

TARTU ÜLIKOOL
Loodus- ja täppisteaduste valdkond
Tehnoloogiainstituut
Arvutitehnika õppekava

Alex Nõomaa

KAUGLABOR LEGO MINDSTORMS EV3 BAASIL

Magistritöö (30 EAP)

Juhendaja:
Kaido Reivelt, PhD

Tartu 2016

Resümee

Loodus-, täppisteaduste ja tehnoloogia valdkonna õppe hädavajalik osa on laboripraktika – praktilised oskused saavutatakse nende nimest johtuvalt läbi kogemuste, mida praegu pakuvad õppurile praktikumid, ekskursioonid ja töö laboris. Võimalus katseid sooritada Interneti vahendusel on sisuliselt küll olemas, kuid praktikas kardetakse nende liigset keerukust ja olemasolevad kauglaborid ei leia täna laialdast kasutust. Kombinatsioon reaalistest katsevahenditest, lairiba-andmesideühendusest ja õppeaine efektiivsest pedagoogilisest ülesehitusest parandaks sellegipoolest õppuri võimet mõista looduslike protsesse.

Antud magistritöö raames antakse ülevaade kaug- ja robotikalaborite valdkonnast, õppeotstarbelise robotikakomplekti LEGO Mindstorms seadmetest ning nende levikust Eesti haridusasutustes, samuti tuuakse välja loodusainete õpetamises Vernier ja Pasco andmekogujate kasutamine ning levik koolides.

Peatükkides 4 ja 5 analüüsitakse ning rakendatakse erinevaid võimalusi kauglabori juhtimiseks Interneti vahendusel ja põhjendatakse läbiproovitud lahenduste ebaõnnestumised.

Käesoleva töö käigus valmis LEGO Mindstorms EV3 juhtmoodulist, servomootoritest ja EV3 ning Vernier anduritest koosnev seade koos leJOS tarkvaral töötava kauglabori töö juhtimiseks kasutatava interaktiivse serverprogrammiga. Peatükis 6 antakse lühike ülevaade valminud komplektist, loetletakse süsteemi puudused koos parandusettepanekutega ning antakse mõned soovitusel kauglabori praktikumitööde koostamiseks.

CERCS P170, T125

Märksõnad: kauglabor, LEGO Mindstorms, EV3, Android, leJOS, http server, kaugjuhtimine, Vernier andur

Abstract

Laboratory practice is an essential part of science, technology, engineering and math education. Currently, students can get experience from labs, field trips or working in a laboratory. Although there is a possibility to perform lab tasks over the Internet, it is considered to be too complicated and existing remote labs are not widely used. The combination of real systems, high-bandwidth data connections and effective pedagogic design would help students' ability to understand natural processes.

This master's thesis gives an overview of remote and robotics labs, the educational robotics kit LEGO Mindstorms in Estonian academic institutions and points out the use of Vernier and Pasco data acquisition devices in teaching natural sciences.

Chapters 4 and 5 analyse and apply various solutions for controlling remote labs over the Internet.

In the course of this work, a LEGO Mindstorms EV3 module was equipped with servos as well as EV3 and Vernier sensors. The EV3 brick is running leJOS software and an interactive server program for controlling the remote lab. Chapter 6 gives a short overview of the kit and a few proposals are given for creating remote lab tasks.

CERCS P170, T125

Keywords: remote lab, LEGO Mindstorms, EV3, Android, leJOS, http server, remote control, Vernier sensor

Sisukord

Resümee	2
Abstract	3
Kasutatud mõisted ja lühendid	5
1 Sissejuhatus	6
2 Ülevaade seni loodud kauglaboritest	7
2.1 Kohvripraktikumid Tartu Ülikoolis	7
2.2 Ülikooli ja keskkooli ühine LEGO-praktikum Peterburis	8
2.3 “iLab” Stanfordinis	8
2.4 Taastuenergia kauglaborid Madridi Avatud Ülikoolis	9
3 LegoKaugLabor	9
3.1 Tark- ja riistvaraplatvorm LEGO Mindstorms	10
4 Kauglabori server Androidil	12
5 Kauglabori server LEGO Mindstorms robotil	14
5.1 Alternatiivpüsivara	14
5.2 Kauglabori riistvara	17
5.3 Kauglabori tarkvara	20
5.4 Veebileht roboti juhtimiseks	20
6. Tulemused	22
6.1 Ülesandeid laboris	23
6.2 Edasine töö	24
Kokkuvõte	25
Summary	26
Viited	27
Lisad	30
Lisa 1 API päringute nimistu	30
Lisa 2 Kasutajaliidese veebileht, index.html	32
Lisa 3 SimpleWebServer.java muutujad ja meetod String apiEndpoint	34
Lisa 3 VernierAnalogSensor.java	40
Lisa 4 LEGO-kauglabori programmikood	41
Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	42

Kasutatud mõisted ja lühendid

API	vt rakendusliides.
Kauglabor	Tark- ja riistvarakogum, mis võimaldab eksperimendi läbiviijast erinevas asupaigas olevate reaalsete seadmete ja katsevahendite kasutust arvutivõrgu vahendusel. Oluline on eristada kaug- ja virtuaallaborit, viimane kasutab laboriseadmete simuleerimiseks virtuaalreaalsust, Flash, Java või muud tarkvara ning selle raames ei viida läbi tegelikku eksperimenti. [1]
Pistikprogramm	ka plugin – Tarkvaramoodul, mis lisab suuremale süsteemile teatud omaduse või teenuse. [2]
Püsivara	(ingl <i>firmware</i>) Talitluslikult põhimälust sõltumatul viisil salvestatud käsu- ja andmekogum. [2]
Rakendusliides	ka programmiliides, API-liides. Rakendusprogrammiga määratud reeglistik, mille alusel rakendusprogramm kasutab operatsioonisüsteemi või teise rakendusprogrammi teenuseid. [2]
URI	(ingl <i>Uniform Resource Identifier</i>) – ühtne ressursi-indikaator. Internetis leiduvate tarkvaraliste võrguressursside unikaalne indikaator, mis näitab, kuidas seda ressursi kätte saada. [2]

1 Sissejuhatus

Laboripraktika on loodus-, täppisteaduste ja tehnoloogia (LTT) valdkonna õppe hädavajalik osa. Teoriaõpe ei anna õppurile kogemust eksperimendi kavandamise, läbiviimise ja tulemuste analüüsimise osas ega tutvusta muid teadustöö olulisi osi nagu näiteks katse ebaõnnestumine. Võimalus sooritada katseid praktikumiruumide või seadmete puudumisel Interneti vahendusel kauglaborites on sisuliselt küll olemas, kuid praktikas ei leia see kuigi palju kasutust. Koolides kardetakse peamiselt liigset keerukust ja pedagoogi suurt ajakulu süsteemi selgeksõppimisel. Kombinatsioon reaalistest süsteemidest, lairiba-andmesideühendusest ja efektiivsest pedagoogilisest ülesehitusest õppeaines parandaks sellegipoolest õppuri võimet mõista looduslikke protsesse. [3]

Euroopa Komisjoni rahastatavas kaug- ja virtuaallaborite portaalis GO-LAB on käesoleva töö valmimise ajal samas kogu maailmast esindatud vaid napilt üle viiekümne kauglabori, vähesed neist on ka töökorras. [4] Töökindla, kättesaadava ja paindliku riist- ning tarkvaraplatvormi puudumine taoliste laborite ehitamiseks on kindlasti üheks kauglaborite väikese arvukuse põhjuseks, laboriseadmed konstrueeritakse universaalse platvormi ja nõudluse asemel üksikkorras ja vastavat võimalustele.

Eestis 2007.a kevadel loodud koolirobootika programmiga muudetakse õpilaste jaoks matemaatikat, informaatikat ning teisi reaalseid robotite abiga huvitavamaks ning pannakse neid omandatud teadmisi kasutama robootikaprojektide teostamisel. Õpetajatele korraldatakse koolitusi ning tagatakse õppematerjalide olemasolu, kooliroboti projekt on 2016. aasta maikuu seisuga toonud LEGO robootikakomplektid käibele üle 260 Eesti haridusasutusse. [5]

Käesoleva töö eesmärgiks on luua lihtne ning interaktiivne tark- ja riistvarakomplekt õppeotstarbeliseks laboriülesannete sooritamiseks Interneti vahendusel. Oluliseks funktsionaalsuseks peetakse servomootorite juhtimist, katsemõõtmiste tegemist

mitmesuguste andurite abil ning võimalust laboris toimuvat jälgida videopildi vahendusel, samuti on eesmärk võimaluse piires kasutada ära olemasolevat riistvara ja pakkuda kirjeldatud lahendust kergesti kohaldatavana erinevates õppeasutustes.

2 Ülevaade seni loodud kauglaboritest

Kauglaborite loomisel ja rakendamisel eksisteerib kaks põhimõtteliselt erinevat lähenemist. Kasutada saab soodsaid ja lihtsaid seadmeid, mis võimaldab edukalt toimivat kauglaborikomplekti edaspidi kergesti replitseerida ka teistes õppeasutustes, teise variandina luuakse kallid süsteemid ja kompleksed rakendused. Suurem investeering võimaldab kasutusse võtta laiema valiku sensoreid ja aktuaatoreid ning seeläbi soodustab kauglaborite ühiskasutust mitmete osakondade ja õppeasutuste vahel mitte ainult kulude, vaid ka teabe jagamise otstarbel.

Soodsad ja madala keerukusastmega süsteemid võimaldavad hinnata reaalseid üksikasjalikke vajadusi kauglaborite kasutamisel enne kulukate keeruliste kauglaborite loomist. [6]

Erialakirjandusest võib leida hulgaliselt näiteid interneti vahendusel kaugjuhitavatest robotitest, mitmed neist on autonoomsed ja küsivad inimese sekkumist vaid keerulisemates situatsioonides, kuid reeglina on tegu tööstuslike või väga spetsiifiliste robotitega. LEGO Mindstorms robotikakomplektide kasutusvaldkond keskendub põhiliselt hariduslikule suunale, nende seadmete rakendusi interneti- või kohtvõrgu vahendusel uuritud praktiliselt ei ole. [7]

2.1 Kohvripraktikumid Tartu Ülikoolis

Automatiseeritud praktikumitöödest, mida tudeng saab vastaval internetilehel sooritada temale sobival ajal, on Tartu Ülikoolis aine LOFY.03.005 Elektrimõõtmised raames kasutusel kodulaborid. Elektriliste suuruste mõõtmise ja mõõtevahendite

kasutamise kogemuse saamiseks on tudengil kasutada elektriskeemi komponentide ja mõõtevahendite komplektid ning praktikumi veebikeskkonnas interaktiivsed tööde juhendid. Töö tulemust ning õpitu omandamist kontrollitakse mõõtmistulemusi arvestavate testide abil.

2.2 Ülikooli ja keskkooli ühine LEGO-praktikum Peterburis

Peterburi Riikliku Ülikooli ning 239. Füüsika- ja matemaatikalütseumi ühise pilootprojektina rajati 2008. a sügissemestril esimesed “küberfüüsika laborid”. Pilootprojekti tulemuste analüüsil selgus, et robotikapraktikumid toimisid kõige paremini kahe haridusastme koostöona. Matemaatika-mehaanikateaduskonna tudengid olid tugevad programmeerimises, ent jäid hätta robotite tehnilise poole teostamisel, samal ajal kui keskkooliõpilased ehtasid küll paremaid konstruktsioone, kuid programmeerimisalast oskusteavet neil nappis. Projekt näitas, et LEGO Mindstorms NXT robotikakomplektid on mehaanika ja programmeerimise õpetamiseks sobilik ja eelarvesõbralik lahendus. Peterburi küberfüüsika laborite projekti kaugem eesmärk on luua LEGO Mindstorms NXT põhised robotika ja mehhatroonika kauglaborid kesk- ja ülikoolides rakendamiseks. [8]

2.3 “iLab” Stanfordis

Prof. Lambertus Hesselinki juhtimisel on Stanfordin Ülikoolis juba 1996. aastast arendatud automatiseeritud kauglaboreid, kus end autoriseerinud üliõpilastel on võimalik laboriseadmetega katseid sooritada. Kuna katseseadmed on automatiseeritud ning ei pea olema inimkäsitsetavad, on võimalik nende mõõtmeid vähendada ja hoida sel viisil võrreldes traditsioonilise laboriruumiga märkimisväärselt kokku ruumi- ja jooksvaid kulusid: sellised “kaugjuhitavad praktikumiruumid” on vaid mõnekümne liitri suurused.

Kaugjuhitavad automatiseeritud laborid on sobilikud madala osalejate arvuga kursustele, ent sadade tuhandete osavõtjatega massikursustel tekiksid tudengite seas aastatepikkused järjekorrad. Massikursuste praktikumidel talletatakse kõik

võimalikud tingimuste kombinatsioonid ning nendele vastavad tulemused, ehk virtuaalse labori kasutajale näidatakse tema käskudele vastavat eelnevalt salvestatud pilti. Niiviisi elimineeritakse laboris liikuvad osad ja seeläbi ka ooteajad, laborit saavad erinevate katsete tegemiseks samaaegselt kasutada mitmed inimesed. [9]

2.4 Taastuenergia kauglaborid Madridi Avatud Ülikoolis

Kaugõppes, kus tudengite osavõtt õppeainetest on enamasti virtuaalne, on praktiline õpe eraldi väljakutse. Kauglabor võimaldab reaalsete vahendite kasutamist üle interneti, vabalt valitud ajal ja kohast, ning on samas rakendatav ka ettevalmistava etapina enne füüsilist praktikumi. UNED (*Universidad Nacional de Educacion a Distancia* e Hispaania riikliku kaugõppekõrgkooli) elektro- ja arvutitehnika osakonna kaugjuhitava praktikumi Mindstorms NXT juhtmooduliga robotid on varustatud LEGO taastuenergia lisakomplektis oleva energiamõõdiku ning tuuliku ja väikese dünamo või päikesepaneeliga. Roboteid juhitakse labori serveris töötava LabVIEW tarkvara ja RESTful veebiteenuste abil, laboriruumis toimuvast saab visuaalse ülevaate läbi veebikaamera. Praktikumitöö käigus sooritatud ülesannete ja kogutud andmete põhjal tehakse iseseisev kodutöö. [6]

3 LegoKaugLabor

Kuivõrd paljudes Eesti koolides on LEGO robotikakomplektid ja seeläbi ka osa labori seadmestikust juba olemas, vaadeldi laboriseadmena lähemalt LEGO Mindstorms robotikakomplekte.

Eesti üldhariduskoolides on veel vähesel määral kasutusel LEGO Mindstorms NXT komplekte, kuid uue põlvkonna EV3 seadmed on neid jõudsalt välja vahetamas – 2015.aastal hangiti ainuüksi Hariduse Infotehnoloogia Sihtasutuse kaudu koolidele 112 baaskomplekti ja hulgaliselt lisasid ning 2016.aastal avaldati riigihange veel 277 EV3 baaskomplekti ostmiseks [10], [11], seetõttu keskendub ka käesolev töö peamiselt EV3 komplektile.

Loodusainete õpetamises on praktilise osa suurendamiseks levinud praktika andmekogujate nagu Pasco ja Vernier kasutamine, õpilased sooritavad laboratoorse töö käigus katsemõõtmised ning arvutused ja analüüs tehakse mõõtetulemuste põhjal. Eesti koolides on valdavalt Vernier seadmed ja nendele sobivad andurid, näiteks 2010.aastal hangiti Haridusameti kaudu 336 andmekogujat [11], enamasti soetavad koolid neid ise. Loodava kauglabori huvi on võimalusel ära kasutada olemasolevat ressursi, seetõttu ei piirdu töö LEGO originaalanduritega.

3.1 Tark- ja riistvaraplatvorm LEGO Mindstorms

Programmeeritavate LEGO robotite ajalugu ulatub aastasse 1998, kui tuli välja esimene Mindstorms seeria komplekt RCX. Eestis saavutas LEGO-robotika laiema populaarsuse suuresti Kooliroboti projekti toel teise põlvkonna komplektiga NXT, mille esimene versioon oli saadaval aastast 2006 ja NXT 2.0 aastast 2009. [12]

2013. aasta sügisel tõi LEGO turule uue robotikakomplekti Mindstorms EV3, millega kaasnesid senisest rohkemate võimalustega juhtmoodul, kahes erinevas suuruses mootorid ja laialdasem valik andureid. [13]

Mitmesuguste laboriplatvormide seas, millega kontrollsüsteemide meetodeid rakendada, paistab LEGO Mindstorms silma vähemalt kolme eelisega:

LEGO on levinud mängukomplekt, seega on õpilastel ja tudengitel olemas baasteadmised selle kasutamisest ja ehitusmeetoditest ning nad on roboti konstrueerimiseks valmis juba kursuse algfaasis. Teiseks võimaldavad komplektid juhtimiseks hõlpsat juhtmevaba Bluetooth- või WiFi-ühendust, ning lõpuks räägib Mindstorms komplektide kasuks nende suhteliselt soodne hind. Mindstorms NXT baaskomplektis sisaldub nelja sisendi ja kolme väljundiga 32-bitine ARM7 põhine NXT juhtmoodul, kolm aktuaatorit (servomootorid), kauguse-, heli-, valguse- ja puuteandurid ning aku. EV3 baaskomplekti maksumus on €428 [10], sellesse kuulub nelja sisendi ja nelja väljundiga ARM9 põhine juhtmoodul, kolm aktuaatorit (servomootorid), värvi-, güroskoop-, kaugus- ja puuteandurid ning laetav aku. Koos arvukate LEGO osadega moodustavad need väga erisuguste robotite jm manussüsteemide ehitamiseks igati sobiliku komplekti. [8]

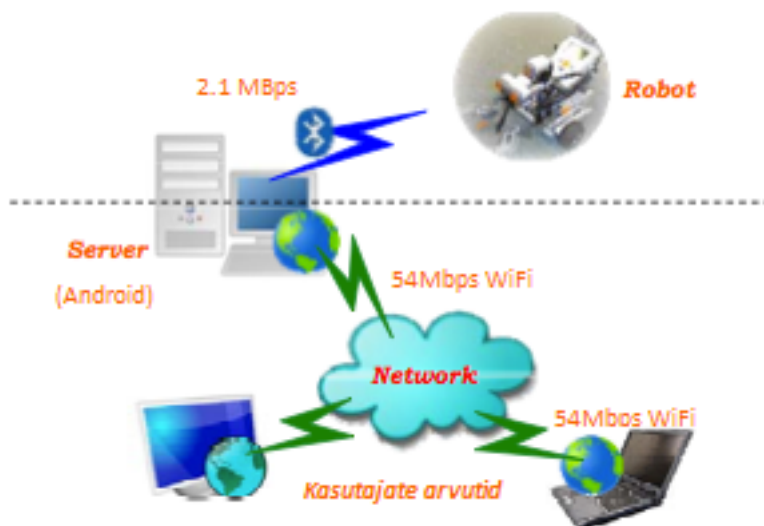
LEGO Mindstorms NXT ja EV3 robotitega on võimalik ühendada Vernier sensoreid, kasutades selleks vastavat BTA-NXT adapterit, mis võimaldab märkimisväärselt laiendada läbiviidavaid eksperimente ning tõsta kauglaborite loomise kasutegurit. Ametlikult on Mindstorms robotitel toetatud üle kolmekümne erineva Vernier sensori, sealhulgas amper- ja voltmeeter, magnetvälja-, temperatuuri- ja UV andurid ning hulk vedelikes kasutatavaid andureid elektrijuhtivuse, soolsuse, happelisuse, lahustunud hapniku ja muu mõõtmiseks. [14]

Mindstorms EV3 roboti liikumise arvestamisel on abiks servomootorites olev 1-kraadise resolutsiooniga analoog-digitaalmuundur. Suurema mootori pöörlemiskiirus ilma koormuseta ja 9V toiteallikaga (so 6x1.5V uued patareid) on 175 p/min ning aeglane hammasülekanne võimaldab 17.3 N.cm pöördemomendi, väiksema mootori näitajad on vastavalt 260 p/min ning 6.64 N.cm. EV3 juhtmooduliga on võimalik kasutada ka NXT komplekti mootoreid koormuseta pöörlemiskiirusega kuni 170 p/min ja pöördemomendiga 16.7 N.cm. [15] Seega pakub LEGO komplekt paindliku lahenduse nii kiiruse kui jõumomendi osas.

Levinuim LEGO Mindstorms NXT robotitel kasutatav programmeerimiskeel on graafiline NXT-G ning EV3 robotitel EV3-G, mõlema interpretaator baseerub National Instruments LabVIEW tarkvaral. Lisaks on LabVIEW tarkvara jaoks võimalik alla laadida LEGO juhtmoodulite programmeerimiseks ja juhtimiseks vajalikke laienduspakette. Käesoleva kauglaboriprojekti juhtimine oli algselt kavandatud LabVIEW keskkonna põhiseks, kuid et süle- või lauarvuti kasutamisel serverina lisanduksid serveri riistvara maksumusele mitmesugused litsentsitasud operatsioonisüsteemi ja LEGO mooduli juhtimise tarkvara kasutamise võimaldamiseks ning see seab küllalt kõrged nõudmised ka labori riistvarale ja valmis komplekti kasutuselevõtmise eelarvele, asuti otsima alternatiivseid lahendusi. Labori kaugjuhtimise probleemi prooviti lahendada kahel platvormil, LEGO juhtmoodulil ning eraldiseisval Android seadmel.

4 Kauglabori server Androidil

Esimese lahendusena valiti kauglaborirakenduse tarkvaraplatvormiks vabavaraline ja avatud lähtekoodiga mobiilne operatsioonisüsteem Android. Sellise lahenduse eeliseks on seadme hea kättesaadavus ja suhteliselt madal hind, samuti mobiilsus ning võrreldes süle- või lauarvutiga ka väga väikesed mõõtmed. Lisaks on Android tarkvaraga nutitelefon või tahvelarvuti paljudel kasutajatel juba olemas, mistõttu kauglabori ülesseadmiseks näiteks koolides piisab vaid olemasolevasse seadmesse rakenduse laadimisest. LEGO Mindstorms robot on läbi serveri kaugjuhitav Bluetooth ühenduse abil, kusjuures server on kohtvõrku ühendatud WiFi liidese kaudu. Serveri veebiliideselega ühendumiseks ja roboti juhtimiseks kasutatakse internetivõimekusega arvutit (klient), nagu on näha ka joonisel 1. Klientarvuti võib füüsiliselt paikneda robotist ja laborist geograafiliselt eemalasuvas kohas, mõnes teises õppehoones, edukalt toimis näiteks videopildi edastus TÜ füüsikahoone ning TÜ raamatukogu vahel.



Joonis 1. Laborikomplekti ühendusprotokollid [7]

Kauglabori komplekti töös on selle lahenduse korral seadmetevahelises andmesides kasutusel kolm erinevat võrguliidest: traadita Bluetooth jadaühendus roboti ja serveri

vahel, traadita internetiühendus serveri ja WiFi pääsupunkti vahel, ning kaabliga kohtvõrguühendus või traadita internetiühendus WiFi pääsupunkti ja kasutaja arvuti vahel. Serveri ehk Android telefoni või tahvelarvuti roll on olla laborirobotiga samas ruumis ja vajadusel sellega kaasa liikuda ning vahendada kasutajalt robotile käsklusi liikumiseks ja mõõtmiseks, robotilt kasutajale tagasisidet mõõtetulemuste kohta ning anda toimuvast ka videoülevaade. Kasutaja jälgib ning juhib protsessi veebibrauseri vahendusel Android seadme sisevõrgu IP-aadressil.

Veebiserveri mooduli rolli täidab kauglabori rakenduses minimalistlik, vaid ühest *Java* failist koosnev HTTP server *NanoHTTPD*. See avatud lähtekoodiga tasuta tarkvara on mõeldud teiste rakenduse siseseks kasutamiseks ning on implementeeritav *Modified BSD* litsentsi alusel, mis lubab lähtekoodi ja tarkvara kasutada ning modifitseerida tingimusel, et säilib viide koodi algsele päritolule. [16] Osaliselt kasutab kauglabori rakendus videopildi edastamisel *GPL v2* litsentsi alusel levitatava *Android-Eye* nimelise programmi lähtekoodi. Nutiseadmes töötav rakendus serveerib telefoni või tahvelarvuti kohtvõrguaadressil veebilehekülje, millel saab jälgida seadme kaamera edastatavat videovoogu ning saata käsklusi roboti servomootoritele ja anduritele.

Androidi rakendustepoes on saadaval kümneid erinevaid Mindstorms NXT ja EV3 robotite üle Bluetoothi kaugjuhtimiseks mõeldud programme. Kauglabori serveri rakenduse loomist alustati videoülekanne funktsionaalsusest ning plaaniti lisada andurite ja mootorite juhtimine. Kaugjuhtimisliides serveriga Androidil osutus mitmes plaanis probleemseks. Testitud nutitelefoniga pea kogu arvutusvõimsus kulus kaamerapildi dekodeerimisele ning juba ligikaudu veerandtunnise töö järel kuumenes seade üle, tekkisid tõrked seadme töös ja info edastamisel WiFi võrku. Kui käesoleva töö alguse ajal oli testtelefoni operatsioonisüsteemi versioon 4.3 värskem saadolev, siis vaid aasta hiljem ei saanud samasse seadmesse laadida isegi eelviimast tarkvara. Uute Androidi versioonide väljatuleku kõrge sagedus eeldab aktiivset haldustegevust ka pärast rakenduse valmimist. Bluetooth liidese kaudu roboti juhtimise lisamist ning selle dubleerimist videopilti edastaval veebilehel ei peetud piisavalt töökindlaks ega ratsionaalseks. Eeltoodust lähtuvalt tunnistati Androidi serveri lahendus töö käigus ebaõnnestunuks.

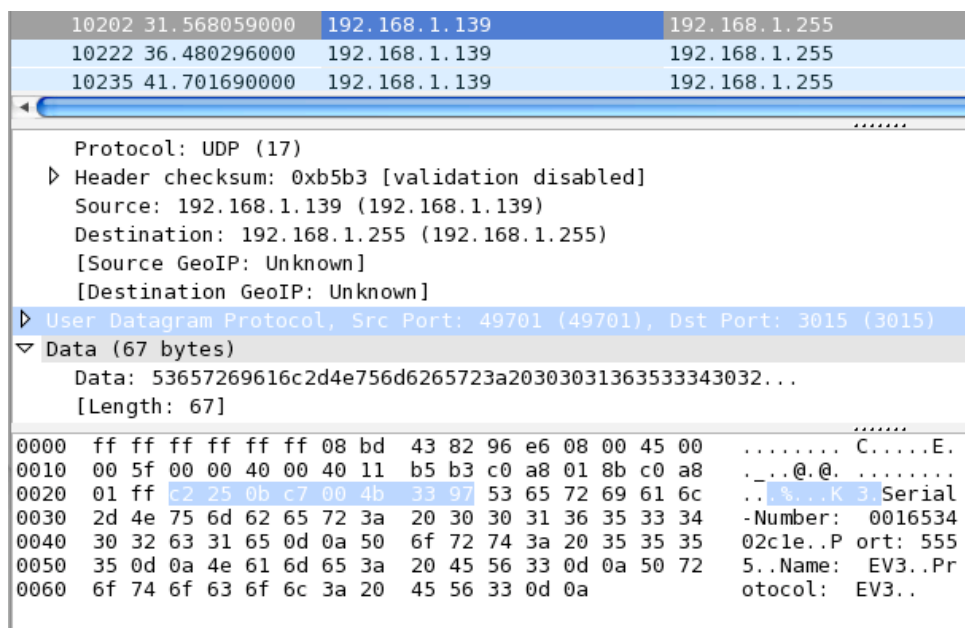
5 Kauglabori server LEGO Mindstorms robotil

Kuivõrd videopildi, veebilehe ja juhtimiskäskluste edastamine ühe mõistliku hinnaga laiatarbeseadme abil ei näidanud soovitud töökindlust, kadus vajadus roboti ja kasutaja vahel oleva eraldiseisva universaalse seadme järele ning teise meetodina prooviti püstitatud ülesanne lahendada LEGO juhtmoodulis. Videopildi edastamine lahendatakse eraldiseisva IP-kaameraga, milliste maksumus jääb töö valmimise ajal reeglina allapoole saja euro piiri. Mindstorms EV3 robotikakomplektil on USB lisaseadme abil WiFi võimekus, seega puudub vajadus aeglasema ja väiksema tööulatusega Bluetooth ühenduse kasutamise järele. Teoreetiliselt on ülesanne sedasi lahendatav ka eelmise põlvkonna NXT juhtmoodulil, mõningaste modifikatsioonidega riistvaras on leJOS platvormil NXT roboteid ühendatud traadita võrku ja sarnaselt käesoleva töö tulemusega juhitud, ent käesoleva projekti raames peeti seda võimalust ebaratsionaalselt keerukaks ning EV3 komplektide kiiresti kasvava arvukuse tõttu ebaotstarbekaks. [17] Originaalpüsivaras on WiFi funktsionaalsus piiratud vaid LEGO graafilises EV3-G programmeerimiskeskonnas loodud programmide edastamiseks juhtmoodulisse ja vaikimisi seadetega robot ei ole väljastpoolt LabVIEW-põhist programmi juhitud. Asuti otsima võimalust robotiga ühendumiseks ilma LEGO tarkvarata.

5.1 Alternatiivpüsivara

Standardse LEGO püsivara kõrval eksisteerib mitmeid alternatiivseid tarkvarapakette, neist levinumad on kahe noore taanlase projekt MonoBrick ja laiema kogukonna poolt arendatav leJOS. Mõlemad võimaldavad LEGO Mindstorms NXT või EV3 juhtmooduliga roboti kontrollimist üle interneti. LEGO originaalpüsivarale lisatav *MonoBrick Communication Library* on C# keeles kirjutatud sideteek, mis muuhulgas võimaldab robotiga USB, Bluetooth, WiFi või ühendustunneli kaudu suheldes saata sellele käsked ja päringuid, sõnumeid, faile, toetab mitmeid järelturu andureid jm. [18]

Monobricki sideteekide eeliseks on töötamine LEGO originaalpüsivaral, seega saab sama EV3 juhtmoodulit kasutajale harjumuspärase tarkvaraga lisaks kauglaborile kasutada ka muudel otstarvetel. Sideteekide abiga saab roboti juhtmooduliga käske vahetada, kuid et EV3 on tavaolekus välisühendustele suletud, siis tuleb see iga sessiooni alustamisel seadistada käske ja päringuid vastu võtma. Selleks tuleb kuulata pealt võrguliiklust, saamaks teada, mis pordil Monobrickiga EV3 moodul parajasti ühendust ootab ning saata robotile õigetes portidesse mõned UDP ja konkreetse juhtmooduli seerianumbrit sisaldavad TCP andmepaketid. *Wireshark* tarkvara kuvatömmisel (joonis 2) näidatud juhul kuulutab EV3 moodul pordilt 49701 oma seerianumbrit (001653402c1e), ühendusporti (5555) ja ühendusprotokolli (EV3). Robotiga ühendumiseks tuleb kõigepealt saata lähteporti suvaline UDP andmepakett ning seejärel kuulutuses näidatud ühendusporti 5555 TCP/IP sõnum sisuga *GET /target?sn=001653402c1e VMTP1.0 Protocol: EV3*



Joonis 2. Võrguliikluse kuvatömmis (*Wireshark*): EV3 juhtmooduli sõnum

Kui robot vastab nõusolekut väljendava “*accept*” sõnumiga, on ühendus loodud ja robotile saab üle WiFi käske ja päringuid saata ehk hakata laboriülesandeid täitma. Peamiselt just igakordse taasühendamise keerukuse tõttu osutus nimetatud lahendus kauglabori jaoks ikkagi ebasobivaks, kuna ei ole mõeldav, et laborikomplekti iga käivitamine on niivõrd keeruline ning selles osaleb ka vastava pädevusega isik.

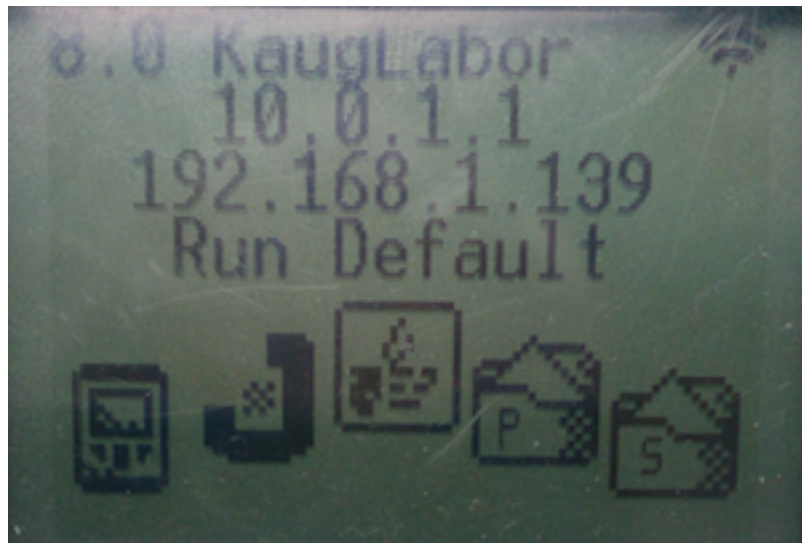
MonoBrick EV3 Firmware asendab LEGO originaalpüsivara ning võimaldab EV3 moodulit programmeerida .NET programmeerimiskeeltes nagu C#, F# ja IronPython. Funktsionaalsusesse kuuluvad LEGO ja HiTechnic andurite tugi ning muuhulgas UART, I2C ja analoogsensorite baasklassid, mootorite juhtimine PID kontrolleri ja nende sünkronimine, samuti SQLite tugi. [19]

Monobrick püsivaraga microSD mälukaardi tegemine ja roboti laboriülesandeks kasutusvalmis seadmine on sideteekide ühenduse avamisest küll märkimisväärselt lihtsam, kuid tarkvara WiFi-funktsionaalsuses ilmnesid probleemid traadita kohtvõrkude mälu hoidmisega ning pärast roboti iga taaskäivitust tuli uuesti sisestada nii võrgu nimi (SSID), parool kui krüpteerimise tüüp. Samuti osutus Monobricki püsivara EV3 seadmel väga aeglaseks, lisaks võrguliiklusest tingitud viidetele toimusid roboti kõik liigutused isegi EV3 füüsiliste nuppudega juhtides ligi sekundilise viivitusega. Kaugjuhitavaid mõõtmisi tehes ei ole usaldusväärsete tulemuste saamine sellisel viisil mõeldav ja nii tunnistati Monobricki arendatav EV3 alternatiivpüsivara kauglabori jaoks ebasobivaks.

Samuti originaaltarkvara asenduseks mõeldud leJOS (*LEGO Java Operating System*, ka viide esimese põlvkonna Mindstorms robotite tarkvarale leGOS) on väike Java virtuaalmasin, mis baseerub kunagisel *TinyVM* projektil. Nii TinyVM kui leJOS loodi algselt Mindstorms RCX robotile, 2006.aastal porditi see NXT juhtmoodulile ning 2013.aastal ka EV3 plokile. Kõigi kolme näol on tegemist Java käituskeskkonnaga, mis sisaldab vastava juhtmooduli rakendusliidese klasse ning võimaldab objektorienteeritud programmeerimist (Java), toetab massiive, rekursiooni, sünkroniseerimist, Java andmetüüpe nagu *float*, *long* ja *String*, ja muud [20], ning selle robotika API on väga hästi dokumenteeritud. EV3 moodulil on võimalik leJOS käitada microSD mälukaardilt, ilma originaalpüsivara üle kirjutamata. [21]

Läbi töötatud lahendustest osutus kõige töökindlamaks ja kasutajasõbralikumaks leJOS, töö valmimise ajal on tarkvara värskeim versioon 0.9.1-beta. Java keskkond võimaldab kasutada lihtsat http-serverit ning põhjalikult dokumenteeritud EV3 API-liides tagab mootorite ja andurite korrektse juhtimise, üle neljasaja leJOS sisese eelimplementeeritud klassi on saadaval andmepaketis *ev3classes.jar*. Kauglaboriks

seadistatud komplekti kasutamahakkamiseks on vajalik vaid vaikimisi programmi käivitamine joonisel 3 näidatud roboti ekraanimenüüs.



Joonis 3. leJOS avakuva, vaikimisi programmi käivitamine

Sarnaselt Android seadmel töötavale kauglabori serverile täidab leJOS EV3 juhtmoodulis veebiserveri rolli minimalistlik *NanoHTTPD*. Nädisimplementatsiooni *SimpleWebServer* vaikimisi toetatud veebilehitsejad on *Firefox*, *Chrome* ja *Internet Explorer*, mis peaks tagama kauglabori probleemideta töö enamikes arvutites. [16] Testimisel ei ilmnunud tõrkeid ka veebilehitsejaga *Safari*, samuti toimivad edukalt nutiseadmete brauserid.

5.2 Kauglabori riistvara

Kauglaborikomplektis kasutatav LEGO Mindstorms EV3 juhtmooduliga robot on varustatud LEGO puute-, kauguse- ehk ultrahelianduri ning LEGO värvianduriga ja kahe Vernier anduriga – jõu andur ja diferentsiaalpinge andur ehk voltmeeter. Kuivõrd käesoleva komplekti üks eesmärke on ka tutvuda LEGO roboti ja andurite sobivusega kauglaboriseadmeks, ei ole konkreetse roboti füüsiline ehitus siinkohal oluline ning kasutatud on lihtsat LEGO baasrobotit. EV3 juhtmoodul töötab 32bitisel ARM9 põhisel Texas Instruments Sitara AM1808 protsessoril, seadmel on sisseehitatud 16 MB välmälu ja 64 MB muutmälu. leJOS

püsivara töötab käesoleval juhul 16 GB Sandisk microSD mälukaartilt, AM1808 MMC/SD kontrolleri toetus kuni 32 GB mälukaarte. [22]

EV3 toetab ametlikult vaid ühte WiFi adapterit, milleks on Atheros AR9271 kiibistikul töötav Netgear N150 (WNA1100). Uusima leJOS tarkvara versiooniga toimib lisaks sellele veel viis erinevat adapterit, nagu nähtub tabelist 4, kuid et Netgari seade toimib testide põhjal just käskude ja päringute ehk väikeste andmepakettidega kuuest võimalikust variandist kõige kiiremini ja testimisel sellega probleeme ei esinenud, on kauglabori kompleksis kasutusel WNA1100. [23]

Seade	Kiibistik
Netgear N150 (WNA1100)	Atheros AR9271
Digitazz 150Mbps Wireless adaptor	Ralink RT 5370
Pi Hut Raspberry Pi Wi-Fi adapter	Ralink RT 5370
CSL USB Wlan for Raspberry Pi	Realtek 8188cu
TP-Link TL-WN725N	Realtek 8188eu
Edimax EW-7811UN	Realtek 8188cus

Tabel 4. leJOS 0.9.1-beta versioonis toetatud WiFi adapterid [23]

LEGO kasutab Mindstorms komplektide mootorite ja andurite ühendamisel juhtmooduliga modifitseeritud RJ-12 pistikut, võrreldes levinud telefonikaabliga on lukusti nihutatud pistiku paremasse serva, nagu on näha joonisel 5. Digitaalsete LEGO sensorite lugem võetakse kuuendalt, SDA kontaktilt, EV3 puuteandur toimib analoogsensorina ja selle olek mõõdetakse pistiku esimeselt kontaktilt. [24] Vernier sensoritel on kasutusel BTA (*British Telecom Analog*) pistik, Mindstorms NXT või EV3 robotiga ühendamiseks on vajalik BTA-NXT adapter. Mõõtetulemus loetakse esimeselt, analoogisendi kontaktilt ning vajadusel teisendatakse korrektseks lugemiks programmi koodis. [25]



Joonis 5. LEGO Mindstorms ja Vernier BTA ühendused

Vernier DFS-BTA on eelkalibreeritud kahe-suunaline tõmbe- ja tõukejõu sensor mõõtetuladusega 0.01 N kuni 50 N. Kahe mõõtepiirkonna vahetamine toimub mehaaniliselt anduril olevast lülitist. [26]

Vernier madalpinge andur DVP-BTA mõõdab alalis- ja vahelduvpinget kahe klemmi vahel vahemikus ± 6.0 V. Pinge andur on samuti eelkalibreeritud, lugem arvutatakse vastavalt tootja ülekandefunktsioonile:

$$V_o = -0.4 (V_+ - V_-) + 2.5$$

kus V_o on pinge lugem ning $V_+ - V_-$ on vastavalt positiivse ja negatiivse potentsiaali vahe. [27]

Võimaldamaks kauglabori järjepidevat tööd ning samal ajal kvaliteetset otseülekannet videos, otsustati käesoleva töö raames videoedastus EV3 juhtmoodulist välja jätta. Labori veebilehel on viide ruumis asuva IP kaamera striimile, mis avaneb eraldi aknas ja seeläbi on kasutajal võimalik korraga ekraanil näha nii juhtnuppe ja mõõtetulemusi kui pilti laboriroboti hetke asukohaga ruumis.

5.3 Kauglabori tarkvara

Kauglabori programm koosneb järgnevatest klassidest

- `InternalRewrite.java` – NanoHTTPD kasutatavad reeglid
- `NanoHTTPD.java` – minimalistliku veebiserveri tuumklass
- `ServerRunner.java` – NanoHTTPD käivitamisel kasutatav eranditöötlus
- `SimpleWebServer.java` – NanoHTTPD serveri näidisimplementatsioon, millele on liidetud kauglabori funktsionaalsus
- `VernierAnalogSensor.java` – Vernier andureid defineeriv abiklass
- `WebServerPlugin.java` – NanoHTTPD kasutatav URI töötlemise pistikklass
- `WebServerPluginInfo.java` – NanoHTTPD kasutatav pistikklass

`SimpleWebServer`ile lisatud funktsioon `apiEndpoint` saab käsu kas veebilehe nupudelt või API vahendusel, viimane võimaldab roboti juhtimist mõne muu veebilehe või rakenduse vahendusel juhul kui selline vajadus peaks tekkima. Täielik ressursiindikaatorite nimekiri on toodud lisas 1.

Analoogsensorite juhtimisel kasutatav klassi `VernierAnalogSensor` ning peaklassi `SimpleWebServer` sisu on toodud lisades 3 ja 4, kusjuures `SimpleWebServer`i väljavõttest on eemaldatud valmistarkvarana kasutatud NanoHTTPD programmikoodi osa. Kogu tarkvaraprojekt on lisatud eraldi failis.

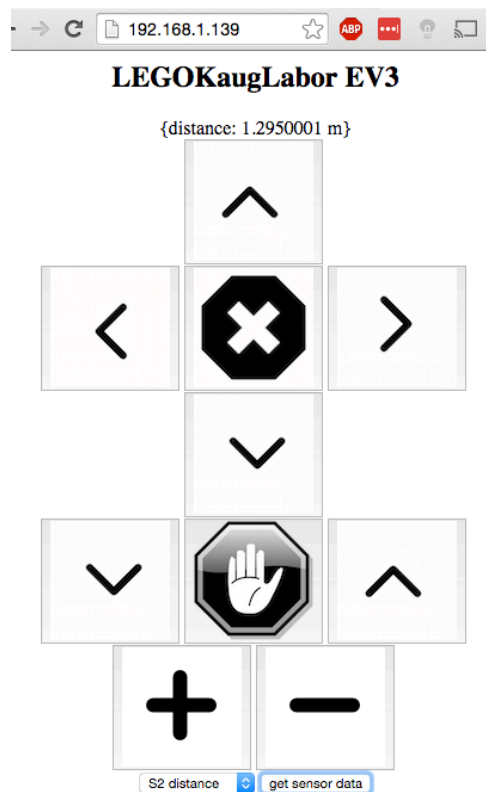
5.4 Veebileht roboti juhtimiseks

Laborikomplekti kaugjuhtimine toimub läbi EV3 moodulis töötava lihtsa html-veebilehe, mille kuvatõmmis on toodud joonisel 6.

Roboti sõidumootoreid B ja C juhitakse nelja suunanupuga, kolmanda mootori liikumissuund valitakse eraldi üles- ja allanoolega nuppudest. Mootorite pöörlemiskiirus on muudetav + ja – nuppudega skaalal 60 kuni 900 kraadi sekundis, mis vastab sõidumootorite 10-150 rattapöördele minutis. Sõidukiiruse seadistussamm on 60 kraadi sekundis ja iga uus väärtus kuvatakse nii roboti LCD-ekraanil kui veebilehel. Mootorite liikumiskiiruse vabalt valitud väärtusi saab käsitsi

sisestada rakendusliidese kaudu aadressil *http://roboti_ip/api/speed/x* kus “x” tähistab soovitud väärtust, vastavalt lisa 1 tabelis 7 toodud dokumentatsioonile on võimalik kasutada ka ülejäänud käske.

Käe ikooniga stopp-nupp lukustab mõõtmise käigus juhusliku veerema hakkamise välistamiseks mootorid nende nupu vajutamise hetkel olnud asendisse.



Joonis 6. Roboti juhtimise liides

Kommunikatsioon EV3 sensoritega on integreeritud leJOS platvormi, sarnaselt on võimalik kasutada ka NXT andureid. Anduritel lugemi võtmiseks valitakse hüpikmenüüst soovitud andur ning klikitakse nuppu “Mõõda”. EV3 puuteandur pordis 1 tagastab väärtuse “0” või “1”, EV3 ultraheliandur pesas 2 arvutab keskmise viielt mõõtmiselt ja annab tulemuse meetrites ning EV3 värviandur ühendatuna porti 3 vastab kolme väärtusega vastavalt punase, rohelse ja sinise värvuse osakaalu kohta mõõdetud punktis.

Vernier andurid on ühendatud analoogsensoritena. Pinge andur ühendatakse porti 3 ja jõu andur porti 4. Kummagi mõõtepiirkonna puhul on jõu anduri kalibreerimiseks

eelsalvestatud analoogsignaale vastav mõõteväärtus ilma koormusega ning kindlaksmääratud koormusega mõõtmistelt. Vastavalt tootja juhendile kalibreeritakse sensor 10 N mõõtepiirkonnas kolmesajagrammise ja 50 N piirkonnas tuhandegrammise raskusega. Vajadusel nullitakse lugem katse algseisu või abivahendite massi fikseerimiseks analoogselt digitaalkaaludel levinud “taara” funktsiooniga, vastav täiendav nupp on veebilehel nähtav vaid jõu anduri valimisel. Diferentsiaalpinge anduri signaali väärtus teisendatakse pinge lugemiks vastavalt punktis 5.2 toodud ülekandefunktsioonile.

6. Tulemused

Valminud kauglaborikomplekt koosneb LEGO Mindstorms EV3 robotist, EV3 anduritest ja Vernier sensoritest koos vajalike ühendusliideste üleminekutega ning tarkvarast, mis loob leJOS EV3 platvormil interaktiivse veebileheküljega HTTP-serveri ja võimaldab nimetatud seadmeid Interneti vahendusel juhtida. Esmajärjekorras on komplekt ette nähtud töötama labori või praktikumiruumi kohalikus WiFi võrgus ning on ligipääsetav kohtvõrgu IP aadressi kaudu. IP aadress kuvatakse võrku ühendatud EV3 roboti LCD ekraanil. Seadmed on kompaktsed ja lihtsasti teisaldatavad, “labori” võib üles seada sobivas asupaigas. Vajadusel on kauglaborikomplekt rakendatav ka üle avaliku Interneti, selle eelduseks on kohaliku arvutivõrgu ruuterisse NAT-suunamisreegli lisamine, millega õppeasutuste kohalikud võrguadministraatorid peaksid toime tulema.

Lahendus on laiendatav ka muude andurite kasutamiseks. LEGO originaalandurite ja järelturul saadavate Dexter, HiTechnic ning Mindsensors andurite juhtimiseks on ev3classes.jar pakis leJOS tarkvara klassid olemas, Vernier sensorid on ühendatavad BTA-NXT ülemineku abil ning võimalik implementeerida analoog-, I2C või UART sensorina.

Süsteemi piirangutest tasub välja tuua, et probleemivabalt toimib WiFi ühendus vaid WPA2-PSK tasemel turvatud võrkudega. Avatud (nt ut-public) ning WPA2-Enterprise krüpteeringuga võrkudesse (nt eduroam) ühendumine nõuab rohkem seadistamist

ning ei pruugi iga kord õnnestuda, konkreetselt TÜ avaliku võrgu ut-public puhul tuleb rõhutada et selle võrgu klientseadmed ei oma ligipääsu teistele sama võrgu klientidele ehk TÜ seadistus ei luba kasutajat roboti IP-aadressil töötavale veebilehele. Lahendus ut-public takistusele on leJOS tarkvara WiFi skripti käsitsi konfigureerimine ja selles eduroam võrgu seadistamine.

Aeg-ajalt esineb leJOS tarkvara hangumist, eriti seadme käivitamise erinevates etappides. Reeglina toimub see juhul, kui toitepinge (aku laetus) langeb liiga madalale, alla 7.2 V. Seadme taaskäivitus võib probleemi edasi lükata, kuid lihtsam on võtta kasutusele uued patareid või ühendada akulaadija.

6.1 Ülesandeid laboris

Loodud kauglabor on universaalne pilootkomplekt, mis võimaldab mitmesuguste andurite kasutamist ja nende abil erinevate ülesannete sooritamist. Ülesande koostamisel on soovitatav järgida kolme kauglaboritele spetsiifilist põhimõtet:

- Sooritatav katse peab olema jälgitav video vahendusel. On hea, kui kogu katse mahub kaamera kaadrisse, liikuvad osad on arvutiekraanil nähtavad, piisavalt suured ja kontrastsed, ning liikumine toimub paraja kiirusega.
- Eksperimendi algseis peab olema automaatselt taastatav, et labor oleks järgmise kasutaja jaoks valmis ilma kõrvalise abita.
- Katse on põnev ja mitmepalgeline, kasutaja peaks saama sooritada mitu erinevat toimingut.

Ühe ülesandena võib välja pakkuda muudetava pikkusega pendli liikumise uurimise. Servomootoriga saab seada pendli pikkuse ja kui pendel varustada magnetiga, siis saab poolilt mõõta pendli liikumise perioodi.

Käesolevas töös kasutatud mootorite ja sensorite komplektiga on näiteks võimalik vedru laiali tõmmata või kokku suruda, vedru avaldatav jõud fikseerida Vernier jõu anduriga ning seejärel mõõta ka vedru venimise kaugus EV3 kaugussensori abil. Mõned neist väärtustest saab ülesande püstituses ette anda ja lasta mõõta ülejäänud suurused ning arvutada vedru omadusi.

6.2 Edasine töö

Kuivõrd tegemist on pilootprojektiga, on tarvis jälgida kauglaborikomplekti kasutuselevõttu ning koguda tagasisidet. Koostöös haridustöötajatega on võimalik välja selgitada tegelikkuses rakendatavad sensorid ning luua konkreetsetele laboriülesannetele vastavate parameetritega robotid. On oluline välja selgitada komplekti tegelik kasutus, et analüüsida vajadust rohkemate kauglaborite järele.

Loodud leJOS EV3 kauglaborikomplektide laialdasema rakenduse saavutamiseks üldhariduskoolides on oluline nende kasutuselevõtmise, sealhulgas olemasoleva Mindstorms EV3 robotikakomplekti kauglaboriks muutmise, veelgi suurem lihtsustamine. Juba seadistatud robotikakomplekti kauglaborina kasutamine on küll väga hõlpsalt kasutatav, kuid alternatiivtarkvara leJOS installeerimine ületab käesolevas faasis keskmiselt põhikooli loodusainete õpetajalt oodatavat taset, mis muudab kasutuselevõtu esimese astme plaanitust keerulisemaks. Lisatavasse funktsionaalsusesse peaks kuuluma automatiseeritud programm, mis konfigureerib roboti vastavalt uues WiFi võrgus saadud IP aadressile.

Kokkuvõte

KAUGLABOR LEGO MINDSTORMS EV3 BAASIL

Alex Nõomaa

Käesoleva töö eesmärk oli Interneti vahendusel laboriülesannete sooritamiseks sobiliku interaktiivse tark- ja riistvarakomplekti loomine, kasutada seda tehes võimalusel ära olemasolevat riistvara ning pakkuda erinevates õppeasutustes kergesti kohaldatavat kauglaborilahendust.

Seadmete platvormiks valiti robotikakomplektid LEGO Mindstorms, mis võimaldavad servomootorite juhtimist, mitmesuguste andurite kasutamist ning mis on ka Eesti koolides laialdaselt kasutuses. LEGO juhtmoodulite üle internetiühenduse juhtimise probleemi prooviti lahendada mitmel meetodil. Peatükis 4 on kirjeldatud videoülekandega serverilahenduse loomist vabavaralisel levinud tarkvaral Android, mis tunnistati ebaõnnestunuks.

Peatükk 5 käsitleb lahendust, kus serverirakendus töötab juhitava LEGO Mindstorms laborirobotiga samas seadmes. Mõõteseadmetena kasutatakse LEGO EV3 andureid ja Vernier andmekogujate sensoreid.

Töö eesmärgi täitmine osutus kavandatust oluliselt keerukamaks, siiski valmis soodne riist- ja tarkvarakomplekt Interneti vahendusel kauglabori lahenduse pakkumiseks ning anti soovitusel kauglaboriülesannete koostamiseks.

Summary

REMOTE LABORATORY BASED ON LEGO MINDSTORMS EV3 ROBOTICS SET

Alex Nõomaa

The main objective of this master's thesis was to create an interactive bundle of soft- and hardware with the purpose of executing lab tasks via the Internet.

LEGO Mindstorms robotics sets were chosen as platform for the remote lab, as they enable the user to control servos and make use of various sensors, while already being widespread in Estonian schools.

We tried to solve the task of controlling the LEGO modules over the Internet in several ways. Chapter 4 describes creating a video streaming server application on the popular open-source software package Android, which was admitted to be a failure. Chapter 5 covers the solution in which the server application runs on the same LEGO Mindstorms module that acts as the remote lab robot. LEGO EV3 and Vernier data logger sensors are used as measuring peripherals.

Achieving the objective of this thesis turned out quite a lot more challenging than initially planned, nevertheless a fair-priced set of hardware and software was created to offer remote lab services over the Internet, while some suggestions for lab tasks were also given.

Viited

- [1] J. G. Zubía and R. A. Gustavo, *Using Remote Labs in Education: Two Little Ducks in Remote Experimentation*, no. September. 2012.
- [2] H. Vallaste, "Vallaste IT ja sidetehnika seletav sõnaraamat," 2015. [Online]. <http://vallaste.ee>. [11-Apr-2016].
- [3] O. Dziabenko and J. Garcia Zubia, "Planning and designing remote experiment for school curriculum," *IEEE Glob. Eng. Educ. Conf. EDUCON*, vol. 2015-April, pp. 874–878, 2015.
- [4] Go-Lab consortium, "Online Labs," 2016. [Online]. [http://www.golabz.eu/labs?f\[0\]=field_lab_type%3A1275](http://www.golabz.eu/labs?f[0]=field_lab_type%3A1275). [20-May-2016].
- [5] MTÜ Robootika, "Koolirobootika programm," 2016. [Online]. https://www.robootika.ee/?page_id=38. [10-May-2016].
- [6] L. Tobarra, S. Ros, R. Hernandez, R. Pastor, A. Robles-Gomez, A. C. Cammero, and M. Castro, "Low-cost remote laboratories for renewable energy in distance education," *Proc. 2014 11th Int. Conf. Remote Eng. Virtual Instrumentation, REV 2014*, no. February, pp. 106–111, 2014.
- [7] K. Chiang-Yu Chin, S. M. Buhari, and W. H. Ong, "Impact of LEGO sensors in remote controlled robot," *2009 IEEE Int. Conf. Robot. Biomimetics*, pp. 1777–1782, 2009.
- [8] S. A. Filippov and A. L. Fradkov, "Cyber-physical laboratory based on LEGO Mindstorms NXT - First steps," in *Proceedings of the IEEE International Conference on Control Applications*, 2009, no. 0, pp. 1236–1241.
- [9] B. Carey and L. Hesselink, "'iLabs' offer a new way to add science experiments to online education," *Stanford News*, 2013. [Online]. <http://news.stanford.edu/news/2013/december/lab-ina-box-120613.html>. [12-May-2015].
- [10] "Estrotech OÜ e-pood." [Online]. Available: www.robomiku.ee. [Accessed: 13-Jan -2016].

- [11] "E-Riigihangete keskkond - Riigihangete register." [Online].
<http://riigihanked.riik.ee/register/RegisterHanked.html>. [13-May-2016].
- [12] "History of LEGO Robotics." [Online]. <http://www.lego.com/en-us/mindstorms/history>. [10-May-2016].
- [13] X. Soldaat, "Comparing the NXT and EV3 bricks," 2013. [Online].
<http://botbench.com/blog/2013/01/08/comparing-the-nxt-and-ev3-bricks/>.
 [09-Sep-2014].
- [14] Vernier, "NXT Sensor Adapter." [Online].
<http://www.vernier.com/products/interfaces/bta-nxt/>. [26-Apr-2016].
- [15] "LEGO® 9V Technic Motors compared characteristics," 2014. [Online].
 Available: <http://philohome.com/motors/motorcomp.htm>. [Accessed: 03-Apr-2016].
- [16] "NanoHttpd." [Online]. Available: <http://nanohttpd.org/>. [Accessed: 26-Feb-2014].
- [17] "WiFi for Lego Mindstorms NXT - ESP8266 RS485 interface using NodeMCU Lua Firmware." [Online]. stefanshacks.blogspot.com/2015/04/wi-fi-for-lego-mindstorms-nxt-esp8266.html. [15-May-2016].
- [18] L. Jeppesen and A. Sjøborg, "MonoBrick Communication Library," 2015.
 [Online]. <http://www.monobrick.dk/software/monobrick/>. [10-May-2016].
- [19] "MonoBrick EV3 Firmware," 2015. [Online].
<http://www.monobrick.dk/software/ev3firmware/>. [10-May-2016].
- [20] M. W. Lew, T. B. Horton, and M. S. Sherriff, "Using Lego Mindstorms NXT and LeJOS in an advanced software engineering course," *Softw. Eng. Educ. Conf. Proc.*, pp. 121–128, 2010.
- [21] "leJOS EV3." [Online]. <http://www.lejos.org/ev3.php>. [19-May-2016].
- [22] Ev3dev.org, "EV3 Processor." [Online]. <http://www.ev3dev.org/docs/kernel-hackers-notebook/ev3-processor/>. [16-May-2016].
- [23] A. Shaw, "Comparing WiFi Adapters," 2015. [Online].
<https://lejosnews.wordpress.com/2015/02/03/comparing-wifi-adapters/>.

[12-May-2016].

- [24] Ev3dev.org, "Sensors and everything else that plugs into input ports on the EV3." [Online]. <http://www.ev3dev.org/docs/sensors/>. [18-May-2016].
- [25] Vernier, "Sensor pinouts." [Online]. <http://www.vernier.com/support/sensor-pinouts/>. [10-May-2016].
- [26] Vernier Software & Technology, "Dual-Range Force Sensor andmeleht, rev 2.16.10." p. 8, 2010.
- [27] Vernier Software & Technology, "Differential Voltage Probe - andmeleht, rev 2/16/10." p. 4, 2010.

Lisad

Lisa 1 API päringute nimistu

Programm ootab URI indikaatoreid aadressil http://EV3_IP/api. Lisatud tabelis on toodud lubatud päringud, mõõdetavad ja tagastatavad arvulised väärtused on tabelis tähistatud tähtedega x, y, z.

Näide korrektsest päringust, millega nullitakse jõu anduri lugem, kui mõõtepiirkonnana kasutatakse 10 N: <http://192.168.1.139/api/calibrate/force10>

URI 1	URI 2	Tegevus	Tagastus
/forward	/	robot hakkab edaspidi sõitma	{action: "forward"}
/back	/	robot hakkab tagurpidi sõitma	{action: "back"}
/left	/	robot hakkab kohapeal vastupäeva pöörama	{action: "left"}
/right	/	robot hakkab kohapeal päripäeva pöörama	{action: "right"}
/thirdUp	/	kolmas mootor liigub ülessuunas	{action: "thirdUp"}
/thirdDown	/	kolmas mootor liigub allasuunas	{action: "thirdDown"}
/stop	/	kõik mootorid lukustatakse paigale	{action: "stop"}
/float	/	kõik mootorid peatatakse ja jäetakse vabakäigule	{action: "float"}
/plus	/	kiiruse tõstmine 60 kraadi võrra sekundis	{action: "plus"} Speed is now x

/minus	/	kiiruse langetamine 60 kraadi võrra sekundis	{action: "minus"} Speed is now x
/speed	/123	Kiiruse seadmine väärtusele 123 - lubatud väärtused 60-900, muud ignoreeritakse	{action: "speed"} Speed set to 123
/calibrate	/force10	nullitakse jõusensori näit 10 N mõõtepiirkonnas	{action: "calibrate"} Calibrated to x
/calibrate	/force50	nullitakse jõusensori näit 50 N mõõtepiirkonnas	{action: "calibrate"} Calibrated to x
/sensor	/touch	Küsitakse puuteanduri olek	{touch: x }
/sensor	/distance	Küsitakse kaugusanduri lugem	{distance: x m}
/sensor	/light	Küsitakse värvianduri lugem	{color: {RED: x, GREEN: y, BLUE: z}}
/sensor	/voltage	Küsitakse pingeaduri lugem	{voltage: "x V"}
/sensor	/force10	Küsitakse jõu anduri lugem, mõõtepiirkond 10 N	{force: "x Newtons."}
/sensor	/force50	Küsitakse jõu anduri lugem, mõõtepiirkond 50 N	{force: "x Newtons."}
/x	/		{action: "Bad action"}

Tabel 7. Kauglabori API juhtimine

Lisa 2 Kasutajaliidese veebileht, index.html

```
<html>
<head><title>LEGO kauglabori veebiliides</title>
<meta charset="UTF-8">
<script>
function action(type, sensor) {
var xmlhttp;
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera, Safari
xmlhttp=new XMLHttpRequest();
} else {
// code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange = function() {
if (xmlhttp.readyState == XMLHttpRequest.DONE) {
document.getElementById("result").innerHTML+xmlhttp.responseText;
}
}
}

//API control
if (sensor) {
xmlhttp.open("GET","api/"+type+"/"+sensor,true);
} else {
xmlhttp.open("GET","api/"+type,true);
}
xmlhttp.send();
}

//extra button for Vforce sensor
function showTare(selection) {
if(selection.value == 'force10' || selection.value == 'force50')
{
document.getElementById('calibrate').style.display = "inline";
}
else
{
document.getElementById('calibrate').style.display = "none";
}
}
</script>
<noscript>Sorry, your browser does not support JavaScript!</noscript>
</head>

<body>
<div style="text-align:center">
```

```

<h2>LEGOKaugLabor EV3</h2>
<div id="result"></div>
<button type="button" onclick="action('forward')"></button>
<br>
<button type="button" onclick="action('left')"></button>
<button type="button" onclick="action('float')"></button>
<button type="button" onclick="action('right')"></button>
<br>
<button type="button" onclick="action('back')"></button>
<br>
<button type="button" onclick="action('thirdDown')"></button>
<button type="button" onclick="action('stop')"></button>
<button type="button" onclick="action('thirdUp')"></button>
<br>
<button type="button" onclick="action('plus')"></button>
<button type="button" onclick="action('minus')"></button>
<br>
<select name="sensor" id="sensor" onchange="showTare(this)">
  <option value="touch">S1 puuteandur</option>
  <option value="distance">S2 kaugusandur</option>
  <!-- hidden in favor of voltage <option value="color">S3 värviandur</option> -->
  <option value="voltage">S3 pinges</option>
  <option value="force10">S4 jõud 10N</option>
  <option value="force50">S4 jõud 50N</option>
</select>

<button type="button"
onclick="action('sensor',document.getElementById('sensor').value)">Mõõda!</butt
on>

<button type="button" id="calibrate" style="display: none;"
onclick="action('calibrate',document.getElementById('sensor').value)">nulli
lugem</button>

</div>
<p>Vaata laborit videost: <a href="http://192.168.1.88" target="_blank">Link IP
kaamerale</a></p>
</body>
</html>

```

Lisa 3 SimpleWebServer.java muutujad ja meetod String apiEndpoint

```
public class SimpleWebServer extends NanoHTTPD {

    static RegulatedMotor leftMotor = Motor.C;
    static RegulatedMotor rightMotor = Motor.B;
    static RegulatedMotor thirdMotor =
MirrorMotor.invertMotor(Motor.A);

    // Get an instance of the Touch EV3 sensor
    static EV3TouchSensor sensorTouch = new
EV3TouchSensor(SensorPort.S1);
    static SensorMode touch = sensorTouch.getTouchMode();

    // Get an instance of the Ultrasonic EV3 sensor
    static SensorModes sensorDistance = new
EV3UltrasonicSensor(SensorPort.S2);
    // get an instance of this sensor in measurement mode
    static SampleProvider distance =
sensorDistance.getMode(sensorDistance.getCurrentMode());

    // Disabling colorSensor for successful booting while
disconnected
    // static EV3ColorSensor sensorColor = new
EV3ColorSensor(SensorPort.S3);
    // static SensorMode color = sensorColor.getRGBMode();

    // Get an instance of the Vernier Dual-Range Force
Sensor
    static VernierAnalogSensor sensorForce = new
VernierAnalogSensor(SensorPort.S4);
    static SensorMode force = sensorForce.getForceMode();
    static float zero10 = 2.915751f; // measured default
value with no
    // load, 10N mode
    static float zero50 = 2.7838829f; // measured default
value with no
    // load, 50N mode
    static final float calibrationData10 = 0.1888482f; //
calculated 1 N value
        // from calibrating
        // with 300g
    static final float calibrationData50 = 0.0376068f; //
calculated 1 N value
        // from calibrating
        // with 1000g
```

```

    // Get an instance of the Vernier Differential Voltage
    Sensor
    static VernierAnalogSensor sensorVoltage = new
    VernierAnalogSensor(SensorPort.S3);
    static SensorMode voltage =
sensorVoltage.getForceMode();
    // Re-using helper class, nothing to do with "force"
    here
    static float voltageOffset = -0.025946498f; // Manual
    correction
    static float voltageSlope = 2.5f; // Datasheet info
    static float voltageMultiplier = -0.4f; // Datasheet
    info

    int speed = 120; // Initial rpm of motor

    >>> // nanoHTTPD osa siin <<<

```

```

String apiEndpoint(String uri) {
    // Extract actions from web api
    String[] params = uri.replace("/api/",
    "").split("/");

    String action = params[0];
    // Show on the LCD what we are doing
    System.out.println("Action = " + action);

    leftMotor.setSpeed(speed);
    rightMotor.setSpeed(speed);
    thirdMotor.setSpeed(speed);

    switch (action) {
    case "forward":
        leftMotor.forward();
        rightMotor.forward();
        break;

    case "back":
        leftMotor.backward();
        rightMotor.backward();
        break;

    case "left":
        leftMotor.forward();

```

```

        rightMotor.backward();
        break;

    case "right":
        leftMotor.backward();
        rightMotor.forward();
        break;

    case "thirdUp":
        thirdMotor.backward();
        break;

    case "thirdDown":
        thirdMotor.forward();
        break;

    case "stop":
        leftMotor.stop(true);
        rightMotor.stop(true);
        thirdMotor.stop();
        break;

    case "float":
        // Release the brakes
        leftMotor.flt(true);
        rightMotor.flt(true);
        thirdMotor.flt();
        break;

    case "plus":
        speed += 60;
        if (speed > 900)
            speed = 900;
        leftMotor.setSpeed(speed);
        rightMotor.setSpeed(speed);
        thirdMotor.setSpeed(speed);
        System.out.println("Speed is now " + speed);
        return "{Speed is now " + speed + " }";

    case "minus":
        speed -= 60;
        if (speed <= 60)
            speed = 60;

        leftMotor.setSpeed(speed);
        rightMotor.setSpeed(speed);
        thirdMotor.setSpeed(speed);

```

```

        System.out.println("Speed is now " + speed);
        return "{Speed is now " + speed + " }";

    case "speed":
        speed = Integer.parseInt(params[1]);
        System.out.println("Speed set to " + speed);
        break;

    case "calibrate":
        int sampleSizeForceCalibrating =
force.sampleSize();
        float[] forceSampleCalibrating =
new float[sampleSizeForceCalibrating];
        force.fetchSample(forceSampleCalibrating, 0);
        String forceSwitch = params[1];
        switch (forceSwitch) {
            case "force10":
                zero10 = forceSampleCalibrating[0];
                System.out.println("Calibrated to " +
zero10);

                break;
            case "force50":
                zero50 = forceSampleCalibrating[0];
                System.out.println("Calibrated to " +
zero50);

                break;
        }
        break;

    case "sensor":
        String sensor = params[1];
        switch (sensor) {

            case "touch":
                // fetch a sample
                int sampleSize = touch.sampleSize();
                float[] touchSample = new
float[sampleSize];

                touch.fetchSample(touchSample, 0);
                return "{touch: " + touchSample[0] + "
}";

            case "distance":
                // stack a filter on the sensor that
gives the running average
                // of the last 5 samples
                SampleProvider average = new

```

```

MeanFilter(distance, 5);
    // initialize an array of floats for
fetching samples
    float[] sample = new
float[average.sampleSize()];
    // fetch a sample
average.fetchSample(sample, 0);
    return "{distance: " + sample[0] + "
m}";

    /*
    * disable light in favour of voltage
    *
    * case "light":
    * // fetch a sample
    * int sampleSizeColor = color.sampleSize();
    * float[] colorSample = new
float[sampleSizeColor];
    * color.fetchSample(colorSample, 0);
    * return "{color: {RED: " + colorSample[0] +
", GREEN: " + colorSample[1] + ", BLUE: " + colorSample[2] +
"} }";
    */

    case "voltage":
    // fetch a sample
    int sampleSizeVoltage =
force.sampleSize();
    float[] voltageSample = new
float[sampleSizeVoltage];
    voltage.fetchSample(voltageSample, 0);
    float voltageDiff = (voltageSample[0] -
voltageSlope) / voltageMultiplier - voltageOffset;
    return "{voltage: \" + voltageDiff + "
V\"}";

    case "force10":
    // fetch a sample
    int sampleSizeForce =
force.sampleSize();
    float[] forceSample =
new float[sampleSizeForce];
    force.fetchSample(forceSample, 0);
    float voltageDiff10 = zero10 -
forceSample[0];
    float forceNewt =
Math.round(voltageDiff10 / calibrationData10);

```

```

        return "{force: \"" + forceNewt + "
Newtons.\"}";

        case "force50":
            // fetch a sample
            int sampleSizeForce50 =
force.sampleSize();
            float[] forceSample50 =
new float[sampleSizeForce50];
            force.fetchSample(forceSample50, 0);
            float voltageDiff50 = zero50 -
forceSample50[0];
            float forceNewt50 =
Math.round(voltageDiff50 / calibrationData50);
        return "{force: \"" + forceNewt50 + "
Newtons.\"}";
    }

    default:
        System.out.println("Unknown");
        return "{action: \"Bad action\"}";
    }

    return "{action: \"" + action + "\"}";
}
}
}

```

Lisa 3 VernierAnalogSensor.java

```
package fi.iki.elonen;

import lejos.hardware.port.AnalogPort;
import lejos.hardware.port.Port;
import lejos.hardware.sensor.AnalogSensor;
import lejos.hardware.sensor.SensorMode;

public class VernierAnalogSensor extends AnalogSensor {

    public VernierAnalogSensor(AnalogPort p) {
        super(p);
        init();
    }

    public VernierAnalogSensor(Port p) {
        super(p);
        init();
    }

    public SensorMode getForceMode() {
        return this;
    }

    @Override
    public int sampleSize() {
        return 2; //for pin1+6
    }

    protected void init() {
        setModes(new SensorMode[] { this });
    }

    @Override
    public String getName() {
        return "Force";
    }

    @Override
    public void fetchSample(float[] sample, int offset) {
        // get data from analog signal pin (+/- 10V)
        sample[offset] = (port.getPin1());
        // get data from digital signal pin (0-5V)
        sample[offset+1] = (port.getPin6());
    }

}
```

Lisa 4 LEGO-kauglabori programmikood

Lisatud programmikood failis Legokauglabor-nanowebserver.zip

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Alex Nõomaa

1. Annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

“Kauglabor LEGO Mindstorms EV3 baasil”

mille juhendaja on Kaido Reivelt

- (a) Reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - (b) Üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile;
 3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seaduses tulenevaid õigusi.

Tartus, 20.05.2016