

TARTU ÜLIKOOL  
LOODUS- JA TÄPPISTEADUSTE VALDKOND  
Arvutiteaduse instituut  
Infotehnoloogia mitteinformaatikutele magistri õppekava

Paul Elberg

Arduino näitel põhinev riistvara  
programmeerimise valikkursus  
gümnaasiumiastmele  
Magistritöö (15 EAP)

Juhendaja: Alo Peets, MSc

Tartu 2020

# Arduino näitel põhinev riistvara programmeerimise valikkursus gümnaasiumiastmele

## Lühikokkuvõte:

Magistritöö käigus valmis algajatele mõeldud eestikeelne õppematerjal „Arduino näitel põhinev riistvara programmeerimise valikkursus gümnaasiumiastmele“. Valikkursuse õppematerjal on mõeldud neile, kellel puudub nii elektroonika kui ka programmeerimise varasem kogemus. Valikkursuse peatükkides antakse praktiline sissejuhatus riistvara programmeerimisse toetamaks kestva huvi teemast. Materjalide koostamisel on silmas peetud nende jõukohasust; peatükke on testitud gümnaasiumiastme vanusegrupist nooremate õpilaste peal. Magistritöös kirjeldatakse nii valikkursuse peatükkide kirjutamise kui ka testimise käigus esinenud probleeme ning antakse ülevaade neile leitud lahendustest.

Materjalid on leitavad veebiaadressil <https://courses.cs.ut.ee/t/nooredkoodi/Arduino/Arduino>

**Võtmesõnad:** Arduino, riistvara programmeerimine, valikkursus, õppematerjal.

**CERCS:** P175 Informaatika, süsteemiteooria, S281 Arvuti õpiprogrammide kasutamismetoodika ja pedagoogika

## An Arduino-Based High School Supplementary Course in Hardware Programming

### Abstract:

In the course of studies for this Master Thesis “An Arduino-Based High School Supplementary Course in Hardware Programming” was completed. This supplementary course has been aimed to teach hardware programming to persons with no prior experience in electronics or programming. The chapters of the supplementary course provide a practical introduction to hardware programming to promote sustained interest in the subject. A key concern throughout this work has been to achieve a material suitable to the entry level abilities; the chapters have been tested on pupils younger than the age group of gymnasium students. The author provides an account of the problems encountered as well as the solutions provided thereto in the course of writing and testing the chapters of the supplementary course.

Created materials are available in Estonian at <https://courses.cs.ut.ee/t/nooredkoodi/Arduino/Arduino>

**Keywords:** Arduino, hardware programming, supplementary course, study material

CERCS: P175 Informatics, systems theory, S281 Computer-assisted education

# Sisukord

Sissejuhatus	3
1. Arduinol põhineva riistvara programmeerimise valikkursuse vajalikkus	5
1.1. „Noored Koodi“ rollist riistvara programmeerimise õppematerjalide loomisel	5
1.2. Arduino olulisus riistvara programmeerimisel	5
2. Õppematerjalides kasutatava riistvara valik	7
2.1. Arduino stardikomplekti kasutatavus kursuse peatükkide läbimisel	7
2.2. Täiendavad riistvara vajadused	8
3. Õppematerjalide loomise protsess	10
3.1. Materjalide loojad	10
3.2. Arduino õppimiseks vajalikud materjalid eesti ja inglise keeles	10
3.3. Mõttekäik ja meetodika valikkursuse koostamisel	13
3.4. Eesti ja inglise keeles elektroonikast rääkimise probleematika	21
3.5. Värvikood takistitel ja juhtmetel	22
3.6. Näidistund—LCD ühendamise keerukusega toimetulek	23
4. Peatükkide testimisel ilmnenuid probleemeid ja nendega arvestamine	26
4.1. Õpilastele sobiva aja leidmine	26
4.2. Peatükkidesse testimise järel tehtud kohandused	27
4.3. Arduino UNO eelistamisest muudele kontrolleriplaatidele	38
4.4. Peatükkide testimise järgsed tähelepanekud ning jätku-ideed edaspidiseks tööks	42
5. Kokkuvõte	45
6. Viidatud allikad	47
Lisa 1. 11. peatükk—LCD kasutamise näide	50
Litsents	56

## Sissejuhatus

21. sajandil sõltub Eesti konkurentsivõime olulisel määral info- ja kommunikatsioonitehnoloogia (IKT) erialade oskuste tasemest Eesti ühiskonnas. Seega on ootuspärane, et oleme riiklikes arengukavades järjest enam tähelepanu pööramas tehnilistele oskustele ja teadvustame ka üldteadmisi ning koostöövajadusi.

Nõudluspõhise teadus- ning arendustegevuse ja innovatsiooni ergutamise esmase meetmena, ent ka meetmena IKT ning teiste võtmetehnoloogiate koostoimeliste mõjude suurendamiseks loetleb „Eesti 2020 tegevuskava 2020-2021“ IKT sektori arenguprogrammi koostamist; üheks osaks sellest on digipööre. [1] „Eesti Elukestva õppe strateegia aastaks 2020“ näeb ette elukestva õppe võtmenäitajate osas, et digipädevusega inimeste osakaal 16-74 aastate rühmas on tõusmas 65% pealt 80%-le. [2]

2011. aastal vastuvõetud „Gümnaasiumi riiklik õppekava“ ning „Põhikooli riiklik õppekava“ käsitlevad digipädevust ühena kaheksast võtmepädevustest. [3, 4] „Digipöörde programm 2018-2021“ toob välja vajaduse pöörata tähelepanu õppevara kättesaadavaks tegemisele: „Lisaks digipädevusele kui infoühiskonnas edukaks toimimiseks vajalikule oskusele on digipöördega ette nähtud ka süvemate IT oskuste arendamine – tehnoloogiline kirjaoskus ehk oskus tehnoloogiat ise luua. Kuigi u 70% üldhariduskoolides pakutakse õpilastele võimalust osaleda IT huviringis, ei ole õpe siiski kõigile õpilastele kättesaadav. Digipöörde programm seab eesmärgi, et IT õpe peab olema kättesaadav kõigile õpilastele. On oluline, et õpilastel oleks kiiresti muutuvast infoühiskonnas piisav arusaam IT kui lahenduste loomise ja rakendamise võimalustest. Automatiseerimise ja tööstuse digitaliseerimise mõju on väga oluline ning tehnoloogilise kirjaoskuse õppe jõustamise eesmärk on kasvatada õpilastes selle valdkonna eestvedajaid.“ Samas dokumendis juhitakse tähelepanu ka parandamist vajavatele momentidele: “Vastavalt õpilaste digipädevuse mudelile kohandati ka HITSA kaudu pakutavate täienduskoolituste sisu. 2017. aastal toetas HITSA koolitustega enim neid koole, kes digiteemalistes tegevustes aktiivsed ei ole olnud. Ministeriumi poolt HITSAle seatud tingimus oli, et koolitused peavad olema koolide vajadustest ja õpetajate oskuste tasemest lähtuvad. Sellegipoolest on valdav tagasiside, et õpetajate digioskused ei ole piisavad.” [5]

Paljudest IKT valdkonda edendama mõeldud kirjalikest materjalidest mainiks siinkohal veel SA Poliitikauuringute Keskus Praxis poolt tehtud uuringut. Praxise uuringus pakutakse välja soovitusi haridussüsteemile: eraldi on välja toodud vajadus pöörata suuremat tähelepanu IT hariduses üldpädevuste ning koostöövalmiduse arendamisele. „Üldpädevuste (nt suhtlemis-,

esinemis-, enesejuhtimis-, projektijuhtimisoskus) arendamise ning praktilisemate teadmiste ja oskuste arendamisega tuleb selgelt enam tegeleda. See on võimalik läbi praktikasüsteemide tõhustamise, aktiivõppemeetodite aktiivsema kasutamise (projekti-põhine õpe), tihedama koostöö erialade vahel (nt majanduse, inseneeria vms ja IT valdkonna õppurite koostööprojektid konkreetsete ettevõtluses esinevate probleemide süstemaatiliseks lahendamiseks) ja vastavalt määratletud õpiväljundite seadmise.“ [6]

Eesti olukorra ja Eesti ees seisvate ülesannete kaardistusest seega saame ka detailiseerida konteksti, millesse asetub valikkursuse loomine riistvara programmeerimise õpetamiseks gümnaasiumiastmele. Vajadus eesmärgipäraste materjalide järgi on ilmne. Põhirõhuga tehnoloogilise sisuga teadmistele kaasneb ühtlasi ka vajadus soodustada probleemilahendusoskuste arengut—orienteeritust meeskonnatööle ning ladusat suhtlusoskust.

Neid eesmärke püüab autor aidata lahendada magistritöö käigus loodud õppematerjaliga „Arduino näitel põhinev riistvara programmeerimise valikkursus gümnaasiumile“.

Magistritöös antakse ülevaade õppematerjalide koostamisel järgitud põhimõtetest ja probleemidest, mis mõjutasid valikkursuse materjalide koostamist; seejärel antakse ülevaade nende materjalide testimise tulemusena laekunud tagasiside kohta ning kirjeldatakse sellele tagasisidele tuginenud täienduste ja muudatuste abil materjali korrigeerimist.

**Töö esimeses osas** antakse ülevaade projekti „Noored Koodi“ rollist riiklike programmide sisuga täitmisel, Arduino olulisusest riistvara-alases väljaõppes ning seonduvatest võimalustest.

**Teises osas** tutvustatakse riistvara valikuvõimalusi ning vajadusi peatükkide lõikes.

**Kolmandas osas** kirjeldatakse õppematerjalide loomise protsessi tulemusena saadud peatüki-kirjeldusi.

**Neljandas osas** vaadeldakse valikkursuse jaoks koostatud õppematerjali testimist ning testimise käigus õppematerjali peatükkides tehtud muudatustest.

# **1. Arduinol põhineva riistvara programmeerimise valikkursuse vajalikkus**

Sissejuhatuses andsin ülevaate riiklikest dokumentidest, mis puudutavad IT õppe arendamist. Suuniste sisuga täitmisel on olnud aktiivne Tartu ülikooli arvutiteaduse instituudi projekt „Noored Koodi“. Projekti eesmärgiks on reaalse panustamisega toetada õpilaste IKT teadmiste arengut, seeläbi praktiliste sammudega aidata kaasa ka Eesti arengueesmärkide täitmisele.

## **1.1. „Noored Koodi“ rollist riistvara programmeerimise õppematerjalide loomisel**

„Noored Koodi“ korraldatud on paljud Eesti koolides läbiviidavad töötoad ning huvilaagrid. „Noored Koodi“ on ka initsiaatoriks rea valikkursuste õppekavade koostamisel (näiteks veebilehtede loomise õppekava, Linuxi (Raspberry Pi) õppekava jt. Seega on olnud loomulik, et ka selle töö aluseks oleva Arduinole põhineva valikkursuse esialgne raamistik on alguse saanud „Noored Koodi“ poolt välja käidud valikkursuse ideest. [7]

„Noored Koodi“ poolt indikeeriti vajadust 32 koolitunni mahus Arduino näitel riistvara valikkursuse programmi loomiseks. Mitte-informaatikute täiend-magistri programmi omandavale autorile oli aga kõigepealt vaja teemasse sissejuhatus. „Noored Koodi“ on autorile võimaldanud assisteerivas rollis osaleda riistvara programmeerimist puudutanud koolitustel ning sellele praktilisele kogemusele tuginedes on osutunud ka võimalikuks antud valikkursuse programmi valmis kirjutada ning ka iseseisvalt selle peatükke pilootida.

## **1.2. Arduino olulisus riistvara programmeerimisel**

Riistvara programmeerimise õpetamisel on Arduino näol tegemist olulise vahendiga ainese selgitamisel ning läbimängimisel. Arduino loojate eesmärgiks oli saavutada rahaliselt mittekallis kontrollerplaat, mis oleks kergesti programmeeritav ning mida iseloomustaks nii käsitlemise mugavus kui ka töökindlus ning teatav universaalsus, kõigele lisaks ka vabavaralikus. Kuna Arduino projekt kujunes edukaks, siis hakkas Arduino arenema paljudes erinevates suundades—nimelt on legaalne kasutada selle vabavaralise platvormi kloone või ka ise toota kloone. Siinkohal aga peab täpsustama, et selliseid kloone Arduino kaubamärgi all turustada on keelatud. Vabavaraline lähenemine on taganud Arduino arendusplatvormile globaalse leviku; esile on kerkinud terve toodete-ökosüsteem ning suur hulk toetajaid kasutajaid

tegeleb omavahelise infovahetuse ning üksteise töödes esinevate huvitavate või keerukamaks osutunud seikade kommenteerimisega. Algajale riistvara programmeerijale on suur mõttekaaslaste kogukond aga võimaluseks saada tasuta nõu ja abi erinevatel riistvaraga ning programmiga töötamise juhtudel [8, 9, 10].

Arduino enda tooteliin on aja jooksul mitmekesisistunud: kasutatakse erinevaid protsessoreid, on tehtud väiksemaid kui Arduino UNO, on olemas kantavaid (*wearables*) ning on olemas ka tööstusele mõeldud versioone. [11] Seejuures on Arduinod kõik programmeeritavad Arduino programmeerimiskeskonna (IDE) abil: võiks näiteks väita, et Arduino algkursuse omandamise järel suudab õpilane põhimõtteliselt juba ka töötada Arduino Mega plaati sisaldava tööstusele mõeldud M-Duinoga. Oluline on tööstuses saavutada resultaadid võimalikult väikeste kuludega—M-Duino hind jääb 200 euro juurde. [12] Siiski nõuab M-Duino töötamine veidi tõsisemat ettevalmistust kui selle magistritöö raames loodud valikkursusel pakutakse.

Arduino õppetöös rakendamise kasuks räägib ka Arduino tehnoloogia rakendamine ArduSat projektis eesmärgiga õpetada huvilistele kosmosega seotud rakendusi. [13] Tegemist on vabavaralist ideoloogiat järgides kosmosega seotud arendusteemasid toetava projektiga.

Seega õppides riistvara programmeerima Arduino näitel võib õpilane olla veendunud, et on omandanud midagi, millele leidub mitmekesiseid rakendusvõimalusi, ning et on teinud oma esimesed sammud rahvusvaheliselt aktiivse ning kiirelt kasvava kogukonnaga [14] liitumisel. Seda arusaama on oluline rõhutada juhul, kui õpilastel kerkib esile küsimus, mis täpselt oleks sellise valikkursuse läbitegemise kasud pikemas perspektiivis. Arduino kasutamine eestikeelse valikkursuse õppematerjali loomiseks on seega õigustatud valik. Järgmises peatükis uurime detailsemalt Arduino stardikomplekti (*Arduino starter kit*) kasutamise võimalusi valikkursuse õpetamisel.

## 2. Õppematerjalides kasutatava riistvara valik

Riistvara valikul on tähtis, et kontrolleraiaati ja temaga ühendamiseks mõeldud muud riistvara oleks võimalik stabiilselt ilma suurema pingutuseta juurde hankida.

Autor on õppematerjali kirjutamisel-läbimängimisel kasutanud erinevaid Arduino tooteid. Kogemustele tuginedes soovitan Arduino UNO-t. Alljärgnevalt puudutame Arduino stardikomplektiga (*Arduino starter kit*) riistvara programmeerimise algõppega alustamist. Tõenäoliselt on algajal mugav omada Arduino stardikomplekti, ent juba antud valikkursuse materjali läbimängimiseks tuleb teha esimesi samme täiendava riistvara hankimisel.

### 2.1. Arduino stardikomplekti kasutatavus kursuse peatükkide läbimisel

Riistvara programmeerimise õpetamisega alustades on oluline, et igaks tunniks oleks olemas piisav riistvara. Ka on harjutuste sooritamiseks ning teadmiste kontrollimisteks valmistumisel õpilasele mugavam, kui tal on võimalik kasutada isiklikku riistvara. Seega vaatleme Arduino stardikomplekti võimalusi.

Eestis jaemüügist kättesaadav kallim stardikomplekt, millesse on juurde lisatud ka projektiraamat, annab mugava võimaluse töötada läbi nii antud projektiraamatus kui ka magistritöö käigus loodud valikkursuses sisalduvaid projekte. Valikkursuse peatükkidest (<https://courses.cs.ut.ee/t/nooredkoodi/Arduino/Arduino>) on Arduino stardi-komplektis sisalduva varustusega tehtavad peatükid 1-5; 8; 11-12; 23; 26. Peatükid 9 ja 10 on stardikomplektis sisalduvaga sooritatavad vaid potentsiomeetri osas. Seejuures peatükid 13; 16; 17; 20; 30; 31; 32—on võimalik põhisosas läbi teha ka vaid Arduinot ning USB ühenduskaablit omades: tegemist on kas kordamise või siis teadmiste kontrolli peatükkidega või on teemaks Fritzing abil skeemide joonistamise harjutus. Siiski, kui teadmiste kontrolli viiakse läbi praktiliste ülesannetega, siis tuleb ka stardikomplekti soetanutel täiendavalt kontrollida, kas kõik vahendid on olemas. Seejuures tuleb ära märkida, et õppematerjalis toodud peatükkides ei ole kasutatud stardipakis sisalduvaid diode, geel-filtreid, H-silda, optopaari, piezo-elementi, transistorit, kaldenurga andurit.

Vaid stardikomplektis sisalduva riistvaraga ei saa tugineda peatükkides 6-7; 14; 15; 18; 19; 21; 22; 24; 25; 27, 28, 29 ning juhtkangi (*joystick*) puudutavates peatükkides 9 ja 10 osades. Seega



on iga peatüki eel mõistlik kontrollida, kas vastav tarvik on olemas või tuleb see hankida eraldi. Tarvikute vajaduses orienteerumise lihtsustamiseks on iga peatüki alguses nimekiri vajaminevast.

## 2.2. Täiendavad riistvara vajadused

Alustuseks tuleb mainida, et mistahes riistvara valiku puhul on hea, kui on ühendamiseks olemas ka külgejoodetud piigiriba (*headers*). Vastasel korral tuleb need eraldi külge joota. Palju riistvara müüaksegi nii, et piigiriba on eraldi komponendina juurde antud ja tuleb enne esimest kasutamist külge joota. Põhjendatud on seda näiteks kulude madalal hoidmisega [15] ning suulistes kommentaarides on prevaleerinud põhjendamine transpordis elemendi kahjustamisest hoidumise vajadusega. Kui sobilik piigiriba puudumine on, jäägu igakordselt eraldi kaaluda enne ostu-otsuse tegemist. Nii koolitundide kui ka muus formaadis õpikohtumiste laduse huvides tasub aga eelistada juba töövalmis piigiribadega elemente. Probleemiks on algajale ka piigiribade jootmisel saavutada see, et ühendused oleks kasutatavad maketeerimislaual. Isegi kui ekspertidest kliendid eelistavad ilma külgejoodetud piikideta versioone, ometi kinnitavad populaarsemate riistvaraelementide müüginäitajad eelistuste kaldumist külgejoodetud piigiribadega varustatud versioonide kasuks. [16]

Eraldi on vaja soetada nii infrapunaseid LED-iseid kui ka RGB LED-iseid, infrapuna-andureid, valgustakisteid, juhtkange (*joystick*), ultraheliga kauguse mõõtmise andureid, õhurõhu andureid, temperatuuriandureid (DHT-22 või DHT-11); *bluetooth*-seadmeid; Digisparke, ESP32 arendusplaat. Tark on juurde hankida ka takisteid ning juhtmeid nii, et neid oleks mugav kasutada: et neid oleks piisavas koguses. Ei tee paha ka täiendavalt soetatud maketeerimislauad.

Mida ei müüda Arduino stardikomplektis, ent mis võiks olla õppetöö toetamiseks soetatud, on multimeeter. Soovitame klassikalist pealt pöörd-lülitiga versiooni—töövõtete harjutamiseks mõistlik. Vt multimeetrit ning Ohmi seadust käsitlev 15. peatükk. Kursuse lõppedes kahe ESP32 peatüki ladusa omandamise tarvis tuleb kontrollida, kas on olemas mikro-USB kaabel

Materjali testimisel on harva midagi katki läinud. Kõige sagedamini on liigse edasi-tagasi painutamistega lõpuks purunenud takistite otsad—ühendusviigud. Takistite osas olen teinud panuse rikkaliku takistivalikuga takistivihikutele: need võimaldavad ka kiirelt leida vajaliku väärtusega või sellele piisavalt lähedase väärtusega takistid. Takistite katkimineku osas on ka tähelepanek, et sagedamini esineb seda maketeerimislaua ebakvaliteetsuse korral—õpilane võib mingil hetkel alateadlikult siiski ka jõudu rakendada.

Test-tundide jaoks riistvara hankimisega on esinenud probleeme, seega ei tee paha korralikud tagavarad. Ka näiteks Arduino kontrolleraalplaate võiks olla rohkem—näiteks võib vaja minna neid puhkudeks, kui on soov valminud töödest midagi tööle jätta—näiteks CSV faili käsitleva 22. peatüki järel võiks olla huvitav mõni süsteem pikemaks ajaks tööle jätta. Sinihamba abil kahte Arduinot omavahel suhtlema panevas 25. peatükis oleme panustanud sellele, et kahe õpilase Arduinosid omavahel suhtlema pannes ei ole vaja kummalegi täiendavat Arduinot. Ent kui on soov 25. peatükki kodus iseseisvalt järgi korrata, siis on vaja ka teist Arduinot.

Kokkuvõtteks: kui riistvara on rohkem, on lihtsam. Ent alustada saab ka Arduino stardikomplektiga (*Arduino Starter kit*). Siinkohal tuleb mainida Arduino stardikomplekti õppimis-hasarti tekitavat mõju—tegemist on nägusa komplektiga, juba selle lahti pakkimine ja detailide uurimine on autori juhendatud noortele testijatele inspireerivalt mõjunud.

### 3. Õppematerjalide loomise protsess

Õppematerjali (<https://courses.cs.ut.ee/t/nooredkoodi/Arduino/Arduino>) loomine algas kõigepealt autori enda õppimisprotsessi kirjeldustega, millele umbes aasta jooksul korduvalt täiendusi lisati (programmi-näited, Fritzing skeemid, lingid lisamaterjalile). „Noored Koodi“ raames mulle antud juhiste vastavalt teemasid internetist saadaolevate ingliskeelsete materjalide põhjal [17] iseseisvalt läbi tehes tuli kõigepealt algaja rollis olles kirja panna kogemused nii, et oleks endal ülevaade ainesest (seosed eelnenud peatükkidega, seejärel töös oleva peatüki sisu lühiülevaade, vajalikud tarvikud ja riistvara, lahenduskäik ja tööde kirjeldused). Kasulikuks osutusid ka autori enda algajana tehtud vigade kohta tehtud ülestähendused.

Läbivaks ideeks oli saavutada materjal, milles sisalduks algajale vajalik—ning seda, mis tegelikult algajale vajalik on, tunnetab kõige paremini algaja ise. Materjalide läbimängimisele ning kirjeldamisele järgnes nende viimine sellisesse vormi, et õnnestuks anda esimesi tunde teistele algajatele—olles nüüd juba ise teemat selgitava õpetaja rollis.

#### 3.1. Materjalide loojad

Valikkursuse peatükkide testimisel selgus, et illustratiivset materjali võiks olla rohkem. Kõige aktiivsemaks testijaks on olnud autori enda tütre tütar. Autori juhendamisel on tema poolt tehtud peatükkides 1, 2, 3, 11, 19, 24 kasutatud fotod ning peatükis 20 tehtava töö näidisenäite kasutatav Fritzing-skeem. Muud pildid ja skeemid on autori enda tehtud. Mõistagi on vastutus kõigi magistratöö raames esitatud valikkursuse piltide eest autori kanda. Enamus pilte on tulnud teksti lisamisel lõigata parajasse mõõtu. Tekstiosa on üleni autori looming.

#### 3.2. Arduino õppimiseks vajalikud materjalid eesti ja inglise keeles

Kuivõrd Arduino edulugu rajaneb vabavaralisusel [18, 19], siis on igati ootuspärane, et internetist leiab piisaval hulgal informatsiooni konkreetseid projekte puudutava riistvara ning programmidega töötamise üksikasjade kohta [17]. Sellised ingliskeelsed materjalid, eriti aga Arduino kodulehel sisalduv [8, 9, 20] olid ka autori enda Arduino-oskuste omandamisel esikohal; õigete märksõnadega otsinguid tehes oli võimalik puuduv info kiirelt kätte saada.

Lisaks leidmise kiirusele ei olnud vähetähtis ka tõsiasi, et Arduino ametlikul kodulehel asuv usaldusväärne materjal on saadaval tasuta. Raha kulub Arduinot tundma õppides pigem riistvara soetamisele. Tasulistest õppematerjalidest hoidumist motiveerib ka tähelepanek, et tihti on võimalik enne ostu vaid osaliselt veenduda, millist sisu täpselt paber kandjal trükise kujul ostaksid [21]. Internetis väljaspool tunnustatud ametlikke kanaleid levitatavate materjalide kvaliteedi osas võib ette tulla kõikumisi nii tasuliste kui ka vabalt kättesaadavate korral. Seega on mõistlik raha kokku hoides eelistada seda informatsiooni, mille sisuga tutvumisele eeltingimusi ei esitata, eelkõige aga otsida Arduino ametlikult kodulehelt. See aga toob esile paljude noorte huviliste probleemid inglise keelega. Gümnaasiumi või põhikooli õpilastele Arduino tundi andes on mõistlik seda ikkagi teha eestikeelsete materjalidega.

Juba olemasolevat eestikeelset materjali iseloomustab kas tasulisuse või muu sarnase mõjuga põhjusel materjalile ligipääsu piiratus—näiteks võib mõningase netis sisalduva vihje põhjal eeldada, et õppematerjalid on olemas, ent neile on ligipääs näiteks vaid antud koolis või kursusel õppijatel-õpetajatel [22]. Selline lähenemine, mis ei jaga materjali huvilistega, ei ole kooskõlas Arduino kogukonda iseloomustava mõtteviisiga. Toetudes enda poolt algajana tehtud valikute näidetele väidaks autor, et mitte-vabavaralised materjalid on enamusel juhtudel mõistetud jääma kõrvale aktiivsest kasutusest.

Eesti keeles kirjutatud ning seejuures vabalt internetist kättesaadavatest materjalidest (nt leitav otsisõnadega 'Arduino õppematerjalid') on selle magistritöö kirjutamise hetkel esile tuua TÜ Arvutiteaduse Instituudis 2016 valminud Selena Lubi bakalaureusetöö, millest algaja leiab eesti keeles piisavad instruktsioonid alustamiseks, ent mille teemade valik on mõeldud lühema kursuse jaoks kui seda on antud teeside aluseks oleva valikkursuse tarvis koostatud õppematerjal. [23] 2017 on valminud Marvin Marani töö GSM mooduliga Arduino baasile loodud telefonist. [24] Mittealgajatele mõeldult on githubist leida Lauri Võsandi poolt IT-kolledžile koostatud materjalid [25], mille alusel oleks võimalik teha jätkukursust juba algajale mõeldud õppe läbinutele.

Tasulise eestikeelse materjalina, mida õnnestus magistritöö kirjutamisel eestikeelsete materjalidega tutvumise tarvis soetada, on autorile paber kandjal kättesaadavaks osutunud Raivo Selli ning Kaupo Raidi 2017 välja antud „Arduino projektid alustajale“. Tegemist on hea õpimaterjaliga, ent selle alusel oleks keerukas anda tunde varem riistvaraga ja programmeerimisega mitte kokku puutunutele. Ka on materjali pealkirjast võimalik eeldada, et ilmselt ei ole mõeldud (*eelteadmisteta*) *algajale*—õppematerjali pealkirjas sisaldub sõna *alustajale*. Ka õppevahendi keelekasutus toetab arusaama, et tegemist on Arduinot tutvustava

õppematerjaliga, mis orienteeritud pigem arvutite maailmas juba kodus olevale huvilisele—info on detailine. Näiteks on lugedes kasulik teada, mis on AVR ning ARM arhitektuuri vahe. Õpik on seejuures eelkogemusega edasijõudnuile igati heaks teejuhiks, hästi kirjutatud ning võimaldab kiirelt teemasid visualiseerida. Alles hiljuti ise algaja rollis olnuna aga leian, et kindlasti oluaksin takerdumas näiteks Arduino arenduskeskkonda puudutavas kolmandas peatükis juba esimesse lausesse: „Arduino integreeritud programmeerimiskeskond (IDE) on tarkvaraplatvormist sõltumatu rakendus, mis on tuletatud *Wiring* ja *Processing* raamistikest.“ [26] Olgu aga siinkohal üle kinnitatud, et algaja tasemest edasi jõudnud lugejale on Raivo Selli ja Kaupo Raidi „Arduino projektid alustajale“ igati väärtuslik materjal. Autor leidis sellest kinnitust arusaamale, et Arduino on algajale kõige sobilikum arendusplaat [26]. Seda raamatut soovitan algajal lugeda Arduino näitel riistvara programmeerimise kursuse lõpuosas jões.

Erinevalt väljatoodud igati kasulikest eestikeelsetest materjalidest sisaldab autori koostatud eestikeelne materjal suuniseid iseseisvalt ingliskeelsest Arduino-infost vajaliku leidmiseks ning puudutab ka riistvara ingliskeelsetena levitatavate andmelehtede lugemise teemat.

Autori isiklik eesmärk Arduino kohta ingliskeelses internetis saadaolevate materjalide abil riistvara programmeerimist õppima asudes oli järk-järgult õppematerjalidest välja kasvada. Sama eesmärki teadvustaks ka kõigi teiste huviliste jaoks. Oluline on, et algaja kogeks eneseusalduse tõusu: veenduks selles, et saab järjest paremini hakkama järk-järgult mahukamaks muutuvate harjutustega ning orienteeruks iseseisvale toimetulekule allikatega. Enamus allikaid maailmas aga jäävad ingliskeelseks. See realiteet seab materjali eesti keeles esitamisele oma raame; autori arvates tingib see vajadust eesti õppekeelega valikkursusel teatud osas anda materjali kohta viiteid pigem ingliskeelsete otsisõnade abil.

Erinevalt autorile kättesaadavatest eestikeelsetest kirjalikest allikatest olen enda koostatud materjalidesse lisanud ka multimeetrit käsitleva peatüki ning selle lõpus Ohmi seaduse toimivust näidates sisse toonud mnemotehnika. Ohmi seaduse meeldejätmise kergendamiseks olen rakendanud Warwicki ülikooli õppejõu Neil Storey poolt 2009 välja antud õpikust „*Electronics: A Systems Approach*“ võetud „virtuaalse kolmnurga“ meetodit. [27] Kui oleme kindlustanud, et valemiga õpilasel probleemi pole, siis saame demonstreerida multimeetriga tehtava mõõtmise matemaatilist kontrollitavust.

Allikate kohta öeldut kokku võttes tuleb ilmselt tõdeda, et õpikute maailm jääb paratamatult tasuliseks. Tasuliste allikate osas on alust arvata, et need ka vähem kasutust leiavad—kuna ilma raha kulutamata ei ole võimalik veenduda, kas need näiteks vastavad algaja tasemele või

vastavad need edasijõudnud alustaja vajadustele. Eestikeelsete materjalide osas tundub seega kõige suurem kasutegur saavutatav läbi vabalt levitatavate materjalide, mis mõeldud kõige noorematele huvilistele ning mis on loodud ja testitud eesmärgiga neid aktiivselt kasutada ning arendada õpetamise käigus. Õeldu aga ei tähenda, et paralleelselt kättesaadavaks tegemisega internetis ei peaks soovijatel olema võimalik sama materjali soetada ka paber kandjal.

### **3.3. Mõttekäik ja metoodika valikkursuse koostamisel**

Palju on räägitud, et Eestis on puudus IKT tööjõust. [28;29;30] Praxise kodulehelt [44] tööjõuvajaduste prognoosist aga leiame, et „IKT erialast ettevalmistust nõudvate ametikohtadel on sektoris vaja töötajaid aastani 2020 juurde sõltuvalt sektorikasvustsenaariumist 2661 – 4456 inimest”. Kusjuures “kõrgharidusega lõpetajaid on vaja suurusjärgus 4200 – 5600, kuid pakkumine jääb samal ajal aga 4550 lõpetaja tasemele” ning “kutseharidusega uute töötajate nõudlus ulatub 2500 – 2900 töötajani, samas pakkumine on suurusjärgus 4000 lõpetajat” ning “tööandjate hinnangul ollakse olemasoleva tööjõuga üldjoontes rahul, mis on seoses ka sellega, et tööandjad on töötajaid ise paljuski oma ettevõttes välja koolitanud, mis kirjeldab omakorda tõsiasja, et valmis ja piisavate teadmiste ning kogemustega IKT spetsialiste pole lihtne leida”. Praxise andmetel “töötajate puhul hinnatakse üha enam üldpädevust – suhtlemisoskust, probleemilahendamisoskust, meeskonnatöö ja juhtimisoskust koos erialaste oskustega.” Tõdetakse, et „tööandjate hinnangul ei ole IKT spetsialistide ettevalmistuse kvaliteet õppurite arvu kasvule järgi tulnud ning kriitikat pälvib ka IKT-alast haridust pakkuvate õppekavade lõpetajate üldpädevuste tase”. Praxise uuring soovitab, et “kõrghariduse osas peab õppurite arvu rõhuasetus olema tarkvaraarendusega seotud spetsialistide ettevalmistusel, üldpädevuste arendamine, tihedam koostöö erialade vahel, ja ettevõtjate ning haridusasutuste vahelise koostöö parandamine” ning “kutsehariduse osas tuleks üle vaadata IKT-erialase kutsehariduse pakkumine nii mahtude kui eelkõige õppe kvaliteedi osas”. Nii kutseharidusele kui ka kõrgharidusele paratamatult eelneva üldhariduse osas leitakse, et “tuleks arendada üldpädevuste omandamist ning suutlikust ise õppida, et elukestva õpihoiaku peale täiendavat baasi ehitada”. [44] Erinevalt kohati esinevatest kurtva sisuga materjalidest IT sektoris tööjõu puuduse üle [31,32] annab Praxise uuring [6] ette konstruktiivsed suunised—ning on seeläbi ka heaks lähtealuseks õpiku või valikkursuse programmi koostamisele.

Valikkursuse loomisel sai lähtunud mõttest, et õpilastele tuleb anda samm-haaval keerukamaks osutuvaid baastadmisi riistvara programmeerimisest nii, et tekiks seos millegi käegakatsutavaga—valmiks riistvara projektid, millega on võimalik ka improviseerida: mängida, katsetada, täiendada, projekte omavahel kombineerida.

Võimalike poolelijätmist vältimiseks peab kogu kursuse jooksul jälgima, et esimeste peatükkide vaimustus ja hasart kestaks kuni valikkursuse lõpuni. Vaimustuse alalhoidmise tarvis saame ära kasutada õppeaspekte, mis sobivad ka eespool viidatud Praxise uuringu soovitusetega.

Sõnastaksin need täiendavad õppeaspektid järgmiselt:

- Üldpädevuste omandamiseks eelduste loomisel suuname uudishimule orienteeritud praktiliste surfamisharjutustega õpilast ise otsima lahendusi internetist, lisaks viitame ka seostele põhiainetes omandatavate teadmistega (matemaatika ja inglise keel, füüsika, ent ka asjasse puutuv avastuste ajalugu);
- Suhtlemisostkust, probleemidele lahenduste leidmise oskust ning meeskonnatööd esile kutsuda on võimalik ka lausa nii, et lubame teatud koostööd isegi ka eksamil—kuna käsitleme eksamit mitte niivõrd teadmiste kontrollina vaid kui viimast õppimisvõimalust enne valikkursuse lõppu.
- Esitlusostkust saame trennida ka eksamiks projekti esitlust korraldades.

Kuigi autori arvates on kõige olulisem ikkagi saavutada riistvara programmeerimise alaseid teadmisi ja vilumusi, on Praxise soovitusel välja toodud üldised aspektid sageli hilisemas tööelus määrava tähtsusega. Ka olemasolevast eestikeelsest inseneripedagoogika kirjandusest leiab soovitusel „keskenduda toetavale õppekeskkonnale ja kontekstipõhisele õpetamisele, tõsielulistele õpituatsioonidele ja tegevuspõhisele õppimisele ning õppima õppimisele. Õpilased peaksid oskama leida vajalikku informatsiooni ja selle õigsust hinnata“ ja et „areng kulgeb ikka ainult läbi protsesside, mida õpilane üle elab, ka läbi negatiivsete ning positiivsete emotsioonide“ [33] Nende põhimõtetele on arvestatud ka valikkursust koostades—nii teemadevalikul, nende peatükkideks vormistamisel, õpilaste peal läbiviidud testimistel ning töö viimases staadiumis õppematerjalidesse vajalikes aspektides muutuste sisseviimisel.

Infotehnoloogia õpieesmärk on saavutada, et õpilane peale valikkursuse läbimist on võimeline läbitöötatud riistvara abil iseseisvalt ja loominguliselt koostama toimivaid süsteeme ning neid muutma vastavalt oma vajadustele:

- A) õpilane valdab tarkvara loomist (seejuures on võimeline leidma ja ära kasutama juba olemasolevaid programminäidiseid, neid kohendama vastavalt oma töö vajadustele, kasutama teeki, oma tööd kommenteerima, korrigeerima, uuesti kommenteerima);
- B) õpilane on omandanud töövilumused aines läbivõetud riistvaraga ja teab, kuidas vajaduse korral infot leida, oskab tööohutust ja tarvikute töökorras olekut puudutavaid detaile silmas pidada ning oskab siluda programmi vastavalt konstrueeritud süsteemi iseärasustele—on hakanud rõõmu tundma improvisatsioonist.

Praktikas tähendab eesmärgi saavutamine ka seda, et kui ei meenu lahenduste võimalused, siis ei lasta tekkida heitumise aistingul: õpilased oskavad leida internetist vajalikku informatsiooni, teevad info leidmise nimel ja mahajääjate edukuse nimel meelsasti koostööd, annavad üksteisele tehnilist nõu—oskavad kommenteerida kaaslaste vigu nii, et kritiseeritaval sellest kasu on.

### **Peatükkide sisuline ülesehitus**

Peatükid võime üldjoontes jagada a) teema-peatükkideks (aine selgitused ning illustreerivad materjalid, lisaks ka harjutused) ning b) kordamis-peatükkideks (läbivõetu kordamine ning esitlemine ning tagasiside võimaldamine—õpilasele õpiedukuse ning ja õpetajale õpetamise edukuse kohta). Samas peame arvestama, et selline jaotus on tinglik: kordamine esineb tegelikult kõigis peatükkides—need baseeruvad eelnenud peatükkides läbivõetud materjalil ning seega on kordamine sisse ehitatud igasse peatükki. Toome järgnevalt ära valikkursuse sisu jagunemise eesmärgiüksuste lõikes. Selguse huvides esitame jaotuse kuue tabelina.

Alustame ABC-ga. Selles peatükkide osas on need teemad, millele kogu järgnev üles ehitub. (Vt tabel 1.) Seejärel konsolideeritakse kolme peatüki abil teadmisi programmeerimisest, korratakse praktikas üle sisendi ja väljundi teema ning jõutakse multimeetri ja Ohmi seaduseni. (Vt tabel 2.)



**Tabel 1. Klassikaline ABC-osa, põhilised teadmised tuleb omandada jõukohases tempos**

Peatükk	Eesmärk
1. Sissejuhatus: millist Arduinot valida, programmeerimiskeskond Arduino IDE	Õpilase arvutis on töövalmis Arduino IDE
2. Töötamine Arduino IDE-ga, Arduino ühendamine arvutiga ja sisseehitatud LED-i vilgutamine	Õpilane oskab kasutada elementaarseid töövõtteid.
3. Vooluahela koostamine (LED, takistid, maketeerimislaud, juhtmed) ja värvikood	Õpilane on koostanud esimese riistvara projekti ja saanud selle toimima, mänginud programmiga ja riistvaraga erinevaid kombinatsioone katsetades. Õpilane oskab kasutada muutujaid.
4. LED-ide vilgutamine FOR tsükliga, lokaalmuutuja kasutamine	Õpilane on selgeks saanud FOR-tsükli kasutamise, tutvunud lokaal-muutuja kasutamisega
5. Lüliti ühendamine ja LED-i vilgutamine vastavalt lüliti asendile: IF/ELSE tingimuse kasutamine	Õpilane on selgeks saanud IF/ELSE tingimuste seadmise.
6. RGB LED-i ühendamine ja RGB võimaluste katsetamine analogWrite() abil	Õpilane on saanud esimese kogemuse digitaal-signaali ning analoogsignaali erinevustest ning oskab kasutada analogWrite() funktsiooni.
7. Valgustundliku takisti abil valguse mõõtmine ja kuvamine arvuti ekraanil—Serial monitor	Õpilane on saanud esimese kogemuse Serial monitori kasutamisest, ühtlasi saanud esimest korda toimima ka analoog-signaali lugemise (analogRead() funktsioon)
8. UART liidese abil andmete vahetamine Arvuti ja Arduino vahel, ASCII tabel	Õpilane on saanud selgeks Serial monitori kasutades Arduinoga suhtlemise ning teab, mis ühik on baud; on tutvunud asünkroonse versus sünkroonse side erinevuse põhiliste aspektidega.
9. Potentsiomeeter: töö-põhimõte, ühendamine, juhtkang (joystick)	Õpilane on saanud selgeks potentsiomeetri kasutamise ning seega oskab kasutada ka juhtkangi (joystick), eelmistest tundidest on meenutatud valgustundliku takisti tunnis õpitut.
10. Servomootori juhtimine—pulsilaiusmodulatsioon ja juhtkang (joystick)	Õpilane on tutvunud pulsilaiusmodulatsiooniga ning on koostanud süsteemi juhtkangiga kahe servo juhtimiseks. Kordab eelmist tundi, ent lisab kaks servot. Ühtlasi on tutvunud servode pöördeulatuse problemaatikaga.
11. Ekraani ühendamine ja teksti kuvamine 1602 LCD ekraanil	Õpilane on tutvunud LCD ekraani ühendamise ja selle tööle saanud. Peale selle peatüki materjali omandamist ei heiduta enam suur hulk juhtmeid.
12. USART ja 1602 LCD: Serial monitorist ekraanile!	Õpilane oskab Serial monitori abil kuvada infot LCD ekraanile ning ei pelga veateateid.

**Tabel 2. Järgnevas osas konsolideeritakse programmeerimise võtteid, korratakse sisendi ja väljundi kohta õpitut ning tutvutakse multimeetri kasutamisega**

Peatükk	Eesmärk
13. Kokkuvõtvalt Arduino IDE abil programmeerimisest	Õpilane on üle korranud programmeerimise töövõtteid ning nüüd teab, kust kohast nende ununemise korral asju üle vaadata. Peatükk on pikk, detailirohke—seega korrage raske meelde jätta, ent sobib kenasti juba läbitehtu kordamiseks ning hiljem programmeerimise töövõtete osas kiireks mälu värskendamiseks.
14. LED heleduse muutmine vastavalt keskkonna valgustugevusele	Õpilase ettekujutus ühe protsessi väljundi kasutamisest teise protsessi sisendina on saanud täiendust, paranenud on programmitekstide lugemise oskus ning on tekkinud soov eelnenud peatükkide ainest kombineerides improviseerida.
15. Ohmi seadus, multimeetri kasutamine	Õpilane oskab kasutada multimeetrit ning on jätnud meelde Ohmi seaduse—ühtlasi on saanud selgemaks, mis on mnemotehnika võimalused.

Järgnevalt peame aru saama, mil määral omandati materjal ning seejuures saame ka stimuleerida õpilaste endi aktiivsust aine omandamisel. Seega ei ole esimesele teema-

peatükkide osale järgneva teadmiste vahe-kontrolli ideeks mitte niivõrd panna hindeid vastavalt ilmutatud teadmistele—eesmärgipärasem on panustada eespool omandatu kinnistumisse läbi materjali taas-esitamise. Hea on teoreetilisel pinnal mõelda teadmiste kontrollimisest kui metakognitiivse strateegiale suunatud võttest stimuleerida õpilasi aine omandamisel. Metakognitiivne strateegia tähendab õppima õppimist: „Oma õpistrateegiate analüüsimist ja nende kohandamist nimetatakse metakognitiivseks strateegiaks, mis sisuliselt tähendab õppima õppimist. Sellised õppijad suudavad ise oma õppimist kontrollida ja hinnata.“ [33] (Vt tabel 3.)

**Tabel 3. Kordamine ja teadmiste kontroll**

Peatükk	Eesmärk
16. Kordamisküsimused	Loend teemadest on antud peatükis. (Kui on selline võimalus, siis õpetaja poolt koostöös õpilastega võib seda loendit täiendada, heade täienduste eest võiks hindamissüsteemi kasutamise korral anda häid hindeid.) Palutakse eraldi tähelepanu pöörata ohutuse kirjeldamisele (nt kuidas ei tohi ühendada LED-i). Kordamine viiakse läbi eelmiste tundide materjalist olulisi kohti koos õpilastega üle vaadates ning palutakse teha märkmeid. Märkmeid on lubatud järgmisel kokkusaamiskorral teadmiste kontrolliks tehtava kontrolltöö sooritamisel kasutada.
17. Teadmiste kontrollimine	Teadmiste kontroll viiakse läbi kirjalikult—sisuliselt kirjandi vormis. Kontrollimiseks vahetavad õpilased kontrolltöid omavahel. Seejärel tuleb igal õpilasel oma tööd esitleda suuliselt ja see oleks kõige raskem osa teadmiste kontrollist—kas õpilane suudab esitada oma teoreetilist kirjutist nii, et ka vähik kenasti asjast aru saab või asja ära unustanud kolleegile kõik kenasti meenub. Oponeerib kirjalikku osa kontrollinud õpilane.

Tiia Rüütman toob välja McKeachie (2006) õpistrateegiate süsteemi selle jaotumises kognitiivseteks, metakognitiivseteks ning ressursistrateegiateks. Seejuures „kognitiivsete strateegiate alla kuuluvad:

- Harjutamine – näiteks materjali kordamine, kopeerimine (ümber kirjutamine), olulise info kohta märkmete tegemine;
- Viimistlemine – seoste loomine uue ja varem õpitu vahel – parafraseerimine, kokkuvõtete tegemine, analoogiatega toomine, üldistamine, küsimuste esitamine ja küsimustele vastamine;
- Organiseerimise strateegiaks on põhiideede väljaselekteerimine, seoste laiendamine, illustreerimine ja skemaatiline kujutamine.“ [33]

Antud süsteem on heaks teoreetiliseks aluseks valikkursuse mõlema osa koostamisel. Antud valikkursuse kontekstis aga on osutunud oluliseks mõista, et töö raames ettenähtud ulatuses kirjeldatud materjal ei pruugi mahtuda ära 32 õpikohtumise sisse, eriti kui tegemist on klassikalise 45 minutit kestva koolitunniga. Seega tuleb järgnevas osas mõista, et peale ABC läbimist ei pruugi näiteks 32 õpikohtumiseks olla võimalik läbi käia kogu materjali.

Peale vahe-eksami seega jätkatakse kuni kursuse lõpliku ajalise piirini jõudmiseni eelnenud kordavate, ent siiski ka uut ainet sisaldavate teemade ja harjutustega.

Teise osa teemapeatükkide eesmärgiks on arendada improvisatsiooni. Õpetajale antakse siin ka valikuvõimalusi—olenevalt sellest, kuidas jagub ajalist ressursi ning kas on saada kõiki vajalikke tarvikuid. (Vt tabel 4.)

**Tabel 4. Teise osa teema-peatükid: võetakse edasi uut osa, ent kombineeritakse seda eespool omandatud teadmistega**

Peatükk	Eesmärk
18. LED juhtimine infrapuna-saatejaga ja –vastuvõtjaga	Õpilane tutvub infrapuna-LED-i ning infrapunaanduri kasutamise võimalustega ning „kirjutab maha“ vanalt televiisoripuldilt signaali, millega saab juhtida mingit süsteemi, mis teeb midagi—näiteks vilgutab LED-i. Ühtlasi mainime ka esimeste LED-ide kohta mõnda fakti ning õhutame õpilasi netist teema kohta lugema.
19. HC-SR04 ultrahelisensori ühendamine ja näitude kuvamine LCD ekraanil	Õpilane on saanud üle korrata LCD ühendamise ning on saanud kuvada ekraanile ultrahelisensoriga saadud kauguse mõõtmise tulemused: seega on olemas esimene kogemus enda tehtud elektroonilisest mõõteriistast.
20. Joonistame ise programmi Fritzing abil korrektse vooluringi skeemi	Õpilane on joonistanud oma esimese Fritzing-skeemi, seega on saanud täiendust võimekus kommunikeerida teistele oma projekti-ideid—näiteks lasta nooremal õel-vennal teha ära ühendused ning siis ette näidata, kuidas töötab programm. Soovitame võtta ette mingi lihtsama skeemi ja selle lõpuni joonistamise järel see skeem ka tööle panna—kui midagi sai algul valesti joonistatud, siis tuleb see Fritzing programmi kasutades ära parandada. Õpilane on võimeline oma tööd selgitama.
21. Sissejuhatavalt sünkroonsest ühendusest (I <sup>2</sup> C ning SPI) õhurõhu anduri BMP280 näitel	Õpilane on tutvunud asünkroonse ning sünkroonse side erinevuste mõistmise raames ka I <sup>2</sup> C ning SPI ühendusmeetoditega. Õhurõhu mõõtmine aitab teooriat praktilise näitega meelde jätta. Ühtlasi toome esile, miks on vajalik töötades märkmeid teha: peatükki läbides täidame harjutusena ühte tabelit—kui oleme selle valmis saanud, siis peaks olema paranenud ettekujutus viikudest ning viikude tähistest. Peatüki materjali omandamisel on kohane õpetaja tähelepanu võimalikele mahajääjatele. Peatüki omandamise järel õpilane mõistab, miks on vajalik teada erinevaid ühendusi ja oskab ka kommenteerida, mis on ühe või teise meetodi eelised ja puudused.
22. CSV ( <i>Comma Separated Values</i> ) faili info salvestamine	Õpilane harjutab iseseisvalt internetist materjali leidmist. Vajadusel aitab õpetaja. Eesmärk on kasutada netist tasuta kättesaadavat programmi andurilt loetavate andmete talletamiseks parema meetodiga kui on olnud seni kasutatud Serial monitor. Õpilane on saanud esimese kogemuse CSV formaadiga anduri andmete säilitamiseks ning saab seega läbi viia ka pikemat andmete jälgimist.
23. Kujundite kuvamine 1602 LCD ekraanil	Õpilane on üle korranud veelkordselt LCD ühendamise—selles enam ei eksi. LCD-l on õnnestunud ka esimeste enda tehtud sümbolite kuvamine. Tegemist on lihtsama sisuga peatükiga, mille saab ajanappuse korral jätta ka koduseks katsetamiseks.
24. Ekraani ühendamine ja teksti kuvamine 1602 LCD ekraanil	Õpilane on omandanud teksti kuvamise oskused ekraanil: osa tekstist on aga mõõtmise tulemus—antud juhul mõõdame näidisprojekti temperatuuri. Valminud on teine isetehtud elektrooniline mõõteriist.
25. Arduino ühendamine <i>bluetooth</i> seadmega	Õpilane oskab töötada <i>bluetooth</i> seadmega ning kahte omavahel suhtlema saada. Õpilane oskab töötada AT käsklustega ning teeb vahet <i>bluetooth</i> seadme olekul. Serial monitoriga töötamisel enam ei eksi, sest töövõtted hakkavad muutuma harjumuspäraseks.
26. Relee juhtimine	Õpilane oskab ühendada releed ning Arduinot kasutades juhtida nõrga pingega juhtvoolu abil suurema pingega vooluahela sisse ning välja lülitamist. Üle on teadvustatud ohutuse teema. Tunnis me ei kasuta aga 220V vooluvõrgu juhtimist.

Teise teema-peatükkide osa sees moodustavad eraldi mainimist vääriva alam-osa kolm viimast tundi, mille viime läbi mitte-Arduino riistvara Arduino IDE abil programmeerides. Selle osa erisus seisneb selles, et kursuse lõpus anda veel täiendav praktiline metakognitiivne kogemus— jõuame nimelt uuel keerukustasemel „tagasi algusesse“: laeme alla meile esialgu tundmatu riistvaraga seotud tarkvara ning jõuame seejärel, nagu Arduinogagi alustades—LED-i vilgutamiseni. Saame käsitleda neid tunde ka kui esimest märki algaja tasemelt väljajõudmisele. (Vt tabel 5.)

**Tabel 5. Teise osa viimases kolmes teema-peatükis programmeeritakse Arduino IDE abil mitte-Arduino riistvara**

Peatükk	Eesmärk
27. Digispark: netist tarkvara leidmine, installeerimine ja esimese lihtsa programmi töölepanek	Õpilane oskab Arduino IDE abil töötada ka mitte-Arduino seadmetega. Selles peatükis ei kasutata Arduino kontrolleraati, ent õpilane programmeerib probleemse „mälu pulka“, mis arvuti külge ühendatuks jäädes hakkab rikkuma Arduino IDE-s sel hetkel lahti olevat programmi. Saab selgeks ka see, et juhuslikku mälu pulka ei tohi ühendada oma arvuti külge.
28. ESP32: tarkvara ja lihtsa esimese programmi töölepanek	Õpilane oskab Arduino IDE abil töötada ESP riistvaraga.
29. ESP32: jõuame sisseehitatud LED vilgutamiseni	Õpilane on ESP32 abil vilgutanud LED-i. Õpilane on saanud veenduda, et iga uue ja algul keerukana tunduva kontrolleriiga tutvumist on mõistlik alustada klassikalise harjutusega—LED-i vilgutamisega.

Kursuse lõpetab iga osaleja poolt esitletud projekt, mille hindamises osalevad kõik kursuselased. Lubatud on igasugune fantaasia ja koostöö—ent peab mahtuma ajalikesse raamidesse ning olema kooskõlas ka muude reeglitega, mille kokkuleppimiseks võib kogu viimase osa sees küsida ka õpilaste ideid ja soove. Idee on taas rakendada teadlikku õppima-õppimist: et oleks õpilastele selge, et õpitakse ise ja endale ning et töötatakse enda arenguhuvist lähtudes. Seda viimast on ehk keerukam läbi mängida noorematega kui gümnaasiumiealised, ent autori hinnangul ei ole see võimatu—tasub näiteks küsida sisendit kontrolltöösse pandavate küsimuste osas ning häid küsimusi teadmiste kontrollis kasutada. (Vt tabel 6.)

**Tabel 6. Viimane võimalus kordamiseks ja sellele järgnev esitlus**

Peatükk	Eesmärk
30. Kordamine	Kordamisküsimused on tegelikult peatükkide pealkirjad—otsida üles eelmise kordamispeatüki läbimisel tehtud märkmed ja teha neile lisa. Koostöö on väga oluline, sest materjali on palju. Soovituslik on, et õpilased jagavad omavahel ära peatükid ning vahetavad omavahel märkmeid.
31. Teooria-esitlus	Iga õpilane esitleb lühidalt vähemalt ühe peatüki materjali. Hea esitluse eest lisapunktid. Kasutada võib kõiki materjale, ka valikkursuse peatükki—tegemist on tegelikult eelviimase õppimisvõimalusega. Kaaslased kuulavad esitlust oma märkmeid ja esitlust võrreldes ning ülestähendusi tehes.
32. Projektiesitlus/Eksam	Vähemalt kolme peatüki materjalist koosneva valmis ehitatud projekti esitlus iga õpilase poolt. Valitakse parim projekt ja parim projekti-esitleja. Lisapunkte annab kirjaliku esitluse olemasolu—koos Fritzing skeemiga.

## Teemapeatüki struktuur

Teemasid ja teadmiste kontrolli puudutavad peatükid moodustavad terviku, mis arvestab vajadusega korrata õpitut ning jõuda tehnilise vilumuseni, et oleks võimalik saavutada ka improvisatsioonivõimekust. Kursuse tervik peab oma osades olema õppurile meeldiv—valikkursus ei ole põhiline ja seega ei tohi olla stressirohkem põhiainetest.

Samas on stressi eest ka võimalik kaitsta harjumuspärase rutiiniga.

Tekitame sellise tööritmi peatükist-peatükki korduva struktuuriga:

- a) Selleks et esile tuua teemade seotust, sisaldab enamus peatükke ka viidet eelnenud õppetundidele. Tähistatud on see osa \*.0. abil.
- b) Teema kohta sissejuhatavad selgitused järgnevad igas teema-peatükis tähistatud \*.1. osas.
- c) Edasi tuuakse välja vajalikud tarvikud osas \*.2. —reeglina on mustaga tähistatud see vajalik, mis kordub tunnist tundi ning punasega see riistvara, mis antud peatükis võrreldes eelmisega täiendavalt juurde võtta tuleb.
- d) Teema enda arendus ja harjutused sisalduvad osas \*.3. Samas näiteks potentsiomeetrit ja juhtkangi (*joystick*) puudutav tund annab juurde ka \*.4.—teema jaguneb kaheks—potentsiomeetri ning seejärel juhtkangi osadeks.
- e) Koduste ülesannete osa on reeglina paigutatud kõige lõppu—samas on viiteid kodusele ülesandele võimalik leida ka peatükki kirjeldavast osast—näiteks on 21. peatükis (*Sissejuhatavalt sünkroonsest ühendusest (I<sup>2</sup>C ning SPI) õhurõhu anduri BMP280 näitel*) toodud ära koduse ülesandele viitamine juba osas \*.3.

## Kasutame pildimaterjali:

Kuivõrd on oluline, et vooluahela kokkupanek ei valmistaks liigset raskust, siis on antud reeglina ka pilt Fritzing abil joonistatud skeemist. Ilmestamiseks on kasutatud fotosid. Fotoga illustreeritud selgituste korral, näiteks multimeetrit tutvustavas osas, on osutunud vajalikuks värviliste täppidega jooniselt õiget kohta tähele panema suunata—saame näidismõõtmist edasi anda vaid fotodega, seega tähistame tähelepanu nõudvad kohad fotodel värvilise markeriga.

## Programmide annamine sageli tervikuna ette:

Uut osa puudutavad programmid on reeglina antud täies mahus—neid on soovitatav üritada mitte kopeerida vaid iseseisvalt läbi kirjutada. Samas on peatükkides ka selgelt öeldud, et kopeerimine ei ole keelatud, pigem soositud—ent seejuures peab õpilane oskama programme vajadusel muuta ning peab ka oskama oma otsust teistele selgitada. Näiteks, kui ühendatud on

teisiti kui programmis antud viikudega, siis peaks oskama programmi muuta nii, et see oleks vastavuses tegeliku viikude kasutusega.

### **Materjal on jaotatud peatükkideks—ei ole jaotatud tundideks**

Keerukas oleks õpetajale, kes on peatükkide teemad ise läbi mänginud ja naudib õpetamist, ette öelda täpset ajastust. Eeldame, et valikkursuse puhul ei ole tähtis mitte minutipealt tundi ette kirjutada, vaid et tund „valmib“ õpetaja-õpilase koostöös ja arvestab individuaalsete arenguvajadustega. Sellisel juhul on vajalik paindlikkus. Paindlikkuse saavutamiseks võib ajapuuduse korral piirduda ka vaid valitud peatükkidega—näiteks nendega, mille tarvis on olemas riistvara. Valikkursuse keskel asuvale teadmiste kontrolli osale järgnevad peatükid on kasutatavad materjalina, mille seast teha valikud erinevaid projekte kombineerivate tundide tarvis. Etteruttavalt olgu öeldud, et testimisel oli soov pigem tagada õpilasele individuaalset lähenemist—ja seega esines pikemate selgituste puhul nii „graafikust mahajäämist“ kui ka seda, et peatükile mõeldud aja jooksul said valmis ka selle kodutööd. Sellistel juhtudel küsiti autori käest kodutööle lisa või sooviti ette võtta järgmist peatükki.

Kindlasti ei ole keelatud mahutada mitut peatükki ühe õpikohtumise („tunni“) sisse. Eriti alguspeatükkide (näiteks esimesed kolm-neli peatükki) testimisel juhtus seda, et õpilane soovis kohe veel mõnda peatükki katsetama asuda. Autor pidas seejuures tähtsaks õpilase soovidele vastutulekut ning riistvara programmeerimise teemasse positiivse suhtumise toetamist. Näiteks kui nooremad õpilased tahavad kangesti ära proovida RGB värvidega mängimist, siis on mõistlik, et toome seda sisaldava kuuenda peatüki veidi ettepoole, ent enne RGB juurde jõudmist siiski võtame ette tavalise LED-iga tehtava sissejuhatuse ning näitame ära analoogsignaali ja digitaalsignaali erinevuse teema. Ilusana tunduv teema õpikohtumise lõpetuseks tekitab soovi osaleda järgmises tunnis: poolelijätmise kõhklustega võitlemise olulisust ei maksa alahinnata.

### **3.4. Eesti ja inglise keeles elektroonikast rääkimise problemaatika**

Probleeme on valikkursuse koostamisel esinenud eesti keeles töövahendite kirjeldamisega. Näiteks tundub ebatäpne nimetada kahekordset potentsiomeetrit (*joystick*) juhtkangiks. Ent paremat vastet leidmata olen ka pöördunud vastavas peatükis oma vaimukate õpilaste poole—öelge ise, mis selle asja nimi on—ent valige tehniliselt pädevana mõjuv väljendusviis.

Koloriitsemaid naljakesi ei ole autor lubanud pakkuda. Siinkohal ei saa jätta tähelepanuta, et elektroonikakomponentide kohta käivates terminites ning ka programmeerimiskeeltes levinud sõnastuste osas on viimasel ajal esinenud viiteid muudatuste vajalikkusest ka ingliskeelse maailma uudistes. Leiame terminite muutmisest vastavalt *workers* ja *helpers* ning *master process* asmele *parent process* soovitamist. Tegemist on pikema poliitilise sisuga probleemiga, mille juured on kahtlemata kahetsusväärse orjanduslikus ühiskonnakorralduses. Protsessis osaleva riistvara personifitseerimisega jätkamine uuenevas ingliskeelses terminoloogias on aga endiselt ebatäpsust sisaldav. Parimaks ingliskeelseks vasteks pean aga Django keelde pakutav *leader* ja *follower*. [34]

Hiljuti poleemikat tekitanud *master* ja *slave* terminitesse on autor suhtunud võimaluse korral vältivalt. Ei ole mõeldav *master* komponenti eestindada isandaks ja *slave* komponenti orjaks—saaksime ülimalt markeeritud keelekasutuse, mis autori hinnangul ei haakuks piisavalt mõiste 'takt' sisuga. Seega ei saanud jätta neid eestindamata seal, kus see võimalik—olen kasutanud sõnastust *takti etteandaja*<sup>1</sup> ning *takti kuulaja*. Ent igal pool ei pruugi olla võimalik *master* ja *slave* termineid vältida: nimelt on kujunenud kindlateks põhimõisteteks MOSI (*Master Out, Slave In*) ja MISO (*Master In, Slave Out*) ühendused. Kindlasti aga MOSI ja MISO selgitamisel tuleb kasutada ka mõisteid *master* ja *slave*—ning autor eelistab seda teha ingliskeelseid termineid kasutades. Ka eesti erialakeeles on ju kasutusel MOSI ja MISO.

Samasugune oleks autori poolt kasutatud sõnastustaktika ka näiteks *male* ja *female* puhul. Ehkki ka eestikeeles erialasuhtluses on laialt kasutusel mõiste-sõnastused nagu *isane juhe* ning *emane juhe*—olen nende terminitega ettevaatlikult ümber käinud. Autor ei taotle seisukohavõttu koloriitsemate sõnakastuste suhtes, neid teemasid aga ei ole ka mõtet antud valikkursuse tunde segama lasta.

### 3.5. Värvikood takistitel ja juhtmetel

Värvikood ning sellele sarnaseid võimalusi pakkuvad muud tähistused on algajale kergenduseks: näiteks LED erineva pikkusega „jalad“ annavad aimu õiget pidi ühendamisest seni, kuni asjast on saanud kätteharjunud liigutus; punase ja musta juhtme reserveerimine teatud korduvateks otstarveteks—näiteks punase ja musta juhtme korrektne ühendamine leiab aset ka

---

<sup>1</sup> Täpsuse huvides tuleb märkida, et *master* ei tegele vaid takti etteandmisega. Antud juhul aga oleme rõhutanud takti—tõlkekaalutlustel.

multimeetrit töökorda seades. Eesmärk on olnud kasutada kogu teadaolevat arsenalit tööd lihtsustavaid võtteid selleks, et algajale aine omandamisel tuge pakkuda—võimaluse korral on olnud keerukamas skeemis iga juhe erinevat värvi: tehes iga juhtme seega ka kergemini kontrollitavaks. Materjalide koostamisel lisati neile ka Fritzing keskkonnas koostatud skeemid. Paljudes algajatele mõeldud ning Fritzing skeemidega varustatud materjalides, mida ka ise kunagi kasutanud olen, on ühendusi kujutatud värvivõimalustele vähe tähelepanu pöörates. Leian, et näiteks juba üle seitsme juhtme korral ei ole selline esitusviis enam algaja suhtes sõbralikku tooni esindav. Seega materjalides pigem rohkem värvisid. Näitena toon antud valikkursuses LCD ühendamise kõigis LCD-d puudutavates peatükkides—12 juhtmega. Ka on oluline siinkohal märkida, et Fritzing abil juhtmeid kujutades on kasutegur väike, kui juhtmed ei asetsevad skeemil üksteise otsa asetunult ning on tähistatud ka sama värviga.

### **3.6. Näidistund—LCD ühendamise keerukusega toimetulek**

Valisin magistritöös näidistunnina kirjeldamiseks 11. peatükis sisalduva LCD ühendamise. (Vt Lisa 1. 11. peatükk—LCD kasutamise näide.) Tegemist on esimese LCD-d kasutava tunniga. LCD-käsitsemise õppimise algus on paljudele noorematele õpilastele keerukas harjumatult suure juhtmete hulga tõttu.

Tundi esindava peatükikirjelduse juhatab sisse traditsiooniline ülevaade eelnenud peatükkidest leitava olulise materjali kohta. Antud juhul osundab 11.0. eelnevalt potentsiomeetri osas omandatule—sest potentsiomeetriga on vaja reguleerida taustavalgust LCD-l—et saaksime õige kontrastsuse korral pingutamata lugeda näitu või muud teksti. Järgmisena anname 11.1. peatüki materjali kohta eesmärgi ja selgitused. Antud juhul on eesmärgiks eelkõige õppida Arduinot ühendama LCD ekraaniga, kuvada LCD-le teksti ning mitte lasta ennast häirida sellest, et LCD puhul tuleb 12 viiku õigesti ühendatud saada.

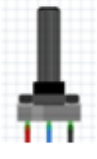
Edasi anname osas 11.2. töövahendite loetelu. Töövahendid jagunevad mustaga tähistatud tunnist tundi samaks jäävatest töövahenditeks ning punasega välja toodud selle tunni jaoks spetsiifiliselt vajalikeks.

Osas 11.3. kirjeldame peatüki tegevusi. Alustuseks mainitakse ka seda, et enne vooluahela sisselülitamist on vaja kontrollida, kas juhtmed on õigesti ühendatud. On juhtunud igasuguseid ühendamisvigasid ja harilikult jääb vigase ühendamise korral tulemus saavutamata, halvimal



juhul aga rikutakse vahendeid. Kuivõrd juhtmeid on võrreldes eelnenud peatükkides läbi tehtuga palju, siis peale testimisi otsustasin, et ei saa anda ette juhtmete ühendamist Fritzing skeemiga—pean seda tegema pigem värve kasutava tabelina, sest selle alusel on eksimist tulnud ette vaid loetud kordadel. (Vt joonis 1.)

LCD ühendused:

LCD-l asuva viigu tähistus:	Kuhu see ühendada:	Märkused: Juhtme värv meie skeemil:
VSS	GND	must juhe
VDD	5V	punane juhe
V0	Potentsiomeetri keskmise viigu külge 	sinine juhe potentsiomeetri ja LCD vahel Potentsiomeetri keskmiselt viigult analoogsignaali abil reguleerime LCD heledust/tumedust
RS	Arduinol digitaal-viik 10*	hall juhe
RW	GND	must juhe
E	Arduinol digitaal-viik 9*	violetne juhe
<i>Me ei ühenda millegagi järgmisi viik e: D0, D1, D2, D3.</i>	-	
D4	Arduinol digitaal-viik 5*	valge juhe
D5	Arduinol digitaal-viik 4*	kollane juhe
D6	Arduinol digitaal-viik 3*	roheline juhe
D7	Arduinol digitaal-viik 2*	oranž juhe
A	5V	punane juhe ning 220 Ω takisti
K	GND	ja must juhe

\* tärniga tähistatud viigud on vaja deklareerida programmis LCD-ekraanile vastavat objekti luues. Ülejäänud, tärniga tähistamata ühendused on kas GND või toide—neid ei ole vaja objekti luues deklareerida. Jälgime, et ühendused ja programm oleks omavahel kooskõlas.

**Joonis 1.** LCD ühendamist käsitleva 11. peatüki värvidega tabel aitab orienteeruda viikude ühendamise keerukuses. Tavaliselt kasutavad nooremad õpilased eri värvi juhtmeid, sest nendega on lihtsam juhehaaval oma tööd kontrollida.

Anname ette ka programmi ning edasine kujuneb tööle saadud LCD korral huvitavaks eksperimenteerimiseks ekraanile kirjutamisega. Programmis tuleb aru saada LCD kursoripunktidest—sellest, et arvutimaalimas alustatakse lugemist nullist. Algpunktiks seega ei saa olla 1;1. Peab olema 0;0. Kuivõrd on tegu Hitachi tehnoloogiaga, siis annan improviseerimiseks juurde ka idee, et ekraanile saab õigeid jaapani keele märke (katakana) kasutades kirjutada jaapani keeles Arduino. Peatükki läbides on tutvutud andmetüüpidega String ja char.

Sarnaselt on üles ehitatud enamus peatükke. Peatüki struktuuri valik on ülesehituslikult nagu koolitund: algab peatükist arusaamise eeldustega ning peatüki omandamiseks olulise infoga, materjali läbitegemisel kasutatava riistvara loendiga. Sellele järgneb uus osa, mis ehitub eelnenud peatükkides läbivõetule. Peatüki uus osa on alati interaktiivne—seda peab õpilane mitte niivõrd jälgima või kaasa tegema kui võrd ise läbi lahendama. Peatükkides on enamus vajalikku infot olemas. Seega on õpetaja roll vaid vajadusel suunata ning abistada. Kui juhuslikult on õpilased kiiremad kavandatust, siis on õpetaja roll suunata improviseerima, sest peatükkide materjal on omavahel kombineeritav. Peatüki lõpus on kodused ülesanded, mille mõte on pakkuda peatüki ainese üle järgi mõtlemise võimalust õppekohtumiste vahele jääval ajal—eesmärk on õpitu kinnistamine. Kui peab hindeid panema, siis on võimalus selleks kasutada koduste tööde esitamist.

Peatükkide tekstiosa koostamisega paralleelselt koostas autor ka vastavad Fritzing skeemid ning veendus piisava riistvara olemasolus järgnevat peatükkide testimiseks õpilaste peal. Peale koostamist sai eesmärgiks testimise abil õppematerjali viimistlemiseks täiendavat sisendit hankida. Järgmises peatükis vaatleme testimisel selgunut ning mida testimisel teadasaaduga ette võeti.

## **4. Peatükkide testimisel ilmnunud probleemid ja nendega arvestamine**

Töö esialgne eesmärk oli riistvara programmeerimise valikkursuse materjali koostamine ning seejärel testimine gümnaasiumiastme katseisikute grupi peal eesmärgiga testimisel saadud info põhjal vajalike muudatuste sisseviimine ning materjali lõplik kokkupanek. Eelnevalt riistvara programmeerimisega mitte kokku puutunud isikute rakendamine katseisikutena oli kavandatud selleks, et reaalseste algajate peal peatükkide testimise järel saavutada jõukohane sissejuhatav õppematerjal kasutamiseks gümnaasiumitaseme valikõppes algajatele. Materjali luues ei seatud eesmärgiks õpetada isikuid, kellel on juba eelteadmised—näiteks kogemused mingi muu kontrolleri-plaadiga.

Eespool on juba viidatud COVID-19 tõttu tekkinud eriolukorrale, mis tingis muudatused gümnaasistide peal valikkursuse peatükkide testimise plaanidesse; põhilised testimised viidi läbi oluliselt nooremaste vanuserühma kuuluvatel õpilastel. Vähesed järelejäänud gümnaasistidest huvilised, kelle puhul oli ideeks eratundidega siiski teha läbi kogu materjal, loobusid üksteise järel—keegi neist ei jõudnud kaugemale viiendast peatükist. Hoolimata oma väikesest panusest testimisse, on gümnaasistid ja nende vanemad olnud heaks kriitilise tagasiside andjaks. Nimelt valmistas ka lapsevanematele muret see, et segadusega alanud kriisiaeg teises kiirelt aineprogrammidele järgijõudmise harjutuseks. Õpilased kurtsid teatavat ülekoormust. Sellises olukorras on loomulik, et kui valikkursus on üles ehitatud sarnaselt põhiainetes õppekavadele ning võib hakata konkureerima õppetööga, siis et õpilased eelistavad põhiainetes edasijõudmist valikkursusele.

Ajapuuduse probleemi teadvustades ent endiselt huvitades ka riistvara programmeerimisest pakuti välja kahest ideest koosnev lahendus—esiteks lükata kogu peatükkide pilootprojekt 2020. aasta sügisele ning teiseks teha see ümber nii, et teadmiste kontroll ei satuks samale ajale tavapärase kontrollitööde hooaegadega gümnaasiumis. Idee oli hajutada koormust, et oleks võimalik töötada põneva valikainega. Mõlemad lahendus-ideed on head, ent enne magistri kaitsmist ei õnnestunud realiseerida.

### **4.1. Õpilastele sobiva aja leidmine**

Arutluste teemaks oli 45 minutise koolitundi mõõde. Algselt oli kursus kavandatud 32 õpikorraks ning iga kord mõeldud 45 minutilise koolitundi formaati järgivana. Praktikas aga on soovitud tegeleda valikainega terve päeva—ning ühe päeva jooksul on kõige rohkem

õnnestunud võtta läbi peatükid 1-6. Laste ja lastevanemate soovidele vastutulek ning nende vaba aja olemasolu on olnud seniste test-tundide jaoks võimaluste leidmisel määravaks. Targu on materjalis teema-üksused nimetatud mitte tundideks vaid peatükkideks.

Ülesehituslikult on kogu õppematerjal jaotatav kaheks teema-peatükkide grupiks, mille „vahel“ asub teadmiste vahekontroll (kordamine ja kirjalik töö koos kõigi vahendite kasutamise lubamisega). Õpilaste vanematelt laekunud tagasiside ja soovitude järgi võiks nn „esimene pool“ teemapeatükke saada läbi lahendatud õpilastele jõukohases tempos—mitte kiirustades. Esimeses pooles on tegu peatükkidega, mis käsitlevad alusteadmisi, seega soovituslik on mitte tingimata üritada selle olulise osaga jõuda nn „õigel ajal“ teadmiste kontrollimiseni vaid jätkata ainet avavate peatükkidega nii, et testimine ei kujuneks ajastuse osas konkureerimiseks põhiainetes valdavalt veerandilõppudele langevate kontrolltöödega.

Palju oleneb õpilaste endi otsustest materjaliga edasi liikuda. Näiteks testimisel on juhtunud, et esimese päeva hasart ei vaibu enne neljanda peatüki läbimist. Koloriitsem on näide, mil neljanda peatüki järele paluti, et autor näitaks ette ka kuuendas peatükis `analogWrite()`—kuna väga taheti katsetada „ilusate värvidega“ RGB LED-idega. Sellisel juhul on juba esimese õpikorra järel saavutatud teatav ajaline edumaa ning ka ajast ettejäudmine võiks anda võimaluse teadmiste kontrollil sattuda ajale, mil ei ole põhiainetes vaja niivõrd palju pingutada. Eesmärki mitte sattuda õpilastele keerukama ajastusega perioodidel nende riistvara õppimise lusti rikkuma koolikoormust ülemääraseks suurendes on mõistlik juba kursuse esimeste õpikohtumiste järgselt õpilastega läbi arutada. Tõenäoliselt väldib ette planeerimine just hoolsamate huviliste kõhkclusi ja loobumisi.

Vajadusel on võimalik ka teisest teemapeatükkide osast peatükke ettepoole tuua—nii et need leiaks aset enne 16. ja 17. peatükkides kirjeldatud teadmiste kontrolli. Sobivaimad ettepoole nihutamiseks on näiteks harjutused LCD-dega, relee ühendamine, Fritzing skeemi joonistamine. Ettepoole toomisel on mõistlik uurida \*.0 osadest eelnenu meenutamise vajadust.

## **4.2. Peatükkidesse testimise järel tehtud kohandused**

Kuna gümnaasiumiastme peal testimine jäi lünklikuks ning puudutas vaid alguspeatükke, siis oli vaja leida alternatiivset lahendust kogu materjali eelteadmisteta katseisiku peal testimiseks. Sellise võimaluse pakkusid aga välja mitte veel gümnaasiumiealiste laste vanemad. Kõige noorem testija on olnud esimeste testimiste ajal kaheksa-aastane. Ehkki ka selle grupiga jäid reeglina ära viimaste peatükkide (27-29) testimised ning eelnenud peatükke tuli kohendada, et

viia need vastavusse eelteadmistega, siiski jõudsin nooremate katseisikute peal läbi proovida oluliselt rohkem peatükke. Samas ei õnnestunud läbi viia kavandatud kujul teadmiste kontrolli; viimastes osades, milleks Digisparki käsitlev 27. peatükk ning ESP32 kohta 28. ja 29. peatükid, otsustasin ise teha demonstratsioon-esinemise ja piirduda väsimuse tundemärke ilmutava publiku küsimustele vastamisega. Ka sellise meetodiga ainese testimine andis sisendit, praeguses versioonis on viimaseid teemasid muudetud kergemaks.

Enamuse peatükkide osas tehtud muudatustest tingiski alternatiivne testgrupp, kes oli oluliselt noorem kavandatud vanusegrupist ning kelle eelteadmised mõjustasid ainet hoolikamalt sõnastama. Kõik teemapeatükid tegi ainsana läbi 10-aastane katseisik, kellel eelneva Lego-robotika hea kogemuse kõrval ka tugev isiklik veendumus matemaatika olulisusest. Selle katseisiku poolt õppetöö käigus antud kriitiline tagasiside on ka kõige rohkem mõjutanud nii töö stiili, illustatsioonide rohkust, värvide kasutamist ning pannud autorit mnemotehniliste võimaluste väljatoomisele tähelepanu pöörama.

### **Pildimaterjali mitmekesistamise vajadus**

Kursuse materjali esitamisel on mõnede eranditega kasutatud ka selle 10-aastase katseisiku poolt peatükke läbides autori juhendamisel tehtud fotosid. Suurim mõju aga on katseisikul olnud töö stiilile, mis on järk-järgult arenenud pigem Arduinoga töötamise lugu jutustava teksti poole. Seda valikkursust kirjutama asudes laekunud soovitus pigem kasutada koomiksi-formaati tundus esialgu liialt keerukana ning isegi kooliga sobimatuna. Tegelikuses aset leidnu alusel aga mõõnan, et ära jääma kippuva testimise konteksti üle juureldes olen selle nõuande ajal suurendanud pildimaterjali osakaalu ning arvan, et pildimaterjali võinuks veelgi rohkem kasutada. Koomiksistiili häiris aga korralikult iga „sele“ alla selle järjekorranumbri ning veidi väiksemas kirjas pildi allkirja lisamine. Pildid on seega pigem integreeritud teksti, sest täiendav liigendamine ei mõjunud piltide arvu suurendamise korral enam teksti mõjulepääsu soosivalt.

### **Õpik võiks olla „lugu jutustavam“**

Jutustava stiili rakendamine teenib formaalsema stiili vältimise eesmärki. Formaalsem esitluslaad on sobilikum teemat tutvustama neile, kellel on juba eelnev hariduslik taust, kes ei ole enam arendusplaadiga esimest korda kohtuvad algajad. Esmatutvust tegeva 'algaja' versus arendusplaatide osas juba eelistusi omava 'muuhulgas ka Arduinoga alustaja' tähelepanuvõime ning infovajadused on erinevad.

Stiili on seega mitmel korral kujundatud „jutustuse“ poole, milles võiks lugejale tunduda, et üheks tegelaseks on õpetaja ise. Sellisesse konteksti oli ka suurem julgus lisada ainest üksteise aitamisest ning koostööst, näiteks küsida, et mida võiksid põhikooli õpilased arvata, mis põhjusel on edukad Eesti robotikaettevõtted niivõrd tublid. Vastuseks loomulikult see, et isegi kiirel ajal nendes ettevõtetes kõik aitavad kõigi ühist ülesannet lahendada. Stiili on ka kritiseeritud, ent kuna näen olulisi eeliseid Arduinoga pigem enne gümnaasiumit alustamises, siis põhikoolile sama ainese lihtsustatud esitus oleks üks võimalikke projekti edasiarendusi.

### **Teadmiste kontrollimine versus viimane võimalus õppida**

Noorte katseisikute siiras kriitika osutus ka määravaks teadmiste kontrolli sisaldavate peatükkide tarvis sellise kontrolli-formaadi leidmisel, mis ei lisaks meeldivale elektroonikakogemusele kontrolltööle iseloomulikku stressi. Seega teadmiste kontrolli peatükid, mis endiselt on pigem vanemale katseisikute rühmale mõeldud, muutusid lühemaks ja loovamaks. Teadmiste kontrollil on lubatud kasutada kõiki allikaid. See muudatus on mõeldud andma ka eksamile vormingut, millest on aru saada, et tegemist on pigem viimaste õppimisvõimalustega just õppurile endale; see testimismeetod ei ole midagi väga uuenduslikku, seda on kasutatud ka mitte-informaatikutele mõeldud õppes. Kusjuures valikkursuse osas mainisime eespool ka vajadust mitte konkureerida põhiainete kontrolltöödega.

Siinkohal kõige ootamatumaks täiustuseks oli võimalus ära kasutada noorte katsetajate soovi pigem joonistada kui kirjutada. Peale esimeses peatükis gümnaasiumile mõeldud õpikust laenatud tsitaadi toel roboti osade selgitamist [35] palusin katseisikul joonistada vastavalt esimesel tunnil mõistetule robotit, mida ta ise hea meelega tahaks ehitama asuda. Toon ära pildi „kuurobotist“, millele palusin lisada ka kirjeldused, et oleks võimalik aru saada, mida pildil kujutatud. (Vt joonis 2.)

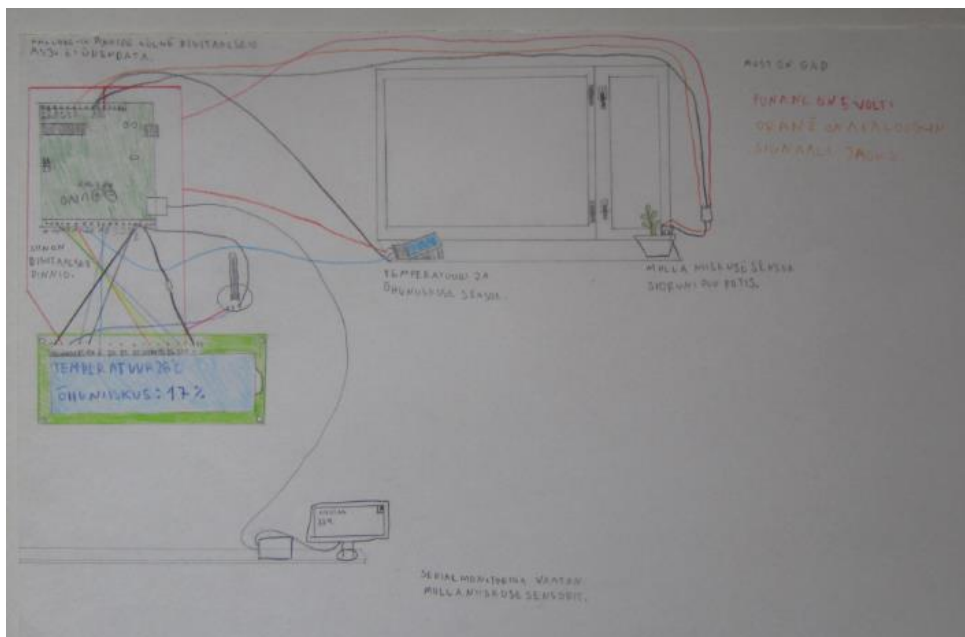


**Joonis 2. Kümneaastase nägemus robotist kursuse alguses on fantaasiaküllane.**

Kursuse lõppedes teavitasin vajadusest esitada peatükkide põhjal „lõputöö“ ning enne seda ka teha märkmed vähemalt ühest keerukamast tunnist. Valik langes LCD ekraani ühendamisele ning sellega temperatuuri ning õhuniiskuse anduri näitude kuvamise kohta pildi joonistamisele. LCD eelistamise põhjuseks sai see, et seda ei olnud kerge omandada, ent kõik ehitatud süsteemid nägid LCD-ga iseäranis uhked välja. Pilt sisaldab lisaks LCD-le ka niiskusanduriga aknalaua oleva sidrunipuu seemiku kasvumulla niiskuse jälgimist läbi Serial monitori. (Vt joonis 3.)

„Eksamiks“ kujunes aga iseseisevalt LCD-ga temperatuuri kuvamise peatükki läbi tehes skeemi koostamine ja töölepanek—võidukas demonstratsioon esinemine viimaseks tunniks kohale tulnud lapsevanema ees.

Kohendatud versioonis aga on sisse kirjutatud esimese tunni järel palve kas kirja panna või joonistada oma nägemus robotist. Sama palvet korratakse 30 peatükis: idee on õppurile endale kursuse lõppedes ette näidata, et tema arusaam robotikast on muutunud.



**Joonis 3.** kümneaastase õppuri nägemus oma toas Arduino abil töötavast süsteemist kursuse lõpus. Joonistamine aitas ka korrata LCD ühendamist, rohkete juhtmete valesti ühendamise kartusest saadi üle „märkmeid tehes“.

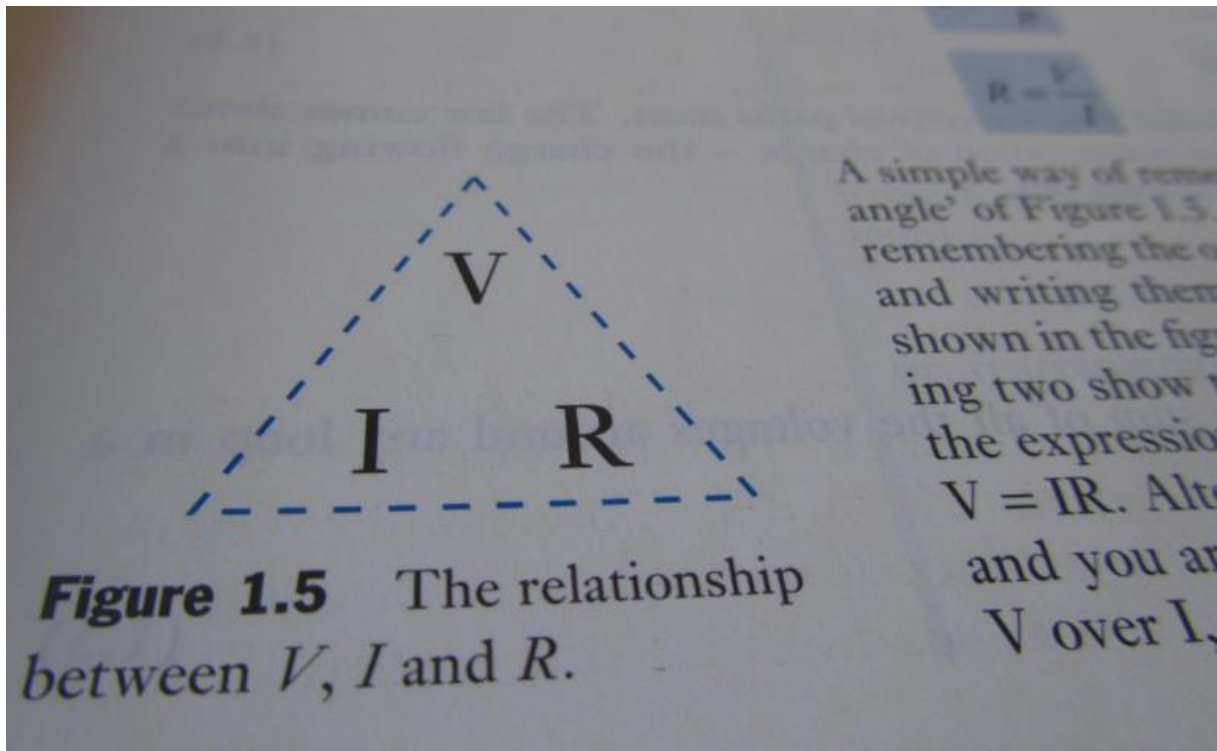
Noore peatükitestija robotinägemuses on pilte vaadates võimalik täheldada muutust. Tegemist ei ole enam pliiatseid kasutada oskava kuurobotiga. Põnevam on teha midagi, mis aitab aknalaua väikest sidrunipuud kasvata. Seega sai õpikusse sisse kirjutatud idee, et kursuse alguses tuleks ka vanemasse eagruppi kuulujatel paluda, et nad kirjalikult ja/või joonisena fikseeriks oma arusaama mingist reaalsest robotist, mida sooviksid ehitada. Selle ülesande saab anda ka kodutööna, mis tuleb esitada enne järgmist õpikohtumist õpetajale. Õpetaja aga hoiab need tööd alles kuni kursuse lõpuni ja siis jagab katseisikutele tagasi—et nad võiks ise veenduda, kuidas nad enne esimest kogemust Arduinoga teemat on ette kujutanud. Kui on tõsiselt palju aega, siis võib ka enne selle „joonistuse“ uuesti kätte andmist paluda teha veel ühe joonise—näiteks enne eksamit panna paberile, et millist robotit tahetakse ehitada kursuse lõpus. Kui õpilased saavad võrrelda oma esialgset „joonistust“ sellega, mis tehtud pikema praktika järel, siis on sel usutavasti huvitav efekt motivatsioonile jätkata enda arendamist—metakognitiivse suunitlusega võtte rakendamine.

### **Mnemotehnika kasutamine**

Eespool olen maininud 15. peatükki seoses multimeetri tundmaõppimisega; peatükis kontrollitakse mõõtmiste paikapidavust Ohmi seadust rakendades. Ohmi seaduse meeldejätmisel kasutatakse nn „virtuaalse kolmnurga“ võtet. Noorte peatükitestijate jaoks ei olnud koheselt jõukohane mõista  $U = IR$ ;  $I = U/R$  ning  $R = U/I$  kujul. Teemat selgitades näitasin



õpilastele Warwicki ülikooli õppejõu Neil Storey õpikus olevat nn virtuaalset kolmnurka. (Vt joonis 4.)



**Joonis 4. Noorematele katsetajatele näitasin Ohmi seadust sellisel kujul nagu see on leida Neil Storey õpikust *Electronics: A Systems Approach*. [27]**

Ohmi seaduse teeb õpilastele huvitavaks see, et ehtsate mõõtmiste andmetega tehte sooritamisel saadakse ligilähedaselt üsna täpne tulemus. Multimeeter on üks elektrooniku olulistest tööriistadest. Antud valikkursusel me seda küll eriti ei kasuta, ent mõeldes võimaliku jätkukursuse materjalide sarnaselt valikkursuseks vormistamisele ei saanud seda teemat kõrvale jätta ka kõige nooremate testijate puhul. Faktiks aga jääb, et noorimate peatükitestijate instrueerimisele kulus ebaproportsionaalselt palju aega.

### **Värvikood ja sellele sarnased tähistused**

Professionaalile ei pruugi olla tähtis ei valemite sisu edastamine kolmnurkse kujutisega ega ka mitte juhtme värv—professionaalid ei tegele põhiteadmiste meeldejätmisega. Ent õpilastele värvikoodi kõigil võimalustel ettenäitamine on pannud nad tähistuse üle järgi mõtlema ning sellest kinnipidamist jälgima. Siiski on tulnud ette päris palju näiteks LED-i ühendamist vales suunas—sellistel juhtudel mõistagi kulus lisa-aega sellele, et leida üles viga. Sageli eeldati, et viga võiks olla enda muudetud programmis. Ei eeldatud, et see võinuks tekkida vooluahela koostamisel LED-i valesti ühendades. Huvitava tähelepanekuna tooks välja, et RGB LED-i ei

ühendatud kordagi valesti—pigem võrreldi selle „keerukat“ nelja „jalga“ pikemalt ning siis tehti õiged otsused.

### **Huupi ühendamise ja faktoriaal**

Huupi ühendamise osas tundus noorematel kasutajatel olevat keerukas mõista probleemi tegelikku sisu—eriti kui praktikas probleemi lahendamine kolme-nelja juhtmega võib ka näiteks teisel katsel õnnestuda. Seega tuli eraldi selgitada, et selline tegevus mitte ainult et võib rikkuda komponente, vaid on olemuslikult mõttetu. Tõin asja illustreerimiseks sisse faktoriaali mõiste: tegemist on korrutamist rakendava võttega ning kolmanda klassis käivale katseisikule ei tekita korrutamine raskusi. Tõime välja, et kolme juhtme korral on ühendamiseks võimalike variantide hulk veel „väike“ ( $3! = 1 \times 2 \times 3 = 6$ ). Seejuures teadvustasime, et iga valesti ühendamine võib detailile kurvalt lõppeda. Ent näiteks 7 puhul on hea õnne peale vaikselt nokitsemine selgelt mõttetu. Sest näiteks  $7! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 = 5040$  erinevat võimalikku kombinatsiooni. Tark on siis pigem esitada küsimusi õpetajale.

Kui LCD-l on 12 juhet vaja õigesti ühendada, siis neid ei saa kohe kindlasti enam nii katsetada nagu LED-i puhul veel võimalik kahe juhtmega või kahe jalaga. Kui kolme viigu korral oli veel mingi šanss, siis 12 juhtmega „katsetamine ja proovimine,“ ei tule enam kõne alla ( $12! = 4.790016 \times 10^8$  puhul osutus vajalikuks ka ekskursioon astmetesse ja õpikohtumine läks päris pikalt üle aja).

### **Paljude juhtmete frustratsioonist üle saamine**

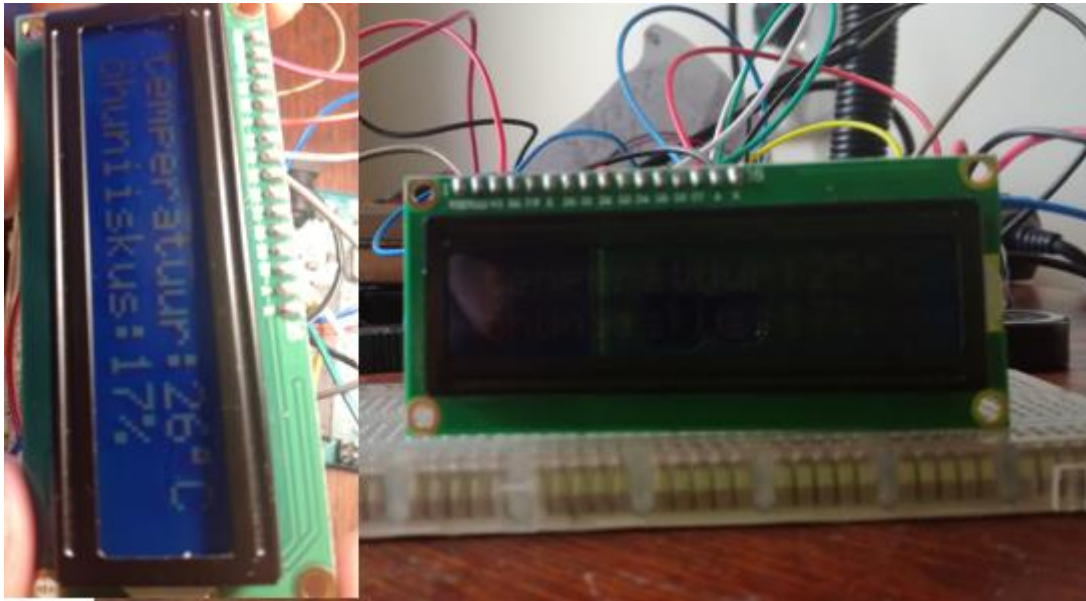
Esimest korda LCD peale sattudes 12 viigu ühendamise vajadust tunnetades tekkis LCD peatükke testinutel midagi, mida nimetaks „paljude juhtmete frustratsiooniks“. LCD ühendamist on kirjeldatud seega pigem tabelina—matkides andmelehtede stiili.<sup>2</sup> Kasutan tabelis ka värve—värvilistele juhtmetele tuginedes on sellise LCD viigu-ühendamise tabeli jälgimine oluliselt kergem. Kuna LCD on mitmetes peatükkides korduv, siis jäävad mingil hetkel meelde ühendamise üksikasjad. Ka neil juhtudel, kui on näha, et õpilasel ei ole arusaama, on sellise värvikoodi kasutava tabeli abiga suhteliselt kerge jõuda töötava LCD-ni.

Huvitavatest vigadest, mis on tehtud põhikoolis käiva katsetaja poolt antud värvikoodi alusel LCD-d ühendamisel eksides, on välja tuua liiga tugeva takistiga LCD taustavalguse häirimine.

---

<sup>2</sup> Olen sellistel kordadel lisa-aja olemasolul ka näidanud ehtsaid andmelehti internetist.

(Vt joonis 5.) Muus osas tema poolt kokku pandud süsteem töötab—ka liiga tumeda taustaga LCD pealt sai teatud nurga all välja lugeda seal kuvatavat infot.



**Joonis 5.** Valesti ühendatud takisti on teinud LCD tumedamaks kui normaalne, potentsiomeetrist ei ole seega abi. Samas aga on info teatud nurga alt välja loetav.

### **Programmeerimine ja inglise keele oskus**

Noorematel on probleemid inglise keelega—näiteks ei saada aru, mida tähendab „if“ või mida tähendab „for“. Seda on võimalik tunnis selgitada, ent tund on produktiivsem ilma sellise selgituseta. Seega on ilmselt mõistlik põhikoolis Arduino algõpet kavandades kõigepealt mõelda klassi keeleoskuse taseme peale—kas on vajalik selgitada pikemalt mis on 'for' ja mis on 'if'. Probleemi ilmumine peatükkide testimisel oli seda arusaamatum, et antud õpilased vaatavad kodus inglisekeelseid animafilme ning oskavad inglise keelt kasutada tavaolukorras.

Programmeerimist puudutavate tunniosade testimistest on sisend järgmine: näiteks kümneaastaselt oli teadmiste kontrolli ja esitluse hetkeks meelde jäänud „muutujate kõige ette panek“, et `void setup()` osas tehakse asju üks kord, ent `void loop()` —nagu nimi ka ütleb, jääbki „luupima“. Kursuse lõpuks arenes välja oskus kontrollida, kas viikude numbrid on vastavuses programmis sisalduvatelga—saavutati see oskus ühenduste ümbertöstmise ja seejärel programmi vastavusse viimise harjutustega. „Keerukamate asjadest“ pakkus eksperimenteerimist for-tsükkel. Analoogsignaali programmis töötamine versus digitaalsignaali töötamine aitas teadvustada Arduino UNO-l viikude tähistuste erinevusi—digitaalviikude real eristame neid, mis on pulsilaiusmodulatsiooni tähisega nendest, millel vaid

digitaalsed sisend ja väljundvõimekused; ning digitaalviikudele vastanduvad *Analog-In* tähistusega viigud—millelt saame signaali kätte `analogRead()` abil. Abistama pidi if-lausetega ning teekide allalaadimisel ja kasutamisel—seejuures põhiline osa probleemidest sügenes pigem napist inglise keele oskusest. Veateadete uurimist ette näidates õnnestus lahti saada sellest, et alatihti ununes ridadele lisada semikoolonit. Serial monitori osas aga jäi meelde see, et baudikiirused peavad olema samad nii programmis kui ka Serial monitori all ääres asuvas menüüs. See hulk teadmisi ei ole suur, ent mida varem alustada, seda tõenäolisem on õpilaste huvi jätkukursusel midagi keerukamat programmeerida.

### **Oma töö esitlemine või oma tegevusest jutustamine**

Tunnikontrolli tegemine tundus katseisikutele mõttetult väsitav—lihtsam on jutustada seda, mida teed, eriti kui saab ka vajadusel järgi vaadata oma vihikust. Seega on peatükkide katsetamisel suunatud õpilasi märkmeid tegema, üheskoos süsteemi kokku panema ning kõigil võimalikel unenemisjuhtudel üksteist aitama (vrd Praksise väljatoodud suhtlusoskuste vajadused. [6])

Internetist iseseisvalt info leidmise oskuste tekitamine nooremate peal ei õnnestunud. Põhjuseks inglise keele vähene oskus, mitte huvipuudus. Testimiseks tehtud õpikohtumistel vaadeldu on viinud mõttele, et kui kirjutada Arduino kursusele järge, siis peaks kaaluma selle tegemist inglise keele õpiku formaadis. Filoloogina leian, et see on teostatav.

### **Joonistamine Fritzing abil**

Fritzing joonestuskeskkonnas skeemide joonistamine on lisatud 20. peatükis. Õpilase poolt tehtud süsteeme saab ta ise väikese vaevaga korrektselt kujutada ning ühtlasi on läbijoonistamine ka märkmete eest: Fritzing abil tehtud joonistust saab kasutada ka valikkursuse lõpus toimival esitlusel. Seega igati eesmärgipärane harjutus—kusjuures lisaboonuseks ka see, et need, kellel ei ole piisavalt õnnestunud näidata oma tehnilist võimekust, saavad võimaluse aines esile tõusta hoopis joonistamisega. Rahulolu tundmine on oluline kursusega jätkumiseks.

## **Võimalus selgitada legaalse tarkvara ja intellektuaalomandi teemat**

Fritzing on pikka aega olnud tasuta keskkonnaks ning üsna suur oli üllatus, kui õpilastelt tuli info, et küsitakse allalaadimise eest raha. Probleemi esmalahenduseks sai autori arvutis töötamine. Arvutigraafikaga tegelemine, eriti raskusi valmistanud peatükkide skeemide läbijoonistamine aga tõi esile nooremate õpilaste õppimise positiivsed iseärasused—läbijoonistamistega kinnistusid töövõtted ja kasvas vahepeal probleemiks saama kippunud tahe kursusega jätkata. Ka sai Tinkercad-Fritzing võrdluses rohkem plusspunkte Fritzing ning seega sai arutelu uue suuna—kas 8 eurot või 25 eurot on suur või väike raha programmi eest maksta. Võrdluses Arduino UNO hinnaga, samuti võrdluses kioskist ostetava hamburgeri hinnaga jäi nooremate õpilaste aruteludes peale ikkagi ausalt Fritzingu kasutajaõiguste ostmine. Lapsevanemad ei pidanud 8 eurot suureks rahaks. Arutelu aga võttis uue suuna—intellektuaalomandi kasutamisel on oluline, et kõik tehtaks reegleid järgides. Autor hindab seda tulemust ning seega ei näe enam ka väga suurt vastuolu esmase eesmärgiga luua tasuta kasutamiseks valikkursust.

## **Vajadus materjali muuta**

Fritzing tasuliseks muutmine ei olnud kooskõlas varasema tööga materjalide ettevalmistamisel. Olen Fritzing kasutaja juba pikalt, alguses oli tegemist tasuta allalaetava programmiga ning sellisena sai see ka võetud valikkursuse skeemide joonistamise keskkonnaks. Antud juhtum aga näitas seda, kui kergesti on väike muudatus võimeline tekitama probleeme sarnase valikkursuse õpetamisele. Sellistel juhtudel on kindlasti vajalik õppematerjale kui mitte üleni muuta, siis vähemalt panna sisse selgitused—20.peatükki sai täiendatud mõttekäiguga intellektuaalomandi legaalsel teel omandamisest ning see sai kajastatud ka peatüki eesmärkides. Seega kinnitas Fritzing-keskkonna tasuliseks muutumine kahte tavateadmist: esiteks on sarnase valikkursuse alusel õpetades vaja õppematerjalidega pidevalt tööd teha, teiseks on oluline teha sarnased materjalid kättesaadavaks internetis, et oleks operatiivne võimalus neid vajadusel korrigeerida.

## **Palju siiraid küsimusi**

Teatud küsimusi vanemas eagrupid ilmselt naljalt ei esitata. Näiteks, mida täpselt tähendab sinihammas ja miks on sellel nii imelik nimi. Sellele küsimusele vastamiseks tuli idee ka kirja panna lisalugemiseks mõeldud osi—vahelduseks muule töötamisele nende jaoks, kes ehk kogu kursuse vältel väsimuse tundemärke hakkavad avaldama. Muide, sinihamba osas on noorematel

katsealustel reeglina juba kerge tüdimus välja arenemas—kui aga sinihambad lõpuks omavahel suhtlema saadakse, siis naaseb huvi AT käskude juures ning sinihamba olekud ja neile vastav vilkumine jääb ka noorematele erksalt meelde. Samas täpne seos vilkumismustrite ja olekute vahel vajab mõne päeva pärast uuesti üle kordamist.

Kui sinihamba osas vajati abi, siis infrapuna LED-iga töötamine, eriti teleripuldiga katsetamine tõi elevust, ühtlasi õnnestus ka selgitada heksadetsimaal-koodi matemaatilist alust ning jutustada tänapäeva LED-idele eelnenud tehnoloogia-lugu. Heksadetsimaalkoodi juurde sobib ka lugu kunagi Californias elanud indiaani hõimust, kes lugesid mitte sõrmi vaid sõrmevahesid—sobib ilmestama kümnendsüsteemi konteksti. Tegelikult on kaheksandsüsteemi alusel rehkendanud hõime olnud regioonis rohkem, näiteks Pamea hõimud Mehhikos [36] Maiadel aga oli kasutusel kahekümnendsüsteem. Algaja tasemel mõeldud õpperaamatutes on teemat kasutatud vahelugemisena näiteks G.Michael Schneide ja Judith L.Gersting „Invitation to Computer Science“ kuuendas väljaandes [37]

Peatükkide testimisel osalenutele jäid pisut kaugeks kolm viimast—sisaldavad pikemat tarkvara allalaadimist ning installeerimist. Kursuse lõpus selgus, et eriti noorematele on mingist hetkest selline kuiv väljaõpe väsitav isegi ka vaid jälgida. Samas Digispark suutis õpikohtumisele elu sisse tuua üllatava lõpuga programmeerimis-tagajärjega. Arduino IDE Digisparki-näidetest on lihtsasti võetav „pahatahtlikult programme rikkuvaks isekirjutavaks klaviatuuriks“ programmeerimise harjutus. Selline meelega programmeeritud äpardus tegi selgeks, et suvalist mä lupulka ei ole tark oma arvutiga ühendada. Igavavõitu väsitava tunni lõpp on 27. peatüki testimisel osutunud mõnedele gümnaasiumieast noorematele katsealustele sedavõrd virgutavaks, et ka samamoodi „igavate“ ESP32 28. ning 29. peatüki osas oodatakse õpikohtumise lõpus mingit „puänti“. Lisan selgituseks, et ESP32 algab samamoodi „tüütava“ allalaadimisega ja esmase käimasaamisega, ent õpilasel on eelmise õpikohtumise Digisparki kogemus värskelt meeles.

ESP32 on Arduinoga harjunule kindlasti keerukam mõista. Mõnda aega tagasi oli katsetamises osalenutel sama vähe kogemust ka Arduinoga. Mõlemal juhul on uue materjaliga alustamiseks vaja teha läbi klassikaline käimasaamise ning tutvumise rutiin—jõuda LED vilgutamiseni. ESP32 puhul omandavad õpilased tegelikult alles viimases peatükis programmi veateadete rahulikuma reageerimise oskuse—ning ka vajaduse uurida veateate sisu. Seega ESP32 sobib lõpetama Arduino valikkursust—annab õppurile kinnitust sellest, et ta on jõudnud uuele tasemele: ei ole enam algaja.

ESP32 toomine kursuse lõppu testimisel erilist sisendit peatükki muudatuste tegemiseks ei andnud, kuna pidin ise materjali demonstreerima. Ka oli kursuse lõpus tekkinud ärevus seoses välja valitud demonstratsioon-esinemisega—peast kogu LCD-de teema kommenteerimine.

### **Koordinaat-teljestiku näide potentsiomeetritega—juhtkang (*joystick*)**

Potentsiomeetrit puudutavas 9 peatükis tutvusime ka potentsiomeetri programmiga ning veendusime, et „juhtkangi“ abil töötamise programm sisaldab endas sisuliselt „kahte“ potentsiomeetri „programmi“. Seejärel 10. peatükis servo juhtimine juhtkangiga, mis on sisuliselt topelt-potentsiomeeter—annab võimaluse selgitada koordinaat-teljestikku. Kui potentsiomeetriga andsime väärtusi, mida võib ka kajastada ühel teljel, siis juhtkangiga saame näidata kaheteljelist süsteemi. Koordinaat-teljestikku kõigil noorematel katsetajatel koolis läbi võetud ei olnud, ent selgitustele toodi peagi ka oma paralleele—näiteks viidati malelual liikuvale malendile. Tegemist on võimalusega teha huvitavat kõrvalepõiget matemaatikasse.

## **4.3. Arduino UNO eelistamisest muudele kontrolleraalplaatidele**

Kuna antud magistr töö raames loodud Arduino näitele põhineva riistvara programmeerimise kursuse kohta on nii õpilaste kui ka lapsevanemate poolt sageli uuritud, et miks ei põhine kursus oluliselt võimsama mäluuga Raspberry Pi-1, siis toon välja põhjused, miks autor eelistab algajale mõeldud kursuse jaoks Arduino UNO-t ning Arduino IDE kasutamist.

Teine sagedane küsimus on puudutanud kloonide odavust võrreldes Arduino originaalplaatidega: miks ei kasutata Arduino UNO-st odavamaid võimalusi? Sellele vastamiseks toon näite, miks ma ei soovita algajale näiteks sõbra garaazist leitud odavlahendusi.

Selle magistr töö käigus loodud riistvara programmeerimise valikkursuse sissejuhatavas esimeses peatükis arutletakse lühidalt selle üle, miks eelistada Arduino UNO originaalplaati. Ühte lausesse kokku võetuna: sellel on testitult õiged Arduino UNO omadused ning seega on eksimisvõimalusi vähem, õppeprotsess lüüsam.

Originaal-plaate kasutades ei ole algaja sunnitud nuputama talle üle jõu käivatel teemadel ning saab pigem peatükkide sisuga edasi liikuda. Edasijõudmise lüüsam aga tähendab sageli õpilase jaoks otsust kas ainega jätkata või riistvara programmeerimise valikkursus pooltli jätta. Näite toon ADC (*Analog-to-Digital Converter*) kontekstis ise päris alguses kogetu põhjal.

Kursuse jooksul korduvalt analoogsignaali tegeledes pöörame tähelepanu analoogsignaali ning digitaalsignaali erinevustele. Valikkursuse 6. peatükis (*RGB LED-i ühendamine ja RGB võimaluste katsetamine analogWrite() abil*) juhatame sisse analoogsignaali teema ning seejärel 7. peatükis (*Valgustundliku takisti abil valguse mõõtmine ja kuvamine arvuti ekraanil—Serial monitor*) jõuame ka pulsilaius-modulatsioonini. Materjali algajale õpilasele on siinkohal piisavalt ning koolitundides ainesega toimetuleku peale mõeldes ei saa väga palju materjali enam peatükkidele juurde lisada—seega on mõneti paratamatu, et me valikkursuse materjalides seda olulist konverterit (MCP3008) eraldi välja ei too, ent saame ära kasutada ADC olemasolust tulenevat võimalust tegeleda algajale olulisemaga—nii ühendamise kui ka programmeerimise seisukohalt oluliste analoog-digitaal erinevuste tutvustamisega. (Vaadates MCP3004/3008 faktilehte näeme, et tegemist on nõudliku ainesega, mis nõuab algaja tasemest paremaid oskusi. [38])

Arduinot kasutades ei ole analoogsignaali teemaga esmatutvuse alguses koheselt vaja ka ADC konverteri ühendamise teemat käsitleda. Seega saame anda algajale aega ka näiteks töövõtetes orienteerumiseks. Katsed esialgsete nn „tunnikirjeldustega“ on veenvalt näidanud, et algajale on siin vaja anda aega.

Arduinot kasutades seega võidab õppematerjali esmane ülevaatlikkus, edasimineku teemadega ei kaota hoogu ning seega loodetavasti on väiksemad ka frustratsiooni-aistinguid nõrgema tehnikahuviga õpilastel. Koolituse organiseerimisel tuleb Arduino eelisena kindlasti välja ka Raspberry Pi-st odavam hind.

Õeldu ei tähenda, et järgmisena ei võiks tutvustada ka ADC kasutamist näiteks MCP3008 näitel koos Raspberry Pi-ga—ent see võiks leida aset näiteks Arduino kursuses õpitule järgneval Raspberry Pi kursusel.

Siinkohal pean kommenteerima ka originaalplaadi versus odav-klooni vahetegu. Veel mõned aastad tagasi iseseisvalt esimesi katsetusi tehes olen kasutanud sellesama analoog-ainese läbimängimiseks sõprade töötubadesse seisma jäänud odavaid kloone. Kui autori praeguste kogemuste pinnalt oleks tegu väljakutset pakkuva uurimisainesega, siis algajana oli tegemist töö seismajäämise ning nõutusega: lisaks probleemidele portidega oli probleem ka ADC konverteriga. Näiteks on autoril Arduino õppimisega alustades juhtunud, et ise esimest korda skeemi käivitada üritades ei ole saanud ühendatud analoogsignaali üheltki selleks ettenähtud viigult.



Sellise probleemi heidutussuurust algajale näitlikustan sellega, et peale nädalapikkust äpardumist töö lihtsalt seisiski seni, kuni kõrvaline ekspertabi soovitas mul kui algajal jääda pigem originaal-Arduinode kasutajaks, mitte kiirustada mulle tundmatus kohas. Odavklooni kohta aga sain selgituseks, et ju on kellelgi terve suurem kilekotitais tarvikuid kunagi maha kukkunud, ühendused sees kergelt katki... Tagantjärei ei oska klooni suhtes täpsemat seisukohta võtta, toon välja vaid selle, et ajakasutuse seisukohalt on tark kasutada originaal-komponente vaatamata sellele, et tuleb veidi rohkem maksta.

Ehtsate Arduinode kasuks väidaks ka nende tugevust. Katsetundides on ette tulnud Arduino UNO-ga ühendatud süsteemi põrandale kukkumist; sellistes äpardustes osalenud Arduino UNO arendusplaadid on aga kõik tõrgeteta edasi töötanud. Siiski olen palunud noortel entusiastlikult kiirustavatel katseisikutel olla hoolikas ja mitte lasta ühelgi detailil kukkuda.

Teiseks tooks välja, et Arduino on suhteliselt tugev ka hetkeliste lühistamiste korral—neid on ette tulnud kõige nooremal katseisikute grupil: väikese kõhkluse järel võidakse huupi ühendamist proovida.

Originaal-Arduinodest oli valikkursuse kirjutamise ja läbimängimise alguses kasutusel Arduino Nano. Meeldivaimaks omaduseks on võimalus Arduino Nano selle all asuvate viikude abil kinnitada maketeerimislaua külge. Peatükkide materjali katseisikute peal tõsisemalt testida üritades selgus, et kõik ei pruugi võrdselt hästi näha viikude markeeringuid, mis Arduino Nano väiksematest mõõtudest tulenevalt on peenemas kirjas. Probleemi lahendamiseks tuli kõigis peatükkides, välja arvatud nendes, kus me midagi ei ühenda, täiendada tarvikute nimekirja luubiga. Ent kuna luubiga uurimine aeglustab tööd iga kord kui on vaja veenduda õige viiguga ühendamises—ja kuna aeglustamine sarnasel põhjusel võib õppuril tekitada frustratsiooni—siis otsustasin võtta kasutusele Arduino UNO. Kergesti kättesaadava Arduino UNO versioon 3 originaalplaatidel on ka viigupesade külgedele vastavad sildid kenasti loetavas selges ja suures kirjas peale kantud. Selge markeering vähendab oluliselt õpilaste kärsitusest aegajalt ette tulevat eksimisvõimalust. Frustratsiooni vältimise vajadust on arvestatud seega kogu kursuse ülesehituses—algajate väljalangevust või loobumist saab aidata vältida ka sellistele pisiasjadele tähelepanu pöörastes. Sõbra garaazist tasuta saadud odavkloonide osas peab lisama, et enamusel autori kollektsiooni sattunud kloonidel puudusid piisavalt hästi loetavad viigumarkeeringud. Odavklooni puhul on loogiline, et hoitakse kokku kõige pealt, mis seda kaupa veidigi kallimaks võiks teha.

Kuna peatükkide testimiseks antud tundide puhul oli prioriteediks toimivus ja ladusus, siis Arduino UNO versioon 3 plaatidele tõsisemat vastast esile ei kerkinudki. Soovitan riistvara

programmeerimises algajaile Arduino UNO-t—see arendusplaat hoiab kokku väärtuslikku õppeaega ja säästab detailidest, millesse süvenemine algaja edasijõudmise seisukohalt ei ole ratsionaalne.

Raspberry Pi-ga saame töötada erinevaid programmeerimiskeskondi kasutades, ent Arduinol toimetame vaid Arduino IDE-ga. Kui Raspberry Pi-l on vaja tööle saada ka operatsioonisüsteem, seejärel teha valik allalaetava programmeerimistarkvara osas, siis Arduino IDE kasutab õppuri arvutis olemasolevat operatsioonisüsteemi (õppuri arvutile me operatsioonisüsteemi reeglina uuesti laadima ju ei pea) ning seega on esimese töötava näiteni jõudmine kiirem ja kergem.

Arduino IDE kasuks tuleb välja tuua, et Arduino IDE toetab lisaks Arduino enda plaatidele veel hulka erinevaid kontrollereid—kursuse viimases kolmes peatükis (peatükid 27-29) tutvustame näiteks Digispark-i ning ESP32; kumbki ei ole Arduino riistvara. Neid saab aga programmiga varustada Arduino IDE abil, lisaks on nii *Digispark*-ile kui ka ESP-dele Arduino IDE-s ohtralt õppenäiteid. Seega saame Arduino IDE-t kasutada algajale sissejuhatuse võimaldamisel ka mitte-Arduino tehnoloogiasse. Allajäämine Raspberry Pi-le ei ole niivõrd ühene.

Raspberry Pi-d soovitaks algaja tasemele koostatud valikkursusele järgnema mõeldud kursuseks. Arduinoga omandatud kogemustele toetudes on Raspberry Pi võimaluste avamine ladiusam ja põnevat äratundmist võimaldav: Raspberry Pi on väike kompuuter, Arduino on praktiline kontrolliplaat. Kummagi eelised ilmnevad erinevates olukordades.

Kui võrdlesime Arduino UNO-t Raspberry Pi-ga, siis võiks tuua võrdluse ka Arduino UNO ja ESP32 vahel.

ESP32 arendusplaatide hinnavahe on ligilähedaselt sama, mis Arduinode oma. Võrrelduna Arduino Uno Rev3 arendusplaadiga võib olla näiteks ESP32 Wifi BLE CH340 isegi kallim [39, 40] , ent ESP32 tuleb koos wifi ning BLE-ga (BLE = Bluetooth Low Energy), mis teeb ESP32 arendusplaadist võrreldes Arduino UNO-ga ülimalt soodsa hinnaga riistvaralahenduse. Tõime eespool ka juba välja programmeeritavuse Arduino IDE-ga: lisame sellele veel võimaluse, et paljudel juhtudel saame ESP32 peal kasutada ka Arduino teeki. See teeb lisaks Raspberry Pi-le ka ESP32-st tugeva kandidaadi Arduinoga sooritatud kursuse võimaliku jätkukursuse tarvis.

Kas Arduino UNO ning ESP32 võrdluses oleks seega Arduino justkui allajääjaks? Mitte tingimata—valik oleneb endiselt sellest, mida me teha tahame ning kas oleme algajad või juba teatud määral edasijõudnud. Kursuse viimastes peatükkides on õpilasele hea vihjata, et tal on esimesed sammud ESP-ga tehtud juba Arduino valikkursuse lõppedes—enam ei olda algaja.

ESP32 ei olegi nii keerukas kui kohati interneti allikatest mulje võiks jääda; võib-olla on raskused selles, et ESP32 puhul peaks mõtlema üksikute viikude täpsemate omaduste kontekstis [41], mis on ehk tunnetuslikult teistmoodi kui me seda siiani Arduinoga harjunud oleme. Selge on aga see, et ESP32 õppima asumisel peame vahetama põhilise infoallikana Arduino kodulehe Espressif kodulehe vastu. [42]

Vaikkursuse kirjutamise projekti testides kerkis esile ka küsimus, et miks ei kasuta autor Arduino laiendusplaat—näiteks LCD-d puudutavates osades ei tekiks küsimust „juhtme-frustratsioonist“ ning see oleks kenasti vastavuses sooviga vältida väljalangevust. Tõsi, laiendusplaadil on viigud, mille abil saame ta kinnitada Arduino UNO põhiplaadi viikude kontaktidesse ja kiirelt teemaga edasi. Samas on mulle tundunud selline võte omamoodi „ebaauus“ õppuri suhtes: *plug-and-play*. LCD-de ühendamist sai seega kergemaks tehtud pigem instruksioonidesse eri värvi juhtmete abil ühendamist toetava tabeliga.

#### **4.4. Peatükkide testimise järgsed tähelepanekud ning jätku-ideed edaspidiseks tööks**

Esialgne eesmärk oli gümnaasiumiastmele mõeldud peatükitekste testida vastavas vanusegrupis õppurite rühma peal koolimiljões. Ent COVID-19 eriolukorras töoga edasi liikuda soovides mängis juhus kätte võimaluse testida oluliselt nooremate õpilaste peal. See tõi välja aspekte, mis õige vanusegrupi peal peatükke testides suure tõenäosusega ei oleks sellisel kujul ilmnunud. Nooremate peal testimise tulemuseks oli peatükkide sisu ümbertegemine suurema illustratiivsuse saavutamiseks. Usutavasti paranes nii ka valikkursuse jõukohasus gümnaasiumiastmele.

##### **Ei pea ootama gümnaasiumiastmeni**

Samade teemade testimine plaanitud eagrupid nooremate peal on ka viinud mõttele, et tingimata ei peaks riistvara valikkursusega ootama õpilaste gümnaasiumi-ikka jõudmiseni, sest nooremas eagrupid on aine omandamisel teatud eelised. Kui algkursust võtta põhikoolis, siis gümnaasiumi astmes oldaks võimelised võtma juba järgmise raskustaseme valikkursusi. See vastaks ka eespool toodud riiklike programmide suunistele. Piisavalt varases eas erialase temaatika tutvustamine ning seejuures sellele eale iseloomuliku uudishimu ja koostööhuvi soodustamine on usutavasti pikemas perspektiivis majandusliku tähendusega ka Eestis.

## **Üllatuslikud probleemid inglise keelega**

Õpetamisel on tulnud välja kohatised puudujäägid inglise keele osas. Autori arvates ei ole keerukas kirjutada inglise keele õpikut, mis ilmnenud puudujääke adresseeriks. Mõnede eakohaste matemaatiliste illustratsioonide soe vastuvõtt noorema testijaskonna poolt on viinud mõttele, et võimalikus jätkuprojektis on mõistlik sisse tuua rohkem matemaatika põhivarasse kuuluvaid mõisteid ning viia läbi ka vastavaid harjutusi.

## **Matemaatikast toodud illustratsioonidele soodus vastuvõtt nooremalt testijatelt**

Mitmes peatükis on ettevaatlikult sisse toodud huvitavat matemaatikat; seda suundumust on toetanud nii lapsevanemad kui teemadest rõõmu tundnud peatükkide testijad. Töö võimaliku arenduse korral on (lisaks ideedele Arduinoga süvitsi mineku osas, Raspberry Pi või ESP32-osas, inglise keele õpiku formaadi kasutamise osas) ka kindlasti vaja kaaluda suunavõttu matemaatikale.

## **Kursuse poolijätmise põhjustest**

Kursuse poolelijätmisel ei olnud tegemist poolelijätja laiskuse või suutmatusega. Kursusest mõne peatüki läbi testinud, ent hiljem eriolukorra ajal põhiainetes ülekoormust põhjusena tuues töö pooleli jätnud olid huvitatud võimalusest testida sama programmi sügisel õppeaasta alguses—kui ei ole veel saanud kontrolltööde hooaeg. Sama kinnitasid ka lapsevanemad. Et aga poolelijätmine on teemaks, seda kinnitavad ka mõned sama teemat puudutavad Eesti kohta ilmunud materjalid. [43].

Seega on võimalikus jätkutöös olulisel kohal ka info kogumine kursusel osaleda tahtnutelt, kes ikkagi otsustavad loobumise kasuks, ja nende vanematelt. Võimalikel probleemidel on suure tõenäosusega ka suhteliselt lihtsad lahendused—üks võimalusi ongi kirjutada valikkursus, millel ei oleks põhiaine loogikat järgivat „ajagraafikut“. Selline lähenemine võiks saada testitud ja ka lahkelt mõistetud just innovaativisusele suunatud aines.

## **Iseõppijale mõeldud materjal**

Kuna kursuse materjalide testimine langes COVID-19 perioodile, siis tuli muuta materjali iseõppija-sõbralikumaks: tuli arvestada kriitikaga õpetaja kättesaadavuse osas ning näiteks lisada rohkem illustratsioone, teha selgitused pikemad ja detailsemad, korrata sama mitmel korral uuesti üle. Vaja oluks täiendavat tööd nii õpikeskkonnaga kui ka iseõppija võimalike vajaduste paremat tundmist. Iseseisva õppimise tarvis materjali kohandamine sobiks samuti jätkuprojektiks. Eesmärgiks võiks seejuures olla eesti keeles tasuta materjali kättesaadavaks tegemine täiskasvanud iseõppijast algajale.

Testimist kokku võttes—kui ei saa testida selle eagrupi peal, mille jaoks materjal kavandatud, siis see ei ole tingimata projekti hukutavaks probleemiks. Noorema eagrupi peal testimisega selgus materjali kohendamise vajadus oodatust erinevalt, ent ometi piisava veenvusega. Lisatulemuseks õnnestunud õpikohtumistele on ka veendumus, et gümnaasiumini oodates oleme õppega hiljaks jäämas ning tõenäoliselt kaotamas ka kodumaa konkurentsivõimes.

## 5. Kokkuvõte

Noored Koodi soovitusel alustatud magistritöö käigus loodi kokku 32 peatükist koosnev õppematerjal Arduino näitele tugineva riistvara programmeerimise valikkursuse õpetamiseks gümnaasiumiastmele. Antud teemas õppematerjalide loomiseflisa 1. ülesanne on kooskõlas ka Eesti arengukavade ning IKT sektori tööturgu puudutavate uuringute soovitustega tõhustada ning mitmekesistada nii erialateadmiste omandamisele kui ka näiteks koostöövõime ja üldpädevuste tugevdamisele suunatud võimekusi.

Esimeses peatükis vaatlesime, et Arduinol, eriti aga Arduino UNO-1 on rida häid omadusi, mis teevad sellest hea vahendi, millele tugineda Eesti arengukonteksti sobiliku emakeelse programmeerimise valikkursuse loomisel.

Teises peatükis vaatlesime kursuse materjalide läbimiseks vajaliku riistvara sisalduvust Arduino stardikomplektis ning veendusime, et lisaks stardikomplektile tuleb antud kursuse peatükkide läbitegemiseks hankida täiendavat varustust.

Kolmandas peatükis kirjeldasime valikkursuse loomise algetappi: peatükkide koostamist, nende tarvis jooniste tegemist enne peatükkide testimist. Testimiseks ettevalmistatud materjal jaotus teema-peatükkideks ning kordamiseks ning teadmiste esitluse läbiviimiseks vajalikeks peatükkideks. Materjalid koostati arvestusega, et neid testitakse gümnaasiumiealistest õpilastest katseisikute grupi peal eesmärgiga peatükke täiendada.

Neljandas peatükis näitame testimise tegelikku kulgu. Seoses COVID-19 ja eriolukorraga pidanuks testimised edasi lükkama. Olukorra lahenduseks kujunes materjali testimine planeeritust noorematega. Tulemuste osas tuleb märkida tekstide kohendamist jõukohasemaks, illustatsioonide ning muude meeldejätmist soodustavate elementide osatähtsuse suurendamist.

Gümnaasiumiastmega testimise väheseks jäämise põhjuste analüüs näitas gümnaasiumiealiste huvi olemasolu, samas vajadust valikkursuste puhul vältida õpilase koormuse suurendamist põhiainete kontrolltööde hooaegadel.

Kavandatust nooremate õpilaste peal gümnaasiumiastmele mõeldud materjalide testimist võiks pidada õnnestunuks ses osas, et sellega juhatati autorile kätte mitmed võimalused kohendada materjale kergemini omandatavaks. Ei puudunud ka üllatused: autor ei oodanud võimalust sisse tuua intellektuaalomandi legaalse ostmise teemat. Põhikoolis käivate õpilastega töötamise

edukusest tulenevalt soovib autor Arduinol põhineva riistvara programmeerimise koolitusega võimaluse korral mitte tingimata oodata õpilaste gümnaasiumiealiseks saamiseni vaid anda võimalus esmatutvuseks juba põhikoolis. Varasem alustamine võiks anda head efekti ka Eesti rahvuslike eesmärkide ning arengukavade elluviimisel.

## 6. Viidatud allikad

- [1] “Eesti 2020“ tegevuskava 2020-2021, lk 8.  
[https://www.riigikantselei.ee/sites/default/files/content-editors/Failid/eesti2020/ee2020\\_tegevuskava\\_30.04.2020.pdf](https://www.riigikantselei.ee/sites/default/files/content-editors/Failid/eesti2020/ee2020_tegevuskava_30.04.2020.pdf) (29.07.2020)
- [2] Eesti Elukestva õppe strateegia aastaks 2020, lk 3.  
<https://www.hm.ee/sites/default/files/strateegia2020.pdf> (28.07.2020)
- [3] Gümnaasiumi riiklik õppekava RT I, 14.01.2011,2, §4 lg 8.  
<https://www.riigiteataja.ee/akt/129082014021> (28.07.2020)
- [4] Põhikooli riiklik õppekava RT I, 14.01.2011,1, §4 lg 8.  
<https://www.riigiteataja.ee/akt/129082014020> (28.07.2020)
- [5] Digipöörde programm 2018-2021, lk7.  
[https://www.hm.ee/sites/default/files/2\\_digipoorde\\_programm\\_2018-2021.pdf](https://www.hm.ee/sites/default/files/2_digipoorde_programm_2018-2021.pdf) (29.07.2020)
- [6] SA Praxis. Eesti IKT kompetentsidega töäjõu hetkeseisu ja vajaduste kaardistamine, 2013, lk 19-20.
- [7] Noored Koodi <http://nooredkoodi.ut.ee/pealeht> (11.07.2020)
- [8] Arduino foorumid <https://forum.arduino.cc/> (18.07.2020)
- [9] Arduino blogid <https://blog.arduino.cc/> (18.07.2020)
- [10] Arduino project hub <https://create.arduino.cc/projecthub> (18.07.2020)
- [11] Arduino: A Technical Reference by J. M. Hughes, Chapter 1. The Arduino Family  
<https://www.oreilly.com/library/view/arduino-a-technical/9781491934319/ch01.html>  
(18.07.2020)
- [12] Industrialshields, M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS  
<https://www.industrialshields.com/shop/product/is-mduino-21-m-duino-plc-arduino-ethernet-21-i-os-analog-digital-plus-3> (18.07.2020)
- [13] ArduSat. International Space Apps Challenge  
<https://2013.spaceappschallenge.org/challenge/ardusat> (18.07.2020)
- [14] Paris Innovation Review. Arduino: Our Real Boss is the Community  
<http://parisinnovationreview.com/articles-en/arduino-our-real-boss-is-the-community>  
(03.08.2020)
- [15] Pin headers—soldering, cutting, both male and female  
[https://rimstar.org/science\\_electronics\\_projects/pin\\_headers\\_soldering\\_cutting\\_male\\_female.htm](https://rimstar.org/science_electronics_projects/pin_headers_soldering_cutting_male_female.htm) (20.07.2020)
- [16] Headers or No Headers? <https://www.sparkfun.com/news/2027> (20.07.2020)



- [17] Arduino tutorials <https://www.arduino.cc/en/Tutorial/HomePage?from=Main.Tutorials> (18.07.2020)
- [18] P2P Foundation. Patterns of Commoning: Arduino and the Open Hardware Revolution <https://blog.p2pfoundation.net/patterns-commoning-arduino-open-hardware-revolution/2017/06/13> (03.08.2020)
- [19] Creative Commons. Attribution-ShareAlike 3.0 (CC BY-SA 3.0) <https://creativecommons.org/licenses/by-sa/3.0/> (18.07.2020)
- [20] Arduino reference <https://www.arduino.cc/reference/en/> (18.07.2020)
- [21] The Best Arduino Books You Can Read: Best Arduino Books for Beginners, Experts, and More <https://www.electromaker.io/blog/article/the-best-arduino-books-you-can-read> (29.07.2020)
- [22] Digiõpilood: Arduino mikrokontrollerid: 11. ja 12. klass, programmeerimise tunnid majanduse ja IT õppesuunal [https://oppevara.hitsa.ee/opilood/arduino-mikrokontrollerid\\_11-ja-12-klass\\_programmeerimise-tunnid-majanduse-ja-it-oppesuunal/?\\_ga=2.263617029.841844536.1596124279-1343509621.1596124279](https://oppevara.hitsa.ee/opilood/arduino-mikrokontrollerid_11-ja-12-klass_programmeerimise-tunnid-majanduse-ja-it-oppesuunal/?_ga=2.263617029.841844536.1596124279-1343509621.1596124279) (29.07.2020)
- [23] TÜ arvutiteaduse instituudi lõputööderegister. Arduino Stardikomplekti õppematerjalid [https://comserv.cs.ut.ee/ati\\_thesis/datasheet.php?id=53783&year=2016](https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=53783&year=2016) (24.07.2020)
- [24] TÜ arvutiteaduse instituudi lõputööderegister. Arduino platvormiga ühilduvad andurid ja liidesed ning nuputelefoni ehitamine [https://comserv.cs.ut.ee/ati\\_thesis/datasheet.php?id=58153&year=2017](https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=58153&year=2017)
- [25] Lauri Võsandi laurivosandi/arduino-book <https://github.com/laurivosandi/arduino-book> (24.07.2020)
- [26] Sell R., Raid K. Arduino projektid alustajale, 2017, lk 4-6 ja lk 8.
- [27] Storey N. Electronics: A Systems Approach, 2009, lk 9.
- [28] Erialase haridusega IKT-spetsialistidest on puudus õpingute katkestamise tõttu <https://oska.kutsekoda.ee/2018/03/erialase-haridusega-ikt-spetsialistidest-puudus-opingute-katkestamise-tottu/> (03.08.2020)
- [29] ERR. Tööturul on suurim puudus tarkvaraarendajatest ja kassapidajatest <https://www.err.ee/898886/tooturul-on-suurim-puudus-tarkvaraarendajatest-ja-kassapidajatest> (03.08.2020)
- [30] CV keskus. Kvalifitseeritud tööjõu nappus <https://www.cvkeskus.ee/karjaarikeskus/artiklite-arhiiv/kvalifitseeritud-toojou-nappus> (03.08.2020)

- [31] Personaliuudised. Eesti IT sektori tööjõu kriisi tasub leevendada nii eestimaalaste kui välistööjõuga <https://www.personaliuudised.ee/uudised/2019/03/13/it-sektori-toojoukriisi-tasub-leevendada-nii-eestimaalaste-kui-valistoojouga> (03.08.2020)
- [32] Postimees IT töötajate põud on muutunud kriitiliseks <https://tehnika.postimees.ee/6443227/it-tootajate-poud-on-muutunud-kriitiliseks> (03.08.2020)
- [33] Tiia Rüütmann. Inseneripedagoogika I, 2016, lk 12 ja lk 98-99.
- [34] 'Master-Slave' Terminology Was Removed from Python Programming Language [https://www.vice.com/en\\_us/article/8x7akv/masterslave-terminology-was-removed-from-python-programming-language](https://www.vice.com/en_us/article/8x7akv/masterslave-terminology-was-removed-from-python-programming-language) (28.0.2020)
- [35] Altin H. jt. Robotika gümnaasiumile. Mehhatroonika ja roboootika õpik, 2013, lk 13. [https://www.robotika.ee/wp/wp-content/uploads/2015/04/robotika\\_gymn\\_opik.pdf](https://www.robotika.ee/wp/wp-content/uploads/2015/04/robotika_gymn_opik.pdf) (27.07.2020)
- [36] Round Valley Indian Tribes <https://www.rvit.org/about> (19.07.2020)
- [37] Schneide M., Gersting J. Invitation to Computer Science, 2013, lk 154.
- [38] MCP3004/3008 2.7V 4-Channel/8-Channel 10-Bit A/D Converters with SPI Serial Interface [https://drive.google.com/file/d/1ebN1cMSdQ4JCTuU\\_vqnXUXZfS90PmnbJ/view](https://drive.google.com/file/d/1ebN1cMSdQ4JCTuU_vqnXUXZfS90PmnbJ/view) (02.08.2020)
- [39] Oomipood. ESP32 [https://www.oomipood.ee/product/espduino\\_32\\_esp\\_32\\_arendusplaat\\_wifi\\_ble\\_ch340](https://www.oomipood.ee/product/espduino_32_esp_32_arendusplaat_wifi_ble_ch340) (25.07.2020)
- [40] Oomipood. Arduino UNO Rev 3 [https://www.oomipood.ee/en/product/a000066\\_arduino\\_uno\\_rev3](https://www.oomipood.ee/en/product/a000066_arduino_uno_rev3) (25.07.2020)
- [41] ESP32 DevKitC <https://components101.com/microcontrollers/esp32-devkitc> (26.07.2020)
- [42] Espressif. ESP32 Series of Modules <https://www.espressif.com/en/products/modules/esp-wroom-32/overview> (25.07.2020)
- [43] Tõnisson E, Informaatikaõpe koolis – kas ja kuidas õpetada? <https://opleht.ee/2015/06/informaatikaope-koolis-kas-ja-kuidas-opetada/> (23.07.2020)
- [44] SA Praxis koduleht <http://www.praxis.ee/tood/eesti-ikt-toojou-uuring/> (08.08.2020)

# Lisa 1. 11. peatükk—LCD kasutamise näide

## 11. Ekraani ühendamine ja teksti kuvamine 1602 LCD ekraanil

### 11.0. Seos eelmiste peatükkidega

Kümnendas peatükis juhtisime potentsiomeetriga servomootori tööd. Kas saame potentsiomeetriga veel midagi juhtida? Näiteks LCD ekraanil kuvatava kirja ja taustavalguse kontrastsust?

Ka oleme juba töötanud esimese teegiga. Ka selles peatükis peame tööle panema teegi.

### 11.1. Eesmärk ja selgitused

Peatüki eesmärgiks on kuvada LCD ekraanile tekstilist infot. Alustuseks aga peame oskama LCD ekraani ühendada Arduinoga.

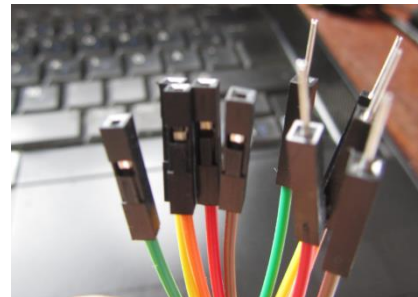
Esmapilgul tundub, et on heidutavalt palju juhtmeid—üheski peatükis ei ole olnud vajalik niivõrd suurel hulgal ühendusi. Anname sedakorda info ette pigem tabeli formaadis kasutamiseks—kas nii on kergem? Suurem kogus juhtmeid joonistatuna Fritzing-skeemile ei pruugi mingist hetkest enam kergesti loetav olla.

Selle peatükki järel õpilane oskab

- a) ühendada vedelkristall-ekraani (1602 LCD) Arduinoga;
- b) kuvada 1602 LCD-le teksti ning
- c) ei karda juhtmerohkust.

### 11.2. Töövahendid

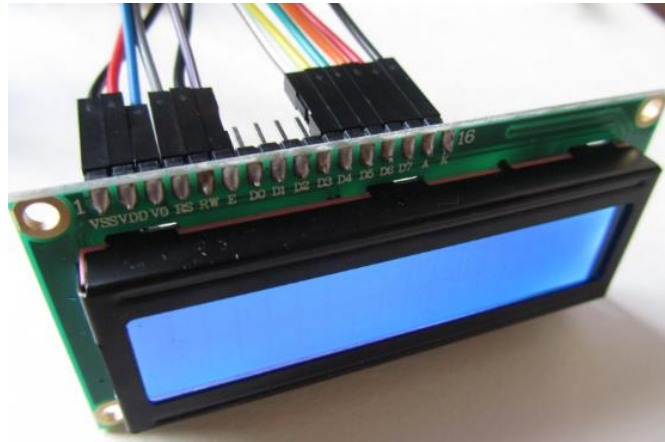
- 1) **Arduino Uno;**
- 2) **USB ühenduskaabel;**
- 3) **arvuti;**
- 4) **internetiühendus;**
- 5) **luup;**
- 6) **juhtmed** (nii mõlemast otsas metallist ühendusega kui ka ühest otsast metallist ühendusega kui ka mõlemast otsast tühja otsaga);
- 7) **maketeerimislaud;**
- 8) **LCD ekraan;**
- 9) **potentsiomeeter**—et saaksime reguleerida heledust-tumedust ekraanil.



### 11.3. Tegevused

#### 11.3.1. LCD ühendamine Arduinoga, potentsiomeetriga, temperatuuri ja niiskuse anduriga

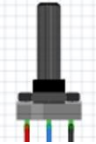
Seekord on vooluahelas juhtmeid ja viikusid üksjagu rohkelt. Tundub keerukas. Selle kogemuse tarvis anname 16X2 LCD ekraani ühendamiseks info ette pigem tabeli formaadis. Ülesanne on vastutusrikas—LCD kasutatavuse huvides ärge enne oma töö lõpliku resultaadi kontrollimist Arduino t vooluallikaga ühendage.



#### Harjutus A

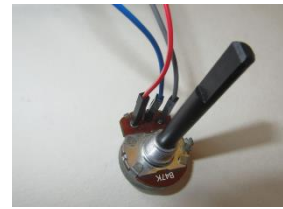
Järgmisel leheküljel asuva tabeli alusel koostage vooluahel. Seejärel aga kontrollime üle—enne vooluallikaga ei ühenda.

LCD ühendused:

LCD-l asuva viigu tähistus:	Kuhu see ühendada:	Märkused: Juhtme värv meie skeemil:
<b>VSS</b>	GND	must juhe
<b>VDD</b>	5V	punane juhe
<b>VO</b>	Potentsiomeetri keskmise viigu külge 	sinine juhe potentsiomeetri ja LCD vahel Potentsiomeetri keskmiselt viigult analoogsignaali abil reguleerime LCD heledust/tumedust
<b>RS</b>	Arduinol digitaal-viik <b>10*</b>	hall juhe
<b>RW</b>	GND	must juhe
<b>E</b>	Arduinol digitaal-viik <b>9*</b>	violetne juhe
<i>Me ei ühenda millegagi järgmisi viik e: D0, D1, D2, D3.</i>	-	
<b>D4</b>	Arduinol digitaal-viik <b>5*</b>	valge juhe
<b>D5</b>	Arduinol digitaal-viik <b>4*</b>	kollane juhe
<b>D6</b>	Arduinol digitaal-viik <b>3*</b>	roheline juhe
<b>D7</b>	Arduinol digitaal-viik <b>2*</b>	oranž juhe
<b>A</b>	5V	punane juhe ning 220 Ω takisti
<b>K</b>	GND	ja must juhe

\* tärniga tähistatud viigud on vaja deklareerida programmis LCD-ekraanile vastavat objekti luues. Ülejäänud, tärniga tähistamata ühendused on kas GND või toide—neid ei ole vaja objekti luues deklareerida. Jälgime, et ühendused ja programm oleks omavahel kooskõlas.

Potentisomeetri ühendused: vt



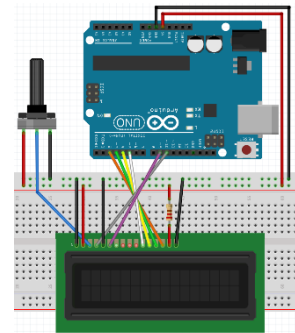
Vasak viik	5V	punane juhe
Keskmine viik	LCD peal viik V0	sinine juhe
Parem viik	GND	must juhe

### 11.3.2. Näidisskeem on meelega hästi pisikene

Vajadusel siiski saate ka nii võrrelda oma koostatud skeemi näidisskeemiga.

Kui oleme kontrollinud, kas kõik ikka õigesti paigas, alles siis on ohutu kokkupandud vooluahel USB kaabli abil ühendada oma arvutiga. Alates USB kaabliga ühendamisest on Arduinol ja temaga ühendatud vooluahela elementidel olemas toide.

Kui midagi nüüd muuta soovime, siis võtame kõigepealt toite maha— lihtsaim on eemaldada USB kaabel arvutil asuvast USB-kontaktist. Seejärel teeme vajalikud muudatused ning siis võime taas ühendada Arduino USB kaabli abil oma arvutiga.



### 11.3.3. Ja lisame ka programmi

```
#include <LiquidCrystal.h> /*laeme LCD ekraani library */
```

```
LiquidCrystal LCD_ekraan(10, 9, 5, 4, 3, 2); /*loome objekti  
LCD_ekraan,
```

millel deklareerime ühendused Arduino viigudele järgmises järjekorras:

```
RS (10), E (9), D4(5), D5(4), D6(3), D7(2).
```

Vaata eespool ka kommentaari LCD viigude ühendamist puudutavas värvilises tabelis--tegemist on antud värvilises ühendamistasbelis tärniga tähistatud viigudega--st nendega, mis ei ole GND või toide. Objekti LCD\_ekraan deklareerimisel on oluline viigude järjekord.

Näiteks:

kui anname ette teistsuguse järjekorra, siis tuleb ka tõsta ümber juhtmed. Näiteks LiquidCrystal LCD\_ekraan(10, 9, 5, 3, 4, 2); puhul vaheta omavahel kollane ja roheline--peale uuesti käivitamist hakkab tööle.

\*/

```
void setup() {
```

```
Serial.begin(9600);
```

```
LCD_ekraan.begin(16,2); /* Ütlesime programmile,
```

```
et vaja on käivitada kaherealine ja kuusteist märgiga LCD ekraan
```

```
--NB! järjekord on oluline: 16, 2! */
```

```
LCD_ekraan.setCursor(0,0); /* ütleme LCD ekraani kursorile, kus asub kursori algpunkt*/
```

```
}
```

```
void loop() {
```

```
/*korduma pidavas osas:
```

```
/*A) ülemine rida*/
```

```
LCD_ekraan.setCursor(0,0); /* esimene null tähistab esimese märgi asukohta,
```

```
teine null märgib ülemist (st esimest) rida:
```

```
arvutimaalimas alustame numbrite lugemist nullist!*/
```

```
LCD_ekraan.print("harjutus");
```

```
LCD_ekraan.print((char) 223); /*lisame midagi huvitavat--näiteks edaspidi läheb meil ka vaja kraadi-tähist. Detsimaalkujul see on 223. Teabe leiame aga LCD fact sheeti abiga*/
```

```
/*B) alumine rida*/
```

```
LCD_ekraan.setCursor(0,1); /* esimene null tähistab esimese märgi asukohta,
```

```
üks aga märgib alumist (st teist) rida:
```

```
arvutimaalimas alustame numbrite lugemist nullist!*/
```

```
LCD_ekraan.print((char) 239); /*lisame ö-sümboli */
```

```
LCD_ekraan.print((char) 239); /*lisame veel ühe ö-sümboli*/
```

```

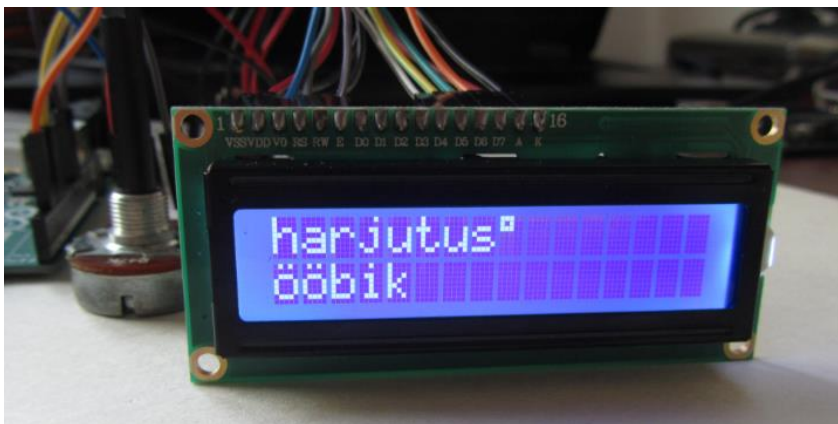
LCD_ekraan.print("bik"); /*jätkame eelmisel programmireal
alustatud sõna*/

delay(10); /*imepisiike delay vajalik süsteemil alguspunkti
naasmiseks--nii väikest viivitust inimsilm vaevalt et tähele paneb*/
}

```

Kui olete programmi käivitanud, tutvuge ka potentsiomeetriga: kruttige seda ja jälgige, mis toimub LCD ekraaniga—selgitage, milleks on vajalik potentsiomeeter ja kas ilma potentsiomeetrita võiks ka hakkama saada.

Kas saite samuti ekraanile midagi sellist?

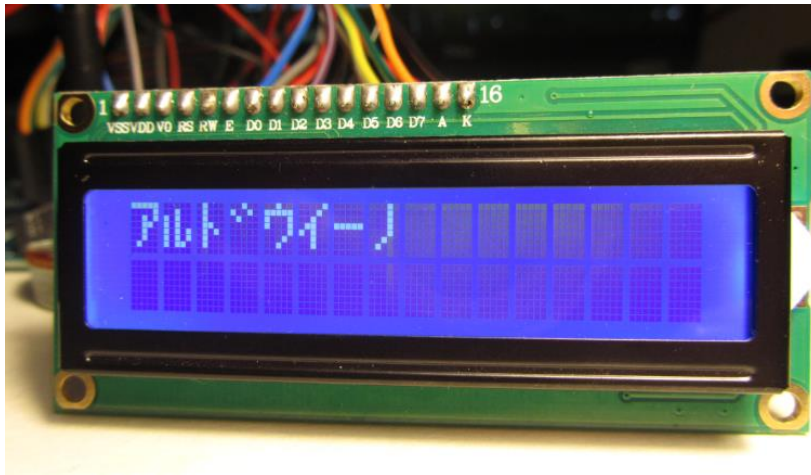


Proovige sõne (*string*) (vt „bik“, „harjutus“) asemel või siis koodiga tähe väljatoomise (vt 'ö' andmine koodiga 239) asemel nüüd lisada üksik täht meetodiga `LCD_ekraan.print('b')`. Kuidas toimis? Mängige, sest nii saab asi selgemaks.

Liquid Crystal Library <https://www.arduino.cc/en/Reference/LiquidCrystal> töötab nendel LCD-del, mis on rajatud Hitachi HD44780 kiibile või sellega sarnaselt toimivale. Veendusime, et taas tegi teegi kasutamine meie töö lihtsamaks.

### **Kõrvalepõige**

Kuna tegu on jaapani tehnoloogiaga, siis ei ole imestada, et HD44780 „mõistab“ ka jaapani keeles suhelda. Vaata [http://www.electronic-engineering.ch/microchip/datasheets/lcd/lcd\\_data\\_sheets.html](http://www.electronic-engineering.ch/microchip/datasheets/lcd/lcd_data_sheets.html) sümbolite tabelit: selles sisalduvad jaapani katakana (<https://en.wikipedia.org/wiki/Katakana>) silptähestiku tähed. Mina valin endale LCD-ga mängimiseks nende seast ekraanile järgmised アルドゥイーノ.



Kusjuures 卜 (*do*) olin sunnitud jagama kaheks sümboliks: esiteks kirjutan 卜 (*to*) ja siis lisan kaks kriipsukest, mis teeb *to*-st *do*.

Mida temaatilist olen ma jaapani keeles siia kirjutanud?

#### **11.4. Koduseks tööks**

Kui peatükkis ei jõudnud piisavalt katsetada, siis pange kodus skeem taas kokku ja mängige— kirjutage sõnu ja sümboleid. Järgmises peatükkis jätkame LCD võimaluste uurimist.

\*\*\*



## Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Paul Elberg,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

**Arduino näitel põhinev riistvara programmeerimise valikkursus  
gümnaasiumiastmele**

mille juhendaja on Alo Peets, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Paul Elberg  
10.08.2020