

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKA TEADUSKOND

Arvutiteaduse Instituut

Informaatika eriala

Angelina Kozina

Helivaljuse mõõtja

Bakalaureusetöö (6 EAP)

Juhendaja: lektor Anne Villems

Kaasjuhendaja: spetsialist Taavi Duvin

TARTU 2014

Helivaljuse mõõtja

Lühikokkuvõte:

Antud bakalaureusetöö eesmärgiks on anda õpetajatele ning õpilastele põhjalik eestikeelne kasutusjuhend firma *Vernier* poolt toodetud helianduri kohta ning tutvustada anduri kasutamise võimalusi koos LEGO *Mindstorms* komplektiga. Töö on koostatud kolmes peatükis. Esimeses peatükis iseloomustatakse heli ja kirjeldatakse heli mõisteid andmaks lugejale ülevaate heli omadustest. Teises peatükis kirjeldatakse ja lahatakse *Vernier*'i helivaljuse mõõtjat ja tema töö põhimõtteid. Kolmandas peatükis on toodud helivaljuse mõõtjale mõeldud ülesanded koos kirjeldustega ja võimalike lahendustega.

Võtmesõnad:

Lego *Mindstorms* NXT, *Vernier*, helivaljuse mõõtja, robotika

Vernier Sound Level Meter

Abstract:

The aim of this bachelor thesis is to create a comprehensive user manual about the Vernier sound level meter in Estonian and to introduce various possibilities to use the sound sensor with LEGO Mindstorms NXT kit. This thesis consists of three chapters. The first chapter provides a brief overview of the concept of the sound and other related definitions. In the second chapter the author focused on the Vernier's sound level meter and compared it to the basic NXT sound sensor. The third chapter consists of five practical exercises for the sound level meter and the NXT set. There are two simple exercises to introduce NXT-G programming environment and three for collecting and analyze sound level meter output data.

Keywords:

Lego Mindstorms NXT, Vernier sound level meter, robotics

Sisukord

| | |
|-----------------------------------------------------------------------------------|----|
| Sissejuhatus | 4 |
| 1. Heli iseloomustus | 6 |
| 1.1 Heli (heli lained) | 6 |
| 1.2 Heli allikad | 7 |
| 1.3 Helivaljus | 8 |
| 1.4 Heli kõrgus | 10 |
| 1.6 Heli kiirus | 11 |
| 1.5 Heli peegeldus | 13 |
| 1.7 Infraheli | 13 |
| 1.8 Ultraheli | 15 |
| 1.9 Müra | 16 |
| 2. Vernier helivaljuse mõõtja | 18 |
| 2.1 Vernier' helivaljuse anduri kirjeldus | 18 |
| 2.2 Vernier'i helivaljuse mõõtja võrdlus standardkomplektiga | 20 |
| 2.3 Robootikakomplekti LEGO Mindstorms NXT üldine kirjeldus | 21 |
| 2.4 LEGO Mindstorms NXT-G tutvustus | 23 |
| 2.5 Vernier'i helivaljuse anduri ühendamine LEGO Mindstorms NXT komplektiga | 23 |
| 2.6 Helivaljuse anduri programmeerimine keeles NXT-G | 24 |
| 2.6.1 Vernier andurite ploki importimine | 24 |
| 2.6.2 Anduri testimine ja olulisemate funktsioonide tutvustamine | 25 |
| 2.6.3 Uute muutujate lisamine | 28 |
| 2.6.4 Failidega opereerimine | 29 |
| 3. Ülesanded | 31 |
| 3.1 Sissejuhatav ülesanne | 31 |
| 3.2 Ülesanne andmete analüüsi operatsiooniga | 34 |
| 3.3 Andmete kogumise ja analüüsimise ülesanne | 37 |
| 3.4 Maksimaalse väärtuse leidmine | 41 |
| 3.5 Keskmise leidmine | 43 |
| Kokkuvõte | 46 |
| Kasutatud kirjandus | 47 |
| Summary | 50 |
| Lisad | 51 |
| Lisa 1. Ülesannete lahenduste failid | 51 |
| Lisa 2. Litsents | 52 |

Sissejuhatus

Tänapäeva hariduselus üheks probleemiks on see, et kool ei soodusta mõtlemist. Koolis on jätkuvalt olulisel kohal mälule rajanev õpe, seetõttu jäävadki sellised oskused nagu kriitiline ja süsteemne mõtlemine, probleemide lahendamise oskused õpilastel välja arendamisel tahaplaanile. [1] Õpilasel, kellel teooria on pähe õpitud, puudub seoste tunnetamine, ta ei tea, kuidas oma teadmisi rakendada.

Enim märgatav on see probleem reaalainete puhul. Õpilastele jääb mulje, et ained nagu füüsika või keemia on väga teoreetilised ja seepärast igavad. Koolides on olemas abimaterjalid praktiliste ülesannete jaoks, mis tihti ei ole kaasaegsed, neid on vähe ja õpilased nendega väga katsetata ei saa. Selle olukorra parandamiseks on tulnud välja erinevate projektidega: Teadusbuss [2], Kooliroboti projekt [3] ja saatesari "Rakett 69". [4]

Kooliroboti projekt sai alguse 2007. aasta kevadel eesmärgiks edendada inseneriteadust Eesti koolides. [3] Projekti eestvedajateks on Hariduse Infotehnoloogia Sihtasutus (endine Tiigrihüppe Sihtasutus), Tartu Ülikool ning MTÜ Robotika. Peamisteks tegevusteks on LEGO NXT robotite tutvustamine, eestikeelsete juhendite loomine ning õpetajate koolitamine. Valminud õppevahendite abil saab õpilane NXT roboteid programmeerida ja nende abil reaalainete tundides katseid läbi proovida.

LEGO *Mindstorms* NXT komplekti kuuluvad: juhtplokk (*Brick*), kolm mootorit, ultraheliandur, heliandur, valgusandur, värviandur, kaks puuteandurit ning ühenduskaablid. Juhul, kui baaskomplekti kuuluvatest andurist jääb väheseks, on võimalik andureid kolmandatelt osapooltelt juurde hankida. NTX-ga ühilduvate andurite tootjaid on mitu - *Vernier*, *HiTechnic* ja *Mindsensors*. Nende firmade anduritega kaasaskäivad kasutusjuhendid on tihti ainult ingliskeelsed ja ei kirjelda kasutamise võimalusi täismahus.

Antud bakalaureusetöö eesmärgiks on anda õpetajatele ning õpilastele põhjalik eestikeelne kasutusjuhend firma *Vernier* poolt toodetud helianduri kohta ning tutvustada anduri kasutamise võimalusi koos LEGO *Mindstorms* komplektiga.

Töö koosneb kolmest osast. Esimeses käsitletakse heli mõistet ja põhilisi omadusi. Eesmärgiks on luua ülevaatlilik heli kirjeldus, mis sobib baasiks antud töö järgmiste peatükkide käsitlemiseks ning lisamaterjalina kooli programmi raames õpetatava juurde. Teises peatükis kirjeldatakse *Vernier*'i helivaljuse mõõtjat, tema tööpõhimõtteid. Antakse võrdlus LEGO *Mindstorms* standardse komplektiga ning tutvustatakse NXT programmeerimiskeskonda. Kolmandas peatükis on toodud helivaljuse mõõtjale mõeldud ülesanded koos kirjeldustega ja võimalikke lahendustega.

Antud bakalaureuse töös on lisaks välja toodud NXT-G võrdlused teiste programmeerimiskeeltega, eesmärgiga anda õpilastele täiendavat ülevaadet programmeerimisest ja tekitada nendes huvi programmeerimise vastu.

1. Heli iseloomustus

Meid ümbritsevad igasugused helid. Loomad häälitsevad, puulehel sahisevad, masinad mürisevad, muusika mängib. Igaüks on vähemalt korra mõelnud, mis on heli. Alguses arvati, et heliks võib pidada kõike, mida inimesed olid võimelised tajuma. Siiski, väga kiiresti aimati, et loomad tunnetavad helisid, mida inimese kõrv ei taju. Kuidas tekivad helid ja kuidas inimene neid kuuleb?

Edaspidi tutvustatakse heliga seotud mõisteid: heliallikad, heli peegeldus, heli kõrgus, kiirus ja otseloomulikult - helivaljus. Seda kõike on vaja, et paremini aru saada heli omadustest ja paremini orienteeruda järgnevates peatükkides.

1.1 Heli (heli lained)

Järgnev osa on koostatud kasutades allikaid [5] ja [6].

Kõike, mis inimene on harjunud heliks nimetama on tegelikult üks osa võnkumistest, mida inimese kõrv on suuteline tajuma. Heliks nimetatakse kõiki elastses keskkonnas (õhk, vesi, metall, kivimid jne) levivaid mehaanilisi võnkumisi. Kitsamas mõttes mõeldakse heli all võnkumisi, mida tajub inimese kõrv, see tähendab võnkumisi võnkesagedusega umbes 16 ... 20 000 Hz (**Herts** on perioodilise protsessi sageduse ühik. 1 herts on niisugune sagedus, mille korral 1 sekundi jooksul toimub üks perioodilise võnkumise protsessi tsükkel). Neid võnkumisi nimetatakse heli sageduseks. Heli sagedusega alla 16 Hz nimetatakse infraheliks ja üle 20 000 Hz – ultraheliks. Neid üldjuhul inimese kõrv ei taju, väljaarvatud väiksed lapsed, kes võivad tajuda kuni 40 000 Hz sagedusega heli.

1.2 Heliallikad

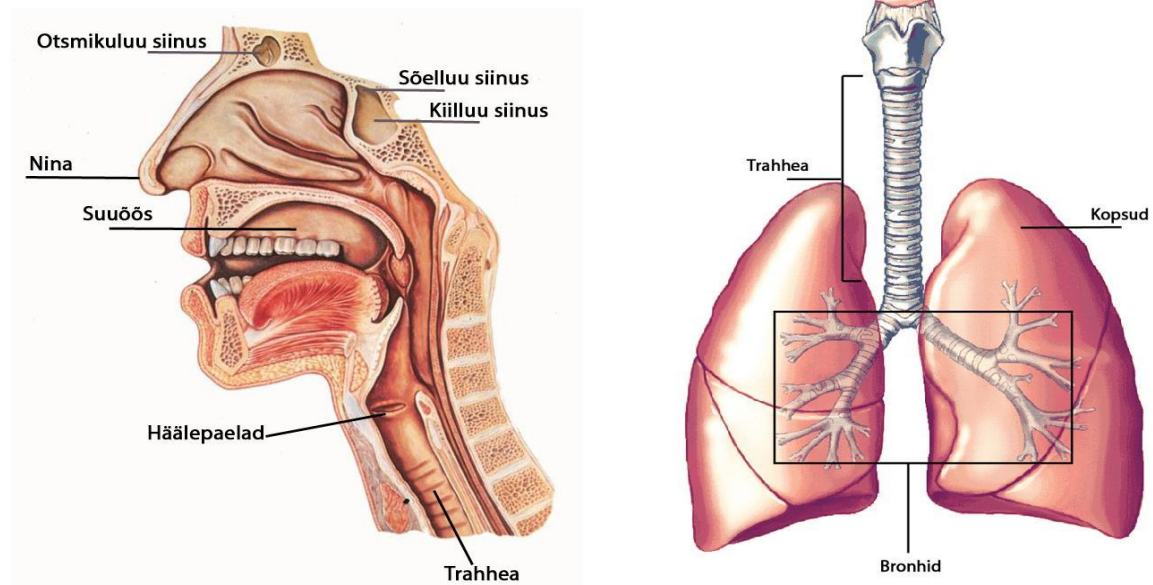
Järgnev osa on koostatud kasutades allikaid [5] - [9].

Iga keha, mis võngub teatud sagedusega, on heliallikas, sest ümbritsevas keskkonnas tekivad temast levivad lained. On olemas nii looduslikud heliallikad, näiteks puulehtede sahin, kui ka kunstlikud, näiteks helihark (vaata joonis 1.2). Helihargi leiutas aastal 1711 Inglise muusik John Shore, kes tegutses trompetistina ning lautomängijana õukondades. Helihark ehk kammertoon on u-kujuline metallhark, mida kõige sagedamini kasutatakse muusikas pillide noodikõrguste määramiseks. Helihark tekitab heli võnkuma pandult lüües seda vastu mõnda objekti või pinda. Kammertooni otsad hakkavad vibreerima, tekitades enda ümber vahelduvaid õhu tihendusi ja hõrendusi, mis omakorda tekitavadki helilaine. Helihargi standardne sagedus on 440 Hz. Silmale pole helilained nähtavad, aga kui puudutada kammertooni käega, on tunda vibreerimist.



Joonis 1.2 Helihark [10]

Inimene ise saab olla heliallikas. Evolutsiooni käigus on inimese kuulmine ja kõne kujunenud välja omavahelt sobitatuna ehk ühildatuna, see tähendab, et inimene suudab häälega heli tekitada samas ulatuses, kui tema kuulmine suudab tajuda. Hääle tekkes osalevad kolm põhilist hingamisaparaadi osa: 1) alumised resonaatorid 2) hääleaparaat; 3) ülemised resonaatorid. **Resonaator** (ladina keeles *resonare* - vastu kajama) on füüsikas keha või kehade süsteem, mis on võimeline hakkama välise ergutuse toimet võnkuma resonantsi tõttu. Alumised resonaatorid on kopsud, bronhid ja trahhea. Hääleaparaadi all mõistetakse häälepaelu. Ülemised resonaatorid on nina- ja suuõõs ning nina kõrvalkoobaste süsteem. Nina kõrvalkoobaste alla käivad otsmikuluu siinus, sõelluu siinus, kiilluu siinus. (Joonis 1.3)



Joonis 1.3 Kõneelundid [11],[12]

Hääl tekib kõris häälelihaste toimele kui ka häälepilu ahendavate lihaste kokkutõmbel. Sellega sulgub häälepilu, sest häälepaelad/kurrud satuvad servapidi üksteisega kontakti. Kopsust väljahingatav õhk surub häälepilu lahti, kuid häälepilu sulgub uuesti, niipea kui pinge, mis hoiab häälepaelu koos, muutub suuremaks kopsust väljahingatava õhu rõhust; väljahingatav õhk surub end häälepilust läbi, avaneb see jälle. Nii tekib häälepaelte liikumine horisontaaltasapinnas. Häälepaelte poolt pidevalt katkestatud õhusammas põhjustab perioodilise õhuvõnkumise, mida tajutakse häälina akustilises mõttes. Kõris tekib häälepilu sulgudes üks põhitoon ja hulk harmoonilisi võnkumisi ehk ületoone. Kõris tekkinud hääl saab kõla ja värvingu ülemises resonantsüsteemis [8].

Heli iseloomustavad helivaljus, heli kõrgus, kiirus ning peegeldus.

1.3 Helivaljus

Järgnev osa on koostatud kasutades allikaid [5],[6],[9] ja [13] - [15].

Helivaljus on heli intensiivsuse subjektiivne tajus. Subjektiivsus tähendab seda, et erinevad inimesed võivad tajuda üht ja sama heli erinevalt, kuigi heli intensiivsus tegelikult ei muutu. Helivaljuse mõõtühikuks on 1 bell (1 B). Nimetus on antud telefoni leiutaja, ameerika teadlase Alexander Belli auks. Sagedamini kasutatakse ühikut 1 detsibell (1 dB). $1\text{dB} = 0,1\text{B}$

[5]. Detsibell on suhteline mõõtühik, mis väljendab kahe füüsikalise suuruse (tavaliselt võimsuse või pinget) suhet või ühe suuruse ulatust võrreldes mingi etalonsuurusega.

Helivaljuse mõistet on hea selgitada näitega. Olles vanaema juures külas, võib märgata, et tema televiisor mängib valjemini. Siiski, ei tundu vanaemale televiisor valjuna, sest vanuse tõttu on tema kuulmine halvemaks läinud.

Heliintensiivsuseks ehk helitugevuseks (I – väike L) nimetatakse energiahulka, mida kannab helilaine ajaühiku jooksul läbi ühikpinna, mis on risti laine levimissuunaga. Mõõtühikuks on $J/(s \times m^2) = W/m^2 = 100 \mu W/cm^2 = 10^3 \text{ erg}/(s \times cm^2)$. Füüsikast on teada, et energiat mõõdetakse džaulides (J). Samuti on teada, et pinnaühikuks on ruutmeeter (m^2), ja ajaühikuks on sekund. Seega, peab olema selge, mida tähendab $J/(s \times m^2)$. See on

$$\frac{\text{heli energia}}{\text{meeter}^2 \times \text{sekund}}$$

Eelpool toodud ühik on samaväärne ühikuga W/m^2 , kus W tähendab helivõimsuse mille mõõtühikuks on vatt [15].

$$\frac{\text{heli võimsus}}{\text{meeter}^2}$$

Selleks, et helisagedusega laine tekitaks heliaistingut, peab heliintensiivsus kõrva vahetus läheduses ületama teatud minimaalse suuruse ($\sim 0 \text{ dB}^*$), mida nimetatakse kuuldeläveks. Kuuldelävi sõltub helisagedusest ja inimese vanusest. Sagedusel 1000 Hz on inimese kuuldelävi keskmiselt $I_0 = 10^{-12} \text{ W/m}^2$, mis on võetud heliintensiivsuse standardseks nulltasemeks [6].

Sagedusel 1000 Hz algab valutunne heliintensiivsusel 100 W/m^2 , mis on nn standardne valulävi. Standardseks valuläveks nimetatakse sellist piiri, millest suurema intensiivsuse heli ei tajuta enam helina vaid kõrvades tekkib valutunne.

1.4 Heli kõrgus

Järgnev osa on koostatud kasutades allikaid [5],[9] ja [14].

Peale valjuse, iseloomustab heli kõrgus. Heli kõrgust määratakse tema sagedusega: mida suurem võnkumiste sagedus, seda kõrgem heli. Väiksema sagedusega võnkumistele vastavad madalad helid. Filmis “Uus Gulliver” (Joonis 1.4) räägivad kääbused peenikese häälega, mis vastab nende väikesele kõrile, aga hiiglane Gulliver räägib madala häälega. Võtetel mängisid kääbuste osi täiskasvanud ja Gulliveri – laps. Näitlejad rääkisid võtetel oma normaalse häälega, tooni muudeti võtete ajal originaalsel viisil. Selleks, et teha kääbuste hääli kõrgeks, Gulliveri oma aga madalaks, kirjutas režissöör kääbuste mängivate näitlejate juttu aeglustatud liikuvale lindile, hiiglase hääle aga, vastupidi, kiirendatult liikuvale lindile. Ekraanile projitseeriti film normaalse kiirusega. Kerge on taibata, mis siin juhtub. Kääbuste kõne jõuab kuulajani normaalse võrreldes suurenenud võnkesagedusega ja tooni kõrgus tõuseb vastavalt. Hiiglase häält võetakse vastu aeglustatud võnkesagedusega ning toon muutub madalamaks [14].



Joonis 1.4 Kaader filmist “Uus Gulliver”. [16]

Kindla sagedusega lainet teisiti nimetatakse muusikalises keelekasutuses tooniks, seega heli kõrgusest tihti räägitakse kui tooni kõrgusest. Muusikalised helid jaotatakse sageduse järgi kaheksasse oktavi. Kõige madalamad on subkontraoktavi helid ja kõige kõrgemad on neljanda oktavi helid (Tabel 1). Oktav jaguneb 12 pooltooniks.

| Oktavi nimetus | LA-noodi sagedus |
|----------------|------------------|
| subkontraoktav | 27,50 Hz |
| kontraoktav | 55,00 Hz |
| suur oktav | 110,00 Hz |
| väike oktav | 220,00 Hz |
| esimene oktav | 440,00 Hz |
| teine oktav | 880,00 Hz |
| kolmas oktav | 1760,00 Hz |
| neljas oktav | 3520,00 Hz |

Tabel 1. Oktavid ja neile vastavad sagedused. [5]

Pillikeel häälestatakse mingile sagedusele. Seda sagedust nimetatakse põhitooniks. Lisaks põhिसagedusele tekitab pillikeel ka ülemtoon, mis on põhिसagedusest väiksema või suurema sagedusega. Põhitoon, koos ülemtoonidega moodustab muusikalist heli. Näiteks, viiuli helid võivad koosneda kuni 15-20 erinevatest sagedustest. Muusikariistadel on kõlakast, mis tõstab mõned ülemtoonid esile ja selle tulemusena tekib muusikariistale omane kõlavärving ehk tämber. Tämber sõltub ülemtoonide arvust ja nende tugevusest [5].

Ka inimese hääl erineb tämbri poolest - mehel esinevad sagedused tavalisel kõnelemisel vahemikus 100-7000 Hz, naisel – 200-9000 Hz [9].

1.6 Heli kiirus

Järgnev osa on koostatud kasutades allikaid [5],[13] ja [17].

Helilaine tekkimiseks ja levimiseks on oluline elastse keskkonna olemasolu. Vaakumis võnkumist ei toimu. Selles on lihtne veenduda lihtsal katsel. Pannes elektrilise kella klaasist

kupli alla ning hakates selle alt õhku välja pumpama, heli läheb vaiksemaks, kuni lõpuks kaob üldse ära.

Helilaine õhus kujutab endast vahelduvaid hõrendusi ja tihendusi. Teisisõnu, tihenduste korral on õhu molekulid on väga tihedalt koos, hõrenduste korral – molekulid asuvad teineteisest kaugemal. Tihenduste korral kasvab õhurõhk ja seega, suureneb elastsus. Vastupidi, hõrenduste korral õhu elastsus väheneb. Peale seda, õhk tihendamisel soojeneb, aga hõrendamisel jahtub. See samuti mõjutab õhu elastsuse, soojenemisel elastsus mõnevõrra suureneb, jahtumisel – mõnevõrra väheneb.

Igas keskkonnas heli kiirus avaldub valemiga [17].

$$kiirus = \sqrt{\frac{elastsus}{tihedus}}$$

Seega heli muutub kiiremaks elastsuse suurenemisel ja tiheduse vähenemisel. Kuigi elastsuse määramine on igas keskkonnas erinev, sobib toodud sõltuvus igas olukorras. Vedeliku keskkonna tihedus on suurem kui gaasi oma, millest võiks eeldada, et vedelikus on heli kiirus väiksem. See aga ei vasta tõele, sest ka elastsus on gaasi omast palju suurem, millest võib järeldada, et vedelikkudes on heli kiirus õhu omast suurem. Sama asi kehtib ka tahke keskkonna korral. Tegelikult heli kiirus sõltub temperatuurist, rõhust ja õhu puhul ka niiskusest (ideaalse õhu puhul õhuniiskusega ei arvestada, siiski reaalses elus, õhk koosneb ka veeaurust).

Nagu näha tabelist (Tabel 2), on heli kiirus õhus olulisemalt aeglasem, kui valguse kiirus – 299 792 458 m/s.

| Keskkond | Kiirus |
|----------|----------|
| Õhk | 330 m/s |
| Vesi | 1450 m/s |
| Raud | 5850 m/s |

Tabel 2. Heli kiirus erinevates ainetes. [5]

See seletab asjaolu, miks äikselisel ilmal on alguses näha välku ja alles mingi aja möödudes on kuulda müristamist.

1.5 Heli peegeldus

Järgnev osa on koostatud kasutades allikaid [9],[14] ja [18].

Heli levimisel ruumis põrkuvad helilained vastu seina ning jõuavad tagasi kuulajani ning ühinevad peamise heliga, seega inimene erinevust ei taju. Juhul, kui peegeldunud helilained jõuavad kuulajani olulise hilinemisega, tajub inimene kahte iseseisvat heli. Seda olukorda nimetatakse kajaks. Kaja ei tekki ainult ruumis, vaid ka looduses, üldjuhul mägistes kohtades.

Eespool toodud heli omadused kirjeldasid eeskätt heli, mille inimkõrv on suuteline tajuma. Järgmistes peatükkidest kirjeldatakse infra- ja ultraheli omadusi.

1.7 Infraheli

Järgnev osa on koostatud kasutades allikaid [9],[19] ja [20].

Helilaineid, sagedusega alla 16Hz, nimetatakse infraheliks, mida inimese kõrv ei taju. Vaatamata sellele suudab infraheli mõjutada inimest ning on kahjulik inimese tervisele. Selline nähtus on seletatav kaasavõnkumisega. Inimese siseelunditel on olemas enda sagedused vahemikus 2 – 20 Hz (Tabel 3). Infraheli, mille sagedus on samas vahemikus, paneb inimese siseelundeid vibreerima ja suure intensiivsuse korral võib tekitada kehasisest verejooksu. Katsed on näidanud, et infraheli võib inimesel tekitada tasakaaluhäireid, iiveldust jt.

| Elund/ Keha osa | Sagedus |
|-----------------|-----------------|
| Silmad | 40,0 – 100,0 Hz |
| Pea | 20,0 – 30,0 Hz |
| Neerud | 6,0 – 8,0 Hz |
| Süda | 4,0 – 6,0 Hz |
| Selgroog | 4,0 – 6,0 Hz |

| | |
|-----------------------|---------------|
| Käed | 2,0 – 5,0 Hz |
| Soolestik | 2,0 – 4,0 Hz |
| Magu | 2,0 – 3,0 Hz |
| Vestibulaarne süsteem | 0,5 – 13,0 Hz |

Tabel 3. Siseorganite omad sagedused. [19]

Mõned loomad on kohanenud infraheli kuulma, näiteks meduus suudab tajuda infraheli sagedusega 8 – 13 Hz. Infraheli, mis tekib tormi ajal, õhuvoogu ja laineharja vastastikusel toimel, ennustab meduusile tulevast tormi.

Infraheli allikateks saavad olla äike (Joonis 1.5), paugud, vulkaanipurse, tuul jne. Infraheli lainetele on iseloomulik kõrge läbitungiv võime: need levivad kaugematele vahemaadele jäädes sama tugevaks. See aitab kindlaks teha suurte plahvatuse asukohad, ennetada tsunamist võimaliku kahju.



Joonis 1.5 Äike on infraheli allikas. [21]

Kuna infraheli on inimesele ohtlik, ei ole sellele leitud laia praktilist kasutust, nagu seda on tehtud ultrahelile.

1.8 Ultraheli

Järgnev osa on koostatud kasutades allikaid [9] ja [14].

Võnkumisi sagedusega üle 20 000 Hz nimetatakse ultraheliks. Delfiinid suhtlevad ultraheli vahendusel, nahkhiired orienteeruvad ruumis samuti ultraheli abil.

Koeradki kuulevad ultraheli, seda omapära kasutatakse tsirkuses. Tsirkuses näidatakse trikki, nagu koerad oskaksid arvutada. Näiteks arvude kaks ja kolm liitmise tulemuse teatab koer haukumisega. Koera dresseerijal on aga vile, mis tekitab üksnes ultraheli. Kui koer on haukunud viis korda annab dresseerija ultrahelivilega märku, et rohkem pole vaja haukuda. Tsirkuse külastajad ultraheli ei kuule ja neil jääb mulje, et koer tõepoolest oskab arvutada.

Ultrahelivõnkumised võivad avaldada piisava intensiivsuse puhul tugevat mõju elusorganismidele: loomsed rakud lõhkevad, verelibled lagunevad, väikesed konnad ja kalad surevad 1 – 2 minuti jooksul [14]. Ultraheli rakendatakse meditsiinis, kus ultraheli abil otsitakse vähki ehk halvaloomulist kasvajat, teostatakse sünnituseelseid uurimusi jne. Ultraheli bioloogiline mõju (põhjustab mikroobide surma) võimaldab kasutada seda piimatoodete töötlemises.

Metallurgias kasutatakse ultraheli mõrade, õhumullide ja teiste defektide avastamiseks massiivses metallitükis. Metallil "läbivalgustamiseks" ultraheliga niisutatakse seda algul õliga ja allutatakse siis ultraheli mõjule. Ebaühtlused metallis hajutavad heli, tekitades nagu helivarju; seega defekte on väga lihtne eristada.

Ultraheli allikaid paigaldatakse laevadele ja allveelaevadele. Üheks selliseks allikaks on kajalood (Joonis 1.6). Kajaloodi kasutatakse merepõhja sügavuse määramiseks ja kalaparvede leidmiseks. Laevalt suunatakse merepõhja lühike ultrahelisignaali, mis peegeldub põhjast tagasi. Arvuti registreerib väljasaadetud ja peegeldunud signaali ajalise vahe ning arvutab kauguse merepõhjani. Vajadusel saab tekitada kajaloodi ekraanile ka merepõhja reljefi.



Joonis 1.6 Kajalood [22]

1.9 Müra

Järgnev osa on koostatud kasutades allikat [17].

Tehnilise progressi mõjud on nähtavad igal pool: prügiaugud, reostunud jõed ja järved, isegi õhk, mida inimene sisse hingab, on reostunud tolmu ja heitgaasidega. Samas, aga on olemas veelgi ohtlikum mõju inimesele – müra (Joonis 1.7).



Joonis 1.7 Müra võib pidada sama ohtlikuks, kui iga teine reostus. [23]

Müra tekitavad korrapäratult võnkuvad kehad. Merelainete kohin, tuule vihin, äikse kärgatus on looduslikud mürad. Kodus tekitavad müra töötav külmkapp, televiisor, pesumasin, ventilaator, personaalarvuti. Tänaval tekitavad müra transpordivahendid.

Müra tekitab inimestele ebameeldivusi ning selle mõju on raske mõõta. On võimalik mõõta helivaljust, aga pole olemas näiteks mõõtvahendid, mis saaks mõõta ärrituse taset. Kraanist tilkuv vesi ei ole suure intensiivsusega heli, samas võib kell 3 öösel ärritada inimest ning ei lase tal magada. Mõned teadlased arvavad, et müra on inimesele sama ohtlik kui reostatud vesi ja õhk. Müra võib mõjutada vaimset ja füüsilist tervist. Müra väsitab inimest, tekitab peavalu, valu kõrvades, kuulmine nõrgeneb. Inimene võib müraga harjuda, kuid müra kahjulik mõju sellest ei vähene.

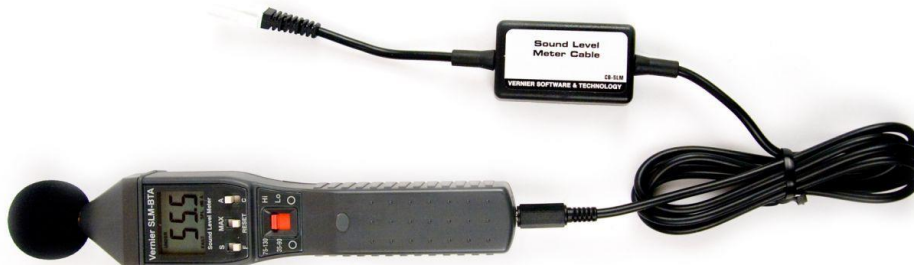
Esimeses peatükis tutvustati heli mõistet ja selle omadusi. Järgnevas peatükis antakse ülevaade firma *Vernier* helivaljuse andurist, selle tööpõhimõttest ja kasutamisevõimalustest.

2. Vernier helivaljuse mõõtja

Selles peatükiks tutvustatakse Vernier'i helivaljuse mõõtjat ja selgitatakse, kuidas seda tuleb kasutada koos NXT komplektiga. Vernier helivaljuse anduri tööpõhimõtet tutvustakse inglise keelse kasutusjuhendi põhjal [24].

2.1 Vernier' helivaljuse anduri kirjeldus

Andur (vaata joonis 2.1) kasutab mõõtmiseks käepideme otsa paigutatud elektrilist mikrofoni, mille väljund läbib ühtlase helirõhu taseme saavutamiseks hulga filtreid ja võimendeid. Andur on suunatav, see tähendab, et mõõtmiseks tuleb seda suunata kindla heli allika poole, muidu on lõpptulemus ebatäpne.



Joonis 2.1 Vernier'i helivaljuse mõõtja

Antud seade mõõdab helivaljusust detsibellides. Seda on võimalik kasutada kas iseseisva mõõturina või ühendades NXT või mõne teise spetsiaalse seadmega (nt. LabQuest 2).

Helivaljuse mõõtja reguleeritakse juhtpaneeli abil, kus paiknevad mõõtmise seadistamiseks vajalikud lülitid. (Vaata joonis 2.2)

Panel Switches



Joonis 2.2 Paneeli lülitid

Helivaljuse mõõteri juhtpaneeli lülitid (*Panel Switches*)

- Sisse lülitamine/ Mõõtmisvahemiku valik (*Power/Measurement Range Switch*) - Liuglülit, mis lülitab anduri sisse ja seab mõõtmispiirkonna.
 - Valides vahemiku 35-90 dB (Lo) lülitakse seade sisse. Selles asendis mõõdab andur mürataset vahemikus 35 kuni 90 dB.
 - Asendis 75-130 dB (Hi), mõõdab andur vastavalt mürataset vahemikus 75-130 dB.

Juhul, kui mõõdetav heli ületab etteantud vahemikku, ilmub hoiatus. Siis tuleb muuta seadistusi vastavalt heli valjusele.

- Aja osakaal (*Time Weighting*) - S/F lüliti muudab aja osakaalu. Tavalisel mõõtmisel sobiks aeglane seadistus "S", mis teeb 1s pausi [25] mõõtmises. Võnkuva müra jaoks sobib aja osakaaluks kiire seadistus "F", see teeb mõõtmises 125ms pausi [25].
- Maksimaalse taseme hoidmine (*Maximum Level Hold*) - MAX/RESET lüliti on mõeldud selleks, et kuvada maksimaalne mõõdetud helivaljus mõõtmise jooksul. RESET seades andur jätkuvalt kuvab mõõdetud näite.
- Sageduse osakaal (*Frequency Weighting*) - A/C lüliti kasutatakse mõõteskaala seadistamiseks

- “A” - väljastab andmeid skaalas, mis on kohandatud inimkõrva tundlikkusele, teisisõnu inimkõrva eristavate helide vahemik.
- “C” - kasutatakse plahvatuste, masinate töö ja sarnaste heliallikate jälgimiseks.

Selleks punktis tutvustati firma *Vernier* helivaljuse anduri tööpõhimõtte, järgmises alampeatükis võrreldakse helivaljuse anduri standardkomplekti kuuluva helianduriga.

2.2 Vernier'i helivaljuse mõõtja võrdlus standardkomplektiga.

Järgnev osa on koostatud kasutades allika [26].

Standardkomplekti heliandur (vaata joonis 2.3) töötab sarnaselt tavalise mikrofonile. Heli andur mõõdab helivaljusust 90 dB ulatuses. Sellest tugevamat heli andur enam ei tuvasta. Mõõdetuid andmeid väljastatakse protsentides.

Näiteks:

4 - 5 % - vaikne ruum

5 - 10 % - kõneleja on andurist kaugel

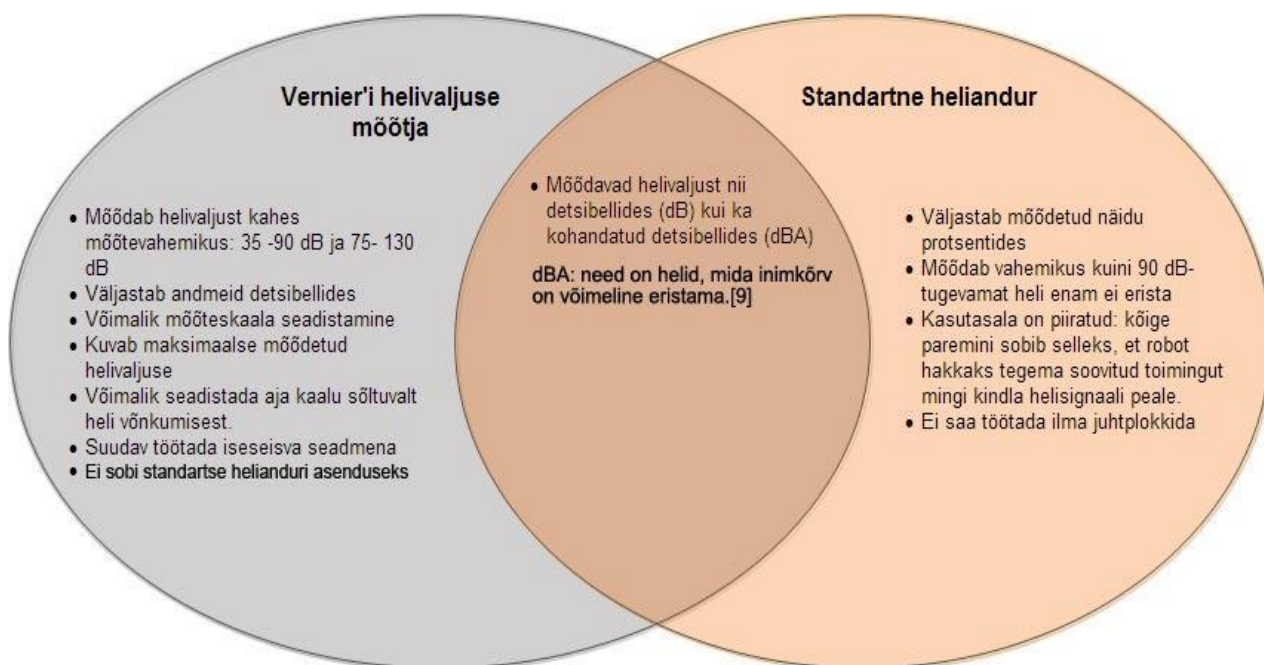
10 - 30 % - kõneleja on anduri juures või ruumis mängib muusika (normaalse helivaljusega)

30 - 100 % - karje või valju muusika



Joonis 2.3 Standardne heliandur [27]

Täpsema mõlema anduri omaduste võrdluste leiab jooniselt 2.4



Joonis 2.4 Vernier'i ja standardse anduri võrdlus.

Vernier'i heliandurit kasutatakse täpseks helivaljuse mõõtmiseks, sest tulemusi väljendatakse detsibellides mitte protsentides, nagu seda teeb standardkomplekti heliandur. Siiski, ei ole Vernier'i heliandur mõeldud selleks, et lasta robotil sooritada mingit tegevust helisignaali peale, kuna selle kuju ei sobi roboti külge panemiseks nii, et andur roboti liikumist ei takistaks. Juhul, kui tahetakse roboti tegevust helisignaaliiga juhtida, siis tuleb valida standardkomplekti helianduri.

Järgmisena kirjeldatakse LEGO Mindstorms NXT juhtplokki ning tutvustatakse lühidalt NXT-G programmeerimiskeskonda.

2.3 Robotikakomplekti LEGO Mindstorms NXT üldine kirjeldus

Selles alampeatükiks tutvustatakse lühidalt NXT juhtimisblokki toetudes LEGO Mindstorms NXT kasutusjuhendile [26]. Seda on küll varasemates töödes tutvustatud, kuid siiski tundub loogiline selles töös veelkord põhilisi asju ära mainida, et edaspidine töö paremini läheks.

NTX juhtplokk on roboti aju - see täidab ülesandeid ja hoiab mälus erinevaid faile ning programme. Juhtplokil on 4 sisendporti andurite ühendamise jaoks, mis on tähistatud 1-4 (edaspidi huvitavad just need) ning 3 väljundporti mootorite jaoks, tähistuseks A,B,C. Andmete väljastamiseks on võimalik info kuvada LCD ekraanile. LCD ekraan on mõõdus 60x100 pikslit. Juhtplokil on samuti olemas 8 KHz sagedusega kõlar. [26] Lisaks on olemas USB port, mille kaudu saab seda arvutiga ühendada. Juhtploki esipaneelil on neli nuppu on 4 nuppu, mille tähendust kirjeldatakse all oleval joonisel 2.5.



Joonis 2.5 Juhtploki kirjeldus [26]

Juhtploki menüüst on võimalik NXT mällu salvestatud programme käivitada ning vajadusel programmi käiku suunata, kui vastane tegevus on programmis kirjeldatud.

2.4 LEGO Mindstorms NXT-G tutvustus

NXT-G on graafiline arenduskeskkond LEGO robotite programmeerimiseks. NXT-G tarkvara on LEGO *Mindstorms* NXT standardkomplektis, *Educational* komplekti korral tuleb tarkvara lisaks installida. NXT-G põhilisteks eelisteks on lihtsus – kasutaja ei pea oskama programmeerimist, kogu töö käib plokkide ehitamise põhjal. Tegu on visuaalse programmeerimisega. Sõltuvalt oskustest võib teha nii lihtsamaid programme (näiteks edasi-tagasi liikumine) kui ka keerulisemaid, mis kasutavad tsükleid või siis lisaplokkide lisamisel võimaldavad kasutada mitmesuguseid andureid.

Kuidas NXT-G keskkonnas tööd alustada siin ei räägita, kuna sellest on juba varem palju kirjutatud [28].

Järgmisena tutvustatakse, kuidas *Vernier* helivaljuse andurit NXT juhtplokkiga ühendada ning kuidas selle abil helivaljust mõõta.

2.5 Vernier'i helivaljuse anduri ühendamine LEGO Mindstorms NXT komplektiga

Firma *Vernier* helivaljuse anduriga töötamiseks LEGO *Mindstorms* NXT-G keskkonnas, tuleb esiteks selle ühendada NXT juhtplokkiga läbi *Vernier*'i adapteri. Ühendamiseks on vajalik:

- NXT juhtplokk
- *Vernier*'i adapter
- USB kaabel
- LEGO *Mindstorms* NXT-G keskkond

Adapterit ennast on võimalik ühendada portidega 1,2,3 või 4. Joonisel 2.6 on näha, et ühenduspordiks on valitud teine port.



Joonis 2.6 Andur on ühendatud juhtplokkiga.

Järgmisena tuleb juhtplokk arvutiga ühendada. Ühendamine käib USB-juhtme abil. Seade tuvastatakse siis, kui juhtplokk on sisselülitatud.

Nüüd vaadatakse lähemalt, kuidas NXT robotit programmeerida keeles NXT-G.

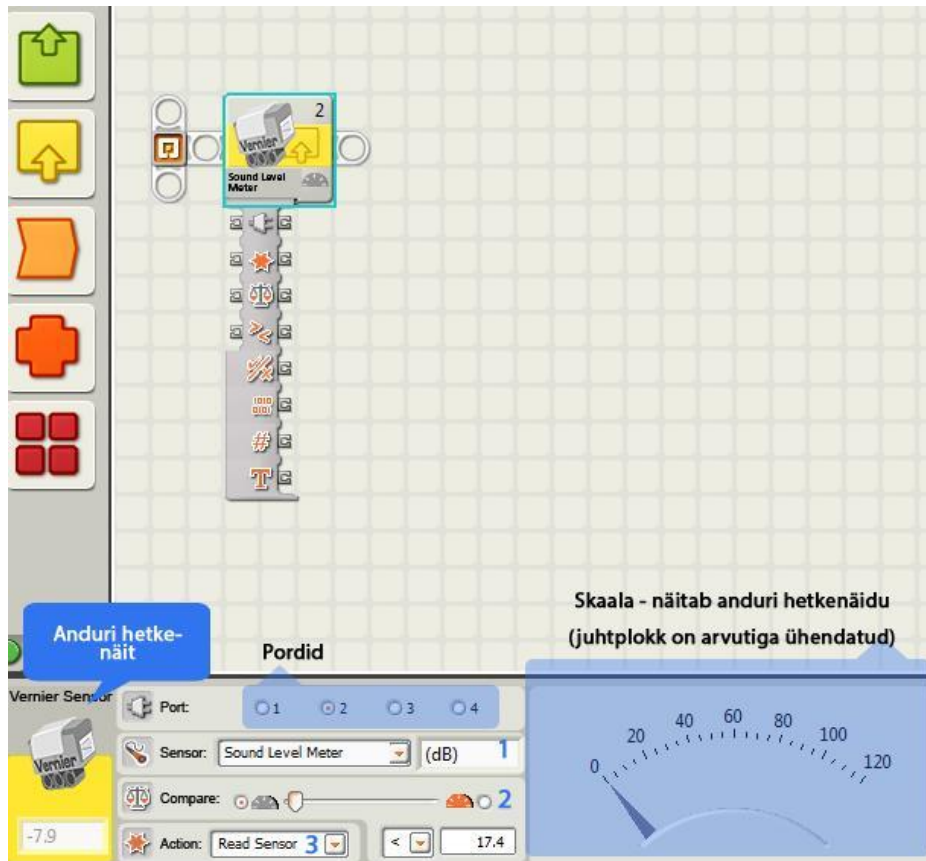
2.6 Helivaljuse anduri programmeerimine keeles NXT-G

2.6.1 Vernier andurite ploki importimine

Enne programmeerimise alustamist tuleb importida *Vernier* andurite plokk NXT-G keskkonda. Selleks tuleb *Vernier*'i kodulehelt alla laadida vastav *zip*-fail [29] arvutisse ja see lahti pakkida. Importimise kohta on võimalik täpsemalt lugeda Siim Jalaka bakalaureusetööst “LEGO Mindstorms NXT komplektiga ühilduv soolsuse andur” [28].

2.6.2 Anduri testimine ja olulisemate funktsioonide tutvustamine

Selleks et kontrollida anduri töökorras olekut, tuleb koostada lihtsa programmi, mis koosneb ainult anduri plokiist. Alguses lohistatakse anduri ploki ikooni programmeerimisalale nagu näha pildil (joonis 2.7).



Joonis 2.7 Programmeerimisala koos valitud anduri omaduste paneeliga.

Selleks, et programmist andurile korraldusi anda, peab anduri ikooni peale vajutama. Allpool ekraanile avaneb omaduste paneel, kus on võimalik muuta järgmisi seadeid:

- **Port** - võimalik määrata, mis juhtimisploki pordiga on andur ühendatud
- **Sensor (1)** - võimalik valida Vernier'i sensoritest helivaljuse andur.
- **Võrdlus (2)** - võimalik valida väärtus, millega anduri näitu võrrelda
- **Tegevus (3)** - võimalik valida 3 tegevuse vahel, kuidas andurit kasutada:
 - *Read sensor* - anduri näidu lugemine
 - *Zero/Calibrate* - anduri nullimine/kalibreerimine
 - *Reset* - algseadete taastamine
- **Skaala** - näitab anduri hetkenäitu graafiliselt

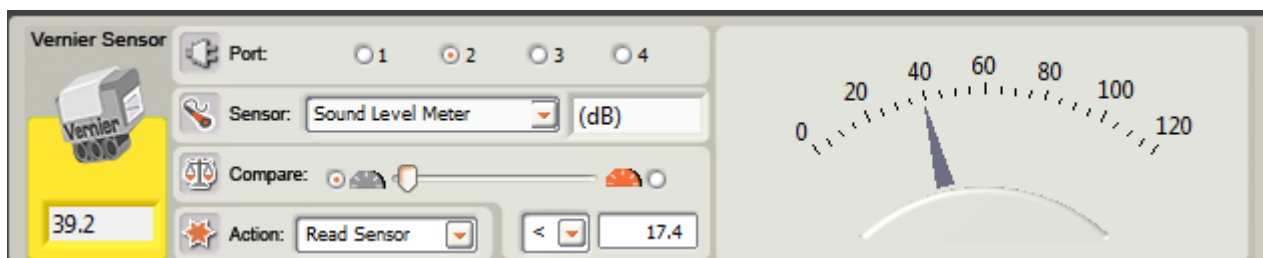
- **Hetkenäit** - näitab anduri hetkenäidu numbrites [28]

Ühendatud helivaljuse andurit tuleb seadistada. Eelkõige tuleb valida pordi, millega andur on ühendatud. Eelnevalt oli helivaljuse andur ühendatud pordiga 2, nii siis peab valima sama pordi. **Sensor** rippmenüüs on võimalik valida *Vernier*'i andurite hulgast õige anduri: valida tuleb **Sound Level Meter**. Muid seadeid ei ole vaja muuta. Kui programm on loodud, peab seda laadima juhtmooduli plokki. Selleks tuleb vajutada all paremas **Download and Run** klahvi (pildil 2.8 märgitud kollasega). Enne programmi juhtploki laadimist tuleb veenduda, et andur ise on sisselülitatud (punane liuglüliti on asendis **Lo**)



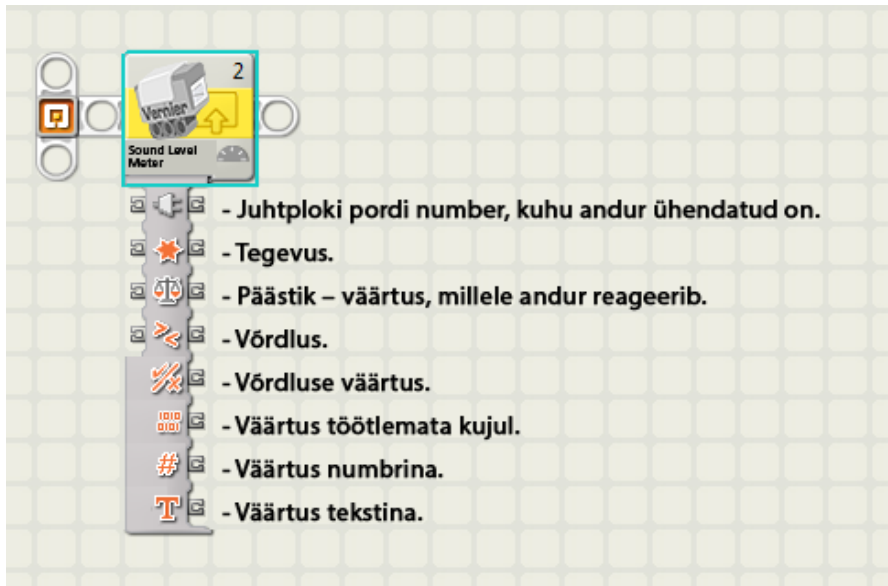
Joonis 2.8 Roboti haldamise nupud.

Kui kõik sammud on edukalt täidetud, hakkab nii anduri ekraanil kui ka omaduste paneelil olev osuti kuvama reaalaajas näite. (vaata joonist 2.9)



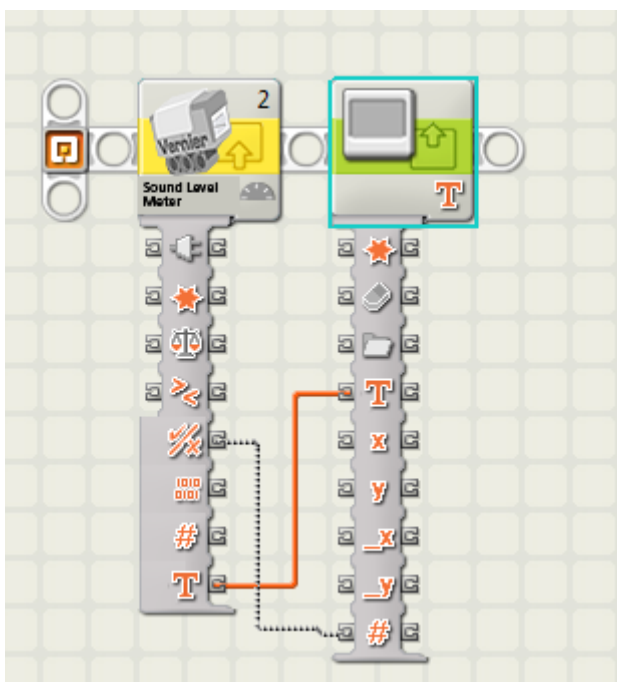
Joonis 2.9 Töötav andur

Anduri plokil on olemas rippmenüü, milles on *Vernier*'i plokki sisendid ja väljundid, nende tähendus on kirjeldatud joonisel 2.10



Joonis 2.10 Anduri plokki sisend ja väljundid.

Kahte plokki omavahel suhtlemiseks, peab looma nende vahel ühendust. Joonisel 2.11 on näide programmist, mis väljastab anduri näitu juhtplokki ekraanile, mille jaoks on programmeerimisalale lohistatud *Display* plokk. Nii sisend, kui ka väljund peavad olema omavahel loogilises seoses, nagu joonisel 2.11 on näha Tekst >> Tekst. Kui ühendus on loodud süntaktiliselt õigesti, märgitakse ühendjoont erksa värviga, vastasel juhul ühendjoon on hall.



Joonis 2.11 Õige ja vale ühendus.

Järgmistes alampeatükkides kirjeldatakse uue muutuja lisamise ning faili plokki kasutamist, eesmärgiks aidata kolmanda peatükki ülesandeid lahendada.

2.6.3 Uute muutujate lisamine

Ülesannete lahendamiseks läheb vaja uute muutujate lisamist. Neid on vaja selleks, et salvestada erinevaid väärtusi, ilma, et need sassi läheks ning iga muutuja hoiaks ainult talle mõeldud väärtuse.

Programmeerimiskeeltes muutuja on teatavat tüüpi väärtuste hoidmiseks ettenähtud mäluväli. Muutujale võib omistada teatava väärtuse ning kasutada seda avaldise väärtuste arvutamisel. Muutujale viidatakse tema nime järgi. [30] NTX-G programmeerimises muutujat (*Variable*) tähistatakse ikooniga, mis väliselt meenutab kohvri. (Vaata joonis 2.12)

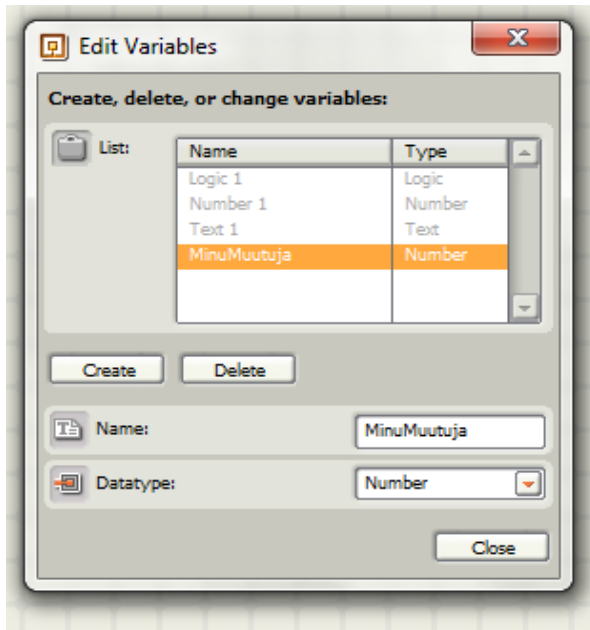


Joonis 2.12 Muutuja plokk

NXT-Gs on juba vaikimisi defineeritud 3 muutujat:

- **Logic 1** - muutuja, millele saab omistada tõeväärtuse - õige või vale (*True or False*)
- **Number 1** - muutuja, millele saab omistada numbrilist väärtust
- **Text 1** - muutuja, millele väärtuseks on sõne.

Juhul, kui vajatakse mitu numbrilist muutujat, mis programmi töötamise ajal ei ole võrdsed, siis ei saa piirduda ainult ühe numbrilise muutujaga. Tuleb lisada uusi muutujaid. Selleks tuleb minna **Edit > Define Variables**. Avaneb aken, kus saab lisada uusi muutujaid (vaata joonis 2.13)



Joonis 2.13 Uue muutuja lisamine

Uue muutuja puhul on kindlasti oluline anda talle nimi, ning tüüp - mis väärtusi talle hakatakse omistama. Muutujaid luuakse, kasutades klahvi **Create**.

2.6.4 Failidega opereerimine

Faili ploki (Joonis 2.14) abil saab NXT mällu luua uusi faile, neisse informatsiooni **kirjutada** (*write*), failist andmeid **lugeda** (*read*), faile saab **sulgeda** (*close*) ja ka **kustutada** (*delete*).

Plokis on kolm parameetrit:

- faili nimi
- tekst
- number



Joonis 2.14 Faili plokk

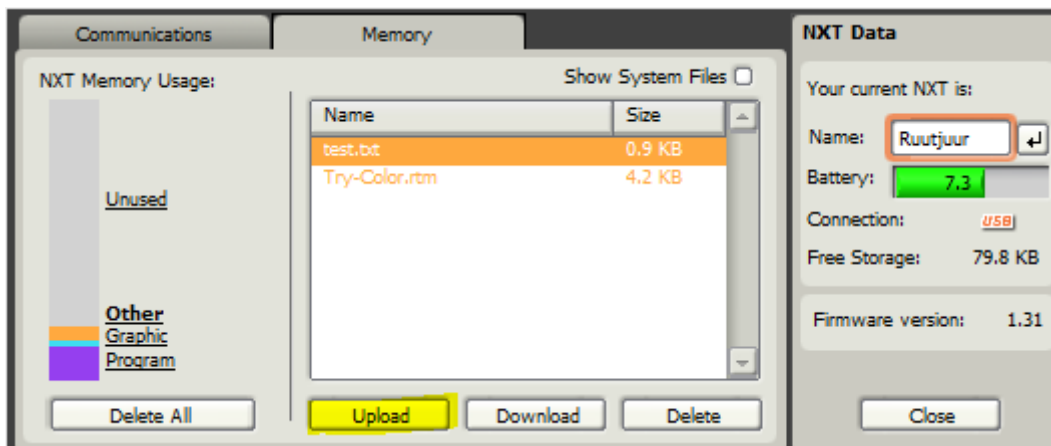
Ploki tegevuse määramiseks, tuleb **Toimingute** (*Action*) alt valida vajalik tegevus.

Failile nime andmine käib Toimingute all, sinna lahtrisse **Simple Text** tuleb kirjutada sobiv nimetus, ilma laiendita (ei kirjutata lõppu .txt). Selleks, et loodud tekstifaili ka arvutis analüüsida, tuleb see NXT mälust arvutisse importida. Alustamiseks tuleb vajutada **NXT Window** (vaata joonis 2.15)



Joonis 2.15 Roboti mälu haldamise nupp.

Avanenud aknas valida teek **Memory** ning **NXT Memory Usage** valida **Other**, kus paikneb loodud tekstifail. (vaata joonis 2.16). Valides õige faili vajutada nuppu **Upload**. Salvestada fail vajalikku kohta oma arvutis.



Joonis 2.16 NXT mälu haldamine

Selleks aknas saab ka ebavajalikke faile, programme kustutada, juhul kui NXT mälu saab täis.

Käesolevas peatükis kirjeldati *Vernier*' helivaljuse andurit ning toodi välja olulisemad punktid, mida peab teadma anduri kasutamisel LEGO *Mindstorms* NXT komplektiga. Järgmises peatükis on esitatud helivaljuse anduriga ja NXT komplektiga tutvumiseks mõeldud ülesanded.

3. Ülesanded

Antud peatükis on pakutud ülesanded, mida õpilane saaks iseseisvalt tunnis lahendada. Igal ülesandel on juures tema raskustase, eesmärk ning ülesande sooritamiseks oluliste vahendite loetelu, mille alla käivad kindlasti ka LEGO *Mindstorms* NXT komplekt, *Vernier*'i helivaljuse mõõtja ning NXT programmeerimistarkvara.

Kokku on viis ülesannet. Esimesed kaks on lihtsad ülesanded ning on mõeldud selleks, et kontrollida anduri korrasoleku ning tutvustada NXT-G programmeerimiskeelt. Ülejäänud kolm on juba keerulisemad ülesanded, mis nõuavad õpilaselt NXT-G keerulisemate konstruktsioonide kasutamist.

3.1 Sissejuhatav ülesanne

Tase: Väga kerge

Eesmärk: Testida helivaljuse mõõtjat ning tutvustada õpilasele NXT-G programmeerimiskeelt.

Ülesande sooritamiseks olulised vahendid:

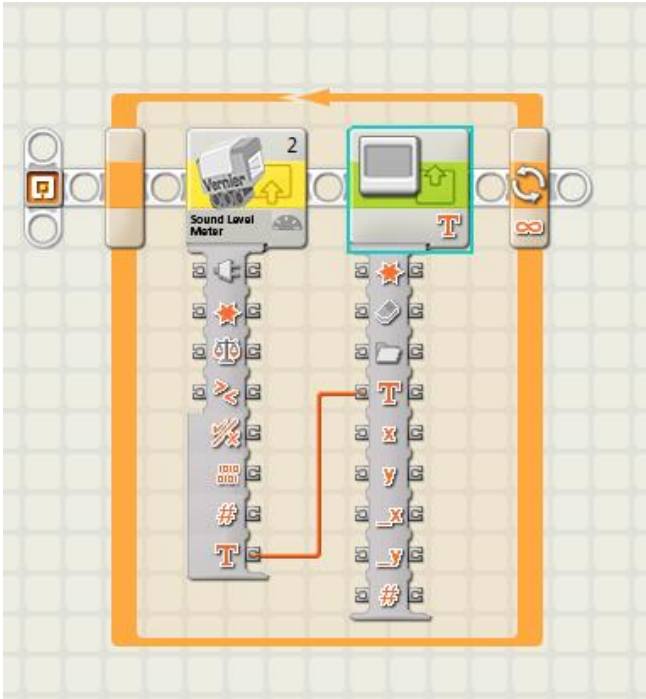
- NXT juhtplokk
- *Vernier*'i adapter
- *Vernier*'i helivaljuse mõõtja
- USB kaabel
- LEGO *Mindstorms* NXT-G keskkond

Ülesande püstitus: Ühendada helivaljuse mõõtja juhtplokkiga ning väljastada lõpmatus tsüklis juhtplokki ekraanile anduri näitu.

Lahendus:

See ülesanne on sissejuhatuseks ning peab olema õpilastele lihtne. Põhiidee on väljastada anduri näitu juhtploki ekraanile, *Cancel* nupu vajutusega programmi töö katkestatakse.

Lahenduseks sobib peatükis 2 toodud näide (Joonis 2.11), mis on paigutatud lõpmatusse tsüklisse. (vaata joonis 3.1)



Joonis 3.1 Anduri näitu kuvamine lõpmatu tsükli.

Tsükli ülesandeks on mingit tegevust mitu korda täita nii kaua, kui jätkamistingimus on täidetud.

NXT-G on tsükli kontrollimiseks hästi mitu võimalust:

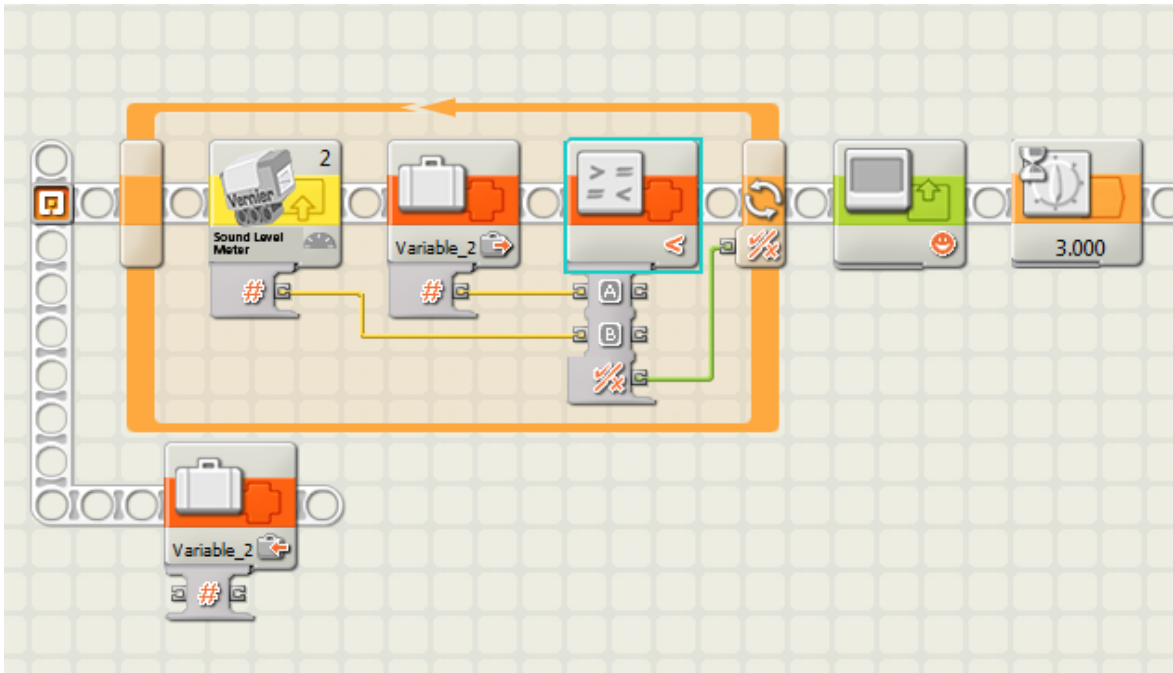
- **Forever**- lõpmatu tsükkel. Töötab niikaua kuni **Cancel** nuppu vajutatakse
- **Sensor**-tsükkel teeb tööd niikaua, kuni valitud sensoriga mida tehakse.
- **Time**- tsükkel lõpetab tööd mingi kindla aja möödumisel
- **Count**- iga kord, kui tsükkel on algusest lõppuni läbitud, kasvab count ühe võrra. Andes ette mingi arvu, millest suuremaks *count* minna ei saa, siis nii palju kordi tsükli läbitakse. Analoogne tsükli töö põhimõttega teistes programmeerimiskeeltes.

```
for (i=0; i < 5; i+1) { //for tähendab siin tsükli direktiivi
    //tegevus
};
```

Esialgne *count* (*i*) on võrdne 0, kui tsükli sisemist tegevust ära tehakse, kasvab *i* ühe korda. Kui *i* jõuab väärtuseni 5 tsükkel lõpetab tööd.

- **Logic**- tsükkel töötab nii kaua, kui tsüklitingimus on kas Tõene või Väär. [31] Selline tsükkel nõuab kindlasti sisendit - mis on *Logic* tüüpi.

Joonisel 3.2 on näide tsüklist, mis kasutab *Logic* - tingimust. Selle programmi põhimõtte on lihtne - töötada nii kaua, kuni anduri näit on suurem kui 85,5 dB, siis väljuda tsüklist ja näidata ekraanile *smail* 'i.



Joonis 3.2 *Logic*-tsükli näide

Võimalikud probleemid ülesande lahendamise ajal:

- Kontrollida, et andur on ühendatud juhtplokkiga ning juhtplokk ise on ühendatud USB-kaabli vahendusel arvutiga.
- NXT juhtplokk ja andur peavad olema kindlasti sisselülitatud.
- Tuleb kontrollida, et anduri plokk on õigesti seatud. Valitud õige andur ning port, millega ta on ühendatud.
- Kui ekraanile ilmub midagi muu, siis võimalik, et on *Display* ploki vale tegevus seatud
- Tuleb kontrollida, et ühendus anduri ploki ja *Display* plokkide vahel on süntaktiliselt korrektne.

3.2 Ülesanne andmete analüüsi operatsiooniga

Tase: Kerge

Eesmärk: Õpetada õpilasi *DataLog* operatsiooni kasutama

Ülesande sooritamiseks olulised vahendid:

- NXT juhtplokk
- Vernier'i adapter
- Vernier'i helivaljuse mõõtja
- USB kaabel
- LEGO *Mindstorms* NXT-G keskkond
- *DataLog*

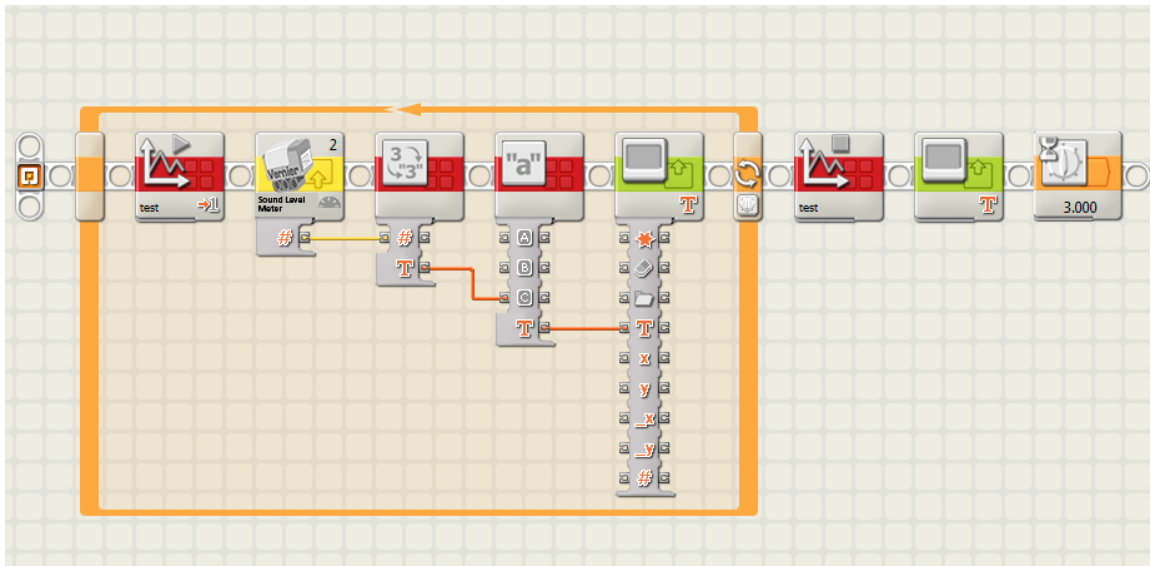
Ülesande püstitus: Mõõta mürataset vähemalt kahes erinevas keskkonnas näiteks:

- klassiruum
- kooli koridor vahetunni ajal

Analüüsida operatsiooniga *DataLog* tehtud diagramme.

Lahendus:

Üks võimalik selle ülesande lahendus on toodud joonisel 3.3.



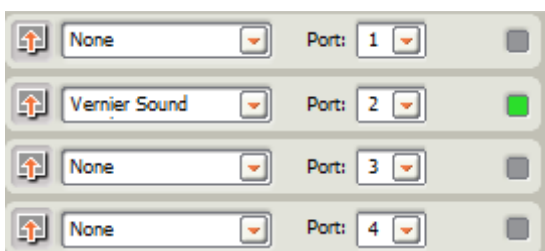
Joonis 3.3 *DataLog* väljakutsumine ja anduri näitu kuvamine juhtplokki ekraanile.

Operatsioon *DataLog* alustab tööd siis, kui tema plokk on seda välja kutsunud ning ei sõltu teistest plokkidest, seega *DataLog* plokki võib katsetada helianduri ja sellele järgneva

konverteerimise ja väljastamise plokk programmeerimisalale lohistamata. Teine asi, et siin tutvustaks lisaks sõne loomist ja lõpliku tsükli tööd, mida läheb õpilasel järgmiste ülesannete lahendamiseks vaja.

Mida siis eelpool toodud programme teeb? Luuakse tsükli, mille kestvuseks on pandud 20 sekundit. Tsükli ajalast kulgemist võib määrata tsükli juhtimispaneelist, **Control** menüüst. Tsükli sees alustab tööd *DataLog* plokk. Konkreetsele logile on antud nimi - “**test**”, vaikimisi on see “**myData**”.

DataLog plokki juhtimispaneelis valida **Kestvuse** (*Duration*) - Ühekordne mõõtmine (**Single Measurement**). Seda on vaja selleks, et *DataLog* lõpetaks enda tööd siis, kui tsükkel lõpetab tööd. Programm lõpetab töö veaga, kui *DataLog* ei saanud endale ette anduri, mille väärtusi ta salvestab. Seega tuleb seda kindlasti talle ette määrata (vaata joonis 3.4)



Joonis 3.4 Operatsioonile *DataLog* kindla anduri määramine.

Samal ajal, kui *DataLog* teeb enda tööd, soovitav on, et ka NXT juhtplokki ekraanile näitaks hetkenäitu, aga siin ei taheta, et ta neid nagu esimeses ülesandes näitaks. Tuleb andma talle soliidsema kuju: <values: anduri hetkenäit>. Selleks peab koostama sõne, mida edastatakse ekraanile. **Text** plokk seab ühtseks lauseks mitu sõne. Teksti plokk aktsepteerib sisendiks ainult sõnesid, seega anduri väärtusi peab konverteerima sõneks. Numbri sõneks konverteerimine käib **Number to Text** plokki kaudu.

Text plokkis on kolm sisendit:

- **A – Values:**
- **B – ei midagi või tühik**
- **C – Anduri väärtus.**

Nii pea, kui tsükkel lõpetab oma tööd, lõpetab seda ka *DataLog*. Siis väljastatakse NXT juhtplokki ekraanile sõne “Olen lõpetanud!”, oodatakse 3 sekundit ja programm lõpetab töö.

Joonisel 3.5 on näha mitmekordse mõõtmise tulemusi. Selleks, et graafik oleks loetav, võib muuta iga joone värvus ning anda igale mõõtmisele ka nimi, näiteks mis keskkonnas helivaljust mõõdeti.



Joonis 3.5 Kolme mõõtmisessiooni tulemusdiagramm.

Võimalikud probleemid ülesande lahendamise ajal:

- Kontrollida, et andur on ühendatud juhtplokkiga ning juhtplokk ise on ühendatud USB-kaabli vahendusel arvutiga.
- Kontrollida, et kõik ühendused on legaalsed.
- *Data Log* ei hakka tööle ning programm lõpetab veaga tööd - tuleb kindlasti kontrollida, et *Data Log* plokk juhtimispaneelil on õige andur ja port ette seatud. Kui see ei ole tehtud, siis *Data Log* ei tea, kust andurist ta andmeid loeb
- Programmi ei edasta juhtplokkile - tekkib mälu viga. *Data Log* operatsiooniga eksperimenteerimisel võib logi-fail väga mahukaks muuta. Sellise probleemi tekkimisel tuleb varasemad logid juhtplokkile mälust eemaldada.
- Andur ei lõpeta tööd korrektse sõnega - tuleb kindlasti kontrollida, mis peatustingimus on tsüklile ette seatud.

3.3 Andmete kogumise ja analüüsimise ülesanne

Tase: Keskmine

Eesmärk: Õppida anduri abil ilma *DataLog* operatsioonita andmeid koguma.

Ülesande sooritamiseks olulised vahendid:

- NXT juhtplokk
- Vernier'i adapter
- Vernier'i helivaljuse mõõtja
- USB kaabel
- LEGO *Mindstorms* NXT-G keskkond
- *Microsoft Excel* või sarnane programm

Ülesande püstitus:

Mõõta mürataset vähemalt kahes erinevas keskkonnas näiteks:

- klassiruum
- kooli koridor vahetunni ajal

Salvestada kogutud andmeid koos ajatempliga tekst-faili ja neid hiljem analüüsida (näiteks tekitada Excelis diagramm)

Lahendus:

Programmi koostamise alustamisel peab arvestama, et kuna ülesanne püstituses on nõutud mitu korda programmi läbimist, siis ei tohi mitme sessiooni mõõtmised failisse järjest salvestuma. Selliste andmetega ei ole midagi targemat teha. Seega iga mõõtmise lõpus peab tekstilist faili arvutisse importima ja programmi töö alguses faili kustutama. Seda võib teha ka siis, kui teksti faili importi tehakse, aga sellist lahendust ei soovitata. Õpilane peab faili plokkiga harjutama. Järgmisena tekitatakse tsükli. Tsükli kestvuse määratakse vastavalt sellele, kui kaua kavatsetakse helivaljust mõõta.

Sarnaselt operatsiooniga *DataLog* peab lisama ajatemplit anduri näitude ette, mis hetkel on helivaljust mõõdetud. Väärtused tuleb kirjutada tekstifaili kujul: <ajatempel; anduri näit>.

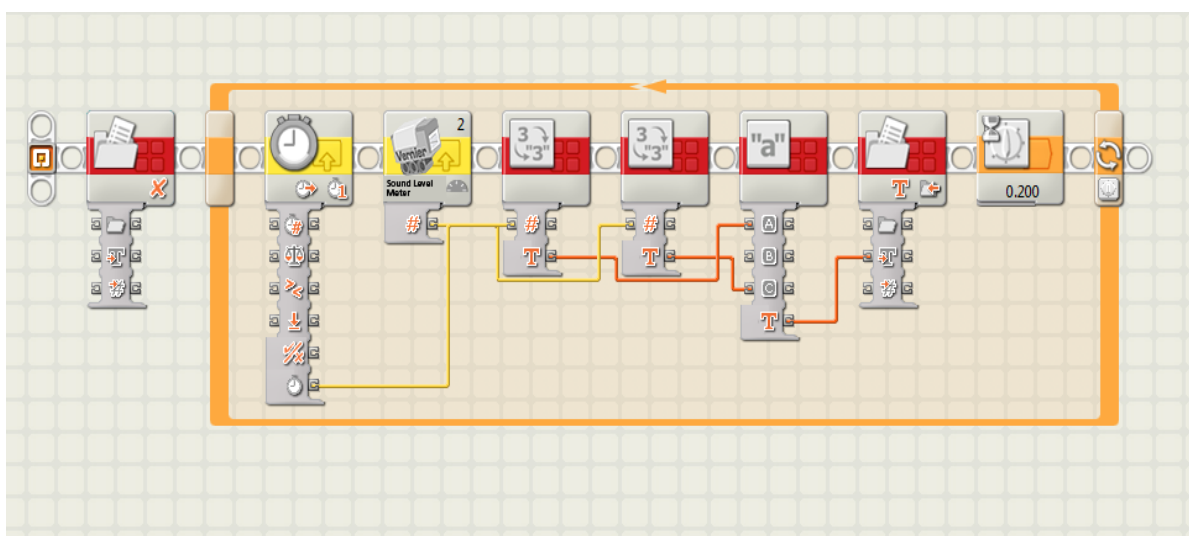
Selleks, et see õigelt kujul failisse jõuaks, peab ajatemplit ja anduri näitu üheks sõneks kokku panema - **Text** plokkis.

Text plokki on kolm sisendit:

- **A – Timer'i väärtus**
- **B – ; (semikoolon)**
- **C – Anduri väärtus.**

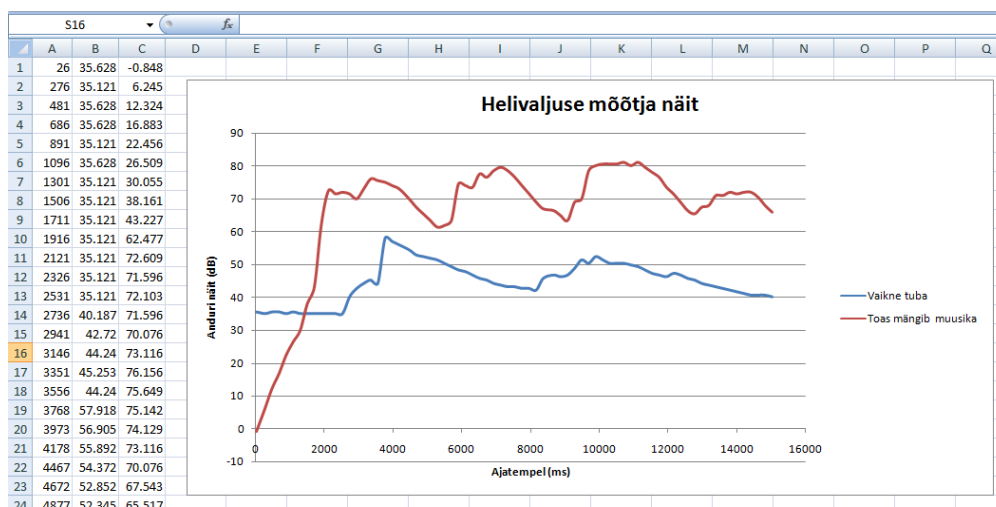
Semikoolonit tuvastavad statistika programmid kui eraldajat. Exceli puhul eraldatakse ajatempli näitu ja anduri näitu erinevatesse tabelitesse.

Üks võimalik lahendus on näidatud allpool oleval Joonisel 3.6.



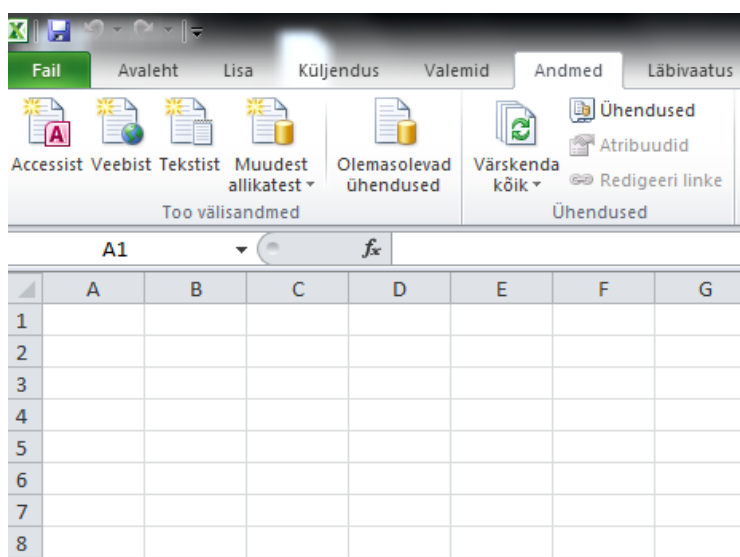
Joonis 3.6 Anduri näitu koos ajatempliga failisse kirjutamine

Joonisel 3.7 on näide väärtuste analüüsist Excelis.



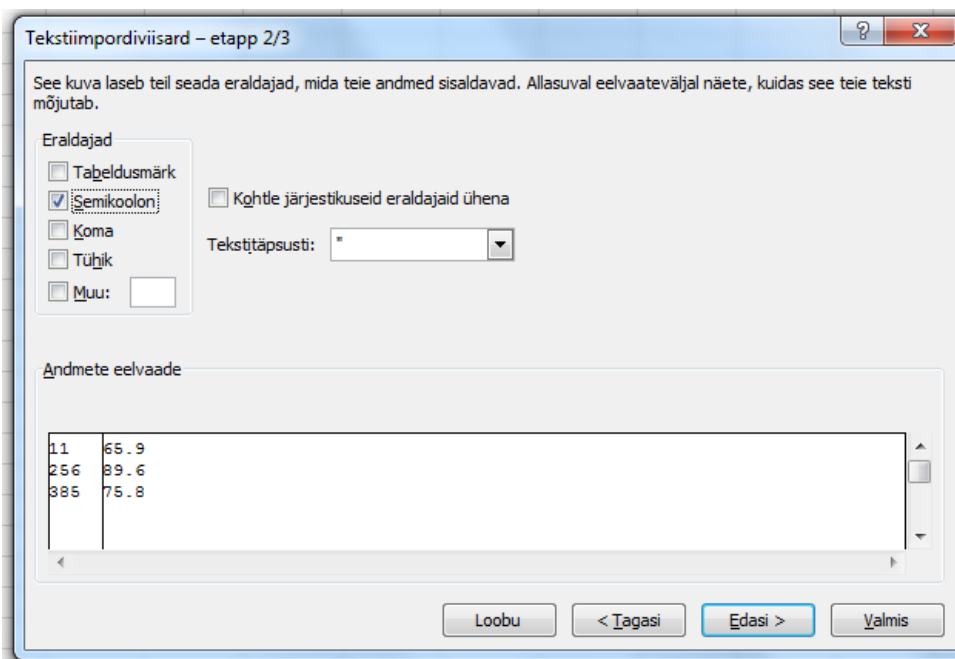
Joonis 3.7 Kogutud andmete põhjal koostatud diagramm

Excelis käib andmete importimine teksti failist väga lihtsalt. Teegist **Andmed** tuleb valida **Tekstist**. (vaata joonis 3.8) Avanenud failisse sirvijast leida soovitud fail.



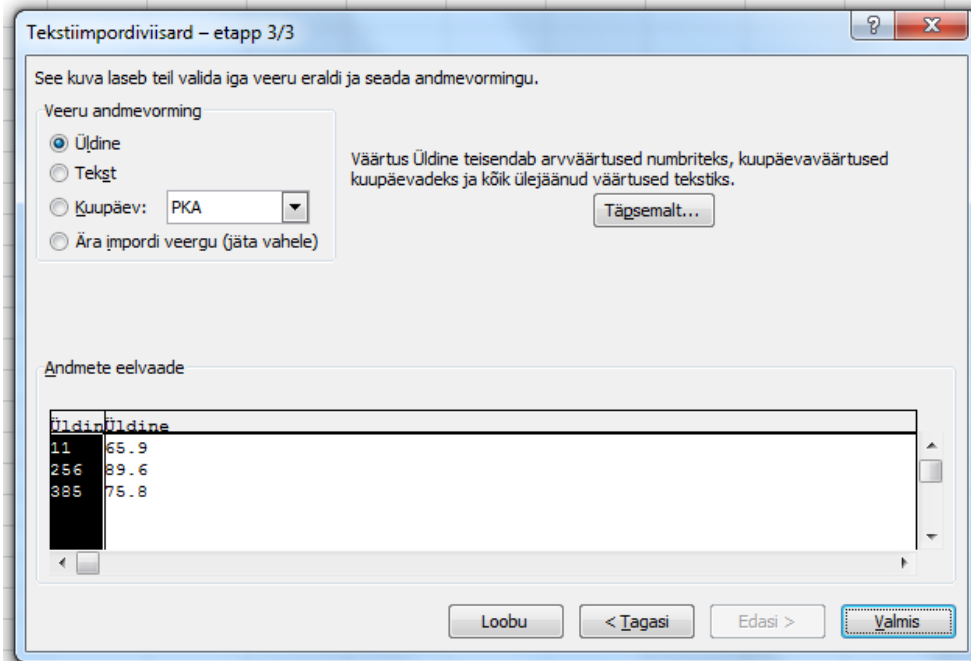
Joonis 3.8 Excelis Andmete teek valitud

Järgmisena avaneb dialoog aken. 1 etapis ei muuda midagi ning vajuta **Edasi** >
2 etapis valida eraldajaks semikooloni. Andmete eelvaatest saab kohe näha, et andmed on nüüd kahes tulbas. (vaata joonis 3.9)



Joonis 3.9 Andmete importimise protseduuri 2. etapp

3 etapis ei muuda midagi esmasel importimisel. (vaata joonisel 3.10) Kui soovitakse lisada hiljem juurde veel andmeid, näiteks peale 2 mõõtmisessiooni, siis tuleb valida “Veerude andmevormingus” **Ära impordi veergu (jätta vahele)**.



Joonis 3.10 Andmete importimise protseduuri 3. etapp

Eduka importimise tulemuseks on kaks tabelit nagu joonisel 3.11

| | A | B | C |
|---|-----|------|---|
| 1 | 11 | 65.9 | |
| 2 | 256 | 89.6 | |
| 3 | 385 | 75.8 | |
| 4 | | | |
| 5 | | | |

Joonis 3.11 Andmed on teksti-failist edukalt imporditud.

Võimalikud probleemid ülesande lahendamise ajal:

- Failisse ei kirjutata midagi - kindlasti tuleb üle kontrollida, et vajalikku sõne edastatakse faili. Faili plokkis peab toimingute seast valitud **Kirjuta** (*Write*)
- Fail ei kustu uue sessiooni alguseks - see võib juhtuda siis, kui faili ploki on määratud erinevatele failidele. Asi on selles, et failiga opereerimise ikooni töölauale

lisamisel antakse vaikimisi failile nimi. Seega siin puhul peab jälgida, et nii kustutamine kui ka kirjutamine käib sama failiga.

3.4 Maksimaalse väärtuse leidmine

Tase: Raske

Eesmärk: Õppida kasutama programmi koostamisel muutujaid ning võrdlusplokki.

Ülesande sooritamiseks olulised vahendid:

- NXT juhtplokk
- Vernier'i adapter
- Vernier'i helivaljuse mõõtja
- USB kaabel
- LEGO *Mindstorms* NXT-G keskkond

Ülesande püstitus: Mõõta klassiruumis mürataset ja väljastada maksimaalse väärtuse juhtplokki ekraanile.

Lahendus:

Selle ülesande lahendamiseks peab enne tsükli deklareerima muutujat (edaspidi **uus**), millega hakatakse anduri näitu jooksvalt võrdlema. Muutuja peab olema number-tüüpi ning vaikimisi on tema väärtuseks pandud 0.

Tsükli kestvuse määratakse vastavalt sellele, kui kaua kavatakse helivaljust mõõta. Tsükli sees luuakse uus muutuja (edaspidi **esialgne**), mis hoiab anduri hetkenäidu. Iga kord kui **esialgne** saab endale uue väärtuse, võrreldakse see **uue** väärtusega. Juhul, kui **esialgne** väärtus on suurem kui **uue** väärtus, siis antakse muutujale **uus** see sama **esialgne** enda väärtuse ja minnakse uuesti mõõtmata. Kui aga **esialgne** on **uuest** väiksem, siis **uue** väärtus jääb samaks ja mõõdetakse uuesti.

Vastavalt konkreetsele väärtusele programmi täitmise kulgu muutmist kõikides programmeerimiskeeltes teostatakse tingimusdirektiivi abil. Üldine kuju on järgmine:

if (loogiline avaldis)

tegevus1

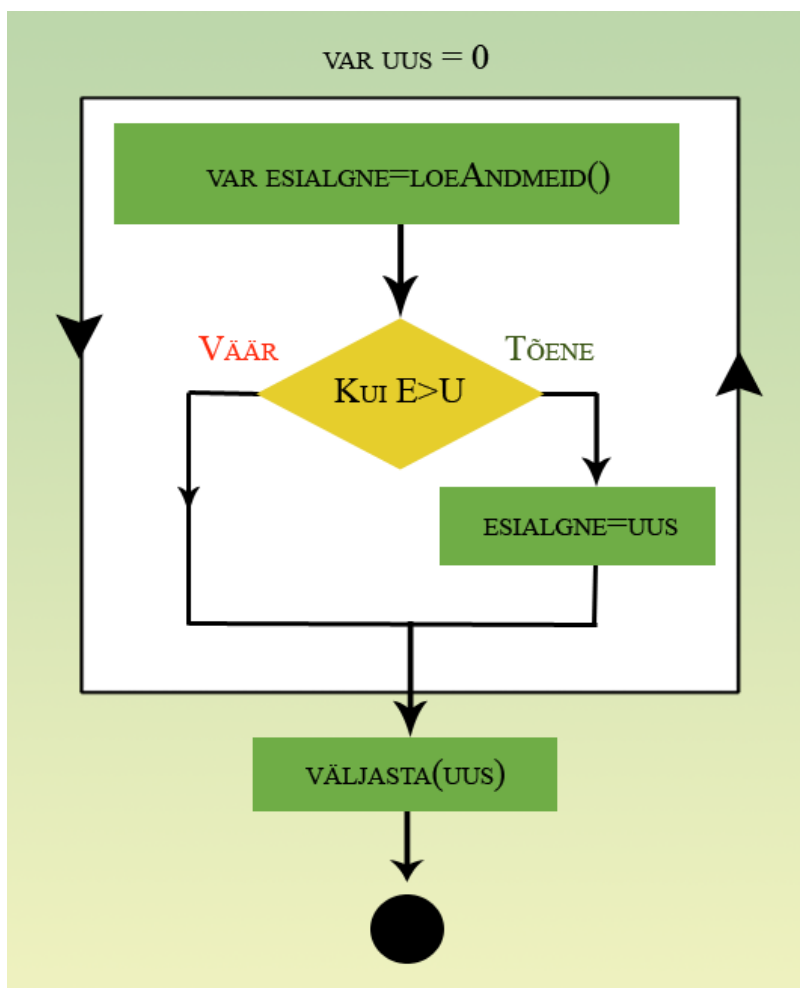
else

tegevus2

Tingimusdirektiiv algab sõnaga kui (*if*), millele järgneb loogiline avaldis (mille väärtus saab olla kas tõene või väär) ja sellele omakorda tegevus, mis täidetakse siis, kui avaldis on tõene. Sõnale muidu (*else*) aga järgneb tegevus, mis täidetakse siis, kui loogiline avaldis on väär. Tingimusdirektiivi else-osa võib ka üldse puududa. [30]

NXT-Gs on tingimusdirektiivi rollis plokk *Switch*. *Switch*'i kasutamist loogilise avaldise kontrolliks tuleb juhtimispaneelist valida *Control* menüüst *Sensor*'i asemele **Väärtus** (*Value*) ja *Type* valida *Logic*.

Maksimaalse väärtuse tuleb väljastada NXT juhtploki ekraanile. Väljastamisprotsess peab olema kindlasti kirjeldatud tsüklist väljas, sest ei taheta vahetulemusi vaid lõpliku tulemust. Programmi algoritmi kirjeldava plokk-skeemi on näha Joonisel 3.12



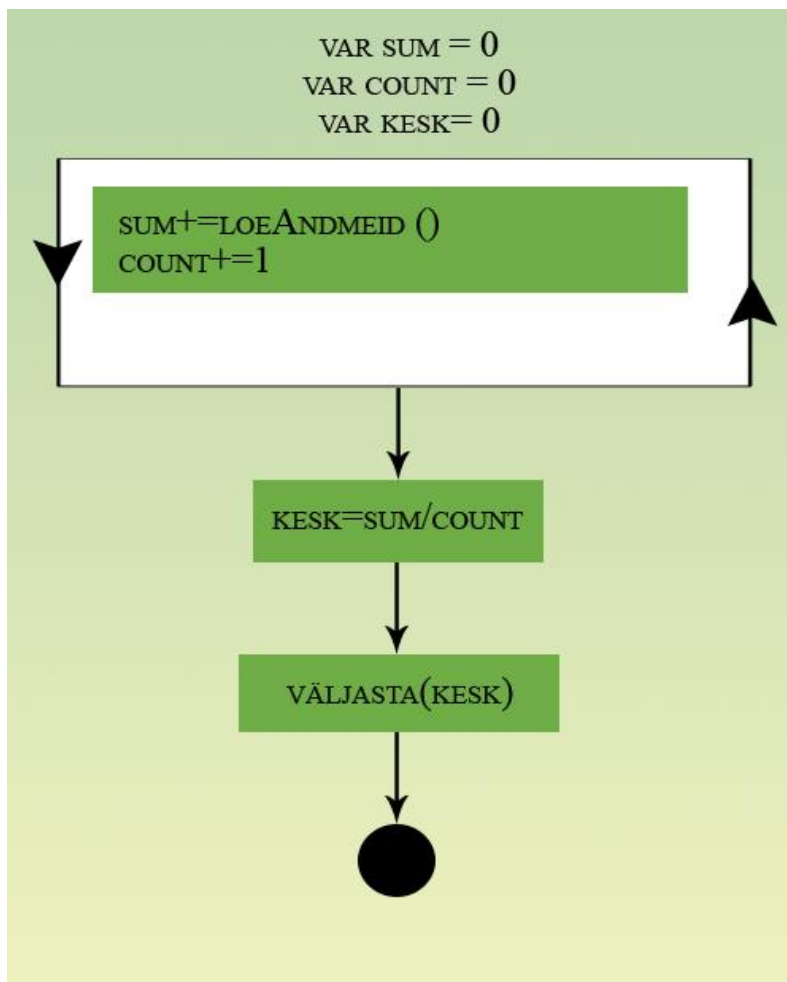
Joonis 3.12 Maksimaalse väärtuse leidmise algoritm

Selle ülesanne puhul ei pea piirduma ainult ühekordse mõõtmisega. Võib kavandada mitu mõõtmisessiooni ja omavahel tulemusi võrrelda.

Lahendus:

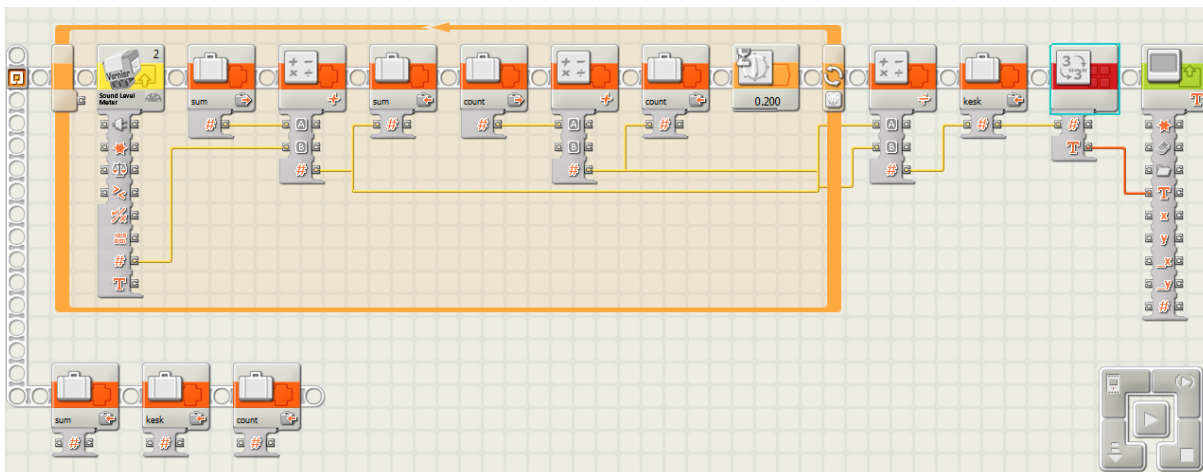
Lahenduse idee sarnaneb maksimaalse väärtuse leidmisel. Vahe on selles, et muutujale *sum*, mis on esialgselt võrdne 0, liidetakse iga andurist tuleva väärtuse juurde ning see järel kasvatatakse muutuja *count* väärtuse ühe võrra suuremaks. Muutuja *count* on vajalik selleks, et hiljem saadud summa jagada saadud väärtuste arvuga. Summa leidmine käib tsüklis, mis töötab nii kaua, kui mõõta soovitakse.

Muutuja *kesk* saab endale väärtuseks $sum/count$ ning seda väljastatakse juhtplokki ekraanile. Programmi algoritmi kirjeldava plokk-skeemi on näha Joonisel 3.14



Joonis 3.14 Keskväärtuse leidmise algoritm

Üks võimalik ülesande lahendus on näidatud allpool oleva joonisel 3.15.



Joonis 3.15 Keskmise leidmine ja selle kuvamine juhtplokki ekraanile

Võimalikud probleemid ülesande lahendamise ajal:

- Kontrollida, et andur on ühendatud juhtplokkiga ning juhtplokk ise on ühendatud USB-kaabli vahendusel arvutiga.
- Kontrollida, et kõik ühendused on süntaktiliselt korrektsed.
- Muutujad peavad olema deklareeritud tsüklist väljas, muide neile alati tagastatakse nende esialgne väärtus.

Kokkuvõte

Käesoleva bakalaureusetöö eesmärgiks oli luua põhjalik eestikeelne õppevahend firma Vernier helivaljuse mõõtja kasutamiseks koos NXT komplektiga, mida õpetajad ja õpilased saavad kasutada füüsika tundides või robotika ringides.

Töö loomisel jälgiti struktuuri, mida kasutatakse sarnaste tööde puhul. Töö koosneb kolmest peatükist. Esimeses andi ülevaade heli omadustele, toodi välja olulisemaid mõisteid. Teises peatükis tutvustati *Vernier*'i helivaljuse mõõtjat, tema omadusi ja kasutamisevõimalusi. Lisaks tehti sissejuhatust NXT-G programmeerimiskeskonda - tutvustati olulisemaid funktsioone selleks, et kolmanda peatükki ülesandeid oleks lihtsam lahendada. Kolmanda peatükki ülesannetele on erineva raskusastmega, mis on mõeldud NXT komplekti ja anduriga tutvumiseks.

Iga ülesanne loomisel kasutati ühtset malli:

- Ülesanne tase
- Ülesanne eesmärk
- Ülesanne lahendamiseks olulised vahendid
- Ülesanne püstitus
- Võimalik lahendus
- Võimalikud probleemid.

Õppematerjali loomisel arvestati sellega, et nii õpetajal kui ka õpilasel võib täielikult puududa programmeerimisoskused, seega iga funktsiooni ja ka ülesanne lahendust üritati põhjalikult ära seletada.

Antud materjali põhjal võib koostada uusi ja raskemaid ülesandeid, mida võib luua antud töös kirjeldatud ülesannete kombineerimisel.

Kasutatud kirjandus

1. Tenno, Toomas. *Urimuslik õpe*
<http://www.ut.ee/et/294390> – viimati vaadatud 12.05.2014.
2. Teadusbuss
<http://www.teadusbuss.ee/teadusbuss/avaleht> – viimati vaadatud 12.05.2014.
3. Kooliroboti projekt.
<http://www.robootika.ee/lego/projekt/> – viimati vaadatud 12.05.2014.
4. Rakett 69
<http://www.rakett69.ee/> – viimati vaadatud 12.05.2014.
5. E.Pärtel. *Füüsika VIII klassile*. Tallinn : “Koolibri”, 2000. lk.149-161
6. A. Sillart I. Eiskop. *Akustika ja helitehnika*. Tallinn : “Valgus”, 1988. lk. 7, 17
7. Wikipedia. *Tuning Fork*, 24.03.2013.
http://en.wikipedia.org/wiki/Tuning_fork – viimati vaadatud 12.05.2014.
8. Sibul, S. *Kõne - ja kuulmiselundite füsioloogia ja pataloogia*. Tartu : “Tartu Ülikooli Kirjastus”, 1997. lk.49-51
9. С.В. Громов, Н.А. Родина. *Физика. 8 класс*. Москва : “Просвещение”, 2002. lk. 176-178,184
10. Joonis “Helihark”.
<http://et.wikipedia.org/wiki/Pilt:Stimmgabel90degree.jpg> – viimati vaadatud 12.05.2014.
11. Joonis „Kõneelundid”
http://www.why.gr/MediaFolder//A5A0586E7D43F2488866BF45C44BF4A9/2081_L.jpg
– viimati vaadatud 12.05.2014.
12. Joonis “Kopsud”.
<http://blog.tauedu.org/wp-content/uploads/2012/05/lungs.png> – viimati vaadatud 12.05.2014.
13. Шебалин, О.Д. *Физические основы механики и акустики*. Москва : “Высшая школа”, 1981. lk.223-226, 231
14. Perelman, Jakov. *Huvitav füüsika*. Tallinn : “Valgus”, 1985. lk.229-232
15. Wikipedia. *Sound intensity*, 9.05.2013.
http://en.wikipedia.org/wiki/Sound_intensity – viimati vaadatud 12.05.2014.

16. Joonis “Kaader filmist Uus Gulliver”.
http://ic.pics.livejournal.com/kinoumerlo/42962094/30019/30019_640.jpg – viimati vaadatud 12.05.2014.
17. Chedd, Graham. *Sound: From communication to Noise Pollution*. Garden City, N.Y: “Doubleday”, 1970
 Грэхем Чедд. *Звук*. Москва : “Мир”, 1975, lk 9-16, 165-171.
18. А.П Ефимов, А.В Никонов, М.А Сапожков, В.И. Шоров. *Акустика: Справочник*. Москва : “Радио и связь”, 1989. lk 14-15.
19. G. I. Sokol, O. M. Duplischeva, T. A. Rybalka. *Influence of the sound and infrasonic acoustic vibrations on the living organisms*, 2009. lk 18
http://uenj.cv.ua/Ecology_and_noospherology_2009_20_3-4/Sokol.pdf – viimati vaadatud 12.05.2014.
20. Хорбенко, И.Г. *Звук, ультразвук, инфразвук*. Москва : “Знание”, 1986. lk 176-178, 184.
21. Joonis “Äike”.
<http://www.uer.varvar.ru/images/erheben1.jpg> – viimati vaadatud 12.05.2014.
22. Joonis “Kajalood”.
<http://images.shipstore.com/ss/images/low/low11070.jpg> – viimati vaadatud 12.05.2014.
23. Joonis “Müra mõju”.
<http://www.shumanet.com.ua/images/img-need/stophthenoise.jpg> – viimati vaadatud 12.05.2014.
24. Vernier. *Sound Level Meter*.
<http://www.vernier.com/files/manuals/slm-bta.pdf> – viimati vaadatud 12.05.2014.
25. NoiseMeters Inc, *Time Weightings - Fast, Slow or Impulse?*
<http://www.noisemeters.com/help/faq/time-weighting.asp> – viimati vaadatud 12.05.2014.
26. LEGO. *LEGO® MINDSTORMS Education NXT User Guide*, 2006
http://cache.lego.com/r/education/-/media/LEGO%20Education/Home/Downloads/User%20Guides/Global/MINDSTORMS/ts.20101019T110252.9797_LME_UserGuide_US_low.pdf – viimati vaadatud 12.05.2014.
27. Joonis “LEGO Mindstorms standardkompleki kuuluv heliandur”.
http://www.brickpicker.com/images/set_images/brickpicker_set_9845_1.jpg – viimati vaadatud 12.05.2014.

28. Jalakas, Siim. *LEGO MINDSTORMS NXT komplektiga ühilduv soolsuse andur*. Tartu : Tartu Ülikool, 2012.
29. Vernier. *Vernier Sensor Block*.
<http://www.vernier.com/engineering/lego-nxt/vernier-sensor-block/> – viimati vaadatud 12.05.2014.
30. I. Meho, H. Uibo, J. Jaeger, J. Kiho, R. Palm, A. Peder, U. Puus, P. Salumaa, E. Tõnisson. *Programmeerimise praktikumid. Algklassid*. Tartu : Tartu Ülikool. Arvutiteaduse Instituut, 2003.
31. Kell, James Floyd. *LEGO MINDSTORMS NXT-G Programming Guide*. s. l. : “Apress”, 2010.

Vernier Sound Level Meter

Bachelor's thesis (6 ECTS)

Angelina Kozina

Summary

The aim of this bachelor thesis is to create a comprehensive user manual about the Vernier sound level meter in Estonian and to introduce various possibilities to use the sound sensor with LEGO Mindstorms NXT kit.

This thesis consists of three chapters. The first chapter provides a brief overview of the concept of the sound and other related definitions. In the second chapter the author focuses on the Vernier's sound level meter and compares it to the basic NXT sound sensor. Author also describes how to connect the sensor with LEGO Mindstorms NXT set and use it in the NXT-G programming environment. The third chapter consists of five practical exercises for the sound level meter and the NXT set. There are two simple exercises to introduce NXT-G programming environment and three for collecting and analyze sound level meter output data. Every exercise is accompanied with a set of instructions, a list of requirements, a possible solution and a list of possible problems. All solutions are available as attachment to this thesis.

This bachelor thesis has a similar structure to other theses from the same field and can be used as teaching material in Estonian schools. In this thesis, author considers that both teachers and students may lack programming skills, so each function and solution has a detailed explanation. This work aims to make learning natural sciences more fun.

Lisad

Lisa 1. Ülesannete lahenduste failid

Tabelis 3 on välja toodud kolmandas peatükis olevate ülesannete lahenduste failid.

| Faili nimetus | Kirjeldus |
|-----------------------|---------------------------------|
| sissejuhatavYl.rbt | Ülesanne 3.1 lahenduse programm |
| dataLog.rbt | Ülesanne 3.2 lahenduse programm |
| andmeteKogumine.rbt | Ülesanne 3.3 lahenduse programm |
| maksimumiLeidmine.rbt | Ülesanne 3.4 lahenduse programm |
| keskmiseLeidmine.rbt | Ülesanne 3.5 lahenduse programm |

Tabel 3. Ülesannete lahenduste failid.

Lisa 2. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina **Angelina Kozina** (sünnikuupäev: 12.10.1991)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „**Helivaljuse mõõtja**”, mille juhendajad on **Anne Villems, Taavi Duvin**,
 - 1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus/Tallinnas/Narvas/Pärnus/Viljandis, **12.05.2014**