

TALLINNA ÜLIKOOL

Digitehnoloogiaste instituut

# Vähetaunatud tootjate tahvelarvutid ja nende täiustamine

Seminaritöö

Autor: Janek Kossinski

Juhendaja: Jaagup Kippar

Autor: ..... ” ..... ” ..... 2017

Juhendaja: ..... ” ..... ” ..... 2017

Instituudi direktor: ..... ” ..... ” ..... 2017

Tallinn 2017

## Autorideklaratsioon

Deklareerin, et käesolev seminaritöö on minu töö tulemus ja seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

.....

(kuupäev)

.....

(autor)

## Sisukord

Sissejuhatus .....	4
1. Sihtgrupp .....	5
1.1 Mis hetkest võib tahvelarvutit liigitada VTTT-de hulka? .....	5
1.2 Järeldused VTTT-de olemuse kohta.....	8
1.2.1 Hind .....	8
1.2.2 Tarkvara .....	9
1.3 Konkreetne järeldus.....	9
2. Täiustamise teooria ja võimalused .....	10
2.1 Millest sõltub kiirus? .....	10
2.1.1 Riistvara .....	10
2.1.2 Tarkvara .....	11
2.1.3 Muu .....	12
2.2 Täiustamise võimalused .....	13
2.2.1 Kasutaja tase.....	13
2.2.2 Juurõigustega kasutaja tase .....	14
2.2.3 Süsteemi tase .....	14
3. Testimine ja tulemused.....	16
3.1 Testimismeetodid .....	16
3.2 Testid.....	16
3.3 Järeldused .....	20
Kokkuvõte .....	22
Kasutatud kirjandus.....	23
LISAD .....	24
LISA 1. build.prop faili muudatused.....	25

## Sissejuhatus

Siinse seminaritöö teemaks on vähetuntud tootjate tahvelarvutid ja nende täiustamine. Autor on motiveeritud kirjutama siinsel teemal, sest teda ennast huvitab Android operatsioonisüsteem, nutiseadmed üldiselt ning nende tarkvara täiustamine, muutmine või vahetamine kasutaja enda soovide järgi.

Autori arvates on siinne teema aktuaalne, sest vähetuntud tootjate tahvelarvutid on sisuliselt dokumenteerimata grupp nutiseadmeid, millel pole head kasutusala, kui võtta arvesse kui halvad nad tavaliselt on.

Siinse töö eesmärgiks on anda lugejale ülevaade vähetuntud tootjate tahvelarvutitest, selgitada kuidas neid täiustada saab ning testida mõned välja toodud täiustused.

Välja toodud täiustuste testimine oleks läbi viidud autori enda vähetuntud tootja tahvelarvuti peal. Igat täiustust testitaks kolm korda ning iseseisvalt (ainult üks täiustus rakendatud seadmele korraga) selleks, et täiustuste kasu saaks võrrelda.

## 1. Sihtgrupp

Tänapäeval tahvelarvutite soetamine tavakasutajate jaoks tehtud tootjate poolt küllaltki raskeks, kui võtta arvesse, et sisuliselt iga võimalikku tahvelarvuti riistvara kombinatsiooni on kasutatud nende loomiseks. On tahvelarvuteid: väikeste ja suurte ekraanidega, kiirete 4-tuumaliste ja aeglaste 10-tuumaliste protsessoritega, graafikakaartidega, mälukaartide lugejatega, SIM-kaartide pesadega, erinevate suuremate arvutite sisendpesadega, klaviatuuridega ja paljude teiste omadustega. Tavakasutajale võib nii suur valik erinevaid tahvelarvuteid jätta arusaama, et kõik tahvelarvutid on sisuliselt samad ning erinevus on ainult hinnas ja ekraanis või mõnes muus ebatähtsas aspektis.

Kui sellise arusaamaga kasutaja mõne suurema toidupoe elektroonikaosakonnas märkab tahvelarvutit, mille hind on üks kümnendik mõnest kuulsama ettevõtte tahvelarvutist ja sellel kasutajal on parasjagu tahvelarvuti soetamishuvi, siis tõenäoliselt ostetakse just see toidupoes müügil olev tahvelarvuti. Kasutaja kahjuks see ost tõenäoliselt ei olnud väga hea, sest mida ta varsti avastab, on see, et selline tahvelarvuti on väga heal juhul kasutuskõlblik ainult sisseehitatud veebilehitsejas lihtsamate lehtede sirvimiseks. Seda tüüpi tahvelarvutid (edaspidi VTTT – „vähem tuntud tootjate tahvelarvutid“) ongi siinse töö teema.

Eelnevalt sõnastatud ostustsenaariumis võib VTTT definitsioon jääda natuke puudulikuks, seega järgnev peatükk avab laiemalt, mida mõtleb autor VTTT all.

### 1.1 Mis hetkest võib tahvelarvutit liigitada VTTT-de hulka?

Selleks, et teha üldist liigitust tahvelarvutite seas, oleks eelnevalt tarvis paika panna teatud riistvaralised kriteeriumid, millele liigitatav seade vastama peab, et teda saaks nimetada VTTT-ks. Autori arvates sobiksid nendeks kriteeriumiteks järgmised:

- **Protsessor** – esimese asjana võiks pöörata tähelepanu selle taktsagedusele. Tavaliselt taktsagedus ei määra protsessori jõudlust („Protsessor“, kuupäev puudub), kuid kui tahvelarvuti protsessoril on madal taktsagedus (500 MHz – 1GHz), siis juba see võib tähendada, et seade ei ole kiire. Samas kõikidel juhtudel ei tähenda madal taktsagedus aeglast seadet, sest protsessor võib olla mitmetuumaline või protsessoreid on mitu, kuid see omakorda tähendab, et protsessor nõuab rohkem aku ressursi. Ehk kui protsessor on näiteks 10-tuumaline siis seadmel peab olema ka vastava mahuga aku, et töötada tavakasutaja seadme tüüpkasutusaja vältel.

- **Mälu** – selle all on mõeldud nii muut- kui ka püsिमälu. Mõlema puhul on nende kogus tähtis, kuid muutmälu puhul on tähtsus natuke suurem. Seadme normaalseks tööks on tarvis piisavalt suur muutmälu, et lisaks operatsioonisüsteemile saaksid toimida ka kasutaja rakendused. VTTT-del tüüpiliselt on seda vähe (500MB – 1GB), mille tõttu nad kohati lihtsalt seiskuvad, mis omakorda madaldab kasutuskõlblikkust. Vajaminev püsिमälu hulk sõltub pigem kasutajast endast, kuid üldiselt alla 8Gb püsिमälu on tänapäeva tahvelarvutites vähevõitu ning võib viidata VTTT-le. Lisaks muut- ja püsिमälule võib heita pilgu ka mälukaardi toetusele. Kui mälukaardi tugi on olemas, siis tähelepanu on tarvis suunata suurimale toetatud mahule. VTTT-del on see tüüpiliselt kuni 64Gb, mis tänapäeva nutiseadmete puhul on keskmisest kõvasti madalam.

- **Ekraan** – selle puhul on tarvis pöörata tähelepanu neljale aspektile:

- 1) puutetundlikkus - kui kiiresti seade reageerib ja kui täpselt ekraan seadmele vajutusest infot edasi annab;
- 2) resolutsioon - kui kvaliteetne pilt ekraanil on;
- 3) piksli tihedus - kui terav pilt ekraanil on;
- 4) vaatenurgad - kas ekraani saab ainult täpse või ükskõik mis nurga alt vaadata.

VTTT-dele on omane halb puutetundlikkus. Näiteks peab mitu korda vajutama ekraanile, et õige asi reageeriks, lihtsalt peab kõvasti vajutama või mõnes kohas ekraan üldse ei reageeri vajutustele. Lisaks madal resolutsioon - piksli tihedust tavaliselt ei mainitagi, kuid palja silmaga juba näeb, et see pole eriti kõrge. Vaatenurk on tüüpiliselt ainult üks, sest teistel juhtudel hakkab ekraan peegeldama.

- **Kaamera** – VTTT-l on tihti olemas kaamera ainuüksi sellepärast, et oleks võimalik karbil või reklaamis kirjutada, et tal on olemas kaamera, sest selle kaamera resolutsioon üle 1Mpix tavaliselt ei ole.

Võib tekkida selline olukord, kus liigitatav seade vastab kõikidele VTTT-de riistvaralistele kriteeriumitele, kuid on siiski kasutuskõlblik. Sellisel juhul oleks tarvis lisaks riistvaralistele kriteeriumitele vaadelda ka tarkvaralisi ning üldiseid kriteeriume:

- **Operatsioonisüsteem** – tüüpiliselt mõni vanem Androidi versioon, kas 4 või 5, millele kunagi uuendusi ei tehta. Autori arvates saab VTTT-de operatsioonisüsteemide kohta teha üldistatud järelduse ja see seletatakse lahti hiljem selles peatükis.
- **Hind** – selle kriteeriumi abil saab kõige kergemini ära määrata, kas tegu on VTTT-ga. Nende hind on tüüpiliselt alla 60€ Autori arvates on see üks võtmekriteeriumitest, sellest täpsemalt kirjutab autor hiljem käesolevas peatükis.
- **Välimus** – selle all on mõeldud, kuidas seade välja näeb ja mis tunne on seda käes hoida. Alloleval pildil (vt Foto 1) on autori enda VTTT eestvaates, kuid VTTT-sid on küllaltki raske eristada eestvaate järgi, sest paljud tahvelarvutid näevad eest sarnased välja. Võib-olla võib ette heita, et ekraani ja äärte vahed on liiga suured, kuid see pole omane ainult VTTT-dele. Teisel alloleval pildil (vt Foto 2) on sama VTTT tagantvaates. Siin võib näha halvasti paigutatud teksti pildi alumises vasakus nurgas ning odava välimusega tagakaant. Selle VTTT hoidmine jätab odavuse mulje.



Foto 1. Autori VTTT eestvaates



Foto 2. Autori VTTT tagantvaates

- **Tootja** – VTTT tootja on tüüpiliselt täiesti tundmatu ning kohati seadme ja selle karbi peal olev ettevõtte nimi ei ole sama, mis on seadme enda build.prop<sup>1</sup> failis kirjeldatud. Näiteks autori enda VTTT peal on suurelt kirjas TUNEX (vt Pilt 2) ja mudeli nimeks on TAB760, kuid selle VTTT build.prop failis on kirjas „model“ parameetri all „Tablet“ ja „brand“ parameetri all „iNet“. Lisaks, kui teha otsingupäring Google's märksõnaga „Tunex“, siis päringu tulemustes ei leidu seost selle märksõna ja tahvelarvutite vahel. Küll on aga paar müügikuulutust, kus seda märksõna kasutatakse, kuid nendel kuulutustel pole seost ettevõtte endaga.
- **Pakend** – selle järgi ei saa alati kindel olla, kuid tavalisest tihedamini pakend on kas tavaline valge karp või on esile toodud teatud komponentide logod, nagu näiteks:

<sup>1</sup> build.prop – see on fail, mis on omane Android'i operatsioonisüsteemi kasutatavatele seadmetele, mis paikneb seadme juurkaustas ning kus on tootja poolt kirja pandud parameetrid, mida seade, käivitudes, arvesse võtab. Näiteks saab muuta mis sagedustel WIFI töötab või mis formaadis aega kuvatakse süsteemis ja palju muud.

Android, WIFI, Bluetooth ja muude enamlevinud komponentide logod (vt Foto 3) ja selline kujundus kordub neljal teisel küljel (vt Foto 4).



Foto 3. Autori VTTT karp pealtvaates



Foto 4. Autori VTTT küljekaardid

- **Muu** – tähelepanu võib ka pöörata sellele, kus seade parasjagu müügis on. Kasutuskõlblikumad tahvelarvutid on tavaliselt müügis suuremates elektroonika-poodides, samas VTTT-d on enamasti müügil ainult toidupoodide elektroonikaosakondades ja väikestes elektroonika-poodides.

Autor soovib rõhutada, et seade peab vastama valdavale enamusele neist kriteeriumitest, et teda saaks liigitada VTTT-de hulka. On ka tuntud tootjate tahvelarvuteid, mis vastavad teatud siin paika pandud kriteeriumitele, kuid tavaliselt tuntud tootjaga kaasnevad teatud garantiid nagu tarkvara uuendused või head riistvaralised komponendid, mille tõttu ei käsitleta neid siinses töös.

## 1.2 Järeldused VTTT-de olemuse kohta

Eelnevas alapeatükis selgitas autor, kuidas hinna ning tarkvara põhjal saab teha üldistatud järeldusi ning sooviks oma eelnevate kogemuste põhjal teha kummagi kohta järelduse.

### 1.2.1 Hind

Kui rääkida hinnast, siis VTTT-de hind on tavaliselt hästi madal võrreldes teiste tahvelarvutitega. See on ilmselt tingitud sellest, et VTTT-de osad on madala kvaliteediga. VTTT-d ei ole just kõige töökindlamad seadmed, neil lakkab tihti miski töötamast. Nad on väga odavad, sest neis kasutatakse vanemaid ja massiliselt toodetud komponente. Vanemate komponentide kasutuse näitena võib tuua autori enda VTTT, mis oli ostetud toidupoest Maxima aastal 2013, kuid selles on kasutusel 2011. aastal toodangusse tulnud süsteemikiip<sup>2</sup> Allwinner A13 („A13 Datasheet“, 2012). Massiliselt toodetud komponentide näitena võib

<sup>2</sup> süsteemikiip (ingl. *System on a chip, SoC*) - on kiip, mille sisse on ehitatud seadme suurimad komponendid, nagu: CPU, GPU, muutmälu, I/O loogika kontrollid ja palju muud (SoC, kuupäev puudub).



tuua loendamatu hulka teisi VTTT-si, kus kasutatakse samasugust süsteemikiipi. Mõned huvilised on teinud väiksemaid nimistuid tahvelarvutitest, mis kasutavad samasugust süsteemikiipi, nagu näiteks selles foorumi teemas<sup>3</sup> on näha. Kui vaadata, millised spetsifikatsioonid sealsetel tahvelarvutitel on, siis on näha, et need langevad enamjaolt kokku eelnevalt nimetatud VTTT-de kriteeriumitega.

### 1.2.2 Tarkvara

Nagu eelnevas alapeatükis kirjas, siis VTTT-d on väga sarnased üksteisega komponentide poolest ja sama kehtib ka tarkvara puhul. Kui otsida näiteks Google's võimalusi, kuidas kasutaja saaks enda VTTT-d täiustada, siis selle kohta häid nõuandeid ei leidu. Kui kasutaja on natuke pädevam ja on näiteks oma VTTT-s juure<sup>4</sup> privileegidega ringi uurinud, siis väga tõenäoliselt on ta oma järgmises lahenduse otsingus võtnud kasutusele enda VTTT build.prop failis kirjeldatud „brand-i“, „model-it“ ja „name-t“. Seda tehes kõik eelnevad asjatud otsingutulemused kaovad ja tekkivad kirjed, mis enamjaolt räägivad ühest ja samast lahendusest. Selleks on ühtne, kõikidel VTTT-del toimiv tarkvaraline lahendus, mille variatsioonid erinevad ainult protsessori mudeli osas ehk siis paljudele VTTT-dele sobivad ühed ja samad lahendused.

### 1.3 Konkreetne järeldus

Eelneva põhjal võib võita, et valdav enamus VTTT-si on sarnased komponentide ja tarkvara poolest. Neil toimivad samad lahendused, mille tõttu selle töö raames välja pakutud lahendused kehtiksid ka paljudele teistele VTTT-dele. Eelnevad järeldused on tehtud kolme kuni nelja aasta vanuste (selle töö kirjutamise ajast) VTTT-de kohta, kuid tänapäeval (selle töö kirjutamise ajal, aastal 2017) olukord pole muutunud. Need kolme kuni nelja aasta vanused tahvelarvutid on paljudes kohtades tänaseni müügil. Lisaks sellele, kolm kuni neli aastat tagasi ostetud VTTT-d ei ole kuskile kadunud, vaid koguvad siiaamaani tolmu kasutajate riulitel. Seda ütlust toetab see fakt, et tänapäeval on kommuunid, mis tegelevad nende VTTT-de arendamise ja täiustamisega.

---

<sup>3</sup> Link foorumi lehele: <http://www.slatedroid.com/topic/36752-list-of-allwinner-a13-cpu-devices/>.

<sup>4</sup> juurkasutaja (ingl superuser, su) – see on selline kasutaja Linux'i süsteemi keskkonnas, kellel on kõik administratiivsed õigused.

## 2. Täiustamise teooria ja võimalused

Eelnevas peatükis määratles autor täpse sihtgrupi (VTTT-d), mille kohta siinses töös edaspidi ka kirjutatakse. Sellest hoolimata on VTTT-d iseloomult ikkagi Android'i tahvelarvutid, mis tähendab, et VTTT-de ja tuntud tootjate tahvelarvutite kiirused sõltuvad ühtedest ja samadest aspektidest. Selles peatükis kirjeldab autor, millest sõltub tüüpilise Android'i tahvelarvuti kiirus (riistvara, tarkvara ja muud aspektid), ning mis võimalused on selle täiustamiseks.

### 2.1 Millest sõltub kiirus?

Tahvelarvutite ja muude taoliste seadmete kiirus võib sõltuda päris paljudest aspektidest. Kui üldisemalt neid kokku võtta, siis nendeks oleksid riistvara ja tarkvara ja just nende kahe kokkusobitamine. Tahvelarvuti sujuvaks tööks on tähtis, et mõlemad oleksid kooskõlas teineteisega, mida nad tavaliselt, eriti VTTT-de puhul, ei ole. Võib olla olukordi, kus seadmel on kõige uuemad ja paremad riistvaralised komponendid, kuid väga halvasti tehtud tarkvara osa, mis ei kasuta riistvara selle täies ulatuses, ja vastupidi, kus seadmel on mõne eelmise iteratsiooni riistvaralised komponendid, mille tõttu peab tarkvara palju tööd tegema.

#### 2.1.1 Riistvara

Tahvelarvutid (ja paljud teised nutiseadmed) on riistvara komponentide poolest väga sarnased arvutitega (näiteks mõlemal on protsessor, graafikakaart, muut- ja püsimälu ja muud taolised komponendid). Seega need riistvaralised komponendid, millest sõltub arvuti kiirus, mõjutavad ka tahvelarvutite kiirust. Tahvelarvutite komponendid on reeglina madalama jõudlusega ning võivad esineda arvuti omadest erineval kujul, kuid nende ülesanne on üldjoontes sama.

Nagu eelmises peatükis kirjutas, kasutab autori VTTT süsteemikiipi, kuhu on sisse ehitatud selle peamised riistvara komponendid. Seda kasutatakse füüsilise ruumi kokkuhoiu eesmärgil, kuid sealsed komponendid on oma ülesannete poolest ikkagi võrreldavad arvuti komponentidega ning nende seos tahvelarvuti kiirusega on järgmine:

- Protsessor – peamiselt vastutab käskude täitmise ja andmetega manipuleerimise eest („Protsessor“, kuupäev puudub). Ehk, kui palju käske korraga ning kui kiiresti suudab protsessor täita, mõjutab seadme üldist kiirust.
- Graafikakaart – vastutab ekraanile kuvatava sisu eest. Nagu protsessorilgi, mõjutab seadme kiirust see, kui kiiresti ta sisu suudab kuvada (töödelda talle antud infot, mida tarvis kuvada). Võib olla olukordi, kus graafikakaart on kiire, kuid protsessor ei edasta

andmeid piisavalt kiiresti. Näiteks mõne programmi puhul kuvatakse ainult selle välimus, kuid seade ei reageeri teiste sisendite (näiteks näpuvajutus) peale. Võib juhtuda ka täpselt vastupidine olukord, kus protsessoril on andmed edastatud, kuid seade ei kuva, et programm töötab

- Ekraan – see on siia nimistusse toodud ainuüksi sellepärast, et ekraan ning graafikakaart peavad sobima üksteisega. Näiteks võib ekraan olla liiga suure resolutsiooniga ning graafikakaart ei jõua lihtsalt kogu aeg uuendada seda kõrgetel sagedustel, mille tõttu võivad tekkida viivitused reageerimises kasutaja sisendile.
- Muutmälu – see on mälu, kuhu seade töö käigus salvestab erinevaid väärtusi, et neid kasutada mõne ülesande täitmiseks. Mida kiirem ja mahukam see mälu on, seda kiiremini suudab seade ülesandeid täita. Näiteks kasutaja täidab midagi veebilehitsejas, kuid on tarvis samaaegselt teisest programmist midagi vaadata. Piisava muutmälu puhul saab kasutaja teha seda mõlemat programmi paralleelselt jooksumises, sest mälu on ta hetke veebilehitseja sessioon olemas. Vastasel juhul peaks seade algusest peale programmi jooksumise, laadimise (aeglasest) püsivõime viimase sessiooni väärtused ja alles siis laskma kasutajal jätkata ta tegevusi.
- Püsivõime – mälu, kus info salvestub ka peale seadme väljalülitamist. Tegelikult see mälu ei tohiks mõjutada seadme kiirust, kuid autor on täheldanud enda VTTT (ja tegelikult ka teiste sarnaste seadmete juures), et ta seade töötab aeglasemalt, kui püsivõime on täis.

### 2.1.2 Tarkvara

Tarkvara all peab autor täpsemini silmas operatsioonisüsteemi ning erinevaid rakendusi, mis on tootja poolt ja kasutaja enda poolt paigutatud seadmele.

Esmapilgul võib tavakasutajale jääda mulje, et kõikidel tahvelarvutitel on sama operatsioonisüsteem (Android) ning erinevus esineb ainuüksi selle versioonis. Mõned võibolla võrdleks seda Windows'iga sellepärast, et Windows töötab sisuliselt ükskõik, mis riistvara peal, seni kuni riistvara on suuteline Windows'it jooksumise.

Tegelikult on Android võrreldav pigem Apple'i macOS'ga, sest macOS on disainitud töötama ainult spetsiifiliste riistvarakomponentide peal. Kui jooksumise seda teiste komponentide peal, siis kõige halvemal juhul ei lähe see lihtsalt tööle või töötab väga halvasti. Sama moodi on ka igal tahvelarvutil olev Android'i versioon kirjutatud ainult antud tahvelarvuti mudeli jaoks.

Ehk siis tootja saab Google'lt võtta Android'i lähtekoodi ning täiustada seda nii, et see töötaks laitmatult tema tahvelarvuti peal.

Tuntud tootjate tahvelarvutid (ja muud nutiseadmed) ongi sellepärast palju kiiremad võrreldes VTTT-dega, et neil on piisavalt ressursi, mida saab tarkvara arengusse investeerida. Eelnevas peatükis tõi autor välja, et VTTT-de tarkvara on suuresti sama ning just see ongi VTTT-de madala kiiruse põhjus. Nende Android'i versioone ei optimeerita samasuguste peensusteni välja nagu tuntud tootjate tahvelarvutite omi. Ehk siis tahvelarvuti kiirus võib sõltuda sellest, kui hea operatsioonisüsteem on sellele arendatud.

Lisaks operatsioonisüsteemile, võivad tahvelarvuti kiirust mõjutada tootjate poolt lisatud erinevad enda programmid, mida kasutaja ei saa kustutada nagu mõnda enda paigaldatud programmi. Tavaliselt need programmid peale püsimälu (halvemal juhul ka muutmälu) ruumi hõivamise midagi ei tee. Tuntud tootjate tahvelarvutitel võib selliste programmidega olukord olla halvem kui isegi VTTT-del ja on olnud juhtumeid, kus tuntud tootja tahvelarvuti töötab aeglaselt, sest kogu aeg jooksevad tootja enda programmid. Ka kasutaja enda programmid võivad seadet aeglustada, näiteks mõni selline programm, mis läheb tööle siis, kui tahvelarvuti käivitub. Kui mitu sellist programmi kogu aeg töötavad, siis tulemuseks on palju hõivatud muutmälu ruumi, millele järgneb tahvelarvuti aeglane töö.

### 2.1.3 Muu

Lisaks tark- ja riistvarale võivad tahvelarvuti kiirust mõjutada ka teised aspektid, nagu näiteks:

- Temperatuur – riistvaralised komponendid töötavad paremini või halvemini sõltuvalt temperatuurist. Arvutitel on hea õhutussüsteem, mille tõttu temperatuur on kontrollitav ja nad saavad kiiremini tööd teha. Tahvelarvutitel õhutus sisuliselt puudub ning tootjatel on tarvis teha igasuguseid trikke komponentide temperatuuri kontrollimiseks, mille tõttu näiteks külmades oludes tahvelarvuti kasutamine võib tuua esile igasugused anomaaliad, näiteks seade lülitub välja või lakkab reageerimast kasutaja sisendile .
- Seadme seisund – kui seade on näiteks kukkunud või kõva löögi saanud kasutajalt, mille tagajärjel mingi komponent seadme sees on lahti või katki läinud.
- Kasutusviis – see, kuidas kasutaja ise seadet kasutab, näiteks pidev ühe ja sama nupu vajutamine ilma peatuseta, mis ei anna seadmele aega töö teostamiseks.

- Tootjapoolne piirang – on olnud juhtumeid, kus tootjad ise piiravad seadme kiirust, näiteks tootjal on kaks sarnast seadet, kuid ühel on kiirus piiratud selleks, et teine tunduks parem ja oleks mõistlikum ost.

## 2.2 Täiustamise võimalused

Selles peatükis toob autor esile erinevaid võimalusi tüüpilise Android'i tahvelarvuti täiustamiseks. Kui eelmises peatükis oli mainitud nii tarkvara kui ka riistvara komponente, millest sõltub tahvelarvuti kiirus, siis selles peatükis keskendub autor rohkem tarkvarale, sest riistvara täiustamine kogemusteta tavakasutaja jaoks pole otseselt võimalik ja tegelikult nõuaks ka tahvelarvuti lähtekoodi muutmist.

Selle töö raames tahvelarvutite täiustamise võimalused jagatakse järgmiste tasemete järgi: kasutaja tase, juurõigustega kasutaja tase ja süsteemi tase. Lisaks nendele tasemetele oli esimeses peatükis paika pandud, et VTTT-del on kas Android 4 või 5, mille tõttu selles töös ei käsitleta täiustusvõimalusi, mis on võimalikud uuematel Android'i versioonidel.

### 2.2.1 Kasutaja tase

Enamus täiustustest, mida teha saab, on seotud pigem juurõigustega kasutaja ja süsteemi tasemega. On ka mõningaid lihtsamad lahendusi, mida saab ellu viia juurkasutaja õigusteta, kuid kahjuks on need lahendused väga piiratud. Kasutaja tase hõlmab endas kõike, mida kasutaja saab ilma igasugu muudatusteta operatsioonisüsteemis teha enda tahvelarvutil.

Eelnevas peatükis oli mainitud, et üks tahvelarvuti kiirust mõjutav aspekt on muutmälu. Kui korraga jookseb liiga palju programme madalamahulise muutmäluga tahvelarvutil, siis väga tõenäoliselt on see aeglane. Selle probleemi leevendamiseks saab kasutada niinimetatud „App killer“ tüüpi rakendust, mis suretab välja rakendusi mis on:

- 1) mällu laetud, aga pole mõnda aega kasutatud;
- 2) tuntud tootjate omad, mis asjatult jooksevad ja ei tee midagi;
- 3) süsteemi tasandil jooksutatud, mis ei tee midagi.

Veel üks lahendus oleks selliste rakenduste kustutamine, kuid tootja enda ja süsteemi rakenduste kustutamiseks on tavaliselt tarvis juurkasutaja õigusi.

Graafikakaardi töö madaldamiseks ja muutmälu mahu kokkuhoidmise eesmärgil võib proovida paigaldada kolmanda osapoole „launcher-it“. „Launcher“ on graafilise liidese üks osa. Sisuliselt „launcher“ vastutab tahvelarvutisisesest navigeerimisest, olgu see siis ekraani

lahtilukustamine, rakenduste käivitamine ja nende ikoonide sirvimine, menüüsse minemine ja paljud muud navigeerimise tegevused. Teise „launcher-i“ kasutuselevõtu eelis võib olla kiiremini töötav tahvelarvuti, sest uus „launcher“ võib kasutada vähem animatsioone või renderdada programme ikoonide väiksema resolutsiooniga. Tulemused loomulikult sõltuvad uuest „launcher-ist“ endast.

Väikese lisana võib siia tasandisse tuua ka mälukaardi lisamist oma seadmele. Kiiruse ja püsivõime vaba mahu vahel on seos, kus seade toimib kiiremini, kui tal on püsivõime vabam.

### **2.2.2 Juurõigustega kasutaja tase**

See tase on sisuliselt sama, mis eelmine tase, aga siin on kasutajal juurõigused. See tähendab, et kasutaja saab muuta teatud tahvelarvuti muutujaid, mis võivad seda mingil viisil kahjustada. Siin kehtivad samad täiustused, mis olid pakutud eelmisel tasandil.

Mõnede „App killer“ tüüpi rakenduste puhul on nüüd võimalik ka suretada välja süsteemi rakendusi, mida eelnevalt ei olnud võimalik teha puuduvate õiguste tõttu. Mõnede „launcher-ite“ puhul saab nüüd sama moodi kasutada eelnevalt puuduvate õiguste tõttu välja lülitatud funktsionaalsusi.

Sellel tasandil annab kõige parema tahvelarvuti kiiruse suurendamise protsessori ülekiirendamine (ingl over-clocking). See tähendab, et kasutaja saab paigaldada enda tahvelarvutile rakendusi, mis muudavad protsessori taktsagedust. Tootja poolt on protsessori taktsageduse vahemikud paika pandud nii, et seadmes saaks tekitada hea jõudluse ja aku kestvuse tasakaalu. Kasutajal, kes ei tea, mida ta teeb, ei tasuks neid sagedusi naljaga muuta, sest see protsess võib tõsiselt vigastada seadet, kuid kui seda teha õigesti, siis tulemused võivad olla väga head.

### **2.2.3 Süsteemi tase**

Tegelikult ei erine süsteemi tase mitte kuidagi juurõigustega kasutaja tasemest õiguste poolest. Vahe on pigem selles, kus täiustus rakendatud on. Sellel tasemel tehtavad täiustused on väga sügaval süsteemi tasemel.

Eelnevas alapeatükis mainitud keeld süsteemi rakenduste kustutamisele enam ei kehti ning kasutaja võib tahte järgi eemaldada ükskõik mida. Hea näide oleks rakendus „Phone“, mille abil saab teha kõnesid Android'i nutitelefonidel läbi sideoperaatori võrgu. Mõnedel VTTT-del

on see alles jäänud vaatamata puuduvale SIM-kaardi pesale, seega sellises olukorras oleks mõistlik see rakendus eemaldada. Tavaliselt süsteemi tööks kriitilisi rakendusi ei ole hea mõte eemaldada, kuid VTTT-del tihtipeale on ka tüüpilised kasutaja rakendused tehtud süsteemi rakendusteks. Autori enda kogemuse järgi on olnud selline juhtum, kus rakendus Youtube oli tehtud süsteemi rakenduseks, mida ei olnud võimalik uuendada, sest Android'i Play Store ei tunnustanud selle versiooni, ega kustutada, sest see oli süsteemi rakendus.

Veel üks väiksema mastaabi täiendus oleks eelnevates peatükkides mainitud build.prop faili muutmine. Tänu sellele failile saab muuta näiteks: muutmälu haldamist paremaks, audio ja video salvestamise kvaliteeti, internetiühenduse kiirust, aku ressursi haldamist, vigade ja muude sündmuste logimist ja palju muid aspekte ([Tweaks][Guide]Build.prop Tweaks, 2016). Selle muutmiseks on tarvis vaid mõnd tekstiredaktorit.

Selle taseme üks suurimaid täiendusi on kolmanda osapoole või enda tehtud ROM-i paigaldamine. XDA-Developers'ide (tuntud Android'i seadmete täiustamise kommuun) vikipeedia tõlgendab ROM-i kui faili, mis sisaldab kogu süsteemi kuvandit, milles omakorda on failid ja kood mis on vajalik seadme töölepanekuks. Lisaks sellele on ROM-is graafiline kasutajaliides, süsteemi ja muud rakendused ning kernel ise (ROM-VS-Kernel, 2013). Selle tõlgenduse järgi peaks olema arusaadav, et ROM-is, enne selle paigaldamist seadmele, saab muuta peaaegu kõiki aspekte, aga peab väga täpselt teadma, mida muudetakse. Kolmanda osapoole ning enda ROM-e tüüpiliselt luuakse olemasolevast tootja ROM-ist. Selline arendusviis soodustab kiiret ja asjalikku arendust, kuna arendaja käepärast on kõik süsteemi originaalfailid. ROM-e on erinevaid tüüpe, need mis: kiirendavad seadet, toovad mõned ilusad graafilised lahendused, muudavad seadme rohkem serveritüüpi seadmeks ja palju muid.

Viimane ja kõige suurem täiendus, mida kasutaja saab oma tahvelarvutile teha, on oma Android'i pordi kirjutamine tahvelarvutile. See on väga mahukas protsess, mis nõuab sügavaid Android'i süsteemi arhitektuuri teadmisi. Selle täienduse eeliseks on: sügav arusaam sellest, mis kasutaja seadmes toimub; võimalus kasutusele võtta uuemat Android'i versiooni; väga praktiline kogemus Android'i süsteemiga. Selle täienduse kohta on info suhteliselt kesine, mis teeb selle loomise mitu korda raskemaks.

### 3. Testimine ja tulemused

Eelmises peatükis olid paika pandud tüüpilise Android'i tahvelarvuti täiustamise võimalused. Selles peatükis katsetab autor paari võimalust enda VTTT peal.

#### 3.1 Testimismeetodid

Tulemuse jäädvustamiseks kasutab autor järgmisi teste:

- Antutu Benchmark– kolmanda osapoole rakendus Android'i operatsioonisüsteemil töötavate nutiseadmete jaoks, mille põhitest testib seadme riistvara komponente ja tagastab punktide summa. Selle töö käigus oli kasutusele võetud selle programmi versioon 5.7.5.
- Kasutustest – autori enda väljapakutud mitmeastmeline test. Enne testi algust seadmele tehakse taaskäivitus. Test algab seadme lahtilukustamisega. Sellele järgneb veebilehitseja avamine peamenüüst. Peale seda on vaja sisse trükkida ja minna lehele „[www.tlu.ee](http://www.tlu.ee)“. Test lõpeb, kui leht on ära laadinud. Selle testi juures võetakse arvesse ülesande täitmiseks kulunud aega.
- Sisselülitustest – järjekordne autori enda väljapakutud test, kus on tarvis seade lihtsalt tööle panna. Test algab, kui väljalülitatud seisundis seadmel vajutatakse sisselülitus nuppu. Test lõpeb, kui seadme ekraanile tekib lahtilukustamise vaade. Selle testi juures võetakse arvesse, kui kaua seadmel võtab aega tööle saamine.

Igat täiustust testitakse kolm korda. Tulemused on sajandiksekundilised, kuid nende keskmine on ümardatud kümnendiksekunditeni. Lisaks igat täiustust testitakse eraldi selleks, et tulemused oleksid võrreldavad teineteisega. Peale igat täiustuse testimist on täiustuse tulemuse keskmise võrdlus kontrolltesti keskmise tulemusega. Peale kõikide täiustuste läbitestimist rakendatakse kõik täiustused korraga ja testitakse veel üks kord.

#### 3.2 Testid

Oma VTTT täiustamiseks kasutab autor järgmisi lahendusi: asjatute rakenduste eemaldamine, „App killer“ tüüpi rakenduse kasutamine, teise „launcher'i“ kasutamine, protsessori ülekiirendamine ja build.prop faili muutmine. Selle töö raames ei testita uue ROM-i kasutuselevõttu ega Android'i portimist aja- ja oskustepuuduse tõttu.



### *Kontrolltest (Test 1)*

Enne kui saab alustada täiustuste testimist, oleks tarvis teha kontrolltest VTTT-le ilma igasugu täiustusteta, et hiljem oleks, millega võrrelda. Kontrolltesti tulemused on välja toodud allolevas tabelis (vt Tabel 1).

Tabel 1. Kontrolltesti tulemused

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	13251	50.15 sekundit	42.12 sekundit
2	12101	50.52 sekundit	41.96 sekundit
3	12667	48.07 sekundit	41.74 sekundit
Keskmine	12673	49.6 sekundit	41.9 sekundit

### *Asjatute rakenduste eemaldamine (Test 2)*

Esimene täiustus, mida autori VTTT-le rakendatakse, on asjatute rakenduste eemaldamine. Protsess toimub nii juhtõigustega kasutaja kui ka süsteemi tasemel. Rakendused, mida autor eemaldas, on järgmised: „Adobe Reader“, „Documents to Go“, „Aldiko“, „Adobe Flash Player 11.1“, „Hi-Q Audio Recorder“, „Koi Free“, „Live Wallpaper Picker“, „Talk“ ja „Maps“.

Tabel 2. Asjatute rakenduste eemaldamise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	11293	51.14 sekundit	41.78 sekundit
2	11425	50.22 sekundit	41.45 sekundit
3	11352	50.41 sekundit	40.92 sekundit
Keskmine	11356	50.6 sekundit	41.4 sekundit

Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 12% halvem, kasutustesti tulemus 2% halvem ja sisselülitustesti tulemus 1.3% parem.

### *„App killer“ tüüpi rakenduse kasutamine (Test 3)*

Selle täiustuse juures oli kasutusele võetud rakendus „App Killer“ (versioon 1.0.7). Enne kasutustesti algust oli see rakendus käivitatud ning mõttetult töötavad rakendused suretati välja. Sisselülitustest selle täiustuse juures ei ole aktuaalne, sest ei muudeta midagi sellist, mida VTTT muidu arvesse võtab sisse lülitades.

Tabel 3. „App killer“ tüüpi rakenduse kasutamise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	12953	35.58 sekundit	Pole aktuaalne
2	13643	42.89 sekundit	Pole aktuaalne
3	13539	30.44 sekundit	Pole aktuaalne
Keskmine	13378	36.3 sekundit	Pole aktuaalne

Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 6% võrra parem ning kasutustesti tulemus on ligikaudu 27% võrra parem.

#### *Uus „launcher“ (Test 4)*

Selle täiustuse juures oli kasutusele võetud „launcher“ „Zeam launcher“ (versioon 3.1.10), mis on väidetavalt üks kergemaid ning kiiremaid „launcher-eid“ saadaval Android'i seadmete jaoks (Fastest Android Launchers of 2017 {Top 10 Lightest}, 2017)

Tabel 4. Uue „launcher-i“ kasutamise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	10881	52.43 sekundit	40.96 sekundit
2	10857	53.24 sekundit	41.19 sekundit
3	12782	43.98 sekundit	39.63 sekundit
Keskmine	11506	49.9 sekundit	40.6 sekundit

Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 9,2% halvem, kasutustesti tulemus 0.6% halvem ja sisselülitustesti tulemus 3,2% parem.

#### *Protsessori ülekiirendamine (Test 5)*

Tegu on esimese täiustusega, mis nõuab juurkasutaja õigusi. Selle täiustuse juures oli kasutusele võetud protsessori ülekiirendamise rakendus „SetCpu“ (versioon 3.1.2). Algne sageduste vahemik oli 60MHz kuni 1.2GHz . Selle testi käigus olid sagedused muudetud vahemikule 360MHz kuni 1.2GHz, „governor“ valikuks oli valitud „fantasy“ ning oli valitud valik käivitada seade nende sätetega.

Tabel 5. Protsessori ülekiirendamise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	14278	45.29 sekundit	40.74 sekundit
2	13977	43.47 sekundit	39.84 sekundit
3	14190	47.35 sekundit	39.54 sekundit
Keskmine	14148	45.4 sekundit	40.0 sekundit

Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 12% parem, kasutustesti tulemus 9% parem ja sisselülitustesti tulemus 5% parem.

#### *build.prop faili muutmine (Test 6)*

Selle täiustuse juures olid lisatud autori VTTT build.prop faili sisse erinevad väärtused (vt Lisa 1).

Tabel 6. build.prop faili muutmise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	11283	79.74 sekundit	46.48 sekundit
2	11730	64.33 sekundit	44.66 sekundit
3	11452	68.63 sekundit	45.95 sekundit
Keskmine	11488	70.9 sekundit	45.7 sekundit

Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 10% halvem, kasutustesti tulemus 43% halvem ja sisselülitustesti tulemus 9% halvem.

#### *Kõik täiustused korraga (Test 7)*

Selles viimases testis rakendati autori VTTT-le kõik eelnevalt testitud täiustused korraga.

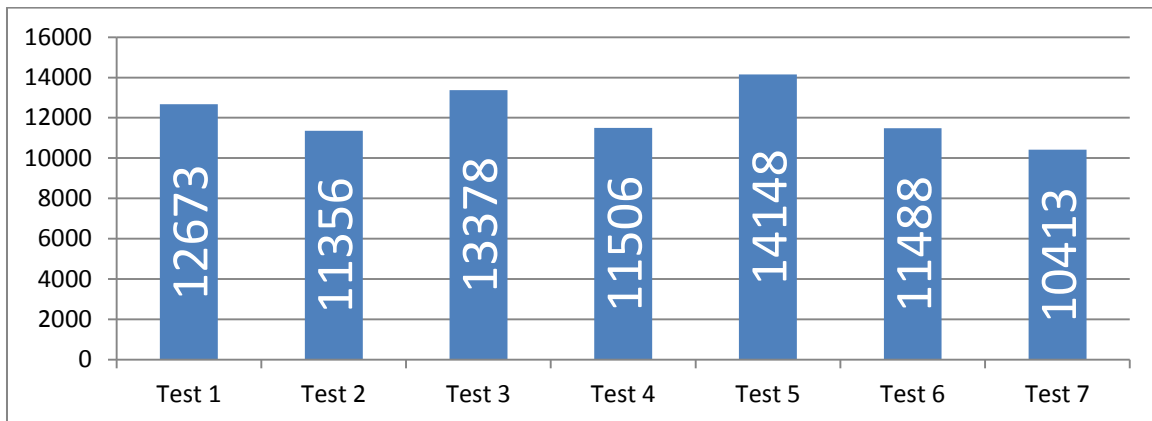
Tabel 7. Kõikide eelnevate täiustuste korraga kasutamise tulemus

Katse	Antutu Benchmark	Kasutustest	Sisselülitustest
1	10348	65.54 sekundit	40.13 sekundit
2	10998	56.43 sekundit	39.86 sekundit
3	9893	59.23 sekundit	39.72 sekundit
Keskmine	10413	60.4 sekundit	39.9 sekundit

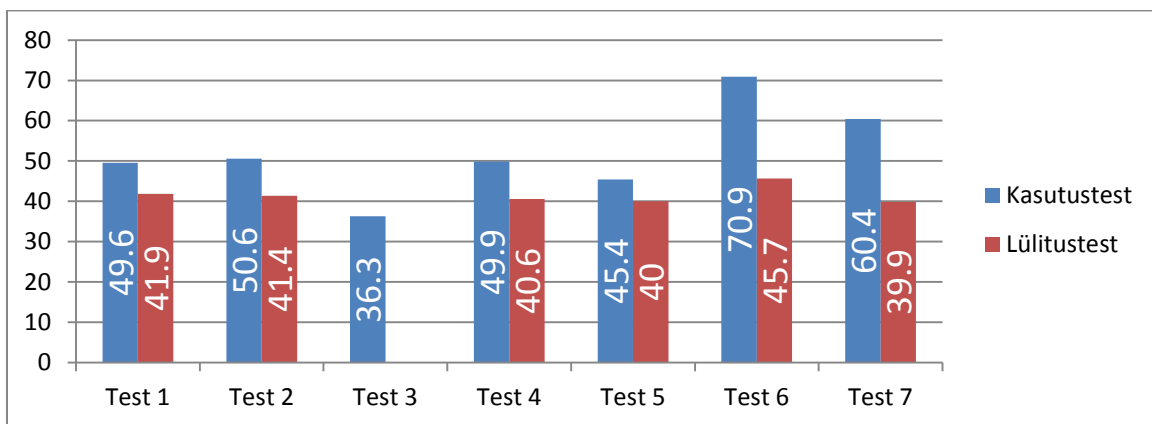
Kontrolltesti keskmisega võrreldes on selle täiustuse juures Antutu Benchmark-i tulemus 18% halvem, kasutustesti tulemus 22% halvem ja sisselülitustesti tulemus 5% parem.

### *Kõikide testide tulemused joonisel*

Loetavuse parendamiseks on kõik tulemused kantud allolevatele joonistele (vt Joonis 2). Antutu Benchmark'i tulemused on eraldi joonisel, sest selle tulemused on kordades suuremad teiste testi omadest (vt Joonis 1).



**Joonis 1. Kõikide testide Antutu Benchmark'i tulemus**



**Joonis 2. Kõikide testide kasutus- ja lülitustesti tulemused.**

### **3.3 Järeldused**

Eelnevate testitulemuste pealt on võimalik teha järgnevad järeldused:

- Kõige parem tulemus Antutu Benchmark-is oli protsessori ülekiirendamisel, tulemusega 14148. See oli mõnel määral oodatav tulemus, kui võtta arvesse, et üks osa Antutu Benchmark-ist testib protsessorit.
- Kõige parem tulemus kasutustesti puhul oli „App killer“ tüüpi rakenduse kasutamise juures, tulemusega 36.3 sekundit. Tõenäoliselt tagaplaanil väiksem arv jooksvaid

rakendusi ning vabam muutmälu maht võimaldas autori VTTT-l teha palutud tööd kiiremini.

- Kõige parem tulemus sisselülitustesti puhul oli kõikide täiustuste kasutamisel korraga, tulemusega 39.9 sekundit. Võimalik, et uue „launcher-i“ kasutuselevõtuga ning asjatute rakenduste eemaldamise tõttu oli autori VTTT-l esiteks tarvis vähem rakendusi muutmällu lugeda, kui käivitus käis, ning teiseks võimalik kiiremini graafiline liidese üles seada käivituse ajal. Selle tulemuse juures on huvitav, et build.prop faili muutmine andis vastupidiselt väga halva tulemuse võrreldes kontrolltesti omaga, kuna build.prop faili oli lisatud koodirida mis nõudis kiiret käivitumist.
- Kõige parem tulemus kokku oli protsessori ülekiirendamisel, Antutu Benchmark tulemusega 14148, kasutustesti tulemusega 45.4 ning sisselülitustesti tulemusega 40.0.
- Kõige halvem tulemus kokku oli build.prop faili muutmisel, Antutu Benchmark tulemusega 11488, kasutustesti tulemusega 70.9 ning sisselülitustesti tulemusega 45.7.

Kõige suurem järeldus mida saab teha saadud tulemuste põhjal on, et vaatamata sellele, et need täiustused võivad töötada tüüpiliste Android'i tahvelarvutite peal ei pruugi need töötada VTTT-de peal.

Autori eelnevate kogemuste põhjal oli autoril ootus, et kõikide täiustuste kasutamise tulemus tuleb mõnevõrra parem kontrolltesti omast, kuid see mis tegelikult tuli, oli väga suur üllatus autori jaoks. Hetkel, kui oli käimas kõikide täiustuste kasutamise test, töötas autori VTTT väga ettearvamatult. Päril tihti oli tarvis autori VTTT-d taaskäivitada, sest see lihtsalt seiskus.

Autori arvamuse järgi on tegu siiski meelega halvasti tehtud tarkvaraga tootja poolt sellepärast, et autori poolt varasemate Android'i nutiseadmete täiustamise jooksul eelnevalt pakutud täiustused vastupidi tegid seadme mõnevõrra kiiremaks.

Saadud tulemuste põhjal oskab autor pakkuda ainult kas uue ROM-i kasutuselevõttu, oma Android'i pordi kirjutamist või täielikku VTTT-de vältimist.

## Kokkuvõte

Selle töö jooksul olid paika pandud täpsed kriteeriumid, mille järgi saab VTTT ära tunda. Samuti oli töö jooksul vaadatud läbi, millest sõltub tüüpilise Android'i tahvelarvuti kiirus ning kuidas seda täiendada saab. Lõpuks olid tehtud eelnevalt väljapakutud täiustuste põhjal testid autori enda VTTT peal.

Siinse töö kõige olulisemad tulemused näitasid, et täiustused, mis teoorias pidid tegema autori VTTT kiiremaks, andsid vastupidise tulemuse, välja arvatud üksikute testide puhul. Kõige suurim järeldus oli, et täiustused mis võivad toimida tüüpiliste Android'i tahvelarvutite peal ei pruugi toimida VTTT-de peal.

Selle töö edasiarendamise võimalus oleks kas kirjutada enda ROM või Android'i port ning võrrelda siis tulemusi.

## Kasutatud kirjandus

Protsessor. (kuupäev puudub). Loetud aadressil <http://heiki.tpt.edu.ee/eucip/protsessor.html>

A13 Datasheet. (2012, 29.märts). Loetud aadressil [http://linux-sunxi.org/images/e/eb/A13\\_Datasheet.pdf](http://linux-sunxi.org/images/e/eb/A13_Datasheet.pdf)

SoC. (kuupäev puudub). Loetud aadressil <http://www.webopedia.com/TERM/S/SoC.html>

[Tweaks][Guide]Build.prop Tweaks. (2016, 11.mai). Loetud aadressil <https://forum.xda-developers.com/android/general/guide-build-prop-tweaks-t3376962>

ROM-vs-Kernel. (2013, 17.oktoober). Loetud aadressil <https://forum.xda-developers.com/wiki/ROM-VS-Kernel>

Fastest Android Launchers of 2017 {Top 10 Lightest}. (2017, 3.august). Loetud aadressil <http://mipbuzz.com/fastest-android-launcher/>

**LISAD**



## LISA 1. build.prop faili muudatused

```
ro.HOME_APP_ADJ=1
media.stagefright.enable-player=true
media.stagefright.enable-meta=true
media.stagefright.enable-scan=true
media.stagefright.enable-http=true
media.stagefright.enable-rtsp=true
media.stagefright.enable-record=false
profiler.force_disable_err_rpt=1
profiler.force_disable_ulog=1
net.tcp.bufferize.default=4096,87380,256960,4096,16384,256960
net.tcp.bufferize.wifi=4096,87380,256960,4096,16384,256960
net.tcp.bufferize.umts=4096,87380,256960,4096,16384,256960
net.tcp.bufferize.gprs=4096,87380,256960,4096,16384,256960
net.tcp.bufferize.edge=4096,87380,256960,4096,16384,256960
ro.mot.eri.losalert.delay=1000 (could brake tethering.)
ro.ril.power_collapse=1
pm.sleep_mode=1
wifi.suppllicant_scan_interval=180
ro.mot.eri.losalert.delay=1000
ro.config.hw_quickpoweron=true
ro.kernel.android.checkjni=0
ro.kernel.checkjni=0
dalvik.vm.checkjni=false
dalvik.vm.dexopt-data-only=1
dalvik.vm.heapstartsize=5m
dalvik.vm.heapgrowthlimit=48m
dalvik.vm.heapsize=64m
dalvik.vm.verify-bytecode=false
dalvik.vm.execution-mode=int:jit
dalvik.vm.lockprof.threshold=250
dalvik.vm.dexopt-flags=m=v,o=y
dalvik.vm.stack-trace-file=/data/anr/traces.txt
dalvik.vm.jmiopts=forcecopy
```

```
ro.config.nocheckin=1
wifi.suppllicant_scan_interval=180
pm.sleep_mode=1
ro.ril.disable.power.collapse=0
debug.performance.tuning=1
```