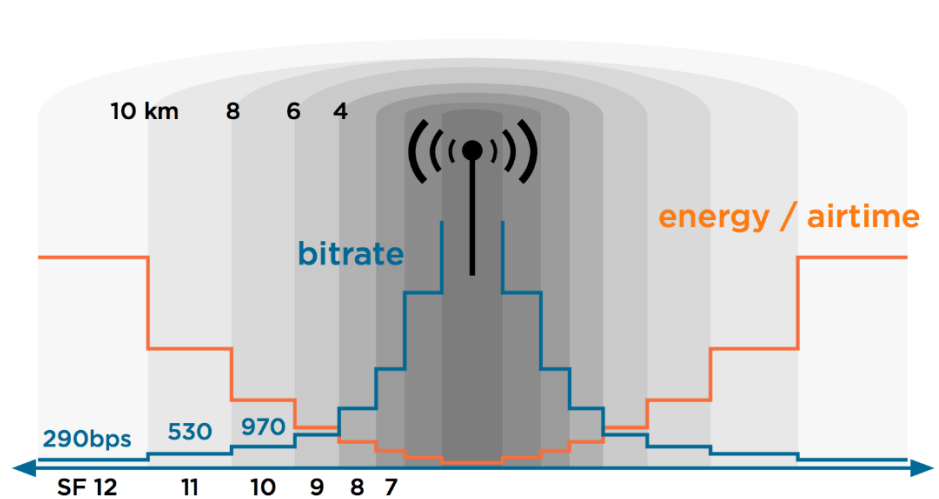
2.1 UHF-taajuuksien vertailu

Vertailussa opinnäytetyö ottaa kantaa UHF-taajuudella toimiviin tiedonsiirtomenetelmiin, jotka ovat yleistyneet maailmalla. Näitä ovat esim. WiFi, ZigBee ja Z-Wave.

UHF-taajuuksilla käytettäviä tiedonsiirtomenetelmiä on vertailtu internetistä löytyviin dokumentaatioihin ja testituloksiin. Testituloksiin vaikuttavia tekijöitä on lähes mahdoton tietää tarkkaan, joten vertailu on suuntaa antava, eikä anna tarkkaa tietoa. Parhaisiin testituloksiin päästäisiin suorittamalla kaikki testit omassa testiympäristössä, joka on muuttumaton testistä toiseen. Tällaisen testiympäristön luominen olisi kuitenkin kallista ja aikaa vievää, joten tässä opinnäytetyössä keskitytään vastaamaan kysymyksiin: Mitkä ovat eri UHF-taajuudella toimivien tiedonsiirtomenetelmien hyvät ja huonot puolet ja mitkä ovat näille soveltuvat käyttökohteet.

2.1.1 UHF-taajuuden kantomatka

Kantomatkaan vaikuttaa useampi tekijä. Tekijöitä ovat vastaanottimen herkkyys, taustakohinan suuruus, lähettimen teho, kuinka vastaanotin ja lähetin muodostavat signaalia sekä kuinka paljon signaalia häviää ilmakehään matkan aikana. Perinteisesti näitä tekijöitä kutsutaan nimellä desibeli, joka helpottaa suorituskyvyn laskemista (Afar communications).



Kuva 2 datasiirtonopeuden vaikutus kantamaan (Innovation Blog).

Yleisesti ottaen mitä pienempi taajuus, sitä suurempi kantomatka, mutta pienemmällä taajuuksilla datasiirtonopeus laskee. Seuraavassa taulukossa on esitelty yleisempien eri langattomien tekniikoiden kantomatkoja suurimmalla sallitulla lähetysteholla Suomessa (25 mW). Kantomatkat ovat suuntaa antavia ja näissä tuloksissa ei ole otettu huomioon lisäantenneja tai tehoantenneja.

Kantomatkojen vertailu:

Tekniikka	Taajuus	Kantomatka	Käyttötarkoitus
ZigBee	868 MHz, 915 MHz, 2.4 GHz	100 m	Automaatio (vähäinen virrankulutus)
Bluetooth	2.4 – 2.485 GHz	10 m – 60 m	Laiteyhteydet jotka vaativat nopeampaa datasiirtonopeutta
WiFi	2.5 GHz, 5 GHz	90 m 40 m	Laiteyhteydet jotka vaativat nopeampaa datasiirtonopeutta
Z-wave	868.42 MHz, 869.85 MHz	100 m	Automaatio
HomeEasy	434 MHz	30 m	Automaatio
LoRa-verkko	434 MHz, 868 MHz, 915 MHz	2 km, (Suunnatulla antennilla, jopa 20 km)	Automaatio (vähäinen virrankulutus, hidas tiedon siirto)

Taulukko 1 Kantomatkojen vertailu (Lora alliance 2017, WikiBooks 2017, How-To geek 2015).

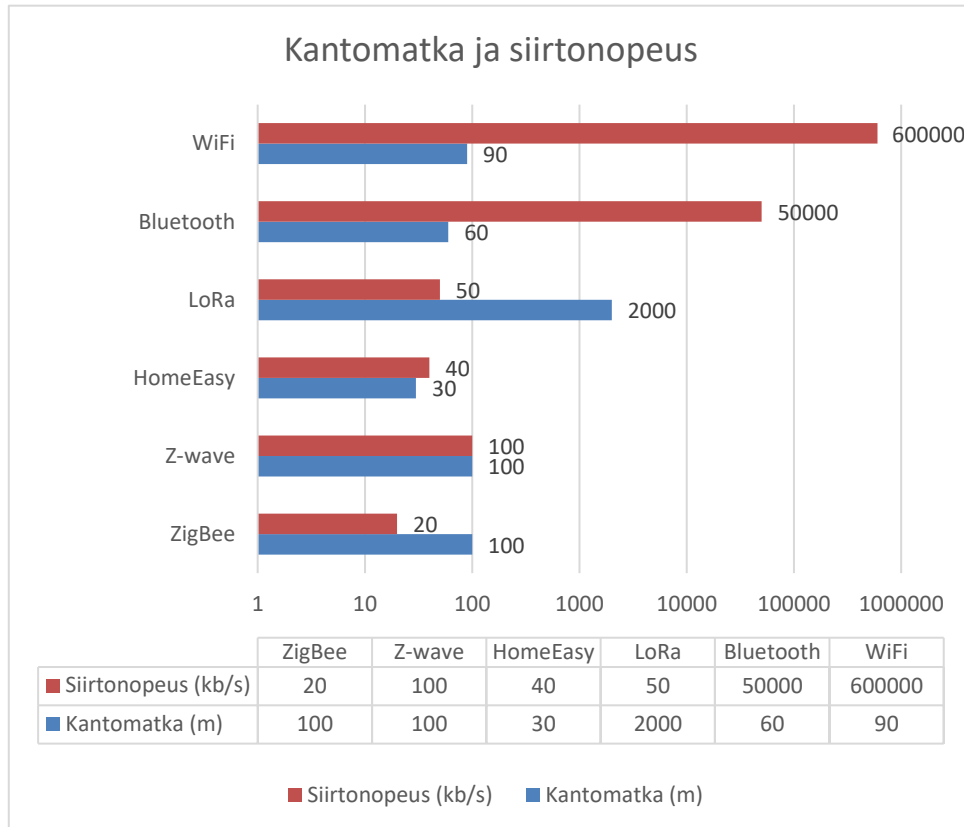
2.1.2 UHF-taajuuksien datasiirtonopeudet

UHF-taajuuksilla toimivien tiedonsiirtomenetelmien datasiirtonopeudet riippuvat taajuudesta ja bittivirrasta (engl. bitrate). Yleisesti ottaen mitä pienempi taajuus sitä hitaampi datasiirtonopeus ja päinvastoin. Seuraavassa taulukossa on esitelty yleisempien eri langattomien protokollien tiedonsiirtonopeuksia. Siirtonopeudet ovat suuntaa antavia.

Siirtonopeuden vertailu:

Tekniikka	Taajuus	Siirtonopeus	Käyttötarkoitus
ZigBee	868 MHz, 915 MHz, 2.4 GHz	20 kb/s, 40 kb/s, 250 kb/s	Automaatio (vähäinen virrankulutus)
Bluetooth	2.4 – 2.485 GHz	25-50 Mbit/s	Laiteyhteydet jotka vaativat nopeampaa datasiirtonopeutta
WiFi	2.5 GHz, 5 GHz	600 Mbit/s 1300 Mbit/s	Laiteyhteydet jotka vaativat nopeampaa datasiirtonopeutta
Z-Wave	868.42 MHz, 869.85 MHz	100 kb/s	Automaatio (vähäinen virrankulutus)
HomeEasy	434 MHz	40 kb/s	Automaatio
LoRa-verkko	434 MHz, 868 MHz, 915 MHz	0.3–50 kb/s	Automaatio (vähäinen virrankulutus, hidas tiedonsiirto)

Taulukko 2 Siirtonopeuden vertailu (Lora alliance 2017, WikiBooks 2017, How-To geek 2015).



Kaavio 3 Kantomatka ja siirtonopeus.

Yllä oleva kaavio (kaavio 3) selventää, kuinka eri tiedonsiirtomenetelmät eroavat toisistaan. Näissä kaavioissa on käytetty suurinta kantomatkaa sekä pienintä datasiirtonopeutta (poikkeuksena LoRa, jossa käytetty suurinta datasiirtonopeutta). Nopeimman tiedonsiirron saavuttaa WiFi ja parhaimman kantomatkan saavuttaa LoRa. Tiedonsiirtonopeudeltaan heikoin on ZigBee ja heikoimman kantomatkan tittelin saa HomeEasy. Näillä tiedoilla ei kuitenkaan voida sanoa mikä tiedonsiirtomenetelmä on paras, koska käyttötarkoitus mihin tiedonsiirtomenetelmää käytetään, on otettava huomioon. Esimerkiksi suuren virrankulutuksen omaava WiFi ei sovellu tiedonsiirtoon missä tarvitaan vähäistä virrankulutusta sekä vähäistä tiedonsiirtonopeutta. Z-Wave taas ei sovellu suuriin tiedonsiirto tarpeisiin, mutta soveltuu hyvin vähään virrankulutusta tarvitseviin käyttökohteisiin.

2.1.3 UHF-taajuuksien esteiden läpäisemiskyky ja käyttökohteet

Läpäisemiskyvyllä tarkoitetaan sitä, miten hyvin taajuus läpäisee esteen. Yleisesti ottaen läpäisemiskyky on sitä parempi mitä alhaisempi taajuus on. Läpäisemiskykyyn vaikuttaa myös tietysti esteen materiaali ja tiheys. Tiheän aallonpituuden omaava taajuus kimpoilee esteestä takaisin päin, kun taas suuren aallonpituuden omaava taajuus läpäisee esteen ja kimpoilee vähemmän esteestä takaisin. Esimerkiksi näkyvän valon taajuus on n. 430-750 THz(Terahertzi) ja kuten kaikki tietävät, valo ei läpäise seinää, mutta 5 GHz (0.005 THz) taajuus läpäisee. Tämä johtuu siitä että, valon taajuuden atomit ovat huomattavasti suurempia kuin 5 GHz taajuuden atomit. (Sciloo 2017, Pacific crest 2011).

Seuraavassa kappaleessa on esitetty ja perusteltu mihin mikäkin UHF-taajuudella toimiva tiedonsiirtomenetelmä soveltuu parhaiten ja mikä olisi sopiva käyttökohde.

Zigbee ja Z-Wave on tarkoitettu lähinnä automaatioon vähäisen virrankulutuksen ja heikon tiedonsiirtonopeuden takia. Signaalin kantomatka ja läpäisemiskyky kattavat suuremman rakennuksen ja täten tämä tiedonsiirtomenetelmä ja taajuus soveltuvat hyvin myös tehtaiden automatisointiin sekä kotitalouksien automatisointiin.

Bluetooth on tarkoitettu nopeaan tiedonsiirtoon ja soveltuu parhaiten laitekytkentöihin, joissa tarvitaan siirtämään paljon dataa lyhyessä ajassa. Näitä käyttökohteita olisi esimerkiksi laiteyhteydet, joissa liikkuu musiikkia tai videokuvaa. Bluetoothin tietoturva on kyseenalainen.

WiFi on ehkä eniten käytetty menetelmä kotitalouksissa sen suuren tiedonsiirtonopeuden takia. Tällä menetelmällä luodaan kotiverkot helposti ja tätä menetelmää tukee suurin osa markkinoilta löytyvistä laitteista, joiden pitää siirtää suuria määriä dataa nopeasti ja tietoturvallisesti.

HomeEasy on menetelmä, jota käyttää moni laite, joka toimii 434 MHz taajuudella. Tämä menetelmä soveltuu lähinnä kotiautomaatiossa tarkoitettuun hitaaseen tiedonsiirtoon, lähinnä valaistuksen hallitseminen. Tämä menetelmä ei sisällä tietoturvaa juuri lainkaan, ja näin ollen mitään arkaluontoista tällä tekniikalla ei kannata siirtää.

LoRa-verkko (LongRange network) on tarkoitettu laiteyhteyksiin pitkillä matkoilla. Vähäisen virrankulutuksen ja pitkän kantaman ansiosta LoRa soveltuu hyvin laitteisiin, jotka lähettävät vähän tietoa ja toimivat akulla. Laitteena voisi esimerkiksi olla maaperän kosteusanturi, kaukana maatilan pellolla.

2.2 Eri tiedonsiirtomenetelmien tietoturva

Tietoturva on tärkeä osa kotiautomaatiota ja useasta kaupan tiskiltä löytyvästä etäohjattavasta sähkölaitteesta puuttuu tämä ominaisuus, tai se on todella heikko. Esimerkiksi suurinta osaa etäohjattavista pistorasioista ei suojata mitenkään ja näin joku voi kuunnella taajuutta ja toistaa taajuudella lähetettäviä käskyjä laitteelle.

Tietoturvan tärkeyttä kotiautomaatiossa ei pidä koskaan väheksyä, jos kodin laitteisiin pääsee käsiksi mistä päin maailmaa tahansa ja tietoturva pettää, voi seuraukset olla vakavat. Otetaan esimerkiksi etähallittavat kiukaat, jotka poistettiin markkinoilta muutamien tulipalojen seurauksena. Jos joku ulkopuolinen pääsee laittamaan kiukaan päälle etänä ja saunassa sattuu-kin olemaan vaatteita kuivamassa, voi tulipalo levitä koko asuntoon. Lisäksi tekijä voisi tehdä kiusaa esimerkiksi, väläyttelemällä valoja tai säätämällä stereoiden äänenvoimakkuuden niin kovalle, että naapurit häiriintyvät. Suuremmassa mittakaavassa, tekijä voisi saada aikaan suuren sähkökatkon, laittamalla päälle jonkin kaupunginosan kaikki etähallittavat sähkölaitteet samalla hetkellä ja näin ylikuormittamalla hetkellisesti sähköverkon. Tietoturva ei siis koske pelkästään sitä henkilöä, joka omistaa laitteen vaan kaikkia alueella asuvia henkilöitä. Kotiautomaatiojärjestelmää rakentaessa on hyvä pitää mielessä se, että jos joku ulkopuolinen pääseekin hallitsemaan laitteita ja mitä hän voi saada aikaan näillä laitteilla.

ZigBee protokolla on monimutkainen ja perustuu "open trust" malliin. Tämä tarkoittaa sitä, että kaikki protokolla pinot luottavat toisiinsa. ZigBeen verkon tietoturvan perustana on kryptatut avaimet. Avaimia on kahdenlaisia. Network Key on 128-bittinen avain, joka jaetaan kaikkien verkossa olevien laitteiden kanssa ja Link key on 128-bittinen avain, joka jaetaan vain kahden laitteen välille. ZigBeen tietoturva on vahva ja luotettava, ja tämä on hyvä ottaa huomioon, kun tarvitaan tietoturvallista automaatiota (Blackhat 2015).

Z-Wave päivitti tietoturva protokollan vuonna 2017, koska edellisessä versiossa oli suuria tietoturva-aukkoja. Uusi tietoturvaprotokolla pitää sisällään AES-128 kryptauksen, Diffie-Hellmann avainvaihtomekanismin ja muita parannuksia, jotka estävät muun muassa "man-in-the-middle" hyökkäyksen.

Z-Wave on nykyään luotettava ratkaisu automaation toteutukseen. (Electronicdesign 2017).

Bluetoothin tietoturva on aina ollut hieman kyseenalainen. Bluetooth on syytä päivittää uusimpaan versioon aina kun mahdollista. Yhteys on syytä pitää päällä vain silloin kun sitä tarvitaan.

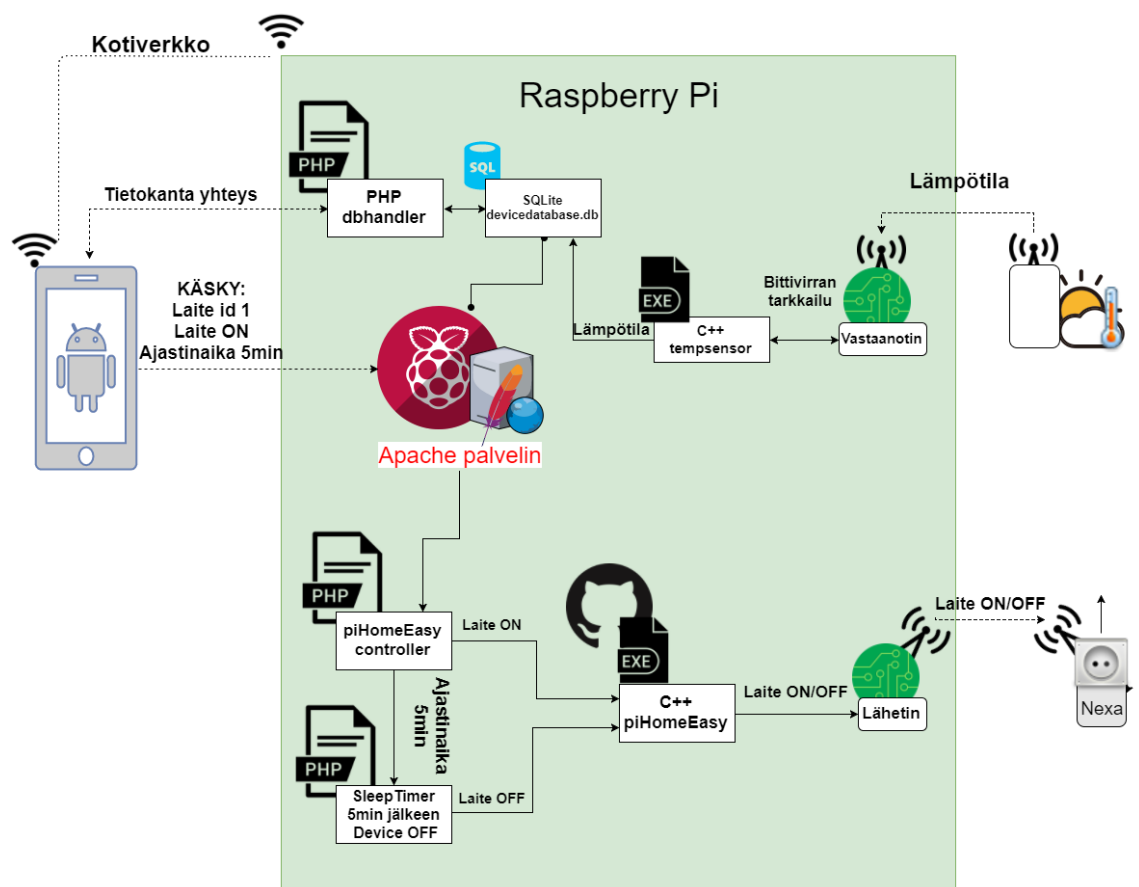
WiFi on pitkälle kehitetty tiedonsiirtomenetelmä, jolla on monta eri tietoturva protokollaa. Oikein käytettynä WiFi on tietoturvallinen ja luotettava tapa siirtää tietoa langattomasti. Suurin tietoturvauhka on käyttäjä itse, esim. salasanat voivat olla liian heikkoja, salasanat ovat kaikkien nähtävillä tai käytettävä protokolla on vanhentunut ja on näin tietoturvariski.

HomeEasy on täysin tietoturvaton. Tätä taajuutta voi helposti kuunnella, koska langatonta viestiä ei kryptata mitenkään, eikä viestin lähettämiseen käytetä avaimia laisinkaan. Tietoturvaa voi hieman parantaa rajoittamalla lähettimen tehoa niin, ettei sen kantama ole muualle kuin tarkoitettuun tilaan.

LoRa-verkon tietoturva on monipuolinen ja luotettava, jos sitä käytetään oikein. LoRa-verkon heikoin kohta on sen avaimien säilyttämisessä ja tämä on otettava huomioon järjestelmää rakentaessa. LoRa-verkosta saa tietoturvallisen, jos se rakennetaan oikein, tiedetään mahdolliset tietoturva-aukot ja niihin osataan varautua. (MWR Labs 2016).

3 KOTIAUTOMAATIO 433 MHZ TAAJUUTTA HYÖDYNTÄEN

Yksi opinnäytetyön aiheista oli luoda kotiautomaatiota 433 MHz taajuutta hyödyntäen ja toimeksiantaja halusi myös nähdä toimivan järjestelmän. 433 MHz taajuudella toimivat laitteet valittiin tähän työhön sen saatavuuden ja edullisen hinnan vuoksi. Kotiautomaatio pitää sisällään 3kpl etäohjattavia pistorasioita sekä yhden sääaseman ja näiden toimintaa ohjaillaan Raspberry Pi 3 -tietokoneella. Järjestelmää on helppo laajentaa tarpeen mukaan.



Kuva 3 Kotiautomaation toiminnan kuvaus.

Käytännönsuudessa luodun kotiautomaatiojärjestelmän toiminnan kuvaus on esitetty yllä olevassa kuvassa (kuva 3), Android puhelin hakee tietokannasta ajastetusti uusimman lämpötilan käyttäen Raspberry Pi:lle luotua tietokantayhteyttä. Lämpötilatietokantaan päivitetään uusin lämpötilasensorilta saatu lämpötila C++ tempensor ohjelmaa käyttäen. Nexa pistorakkeita hallitaan Android -puhelimella lähettämällä tietyille laitteille kolme parametria, jotka ovat id, on/off ja ajastimen aika. PHP sivut käsittelevät nämä parametrit ja suorittavat näiden parametrien asettamat toiminnot C++ piHomeEasy ohjelmassa, joka lähettää tiedot lähettimelle, joka säätelee nexa pistorakkeita päälle ja pois päältä. Näitä toimintoja kuvataan tarkemmin käytännönsuutta käsittelevissä kappaleissa.

Kotiautomaatio ei suinkaan ole täysin automatisoitu älytalo kuten moni näin sen saattaa käsittää. Kotiautomaatiota ovat yksinkertaisetkin liiketunnistimella toimivat valot tai lämpötilan mukaan säätävät lämmittimet. Eli melkein joka kodissa on jollain tavalla käytössä automaatiota.

Kotiautomaation tarkoitus on luoda helpotusta arkeen ja säästää energiaa. Nykyajan kodissa alkaa olla eri laitteita jo niin paljon, että niiden samanaikainen hallitseminen on vaikeaa. Kotiautomaatiolla voidaan keskittää laitteiden hallitseminen yhteen paikkaan ja automatisoida tiettyjä toimintoja. Esimerkiksi kotoa lähdettäessä ei tarvitse kuin painaa yhtä nappulaa niin asunnon valot sammuvat, sähkölaitteet menevät kiinni, lämmitystä lasketaan ja murtohälytinjärjestelmä aktivoituu. Näin säästetään sähköä ja saadaan helpotusta arkeen (luovasähkö 2015).

Tähän opinnäytetyön käytännönsuuteen käytettyjä osia on helposti ja edullisesti saatavilla. Tämän järjestelmän pääpiirteitä ovat laajennettavuus, edullisuus ja käyttötarpeisiin muokattavuus.

Opinnäytetyössä otetaan myös hieman kantaa siihen, kuinka edullista itse rakentaman järjestelmän totuttaminen on verrattuna markkinoilta löytyviin valmiisiin järjestelmiin. Opinnäytetyön käytännön osuuden mukaisen järjestelmän luominen osineen maksaa noin 100 – 120 euroa. Markkinoilla olevien järjestelmien hinnat alkavat 120 eurosta ylöspäin. Verkkokaupoista löytyy myös halvempia järjestelmiä, mutta näiden järjestelmien tietoturva, ominaisuudet ja toimivuus voi vaihdella.

3.1 Raspberry Pi 3 Model B

Raspberry Pi on yhden piirinlevyn tietokone ja ensimmäiset Raspberryt julkaistiin vuonna 2014. Raspberry Pi 3 model B poikkeaa edeltäjästään Raspberry Pi 2 model B:stä paremmalla suorittimella sekä sisäänrakennetulla WLAN- ja Bluetooth -piirillä (Raspberry Pi 2016). Hintaa kyseisellä piirillä on noin 50 euroa. Lisäksi Raspberryyntäytyy ostaa micro sd kortti, johon käyttöjärjestelmä asennetaan. 16 Gb tallennustilalla varustettu class 10 kortti maksaa noin 15 euroa. Raspberry on käytössä myös kansainvälisellä avaruusasemalla ja sillä voi esimerkiksi automattisesti suodattaa mainoksia kotiverkosta, rakentaa robotteja, tehdä pelikonsoleja ja automatisoida lähes kaikkea.

3.2 Langattomat etäohjattavat virtakytkimet

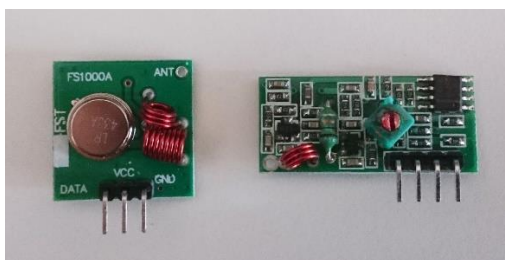
Virtakytkimet toimivat taajuudella 433 MHz ja valmistaja on Nexa. Tätä käytännönsuutta varten päädyin ostamaan 3kpl paketin, johon sisältyy myös kauko-ohjain. Hintaa tälle paketille tuli noin 25 euroa. Virtakytkimet käyttävät HomeEasy protokollaa ja nämä virtakytkimet voidaan parittaa älykkäästi eri laitteiden välillä.



Kuva 4 Nexa virtakytkimet (Clas ohlson).

3.3 433 MHz lähetin ja vastaanotin

Lähettimeä ja vastaanottimia löytyy Internetistä paljon, tähän projektiin tilattu lähetin ja vastaanotin löytyivät suomalaisesta verkkokaupasta. Lähettimen ja vastaanottimen yhteyshinta on noin 8 euroa. Lähetin toimii taajuudella 433.92 MHz eli samalla taajuudella millä projektin hallittavat laitteet. Lähettimen ja vastaanottimen siirtonopeudeksi on merkattu 10 kt/s ja tämä riittää käyttötarkoitukseemme hyvin. Lähettimen käyttöjännite on 3V – 12V, joten Raspberryn 5V -ulostulo riittää käyttöjännitteeksi hyvin ja antaa tarpeeksi kantamaa kattaakseen koko rakennuksen, sillä mitä suurempi käyttöjännite, sitä parempi kantama lähettimellä on. Vastaanotin toimii 5V käyttöjännitteellä, joten tämäkin soveltuu Raspberry Pi:lle loistavasti.



Kuva 5 Lähetin ja vastaanotin.

3.4 Sääasema

Sääasema on aikoinaan ostettu Lidlistä ja toimii 433 MHz taajuudella. Tämä sääasema on yksinkertainen ja näyttää vain ulkolämpötilan. On tärkeää tarkistaa, että sääasema toimii 433 MHz taajuudella ennen ostohetkeä, jotta voimme tarkkailla vastaanottimella sääaseman lähettämää bittivirtaa.

3.5 Raspberry Pi käyttöjärjestelmän ja palvelimen asennus

Raspberry Pi käyttöjärjestelmänä toimii Debian pohjainen Raspbian Jessie käyttöjärjestelmä, joka asennetaan micro sd kortille. Raspbianin voi ladata ilmaiseksi netistä ja asennus kortille on helppoa. Kun Raspbian image on kortille asennettu ja kortti liitetty Raspberryn korttipaikkaan, voi virtaliittimen kytkeä kiinni ja Raspberry käynnistyy. Kun käyttäjäkohtaiset asetukset on asetettu, asennetaan uusimmat päivitykset kirjoittamalla LXTerminaaliiin komento `sudo apt-get update` ja haetaan viimeisimmät versiot komennolla `sudo apt-get dist-upgrade`.

Raspberry Pi palvelimena toimii Apache web server application, joka on helppo asentaa kirjoittamalla Raspbianin LXTerminaaliiin, `sudo apt-get install apache2 -y`. Apachen toimivuuden voi tarkistaa kirjoittamalla nettiselaimen <http://localhost/>. Tämän jälkeen selaimelle pitäisi ilmestyä Apachen oletussivu. Oletussivun tiedostot sijaitsevat `/var/www/html/` kansiossa. Raspberryn uudelleen käynnistäminen voi olla tarpeen, ennen kuin Apache toimii.

PHP asennetaan Raspbianiin kirjoittamalla LXTerminaaliiin käsky `sudo apt-get install php5 libapache2-mod-php5 -y`. PHP hoi-taa Android-sovellukselta tulevat käskyt käyttäen HTTP-protokollaa. PHPn toimivuutta voi testata kirjoittamalla pienen PHP ohjelman. Luo uusi PHP tiedosto, johon kirjoitat yhden komennon `phpinfo()`; (kuva 12), joka tulee tallentaa Apachen html kansioon, joka sijaitsee `/var/www/` kansiossa. Kun sivu on tallennettu kansioon, voit avata tämän selaimella, kirjoittamalla osoitteeksi `http://localhost/-tallentamasitiedostonimi-`. Tämä tulostaa sivulle kaikki tiedot PHP:sta. Jos sivu ei näy voi Raspberry Pi:n uudelleen käynnistäminen olla tarpeen tai PHP sivussa voi olla vikaa. Vian jäljittämiseen on hyvä lukea Apache serverin virhelokia, joka löytyy `/var/log/apache2/` kansioista. Voit avata `error.log` tiedoston käskyllä: `sudo nano /var/log/apache2/error.log` ja selata virheitä nuolinäppäimillä nano editorissa. Nano ohjelman voi lopettaa painamalla näppäimiä `Cntrl + x`.

```

<?php

// Show all information, defaults to INFO_ALL
phpinfo();

// Show just the module information.
// phpinfo(8) yields identical results.
phpinfo(INFO_MODULES);

?>

```

Kuva 6 phpinfo esimerkki (phpnet 2017).

SQLite on relaatiotietokanta johon voi tehdä erilaisia hakuja, muutoksia ja lisäyksiä. Tarvitsemme tietokantaa, jotta voimme tallentaa ja poistaa kytettyjä laitteita ja lämpötilatietoja. SQLite3 asennetaan käskyllä `sudo apt-get install sqlite3` (kuva 13), lisäksi tarvitaan vielä käsky `sudo apt-get install php5-sqlite` (kuva 14), jotta SQLite toimii myös PHP:ssä. Uuden tietokannan luominen tapahtuu käskyllä: `sudo sqlite3 -tietokannan nimi-`. Ota huomioon, että tietokanta luodaan sinne, missä kansiossa tällä hetkellä olet. Esimerkiksi käsky: `pi@raspberrypi: /var/www/html $ sudo sqlite3 devicedatabase.db` luo `html` kansioon `devicedatabase.db` nimisen tiedoston, mihin tietokannan tiedot tallentuvat (kuva 15).

```

pi@raspberrypi: /var/www/html $ sudo apt-get install sqlite3
Luetaan pakettiluetteloita... Valmis
Muodostetaan riippuvuussuhteiden puu
Luetaan tilatiedot... Valmis
Ehdotetut paketit:
  sqlite3-doc
Seuraavat UUDET paketit asennetaan:
  sqlite3

```

Kuva 7 SQLite asennus.

```

pi@raspberrypi: /var/www/html $ sudo apt-get install php5-sqlite
Luetaan pakettiluetteloita... Valmis
Muodostetaan riippuvuussuhteiden puu
Luetaan tilatiedot... Valmis
Seuraavat UUDET paketit asennetaan:
  php5-sqlite

```

Kuva 8 php5-sqlite asennus.

```

pi@raspberrypi: ~/Scripts $ sudo sqlite3 devicedatabase.db
SQLite version 3.8.7.1 2014-10-29 13:59:56
Enter ".help" for usage hints.
sqlite> CREATE TABLE devices (id NUMERIC,deviceon TEXT, deviceoff TEXT,timer NUMERIC);
sqlite> INSERT INTO devices values(1,"on","off",50);

```

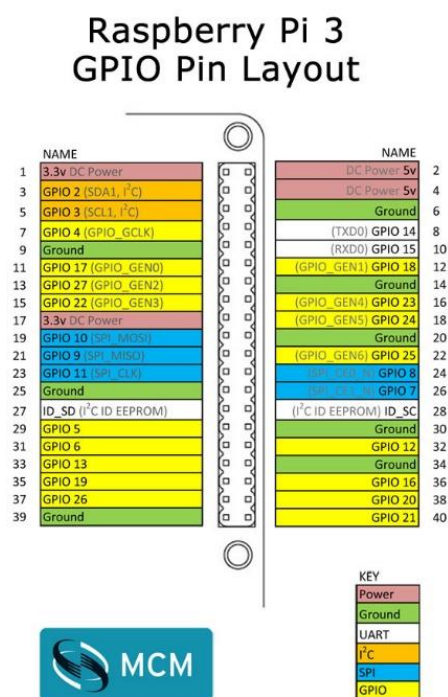
Kuva 9 Uuden tietokannan luominen ja arvojen asettaminen.

Kun tietokanta on luotu, voidaan tietokantaa lisätä uusia tauluja `CREATE TABLE` komennolla ja lisätä arvoja tauluun komennolla `INSERT INTO` (kuva 15).

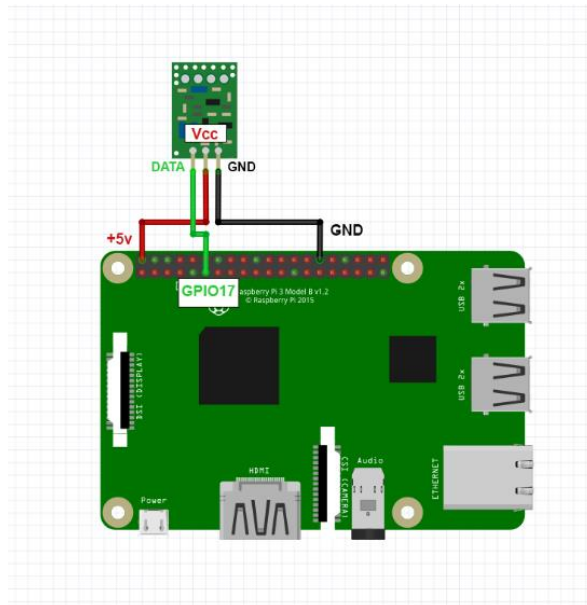
3.6 Lähettimen ja vastaanottimen kytkeminen

Lähetin ja vastaanotin kytketään kiinni suoraan Raspberryn GPIO-liitäntöihin käyttäen liitäntäjohtoja. Johtojen juottamista suoraan kiinni liitäntöihin ei suositella. Lähettimen ja vastaanottimen käyttöjännitteet on hyvä ottaa huomioon ennen kuin ne liitetään. Tässä projektissa käytettiin 5v käyttöjännitteellisiä laitteita, mitkä sopivat Raspberry Pi:lle. On myös hyvä ottaa huomioon, että joidenkin valmistajien vastaanottimet lähettävät suoraan kytketyn käyttöjännitteen dataliitännästä ulos vastaanottaessaan dataa, joten jos laite ei siedä +5v jännitettä GPIO-pinneissä, joudutaan jännitettä laskemaan. Tässä projektissa käytetty vastaanotin laskee data pinnan käyttöjännitettä puoleen kokonais käyttöjännitteestä, eli 5v käyttöjännitteellä, data pinniltä lähtevä jännite input pinniin on 2.5v (Arduinobasics 2014).

Lähettimestä löytyy 3 kytkentäliitäntää: Data joka on tiedon lähettämistä varten, Vcc johon liitetään käyttöjännite ja GND maadoitusta varten. Data liitetään Raspberryn GPIO liitäntään. Tässä projektissa on käytetty GPIO17 pinniä (pinni 11), Vcc liitetään +5v pinniin (pinnit 2 ja 4) ja GND Ground pinniin (pinnit 6, 9, 14, 20, 25, 30, 34, 39), katso alla olevat kuvat 7, 8, 9 ja 10.



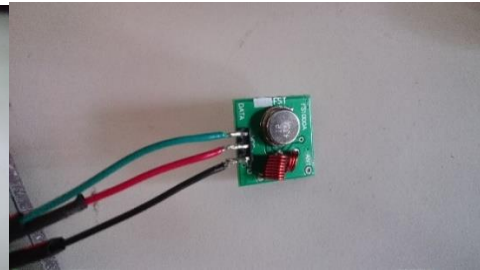
Kuva 10 Raspberry Pi 3 GPIO pinnikartta (mcmelectronics).



Kuva 11 Lähettimen kytkentäkaavio.



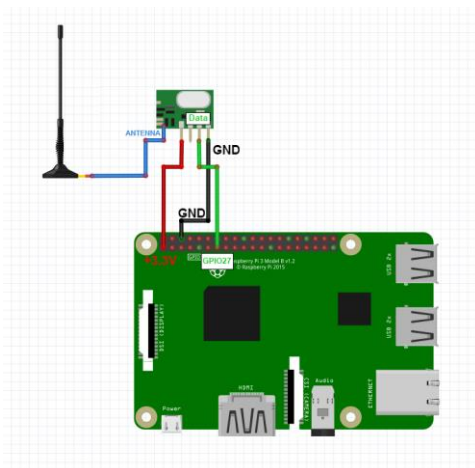
Kuva 12 lähettimen kytkentä Raspberry Pi.



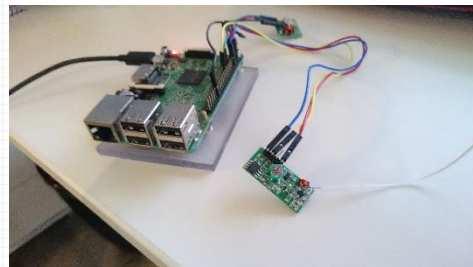
Kuva 13 Lähettimen kytkentä.

Kun tarvittavat kytkennät on tehty, voidaan seuraavaksi kytkeä vastaanotin. Lähettimessä on myös lisäantennin liitännämahdollisuus, tässä projektissa ei tarvittu lisäantennia, koska kantama oli tarpeeksi hyvä ilmankin.

Vastaanottimesta löytyy 4 kytkentäliitännää: 2x Data tiedon käsittelyä varten, Vcc käyttöjännitettä varten ja GND maadoitusta varten. Vcc kytketään Raspberryn +3.3v pinniin ja GND Raspberryn Ground pinniin. Data pinneistä ei tarvitse kytkeä kuin toinen pinni, eikä ole väliä kumman kytkee koska vastaanotin lähettää saman datan molempiin pinneihin. Data pinni kytketään Raspberryn GPIO pinniin, tässä projektissa käytettiin GPIO27 pinniä (Pinni 13). Vastaanottimen voi myös kytkeä +5v liitännään, mutta vaarana on GPIO pinnan vaurioituminen ja tässä projektissa taustakohina kasvoi liikaa ja signaalien tulkinta vaikeutui. Lisäantennilla saadaan kantomatkaa hieman lisättyä.



Kuva 14 Vastaanottimen kytkentäkaavio.



Kuva 15 Raspberry Pi ja vastaanotin.

3.7 Ohjelmakoodit

Tavoitteeseen päästäksemme tarvitsemme vielä muutaman ohjelmakoodi, jotta järjestelmä toimisi halutulla tavalla.

Tarvitsemme vielä:

- PHP sivun joka käsittelee Android-sovellukselta tulevat käskyt (REST-rajapinta)
- PHP sivun joka hallitsee tietokantayhteydet
- PHP sivu ajastimelle, jolla voi ajastaa laitteen sammumaan asetetun ajan jälkeen
- C++ ohjelman joka lähettää käskyt lähettimen kautta seinäpistokkeille
- C++ ohjelman joka tarkkailee 433 MHz taajuutta ja tulostaa sääaseman lämpötilaa käsittelevän bittivirran
- Android-sovellus, joka toimii käyttöliittymänä ja jolla voi ohjalla kytkettyjä laitteita

3.7.1 C++ piHomeEasy ohjelma

Tämä ohjelma osaa ohjalla HomeEasy protokollaa käyttäviä laitteita. Tämä ohjelma on Nenad Bogojevicin luoma ja löytyy ilmaiseksi ladattavaksi osoitteesta <https://github.com/nbogojevic/piHomeEasy>. Tämä ohjelma tarvitsee myös wiringPi ohjelman toimiakseen sekä git-coren, jotta voimme ladata ohjelman github:sta. WiringPi ja git-core asennetaan komennoilla:

```
sudo apt-get install git-core
git clone git://git.drogon.net/wiringPi
cd wiringPi
sudo ./build
```

HomeEasy ohjelma asennetaan hakemalla se ensin github:sta osoitteesta <https://github.com/nbogojevic/piHomeEasy.git>. Tämän jälkeen avataan terminaalilla tiedostosijainti, minne ohjelma on tallennettu ja kirjoitetaan komentoriville komento `make`, joka koostaa ohjelman ja tämän jälkeen suunnistetaan tiedostosijaintiin `/usr/local/bin`, jonka jälkeen annetaan komento `sudo make install` joka asentaa ohjelman käytettäväksi.

Ota huomioon, että jos haluat käyttää eri Raspberry Pi:n GPIO pinniä kuin GPIO 17 (pinni 11), niin joudut muuttamaan koodista pinnin asetuksia. Jos ohjelmat asentuivat ilman ongelmia, nyt laitteiden pitäisi toimia komenolla `sudo piHomeEasy 0 31415 1 on` ja `sudo piHomeEasy 0 31415 1 off`. Ennen on/off komentoa oleva numero on laitteen id, jolla voidaan hallita aina tiettyä laitetta. Id:n arvoksi voi myös asettaa -1, tätä kutsutaan ryhmäkäskeksi, jolla voidaan hallita kaikkia emitteri id:n (31415) laitteita. Ota huomioon, että joudut parittamaan laitteet, ennen kuin käskyt toimivat. Katso laitteidesi ohjeet, jos et tiedä miten parittaminen tapahtuu. Nexa PE-3 laitteiden paritus tapahtuu liittämällä laite pistorasiaan, jolloin laite alkaa vilkuttamaan lediä muutaman sekunnin ajaksi, jolloin laitteelle pitää lähettää ON käsky. Laitteen parittamisen poisto tapahtuu taas samalla lailla, mutta lähettämällä OFF käsky. Eli tässä tapauksessa laite paritetaan laittamalla laite pistorasiaan ja lähettämällä `sudo piHomeEasy 0 31415 1 on` käsky Raspberry Pi:llä. Laite ilmoittaa parittamisen onnistumisen menemällä muutaman kerran päälle ja pois päältä.

3.7.2 PHP REST -rajapinta

Tämä PHP -ohjelma käsittelee Android-sovellukselta tulevat käskyt HTTP -protokollaa käyttäen. Kyseessä on siis REST-rajapinta. REST (REpresentational State Transfer) on lyhyesti selitettynä arkkitehtuurimalli ohjelmointi-rajapintojen toteuttamiseen ja se käyttää HTTP-protokollaa, jossa operaatioina ovat GET, POST, PUT ja DELETE (Paikkatietoikkuna 2012). Tässä PHP -ohjelmassa poimitaan kolme eri parametria, jotka Android-sovellus lähettää. Parametrit ovat laite id, onoff ja timer. Kun parametrit on käsitelty, tiedetään mille laitteelle lähetetään ON tai OFF komento. Timer parametrilla saamme ajastimelle minuutit, ajastimen päätyttyä lähetetään OFF komento piHomeEasy ohjelmalle. Ohjelma voidaan luoda millä tahansa tekstinkäsittelyohjelmalla, Raspbianista kuitenkin löytyy Geany ohjelma, jolla on hyvä kirjoittaa koodia.

```

1
2
3
4 <?php
5 include 'sleepertimer.php';
6 $device = $_GET['device'];
7 $onoff = $_GET['onoff'];
8 $timer = $_GET['timer'];
9 ignore_user_abort(true);
10 set_time_limit(0);
11 ob_start();
12 $message = exec("sudo /home/pi/Downloads/piHomeEasy/piHomeEasy 0 31415 $device $onoff");
13 $text = "Message from exec: ";
14 // $message = exec("ls");
15 print("$text - $message $device $onoff");
16 print("$device $onoff");
17 ob_end_flush();
18 ob_flush();
19 flush();
20 if($timer!=0){
21 sleeper($timer,$device);
22 }
23 // $sleepime = $timer * 60;
24 // sleep($sleepime);
25 // $message1 = exec("sudo /home/pi/Downloads/piHomeEasy/piHomeEasy 0 31415 $device off");
26 // print("$message1");
27
28
29
?>

```

Tila

12:42:17: Tiedosto /home/pi/Scripts/dbconn.php avattu (3).

12:42:17: Tiedosto /home/pi/Scripts/tempsensor.cpp avattu (4).

13:19:59: Tiedosto /home/pi/Scripts/tempsensor.cpp tallennettu.

line: 27 / 29 col: 6 sel: 0 INS SAR mode: Unix (LF) encoding: UTF-8 filetype: PHP scope: tuntematon

Kuva 16 PHP koodi.

Kun ohjelmakoodi on kirjoitettu, tallennetaan ohjelma `/var/www/html` kansioon, jotta Apache pääsee käsiksi koodiin. Jos tiedoston tallentaminen ei suoraan html kansioon onnistu, voidaan tiedosto siirtää toisesta kansioista pääkäyttäjaoikeuksin komennolla `sudo mv -tiedosto sijainti/tiedostonimi- -haluttu tiedostosi-jainti/tiedostonimi- katso kuva 27.`

```
pi@raspberrypi:/var/www/html $ sudo mv ~/Scripts/controller.php controller.php
```

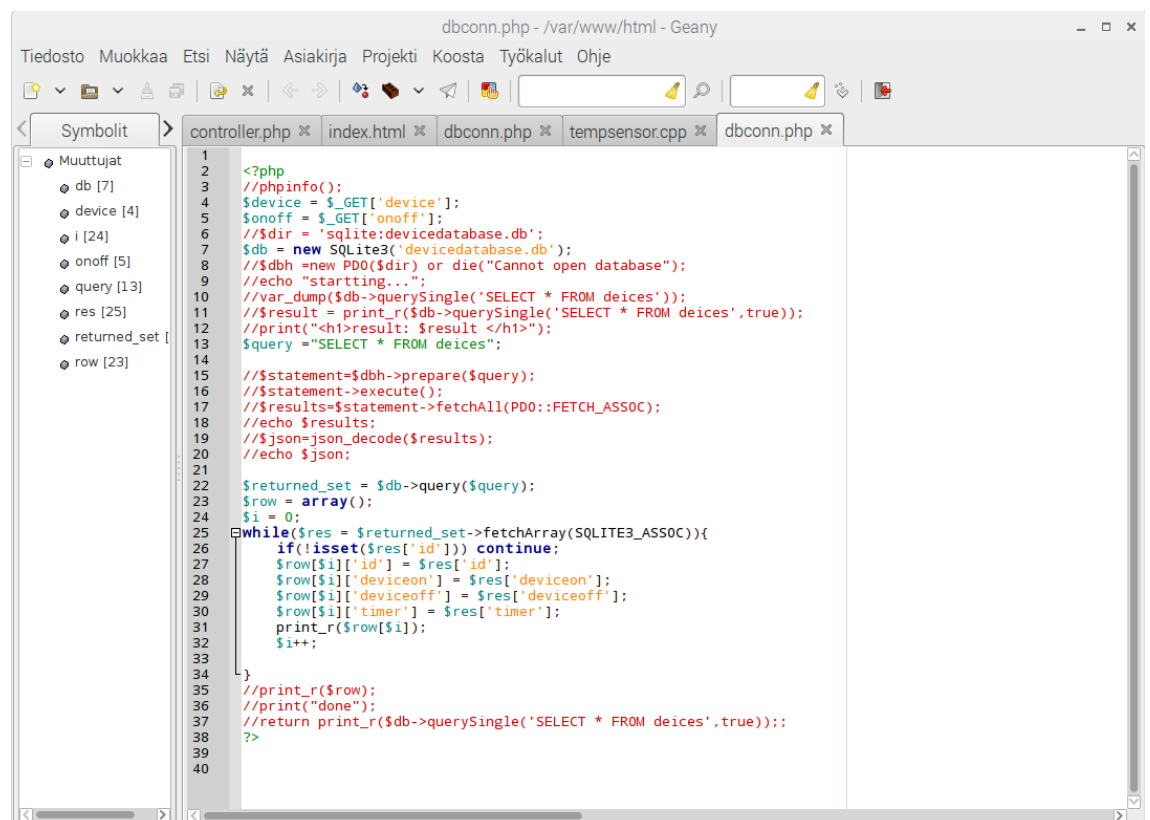
Kuva 17 tiedoston siirtäminen.

Ennen kuin pääsemme testaamaan ohjelman toimivuutta, tarvitaan asettaa käyttöoikeudet ohjelmalle käyttää `exec()` komentoa. Käytännössä `exec()` funktioon kirjoitettu käsky ajetaan terminaalissa. Tässä ohjelmassa lähetämme `exec` funktiolla käskyn `piHomeEasy` ohjelmalle, joka säätelee laitteita kiinni tai päälle. Käyttöoikeudet asetetaan muokkaamalla `/etc/sudoers` tiedostoa lisäämällä `User alias specification` kohtaan rivi `www-data ALL = NOPASSWD: piHomeEasy`. `Sudoers` tiedostoa voi muokata olemalla `etc` kansiossa ja kirjoittamalla terminaaliin komennon `sudo nano sudoers`, kaikki oikeudet voidaan antaa komennolla `www-data ALL=(ALL) NOPASSWD:ALL`

Kun käyttöoikeudet on tallennettu, voidaan ohjelman toimivuutta testata kirjoittamalla Raspberry Pi:n selaimeen <http://localhost/-tiedostonimi-.php/?device=1&onoff=on&timer=1>. Tällä käskyllä laitetaan laite 1 päälle ja ajastimen ajaksi 1 minuutti. Jos tämä käsky ei toimi voidaan palvelimen virhelokista jäljittää missä virhe tapahtuu.

3.7.3 PHP tietokantayhteys

Tämä PHP-ohjelma hallinnoi tietokantaa. Laitteiden tiedot saadaan Android-sovellukselta. Kotiautomaatiojärjestelmä tarvitsee tätä ohjelmaa, jotta voimme lisätä ja poistaa uusia laitteita käyttöliittymää käyttäen.



```

1  <?php
2  //phpinfo();
3  $device = $_GET['device'];
4  $onoff = $_GET['onoff'];
5  // $dir = 'sqlite:deicedatabase.db';
6  $db = new SQLite3('deicedatabase.db');
7  // $dbh = new PDO($dir) or die("Cannot open database");
8  //echo "startting...";
9  //var_dump($db->querySingle('SELECT * FROM deices'));
10 // $result = print_r($db->querySingle('SELECT * FROM deices', true));
11 //print("<h1>result: $result </h1>");
12 $query = "SELECT * FROM deices";
13 // $statement = $dbh->prepare($query);
14 // $statement->execute();
15 // $results = $statement->fetchAll(PDO::FETCH_ASSOC);
16 //echo $results;
17 // $json = json_decode($results);
18 //echo $json;
19
20 $returned_set = $db->query($query);
21 $row = array();
22 $i = 0;
23 while($res = $returned_set->fetchArray(SQLITE3_ASSOC)){
24     if(!isset($res['id'])) continue;
25     $row[$i]['id'] = $res['id'];
26     $row[$i]['deviceon'] = $res['deviceon'];
27     $row[$i]['deviceoff'] = $res['deviceoff'];
28     $row[$i]['timer'] = $res['timer'];
29     print_r($row[$i]);
30     $i++;
31 }
32 //print_r($row);
33 //print("done");
34 //return print_r($db->querySingle('SELECT * FROM deices', true));
35 ?>
36
37
38
39
40

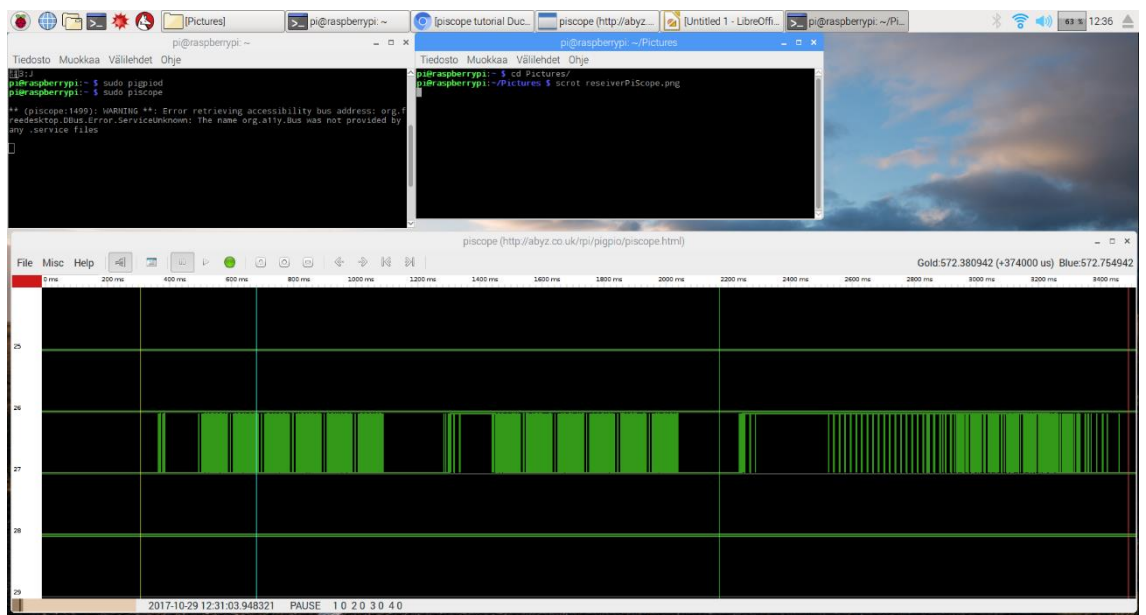
```

Kuva 18 Controller PHP-koodi.

Kun koodi on tallennettu onnistuneesti kansioon /var/www/html/ voidaan testata ohjelman toimivuutta kirjoittamalla selaimeen <http://localhost/-tiedostonimi-.php/?state=add&devicename=kahvinkeitin&timer=30&device=1>. Tällä komennolla lisätään laite nimeltään kahvinkeitin, jonka ajastimeksi on laitettu 30 minuuttia ja laite id:ksi 1.

3.7.4 C++ bittiviran tarkkailu

Tämä ohjelma tarkkailee bittivirtoja 433 MHz taajuudella ja tallentaa saatuja lämpötilatietoja sensorilta. Ohjelman alkuperäinen koodi on kopioitu netistä ja sitä on muokattu omiin tarpeisiin, ohjelman alkuperäisen koodin ja ohjeita löytyy osoitteesta: <https://rayshobby.net/reverse-engineer-wireless-temperature-humidity-rain-sensors-part-1/>. Bittivirran tarkkailuun käytettävä piscope ohjelma näyttää visuaalisesti mitä vastaanotin havaitsee. Piscopen ohjeet löydät osoitteesta: <http://abyz.me.uk/rpi/pigpio/piscope.html> (Kuva 23).

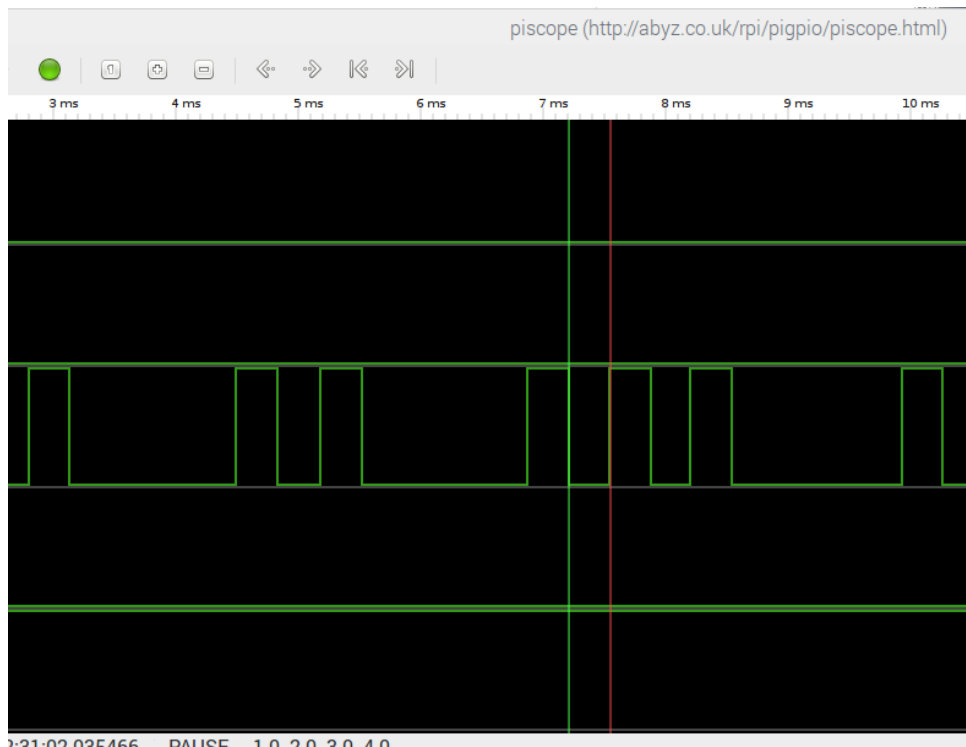


Kuva 19 Bittivirran tulostus piscope ohjelmalla.

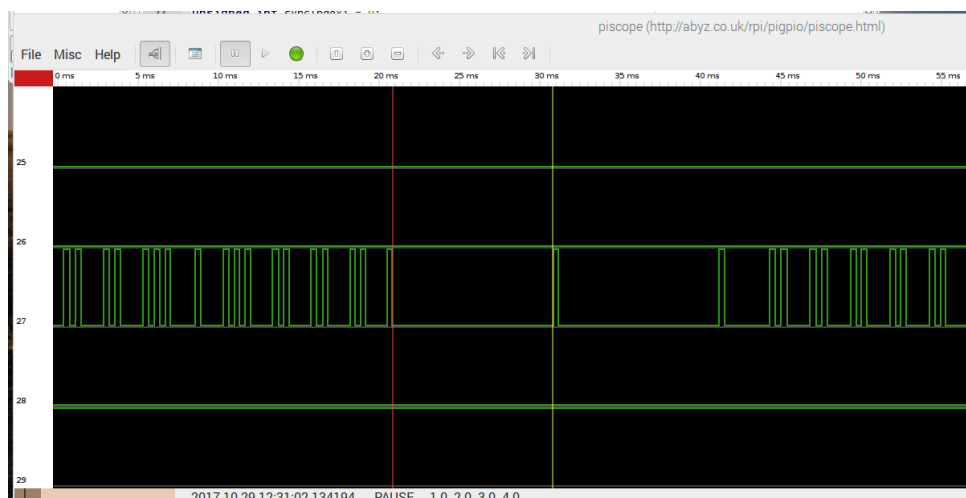
Kuvaa tulkitsemalla voimme päätellä millaisen bittivirran vastaanotin vastaanottaa. Yleensä lähettimet lähettävät saman datan useamman kerran, jotta viesti menee varmasti perille. Tämä selviää myös kuvasta 19, jossa näkyy selvästi kaksi samanlaista bittivirtaa. Jotta ohjelma osaa tulkita bittivirtaa oikein, meidän tulee tietää millaista dataa sääasema lähettää. Tarvitsemme toimivaa ohjelmaa varten neljä muuttujaa (katso kuva 23, koodin rivit 19-26), jotka ovat:

- bittivirran toisistaan erottava SYNC_LENGTH
- kahden eri korkean ja matalan jännitteen välinen aika SEP_LENGTH
- bitti 1 pituus ajassa BIT1_LENGTH
- bitti 0 pituus ajassa BIT0_LENGTH

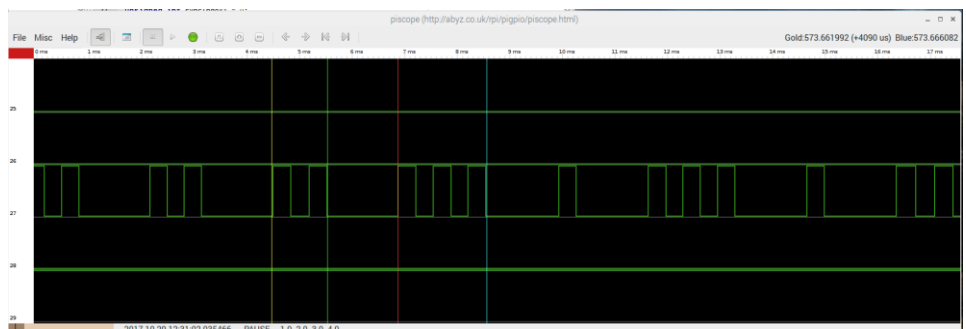
Nämä arvot voidaan tulkita suhteellisen helposti, käyttäen piscope ohjelmaa. Ohjelma näyttää mille GPIO pinnille tulee dataa, tässä esimerkissä vastaanottimen datapinni on kytketty GPIO27 (pinni 13) liitäntään. Suurentamalla aikajanaa voimme päätelle edellä mainitut muuttujat (Katso kuvat 20 – 22). Piscopen ylärivistä näkee kuinka monta milli- ja mikrosekuntia jakso kestää.



Kuva 20 pscope SEP_LENGTH määrittäminen.



Kuva 21 pscope SYNC_LENGTH määrittäminen.



Kuva 22 pscope BIT1_LENGTH & BIT0_LENGTH määrittäminen.

Kuvassa 23 keltaisen ja vihreän viivan välinen aika on BIT1_LENGTH muuttujan arvo ja punaisen ja sinisen viivan välinen aika on BIT2_LENGTH muuttujan arvo.

```

5  L  */
6
7  // ring buffer size has to be large enough to fit
8  // data between two successive sync signals
9
10
11 #include <wiringPi.h>
12 #include <stdlib.h>
13 #include <stdio.h>
14 #include <sqlite3.h>
15 #include <ctime>
16
17
18 #define RING_BUFFER_SIZE 150
19 //Time after data
20 #define SYNC_LENGTH 3200
21 //Time between high and low
22 #define SEP_LENGTH 450
23 //Bit 1 lenght
24 #define BIT1_LENGTH 2000
25 //Bit 0 lenght
26 #define BIT0_LENGTH 1000
27
28 #define DATA_PIN 2 // wiringPi GPIO 2 (P1.13)
29
30 unsigned long timings[RING_BUFFER_SIZE];
31 unsigned int syncIndex1 = 0;
32 unsigned int syncIndex2 = 0;
33 //<-- Make space where value must be, Min max
34 unsigned int seplenghtmin = 250;
35 unsigned int seplenghtmax = 200;
36 unsigned int bitmin = 300;
37 unsigned int bitmax = 600;
38 //-->
39 unsigned int temperature = 0;
40 const char *query;
41
42
43 bool received = false;
44 bool start = false;
45
46

```

The screenshot shows the Geany IDE with the file tempensor.cpp open. The left sidebar displays a symbol list with categories like Funktiot, Makrot, and Muuttujat. The main editor shows the C++ code with line numbers 5 to 46. The status bar at the bottom indicates the current line is 45 of 263, and the file is encoded in UTF-8.

Kuva 23 Muuttujat tempensor ohjelmassa.

Näiden neljän muuttujan lisäksi on määriteltävä neljä muuta muuttujaa, jotta voimme antaa BIT1_LENGTH, BIT0_LENGTH ja SEP_LENGTH:lle toimintasäteen (Katso kuva 23, Koodin rivit 33 - 37). Toimintasäde on asetettava, koska Raspberryn kello ei ole tarpeeksi tarkka toimiakseen mikrosekunnin tarkkuudella, jota koodissa käytetään. Toimintasäde takaa sen, että ohjelma osaa eritellä BIT1 ja BIT0 arvot.

Ohjelma tarkkailee bittivirtaa ja jos muuttujilla asetetut arvot toteutuvat, ohjelma muuttaa bitit luettavaan muotoon ja tallentaa arvon tietokantaan. Tämä ohjelma tallentaa arvon celsiuksena. Tietokantaan tallentaminen tapahtuu avaamalla tietokantayhteys (kuva 24, Koodi rivi 208) ja tallentamalla lämpötila-arvo, päivämäärä ja kellonaika tietokantaan (katso kuva 24).


```

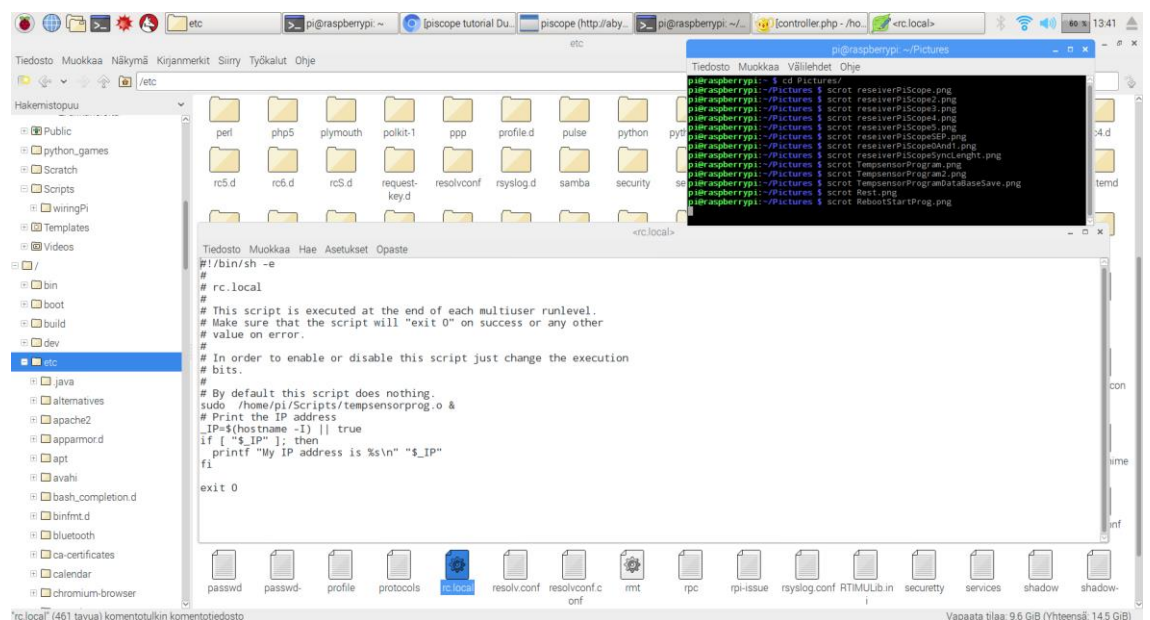
pi@raspberrypi:~/Scripts $ g++ -o tempsensorprog.o tempsensor.cpp -lwiringPi -l
sqlite3
pi@raspberrypi:~/Scripts $ ./tempsensorprog.o
Opened database successfull
Starting1...
syncCount: 1
syncCount: 2
changeCount: 84 != 74
syncCount: 1
syncCount: 2
changeCount: 60 != 74
syncCount: 1
syncCount: 2
changeCount: 102 != 74
syncCount: 1
syncCount: 2
changeCount: 78 != 74
syncCount: 1
syncCount: 2
changeCount: 79 != 74
syncCount: 1
syncCount: 2
changeCount: 74 True!
0011101110000001111001111100000000SYNC
24 C 76 F
done!
^C
pi@raspberrypi:~/Scripts $

```

Kuva 25 ohjelman koostaminen ja ajaminen terminaalissa.

Lopuksi ohjelma on hyvä lisätä käyttöjärjestelmän käynnistyessä aloitettaviin ohjelmiin. Ohjelman tiedostopolku lisätään rc.local tiedostoon, joka löytyy /etc/ kansioista.

Lisää tiedostoon rivi sudo /home/pi/-ajettavan tiedoston polku ja nimi- esim. sudo /home/Scripts/tempsensorprog.o (kuva 26).

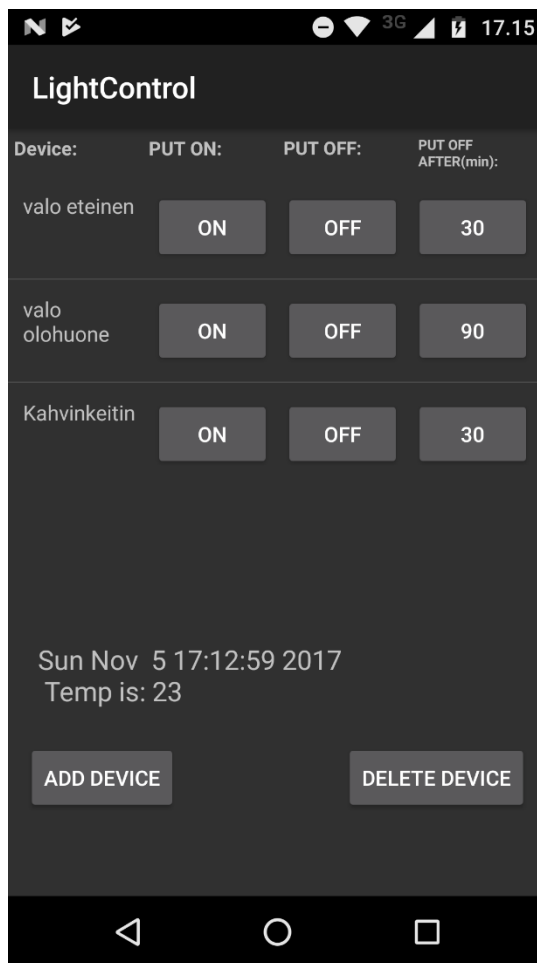


Kuva 26 Käyttöjärjestelmän käynnistyksen yhteydessä ajattavat ohjelmat.

3.7.5 Android-sovellus

Android-sovellus on tehty Microsoftin Visual Studio 2017 ohjelmalla ja kirjoitettu C# ohjelmointikielellä. Android-sovelluksella hallinnoidaan liitetyjä laitteita ja se ilmoittaa käyttäjälle ulkolämpötilan sensorilta. Sovellus sisältää yhdeksän luokkaa jotka ovat:

- MainActivity.cs. Tämä luokka on pääluokka, joka ajetaan sovelluksen käynnistettäessä, luokka listaa olemassa olevat laitteet ja viimeisimmän lämpötilatiedon
- GetDevices.cs, tämä luokka hakee liitetyt laitteet Raspberry pi:n tietokannasta
- AddDevices.cs lisää asetettuja laitteita Raspberryn tietokantaan
- CustomListAdapter.cs on luokka, joka määrittää kuinka sovelluksen laitteet listataan näytölle ja käsittelee listan nappuloiden painallukset
- DBDeviceHandler.cs luokalla voidaan poistaa, lisätä ja hakea viimeisimmät lämpötilatiedot raspberry pi:ltä
- devicesItem.cs luokka on rakentajaluokka, jolla kerrotaan ohjelmalle millainen Raspberry pi:n taulun rakenne on
- HttpWebRequestHandler.cs on luokka, jolla herätetään "fire and forget" menetelmällä Resti kutsu annettujen parametrien mukaan, eli tällä luokalla lähetetään käskyjä laitteille ja asetetaan laitteelle ajastin sekä haetaan lämpötila arvot
- JsonObjects.cs on rakentajaluokka, jolla kerrotaan ohjelmalle mitä Restin palauttama Json vastaus pitää sisällään
- LocalOnClickListener luokalla hallinnoidaan käyttäjän tekemiä klikkauksia



Kuva 27 LightControl sovellus.

Lukijan on syytä tietää, kuinka Rest -rajapinnan kanssa keskustellaan, jotta koodia voidaan helposti muokata tarpeiden mukaan. Rest -rajapinnalle lähetetään parametrit, jotka PHP-sivu tulkitsee ja serveri lähettää vastauksen takaisin.

```

LightControl | LightControl.HttpWebRequestHandler | SubmitUrlPrivate(string url)
25  this.context = context;
26
27
28  public async Task<string> webRestHandler(string deviceId, string deviceName, string deviceState, string timer, string switchcase)
29
30  string temp = "";
31  string url;
32  Log.Debug("WebHandler", "Starting " + switchcase);
33  try
34  {
35      switch (switchcase)
36      {
37          case "add":
38              url = "http://192.168.10.54/dbhandler.php?device=" + deviceId + "&deviceName=" + deviceName + "&timer=" + timer + "&state=add";
39              break;
40          case "delete":
41              url = "http://192.168.10.54/dbhandler.php?device=" + deviceId + "&state=delete";
42              break;
43          case "control":
44              url = "http://192.168.10.54/controller.php?device=" + deviceId + "&onoff=" + deviceState + "&timer=" + timer;
45              break;
46          case "temp":
47              url = "http://192.168.10.54/fetchtemp.php";
48              break;
49          default:
50              url = "http://192.168.10.54/fetchall.php";
51              break;
52          }
53      }
54
55      /**<-FIRE-AND-FORGET->*/
56      Console.WriteLine("Urli= " + url);
57      temp = await SubmitUrl(url);
58  }

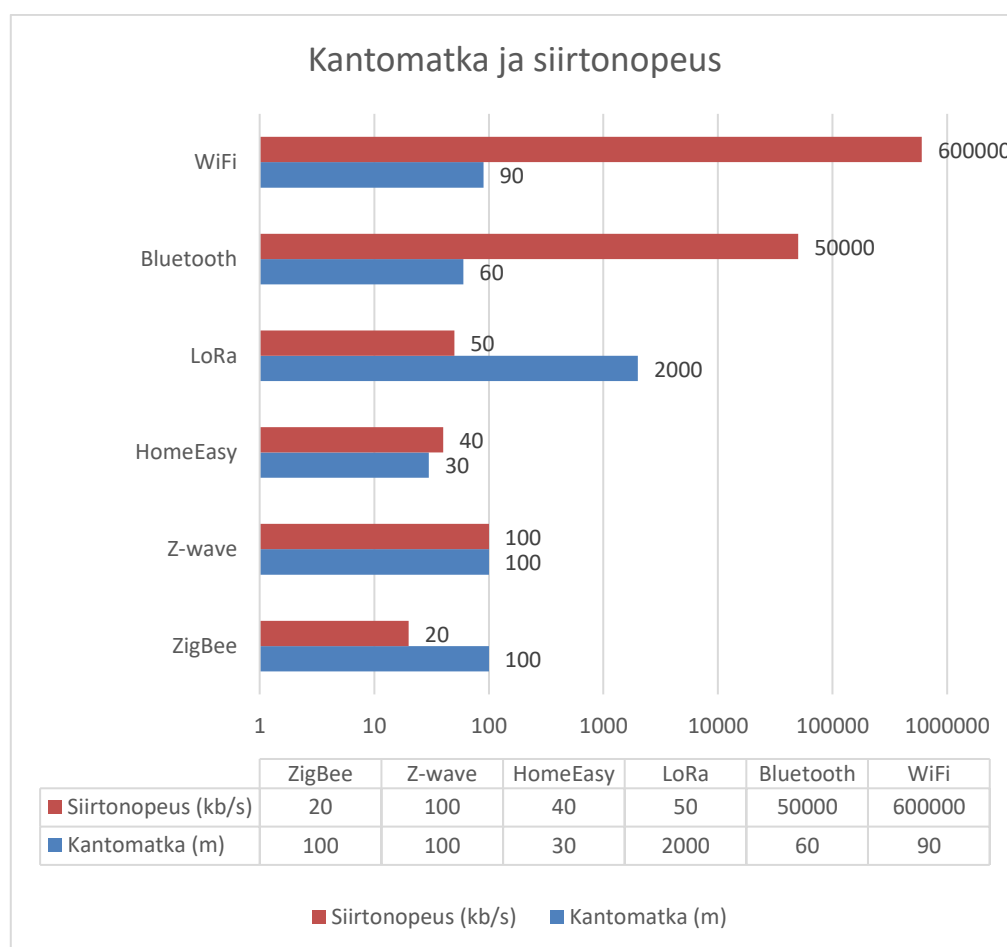
```

Kuva 28 URL-muodostus.

Android-sovellus muodostaa halutun osoitteen `HttpRequestHandler.cs` sivulla, `add`, `delete` ja `control` osoitteet eivät palauta mitään, joten näitä käytetään vain laitteiden hallinnointi tarkoituksissa. `temp` palauttaa viimeisimmän lämpötilan, joka on saatu langattomalta sääasemalta. Sovellus on ladattavista <https://github.com/antti88/LightControl>, ohjelma on kuitenkin vielä kesken, joten ohjelmassa voi ilmetä ongelmia.

4 TULOKSET JA POHDINTA

Opinnäytetyön tulokset eri tiedonsiirtotekniikoiden kantomatkoista ja datasiirtonopeuksista perustuvat Internetistä löytyviin mittauksiin ja teoreettisiin laskelmiin. Tuloksena opinnäytetyö selvitti onnistuneesti, mitkä ovat UHF-taajuudella toimivien yleisimpien tiedonsiirtomenetelmien käyttökohteet, vahvuudet ja heikkoudet. Käytännönsuosus todensi, kuinka UHF-taajuudella siirretään ja luetaan viestejä. Toimeksiantajan tavoitteena oli saada uutta tutkimustietoa sekä kasata jo olemassa olevaa tietoa UHF-taajuudella toimivista tekniikoista ja tässä opinnäytetyö onnistui.



Kaavio 4 Kantomatka ja siirtonopeus.

Opinnäytetyössä tutkittiin eri tiedonsiirtomenetelmien siirtonopeuksia ja kantomatkoja (kaavio 4). Suurimman tiedonsiirtonopeuden saavuttava tiedonsiirtomenetelmä oli WiFi ja suurimman kantomatkan saavuttaa LoRa.

Zigbee ja Z-Wave on tarkoitettu lähinnä automaatioon vähäisen virrankulutuksen ja heikon tiedonsiirtonopeuden takia. Signaalin kantomatka ja läpäisemiskyky kattavat suuremman rakennuksen ja täten tämä tiedonsiirtomenetelmä ja taajuus soveltuvat hyvin myös tehtaiden automatisointiin sekä kotitalouksien automatisointiin.

Bluetooth on tarkoitettu nopeaan tiedonsiirtoon ja soveltuu parhaiten laitekytkentöihin, joissa tarvitaan siirtämään paljon dataa lyhyessä ajassa. Näitä käyttökohteita olisi esimerkiksi laiteyhteydet, joissa liikkuu musiikkia tai videokuvaa. Bluetoothin tietoturva on kyseenalainen.

WiFi on ehkä eniten käytetty menetelmä kotitalouksissa sen suuren tiedonsiirtonopeuden takia. Tällä menetelmällä luodaan kotiverkot helposti ja tätä menetelmää tukee suurin osa markkinoilta löytyvistä laitteista, joiden pitää siirtää suuria määriä dataa nopeasti ja tietoturvallisesti.

HomeEasy on menetelmä, jota käyttää moni laite, joka toimii 434 MHz taajuudella. Tämä menetelmä soveltuu lähinnä kotiautomaatiossa tarkoitettuun hitaaseen tiedonsiirtoon, lähinnä valaistuksen hallitseminen. Tämä menetelmä ei sisällä tietoturvaa juuri lainkaan, ja näin ollen mitään arkaluontoista tällä tekniikalla ei kannata siirtää.

LoRa-verkko (LongRange network) on tarkoitettu laiteyhteyksiin pitkillä matkoilla. Vähäisen virrankulutuksen ja pitkän kantaman ansiosta LoRa soveltuu hyvin laitteisiin, jotka lähettävät vähän tietoa ja toimivat akulla. Laitteena voisi esimerkiksi olla maaperän kosteusanturi, kaukana maatalan pellolla.

Kaikilla tiedonsiirtomenetelmillä on omat heikkoutensa ja vahvuutensa. Tiedonsiirtomenetelmä on aina valittava käyttötarkoituksen mukaan. Käyttötarkoitusta kartoittaessa on huomioitava virrankulutus, tiedonsiirtonopeus ja kantomatka. Esimerkiksi WiFi:llä on hyvä toteuttaa kotiverkkoja, jossa virrankulutuksella ja kantomatalla ei ole suurta vaikutusta. LoRa verkolla saavutetaan puolestaan pitkiä kantomatkoja, mutta tiedonsiirtonopeus on todella pieni. Zigbee, Z-Wave ovat hyviä puolestaan automaatio tarkoituksiin. Tietoturva on myös tärkeä osa järjestelmää ja tämä tulee huomioida järjestelmää rakentaessa.

4.1 Markkinoilta löytyvien kotiautomaatiojärjestelmien vertailu

Markkinoilta löytyy kotiautomaatiojärjestelmiä, joita voi kuluttaja asentaa kotiin tai asentaa itse. Järjestelmien hinta määräytyy oikeastaan monipuolisuuden mukaan. Esimerkiksi Fortumin Fiksu kotiautomaatiojärjestelmän hinta alkaen on 730 € + 9.90 € palvelumaksu kuukausittain ja asennus maksaa 550 €. Tällä järjestelmällä pystyy säätämään asunnon lämpötilaa ja luoda lämmitysprofiileja, jotka säätävät automaattisesti asunnon lämpötilaa vuorokauden ajasta riippuen. Fortumin Kotinäyttö puolestaan näyttää sähkönkulutuksen reaaliajassa. Tämä järjestelmä maksaa 198 €.

Opinnäytetyössä toteutettu kotiautomaatiojärjestelmän kustannukset olivat noin 120 €. Tätä järjestelmää pystyy laajentamaan HomeEasy protokollaa käyttävillä laitteilla, jotka voidaan parittaa lähettimen kanssa. Tähän järjestelmään voidaan myös lisätä liiketunnistimia, mutta tämä vaatii hieman muutoksia ohjelmaan, joka tulkitsee vastaanottimelle saapuvia viestejä. Tämän järjestelmän luominen on kuitenkin hankalaa sekä se vaatii käyttäjältä tietämystä ohjelmoinnista, taajuuksista ja palvelimista.

Cozify Hub on puolestaan hieman samanlainen ratkaisu, mitä tässä opinnäytetyön käytännönsuudessa toteutettiin. Cozify on yksi reititin, joka hallitsee monen eri tuotevalmistajan laitteita. Cozifyllä voi hallita esimerkiksi valoja, sähkölaitteita, kameroita, tarkkailla virrankulutusta, liiketunnistimia ym. ja tämän järjestelmän hinta n. 250 €. Valmiit Cozify paketit, joissa tulee valmiina liitettäviä laitteita, maksavat 299 € - 399 € kpl. Paketteja ovat mm. Tunnelma, Mökkiturva ja Turvaa kotisi paketti. Cozify on Suomessa suunniteltu ja valmistettu.

TaHoma on myös keskitetty järjestelmä, jolla voi hallita laitteita etänä. Tämä järjestelmä ei kuitenkaan tue yhtä monta eri valmistajaa kuin Cozify Hub. Tämän järjestelmän hinta on 349 € ja esimerkiksi yksi liiketunnistin sisäkäyttöön maksaa 79 €, joten älykodin rakentaminen näillä laitteilla tulisi kuluttajalle kalliiksi.

TellStick Net gateway on halpa reititin, joka toimii 433,92 MHz taajuudella. TellStickiin on helppo lisätä laitteita ja asettaa ajastustoimintoja. TellStickin hinta on noin 85 euroa, tämä ei kuitenkaan sisällä yhtään laitetta, jota ohjata. TellStickille löytyy myös Android ja iPhone -sovellukset, joilla laitteita voi ohjaila ja tarkkailla. Opinnäytetyössä toteutettuun järjestelmään verrattuna samanlaisen järjestelmän luominen, tulisi TellStickillä samantaiseksi, eli noin 120 euroa. TellStick taipuu kaikkeen ja enempäänkin, kuin opinnäytetyössä toteutettu järjestelmä. TellStick on hyvä vaihtoehto vähemmällä vaivalla kotiautomaatiota haluaville.

4.2 Pohdinta

Opinnäytetyössä saatiin vastaukset tutkimuskysymyksiin. Tutkimustulosta opinnäytetyö kasasi eri lähteistä ja käytännönoisuus osoitti, kuinka viestejä siirretään 433 MHz taajuudella. Tarkempia testituloksia eri tiedonsiirtomenetelmistä ei kuitenkaan ollut mahdollista toteuttaa. Resursseja luoda täysin muuttumaton testiympäristö, missä olisi testattu eri tiedonsiirtomenetelmiä, olisi ollut lähes mahdotonta toteuttaa. Testiympäristön täytyisi olla muuttumaton testistä toiseen sekä sään vaikutusta olisi mahdoton tietää tarkalleen eri testeissä, koska radioaallot kulkevat väliaineiden kautta ilmassa. Lisäksi testiympäristön pitäisi olla yli 2 kilometriä pitkä sekä täysin esteetön. Olen kuitenkin sitä mieltä, että opinnäytteeni antaa lukijalleen hyvän yleiskuvan eri tiedonsiirtotekniikoista ja mitä on otettava huomioon näitä tekniikoita käytettäessä. Tietoperustana on käytetty Internetistä löytyvää materiaalia, jotka pääsääntöisesti perustuvat teoreettisiin laskelmiin. Todelliset katomatkat ja tiedonsiirtonopeudet varmasti jäävät alemiksi, kuin teoreettiset arvot ja tämä on hyvä ottaa huomioon tekniikoita käytettäessä.

Opinnäytetyön käytännönoisuus puolestaan tuo lukijalle tarkempaa tietoa, kuinka tiedonsiirtotekniikkaa voi käyttää siirtämään viestejä ja hallitsemaan kodin sähkölaitteita. Käytännönoisuus on kuitenkin haastava ja lukijalla voi olla hankala toteuttaa vastaavaa järjestelmää. Opinnäytetyössä on kuitenkin viitattu mistä saa lisää tietoa eri vaiheista, mitä käytännönoisuudessa käytettiin. Kannustankin lukijaa ottamaan ensin selvää automaation ja ohjelmoinnin perusasioista, ennen kuin aloittaa rakentamaan omaa kotiautomaatiojärjestelmää.

Opinnäytetyöhöni olen tyytyväinen, se antaa hyvän yleiskuvan lukijalle eri tiedonsiirtotekniikoista ja todentaa kuinka viestejä lähetetään 433 MHz taajuudella.

5 YHTEENVETO

Opinnäytetyön tavoitteena oli selvittää mitkä ovat UHF-taajuudella toimivien yleisempien tiedonsiirtomenetelmien käyttökohteet, vahvuudet ja heikkoudet. Käytännönsuudessa pyrittiin todentamaan, kuinka UHF-taajuudella siirretään ja luetaan dataa. Toimeksiantajan tavoitteena oli saada uutta tutkimustietoa ja kasata jo olemassa olevaa tietoa UHF-taajuudella toimivista tekniikoista.

Tutkimuskysymys mihin opinnäytetyössä vastattiin: Mitkä ovat eri UHF-taajuudella toimivien tiedonsiirtotekniikoiden vahvuudet, heikkoudet ja käyttökohteet?

Opinnäytetyö kertoi lukijalle mitä tarkoittaa taajuus, radioaallot ja miten näitä käytetään Suomessa sekä otti kantaa myös eri tiedonsiirtotekniikoiden tietoturvaan ja häiriötekijöihin. Opinnäytetyö myös selvittää millaisia ja minkä hintaisia valmiita kotiautomaatio ratkaisuja markkinoilta löytyy.

Käytännönsuus todentaa kuinka viestejä lähetetään 433 MHz lähettimellä ja kuinka viestejä vastaanotetaan vastaanottimella. Käytännönsuus myös opastaa käyttäjää luomaan oman kotiautomaatiojärjestelmän sekä kertoo kuinka paljon kotiautomaatiojärjestelmän osat kustantavat.

Yksi suurimmista ongelmista kotiautomaation teossa oli lyhyt kantama vastaanottimessa. Sääaseman täytyi olla noin 50 senttimetrin päässä vastaanottimelta, jotta lämpötilaa pystyi lukemaan. Luultavasti vastaanotin on vaurioitunut tai on muuten vain heikkolaatuinen. Vastaanotinta kokeiltiin myös suuremmalla käyttöjännitteellä, mutta tällöin taustakohina kasvoi liian suureksi, eikä bittivirtaa voitu enää tulkita. Ongelmana oli myös löytää tietoa, kuinka Raspberry saadaan keskustelemaan laitteiden kanssa käyttäen piHomeEasy ohjelmaa sekä kuinka bittivirtaa kuunnellaan. Näihin ratkaisu kuitenkin löytyi testailemalla ja etsien tietoa internetistä.

Ongelmia oli myös luotettavan tiedon etsiminen Internetistä. Internet on pullollaan eri tietolähteitä ja näistä luotettavan tiedon etsiminen on haastavaa ja vaikeaa. Olen kuitenkin pyrkinyt todentamaan tietojen paikkaansa pitävyyden vertailemalla useampaa lähdettä.

LÄHTEET

Afar communications. Kuinka kauas ja kuinka nopeasti. Haettu 8.5.2017 osoitteesta <http://www.afar.net/tutorials/how-far/>

Arduinobasics (2014). 433 MHz RF module with Arduino Tutorial 1. Haettu 8.5.2017 osoitteesta <http://arduinobasics.blogspot.fi/2014/06/433-mhz-rf-module-with-arduino-tutorial.html>

Blackhat (2015). ZigBee Exploited 2015. Haettu 30.7.2017 osoitteesta <https://www.blackhat.com/docs/us-15/materials/us-15-Zillner-ZigBee-Exploited-The-Good-The-Bad-And-The-Ugly-wp.pdf>

Clas ohlson (2017). Nexa PE-3 kaukokytkinsarja 2017. Haettu 8.10.2017 osoitteesta <http://www.clasohlson.com/fi/Kaukokytkinsarja-3-kpl-Nexa-PE-3/36-4602>

Digitaltrends (2017). Z-Wave just made your smart home devices safer via a new security protocol. Haettu 30.7.2017 osoitteesta <https://www.digitaltrends.com/home/z-wave-security-protocol/>

Electronicdesign (2017). S2's impact on Z-Wave and IoT security 17.1.2017. Haettu 30.7.2017 osoitteesta <http://www.electronicdesign.com/embedded-revolution/computers-101-can-you-identify-these>

How-To geek (2015). Wi-Fi 2.4GHz ja 5GHz eroavaisuudet 2015. Haettu 8.5.2017 osoitteesta <https://www.howtogeek.com/222249/whats-the-difference-between-2.4-gHz-and-5-gHz-wi-fi-and-which-should-you-use/>

Innovation Blog (2016). LoRa, the Internet of Things 25.1.2016. Blogijulkaisu haettu 8.10.2017 osoitteesta <https://blog.surf.nl/en/lora-the-internet-of-things/>

Lora alliance (2017). What is LoRa 2017. Haettu 8.5.2017 osoitteesta <https://www.lora-alliance.org/what-is-lora>

Livescience (2015). Mitä ovat radioaallot, Jim Lucas 6.4.2015. Haettu 8.5.2017 osoitteesta <http://www.livescience.com/50399-radio-waves.html>

Luovasähkö (2015). Mikä ihmeen kotiautomaatio 1.4.2015. Haettu 11.5.2017 osoitteesta <http://luovasahko.fi/mika-ihmeen-kotiautomaatio/>

Mcmelectronics (2016). Raspberry Pi 3 GPIO pin layout 26.2.2016. Haettu 25.5.2017 osoitteesta <http://blog.mcmelectronics.com/post/Raspberry-Pi-3-GPIO-Pin-Layout#.WSaSmmiyiM8>

MWR Labs (2016). LoRa Security Building a Secure LoRa Solution, Robert Miller. 22.3.2016. Haettu 8.10.2017 osoitteesta <https://labs.mwrinfosec.com/assets/BlogFiles/mwri-LoRa-security-guide-1.2-2016-03-22.pdf>

Pacific crest (2011). Ohje langattomiin datalinkkeihin joulukuu 2011. Haettu 8.5.2017 osoitteesta http://pacificcrest.com/library/BRO_Guide_Wireless_Data_Links.pdf

Paikkatietoikkuna (2012). REST-rajapinta 3.2012. Haettu 3.6.2017 osoitteesta https://www.paikkatietoikkuna.fi/c/document_library/get_file?uuid=97cc3ed7-bbee-4230-96a8-6a0615f14515&groupId=108478

Peilausauto (2017). Näin toimii Viestintäviraston salaperäinen peilausauto 29.1.2017. Haettu 8.10.2017 osoitteesta <https://yle.fi/uutiset/3-9425002>

phpnet (2017). PHP info 2017. Haettu 4.6.2017 osoitteesta <http://php.net/manual/en/function.phpinfo.php>

Raspberry PI 3 (2016). Raspberry Pi 3 Model B 2016. Haettu 11.5.2017 osoitteesta <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

Rikoslaki 578/1995. Haettu osoitteesta 8.10.2017 <http://www.heikniemi.fi/kirj/jur/rikos/tlhair.html>

Rouse, M. (2007). UHF (Ultrahigh frequency) definition, Blogijulkaisu huhtikuu 2007. Haettu 1.4.2017 osoitteesta <http://searchnetworking.techtarget.com/definition/UHF>

Sciloo (2017). How WiFi/Radio crosses/penetrates/pass through the wall where light fails/doesn't? 3.2.2017. Blogijulkaisu haettu 8.10.2017 osoitteesta <https://sciloo.com/2017/02/03/wifiradio-crossespenetratespass-wall-light-failsdoesnt/>

Tekstenuitleg (2012). Kuinka radioaallot luodaan. Haettu 8.5.2017 osoitteesta <http://en.tekstenuitleg.net/articles/networking/how-wireless-communication-works/how-radio-waves-are-created-by-electricity>

Viestintävirasto (2016). Taajuusjakotaulukko 22.12.2016. Haettu 28.4.2017 osoitteesta https://www.viestintavirasto.fi/attachments/maaraykset/Taajuusjakotaulukko_22.12.2016.pdf

WikiBooks (2017). Radiotaajuuskirja/UHF tietoa ultra korkeista taajuuksista 28.3.2017. Haettu 1.4.2017 osoitteesta <https://fi.wikibooks.org/wiki/Radiotaajuuskirja/UHF>

Liite1

UHF-taajuusalueen käyttö Suomessa

käyttötarkoitus	nimi	taajuusalue MHz
sotilaskäyttö	ilmavoimat	240-328
ääniohjelmansiirto	yleisradio-asetat	308-319
ilmailu	ILS	328-336
sotilaskäyttö		336-378
viranomaisverkko	VIRVE	378-385
elinkeinoelämä	TETRA	385-390
viranomaisverkko	VIRVE	390-395
elinkeinoelämä	TETRA	395-399,8875
satelliittiliikenne, ilmatiede	satelliitit, navigointi, sondit, sääasemat, lääketiet. implantit	399,900 - 406
pelastuspalvelu	EPIRB	406,000-406,100
elinkeinoelämä	puheradio, valvonta, kaukomittaus, hälytys ja muu datasiirto	406,125 - 409,975
elinkeinoelämä	puheradio, DMR ja radiolinkit	410-430
elinkeinoelämä	D-GPS	430-432
amatööriradio	70 cm aallot	432-438
elinkeinoelämä, viranomaiset	puheradio, savusukellus, nostotyö, mittarinluku ja muu datasiirto	438-446
luvasta vapaa käyttö	PMR- ja dPMR446	446,00625-446,200
elinkeinoelämä	DMR, valvonta, kaukomittaus, hälytys ja muu datasiirto	446,21875-449
väestönsuojelu	puheradio	449,025-449,525
elinkeinoelämä, viranomaiset	puheradio	449,550-450
elinkeinoelämä	puheradio, laajakaista, henkilöhaku	450-470
digi-tv	SDTV, langattomat mikrofonit	470-790
matkaviestintä	4G	791-821
luvasta vapaa käyttö	radiomikrofonit, kuunteluapuvälineet, korvamonitorit	823-832
matkaviestintä	4G	832-862
luvasta vapaa käyttö	radiomikrofonit, langattomat kuulokkeet, johdottomat puhelimet, RFID, valvonta, hälytys	863-870
raideliikenne	GSM-R	876,100-879,900
matkaviestintä	GSM 900 ja 3G	880,100-914,900
raideliikenne	GSM-R	921,100-924,900
matkaviestintä	GSM 900 ja 3G	925,100-959,900
ilmailu	navigointi, tutkat, DME ja ADS-B	960-1240
amatööriradio, navigointi, tutkat	23 cm aallot	1240-1300
ilmailu	navigointi, tutkat	1300-1350
radiolinkit	elinkeinoelämän ja sotilaskäytön linkit	1350-1400
radioastronomia		1400-1427
sotilaskäyttö		1427-1452
sotilaskäyttö, yleisradio	T-DAB ja S-DAB	1452-1492
radiolinkit, satelliittiliikenne		1492-1525
merenkulun satelliittipalvelut	Inmarsat yms	1525-1559
satelliittinavigointi	GPS, GLONASS	1560-1610
merenkulun satelliittipalvelut	Inmarsat yms	1610-1660
radioastronomia, ilmatiede	sääsondit	1660-1670
satelliittiliikenne	sääsatelliitit, sondit	1670-1710
matkaviestintä	GSM 1800 ja 3G	1710-1785
radiomikrofonit		1785-1800
matkaviestintä	GSM 1800 ja 3G	1805-1880
luvasta vapaa käyttö	DECT	1882-1897
matkaviestintä		1900-1980
satelliittiliikenne	Inmarsat	1980-2010
laajakaista	digitaalinen laajakaistainen 2000 verkko	2010-2025
satelliittiliikenne	satelliittien ohjaus, kaukokartoitus, radiolinkit	2025-2110
matkaviestintä		2110-2170
satelliittiliikenne	satelliittien ohjaus, Inmarsat, radiolinkit	2170-2290
langattomat kamerat		2290-2300
amatööriradio, langattomat kamerat, luvasta vapaat laitteet	2,4 GHz langaton lähiverkko, RFID, automaattinen vaununtunnistus, ohjaus/valvonta/hälytystutkat	2300-2483,500
satelliittiliikenne, lääketiet. implantit		2483,500-2500
matkaviestintä	sähköisten viestintäpalvelujen tarjontaan soveltuvat maanpäälliset järjestelmät	2500-2690
radioastronomia		2690-2700
ilmailu	navigointi, tutkat, sotilaskäyttö	2700-3100