

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Arvutisüsteemide instituut

Simo Oskar Juvanen 186064IACB

# **Keskkonna mõõtepunkti jälgimine madala energiatarbega sidemoodulite jadaga**

Bakalaureusetöö

Juhendaja: Kalle Tammemäe  
PhD

Tallinn 2021

# Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor:           Simo Oskar Juvanen  
                      12. Mai, 2021

# Annotatsioon

Käesoleva lõputöö eesmärgiks on luua keskkonna mõõtepunkti jälgimiseks lahendus, mis kasutab madala energiatarbega sidemoodulite ahelat, et oleks võimalik potentsiaalselt kaugetest kohtadest, näiteks loodusest, mugavalt andmeid koguda ning need andmed ka pilve andmekogumist kätte saada.

Töös on kirjeldatud sissejuhatavalt andmesidevõrkudest tänapäeval ning lähemalt ZigBee protokollist. Lahenduse loomisel on kirjeldatud ülesandele iseloomulikud tunnused ning analüüsitud valiku tegemisel nende kriteeriumite kohaselt erinevaid sidetehnoloogiaid. Samuti on kirjeldatud XBee võrgu üles seadmist, andmete lugemist ja pilve üles laadimist ning erinevaid katsetusi moodulitega.

Töö tulemusena valmis prototüüplahendus, kolmest sidemoodulist koosnev juhtmeta ning madala energiatarbega võrk, mis võimaldab saata sensorist saadud andmeid pilveandmepunkti ning need pilve üles laadida. Samuti on võimalik ahelat pikendada lisamoodulitega.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 34 leheküljel, 6 peatükki, 12 joonist, 3 tabelit.

# **Abstract**

## **Environmental monitoring using a series of low-power communication modules**

The purpose of this bachelor's thesis is to create a solution for measuring an environmental point, using a series of low-power communication modules to provide a way to comfortably gather data from potentially remote places e.g bodies of water or a forest, and make this data available through a cloud storage service.

The thesis briefly goes over the trends, principles and technologies used in modern data networks and takes a specific look at the ZigBee protocol. The thesis also describes the process of choosing the most fitting communication technologies to use according to the intrinsic characteristics of the proposed problem. In addition, the setup of an XBee network, including security and power saving methods, and the method of cloud storage with the algorithm of the program responsible for updating the data, are thoroughly explained. The various experiments done with the network are also described along with their positives and areas, which could be improved.

As the result of this work, a prototype network, consisting of three wireless low-power communication modules, which allowed sending data from a node with a temperature sensor to the cloud storage point and then uploaded to the cloud, was formed. The network is expandable through additional modules. The conclusion provides some possible areas of interest to further improve upon the current result.

The thesis is in Estonian and contains 34 pages of text, 6 chapters, 12 figures, 3 tables.

# Lühendite Loetelu

IP	Internet Protocol
ATM	Asynchronous Transfer Mode
ISDN	Integrated Services Digital Network
DWDM	Dense Wavelength Division Multiplexing
LAN	Local Area Network
WAN	Wide Area Network
ARPANET	Advanced Research Projects Agency Network
TCP	Transmission Control Protocol
SAP	Service Access Point
WPAN	Wireless Personal Area Network
IEEE	Institute of Electrical and Electronics Engineers
MAN	Metropolitan Area Network
PHY	Physical
APS	Application Support Sublayer
ZDO	ZigBee Device Object
MAC	Medium Access Control
CSMA-CA	Carrier-Sense Multiple Access with Collision Avoidance
CBKE	Certificate-Based Key Establishment
BLE	Bluetooth Low Energy
USB	Universal Serial Bus
XCTU	XBee Configuration and Test Utility
JST	Japan Solderless Terminal
ID	Identifier
PAN	Personal Area Network
ASCII	American Standard Code For Information Interchange
API	Application Programming Interface
ADC	Analog-Digital Converter

# Sisukord

<b>Jooniste Loetelu</b>	<b>vii</b>
<b>Tabelite loetelu</b>	<b>viii</b>
<b>1 Sissejuhatus</b>	<b>1</b>
1.1 Ülesande püstitus . . . . .	2
1.2 Töö struktuur . . . . .	2
<b>2 Kommunikatsioon</b>	<b>3</b>
2.1 Pikamaakommunikatsioon enne raadioside leiutamist . . . . .	3
2.1.1 Kommunikatsioon pärast telegraafi leiutamist . . . . .	4
<b>3 Andmesidevõrgud tänapäeval</b>	<b>5</b>
3.1 Trendid andmesides . . . . .	5
3.1.1 Trend odavama ning kiirema poole . . . . .	6
3.1.2 Võrkude intelligentsus . . . . .	6
3.1.3 Migratsioon veebipõhiseks . . . . .	6
3.1.4 Võrkude liikuvus . . . . .	7
3.2 Andmesidevõrgude liigitus . . . . .	7
3.2.1 Laivõrk . . . . .	7
3.2.2 Kohtvõrk . . . . .	7
3.3 Internet . . . . .	8
3.4 Protokollipõhine arhitektuur . . . . .	9
3.5 Standardiseerimine protokollarhitektuuris . . . . .	11
3.6 Kõige tüüpilisemad juhtmeta andmesidestandardid . . . . .	11
3.6.1 Personaalvõrgud (PAN) . . . . .	12
3.6.2 Kohtvõrgud (LAN) . . . . .	12
3.6.3 Metropolvõrk (MAN) . . . . .	12
3.6.4 Laivõrk (WAN) . . . . .	12
<b>4 Zigbee</b>	<b>13</b>
4.1 Zigbee arhitektuur . . . . .	13
4.2 ZigBee võrgu topoloogia . . . . .	14
4.3 Võrdlus konkureerivate juhtmeta standarditega . . . . .	15
4.4 ZigBee võrgu turvalisus . . . . .	16
4.4.1 Turvalisuse tagamise meetmed . . . . .	16

4.4.2	Turvaohud . . . . .	17
<b>5</b>	<b>Püstitatud ülesande lahendus</b>	<b>18</b>
5.1	Probleemi analüüs . . . . .	18
5.2	Sidemoodulite tehnoloogi valik . . . . .	19
5.2.1	Wi-Fi . . . . .	19
5.2.2	Z-Wave . . . . .	19
5.2.3	Bluetooth Low Energy . . . . .	20
5.2.4	ZigBee . . . . .	20
5.2.5	Valitud sidemooduli tehnoloogia . . . . .	21
5.3	Valitud sidemoodul . . . . .	22
5.3.1	Digi XBee . . . . .	22
5.4	Muud vajalikud vahendid . . . . .	23
5.4.1	Keskkonna mõõtesensor . . . . .	24
5.4.2	Pilveandmepunkt . . . . .	24
5.4.3	Akud . . . . .	24
5.4.4	Antennid . . . . .	25
5.5	Võrgu konfiguratsioon . . . . .	25
5.5.1	Võrgu baassätted . . . . .	25
5.5.2	Turvameetmed . . . . .	27
5.5.3	Kindla mooduli spetsiifilised sätted . . . . .	28
5.5.4	Energiatarve . . . . .	29
5.6	Andmete kättesaamine ning laadimine pilve . . . . .	30
5.7	Katsetused . . . . .	31
5.8	Järeldused . . . . .	33
<b>6</b>	<b>Kokkuvõte</b>	<b>34</b>
	<b>Kasutatud kirjandus</b>	<b>35</b>
	<b>Lisad</b>	<b>38</b>
	<b>Lisa 1 - Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks</b>	<b>38</b>
	<b>Lisa 2 - Sidemoodulite täiskonfiguratsioon</b>	<b>39</b>
	<b>Lisa 3 - Vahendite maksuvus</b>	<b>39</b>
	<b>Lisa 4 - Programmikood kogutud andmete kättesaamiseks ning pilve üleslaadimiseks</b>	<b>40</b>

## Jooniste Loetelu

1	<i>Interneti põhikomponendid[14]</i> . . . . .	9
2	<i>TCP/IP kihid koos protokollid näidetega[15]</i> . . . . .	10
3	<i>ZigBee Arhitektuur[21]</i> . . . . .	14
4	<i>ZigBee toetatud topoloogiad[22]</i> . . . . .	15
5	<i>SparkFun Thing Plus - XBee 3 Micro (U.FL)</i> . . . . .	22
6	<i>XCTU Arenduskeskkond</i> . . . . .	24
7	<i>Temperatuurisensor LM35[34]</i> . . . . .	25
8	<i>Raspberry Pi 3 Model B[35]</i> . . . . .	26
9	<i>ADA1578 liitium-poliümeer aku[36]</i> . . . . .	26
10	<i>Bingfu antennid[37]</i> . . . . .	27
11	<i>Lõpliku sidevõrgu topoloogia</i> . . . . .	28
12	<i>Andmete formaat pilves</i> . . . . .	31



## Tabelite Loetelu

1	<i>Võrdlus juhtmeta standardite vahel[23]</i> . . . . .	15
2	<i>Võrgu moodulite sätted</i> . . . . .	39
3	<i>Vahendite maksumus</i> . . . . .	40

# 1 Sissejuhatus

Käesoleva töö teema sai välja pakutud juhendaja poolt ning hõlmab sensorandmestiku kandmist kaugel distantsil, kasutades madala energiatarbega vahemooduleid ja ZigBee protokollid, toimetades sensorandmestiku internetilahendusega punkti selle pilvesaatmiseks. Lahenduse kasutusnäideteks on mõne kaugema keskkonna punkti seisundi mõõtmine (seire), näiteks jõe nivoo ja voolukiiruse, veekogu temperatuuri, õhu kvaliteedi, metsloomade ja lindude liikumise jms. info. Samuti eeldades, et lahendus on kasutatav väga erinevate distantside korral lisades ülekandeahelasse vajalikul määral täiendavaid releesõlmi.

Keskkonnaseire on kriitiline, et olla kindel kas meid ümbritsev keskkonnakvaliteet on paranemas või muutumas halvemaks. Keskkonnatingimused mõjutavad väga paljusid valdkondi, näiteks tervishoidu - halb õhukvaliteet või otseselt kahjulike osakeste olemasolu õhus on suure tähtsusega avalikkuse tervise poole pealt. Samuti näiteks talumehed peavad olema teadlikud lähiajalistest ilmaprognoosidest aga ka pikaajalistest kliimatrendidest, et otsustada kuidas kõige parem vilja on kasvatada.

Tänapäeval on väga palju erinevaid meetodeid, kuidas mõõta erinevatest punktidest andmeid. Näiteks on asutatud jaamasid, mis tegelevad ainult ultraviolettkiirguse taseme mõõtmisega või kosmosesse lennutatud satelliite, mis jälgivad näiteks lumekatet või pilvisust[1]. Käesoleva töö raames lähenetakse probleemile sellelt poolelt, et lahendus peaks olema võimeline saada andmeid kaugel distantsil tagant, ta oleks adaptiivne ning madala energiatarbega, ehk lihtsasti laiendatav või paigaldatav ning toimiks hooldamata võimalikult kaua.

Töö ülesande plaanisel võeti inspiratsiooni mõningatest olemasolevatest lahendustest. Üheks neist oli Thinnecti lahendused Tallinna linna keskkonna mõõtmiseks (Smart City), kus jälgitakse sensoritega linna liiklust, õhukvaliteeti, mürataset ja muud.[2]. Samuti võeti eeskuju paarist eelnevast lõputööst, milles kajastusid sarnaseid implementatsioone, näiteks Igor Golmi "ZigBee võrkude projekteerimine VKG Ojamaa kaevanduses", kus kasutati ZigBee mooduleid, et luua traadita võrk kaevandusele.[3]

## 1.1 Ülesande püstitus

Käesoleva töö eesmärgiks on luua sidevõrk, mis töötaks madala energiatarbega ning võimaldaks sidemoodulite jada kasutades koguda andmeid keskkonna mõõtepunktis. Samuti kuulub lahenduse sisse pilveandmepunkt ehk koht, kus laetakse andmed pilve andmekogumisse, kust neid on võimalik läbi interneti kätte saada.

Seega ülesande tingimused on järgmised:

1. Keskkonna mõõtepunktis seisundi jälgimine (sensor/sensorid)
2. Andmed edastatakse pilve andmekogumisse
3. Madal energiatarve
4. Kasutatakse sidemoodulite jada

Vastavalt eeldusest, et keskkonna mõõtepunkt võiks olla ka näiteks looduses asuv veekogu või muu, siis sidevõrk eeldatavasti oleks oma ulatuses suurem kui tavalised Wi-Fi või Bluetoothi võrgud ning samuti peaks olema võimalik tema ulatust suurendada sidemoodulite jada pikendades.

## 1.2 Töö struktuur

Töö esimeses osas käiakse sissejuhatavalt üle andmesidevõrkudest tänapäeval - trendidest, tehnoloogiast. Seejärel kirjeldatakse spetsiifiliselt ZigBee protokollide arhitektuuri, võrku, võrdlust konkureerivate standarditega ning turvalisust.

Töö teises osas analüüsitakse püstitatud ülesande tingimusi ning neile vastavalt seda, milline tehnoloogia oleks kõige sobilikum vastavalt neile tingimustele. Kirjeldatakse valitud sidemooduleid ning muid vajalikke vahendeid. Seejärel kirjeldatakse loodud võrgu konfiguratsiooni, ning pilve andmete laadimise programmi algoritmi põhjalikult. Luuakse prototüüp kolmest sõlmest sensorist pilve laadimise punktini. Samuti on kirjeldatud erinevaid katsetusi prototüübi loomisel ning sealhulgas ka toodud välja alad, mis vajaksid rohkem uurimist ja proovimist lahenduse edasiarendamiseks realselt kasutatava tooteni.

## 2 Kommunikatsioon

Sõna kommunikatsioon tuleneb ladina keelsest sõnast *communico*, mille tähendus on "jagama", "osa võtma" või "tooma ühiskasutusse" [4]. Antud definitsioonis "tooma ühiskasutusse" vihjab, et peab olema olemas ühine arusaam sõnumist allika ning saaja vahel.

Kommunikatsioon üldiselt kaasab endas saatjat(allikat), saajat(vastuvõtjat) ning nende vahelist signaali. Signaalid võivad näiteks olla verbaalsed, graafilised - ütlus, žest, kasvõi fotograaf. Kasutatakse koode, mis on tehtud silmadega, kehaliigutustega, hääliitsustega või muuga. Kuidas iganes see läbi on viidud, toimub alati protsess, mille käigus saatja edastab tähendust saajale.

Üheks tähtsaimaks kommunikatsioonitasandiks inimolendite vahel on inimkeeled. See on vahend, mida me kasutame, et kanda edasi oma ideid, tundeid, teadmisi ning vajadusi. Kommunikatsiooni edasiarenguta ei oleks tekkinud tsivilisatsioonid ning ühiskonda kui sellist nagu me tunneme tänapäeval. Seega selle säilimine ning areng on alati olnud ülimalt tähtis[5].

### 2.1 Pikamaakommunikatsioon enne raadioside leiutamist

Kommunikatsioon tänapäeval on arenenud vägagi keerulisele tasemele, kuid eeldused selleks said alguse juba väga muistsetel aegadel.

Esmane viis suhtlemiseks oli iseenesest mõistetavalt inimkeel. Samuti olid juba muistsetel aegadel olemas ka pikamaakommunikatsiooni võimalused, mis ei põhinenud sõnadel. Näiteks Ameerika Indiaanlased kasutasid suitsusignaale - süüdati lõkkeid järjekorras küngaste peal. Kuid sellised viisid on ainult võimelised edasi andma märku eelnevalt selgeks tehtud signaale nagu näiteks "oht" või "võit".

Mõningad mitteverbaalsed valikud on aga pisut kõrgetasemelisemad, näiteks Kanaari saartel Gomera rahvas kasutas vilekeelt, et suhelda pikkades sügavates orgudes, kus vile levis kaugele. See oli küll piisav saarteelanike hetkevajaduste jaoks, kuid ei suuda siiski kirjeldada nii täpset sõnumit kui seda saab teha läbi kõne või kirjakeele[6].

Umbes 3200 aastat eKr leiutati Iraagis ning Egiptuses kirjaviisid. Järgmiseks suureks sammuks oli esimese tähestiku loomine, nüüdseks Iisraeli ning Leebanoni aladel, 1600 aastat Ekr. Tolles ammus ajastus olid paljudes tsivilisatsioonides, näiteks Egiptuses, Pärsias, Roomas olemas efektiivsed postisüsteemid, et edastada kirjalikke sõnumeid üle kogu oma valduse, kasutades hobustel sõnumitoojaid. Need sõnumid olid kirjas papüürusel või pärgamendil kuniks leiutati paber.

Edasiseks suurimaks edusammuks osutus printimise võimaluse loomine. Esimesena tuldi selle peal Hiinas 6. sajandil pKr, kasutades tindi sisse kastetud plokket, millesse on sisse lõigatud sümbolid. Kõige esimene printitud raamat oli Teemant Sutra aastal 686. Euroopasse jõudis printimine 15. sajandi keskel Johannes Gutenbergi trükipressiga, mis muutis raamatud palju odavamaks ning lõi võimaluse hakata printima ajalehti. Esimesed ajalehed printiti 17. sajandil. Ajalehtede laiali viimiseks loodi postiteenuseid.

### **2.1.1 Kommunikatsioon pärast telegraafi leiutamist**

19. sajandil muutus infovahetus palju efektiivsemaks - posti saatmise võimalused muutusid aina mugavamaks ning odavamaks. 1837 aastal leiutati telegraaf ning aastaks 1866 oli rajatud kaabel üle Inglise kanali, luues võimaluse saata sõnumeid üle Atlandi ookeani. Aastal 1876 lõi Alexander Graham Bell esimese telefoni ning juba 3 aasta hiljem avati Suurbritannias esimene telefonijaam.

20ndasse sajandisse jõudes hakkas kommunikatsiooni areng eksponentsiaalselt kiirenema. Näiteks 1901. aastal saadeti üle Atlandi ookeani esimene raadiosõnum ning 1933. aastal oli Suurbritannias pooltes kodudes olemas oma raadio. Samuti näiteks televisioon jõudis 40 aastaga, alates oma leiutamisest 1925. aastal, Suurbritannias 90% inimesteni ning oli selle ajaga muutunud peamiseks meelelahutusallikaks[7].

20. sajandi keskel leiutati transistor - see oli tähtis hetk elektroonika arengus. Nimelt saadi luua integreeritud elektroonikaseadmeid. Nende tehnoloogiate peale arendati Ameerika Ühendriikide valitsuse poolt, saates kosmosesse 1958 aastal esimese kommunikatsiooni-satelliidi. Peale esimest faksi ülekannet levis populaarsuses ka andmete moduleerimine heliks üle telefoniliinide.

Üheks suurimaks verstapostiks tänapäevase kommunikatsiooni rajamisel oli IP protokollide areng. Sellele järgnesid järgmistel aastakümnetel varajased paketi põhised andmeside-tehnoloogiad nagu ATM, Frame relay, ISDN täites järjest kasvavaid kommertsiaalseid ning personaalseid andmevajadusi. 20. sajandi lõpuks oli IP tehnoloogia kasutajaid juba miljonid ja 1990ndate keskpaigaks kerkis esile interneti põhiosana World Wide Web [8].

## 3 Andmesidevõrgud tänapäeval

Tänapäeva maailmas on efektiivne ning tõhus andmevahetus elutähtis ükskõik millisele ettevõttele. Informatsiooni liiklus, nii lokaalne kui ka pikamaa, on kasvanud kiirelt ning sujuvalt juba aastakümneid. Võrkudes saadetud andmed ei ole enam limiteeritud ainult heli või andmebaitidele, vaid hõlmavad ka enda alla järjest rohkem pilte, videosid ja muud. Järjest enam suureneb rõhk veebiteenuste, kaugjuurdepääsu, veebitehingute ning sotsiaalvõrgustike peale ning see trend ei paista lõppema. Seega ettevõtted on pidevalt survestatud suurendama oma kommunikatsiooni võimaluste mahtu kulutõhusal viisil[9]. Samuti on juba algust tehtud globaalsete kiiret interneti pakutavate teenustega nagu näiteks Starlink, mille eesmärgiks ongi pakkuda kiiret, madala latentsusega ülemaailmset satelliit-interneti[10].

Jaanuar 2021 aasta andmete põhjal on aktiivseid interneti kasutajaid üle maailma 4.66 miljardit - 59.5% kogu maailma elanikkonnast ning sellest 92.6% kasutasid interneti läbi mobiilseadme[11]. Aastakümne tagasi oli sama statistika põhjal interneti kasutajate arv vaid 1.9 miljardit - 2.45 korda vähem[12]. Seega andmesidevõrkude võimaluste kasutajate arv kasvab üüratult kiirelt.

Mida rohkem me sõltume infotehnoloogiast, seda suurem soov tekib kasutajatel kasutada mitmesuguseid teenuseid. Näiteks mobiili andmeside liiklus kasvab plahvatuslikult ning sellega kaasas ka kasutajate poolt kasutatav andmete maht läbi nutitelefonide ja muude seadmete. Lisaks sellele, aja jooksul tekivad kasutajatel nõudlused üha kõrgema kvaliteediga teenuste poole, mis toetaksid uuemaid võimalusi, näiteks kõrge resolutsiooniga kaamera pilte, video voogedastust või tippasemel heli. Samasugust kasvu saab ka täheldada kaabliga interneti ja privaatsete võrkude korral. Selleks, et arenguga sammu pidada, peavad teenusepakkujad pidevalt investeerima kõrgemahulistesse võrkude ning ülekandegaamade edendamisse. See omakorda stimuleerib laienemist erinevates teenustes - võrkude maht, efektiivsus ning teenuste kasv käivad käsikäes[9].

### 3.1 Trendid andmesides

Erinevad trendid tänapäeva tehnoloogia arengus loovad eelduse andmesidevõrkude mahu ning teenuste suurendamiseks. Eriti märkimisväärtteks trendideks tehnoloogias on hetkel

need neli järgmist.

- Trend odavama ning kiirema poole
- Võrkude intelligentsus
- Migratsioon veebipõhiseks
- Võrkude liikuvus

### **3.1.1 Trend odavama ning kiirema poole**

Nii kommunikatsiooni kui ka arvutite alal jätkub trend odavuse ning kiiruse poole. Arvutite koha pealt see tähendab üha võimsamaid seadmeid ning arvutite võrke, mis on suutelised toetama järjest nõudlikemaid rakendusi, näiteks multimeediat. Mis puutub kommunikatsiooni, siis tõusev optilise kaabli ning kõrge kiirusega traadita võrkude kasutus on andmevahetuse hinnad toonud alla, kuid samal ajal suurendanud mahutavust märgatavalt. Näiteks pika vahemaa telekommunikatsiooni korral, kasutades DWDM tehnoloogiat, on võimalik üle optilise kaabli liigelda kiirusega kuni mitu terabiti sekundis. Lokaalvõrkudes (LAN) on juba tavaline 100 Gbps Etherneti baasvõrk.

### **3.1.2 Võrkude intelligentsus**

Tänapäeva andmesidevõrgud on "intelligentsemad" kui kunagi varem. Nimelt võrgud saavad pakkuda erinevatel tasemetel teenusekvaliteeti, mis hõlmab enda all näiteks maksimaalse viite ning minimaalse läbilaskevõime spetsifikatsioone, et kindlustada kõrge kvaliteediga tugi rakendustele ja teenustele. Samuti on võrkude turvameetmed ning majandamine palju kohandatavam kui kunagi varem.

### **3.1.3 Migratsioon veebipõhiseks**

Internet, veeb ja kõik seotud rakendused on kerkinud esile kõige dominantsemana nii ettevõtete kui ka personaalvõrkude maastikul. Jätkumas on pea iga võimaliku asja migratsioon IP peale, mis on tekitanud palju uusi võimalusi ja väljakutseid info- ja kommunikatsioonitehnoloogia maailmas. Lisaks sellele, et kasutada interneti, et suhelda klientide, tarnijate või koostööettevõtetega, on paljud ettevõtted ka loomas sisevõrke, et isoleerida varalist informatsiooni tahtmatu ligipääsu eest.

### **3.1.4 Võrkude liikuvus**

Liikuvus on kõige uuem arengusuund ettevõtetele. Nutiseadmete populaarsuse kasvades on neist saanud ajend võrkude edasiarenduseks. Kuigi trend mobiilsete võrkude poole oli olemas juba aastakümneid, siis see plahvatas koos nutiseadmete arenguga. Rakendusi, mis varem olid toetatud ainult terminalides ning kontoriarvutite peal, on nüüd rutiinselt võimalik kasutada ka mujal. Pilveandmetöötlus on võetud omaks paljude suurte ettevõtete, nagu näiteks Microsofti, Oracle poolt. Ekspertid ennustavad, et see trend jätkub edasi ning võimalus kasutada ettevõtete resursse ja teenuseid ükskõik kus, ükskõik millal muutub dominantseks[13].

## **3.2 Andmesidevõrgude liigitus**

Andmesidevõrkude tüüpide liigitamiseks on palju erinevaid termineid ning mudeleid, kuid kõige lihtsamalt jaotatakse võrgud laivõrkudeks(WAN) ja kohtvõrkudeks(LAN).

### **3.2.1 Laivõrk**

Laivõrk ehk WAN üldiselt katab geograafiliselt suurt ala. Tihtipeale vajab nende paigaldus avalikku eesõigust ning nad sõltuvad teenusepakkujast, kes avalikule publikule vastavaid kommunikatsiooniteenuseid pakub. Tavaliselt WAN koosneb mitmest omavahel ühendatud sõlmedest. Informatsiooni ülekande ühest seadmest marsruutitakse läbi sisemiste sõlmede spetsiifilisse sihtseadmesse. Neid sõlmi ei huvita andmete sisu, vaid nende eesmärgiks on puhtalt andmete juhatamine sõlmest-sõlme kuni jõutakse sihtpunkti.

### **3.2.2 Kohtvõrk**

Sarnaselt WAN-ile, on LAN kommunikatsioonivõrk, mis seob omavahel mitmeid erinevaid seadmeid ning pakub võimaluse informatsioonivahetuseks seadmete vahel. Siiski on nende vahel mõningad suured erinevused:

1. LAN võrgu skoop on palju väiksem, tüüpiliselt ühe hoone või hoonetekobara suurune. See erinevus geograafilises suuruses viib ka erinevate tehniliste lahendusteni.
2. Üldiselt LAN võrgu omanik omab ka seadmeid, mis on võrku ühendatud. WAN puhul ei pruugi nii olla, vaid märkimisväärne osa võrgu seadmetest pole omatud. Sellel distinktsioonil on paar tähtsat tähendust. Esiteks, LAN-i valikul peab hoolikalt kaaluma erinevaid võimalusi, sest selle paigaldus võib nõuda mahukat investeeringut. Teiseks, LAN puhul langeb võrgu haldamise vastutus kasutaja enda peale.



3. LAN-i puhul on võrgusisesed andmekiirused tüüpiliselt palju suuremad kui WAN-is.

LAN võrgud jagunevad omakorda erinevate konfiguratsioonide vahel. Kõige tavalisemateks osutuvad kommuteeritud LAN ning traadita LAN. Populaarseim kommuteeritud LAN on Ethernet LAN, mis võib koosneda lihtsalt üksikust kohtvõrgu kommutaatorist ning tema külge ühendatud seadmetest või mitmetest omavahel ühendatud kommutaatoritest. Tavalisim traadita LAN on Wi-Fi.

Traadita LAN-e kasutatakse tihti ettevõtetes ning avalikes hoonetes ja paikades. Traadita tehnoloogia on laialt levinud ka laia ala heli ning andmevõrkudes. Traadita võrgud pakuvad eeliseid just liikuvuse ning paigalduse lihtsuse aladel.

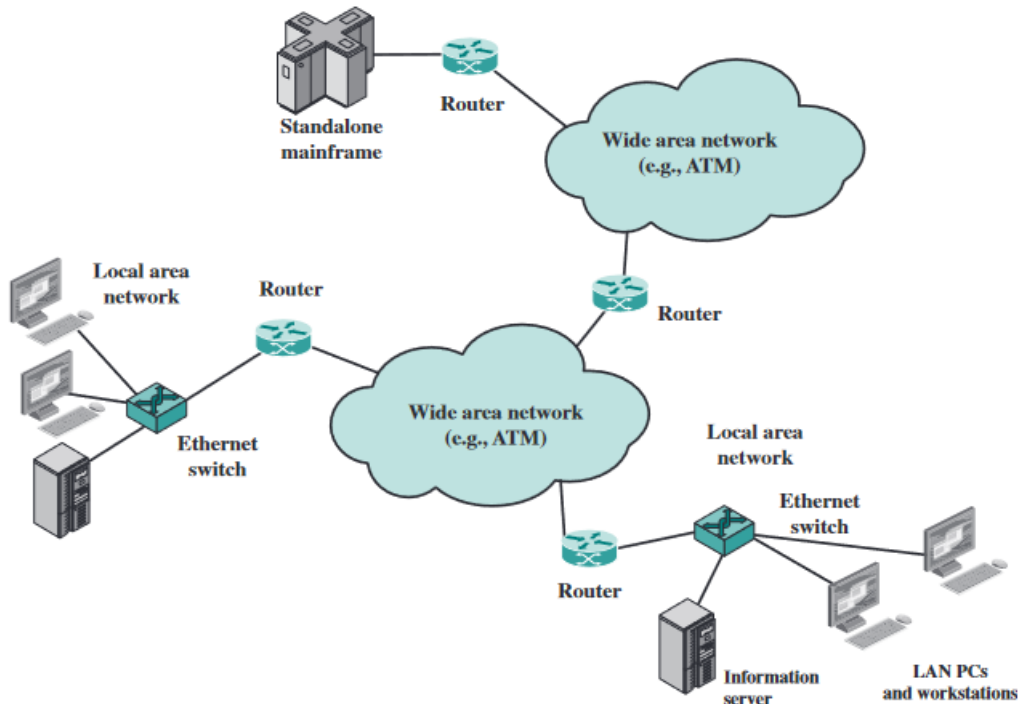
### 3.3 Internet

Internet arenes välja ARPANET-i projektist, mis loodi aastal 1969 Ameerika Ühendriikide kaitseministeeriumi poolt. Tegemist oli esimese töötava pakettkommuteeriva võrguga. ARPANET alustas töötamist neljas asukohas - tänapäeval on seadmete ning kasutate arv miljardites.

ARPANET osutus nii edukaks, et sama tehnoloogiat kasutati taktikalise raadio kommunikatsiooni ning satelliitkommunikatsiooni korral. Kuna need võrgud töötasid väga erinevates keskkondades, olid teatud parameetrite, näiteks pakettide suurused, erinevad. Seega tuldi *interworking* meetodite ja protokollide peale - see tähendab, et saab suhelda arbitraarselt üle mitme pakettkommuteeritud võrgu. Sellest kujunes lõpuks välja TCP ning IP protokollid, mis kujundas baasi TCP/IP protokollide komplekti jaoks.

Joonis 1 illustreerib komponente, mis moodustavad Interneti. Interneti eesmärgiks on siduda omavahel lõppsüsteeme ehk hoste, nende alla kuuluvad arvutid, serverid, tööjaamad ja muud. Enamik hoste, mis kasutavad interneti, on ühendatud võrku, kas LAN-i või WAN-i. Need võrgud on omakorda ühendatud läbi marsruuterite. Iga marsruuter on seotud kahe või enama võrguga. Mõni host võib olla ka ühendatud otse marsruuteri külge, mitte läbi võrgu.

Interneti tööpõhimõte on järgmine: Üks host saab saata andmeid teisele hostile, kes asub ükskõik kus internetis. Saatja host tükeldab saadetavad andmed väiksemateks pakettidejadaks, mida kutsutakse IP datagrammideks või IP pakettideks. Iga pakett sisaldab unikaalset sihthosti aadressi - IP aadressi. Selle aadressi põhjal kõik pakettid rändavad läbi marsruuteride ning võrkude lähtepunktist sihtpunkti. Iga marsruuter, saades kätte paketi, teeb otsuse ning suunab selle edasi sihtpunkti poole[14].



Joonis 1. Interneti põhikomponendid[14]

### 3.4 Protokollipõhine arhitektuur

Kui arvutid, terminalid või teised seadmed vahetavad omavahel andmeid, siis läbiviidavad protseduurid võivad olla üsnagi kompleksed. Võttes näiteks faili saatmise kahe arvuti vahel, peab kõigepealt eksisteerima teekond arvutist arvutisse, kas otseselt või kaudselt läbi andmevõrgu, kuid vaja läheb veelgi rohkem:

1. Saatja peab aktiveerima otsese kommunikatsioonitee või teatama võrku soovitud sihtpunkti identiteedist.
2. Saatja peab kindlaks tegema, kas sihtsüsteem on valmis andmeid vastu võtma.
3. Failiedastusrakendus saatjas peab olema kindel, et failiedastusprogramm sihtsüsteemis on valmis vastu võtma faili ning seda mällu kirjutada.
4. Kui failiformaadid, mida kasutatakse süsteemides, on erinevad, siis üks arvutitest peab tegelema formaatimisega

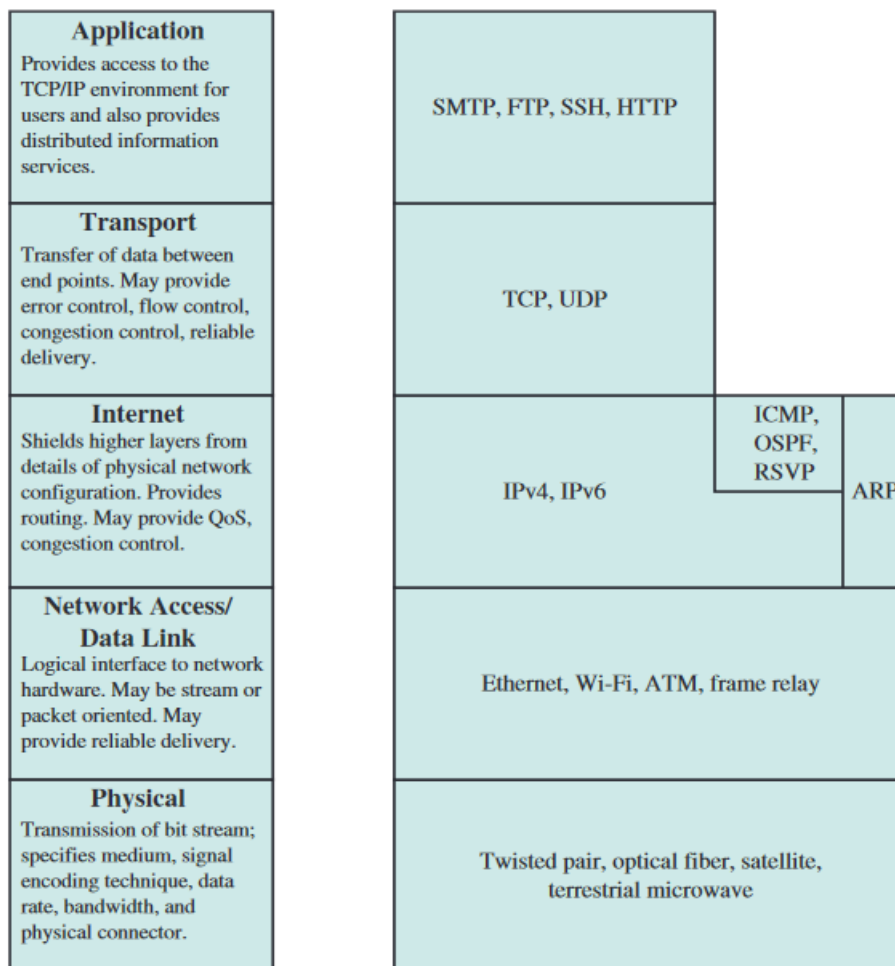
Ilmselt peab olema ka sellise lihtsa näite puhul kahe süsteemi vahel suur hulk koostööd. Selle asemel, et teha see töö korraga, jaotatakse ülesanne lihtsamateks alamülesanneteks, mida saab eraldi läbi viia. Protokollipõhises arhitektuuris on need moodulid vertikaalse pinu ehk virnana. (Joonis 2).

Iga tase pinus viib läbi konkreetseid funktsioone, mida on vaja, et suhelda teiste süs-

teemidega. Ülemine kiht sõltub allolevast kihist ning pakub teenuseid endast üleval olevasse kihti. Ideaalis on iga kiht defineeritud sedasi, et muutused ühes ei vaja muutusi teistes kihtides. Selleks, et andmeid saata või vahetada, on siiski vaja mitut seadet, seega peab sama funktsionaalsus olema olemas mõlemas seadmes. Kommunikatsioon saavutatakse kui vastavad kihid süsteemides suhtlevad. Vastavad kihid suhtlevad omavahel formeeritud andmeplukkide kaudu, mis alluvad kindlale reeglihulgale - protokollile.

Protokoll koosneb üldiselt:

- **Süntaksist:** määrab ära andmeplukkide formaadi.
- **Semantikast:** hõlmab informatsiooni koordineerimist ning vigadega tegelemist.
- **Ajastusest:** hõlmab kiiruste sobitamist ning andmete järjestamist[15].



Joonis 2. TCP/IP kihid koos protokollid näidetega[15]

### 3.5 Standardiseerimine protokollarhitektuuris

Protokollipõhine arhitektuur, nagu näiteks TCP/IP, pakub hea raamistiku standardiseerimiseks. Mudeli sees saab luua ühe või mitu erinevat standardit iga kihi kohta. Mudel defineerib üldiselt ära funktsioonid, mida kihis peaks teostama ning seega hõlbustab standardite loomise protsessi. Kuna funktsioonid igas kihis on kindlalt defineeritud, saab luua standardeid iga kihi kohta üksteisest sõltumatult ning samaaegselt - kiireneb standardiseerimise protsess. Samuti kuna kihtidevahelised piirid on hästi määratletud, ei saa muutused ühes kihis mõjutada teises kihis olevat tarkvara - uusi standardeid on lihtsam välja tuua.

Standard peab defineerima konkreetse kihi kohta järgmised:

- **Protokolli spetsifikatsioon:** Kaks üksust samas kihis, kuid erinevates süsteemides, koostuvad ja suhtlevad protokollide kaudu. Kuna kaasatud on kaks erinevat süsteemi, peab protokoll olema spetsifitseeritud täpselt. See hõlmab enda all protokollide andmeüksuste formaati, kõikide alade semantikat ning lubatud järjestusi.
- **Teenus:** Lisaks protokollidele, on standardeid vaja teenustele, mis iga kiht pakub temast kõrgemale kihile. Üldiselt on see teenuse definitsioon võrdväärne funktsionaalse kirjeldusega, mis defineerib, mis teenust pakutakse, kuid mitte kuidas teda pakutakse
- **Adresseerimine:** Iga kiht pakub teenuseid kõrgemates kihtides olevatele üksustele.

Vajadus pakkuda konkreetne protokollide spetsifikatsioon on iseenesest mõistetav, kuid järgmised kaks nõuavad selgitust. Kuna kahe kõrvuti oleva kihi vaheline suhtlus toimub ühes süsteemis, sõltumatult teisest, võivad teenuse enda detailid olla erinevad, senikaua kuni ülemisele kihile pakutakse mõlemas süsteemis samu teenuseid. Samuti saab sedasi luua liideseid, mis on võimalikult efektiivsed. Seoses adresseerimisega, saab igas kihis adresseerimise mehhanismi implementeerimisel multiplekseerida mitut kasutajat kõrgemast kihist. Seda ei pruugi toimuda igas kihis, kuid mudel pakub selle võimaluse[16].

### 3.6 Kõige tüüpilisemad juhtmeta andmesidestandardid

Üle maailma kasutatakse mitmeid erinevaid meetodeid ning standardeid juhtmeta kommunikatsiooni saavutamiseks. Neid tehnoloogiaid saab jagada nelja kategooria vahel vastavalt spetsiifilistele rakendustele ning ülekanderaadiusele.

### **3.6.1 Personaalvõrgud (PAN)**

PAN võrgus toimub andmevahetus üldiselt ühe inimese lähestikku. PAN-e kasutatakse põhiliselt inimese personaalsete seadmete vahel informatsiooni vahetamiseks või näiteks interneti või teise võrguga ühendamiseks. Juhtmetega PAN võib olla ühendatud näiteks USB kaudu, kuid juhtmeta PAN korral on kasutusel kõige rohkem Bluetooth ning vähem ka infrapuna.

Bluetooth on spetsifikatsioon juhtmeta personaalvõrkude jaoks. Ta pakub viisi lühikese raadiuse sees mitmeid seadmeid omavahel, kasutades lähitoimelist raadiosagedust. Bluetooth on mõeldud madala energiatarve ning kauguse jaoks.

Infrapuna tehnoloogia põhineb infrapunavalgusega andmete saatmist üle lühikeste vahemaade.

### **3.6.2 Kohtvõrgud (LAN)**

Juhtmeta kohtvõrgu peamiseks standardiks on kujunenud IEEE 802.11 ehk Wi-Fi. 802.11 hõljab enda all palju erinevaid meetodeid, kuid kasutades samu protokolle. Wi-Fi leviala on märgatavalt suurem kui Bluetoothil või infrapunavalgusel ning samuti on ka andmevahetuse kiirused suuremad.

### **3.6.3 Metropolvõrk (MAN)**

MAN ehk Metropolitan Area Network on IEEE poolt välja mõeldud termin WiMAX standardi jaoks. WiMAX võrgud võivad olla kuni 16 kilomeetri suurused eeldades väga häid tingimusi ehk midagi LAN ning WAN vahepealset. Seega ta on alternatiiv fikseeritud kaablitega võrkudele ning on võrdlemisi lihtne üles seadistada. WiMax

### **3.6.4 Laivõrk (WAN)**

Laivõrgu korral on võrkude geograafiline suurus tüüpiliselt juba selline, et ei saa enam rääkida ühest võrgu alast, vaid peab vaatama kui mitmete väiksemate võrkude tervikut.

Samas tänapäeval on tehnoloogia arenenud kiirelt mobiilsete andmevõrkude koha pealt. On võimalik saavutada väga häid andmeedastuskiirusi väga suure geograafilise ala sees, näiteks kasutades 3.5G standardit[17].

## 4 Zigbee

ZigBee on kõige populaarsem tööstuses kasutatav traadita võrgustandard sensorite, instrumentide ja kontrollsüsteemide ühendamiseks. ZigBee-d on varem kutsutud "asjade internetiks" - ta on traadita personaalse võrgu(WPAN) spetsifikatsioon. ZigBee on avatud, globaalne, paketi põhine protokoll, mis on loodud, et pakkuda lihtsasti kasutatavat arhitektuuri ja turvatud, töökindlat ning madala voolutarbega traadita võrku. Kõige tavalisemad ZigBee rakendused on:

- Kodu ja kontori automatiseerimine
- Tööstuslik automatiseerimine
- Meditsiiniline jälgimine
- Madala voolutarbega sensorid
- Soojustuse, ventilatsiooni ja õhukonditsioneeride kontrollimine[18]

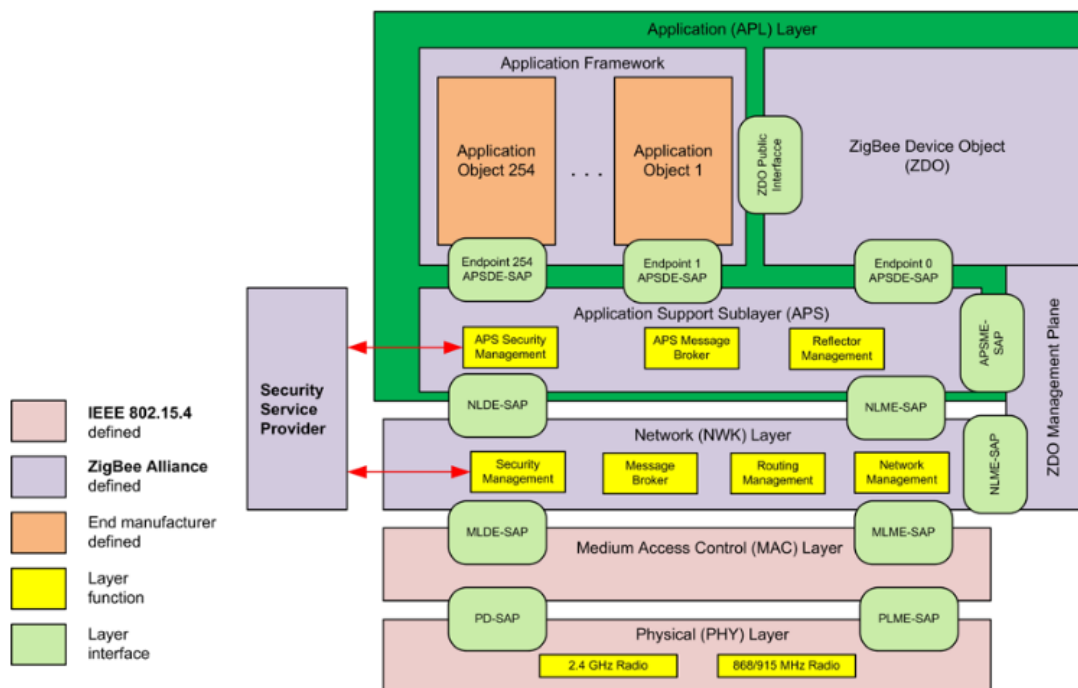
### 4.1 Zigbee arhitektuur

ZigBee protokollu pinu arhitektuur on üles ehitatud erinevatest kihtidest (Joonis 3). Iga kiht viib läbi konkreetse komplekti teenuseid kõrgemale kihile. Andmeüksus pakub andmeedastusteenust ning haldusüksus pakub kõiki muid teenuseid. Iga teenuseüksus näitab ülemisele kihile liidest läbi teenuse pöörduspunkti(SAP-i) ning iga SAP toetab arvu teenuseid, et saavutada vajalik funktsionaalsus.

Kaks alumist kihti on defineeritud IEEE 802.15.4 standardi poolt: füüsiline(PHY) kiht ning meediapäsu reguleerimise alamkihi (MAC). ZigBee Alliance ehitab selle vundamenti peale, pakkudes võrgukihi ning raamistikku rakenduskihi jaoks. Rakenduskihi raamistik koosneb rakenduse toe alamkihist (APS) ning ZigBee seade objektides (ZDO). Tootja poolt defineeritud rakendused kasutavad raamistikku ning jagavad APS ning turvateenuseid ZDO-dega.

PHY kiht opereerib kahes sagedusvahemikus: 868/915 MHz ja 2,4GHz. Madalama sagedusega PHY kiht katab nii 868Mhz Euroopa laineala kui ka 915MHz laineala, mida kasutatakse näiteks Ameerika Ühendriikides ja Austraalias. Kõrgema sageduse PHY kihti kasutatakse praktiliselt ülemaailmselt.

MAC alamkiht kontrollib ligipääsu raadiokanalile, kasutades CSMA-CA<sup>1</sup> mehhanismi. Samuti võib tema vastutuse kätte langeda "beacon" kaardrite<sup>2</sup> edastamine, sünkroniseerimine ning töökindla edastuse tagamine[21].



Joonis 3. ZigBee Arhitektuur[21]

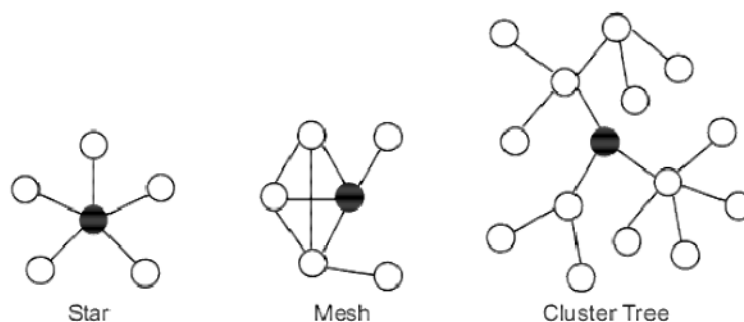
## 4.2 ZigBee võrgu topoloogia

ZigBee võrgukiht(NWK) toetab täht-, puu- ning võrgusilmatopoloogiaid (Joonis 4)[21].

Kui kasutusel on tähttopoloogia, siis võrk on juhitud ühe seadme poolt, kelleks on ZigBee koordinaator. Koordinaator vastutab selle eest, et võrk luuakse ning hoiab korras teiste seadmete tööd võrgus. Kõik teised seadmed, lõpp-seadmed, suhtlevad otseselt ZigBee koordinaatoriga. Puu- ning võrgusilmatopoloogias, koordinaator on vastutuslik võrgu loomise ning teatud võrgu parameetrite valimise ees, aga võrku saab ka suurendada, kasutades ZigBee marsruutereid. Puu-võrkudes marsruuterid liigutavad andmeid ja juhtsõnumeid läbi võrgu, alludes hierarhilsiele marsruutimiskavale. Võrgusilma-võrgud lubavad peer-to-peer suhtlust ehk võrdõigusvõrku[21].

<sup>1</sup>CSMA-CA on meetod kontrollimaks andmeside mitme erineva osaleja vahel, kasutades jagatud detsentraliseeritud kanalit. Saadetakse andmeid ainult kui kanal on vaba, välditakse kokkupõrkeid andmetes, palju seadmeid jagavad sama kanalit[19].

<sup>2</sup>"Beacon" kaader sisaldab kogu informatsiooni võrgu kohta. Neid väljastatakse perioodiliselt, et teatada traadita LAN olemasolust ning sünkroniseerida erinevaid seadmeid[20].



Joonis 4. ZigBee toetatud topoloogiad[22]

### 4.3 Võrdlus konkureerivate juhtmeta standarditega

Tabelis 1 on kujutatud ZigBee võrdlus konkureerivate juhtmeta standardite Bluetooth ning Wi-fi-ga [23].

Tabel 1. Võrdlus juhtmeta standardite vahel[23]

Parameeter	Bluetooth	Wi-Fi	ZigBee
Sagedusriba	2.4GHz	2.4GHz	2.4GHz
PHY/MAC kihid	IEEE 802.15.1	IEEE 802.11b	IEEE 802.15.4
Raadius	9m	75 kuni 90m	Siseruumides: kuni 30m. Välitingimustes: kuni 100m
Voolutarve	60 mA (Tx režiim)	400mA (Tx režiim) 20mA (Ooterežiim)	25-35mA(Tx režiim) 3µA (Oote režiim)
Toores and-meedastuskiirus	1 Mbps	11 Mbps	250 Kbps
Protokolli kuhja suurus	250KB	1 MB	32KB 4KB(piiratud funktsionaalsuseda lõppseadmetes)
Tüüpiline võrguga liitumise aeg	>3 sekundit	Varieeruv, Tüüpiliselt 1 sekund	tavaliselt 30 ms

Jätkub...



Tabel 1 – *Jät kub...*

<b>Parameeter</b>	<b>Bluetooth</b>	<b>Wi-Fi</b>	<b>ZigBee</b>
Interferentsi vältimise meetod	FHSS(frequency-hopping spread spectrum)	DSSS(direct-sequence spread spectrum)	DSSS(direct-sequence spread spectrum)
Maksimaalne arv sõlmesid võrgus	7	32 iga pöörduspunkti kohta	64 K
Kanalite arv	19	13	16

## 4.4 ZigBee võrgu turvalisus

Tänapäeval on vägagi aktuaalseks saamas "asjade internet", ning ZigBee protokoll leiab selles palju kasutust. Kuna "asjade interneti" kuuluvad seadmed on põhiliselt igapäevases kasutuses koduseadmed, nagu näiteks printerid, lukud, pesumasin, siis tuleks erilist rõhku pöörata turvameetmetele ning privaatsusele.

ZigBee spetsifikatsioon tagab võimaluse ülimalt turvaliste juhtmeta võrkude loomiseks. Need turvaseaded pole samas kohustuslikud ning turbetase sõltub sellest, milliseid meetmeid toote tegija on valinud implementeerida. Näiteks ukسلukk vajaks turvalisemat lahendust kui lihtne termomeeter.

### 4.4.1 Turvalisuse tagamise meetmed

Traditsioonilises turvatud ZigBee võrgus on turvamine tsentraliseeritud - see tähendab, et eksisteerib üks sõlm, mis saab autentida seadmeid, kes üritavad võrguga liituda, lubada või keelata võrgu liikmesust ning jagada välja turvavõtmeid. See sõlm kannab nimetust usalduskeskus (Trust Center) ning tavaliselt on selles rollis võrgu koordinaator, kes lõi võrgu.

Sõlme autentimise protsessi käigus saadetakse liituvale seadmele krüpteeritud võti. See võti on juhuslikult genereeritud ning on kõikidel seadmetel võrgus sama. Seda võtit peavad kasutama kõik sõlmed, et krüpteerida ning dekrüpteerida protokollis andmeid, mida vahetatakse. Kui võti edastatakse uuele sõlmele, siis võrguvõti on krüpteeritud teise eelnevalt konfigureeritud võtmega, mis on teada ainult usalduskeskusele ning liituvale

sõlmele.

Samuti on võimalik, et kaks sõlme jagavad omavahel unikaalset võtit, et krüpteerida/dekrüpteerida kommunikatsioone üksteise vahel. See võti tagab veel ühe turvaseme. Mõningad ZigBee rakendusprofiilid, näiteks Smart Energy, kasutavad sertifikaadipõhist võtme kehtestamist (CBKE), et kehtestada unikaalne võti turvaliseks suhtluseks. Iga seade võrgus peab hoidma sertifikaati, mis on väljastatud usaldatud autoriteedi poolt. See meetod pakub võimaluse turvaliselt tuvastada seade ning lubada tal alustada andmevahetust.

ZigBee 3.0 versioonis on olemas võimalus ka detsentraliseeritud turvalisuse haldamise jaoks. Sellises võrgus iga ZigBee marsruuter autentib ja jagab võtmeid seadmetele, mis üritavat võrguga liituda marsruuteri lapsena. Sellisel juhul ei ole ka tsentraalset sõlme, mis näeb kõiki lubatud sõlmesid.

#### **4.4.2 Turvaohud**

Põhilisteks turvariskideks ZigBee traadita võrgus on :

- Tundliku andme vargus sõlmest - see võib olla kasutaja andmed aga tõenäolisemalt võrgu turvaandmed, nagu krüpteerimisvõtmed, millega saab ligi ka ülejäänud võrgule. Seega on väga oluline hoolega valida turvavõtmed ning hoida neid hoolega.
- Sõlme enda vargus - keegi võib võtta endale ühe sõlme ning viia ta teise võrku, kus temale saab juurdepääsu.
- Volitamata sõlme juhtimine - keegi saab püüda raadiokanalist konkreetseid kaadreid ning neid hiljem sõlmele uuesti saata, et saavutada kontroll. ZigBee kasutab selle vastu kaadri loendureid, kuid samuti aitab selle vastu regulaarselt võrguvõtme vahetamine.
- Võrguteenuse kadumine - näiteks keegi ummistab raadiokanali või kogu laiusriba interferentsiga. Rakendused, mis saavad muuta oma sagedusala vaiksema kanali peale, on siinjuhul väga head.

Kõige vähem turvaliseks ehk seega ka kõige tähtsamaks elemendiks ZigBee võrgus osutub võrguvõti. Seega tuleb kõige rohkem tähelepanu pöörata selle kaitsmisele ning valimisele[24].

## 5 Püstitatud ülesande lahendus

Ülesandel on mitu iseloomulikku tunnust, millest tuleb lähtuda. Nendeks on:

- Keskkonna mõõtepunkti jälgimine
- Andmed edastatakse pilve
- Madal energiatarve
- Kasutatakse sidemoodulite jada

Nendest tulenedes, saab asuda konkreetselt analüüsima, kuidas probleemi lahendada ning milliseid vahendeid ja meetodeid oleks vaja kasutada võimalikult sobiliku tulemuseni jõudmiseks.

### 5.1 Probleemi analüüs

Esimeseks tähtsaks atribuudiks on keskkonna mõõtepunkti jälgimine. Selleks, et jälgida mingis keskkonnas kindlaid tingimusi, on vaja konkreetset mõõteandurit. Selleks võib olla tavaline temperatuurisensor, müra sensor, niiskuse sensor - võimalusi on palju. Püstitatud ülesande kontekstis ei mängi see erilist rolli, mis andmeid konkreetne sensor mõõdab, oluline on vaid põhimõtteline sensori olemasolu. Ilmselt ei ole ka siis oluline suur andmekiirus, vaid saab hästi hakkama ka palju "aeglasema" võrguga.

Teisena, oluline on, et andmed edastatakse pilve - ei mängi rolli, millist pilvevõimalust kasutatakse - kas andmebaasi või näiteks lihtsalt faili. Peamine on see, et andmed oleksid kättesaadavad läbi interneti. Siis ei ole vaja andmete kättesaamiseks vaja otseselt olla ühenduses pilveandmepunktiga ning seega laiendub ka lahenduse kasutusvõimalused.

Järgmised kaks tunnust, madal energiatarve ning sidemoodulite jada kasutus, käivad käsikäes, see tähendab, et analüüsides energiatarvet, peame meeles pidama, et lahenduses kasutatakse sidemoodulite jada. Seega kitsendub potentsiaalsete tehnoloogiate valik märgatavalt. Sidemoodulite valikul tuleb peale energiatarbe veel kaaluda mitmeid erinevaid võimalusi, nendeks on näiteks võrkude moodustamise võimalused, turvameetmed, leviala ning antennide võimalused, toitevõimalused, olemasolevad väljaviigud, toetatud tarkvara.

Kuna keskkonna mõõteandur ning pilvekeskkond andmehoidjana on mõlemad üsnagi paindlikud - see tähendab, et nende konkreetne valik ei mängi ülesande kontekstis nii suurt rolli, siis kõige targem on alustada sidemoodulite valikust. Samuti võib sensori valik sõltuda sidemoodulite valikust - mõningaid keerulisemaid sensoreid ei toetata igalpool.

## 5.2 Sidemoodulite tehnoloogi valik

Sidemoduleid valides lähtuti ka eeldusest, et keskkonna mõõtepunkti seire võiks olla võimalusterohke, see tähendab, et mõõtepunkt võib olla siseruumides, aga ka asutustest kaugel looduses, siis mõistlikkuse ning mugavuse mõttes saab järeldada, et sidemoodulid peaksid olema võimelised suhtlema juhtmeta. Seega koheselt kitseneb tehnoloogiate valik ning esile kerkivad järgmised :

- Wi-Fi
- Bluetooth Low Energy
- ZigBee
- Z-Wave

Valiku tegemisel tuleb prioritiseerida esmalt madala energiatarbe poole.

### 5.2.1 Wi-Fi

Wi-Fi kasutab tüüpiliselt märgatavalt rohkem energiat kui näiteks Bluetooth seadmed. Seda põhiliselt tugevama signaali saatmise tõttu. Mõningates rakendustes võib Wi-Fi kasutada isegi kuni 40 korda rohkem energiat kui Bluetooth. Näiteks andmete saatmine 75 baiti sekundis üle Wi-Fi nõuab ligikaudu 80 mW voolu, kuid Bluetoothi kaudu ainult 2 mW[25]. Seega saame välistada valikust Wi-Fi moodulid energiatarbe pärast.

### 5.2.2 Z-Wave

Z-Wave on natuke vähem tuntud, kuid väga paljulubav madala voolutarbe ning andmemahuga tehnoloogia. Ta toimib 800-900 MHz laiusriba peal ning seega ei ole konfliktis Wi-Fi-ga, mis töötab 2.4GHz laiusribal. Samuti on erinevate Z-Wave seadmete vahel hea koostalitlusvõime (sarnaselt ZigBee-le)[26]. Z-Wave võrgul on samuti üsna suur ulatus - seadmed saavad olla ühendatud kuni 160 meetri kaugusele[27].

Samas Z-Wave oli kuni alles 2019 aastani kinnine standard, mis tähendab, kui kellelgi oleks vaja teha muudatusi protokollis, siis seda teha ei saadud ning ainult üks kindel firma

vastutas Z-Wave kiipide tootmise eest. Seega oli varem raskem luua Z-Wave seadmeid kõrvalistel tegijatel ning kasutada selle kogu potentsiaali[26].

Siiski Z-Wave paistab vägagi sobilik tehnoloogia ülesande lahendamiseks.

### **5.2.3 Bluetooth Low Energy**

Bluetooth Low Energy (BLE) toetab madala andmemahuga ühendusi üle lühemate vahemaade, pakkudes samal ajal väga efektiivset toitehaldust. BLE teostab klient/server arhitektuuri, mis võimaldab riistavaral implementeerida ainult kommunikatsiooni funktsionaalsust, säästes sedasi raha, toidet ning andmemahut. BLE võrgud saavad teoorias sisaldada väga suurt hulka seadmeid.

Klient-server mudeli tõttu on kommunikatsioon ka lihtsamini implementeeritav. Seadmed ei pea olema seotud, vaid saavad lugeda ning saata andmeid asünkroonselt ning vastavalt vajadusele koheselt.

BLE suurimaks piiranguks osutub see, et tema võrkude füüsiline suurus on limiteeritud kvaliteetse ülekandesuutlikuse tagamise korral umbes 10 meetri ringi. Suurtematel distantsidel ei saa enam garanteerida sama ülekandesuutlikkust[28]. See on palju madalam kui teistel eelnevalt väljapakutud tehnoloogiatel. 10 meetrit on hea ning piisav kodukontoris, kuid pole nii paindlik oma võimalustes.

### **5.2.4 ZigBee**

Nagu pikemalt kirjutas peatükis 4, on ZigBee madala voolutarbe ja andmemahuga juhtmeta protokoll. ZigBee on mitmeid eeliseid võrreldes eelnevate protokollidega. Peamine eelis seisneb selles, et ZigBee abil saab luua väga mahukaid võrke, mis on palju suuremad kui ühe raadio ulatus. ZigBee võrk võimaldab isegi kuni 65000 sõlme ühes võrgus. Võrk konfigureerib end automaatselt ning suudab ennast dünaamiliselt parandada, kui mõned sõlmed peaksid kaduma või eemalduma. Samuti on tema standardi pärast lihtne panna omavahel suhtlema ZigBee seadmeid erinevatelt tootjatelt[28].

Seega ZigBee paistab samuti väga sobilik lahendus ülesandele.

## 5.2.5 Valitud sidemooduli tehnoloogia

Analüüsid sobilikke võimalusi, paistsid kõige sobilikemaks Z-Wave ning ZigBee. Seega tuleb teha valik nende vahel. Kui vaadata mõlema protokollide võimalusi siis, mõlemad sobiksid väga hästi antud ülesande konteksti. Siiski esinevad erinevused:

- Z-Wave võrgus saab seadmete vahel olla suurem vahemaa võrreldes ZigBee võrgus.
- ZigBee võrk lubab rohkem hüppeid seadmete vahel keskpunktini. Z-Wave võrgus on lubatud teha kokku ainult 4 hüpet, kui seade koos endale 3 ligidaima seadmega ei ulatu jaoturini, siis ühendus kaob. ZigBee võrgus saab teha nii palju hüppeid kui vaja.
- ZigBee vajab vähem energiat.
- Z-Wave töötab vähem kasutatud raadiosagedusel, seega väldib interferentsi näiteks Wi-Fi kanalitel[27].

Järelikult Z-Wave kasutades saab maksimaalselt sidemoodulite jadas kasutada 5 seadet, 1 keskpunkt ning 4 järjest kaugemal olevat seadet. See paneb range piirangu peale sidemoodulide jada kaugusele. Seadmete vahel saab küll olla suurem vahemaa kui ZigBee võrgus, kuid ZigBee võrgus saab selle arvelt moodustada suurema jada. Samas kui on teada konkreetse keskkonna mõõtmed ning see ei piirdu Z-Wave ulatusest välja, siis ilmselt oleks targem kasutada Z-Wave seadmeid, eriti kuna ta väldib võistlemist Wi-Fi sagedustel. Pakutud ülesandes seda ei ole mainitud, ehk eeldatavalt valitakse lahendus, mis on skaleeritav ning paindlikum.

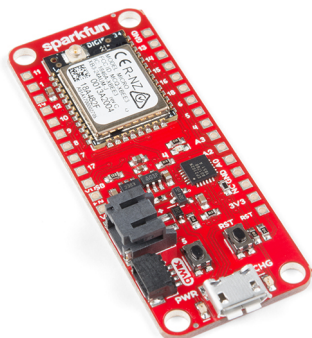
Vaadates puhtalt energiatarbe koha poolt, siis võidab ZigBee, kuid pikemate vahemaade peale nõutakse rohkem seadmeid. Kui on võimalik kasutada Z-Wave seadmeid, siis tõenäoliselt oleks nende kogu energiatarve madalam või sarnane ZigBee ahelaga, kuna seadmete arv on samuti madalam.

Tuleb tõdeda ka tõsiasi, et Z-Wave on olnud suurema enamuse oma elueast kinnine standard, ehk seadmeid on raskem leida ning hinnad on kallid võrreldes ZigBee seadmetega[27]. Samuti tuleb kaaluda mõlemale tehnoloogiale lähenemise lihtsust. Siinkohal on ees ZigBee, millel on olemas vägagi head kasutajasõbralikud edasiarendused just selle tõttu, et ta on olnud avatud standard[29].

Järelikult sobivad pakutud ülesande konteksti kõige paremini ZigBee-d kasutavad moodulid.

### 5.3 Valitud sidemoodul

Arvestades põhiliselt moodulite kättesaadavust ning ka sobivust, valiti sobivaks sidemooduliks SparkFun Thing Plus - XBee 3 Micro (U.FL) (Joonis 5).



Joonis 5. *SparkFun Thing Plus - XBee 3 Micro (U.FL)*

Tegemist on mooduliga, mis pakub XBee 3 (ZigBee 3.0) võimalusi. Plaat on kompaktne, 20 sisend/väljundviiguga ning toetab liitium-polümeer akut viikude kinnijootmiseta. Samuti toetab plaat U.FL tüüpi ühendusega antenni pikema vahemaade kommunikatsioonide jaoks. Andmevahetust saab viia läbi siseruumides kuni 60 meetri kaugusele ning välitingimustes otse vaateväljas isegi üle kilomeetri kaugusele[30].

ZigBee 3.0 on küllaltki uus arendus, nimelt avalikule publikule lasti ta välja 2016 aasta detsembris[31].

Mooduli konfigureerimine toimub läbi AT-käsustiku või kasutades Digi poolt loodud variante, millest täpsemalt on kirjas järgmises alampeatükis. Sättestamiseks saab moodulit ühendada arvutiga kasutades micro-USB kaablit või ka läbi järjestikliidese[30].

Ülesande lahenduse kontekstis on vaja vähemalt kolme sidemoodulit, et moodustada ahel.

#### 5.3.1 Digi XBee

Valitud sidemoodul kasutab Digi XBee 3 moodulit ning see on ka üks tähtsamaid põhjuseid, miks just selline moodul ülesande lahendamiseks valiti.

XBee on erinevate raadiomoodulite perekond, mis on loodud Digi Internationali poolt. XBee brändi alla kuulub palju erinevaid raadioseadmeid ning kasutavad erinevaid protokolle nagu näiteks Wi-Fi või ZigBee. Mõningad XBee seadmed töötavad ZigBee pro-

tokolli põhjal, kuid mitte kõik XBee seadmed.

ZigBee on avatud standard, seega kiibid ja moodulid on kättesaadavad paljude erinevate tootjate käest. Avatud standard panustab palju tema väärtusesse. Digi on loonud oma implementatsiooni ZigBeest, mis on täielikult sertifitseeritud ZigBee Alliance poolt, kes haldab ZigBee spetsifikatsiooni. Standardiseeritud liides pakub kiire ning lihtsasti arusaadava sätestamise, juhtimise ning kommunikatsiooni võimaluse.

Digi XBee on süsteemne tervik, mis koosneb programmeeritavatest moodulitest aga ka adapteritest, tarkvarast, võrguvärvatest - kõik loodud eeldusega, et kiirendada projektide arendust. Kõik XBee moodulid jagavad sama käsustikku ning liidest, ehk on võimalik isegi vahetada protokolle ilma, et füüsiliselt midagi muutma peaks. Samuti on need moodulid olemas erinevates suurustes ning antennide ühenduse valikuga.

XBee micro formaat on ideaalne kompaksete ning portatiivsete lahenduste jaoks[32]. See on täpselt sobilik pakutud ülesande lahenduse jaoks.

### **XCTU arenduskeskkond**

Suur osa sellest, mis ajendab kasutama XBeed, on kasutajakogemus. Selleks on Digi loonud oma üleplatvormilise rakenduse XCTU (Joonis 6).

XCTU on tasuta rakendus, mis on loodud eesmärgiga pakkuda arendajatele lihtsasti kasutatavat graafilist kasutajaliidest Digi raadiomoodulitega suhtlemiseks. XCTU on ühilduv nii Windowsi, MacOS kui ka Linuxiga.

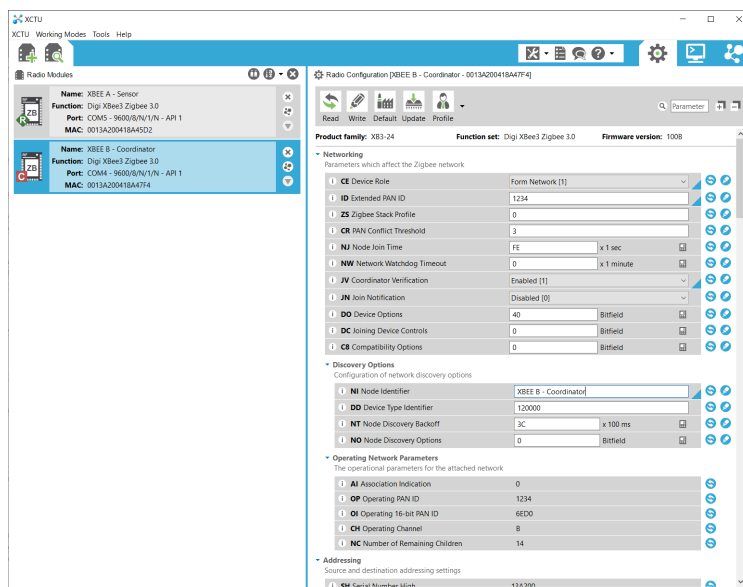
XCTU sisaldab endas kõiki tööriistu, mida võib arendajal vaja minna, et XBee moodul kiiresti tööle panna. XCTU-s saab näiteks seadistada mitmeid raadioseadmeid, uuendada otse seadmete püsivara, kasutada konsooli suhtlemiseks ja muud. Lisaks sellele on XCTU kohta olemas põhjalik dokumentatsioon[33].

## **5.4 Muud vajalikud vahendid**

Peale sidemooduli on veel vaja valida:

- Keskkonna mõõtesensor
- Pilveandmepunkti lahendus
- Akud
- Antennid





Joonis 6. *XCTU Arenduskeskkond*

### 5.4.1 Keskkonna mõõtesensor

Prototüüplahenduse keskkonna mõõtesensoriks sai valitud temperatuurisensor LM35 (Joonis 7). Tegemist on analoog temperatuurisensoriga, mille väljundpinge on lineaarselt proportsionaalne temperatuuriga, pinge tõuseb 10 mV iga 1°C temperatuuri kasvu kohta [34]. Ta on oma kasutuselt lihtne ning prototüüpimiseks igati sobilik.

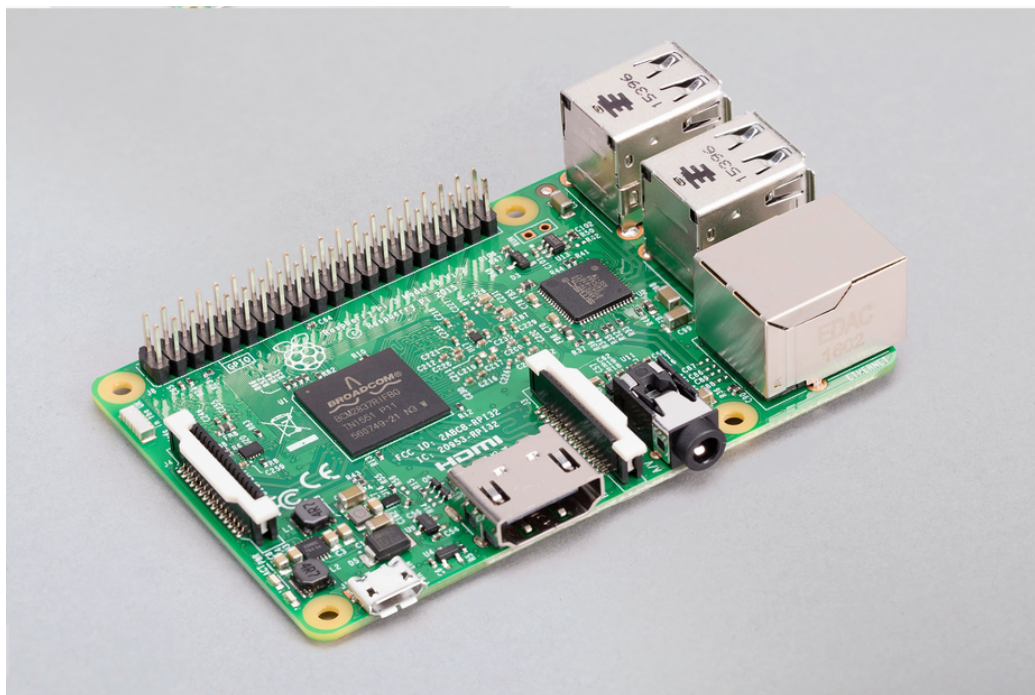
### 5.4.2 Pilveandmepunkt

Pilveandmepunktiks sai valitud Raspberry Pi 3 Mudel B versioon 1.2 (Joonis 8). Teised platvormid oleksid pakkunud täpselt sama funktsionaalsust ning seega polnud valik kriitiline. Sõlme saaks kasutada ka kui kontsentraatorit, mis saab koguda andmeid ning laadida pilve mitmetest sensorahelatest saabuvas infot. Nimelt Raspberry Pi on mõõtmelalt väike, aga samas toetab ühendusi üle USB, nii kaabliga interneti kui ka Wi-Fi-t. Läbi USB saab ühendada XBee plaadi otse Raspberry Pi külge. Pilve andmete laadimine toimib kasutades skripti, mis pannakse jooksma kohe kui arvuti käivitatakse. Seega ei oleks otseselt vaja ka ekraani peale algset seadistamist, vaid saab väga mugavalt ning ruumi kokkuhoidlikult pilveandmepunkt tööle panna.

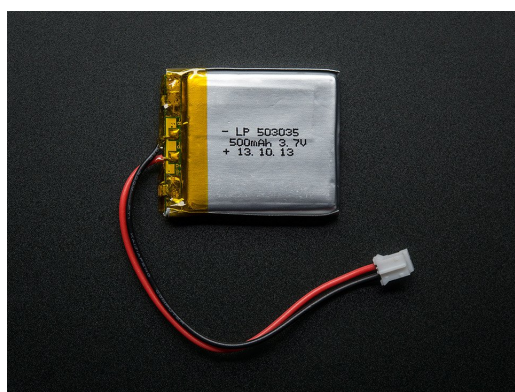
### 5.4.3 Akud

Valitud XBee moodulil on olemas pistik ning laadimisvõimalus plaadi küljes liitium-polümeer akude jaoks. Seega sai valitud kättesaadavuse alusel liitium-polümeer aku ADA1578, mis väljastab 3.7 - 4.2V pinget ning tema mahutavus on 500mAh. Akul on





Joonis 8. *Raspberry Pi 3 Model B*[35]



Joonis 9. *ADA1578 liitium-polimeer aku*[36]

naator vastutab võrgu ülal hoidmise eest, ei saa teda, ning pakutud ülesande juhul ei ole ka vaja ,panna unerežiimi. Kui üks moodul on koordinaatori rollis, siis teised moodulid saavad liituda dünaamiliselt võrguga ning käituvad kui marsruuterid, laiendades võrgu suurust.

XCTU arenduskeskkond muudab konfigureerimise märksa kasutajasõbralikumaks. Selleks, et üks seade koordinaatoriks määrata, tuleb ta ühendada arvuti külge, kus jookseb XCTU, lisada sinna vastav moodul ning määrata säte "CE Device Role" vastama valikule "Form Network[1]".

Seega on olemas koordinaator, kes loob võrgu ning võimaldab teistel moodulitel sellega liituda. Kuid selleks, et teised moodulid teaksid millise võrguga ühenduda, juhul kui näites



Joonis 10. *Bingfu antennid*[37]

tuvastatakse mitu erinevat võrku, tuleb ka määrata ära võrgu ID. Selle saab määrata sättes "ID Extended PAN ID". Tegemist on kuuteistkümnendiksüsteemis arvuga, ning ta peab olema igas moodulis sätestatud samaks, kui tahta, et kõik samas võrgus töötaksid.

Selleks, et hiljem oleks lihtsam XCTU-s moodulitel vahet teha aga ka programmikoodis kindlatele moodulitele viidata, on võimalik määrata ära ka iga mooduli sõlme identifikaator ehk põhimõtteliselt nimi. Selleks on "NI Node Identifier" säte, mis võib olla kuni 20 ASCII tähemärki.

Võimalik on ka muuta moodulite adresseerimist, kui on vaja andmeid liigutada konkreetsest seadmest teise konkreetmesse seadmesse, kuid mitte kuhugi mujale. Antud võrgus on sihtaadressiks koordinaator ehk marsruuterite sihtaadress on koordinaatori aadress ning seda ei ole mõtet võrgu dünaamilisuse mõttes muuta. Kui sideahela lülid on järgalt kindlale järgmisele lülile adresseerima määratud, siis muutub uute lülide lisamine aeganõudvamaks ning tüütumaks[38].

### 5.5.2 Turvameetmed

Kuigi võrgu eesmärgiks on hetkel saada andmeid väga lihtsast temperatuurisensorist, mille turvamine ei paista esmapilgul oluline, peame arvestama selle turvamisega. Kui lahendust rakendatakse mõne tundlikuma andmekogumi kogumiseks, siis peab olema tagatud, et ei oleks võimalik saboteerida sideahelat või varastada andmeid.

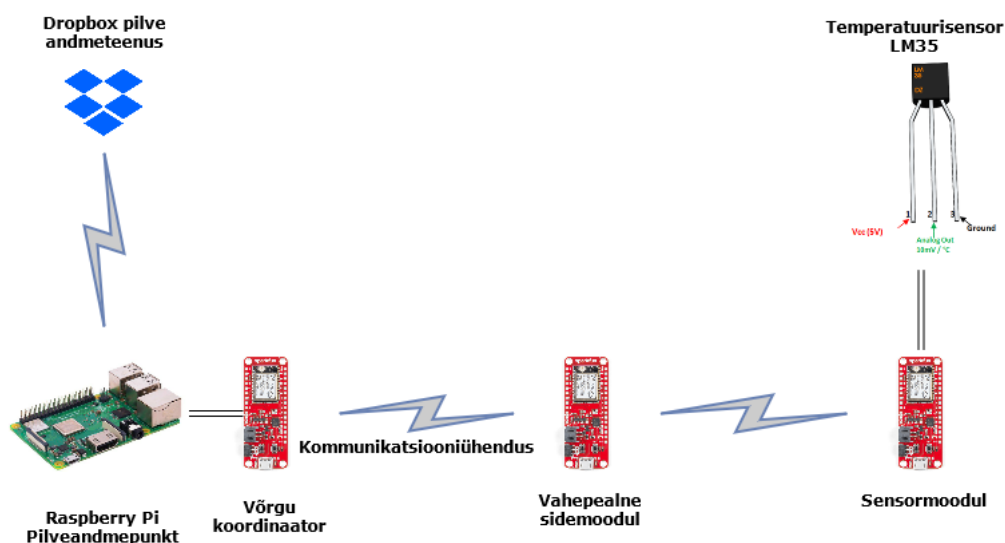
Peatükis 4 on toodud välja mõningad viisid kuidas ZigBee võrgus tavaliselt turvamine

töötab. Antud võrgu turvalisuse tagamiseks tuleb kõigepealt lubada krüpteerimine sättest "EE Encryption Enable". Turvasätete all on palju erinevaid võimalusi, kuid algselt on sätestatud võrk kasutama tsentraliseeritud usalduskeskust (Trust Center). Antud võrgus on selleks koordinaator. Lubades sättest krüpteerimise, hakatakse iga andmevahetust võrgus krüpteerima ja dekrüpteerima vastava võrguvõtmega. Võrguvõti on algselt määratud "0", mis tähendab, et kasutatakse juhuslikult genereeritud võtit - see on sobilik ning pigem just soovitatud.

Võrguvõti ise on juhuslikult genereeritud ehk praktiliselt võimatu ära arvata, kuid teisele sõlmedele saadetakse ta veel omakorda krüpteerituna teise võtmega (Link Key). See lisab veel ühe taseme võtme turvalisusele. Vaikimisi, genereeritakse see võti samuti juhuslikult, kuid siinpuhul pole see nii hea, sest võrguvõtme dekrüpteerimiseks on vaja just seda võtit. Seega liituvatesse sõlmedesse tuleb ta saata krüpteerimata. Asjaolu saab lahendada pakkudes eelkonfigureeritud võtme. Võrguvõtme dekrüpteerimisvõtme (Link Key) saab määrata seades "KY Link Key" ning see tuleb määrata igal võrgu sidemoodulil sama. Tegemist on kuni 128-biti pikkuse kuueteistkümnendsüsteemis väärtusega. Võtme saab ise määrata või soovituslikult kasutada mõnda juhuslikku võtmegeneraatorit, mis moodustab vägagi turvalise väärtuse[38].

### 5.5.3 Kindla mooduli spetsiifilised sätted

Kasutusel on 3 moodulit, üks neist on koordinaator, üks on keskmine sideahel, kes ainult edastab saadud signaali edasi ning viimane, kelle külge on ühendatud sensor ning ta edastab signaali (Joonis 11). Seega keskmine moodul ei peagi midagi rohkemat tegema, kui liituma võrguga ning signaale edastama, ehk tema konfiguratsioon on paigas.



Joonis 11. Lõpliku sidevõrgu topoloogia

Selleks, et oleks võimalik saada andmeid kätte läbi koordinaatori programmikoodi kaudu, on vaja aga lubada koordinaatoril API režiim. API režiim võimaldab näiteks Pythonis kirjutatud programmil avada ühendus temaga ühendatud seadmega ning kirjutada ja/või lugeda andmeid plaadilt. Selleks on vaja muuta koordinaatoris sätte "AP API Enable" valikusse "API Mode Without Escapes[1]" või "API Mode With Escapes[2]".

Selleks, et ühendada ning saada andmeid sensorilt on vaja sätestada sensoriga moodulil sisend/väljundviik. Kuna kasutusel on analoog temperatuurisensor, siis tuleb valida vastav sisend/väljund ning määrata ta analoogsisendiks. Selleks tuleb sätestada näiteks sätte "D3 DIO3/AD3 Configuration" valikule "ADC[2]". Samuti on sensori puhul oluline analoog referentspinge, nimelt sellest sõltub väärtus, mille saame väljaviiku lugedes. Seda sama referentspinget on ka vaja teada, et hiljem õigesti saadud väärtusest temperatuur arvutada. Vaikimisi on selleks 1.25 volti ning see on sobilik. Samuti tuleks sätestada sisend/väljund diskreetimissagedus, kui andmeid kogutakse harvemini, siis võib olla ajavahe suurem ning vastupidi. Siinjuhul määrati arbitraarselt selleks 100ms.[38].

Kogu sidemoodulie konfiguratsioon on toodud tabelina lisa 2-s.

## 5.5.4 Energiatarve

Spetsifikatsiooni järgi tarbib üks moodul vastuvõtja režiimis voolu 17mA ehk kasutades 500mAh akusid saab eeldada, et, optimeerimiseta energiatarvet, aku kestaks ligikaudu 29 tundi[39]. Enamus ajast peaks võrk siiski magama ning sedasi optimeerides oleks võimalik teoorias pikendada seda kestvust märgatavalt.

Kuna koordinaator on pidevalt ühendatud pilveandmepunktiga ning teda ka ei saa unerežiimi panna, siis ei ole tema energiatarbe suhtes võimalik ega ka vaja optimeerida.

Samas teiste sideahela lülide unerežiimi kaudu saaks energiatarvet märgatavalt vähendada ning seega jooksutada väga pikalt. Ideaalselt saaks moodulid sünkroniseerida oma unetsükli sedasi, et sensoriga moodul ärkab üles, saadab andmed järgmisele, kes täpselt just üles ärkas ja nii edasi. Kahjuks paistab, et unerežiimide sünkroniseerimiseks ZigBee 3.0 protokollil kasutades XCTU seadetes ei leidu. Seega pidi välja tulema muu staatilise lahendusega, mis on märksa ebaefektiivsem. Nimelt saab panna moodulid teatud ajaks unne ning teadud ajaks ärkvele, seega kui tahta, et andmed kindlasti alati ära saadetakse, peaks keskmise mooduli kogu tsükli aeg olema lühem kui sensoriga mooduli ärkveloleku aeg. Siin on näha probleemi, kus iga sideahela lisades peab kogu unetsükkel olema eksponentsiaalselt lühem kui eelmisel ning seega tuleb mingi hetk range piirang. Antud ülesande lahendusse see sobib, kuid pole just ideaalne. Siiski sedasi on võimalik saavutada

mitmeid kordi pikem aku kestvus.

XCTU-s on olemas ka võimalus kasutada MicroPythoni, mis on mikrokontrolleritele ning sardsüsteemidele tarkvaraliselt implementeeritud Python 3.0 versioon[38]. Selle võimalusi antud töö käigus ei realiseeritud, kuid tõenäoliselt on selle abil võimalik saavutada suuremal määral kontrolli moodulite üle, sealhulgas unerežiimide üle. See on ala, mis vajab edasiuurimist.

## 5.6 Andmete kättesaamine ning laadimine pilve

Selleks, et saada kätte temperatuurisensori poolt kogutud andmeid, kirjutati Pythoni programmeerimiskeeles programm (Lisa 4).

Pilve andmete laadimiseks loodi lihtne kontseptsiooni tõestus kasutades Dropboxi. Dropbox on failimajutusteenus ning temale on loodud hea dokumentatsiooni ja võimalustega tasuta Pythoni teek. Seega paistis tema kasutamine hea valik.

Programmi tööprotsess käib järgmiselt:

1. Kontrollitakse internetiühendust. Kui interneti pole, siis edasi ei minda senikaua kuni on.
2. Kontrollitakse, kas Dropboxile ligi pääsemiseks vajalik pilet (ligipääsuks vajalik juhuslikult genereeritud sümbolite jada) on olemas ning õige.
3. Laaditakse Dropboxist alla fail, mis sisaldab andmeid. Juhul kui seda pole, siis luuakse fail hiljem ise.
4. Üritatakse luua ühendus arvuti külge ühendatud koordinaatoriga.
5. Üritatakse saada võrgu andmed ning nende kaudu leida sensoriga seade üles.
6. Üritatakse lugeda seadme konkreetsest sisend/väljundviigult väärtus.
7. Arvutatakse väärtusest õige temperatuuriväärtus, kirjutatakse faili ning laetakse uuesti üles Dropboxi.

Sammud 3 kuni 7 toimivad pidevalt tsükliliselt määratud ajavahemiku tagant. Seega on olemas pidev sensori väärtuse lugemine.

Raspberry Pi moodulil on arvutussuutlikust sellise rakenduse jaoks küll ja veel ning seega on otstarbekas teha andmestiku teisendus tema poolt enne pilve laadimist.

Selleks, et LM35 sensorist saada kätte konkreetne temperatuuriväärus Celsiuse skaalal, tuleb teha saadud väärtusega arvutus. Nimelt analoogsisend käitub kui analoog-

digitaalmuundur ehk analoogsignaali muundatakse digitaalseks. Konkreetse ZigBee plaadi analoog-digitaalmuundur on 10-bitise resolutsiooniga[40], ehk sensorist saadakse väärtus vahemikus 0-1023. Seega kui teame, et referentspinge on vaikumisi 1.25V ning väärtused lähevad kuni 1023, siis saame kätte tegeliku pinge, kui korrutame saadud väärtuse 1.25-ga ning jagame 1023-ga. Nagu varem kirjutatud temperatuurisensori LM35 korral vastab iga 10mV pingele kasvule 1°C temperatuuri kasv, siis saame õige temperatuuri, kui korrutame eelnevalt saadud väärtuse 100ga. Näiteks sensorist saadud väärtuse 212 vastaks arvutades väärtus 0.259 ehk korrutades 100ga saame õigeks temperatuuriks 25.9°C.

Seega valem LM35 temperatuurisensorist õige temperatuuri saamiseks oleks:

$$T = 100 * (V_{ref} * \frac{lugem}{ADC_{res}}) \quad (5.1)$$

Joonis 12 näitab millisel kujul on andmed pilves.

```
10/05/2021 16:49:05 Temperature = 25.78 C
10/05/2021 16:49:54 Temperature = 25.66 C
10/05/2021 16:50:49 Temperature = 25.54 C
10/05/2021 16:51:43 Temperature = 25.66 C
10/05/2021 16:52:38 Temperature = 25.66 C
10/05/2021 16:53:33 Temperature = 25.66 C
10/05/2021 16:54:28 Temperature = 25.54 C
10/05/2021 16:55:22 Temperature = 25.66 C
10/05/2021 16:56:23 Temperature = 25.54 C
10/05/2021 16:57:18 Temperature = 25.54 C
10/05/2021 17:06:53 Could not find remote device within 30 seconds
10/05/2021 17:08:01 Temperature = 25.66 C
10/05/2021 17:08:56 Temperature = 25.66 C
10/05/2021 17:09:57 Temperature = 25.66 C
10/05/2021 17:10:57 Temperature = 25.66 C
10/05/2021 17:11:52 Temperature = 25.78 C
10/05/2021 17:12:32 Network configuration/detection error
```

Joonis 12. Andmete formaat pilves

## 5.7 Katsetused

Esimeseks katsetuseks võrguga oli kõik moodulid ühte võrku ühendada ning proovida andmevahetust läbi XCTU konsooli. Seal sai lihtsasti kirjutada sisse teksti ning teise mooduli konsoolist oli näha, kas need jõudsid kohale või mitte. Sedasi on lihtne näha, kas moodulid saavad üksteise sõnumeid kätte. Samuti on XCTU-s võimalus võrku skännida ning näha, mis seadmed on võrgus.

Seejärel sai järgmiseks katsetuseks ühendada sensor mooduli külge ning proovida Pythonis kirjutatud koodiga kätte saada andmeid sama plaadi pealt. Kui oldi XBee teekidega



tutvutud, sai andmed üsnagi kergelt kätte.

Kui andmed sai otse arvuti küljes olevalt moodulilt kätte, siis järgmisena oli mõistlik proovida panna sensoriga moodul tööle aku peal ning saada andmeid läbi koordinaatori, mis oli ühendatud arvuti külge. Katsetus hakkas tööle üllatavalt probleemivabalt. Seejärel prooviti koordinaator ühendada külge pilveandmepunktile ehk Raspberry Pi-le. Seoses Raspberry Pi peal koodi jooksutamiselega tuli uuesti installeerida Python ning vajalikud teegid. Samuti operatsioonisüsteemide erisuse tõttu on erinevad nimetused andmesideliidestel ning see tuli koodis vahetada. Siiski, programm hakkas tööle probleemideta.

Järgmisena prooviti viia sidemoodulid üksteisest piisavalt kaugemale, et moodustada ahel ning näha, kui kaugelt on võimalik sensorist andmeid kätte saada (Joonis 11). Raspberry Pi koos koordinaatoriga paiknesid toas, järgmised kaks moodulit viidi välistingimustesse ning prooviti paigutada maksimaalse vahemaa peale, kus andmeside veel ei katkenud. Saavutati maksimaalne kaugus ligikaudu 350 meetrit, mis on autori hinnangul hea arvestades tingimusi, kuid võimalik oleks palju paremini. Signaali võisid segada järgmised tegurid:

- Antenne polnud võimalik konkreetsetes välistingimustes sättida täielikult üksteise vaatevälja ehk esines mingil määral füüsilisi takistusi signaalile.
- Polnud võimalust antenne maast kõrgemale tõsta, see on seotud eelmise punktiga, kuid kõrgemale paigutades, oleks võimalik isotroopsel antennil potentsiaalselt paremini levida.
- Välistingimustes oli siiski vähemalt kaks moodulit alas, milles on pidevalt olemas Wi-Fi, ehk kuna ZigBee opereerib ka 2.4GHz sageduste peal, võis see mingil määral segada signaali.

Viimase katsetusena prooviti panna sidemoodulid madalama energiatarbega tööle, kasutades vahelduvat unerežiimi. Kuna nagu eelnevalt ka kirjas, ei leitud võimalust sünkroniseerida unerežiime ning tuldi staatilise võimaluse peale. Unetsükliid valiti sellised, et igal juhul sensormooduli ärkveloleku aja jooksul järgnev ahela moodul andmed kätte saaks. Nimelt sensoriga mooduli unetsükkel oli 6.6 sekundit unes ning 3.3 sekundit ärkvel. Keskmise sidemooduli unetsükkel oli 2 sekundit unes ning 1 sekund ärkvel. Sedasi töötas andmevahetus edasi. Tõdedes, et vastavalt 3.3 sekundit ning 1 sekund ärkvel on väga pikk aeg andmesidevõrkude maailmas, saab kindlasti neid veel madalamaks tuua. Samas tuleb ka mängu Pythoni programmikoodi sisesed viited, millega on raske arvestada ning sellise meetodi rakendamisel peaks kasutaja katse-eksituse meetodil ja/või konkreetseid andmevahetuse viitaseid mõõtma, et see võimalikult ära optimeerida. Olenemata sellest, saavutati esimese ideed tõestava katsega teoorias 3 korda pikem aku kestvus ning tõenäoliselt saab seda sama meetodiga veel mitu korda kõrgemaks viia.

## 5.8 Järeldused

Ülesande lahendamiseks valiti ZigBee moodulid, täpsemalt Digi XBee moodulid, just seepärast, et Digi on loonud kasutajasõbraliku arenduskeskkonna. Tõepoolest, XCTU võimaldas väga kasutajasõbralikult konfigurereida võrku, nii baastasemel kui ka seoses turvameetmete ja energiatarbega. Digi pakutav keskkond on juba hea põhjus, miks eelistada XBee kasutamist võrreldes näiteks Z-Wavega.

Kuna ZigBee võrgud töötavad väga dünaamiliselt, siis oli kerge testida võrku, sest moodulid liitusid võrguga väga kiirelt ning suurendasid koheselt ise võrku ehk hakkasid automaatselt tööle ning dünaamiliselt sõnumeid marsruutima.

Tänu lihtsasti kasutatavatele teekidele ning dokumentatsioonile oli üsna lihtne saada kätte sensori andmeid üle võrgu ning neid üles laadida.

Katsetamisel jäi võrgu ulatus spetsifikatsioonis kirjeldatud maksimaalsele ulatusele küll alla, kuid vahemoodulite parema paigutusega ruumis ja/või parema antenni soetades oleks võimalik viia sama arvu sidemoodulitega signaali mitmeid kordi kaugemale.

Ei leitud optimaalset lahendust ZigBee moodulite energiatarbe madalamaks viimise jaoks, peamiselt seepärast, et ei olnud võimalik konkreetselt sünkroniseerida unetsükleid moodulite vahel. Samas võivad eksisteerida muud tarkvaralised lahendused selle suhtes, kasutades näiteks MicroPythoni. Katsetused MicroPython-ga ei mahtunud käesoleva töö skoopi.

Vaadates tagasi ülesande iseloomulike tunnuste peale, siis said kõik lahendatud. Ainuke tegur, millega tegeledes ei leitud konkreetselt head lahendust oli madal energiatarve, kuid isegi seda arvestades kasutavad ZigBee moodulid energiat vähe ning leitud lahendus pikendas aku kestvust vähemalt 3 korda.

## 6 Kokkuvõte

Antud töö eesmärgiks oli luua madala energiatarbega sidevõrk, mis võimaldab koguda andmed keskkonna mõõtepunktist ning kasutab selleks sidemoodulite jada ja andmed laetakse üles pilve andmekogumisse.

Kirjeldatai andmesidevõrkude arengutrende, baastadmisi ning spetsiifilisemalt ZigBee protokollide olemust ja võrdlust konkureerivate standarditega.

Püstitatud ülesande lahendamise käigus analüüsiti probleemile iseloomulikke aspekte ning neile vastavalt valiti kõige sobilikum tehnoloogia ning vajalikud vahendid. Samuti kirjutati Pythoni programmeerimiskeeles programm, mis haldab sensori andmete lugemist ning pilve üleslaadimist. Kirjeldatai loodud programmi tööpõhimõtet ning ka moodustatud sidevõrgu üles seadmist.

Tulemusena valmis prototüüpvõrk, milles oli võimalik saata sensorist andmeid läbi sidemoodulite ahela pilveandmepunkti ning need pilve andmekogumisse üles laadida.

Ülesande algtingimustest lähtudes sai töö eesmärk täidetud, kuid neist ühele - madal energiatarve, ei leitud lahendust, mis oleks optimaalne ning autor pakkus selle koha peale välja võimaliku edasiarengu suuna.

## Kasutatud kirjandus

- [1] *2011 December Report of the Commissioner of the Environment and Sustainable Development*. Office of the Auditor General of Canada. 2011. URL: [https://www.oag-bvg.gc.ca/internet/English/parl\\_cesd\\_201112\\_05\\_e\\_36033.html](https://www.oag-bvg.gc.ca/internet/English/parl_cesd_201112_05_e_36033.html).
- [2] Thinnect, Inc. URL: <https://www.thinnect.com/case-studies/>.
- [3] Igor Golm. “ZigBee võrkudeprojekteerimine VKG Ojamaa kaevanduses”. Tallinna Tehnikaülikool Virumaa Kolledž, 2017.
- [4] ‘*communicare*’ – *WordSense Online Dictionary*. [Kasutatud: 03-05-2021]. URL: <https://www.wordsense.eu/communicare/>.
- [5] PhD Foluke Fatimayin. “What is communication?” [Kasutatud: 03-05-2021]. Oct. 2018.
- [6] *HISTORY OF COMMUNICATION*. History World. URL: <http://www.historyworld.net/wrldhis/plaintexthistories.asp?historyid=aa93>.
- [7] Tim Lambert. *A BRIEF HISTORY OF COMMUNICATION*. [Kasutatud: 03-05-2021]. 2021. URL: <http://www.localhistories.org/communications.html>.
- [8] Arzath Areeff. *History of Data Communication*.
- [9] William Stallings. “Data and Computer Communications Tenth Edition”. In: 2014. Chap. 1, pp. 33–34.
- [10] Starlink. 2021. URL: <https://www.starlink.com/>.
- [11] *Global Digital Population as of January 2021*. Jan. 2021. URL: <https://www.statista.com/statistics/617136/digital-population-worldwide/>.
- [12] *Number of internet users worldwide from 2009 to 2020, by region*. Oct. 2020. URL: <https://www.statista.com/statistics/265147/number-of-worldwide-internet-users-by-region/>.
- [13] William Stallings. “Data and Computer Communications Tenth Edition”. In: 2014. Chap. 1, pp. 34–36.
- [14] William Stallings. “Data and Computer Communications Tenth Edition”. In: 2014. Chap. 1, pp. 46–50.
- [15] William Stallings. “Data and Computer Communications Tenth Edition”. In: 2014. Chap. 1, pp. 56–62.

- [16] William Stallings. “Data and Computer Communications Tenth Edition”. In: 2014. Chap. 1, pp. 69–70.
- [17] *Wireless Standards*. RFID Centre Ltd. URL: [https://rfidc.com/docs/introductiontowireless\\_standards.htm](https://rfidc.com/docs/introductiontowireless_standards.htm).
- [18] Ankur Tomar, ed. *Introduction to Zigbee Technology*. Vol. 1. Global Technology Centre. July 2011, p. 2. URL: <https://www.cs.odu.edu/~cs752/papers/zigbee-001.pdf>.
- [19] *CSMA/CA: definition and explanation of the method*. IONOS Inc. Aug. 2019. URL: <https://www.ionos.com/digitalguide/server/know-how/csmaca-carrier-sense-multiple-access-with-collision-avoidance/>.
- [20] Jim Geier. *802.11 Beacons Revealed*. Aug. 2018. URL: <https://web.archive.org/web/20180805075033/http://www.wi-fiplanet.com/tutorials/print.php/1492071>.
- [21] *ZigBee Specification*. 21st ed. ZigBee Alliance. Aug. 2015, pp. 1–2. URL: <https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf>.
- [22] Ankur Tomar, ed. *Introduction to Zigbee Technology*. Vol. 1. Global Technology Centre. July 2011, p. 6. URL: <https://www.cs.odu.edu/~cs752/papers/zigbee-001.pdf>.
- [23] Ankur Tomar, ed. *Introduction to Zigbee Technology*. Vol. 1. Global Technology Centre. July 2011, p. 7. URL: <https://www.cs.odu.edu/~cs752/papers/zigbee-001.pdf>.
- [24] *Maximizing Security in ZigBee networks*. NXP. Jan. 2017. URL: <https://www.nxp.com/docs/en/supporting-information/MAXSECZBNETART.pdf>.
- [25] Elise Vogler. *Bluetooth vs. Wi-Fi Power Consumption*. URL: <https://itstillworks.com/does-wireless-internet-adapter-cost-4673941.html>.
- [26] Tristan Perry. *Zigbee vs Z-Wave vs WiFi vs Bluetooth: What’s Best?!* Sept. 2020. URL: <https://www.smarthomepoint.com/zigbee-zwave-wifi-bluetooth-comparison/>.
- [27] Josh Hendrickson. *ZigBee vs. Z-Wave: Choosing Between Two Big Smarthome Standards*. Nov. 2018. URL: <https://www.howtogeek.com/394567/zigbee-vs.-z-wave-choosing-between-two-big-smarthome-standards/>.
- [28] Digi International Eugene Fodor Software Engineering Manager. *Zigbee vs. Bluetooth: Choosing the Right Protocol for Your IoT Application*. Mar. 2020. URL: <https://www.digi.com/blog/post/zigbee-vs-bluetooth-choosing-the-right-protocol>.

- [29] *XBee Buying Guide*. Sparkfun. URL: [https://www.sparkfun.com/pages/xbee\\_guide](https://www.sparkfun.com/pages/xbee_guide).
- [30] *SparkFun Thing Plus - XBee3 Micro (U.FL)*. SparkFun Electronics. URL: <https://www.sparkfun.com/products/15435>.
- [31] Philipp Morgner et al. "Insecure to the Touch: Attacking ZigBee 3.0 via Touchlink Commissioning". In: 2017, p. 2. URL: [https://www.researchgate.net/publication/318408908\\_Insecure\\_to\\_the\\_Touch\\_Attacking\\_ZigBee\\_30\\_via\\_Touchlink\\_Commissioning](https://www.researchgate.net/publication/318408908_Insecure_to_the_Touch_Attacking_ZigBee_30_via_Touchlink_Commissioning).
- [32] Rob Faludi. *XBee vs. Zigbee: A Simple Comparison Guide*. Dec. 2020. URL: <https://www.digi.com/blog/post/xbee-vs-zigbee-a-simple-comparison-guide>.
- [33] *XCTU*. Digi International. URL: <https://www.digi.com/products/embedded-systems/digi-xbee/digi-xbee-tools/xctu>.
- [34] *LM35 Temperature Sensor*. Aug. 2017. URL: <https://components101.com/sensors/lm35-temperature-sensor>.
- [35] *Raspberry Pi 3 Model B*. Raspberry Pi Foundation. URL: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>.
- [36] *Lithium Ion Polymer Battery - 3.7v 500mAh*. Adafruit. URL: <https://www.adafruit.com/product/1578>.
- [37] *Bingfu 2 x RP-SMA WiFi Antenna 8dBi 2.4/5/5.8GHz and 15cm IPEX IPX U.FL to RP-SMA Female Cable Compatible with Intel Fenvi HP Dell Wireless Mini PCI Express PCIE Network Card WiFi Adapter*. URL: [https://www.amazon.de/-/en/gp/product/B07YCBGRRCR/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o00\\_s00?ie=UTF8&psc=1](https://www.amazon.de/-/en/gp/product/B07YCBGRRCR/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1).
- [38] *Security on the XBee*. Digi International Inc. May 2021. URL: <http://cms.digi.com/resources/documentation/digidocs/90001539>.
- [39] *Digi XBee 3 Zigbee 3 RF Module*. Digi International. URL: <https://www.digi.com/products/embedded-systems/digi-xbee/rf-modules/2-4-ghz-rf-modules/xbee3-zigbee-3#specifications>.
- [40] *Digi XBee 3 802.15.4 Migration Guide*. Digi International Inc. URL: <https://www.digi.com/resources/documentation/digidocs/pdfs/90002279.pdf>.

# Lisad

## **Lisa 1 - Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks**

Mina, Simo Oskar Juvanen

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose "Keskkonna mõõtepunkti jälgimine madala energiatarbega sidemoodulite jadaga", mille juhendaja on Kalle Tammemäe

1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

12.05.2021

## Lisa 2 - Sidemoodulite täiskonfiguratsioon

Tabel 2. Võrgu moodulite sätted

Parameeter	Moodul 1	Moodul 2	Moodul 3
CE Device Role	Join Network[0]	Join Network[0]	Form Network[1]
ID Extended PAN ID	2221	2221	2221
NI Node Identifier	XBee A - Sensor	XBee B - Router	XBee C - Coordinator
EE Encryption Enable	Enabled[1]	Enabled[1]	Enabled[1]
KY Link Key	59703373367-6397924423F-4528482B4D	59703373367-6397924423F-4528482B4D	59703373367-6397924423F-4528482B4D
SM Sleep Mode	Cyclic Sleep[4]	Cyclic Sleep[4]	No Sleep (Router)[0]
SP Sleep Time	294	C8	20 (default)
ST Wake Time	CE4	3E8	1388 (default)
AP API Enable	Transparent mode[0]	Transparent mode[0]	API Mode Without Escapes[1]
D3 DIO3/AD3 Configuration	ADC[2]	Disabled[0]	Disabled[0]
IR IO Sampling Rate	64	0	0
AV Analog Voltage Reference	1.25v reference[0] (default)	1.25v reference[0]	1.25v reference[0]



## Lisa 3 - Vahendite maksumus

Tabel 3. Vahendite maksumus

Komponent	Müüja	Ko- gus	Ühiku hind (koos KM)	Summa
SparkFun Thing Plus - XBee3 Micro	www.elfa-distrelec.ee	3	48.204 €	144.612 €
Micro USB 2.0 cable 0.5m Black	www.elfa-distrelec.ee	3	1.644 €	4,932 €
Bingfu 2 x RP-SMA WiFi Antenna 8dBi 2.4/5/5.8GHz and 15cm IPEX IPX U.FL to RP-SMA Female Cable	www.amazon.de	3	5.495 €	16.485 €
Li-Ion Polymer aku - 3.7V 500mAh	www.yeint.ee	2	10.46 €	20.92 €
SparkFun LiPo laadija, Micro-USB	www.yeint.ee	1	4.02 €	4.02 €
SanDisk Ultra 16GB microSDHC memory card	www.amazon.de	1	4.84 €	4.84 €
RaspBerry Pi 3 Model B+ 1GB RAM	www.elfa-distrelec.ee	1	47.66 €	47.66 €
LM35DZ Analog Temperature Sensor		1	1.39 €	1.39 €
Maketeerimislaud	robolabor.ee	1	3,60 €	3.60 €
Pikendusjuhe (ema-isa) 20CM (40TK)	robolabor.ee	1	3.00 €	3.00 €
Kokku:				250.069 €

## Lisa 4 - Programmikood kogutud andmete kättesaamiseks ning pilve üleslaadimiseks

```
# File Name      : "get_send_data.py"
# Author        : Simo Oskar Juvanen, 2021
# Last modified on : 07.05.2021
# Description :
# A relatively simple python application, made to receive data from a
# remote XBee
# module and upload it to be accessible from the internet/cloud
# storage.
#
# One module is connected to the host, which is running this
# application, and a
# remote module is sending data, received from a connected sensor.
# For that, Digi XBee python libraries are used.
#
# In terms of cloud storage, the chosen method is a simple Dropbox
# application.

import os
import sys
import time
from datetime import datetime

# Importing required Digi XBee libraries
from digi.xbee.devices import XBeeDevice
from digi.xbee.io import IOLine, IOMode

# Importing required dropbox libraries
import dropbox
from dropbox.files import WriteMode
from dropbox.exceptions import ApiError, AuthError

# Import required libraries to check internet connection
try:
    import urllib
except:
    import http.client as urllib

# Define the unique authorization(OAuth2) token of our DropBox
# application
Token = '
pbfpdys0rh4AAAAAAAAAAYNljeBul_9Y1Acgg6dq5d_smcx_6x3FwGBYbQJX9r83'
# Define local & remote file paths
LocalFilePath = os.path.join( os.path.dirname(os.path.abspath(__file__))
, 'data.txt')
RemoteFilePath = '/data.txt'

# Define local module com port and baud rate
PORT = "/dev/ttyUSB0"
BAUD_RATE = 9600
# Define remote node ID and sensor I/O line
REMOTE_NODE_ID = "XBEE A - Sensor "
IOLINE_IN = IOLine.DIO3_AD3

# Downloads remote data file, creates one if it does not exist
def download_file():
    # If local file with the same name already exists, delete it
```

```

# in case it contains wrong information.
if os.path.exists(LocalFilePath) :
    os.remove(LocalFilePath)
print("Downloading remote data file")
try:
    dbox.files_download_to_file(LocalFilePath,RemoteFilePath)
except:
    print("Remote file not found, creating new local file")

# Upload file to dropbox
def upload_file():
    print("Uploading file to Dropbox")
    with open(LocalFilePath,'rb') as f:
        try:
            dbox.files_upload(f.read(),RemoteFilePath,mode=WriteMode('
            overwrite'))
        except ApiError as err:
            print(err)

# Simple countdown timer
def countdownTimer(countTime):
    i = countTime
    while i >= 0:
        time.sleep(0.5)
        print("\r" + f"Next update in : {i} ", end="\r")
        i -= 0.5
    print(end="\n")

# Checks internet connection ( to google.com)
def checkInternet(url="www.google.com",timeout=3):
    connection = httplib.HTTPConnection(url,timeout=timeout)
    try:
        connection.request("HEAD","/")
        connection.close()
        return True
    except:
        return False

if __name__ == '__main__':
    countdownTime = 30

    # We should not go forward until we confirm that there is
    # a working internet connection
    internet = False
    while internet == False:
        internet = checkInternet()

    # Exit if there is no token defined
    if (len(Token) == 0):
        sys.exit("No Dropbox access token!")

    with dropbox.Dropbox(Token) as dbox:
        # Check if the token is valid
        try:
            dbox.users_get_current_account()
        except:
            sys.exit("ERROR: Invalid access token!")

        local_device = XBeeDevice(PORT,BAUD_RATE)

        while True:
            countdownTimer(countdownTime)

            # Download file from Dropbox
            download_file()

            # Opening local XBee device
            try:

```

```

        local_device.open()
    except:
        with open(LocalFilePath,"a+") as f:
            f.write(datetime.now().strftime("%d/%m/%Y %H:%M:%S") + "
                ") + "        Could not open local device\n")
        upload_file()
        continue
    # Set correct configuration for remote object
    try:
        network = local_device.get_network()
        remote_device = network.discover_device(REMOTE_NODE_ID)
        timestart = time.perf_counter()
        while remote_device is None:
            timeend = time.perf_counter()
            if timeend - timestart > 30:
                break
            remote_device = network.discover_device(
                REMOTE_NODE_ID)
            time.sleep(0.5)
            continue

        if remote_device is None:
            with open(LocalFilePath,"a+") as f:
                f.write(datetime.now().strftime("%d/%m/%Y %H:%M:%S") + "
                    :%S") + "        Could not find remote device
                    within 30 seconds\n")
            upload_file()
            local_device.close()
            continue

        remote_device.set_io_configuration(IOLINE_IN, IOMode.ADC
        )
    except:
        with open(LocalFilePath,"a+") as f:
            f.write(datetime.now().strftime("%d/%m/%Y %H:%M:%S") + "
                ") + "        Network configuration/detection error
                \n")
        upload_file()
        local_device.close()
        continue
    # Get Sensor data and convert to correct value
    # If the value is exactly 1023, that means that
    # the sensor might be disconnected.
    while True:
        try:
            value = remote_device.get_adc_value(IOLINE_IN)
        except:
            print("Could not get value, try again\n")
            countdownTimer(2)
            continue
        if value != 1023:
            break
        else:
            print("Value is 1023 - check sensor connection\n")
            with open(LocalFilePath,"a+") as f:
                f.write("Check sensor connection\n")
            upload_file()
            countdownTimer(countdownTime)
            continue

    convertedValue = 100.0 * (1.25 * value / 1023)
    strValue = datetime.now().strftime("%d/%m/%Y %H:%M:%S") + "
        Temperature = %0.2f C" % (convertedValue) + "\n"
    # Write to local file
    with open(LocalFilePath,"a+") as f:
        f.write(strValue)
    # Upload modified local file
    upload_file()
    local_device.close()

```