

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Tarmo Pajusaar 164275IASB

# **ANDMEHÕIVE HAJUSAS HÄÄLJUHTIMISEGA SÜSTEEMIS**

Bakalaureusetöö

Juhendaja: Arvo Toomsalu  
Lektor

Tallinn 2019

## **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: Tarmo Pajusaar

20.05.2019



TALLINNA TEHNIKAÜLIKOOL  
Arvutisüsteemide instituut

## Bakalaureusetöö ülesanne

*Üliõpilase ees- ja perekonnanimi: Tarmo Pajusaar*

*Üliõpilaskood: 164275IASB*

*Lõputöö teema eesti keeles:*

Andmehõive hajusas hääljuhtimisega süsteemis

*Lõputöö teema inglise keeles:*

Data Acquisition in a Distributed Voice Control System

*Juhendaja (nimi, töökoht, teaduslik kraad, allkiri):*

Arvo Toomsalu, lektor

*Konsultandid (nimi, töökoht): –*

*Lahendatavad küsimused ning lähtetingimused:*

1. Projekteerida korteris kasutatavate olmeseadmete talitluse korraldamisel hääljuhtimist rakendava süsteemi andmehõivega seotud osad, mis tuvastavad, eristavad ja reageerivad erinevatele lihtsatele häälkäsklustele.
2. Projekteeritav hääljuhtimissüsteem peab tagama kolme akustilise anduri, mis paiknevad erinevates ruumides, kontaktivaba andmeside süsteemi keskseadmega.
3. Hääljuhtimise süsteemi keskseadmena kasutada mikroarvutit Raspberry Pi.
4. Välja töötada hääljuhtimisega süsteemi riistvarakomponentidele sobiv elektritoitesüsteem.
5. Hääljuhtimisega süsteemi ohjel tugineda põhiliselt vabavaralisele tarkvarale.
6. Küsimused, mis seonduvad hääljuhtimisega süsteemi poolt juhitavate olmeseadmete koosseisu ja liidestamisega, ei kuulu antud projekti raames käsitlemisele.

*Eritingimused: puuduvad*

*Nõuded vormistamisele: Vastavalt Infotehnoloogia teaduskonnas kehtivatele nõuetele*

*Lõputöö esitamise tähtaeg: 20.05.2019*

Ülesande vastu võtnud: \_\_\_\_\_ kuupäev: 08.05.2019  
(lõpetaja allkiri)

## **Annotatsioon**

Bakalaureusetöös seatakse eesmärgiks projekteerida korteri või väiksema eramu olmeseadmete hajusale hääljuhtimissüsteemile andmehõivet läbiviivad osad koos sobiva toitesüsteemiga. Projekteeritav andmehõivesüsteem võimaldab akustiliste andmete kontaktivaba edastust keskseadmele kolmest eri ruumist. Keskseadme ülesandeks on akustiliste andurite poolt saadetud andmetest käskude tuvastamine. Keskseadme riistvaraks valitakse Raspberry Pi 3 Model B+. Akustiliste anduritena on kasutusel mikrofonimoodulid Adafruit AGC Electret Microphone Amplifier MAX9814. Andurite väljundsignaali digitaliseerimiseks ning kontaktivabaks edastuseks üle WiFi kasutatakse ESP32 mikrokontrolleri arendusplaati Adafruit HUZZAH32. Leitakse, et pärast integreeritud atenuaatori rakendamist mikrokontrolleri analoog-digitaalmuunduri sisendsignaalele sobitub mikrofonimooduli väljundpinge diapsoon analoog-digitaalmuunduri sisendpinge diapsooniga küllalt hästi, et puuduks vajadus täiendavate sobituskomponentide järele. Koostatakse andmehõiveseadmete maketid ning testitakse projekteeritud andmehõivesüsteemi toimimist.

Bakalaureusetöö on kirjutatud eesti keeles ning sisaldab teksti 27 leheküljel, 6 peatükki, 19 joonist, 2 tabelit.

## **Abstract**

### Data Acquisition in a Distributed Voice Control System

The main goal of the thesis is to devise a data acquisition subsystem for distributed voice control of domestic devices, together with the power supply solution to power the hardware components of the system. The data acquisition system is required to support cordless data communication between three acoustic sensors and the central unit. The thesis concentrates on data acquisition and only considers speech recognition to the extent that is necessary for verifying the concept and feasibility of the devised solution.

For the central unit, performing speech (command) recognition, microcomputer Raspberry Pi 3 Model B+ is chosen. Acoustical data acquisition is performed by Adafruit AGC Electret Microphone Amplifier MAX9814. Data digitization and wireless transfer over WiFi is done by Adafruit HUZZAH32 development board, equipped with an ESP32 microcontroller. The microphone amplifier is powered from the output of the +3.3 V voltage regulator on HUZZAH32. For HUZZAH32, a standard USB 5V charger is found to be a suitable power source. Raspberry Pi gets its power from an official Stontronics DSA-13PFC-05 power supply.

After applying a 6 dB internal attenuator to the input of the HUZZAH32 analog-to-digital converter, the output range from MAX9814 is found to match well enough to the input range of the converter. Thus, no additional components are considered necessary for building the data acquisition system.

Algorithms are provided for software on both the central unit and data acquisition units. Three data acquisition units are then breadboarded and the system verified against the desired behavior.

In conclusion, the chosen components and system architecture are found to work satisfactorily within the range of the goal set for the thesis.

The thesis is in Estonian and contains 27 pages of text, 6 chapters, 19 figures, 2 tables.

## Lühendite ja mõistete sõnastik

ADM	Analoog-digitaalmuundur
ISR	Katkestuse teenindusprogramm ( <i>Interrupt Service Routine</i> )
VAD	Kõne (hääle) olemasolu tuvastamine ( <i>Voice Activity Detection</i> )
ESP32	Mikrokontroller ESP32
ESP-IDF	Tarkvaraarenduse raamistik ESP32-le ( <i>Espressif IoT Development Framework</i> )
ESP-ADF	Audiorakenduste arendamise raamistik ESP32-le ( <i>Espressif Systems Audio Development Framework</i> )
FreeRTOS	Vabavaraline reaalaaja operatsioonisüsteem ( <i>Free Real-Time Operating System</i> )
GPIO	Üldotstarbeline sisend-väljund ( <i>General Purpose Input Output</i> )
AGC	Automaatne võimenduse reguleerimine ( <i>Auto Gain Control</i> )
EMA MAX9814	Mikrofonimoodul Adafruit AGC Electret Microphone Amplifier MAX9814
HUZZAH32	Mikrokontrolleri ESP32 arendusplaat Adafruit HUZZAH32
Raspberry Pi	Mikroarvuti Raspberry Pi 3 Model B+
Taimaut	Aegumine ( <i>timeout</i> )
CMP	Mikroarvuti
PS	Toiteplokk
A	Võimendi
BM	Mikrofon
U	Funktsionaalne üksus ( <i>unit</i> )
WA	Antenn
GUI	Graafiline kasutajaliides
IoT	Asjade internet ( <i>Internet of Things</i> )
OTA	Eetri kaudu ehk raadiosidet kasutades ( <i>Over The Air</i> )
LED	Valgusdiod ( <i>Light Emitting Diode</i> )
Objekt	Siin kasutusel tähenduses, nagu seda mõistetakse objektorienteeritud programmeerimises

## Sisukord

Autorideklaratsioon .....	2
Bakalaureusetöö ülesanne .....	3
Annotatsioon.....	4
Abstract Data Acquisition in a Distributed Voice Control System .....	5
Lühendite ja mõistete sõnastik .....	6
Sisukord.....	7
Jooniste loetelu .....	8
Tabelite loetelu .....	9
1 Sissejuhatus .....	10
2 Süsteemi põhikomponentide ja andmesidetehnoloogia valik.....	12
2.1 Keskseadme riistvara .....	12
2.2 Ruumiseadmete riistvara .....	13
3 Toitesüsteem.....	14
4 Süsteemi ülesehitus.....	16
4.1 Keskseade .....	16
4.1.1 Kõnetuvastuse allsüsteem.....	20
4.2 Ruumiseadmed .....	23
4.2.1 Pingete ühildamine .....	23
4.2.2 Andmete vastuvõtt keskseadmest.....	26
4.2.3 Andmete saatmine keskseadmele .....	27
4.2.4 ADM ja VAD .....	28
4.2.5 LED-indikaatori märguanded.....	31
4.3 Maketeerimine ja testid .....	32
5 Edasiarenduse võimalusi .....	34
6 Kokkuvõte .....	35
Kasutatud kirjandus .....	36

## Jooniste loetelu

Joonis 1. Raspberry Pi 3 Model B+.....	12
Joonis 2. Adafruit AGC Electret Microphone Amplifier MAX9814.....	13
Joonis 3. Adafruit HUZZAH32.....	13
Joonis 4. Raspberry Pi 3 Model B+ toiteadapter Stontronics DSA-13PFC-05.....	14
Joonis 5. Apple A1400 toiteadapter.....	15
Joonis 6. Andmehõivesüsteemi plokkskeem.....	18
Joonis 7. Ressursside eraldamine ruumiseadmete teenindamiseks.....	19
Joonis 8. Ühe ruumiseadme teenindamine keskseadmes.....	19
Joonis 9. Kõnetuvastuse allsüsteem lihtsa automaadina.....	22
Joonis 10. Käsustiku JSGF grammatikakirjeldus.....	22
Joonis 11. HUZZAH32 elektritoitega seotud väljundviigud.....	23
Joonis 12. HUZZAH32 analoogsisendid.....	23
Joonis 13. ADMi sisendpinge ning EMA MAX9814 väljundpinge diapasooneid.....	25
Joonis 14. Mikrofonimooduli väljundpinge.....	26
Joonis 15. Keskseadme saadetud vastuste või korralduste vastuvõtt ruumiseadmes.....	27
Joonis 16. Andmeedastus ruumiseadmelt keskseadmele.....	28
Joonis 17. ADMi taimerikatkestust teenindava programmiosa vooskeem.....	30
Joonis 18. LED-indikaatori juhtimine.....	31
Joonis 19. Süsteemi olekud makettidega testimisel.....	33



## **Tabelite loetelu**

Tabel 1. Kommunikatsioonisignaali koodid.....	20
Tabel 2. LED-indikaatori märguanded.....	32

# 1 Sissejuhatus

Häälkäsklustega juhtimine on kasvav trend, mida tehnoloogia – nii vahendite kui oskuste – areng ühelt poolt soodustab, ja milleks teiselt poolt peaaegu sunnib. Esiteks kasvab meid ümbritsevate asjade hulk järjest, ehk inimesel on asjadega järjest rohkem tegemist. Hääljuhtimine, vastandatuna traditsioonilisele vajuta-ja-kirjuta juhtimisele, võimaldab käed vabaks jätta ning läbimõeldud korralduse juures juhtida kasvõi mitut seadet korraga, ka selliseid, mis omavahel ega mõne tsentraalse seadmega ei suhtle. Teiseks on hääljuhitavad seadmed kahtlemata huvitavad, sest tekitavad kogemuse, justkui asjad tõesti mõistaks meid. Kolmandaks ja mitte vähetahtsana oleme harjunud suhtlema eelkõige kõneldes.

Hääljuhtimisega süsteemi üks põhikomponente on muidugi kõnetuvastussüsteem. Kõnetuvastussüsteem omakorda eeldab andmehõivesüsteemi kasutamist, kuivõrd ilma sisendandmeteta või nende mitterahuldava kvaliteedi korral ei saa ka parim kõnetuvastuse algoritm anda rahuldavat tulemust. Kõnetuvastus on reeglina ressursinõudlik, selle praktiline rakendamine seisneb kompromisside tegemises tehniliste võimaluste ning tuvastustäpsuse ja -kiiruse vahel [1]. Hajusad hääljuhtimisega süsteemi projekterides tuleb ühe olulise põhimõttelise otsusena lahendada küsimus, kas süsteemis hakkab ka kõnetuvastus toimuma hajusalt või eraldatakse selleks mingi tsentraalne seade. Üheselt parimat lahendust siin ei ole. Valiku tegemisel tuleb lähtuda eelnimetatud kompromissidest, aga ka nõuetest ning ootustest süsteemi käitamisele ja haldamisele.

Bakalaureusetöö uurimisobjektiks on hajusa hääljuhtimissüsteemi andmehõive väiksemas korteris või eramus. Süsteemilt oodatakse töös paindlikkust, st võimalust viia süsteemi käitumisse (käsustikku) kiiresti sisse muutusi, ning kiiresti ja suhteliselt odavalt laiendada süsteemi uutesse ruumidesse. Nende kriteeriumite alusel valitakse lähenemisprintsipiiks nimelt tsentraalne kõnetuvastus (kiire muutmine) ja n-ö kerged ruumiseadmed (odav laiendamine).

Tsentraalseks kõnetuvastusseadmeks on selles töös mikroarvuti Raspberry Pi 3. Raspberry Pi-d on kõnetuvastuseks muidugi varem juba rakendatud, seda nii kõnetuvastustehnoloogiat sügavuti, algoritmide tasemel uurides [2] kui ka iseseisvana, olemasolevat kõnetuvastustehnoloogiat lihtsalt kasutades [3]. Käesolev töö läheneb kõnetuvastusele viimati nimetatud printsiibil ja keskendub küsimusele, kuidas tuua olemasolevaid kõnetuvastuse vahendeid reaalsele rakendatavusele lähemale.

Rääkides hääljuhtimisega süsteemist, mis ühelt poolt on hajus, kuid milles teiselt poolt eraldatakse kõnetuvastuseks mingi tsentraalne seade, tõstatub loomulikult moel andmehõive korraldamise küsimus – kuidas jõuavad juhtimiskorraldused hajusast osast tsentrisse. See küsimus on töö põhifookuses.

Terminite kasutamisel on töös võimalusel lähtutud e-Teatmik [4] toodud tähendustest.

## 2 Süsteemi põhikomponentide ja andmesidetechnoloogia valik

Süsteemi komponentide valikul tuli ühelt poolt lähtuda ülesandepüstituses nimetatud nõuetest ja kitsendustest, teisalt oli otstarbekas arvestada selliseid tegureid nagu komponentide kättesaadavus, maksumus, kogukonna tugi ning varasem kogemus nende kasutamisel. Kontaktivabadest andmesidetechnoloogiatega otsustati kasutada WiFi-t, sest vastav võimalus on tänapäeval enamasti kodudes juba olemas või kergesti tekitatav. Ka annab see võimaluse laiendada projekti suuremale hulgale seadmetele ning vajadusel kasutada jõudluse tõstmiseks mikroarvuti asemel keskseadmena kasvõi mõnd serverit.

### 2.1 Keskseadme riistvara

Varasemas töös realiseeriti lokaalse hääljuhtimissüsteemi andmehõive osa mikroarvuti Raspberry Pi 3 Model B+ USB-porti ühendatud mikrofonimooduliga ning lisati sellele ka kõnetuvastuseks vajalikud tarkvarakomponendid [3], [5]. Et Raspberry Pi 3 Model B+ oli käesoleva tööga alustades Raspberry Pi mudelitest ka suurima jõudlusega, loeti töös põhjendatuks kasutada keskseadmena sedasama mikroarvutit (joonis 1). Kirjeldatud mikroarvutil on olemas WiFi suutlikkus [6].

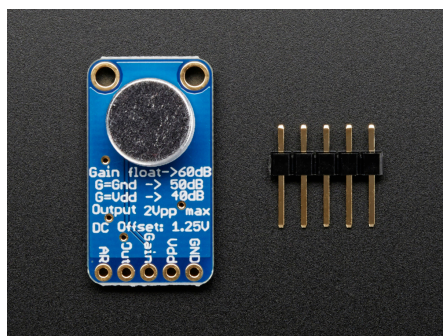


Joonis 1. Raspberry Pi 3 Model B+.

## 2.2 Ruumiseadmete riistvara

Ruumiseadmed paigutatakse eri ruumidesse. Nende ülesanne on registreerida vastavas ruumis inimkõnet (helilaineid), teisendada see elektriliseks analoogsignaals, digitaliseerida ning lõpuks edastada andmesidekanali kaudu keskseadmele. Seetõttu peab ruumiseade sisaldama nii mikrofoni, mikrofonivõimendit, analoog-digitaal muundurit (ADM) kui ka andmesidesuutlikkust.

Projekteeritud lahenduses täidab kaht esimest ülesannet mikrofonimoodul Adafruit AGC Electret Microphone Amplifier MAX9814 [7] (joonis 2). Seadme AGC kohandab võimendustegurit helirõhu järgi automaatselt, mis aitab parandada kõnetuvastuse kvaliteeti. [ibid]



### 3 Toitesüsteem

Kuna projekteeritav süsteem on hajus, osutus otstarbekaks lahendada selle komponentide toitmine hajusalt, vahetult energiavajaduse tekkimise kohas.

Keskseade Raspberry Pi 3 Model B+ töötab +5,1V toiteallikaga üle mikro USB-liidese. Olenevalt kasutatavatest perifeerseadmetest jääb Raspberry Pi voolutarve vahemikku 500...1000 mA [10]. Töös arendatav hääljuhtimisega süsteem vajab perifeerseadmetest üksnes WiFi moodulit, seega jääb voolutarve alla 1000 mA. Tootja poolt soovitatav toiteadapter on Stontronics DSA-13PFC-05 [11] (joonis 4). Adapter on ette nähtud sisendpingele ja -voolule 100...240 V 50/60 Hz 0,5 A ning annab välja 5,1 voldise alalispinge koormusvooluga kuni 2,5 A. [12]



Joonis 4. Raspberry Pi 3 Model B+ toiteadapter Stontronics DSA-13PFC-05.

Mikrokontrolleri arendusplaat HUZAZH32 on ette nähtud töötama mikro USB kaudu liidestuva toiteallikaga. Seadmesse on integreeritud 3,3 V väljundpingega ning kuni 500 mA tippkoormatavusega pingeregulaator. Kuni pool sellest voolust kulub mikrokontrollerile endale, regulaatori väljundit saab niisiis kasutada vaid väiksema energiatarbega seadmete toitmiseks [8]. Arendusplaadiga on võimalik ühendada ka aku, mida USB-toite olemasolul laetakse ja mille energiat USB-toite katkestuse korral kasutatakse [ibid]. Käesolevas töös polnud see vajalik, kuid potentsiaalse laienduse korral suurematesse hoonetesse ja/või kriitilisemate funktsioonide juhtimisel võib akutoide osutada väga kasulikuks. Tootja soovib püsiinstallatsioonide jaoks 5,25V/1A

adapterit, kus 5,25 V on USB-toite lubatud maksimum. Täiendav 0,25 V on adapteril mõeldud kompenseerima USB-seadmete suure voolutarbega kaasnevat pingelangu USB-kaablitel, mistõttu seadmetele jääb tegelikult kasutada vähem kui 5 V [13]. Käesoleva töö raames projekteeritavate ruumiseadmete voolutarbe määrab suuresti mikrokontroller, mis ülal nimetatud andmete põhjal tarbib mõnisada mA ja ei tohiks seetõttu süsteemi toimimist häirivat suurt pingelangu põhjustada. Niisiis võiks pidada HUZZAH32-le sobivaks elektritoiteallikaks suvalist standardset ja kvaliteetset USB-liidesega 5V/1A adapterit. Nutiseadmete ajastul jääb majapidamistes sageli selliseid adaptereid üle. Arenduse käigus ehitatud maketil tehtud mõõtmised näitasid, et HUZZAH32 moodulit arvuti klaviatuuri USB-pessa ühendatud juhtme kaudu toites rakendus moodulile 5,07 V või 5,03 V pinge vastavalt kui ei toimunud või toimus WiFi andmeedastus. Samad näitajad Apple'i A1400 adapterit kasutades olid 5,13 ja 5,12 V, ehk sisuliselt mõõtevea piirides. Kummagi toiteallika kasutamisel toimis süsteem korrektselt, mis toetab eelnimetatud oletust suvalise standardse adapteri sobivuse kohta.



Joonis 5. Apple A1400 toiteadapter.

Ruumiseadme teine põhikomponent, MAX9814 mikrofonimoodul vajab +2,7...5,5 V alalispinget 3 mA voolutarbe juures [7] ja sellisena sobib hästi ühendamiseks HUZZAH32 3,3 V pingeregulaatori väljundisse.

Keskseade ja andmehõiveseadmed varustatakse niisiis individuaalsete autonoomsete elektrivõrgust toidetavate toiteplokkidega (adapteritega).

## 4 Süsteemi ülesehitus

Projekteeritud andmehõivesüsteemi plokkskeemi esitab joonis 6. Süsteem koosneb keskseadmest U10 ning kolmest ruumiseadmest. Ruumiseadmete akustiliste andurite plokkide U1, U4 ja U7 analoogväljundid ühendatakse vastavate juht- ja andmesideplokkidega U2, U5 ja U8, mis edastavad digitaliseeritud helisignaali keskseadmele. Keskseadme ülesandeks on tuvastada ruumiseadmetelt saabunud digitaliseeritud helist häälkäsklused. Seadmeid toidetakse elektrivõrgust individuaalsete toiteadapteritega. Andmeside toimub üle TCP-sokli WiFi võrgu vahendusel. Kuigi ülesandepüstitus seda ei nõua, täidab projekti lõppversioonis WiFi ruuteri ülesannet keskseade U10 ise, seega on süsteem sõltumatu ning esmaseks opereerimiseks ei vaja väliste seadmete tuge. Keskseadmele WiFi ruuteri rolli andmine on autori idee, kuid selle realiseerimiseks kasutati juhiseid [14], [15].

### 4.1 Keskseade

Keskseade loob käivitamisel TCP-sokli liitumisvõimalusega kuni 5 kliendile, millest käesoleva projekti raames kasutatakse 3, ja jääb ootama ühendustaotlusi ruumiseadmetelt [16]. Kui ruumiseade võtab ühendust, luuakse selle teenindamiseks kaks lõime – vastuvõtu- ja saatislõim – ning üks järjekord, kuhu lisada ruumiseadmele edastatavad vastused ja korraldused (joonis 7). Vahetult lõimede loomise järel saadetakse ruumiseadmele VADi tundlikkuse seadistamise korraldus. Igale ruumiseadmele eraldatakse ka individuaalne kõnetuvastuse allsüsteemi objekt.

Vastuvõtulõim edastab ruumiseadmelt saabunud digitaliseeritud helisignaali kõnetuvastuse allsüsteemile. Kõnetuvastuse allsüsteem lisab tuvastuse oleku muutumisel (lähemalt vt jaotis 4.1.1) saatmisjärjekorda vastava teate. Saatislõime ülesandeks on järjekorda lisatud andmed ruumiseadmele edastada.

Ruumiseadmele edastatava info hulka kuulub:

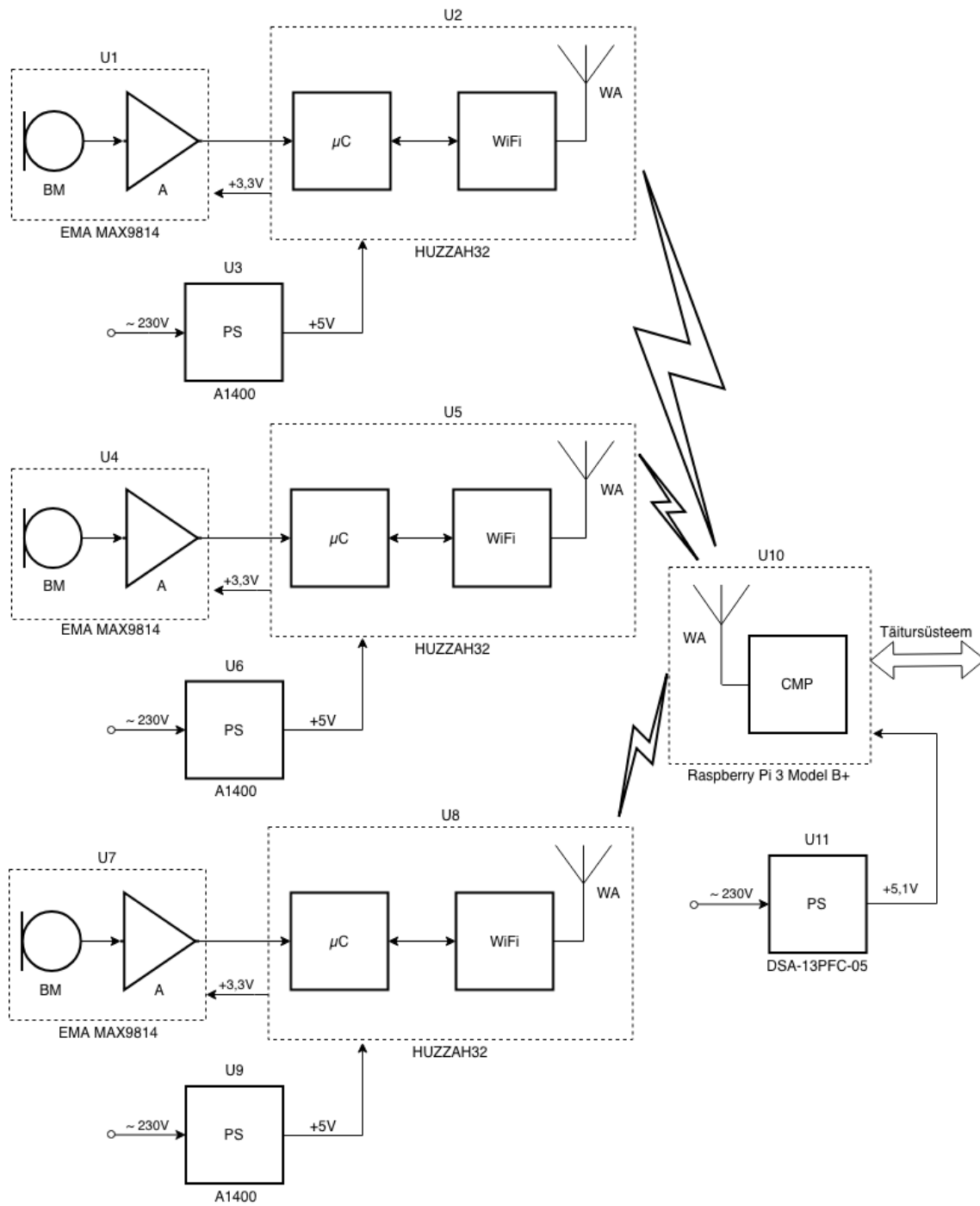
a) kas tuvastati võtmesõna ja oodatakse käsku,



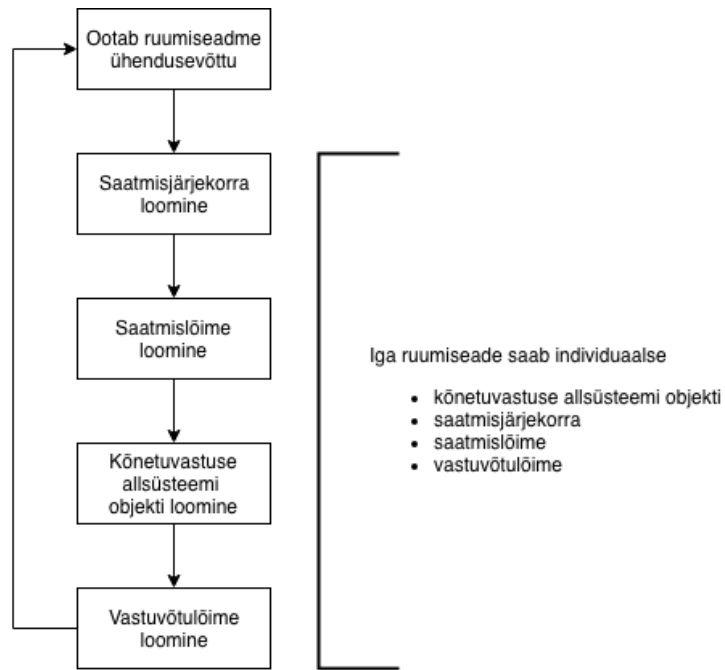
- b) tuvastati käsk,
- c) mõistliku aja jooksul ühtki käsku ära ei tuntud ning oodatakse jälle võtmesõna.

Kõnetuvastuse allsüsteemi vastuseid kasutab ruumiseade visuaalseks indikatsiooniks LED-indikaatori vahendusel ja ka mõningate andmeedastusotsuste tegemiseks (vt jaotis 4.2.4).

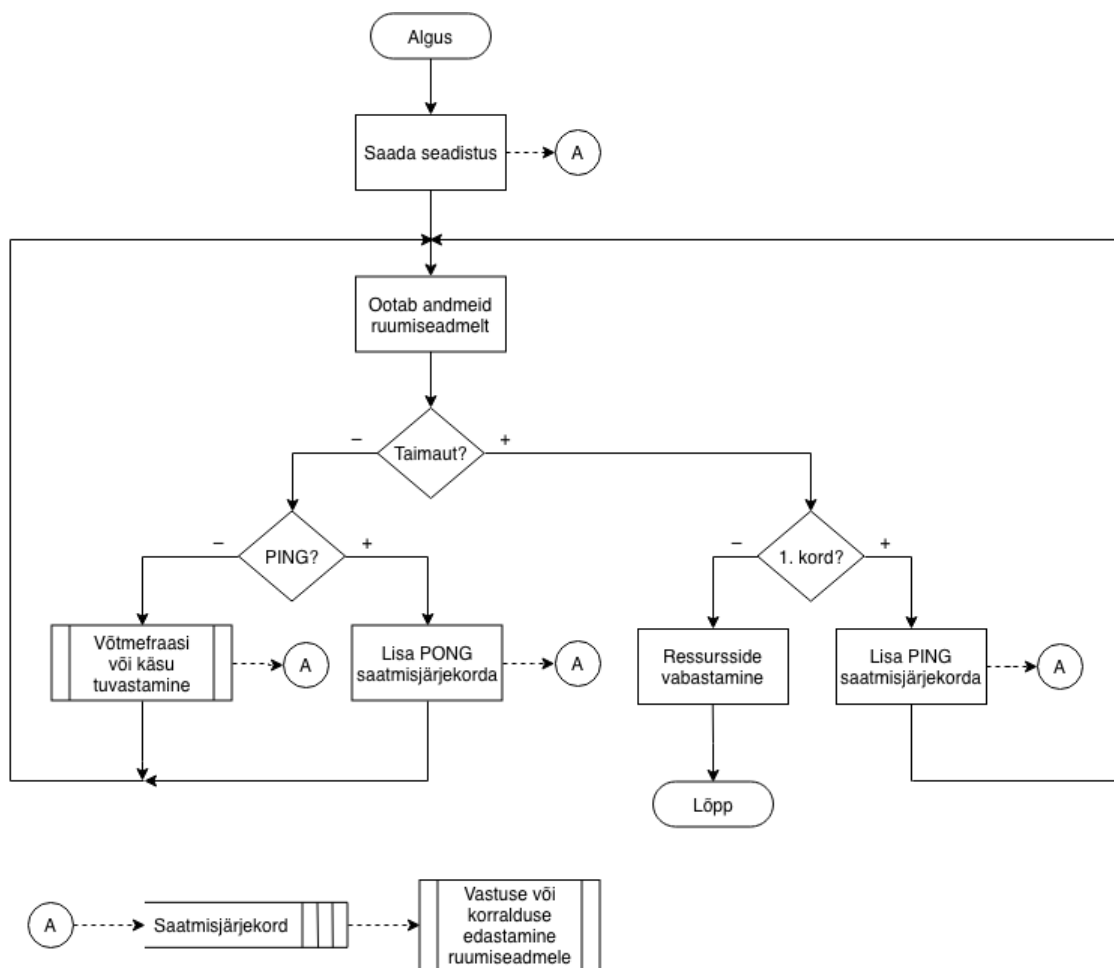
Keskseade hoiab konkreetse ruumiseadme teeninduseks eraldatud ressursse seni, kuni ruumiseade ise on küllaldaselt aktiivne. Kui ruumiseadmelt andmeid oodates tekib sokli taimaut, jätab keskseade selle meelde ning saadab spetsiaalse PING signaali. PING-signaali eeldatakse ruumiseade reageerivat vastava PONG signaaliga. Kui PING-signaali vastust oodates tekib taas sokli taimaut, loeb keskseade ruumiseadme mittefunktsioneerivaks ning vabastab sellele eraldatud ressursid. Ühe ruumiseadme teenindamist kujutab joonis 8. Kesk- ja ruumiseadme vahelise kommunikatsiooni-protokolli koodid on toodud tabelis 1 (tähistused: K – keskseade, L – ruumiseade).



Joonis 6. Andmehõivesüsteemi plokkskeem.



Joonis 7. Ressursside eraldamine ruumiseadmete teenindamiseks.



Joonis 8. Ühe ruumiseadme teenindamine keskseadmes.

Tabel 1. Kommunikatsioonisignaaside koodid.

Kood	Tähendus	Suund	Kommentaar
0	Jõudeolek, võtmefraasi ootel.	K → L	Praktikas ei kasutata. Ruumiseade järeldab selle oleku signaalidest 2 ja 3.
1	Aktiivne, käsu ootel.	K → L	Kõnetuvastuse allsüsteem tuvastas võtmefraasi ja on valmis käsku tuvastama.
2	Käsk tuvastatud.	K → L	Kõnetuvastuse allsüsteem tuvastas käsu.
3	Viga, taimaut.	K → L	Kõnetuvastuse allsüsteem ei tuvastanud etteantud aja jooksul käsku ja läks tagasi võtmefraasi ootamise režiimi.
4	PING	K ← L K → L	Ruumiseade saadab „vaikuse olukorras“ seda signaali välja perioodiliselt. Keskseade saadab signaali välja erijuhul, kui ruumiseadmelt pole pikema aja jooksul andmeid saabunud.
5	Vähendada VAD tundlikkust.	K → L	Praktikas ei kasutata.
6	Suurendada VAD tundlikkust.	K → L	Praktikas ei kasutata.
7	VAD tundlikkuse seadmine.	K → L	Saadetakse kohe, kui ruumiseade on keskseadmega ühendust võtnud.
8	PONG	K → L K ← L	Saadetakse vastusena PING signaalile 4.

#### 4.1.1 Kõnetuvastuse allsüsteem

Kõnetuvastuse allsüsteem oli suures osas juba varem välja arendatud [3], kuid vajab käesoleva projekti jaoks täiendamist. Siinkohal väike ülevaade selle ülesehitusest ja toimimisest, et hõlbustada töö järgnevate osade mõistmist.

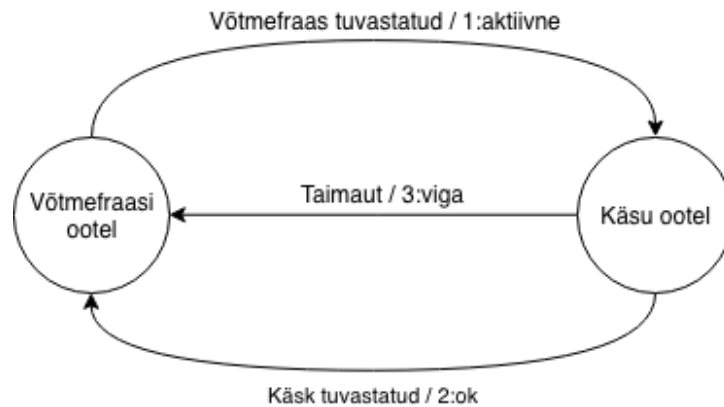
Kõnetuvastussüsteemi põhikomponent on spetsiaalselt mobiilsete seadmete tarvis loodud vabavaraline kõnetuvastustarkvara PocketSphinx [17] koos sellele sobiva eesti keele akustilise mudeliga [5]. PocketSphinxile on võimalik edastada helisignaali n-ö pidevas režiimis. Pidev režiim tähendab, et tarkvarale antakse perioodiliselt töötlemiseks lühikesi helisignaale. Iga töödeldud signaali korral saab PocketSphinxilt teada, kas seda eelmistega kokku pannes formeerus mõni tuvastatav sõna/fraas või mitte. PocketSphinx on kõnelejast sõltumatu süsteem, ehk selle funktsioneerimine ei eelda konkreetse isiku hääletämbri tundmaõppimist.

Häälkäskluste tuvastamiseks saab PocketSphinxi rakendada n-ö grammatika režiimis, milles tarkvarale antakse ette käskude grammatika formaliseeritud kirjeldus. Grammatika kirjelduseks on kasutusel spetsiaalne formaat JSGF [5], [18]. Sellise töörežiimi eripäraks on, et tarkvara analüüsib sisendandmeid ainult läbi grammatikakirjeldusega määratud kitsaste valikuvõimaluste. Seetõttu võib tarkvara tuvastada käskluse ka olukorras, kus seda tegelikult ei öeldud [3]. Probleemi aitab pehmedada võtmesõna või -fraasi rakendamine, sarnaselt „Ok Google“ või „Hey Siri“ kasutamisele Androidi või iOS mobiilsetes seadmetes, kus süsteem hakkab alles pärast nimetatud fraasi tuvastamist tegelikku korraldust ootama, ning teeb seda piiratud aja jooksul. Varasemas projektis [ibid] realiseeriti võtmefraasi tuvastus sama grammatikakirjelduse kaudu, millega ka käsustik. Seega PocketSphinx otsis endiselt vasteid grammatikakirjelduse piiratud valikute hulgast, lihtsalt see kirjeldus sisaldas lisaks käsustikule ka võtmefraasi. Võtmefraasi ja käskude eristamine tuli realiseerida projekti käigus arendatud tarkvaras.

Käesolevas töös muudeti varasemat lahendust selliselt, et kasutada ära PocketSphinxi spetsiaalne võtmefraasi tuvastuse töörežiim. Võtmefraasi tuvastamise režiimis ei otsi PocketSphinx iga hinna eest vasteid grammatikakirjeldusega piiratud võimaluste seast. Selle asemel arvutatakse hinnang sisendsignaali akustilisele sarnasusele võtmefraaside loetus nimetatud fraasidega. Igale võtmefraasile saab kehtestada numbrilise läve, mille ületamine on nõutav, et fraasi tuvastatuks lugeda. See võimaldab kohandada võtmefraaside äratundmise tõenäosust vastavalt vajadusele. Näiteks võib kriitilisemaid seadmeid häälkäsklustega juhtides määrata tavalisest erineva võtmefraasi ja kehtestada sellele rangema tuvastusläve. Kõnetuvastussüsteemi kui kahe olekuga automaati kirjeldab joonis 9. Väljundkoodid („1“, „2“ ja „3“) kuuluvad edastamisele vastava olukorra tinginud ruumiseadmele.

Töö kirjutamise hetkel kasutatud käsustiku JSGF grammatikakirjeldus on toodud joonisel 10. Sellise kirjeldusega tuntakse ära niisuguseid käske, nagu „lülita valgus sisse“, „pane televiisor kinni“, „tõmba kardinad ette“ jne, aga ka nende lühendatud kuju „valgus sisse“, „teler kinni“, „kardinad ette“. Praktikas võib sedavõrd paindlik kirjeldus ka väärtuvastusi põhjustada, sest televiisori saab lülitada nii „käima“ kui „välja“, mida PocketSphinx alati korrektselt ei erista. Samuti esines proovimise käigus väärtuvastusi „kardinad ette“ ja „kardinad eest“ vahel. Praktilise kasutamise planeerimiseks on niisuguste detailide avaldumine hea, mistõttu jäeti probleemsed sõnad siin töös

grammatikakirjeldusse sisse. Sõnade väärtuvastused muidugi vähendavad hääljuhtimisega süsteemi kasutusväärtust. Samas ei tekita need märkimisväärseid probleeme seni, kuni tegu on mittekriitiliste seadmete lülitusliku juhtimisega. Kui süsteem saab lülituskäsu suunast valesti aru, siis lihtsalt midagi ei muutu, sisselülitatud seade jääb sisselülitatuks ja vastupidi, ning käsku saab alati korrata.



Joonis 9. Kõnetuvastuse allsüsteem lihtsa automaadina.

```

#JSGF V1.0;
grammar commands;

<act_kardinad_1> = (tõmba | pane) *;
<act_kardinad_0> = tõmba *;
<obj_kardinad> = (kardin | kardinad);
<kardinad_1> = <act_kardinad_1> <obj_kardinad> ette;
<kardinad_0> = <act_kardinad_0> <obj_kardinad> eest (ära *);

<act_switching> = (pane | lülita) *;
<obj_teler> = televiisor | telekas | teler;
<obj_valgus> = valgus | valgustus | tuli | lamp;

<act_teler_1> = sisse | tööle | käima | mängima;
<act_teler_0> = välja | kinni;
<teler_1> = <act_switching> <obj_teler> <act_teler_1>;
<teler_0> = <act_switching> <obj_teler> <act_teler_0>;

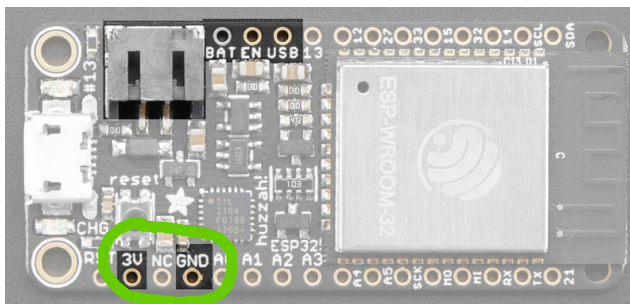
<act_valgus_1> = sisse | põlema;
<act_valgus_0> = välja;
<valgus_1> = <act_switching> <obj_valgus> <act_valgus_1>;
<valgus_0> = (<act_switching> <obj_valgus> <act_valgus_0>) |
  (kustuta <obj_valgus>);

public <commands> = <kardinad_1> | <kardinad_0> | <teler_1> | <teler_0> |
  <valgus_1> | <valgus_0>;
  
```

Joonis 10. Käsustiku JSGF grammatikakirjeldus.

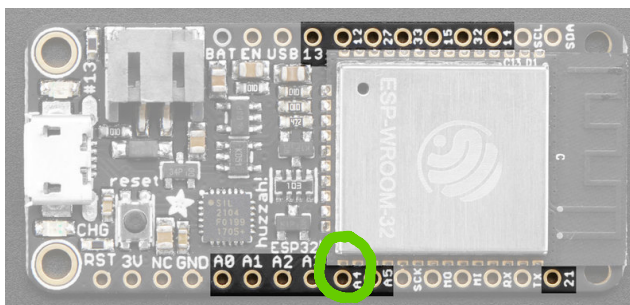
## 4.2 Ruumiseadmed

Nagu kirjeldatud peatükis 3, sobib mikrofonimooduli EMA MAX9814 toiteks kasutada mikrokontrolleri ESP32 arendusplaadi HUZDAH32 pingeregulaatori +3,3 V väljundit. Vastav väljundviik on joonisel 11 tähistatud 3V ja maandus GND. (Foto: [8].)



Joonis 11. HUZDAH32 elektritoitega seotud väljundviigud.

Mikrokontrolleri ESP32 kahest analoog-digitaalmuundurist ADC1 ja ADC2 kasutab WiFi draiver ise ADC2 [9], mistõttu tuli mikrofonisignaali digitaliseerimiseks valida ADC1. HUZDAH32 analoogsisendeid kujutab joonis 12, töös võeti neist kasutusele sisend A4. (Foto: [8].)



Joonis 12. HUZDAH32 analoogsisendid.

### 4.2.1 Pingete ühildamine

Mikrofonimooduli EMA MAX9814 väljundpinge diapsoon on  $1,25 \pm 1V$  [7] ehk 0,25...2,25 volti. ESP32 analoog-digitaalmuunduri väljundskaala maksimum aga vastab 1,1 voldile [9]. 1,1 voldist suuremate pingete muundamiseks saab ADMi sisendile programselt rakendada kas 2,5 dB, 6 dB või 11 dB sisendsignaali atenuaatori. 6 dB atenuaator vähendab muundurile edastatavat pinget ligikaudu 2 korda. See suurendab realselt kasutatava sisenddiapsooni 2,2 voldini [ibid]. Vastavalt kujuneb EMA MAX9814 väljundpinge diapsooniks pärast 6 dB atenuaatori läbimist 0,125...1,125 V. EMA MAX9814 väljundpinge diapsoon paikneb ADMi sisendpinge diapsooni suhtes

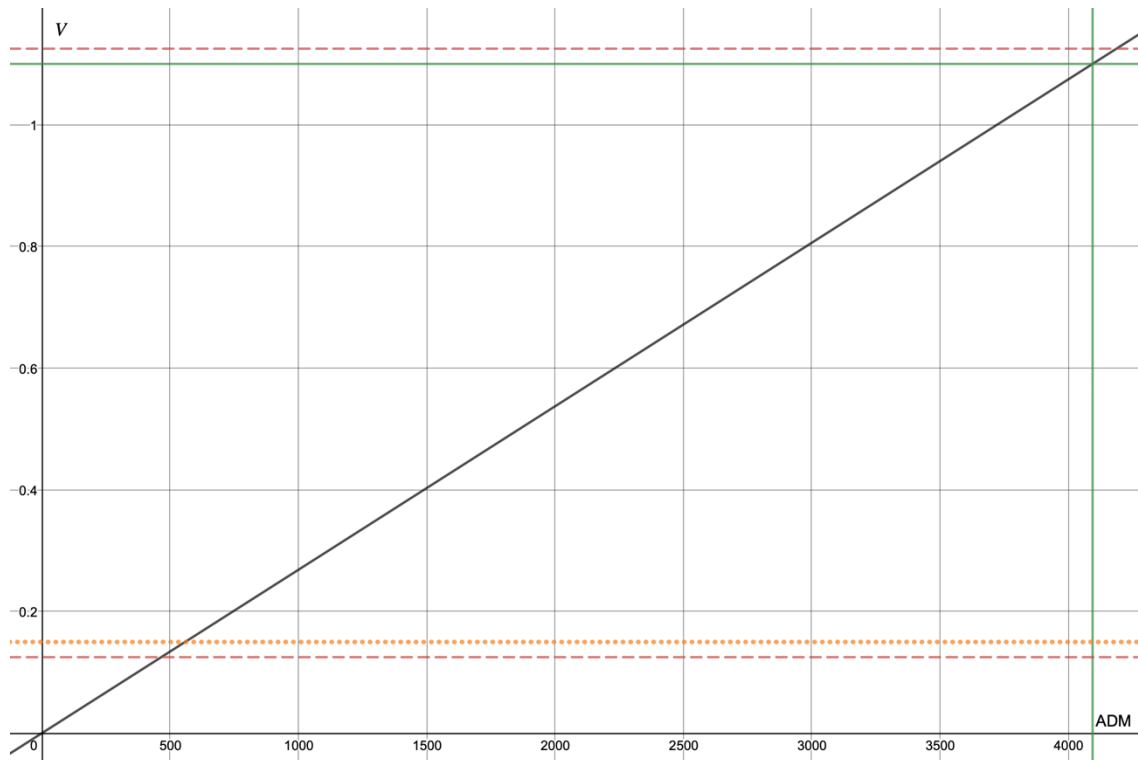
ebasümmeetriliselt, nagu näitab joonis 13. Joonise horisontaalteljel on esitatud 12-bitise ADMi täisskaala ning vertikaalteljel vastav sisendpinge, kui ADMi maksimumpingeks on 1,1 V. Rohelise pidevjoonega on kujutatud sellise ADMi maksimaalsed võimalikud väärtused kummaski dimensioonis, punane katkendjoon esitab pingevahemikku 0,125...1,125 V. Näeme, et EMA MAX9814 väljundpinge ei jõua kunagi väärtuseni 0, tegelikult kasutatav diapasoone jääb altpoolt piiratuks pingega 0,125 V. Üldalt piirab ulatust aga endiselt ADMi maksimaalne 1,1 V, mistõttu jääb osa EMA MAX9814 väljundpinge diapasoone kasutamata. Helisignaali oma olemuselt esitab mingi keskmise väärtuse ümber võnkuvat suurust. Niisuguse signaali moonutatusteta esitusdiapasoone määrab lähim piir, mis keskmisest väärtusest ühes või teises suunas liikudes vastu tuleb. EMA MAX9814 väljundpinge keskpunkt 1,25 V teiseb pärast 6 dB atenuerimist pingeks 0,625 V (vahemiku 0,125...1,125 V keskpunkt). Et ADMi sisendpinge maksimum 1,1 V on keskpunktile lähemal kui EMA MAX9814 väljundpinge miinimum 0,125 V, siis jääb sümmeetriliselt võnkuvat signaali moonutamata esitamiseks alles  $0,625 \pm (1,1 - 0,625) = 0,625 \pm 0,475$  V ehk 0,15...1,1 V. ADMi sisendpinge praktiline alampiir 0,15V on joonisel 13 kujutatud oranži punktiirjoonega.

Vaadeldava 12-bitise ADMi väärtuste skaala 0...4095 vastab pingetele 0...1,1 V, seega jäävad diapasoone 0,15...1,1 V korral kasutamata esimesed 558 väärtust:

$$\frac{0,15 - 0}{1,1 - 0} \cdot 4095 \approx 558$$

Analoog-digitaalmuunduse tegelikuks täpsuseks tuleb sel juhul  $4096 - 558 = 3538$  erinevat väärtust. Bittide arvuna väljendades on täpsuseks  $\log_2 3538 \approx 11,79$  bitti. Võrrelduna ADMi 12-bitise täpsusega ei ole täpsuskadu suur. Alternatiivne ja täpne lahendus EMA MAX9814 väljundi ning ADMi sisendi omavaheliseks sobitamiseks eeldaks minimaalselt nende galvaanilist lahtisidestamist ning pingejaguri kasutamist, ehk vähemalt kolme lisaelementi. Toodud arvutuste ja hilisemate praktiliste testide alusel ei peetud seda otstarbekaks. ADMi töö korraldamise tarkvaralist osa kirjeldab jaotis 4.2.4.

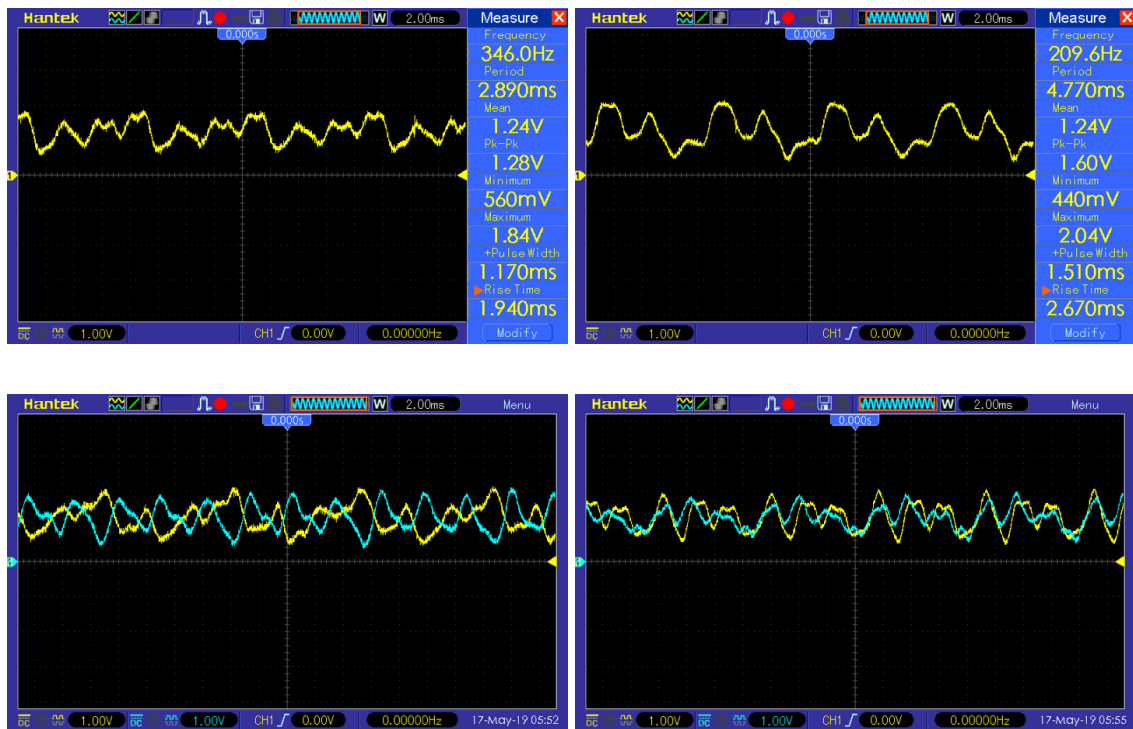




Joonis 13. ADMi sisendpinge ning EMA MAX9814 väljundpinge diapasooneid.

Joonis 14 kujutab EMA MAX9814 väljundpinget, registreerituna hääliku „u“ hääldamisel mikrofonist eri kaugusel ja helivaljusel. Üleval vasakul: kasutades normaalset hääletugevust (autori esituses) mikrofonist 1 meetri kaugusel ja mikrofoni pikiteljega umbes 45° nurga all. Üleval paremal: oluliselt valjema hääletugevusega, 30 cm kaugusel peaaegu otse mikrofoni suunatuna. Tuleb rõhutada, et need tulemused pole omavahel rangelt võrreldavad, sest katsed tehti eri ajal ning „normaalne“ ja „valjem“ hääletugevus pole objektiivselt ühesed ega korratavad suurused. Siiski väärub märkimist, et kauguse enam kui kolmekordne vähendamine koos hääletugevuse märgatava tõstmisega avaldus mikrofonimooduli väljundsignaalis kõigest  $20 \log \frac{1,60 \text{ V}}{1,28 \text{ V}} \approx 1,9 \text{ dB}$  erinevusena. Seega võimenduse automaatne reguleerimine toimus tõhusalt. Oluline on seejuures, et ka valjemal ja ekstreemsemal juhul – millist tegelikkuses tõenäoliselt ette ei tule – ei küündinud EMA MAX9814 väljundpinge spetsifitseeritud maksimumini 2,25 V, vaid maksimaalseks väärtuseks saadi nende testide puhul 2,04 V. Niisiis võib EMA MAX9814 väljundpinge lugeda pärast 6 dB atenuaatori rakendamist sobituvaks mikrokontrolleri ESP32 ADMi sisenddiapasoonega 0...1,1 V vähemalt nendes oludes, milles kasutamiseks projekteeritav süsteem mõeldud on.

Täiendavalt kujutavad joonise 14 kaks alumist pingegraafikut mõõtmisi, mis on tehtud kahe lokaalmooduliga samaaegselt. Häält tekitati kollasele joonele vastavast lokaalmoodulist 90 cm ja sinisega märgitust 180 cm kaugusel (nurgad hinnanguliselt vastavalt 0° ja 60°). Alumine vasakpoolne graafik vastab normaalsele ja parempoolne kõrgendatud hääletugevusele.

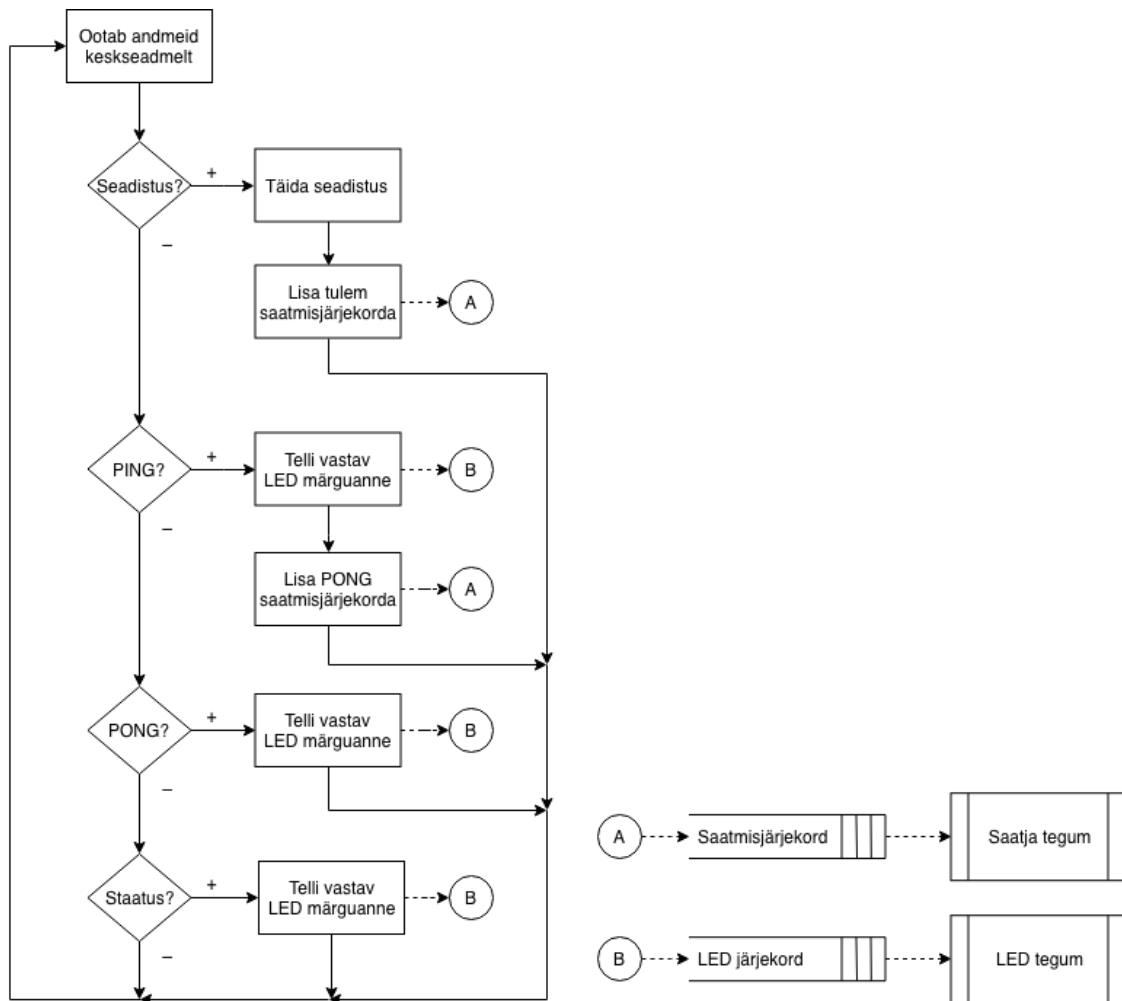


Joonis 14. Mikrofonimooduli väljundpinge.

#### 4.2.2 Andmete vastuvõtt keskseadmest

Ruumiseadme iga põhifunktsiooni viib läbi vastav FreeRTOSi tegum. Vastuvõtutegum ootab TCP-soklist andmeid – keskseadme saadetud vastuseid või korraldusi. Andmete saabudes otsustatakse vastuvõetud koodi alusel, kas ja millist märguannet LED-indikaatoril näidata. Märguande kirjeldus lisatakse LED-järjekorda, millest LED-indikaatori juhtimise tegum järgemööda kirjeldused välja loeb. Normaaljuhul on koodideks keskseadme vastused ruumiseadme saadetud digitaalsele helisignaale või PING-signaale, erijuhul aga VADi tundlikkuse seadmise korraldus. PING-signaali saabumisel algatab vastuvõtutegum lisaks LED-indikaatori märguandele ka PONG-vastuse, lisades selle saatmisjärjekorda. VADi tundlikkuse seadekorralduse puhul muudetakse VADi tundlikkust vastavalt koodiga saadetud parameetri väärtusele 0...4.

Andmete vastuvõttu ruumiseadmes kujutab joonis 15. Andmete saatmist ja LED-indikaatori juhtimist käsitletakse allpool vastavalt jaotistes 4.2.3 ja 4.2.5.

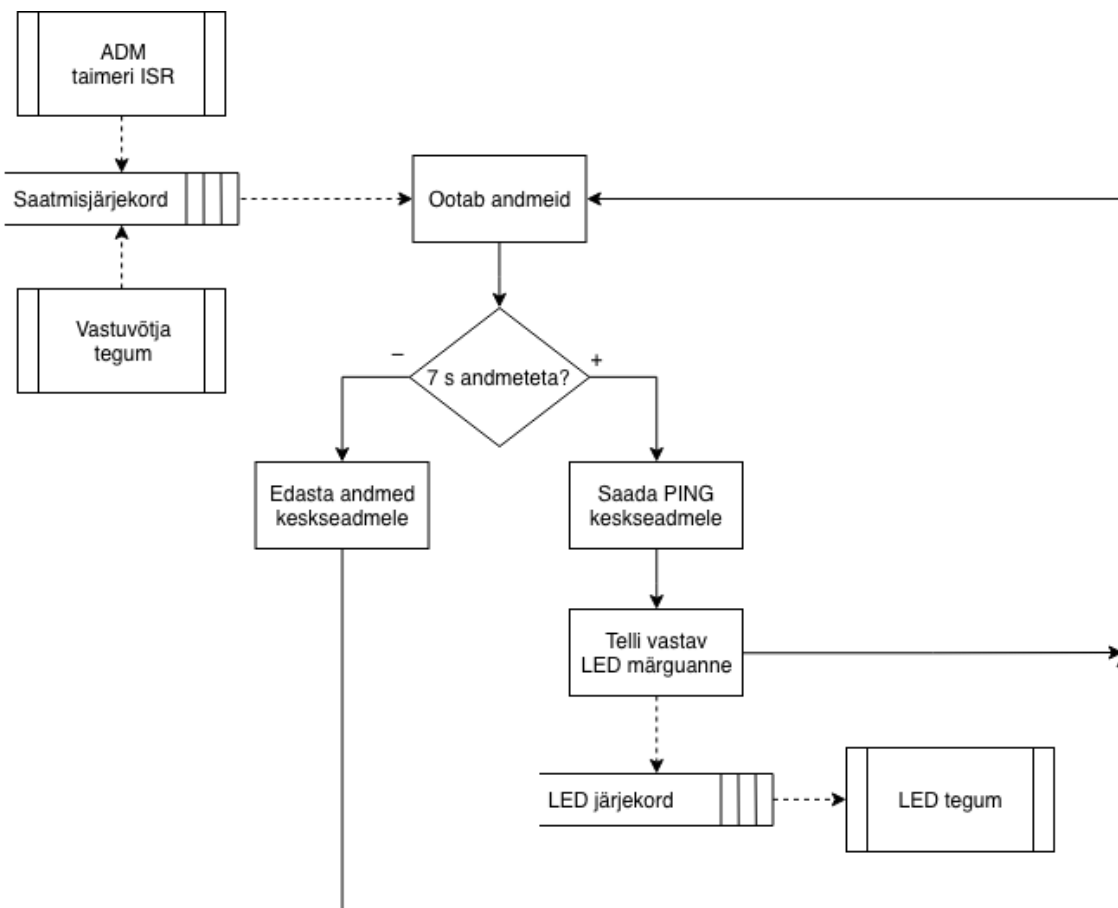


Joonis 15. Keskseadme saadetud vastuste või korralduste vastuvõtt ruumiseadmes.

#### 4.2.3 Andmete saatmine keskseadmele

Ruumiseadme põhiülesanne on akustilise info vahendamine keskseadmele. Seetõttu moodustab valdava osa ruumiseadme poolt saadetavatest andmetest lihtsalt digitaliseeritud helisignaali ADM-ilt. Saatmise tegum võtab saatmisjärjekorrast teiste tegumite poolt sinna lisatud andmed ja edastab need TCP-sokli kaudu keskseadmele. Kui ükski tegum pole teatud aja jooksul (praegu on selleks valitud 7 s – hea pikkus kasutamiseks süsteemi arendamise käigus, kuid praktilises kasutuses võib aeg oluliselt pikem olla) midagi saatmisjärjekorda lisanud, saadab saatmisloim keskseadmele iseseisvalt PING-signaali, hoides niiviisi sokliühenduse aktiivsena. PING-signaaliga kaasneb ka LED-indikaatori vilgutamine LED-tegumi vahendusel.

Andmete edastust keskseadmele kirjeldab joonis 16. Andmete vastuvõtu loogikat on selgitatud jaotises 4.2.2 ülal, ADMi taimerikatkestuse teenindamist ja LED-indikaatori juhtimist käsitletakse vastavalt jaotistes 4.2.4 ja 4.2.5.



Joonis 16. Andmeedastus ruumiseadmelt keskseadmele.

#### 4.2.4 ADM ja VAD

Kõnetuvastuse õnnestumiseks on oluline helisignaali kvaliteet. Digitaalne signaal vastab analoogsignaali seda täpsemalt, mida täpsem on muundamisel rakendatav eraldusvõime (mida suurem on kodeerimisel kasutatavate bittide arv) ja mida sagedamini analoogsignaali hetkväärtusi edastatakse (mida suurem on kvantimissagedus). Mikrokontrolleri ESP32 12-bitise analoog-digitaalmuunduri sisenddiapasoonil ja täpsusel peatuti pikemalt jaotises 4.2.1. Kvantimissageduse määramisel on otstarbekas lähtuda digitaliseeritava suuruse muutumise kiirusest. Nyquisti teoreemi järgi on sagedusega  $f$  muutuva signaali digitaalseks esituseks vajalik kvantimissagedus vähemalt  $2f$ . Teisalt nõuab kõrgem kvantimissagedus aga rohkem ressursse. Seega pole otstarbekas mõõta sagedusega  $\leq f$  muutuvat signaali oluliselt

sagedamini kui 2f. Inimkõne sagedusspekter kontsentreerub tüüpiliselt sagedustele alla 10 kHz, mistõttu valitakse kõnetuvastusel kvantimissageduseks kuni 16 kHz, ressursikriitilisemates süsteemides ka vähem [19]. Käesolevas töös võeti kvantimissageduseks 16 kHz. EMA MAX9814 väljundsignaali perioodiliseks digitaliseerimiseks seadistati mikrokontrolleril sellise sagedusega esilekutsutav taimerikatkestus. Taimerikatkestuse ja ADMi seadistamisel juhinduti ESP-IDF standardinstallatsioonis sisalduvatest näidetest.

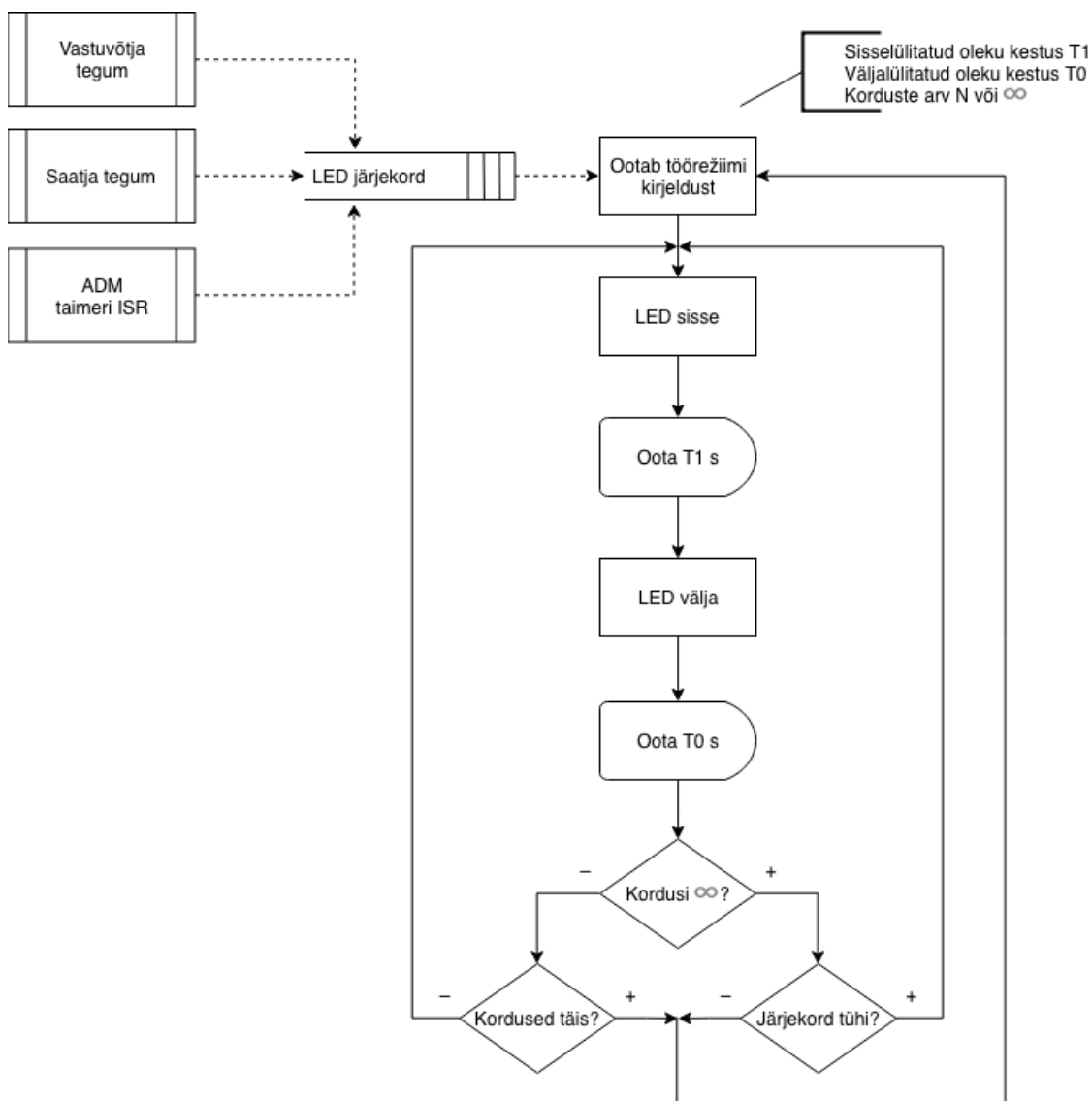
Et mitte edastada andmeid ruumiseadmelt keskseadmele olukorras, kus keegi midagi ei räägi, võeti kasutusele ESP-ADF VAD süsteem [20]. ESP-ADF VAD initsialiseerimisel tuleb näidata VADi soovitud tundlikkus täisarvulisest vahemikust 0...4. Vaikimisi valib ruumiseade tundlikkuseks keskmise väärtuse 2. Ette on nähtud ka võimalus tundlikkusparameetri edastamiseks keskseadme poolt (vt 4.1). Tuleb nimetada, et kuigi VAD viitab just inimkõne olemasolu otsimisele signaalis, osutus ESP-ADF vastav realisatsioon reageerivaks igasugusele helisignaalile, ka näiteks laualekoputusele.

Kasutusel on kaks andmepuhvrit vaheldumisi, kumbki mahutab 30 ms kestusega helisignaali. Samal ajal kui ADMi taimerikatkestuse programm täidab üht andmepuhvrit, saab teise puhvri sisu edastada keskseadmele, kui see on vajalik. Vajalikkuse määrab VAD-funktsioon kombinatsioonis keskseadmelt saabuvate vastustega (vt joonis 17). Kui seni on kestnud „vaikus“, siis käivitatakse puhvri igal täitumisel selle puhvri sisu kohta VAD-funktsioon. Kui VAD-funktsioon tuvastab heli olemasolu, väärtustatakse loendur nVAD, et järgnevad 25 puhvritäit helisignaali keskseadmele saata. Loenduri nVAD nulli jõudes kontrollitakse heli olemasolu uuesti, kui just vahepeal pole keskseadmelt saabunud teadet võtmefraasi tuvastamisest. Viimasel juhul loobutakse VAD-funktsiooni kasutamisest ning andmepuhvri igal täitumisel edastatakse selle sisu alati keskseadmele, kuni keskseade teatab kas käsu tuvastamisest või taimaudist. Kuni keskseadmelt pole vastuseid saabunud, tellib ADMi taimer ISR ka LED-märkuandeid. Heli tuvastamisel VAD poolt lülitatakse LED-indikaator sisse. Pärast 25 puhvritäie andmete edastust otsustatakse VAD-funktsiooni järgmise väljakutse tulemi põhjal, kas jätta LED-indikaator sisselülitatuks või mitte.



#### 4.2.5 LED-indikaatori märguanded

LED-indikaatoriga antavaid märguandeid korraldab LED-tegum vastavalt LED-järjekorda pandud tellimustele. Tellimused sisaldavad infot selle kohta, kui kaua hoida LED-indikaatorit sisselülitatuna, kui kaua väljalülitatuna, ja kui mitu korda seda teha. Olemas on ka ümberlülitumiste lõputu kordamise võimalus. Lõputult korduva märguande katkestab järgmise tellimuse ilmumine LED-järjekorda. Fikseeritud korduste arvuga märguanded seevastu täidetakse alati lõpuni ehk korduste ettenähtud arvu saavutamiseni. LED-indikaatori juhtimise voodiagramm on toodud joonisel 18 ja märguannete loetelu tabelis 2.



Joonis 18. LED-indikaatori juhtimine.

Tabel 2. LED-indikaatori märguanded.

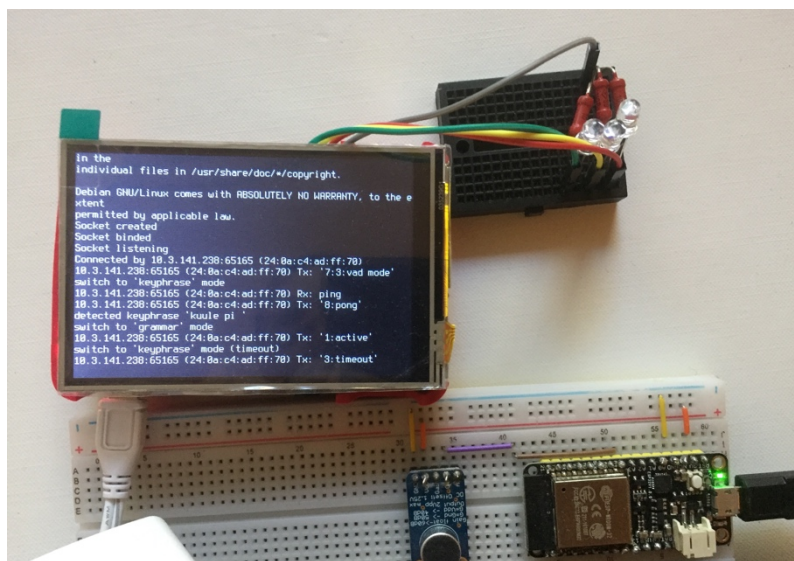
Sündmus	Allikas	LED režiim	Korduste arv
Vaikimisi		Väljas	Katkestamiseni
Käsu ootel	Vastuvõtu tegum, kood 1	Vilgub perioodiga 660 ms	Katkestamiseni
Käsk tuvastatud	Vastuvõtu tegum, kood 2	Sees 120 ms, väljas 120 ms	2
Taimaut, viga	Vastuvõtu tegum, kood 3	Sees 30 ms, väljas 30 ms	5
PING	Saatja tegum	Sees 30 ms, seejärel väljas	1
PONG	Vastuvõtu tegum, kood 8	Sees 30 ms, seejärel väljas	1
VAD = 1	ADM ISR	Sees	Katkestamiseni
VAD = 0	ADM ISR	Väljas	Katkestamiseni

### 4.3 Maketeerimine ja testid

Süsteemi välja töötades kasutati keskseadme toimimise monitooringuks Raspberry Pi-ga ühendatud LCD-tablood [21]. Potentsiaalset täitursüsteemi imiteeriti kolme valgusdiodiga, tüürituna Raspberry Pi GPIOde kaudu [3]. Samuti koostati kõigi kolme ruumiseadme maketid. Lõplik süsteem LCD-tablood ei vaja. Selle olemasolu võib osutada isegi piiravaks teguriks, sest tablo katab mikroarvutis mitmed väljundid, mida ei saa enam GPIO funktsioonis kasutada.

Joonisel 19 on demonstreeritud andmehõivesüsteemi töö kolme põhisündmust – võtmefraasi tuvastus ja järgnev käsu tuvastus või selle ebaõnnestumine. Ülemisel fotol: äsja ühenduse loonud ruumiseadmele on saadetud VADi tundlikkuse seadistuse korraldus. Ruumiseadme poolt edastatud helisignaalist on tuvastatud võtmefraas (seadme punane LED-indikaator vilgub) ning oodatakse käsku. Keskmisel fotol: ettenähtud aja jooksul käsku kas ei saanud või seda ei tundud ära, süsteem läks tagasi rahuolekusse (võtmefraasi ootel). Alumisel fotol: tuvastati järgmine võtmefraas ning seejärel käsk. Käsu täitmist imiteerib Raspberry Pi-ga ühendatud punase LED-indikaatori sisselülitumine.





Joonis 19. Süsteemi olekud makettidega testimisel.

## 5 Edasiarenduse võimalusi

Töös on põhiliselt käsitletud andmehõivet, pööramata suuremat tähelepanu täiturmehhanismidele. Mõistagi eeldab juhtimine ka täitursüsteemi kasutamist. Kõige otsesem võimalus täiturite juhtimiseks on kasutada GPIOd. Raspberry Pi-l on 26 [22] ja HUZZAH32-l 10 GPIOd [8], millest ühe kaudu juhiti töös LED-indikaatorit ning veel ühe hõivas analoogsisend. GPIO kaudu juhtimiseks on aga vajalik füüsiline ühendus juhitava seadme ning keskseadme või ruumiseadme vahel. Niisugune süsteem oleks kohmakas, raskesti standardiseeritav ning keeruline hallata.

Perspektiivsem on integreerida projekteeritud andmehõivesüsteem mõne standardse hooneautomaatika tehnoloogiaga. Raspberry Pi on selles osas küllaltki universaalne seade, sest seda toetavad mitmed IoT ja hooneautomaatika tarkvara raamistikud. Näiteks openHAB võimaldab kasutada mitmeid tehnoloogiaid, sh Bluetooth, Philips Hue, ZigBee, aga ka KNX [23], [24]. LON-tehnoloogia seadmeid toetab IzoT [23],

Ka täitursüsteemi määratlemata on võimalik tööd mitmeti edasi arendada. Näiteks saab optimeerida ruumiseadmete energiatarvet. Töös arendatud andmehõivesüsteemi võib lugeda sobivaks püsiva elektrivõrgutoite olemasolul, kuid ESP32 energiasäästurežiime [25], [26] rakendades ning süsteemi akutoitega täiendades saaks sama lahendust kasutada ka ruumides, kus elektrivõrk puudub.

Täiustada võib ruumiseadmete kasutajaliidest, lisades LED-indikaatori märguannetele ka helitagasiside. Nii ei eeldaks süsteemi kasutamine visuaalset kontakti.

Võimalik on arendada käsustiku loogikat, koostades iga ruumiseadme jaoks individuaalse grammatikakirjelduse. Hea oleks töötada välja tarkvara, millega GUI abil käsustikku muuta ning sellest automaatselt JSGF grammatikakirjeldused genereerida.

Ruumiseadmetel võiks kasutusele võtta tarkvara OTA uuendamise mehhanismi, et tarkvara jätkuval arendamisel tehtud muutusi oleks lihtsam seadmetesse sisse viia.

Kindlasti pole toodud loetelu lõplik.

## 6 Kokkuvõte

Bakalaureusetöös seati eesmärgiks projekteerida andmehõivesüsteem koos selle riistvarakomponentidele sobiva toitesüsteemiga olmeseadmete hääljuhtimiseks korteris või väiksemas eramus. Süsteem pidi tagama kontaktita andmeside kolme eri ruumides paikneva akustilise anduri ning keskseadme vahel. Juhtimissüsteemi täiturmehhanism püstitatud ülesandes käsitlemisele ei kuulunud.

Kõigepealt valiti ja täpsustati andmehõivesüsteemi riistvarakomponente. Keskseadmeks valiti Raspberry Pi 3 Model B+ ning akustilisteks anduriteks mikrofonimoodulid Adafruit AGC Electret Microphone Amplifier MAX9814. Andurite signaali digitaliseerimiseks ning kontaktita edastuseks üle WiFi valiti mikrokontrolleri ESP32 arendusplaat Adafruit HUZAZH32.

Järgmiseks uuriti valitud komponentide elektritoitevajadusi. Leiti, et tootja soovitab Raspberry Pi toiteks toiteplokki Stontronics DSA-13PFC-05 ning HUZAZH32 elektritoiteks saab kasutada standardseid elektrivõrgust toidetavaid USB liidesega 5 V toiteplokkke. EMA MAX9814 elektritoiteks osutus sobivaks HUZAZH32 3,3 V pingeregulaatori väljund.

Seejärel töötati välja süsteemi struktuur ja algoritmid ning uuriti EMA MAX9814 väljundpinge ning HUZAZH32 ADMi sisendpinge diapsoonide sobivust. Leiti, et pärast integreeritud atenuaatori rakendamist ADMi sisendisse antavale signaalile sobitub EMA MAX9814 väljundpinge diapsoon ADMi sisendpinge diapsooniga küllalt hästi, et puuduks vajadus täiendavate sobituskomponentide järele. Viidi läbi andmehõivesüsteemi testimine makettseadmetel.

Töö viimases osas vaadeldi lühidalt teemasid, mis ülesandepüstitusse ei mahtunud või mille põhjaliku käsitlemiseni töö käigus ei jõutud. Kirjeldati võimalusi projekteeritud andmehõivesüsteemi ühendamiseks täitursüsteemiga ning töö võimalikke edasiarendusi.

Autori hinnangul said bakalaureusetöös püstitatud ülesanded täidetud.

## Kasutatud kirjandus

- [1] D. Bharambe ja S. P. Kodgire. Speech Recognition System for Voice Controlled Devices. — *International Journal of Latest Trends in Engineering and Technology*, 2016, 6 (3).
- [2] S. Agrawal ja S. Krithiga. Implementation of Speech Recognition System on Raspberry Pi. — *International Journal of Innovative Research in Computer Science & Technology*, 2015, 3 (2).
- [3] J. Palgi ja T. Pajusaar. Eestikeelne häälujuhtimine Raspberry Pi baasil : IXX1420 Arvutid ja süsteemid kursusetöö. Tallinn, Tallinna Tehnikaülikool, 2019.
- [4] e-Teatmik: IT ja sidetehnika seletav sõnaraamat. [WWW] <http://vallaste.ee/> (15.05.2019)
- [5] T. Alumäe. Tutorial of Estonian speech recognition using PocketSphinx. [WWW] <https://github.com/alumae/et-pocketsphinx-tutorial> (13.05.2019)
- [6] Raspberry Pi 3 Model B+. — *Raspberry Pi*. [WWW] <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/> (13.05.2019)
- [7] Adafruit AGC Electret Microphone Amplifier - MAX9814. — *Adafruit Learning System*. [WWW] <https://learn.adafruit.com/adafruit-agc-electret-microphone-amplifier-max9814?view=all> (13.05.2019)
- [8] Adafruit HUZZAH32 - ESP32 Feather. — *Adafruit Learning System*. [WWW] <https://learn.adafruit.com/adafruit-huzzah32-esp32-feather?view=all> (13.05.2019)
- [9] Analog to Digital Converter. — *ESP-IDF Programming Guide v3.3-beta1-542-gbba89e151 documentation*. [WWW] <https://docs.espressif.com/projects/esp-idf/en/latest/api-reference/peripherals/adc.html> (09.03.2019)
- [10] Power Supply. — *Raspberry Pi Documentation*. [WWW] <https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md> (13.05.2019)
- [11] Raspberry Pi Universal Power Supply. — *Raspberry Pi*. [WWW] <https://www.raspberrypi.org/products/raspberry-pi-universal-power-supply/> (13.05.2019)
- [12] Power Supply for Raspberry Pi 3, 5V(5.1V) 2.5A, Official PSU, AU, Micro USB, Pi 2 – Modtronix. [WWW] <http://modtronix.com.au/product/pwr-rpi3-official/> (13.05.2019)
- [13] 5V 1A (1000mA) USB port power supply - UL Listed. [WWW] <https://www.adafruit.com/product/501> (13.05.2019)
- [14] Create a Wi-Fi hotspot in less than 10 minutes with Pi Raspberry! — *How To Raspberry Pi*, 14-06-2017. [WWW] <https://howtoraspberrypi.com/create-a-wi-fi-hotspot-in-less-than-10-minutes-with-pi-raspberry/> (08.05.2019)
- [15] B. Zimmerman. A simple, responsive web interface to control wifi and hostapd on the Raspberry Pi. 08-05-2019. [WWW] <https://github.com/billz/raspap-webgui> (08.05.2019)
- [16] Socket Programming HOWTO. — *Python 3.7.3 documentation*. [WWW] <https://docs.python.org/3/howto/sockets.html> (14.05.2019)
- [17] PocketSphinx speech recognition engine. 14-05-2019. [WWW] <https://github.com/cmusphinx/pocketsphinx> (14.05.2019)
- [18] JSpeech Grammar Format. [WWW] <https://www.w3.org/TR/2000/NOTE-jsgf-20000605/> (14.05.2019)
- [19] Inside Speech Recognition. [WWW] <https://www.tldp.org/HOWTO/Speech-Recognition-HOWTO/inside.html> (17.05.2019)
- [20] Espressif Audio Development Framework. [WWW] <https://github.com/espressif/esp-adf> (02.05.2019)

- [21] Puutetundlik ekraan 3.5'' Raspberry PI-le. — *Oomipood*. [WWW]  
[https://www.oomipood.ee/product/3\\_5lcdrpi\\_puutetundlik\\_ekraan\\_3\\_5\\_raspberry\\_pi\\_le](https://www.oomipood.ee/product/3_5lcdrpi_puutetundlik_ekraan_3_5_raspberry_pi_le)  
(16.01.2019)
- [22] GPIO. — *Raspberry Pi Documentation*. [WWW]  
<https://www.raspberrypi.org/documentation/usage/gpio/> (19.05.2019)
- [23] A. Rähni. ISS0090 Arukad hooned, praktikum 3. [WWW]  
[http://www.tud.ttu.ee/web/Andres.Rahni/Prx\\_X10p.pdf](http://www.tud.ttu.ee/web/Andres.Rahni/Prx_X10p.pdf)  
Tallinn, Tallinna Tehnikaülitool, 2019
- [24] openHAB. [WWW] <https://www.openhab.org/> (19.05.2019)
- [25] Insight Into ESP32 Sleep Modes & Their Power Consumption. — *Last Minute Engineers*,  
23-12-2018. [WWW] <https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/> (18.05.2019)
- [26] S. Marga. Patareitoitel WiFi andmesidega anduriplatvorm : bakalaureusetöö. Tallinn,  
Tallinna Tehnikaülikool, 2017.