

Jussi Lappi

PONSSE OPTI6-MITTALAITTEEN TESTAUS

Testausmenetelmät ja huoltomiehen kenttätestaussalkun suunnittelu

PONSSE OPTI6-MITTALAITTEEN TESTAUS

Testausmenetelmät ja huoltomiehen kenttätestaussalkun suunnittelu

Jussi-Matti Tuomas Lappi
Opinnäytetyö
Syksy 2014
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

ALKULAUSE

Tämä insinöörityö on tehty opinnäytetyönä Oulun seudun ammattikorkeakoulun tekniikan yksikön tietotekniikan koulutusohjelmassa. Työn valvojana on toiminut lehtori Eino Niemi Raahen kampukselta.

Työn toimeksiantajana oli Ponsse Oyj:n Global Service. Insinöörityön tavoitteena oli kehittää metsäkoneen mittalaitteen testaus- ja huolto-ohjeita siten, että mittalaitteen kenttähuoltoa ympäri maailmaa tekevillä huoltomiehillä olisi käytettävissään vikadiagnostiikkaa helpottava ja yksinkertaistava työkalu ja toisaalta työkalu, joka helpottaa Suomessa toimivan huoltoneuvonnan työtä yhteydenoton jälkeen antaen nykyistä selkeämpiä viestejä mittalaitteiden vioista. Työn valvojana toimeksiantajan puolesta on toiminut Ponsse Oyj:n Global Servicen huoltopäällikkö Terho Tanskanen.

Haluan kiittää työn ohjaavaa opettajaa lehtori Eino Niemeä ja valvojaa huoltopäällikkö Terho Tanskasta sekä kaikkia muita projektiin osallistuneita.

Kinnulassa 20.11.2014

Jussi Lappi

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikan koulutusohjelma

Tekijä(t): Lappi Jussi-Matti Tuomas

Opinnäytetyön nimi: Ponsse Opti6-mittalaitteen testaus. Testausmenetelmät ja huoltomiehen kenttätestaussalkun suunnittelu.

Työn ohjaaja(t): Niemi Eino

Työn valmistumislukukausi ja -vuosi: Syksy 2014

Sivumäärä: 90

Tämä työ tehtiin Ponsse Oyj:n Global Service -osaston toimeksiannosta. Metsäkoneiden ohjausjärjestelmän muuttuessa monimutkaisemmaksi oli tarpeen suunnitella ja tuottaa työkalu, jonka avulla korjaamoilla ja kentällä työskentelevien huoltomiesten on mahdollista tehdä mittalaitteelle ja ohjausjärjestelmälle perusdiagnostiikkaa sekä tehdä diagnostiikan perusteella johtopäätökset siitä, milloin mittalaitte on toimitettava siihen erikoistuneen huollon korjattavaksi.

Työn aluksi perehdyttiin puutavaran mittaukseen yleisellä tasolla sekä mittalaitteen rooliin osana koneenohjausta. Tässä käytettiin aineistona sekä kirjallisuutta että omassa työssä Global Servicen huoltoneuvojana saatuja kokemuksia. Varsinaisen testaussalkun suunnittelun lähtökohdat ja tarpeet tulivat määriteltyä oman työn kautta ja mittalaitteen testauksen suunnittelussa ja toteutuksena suurimpana lähteenä oli oma työ ja siinä eteen tulleet tilanteet.

Työn tuloksena syntyi Ponsse Opti6 -mittalaitteen testaussalkku, joka myös tuotteistettiin Ponssin sopimushuoltopisteille myytäväksi erikoistyökaluksi. Testaussalkun avulla huoltomies pystyy tekemään perustestit mittalaitetietokoneelle sekä hänellä on mukanaan ohjausjärjestelmän peruskomponentit varaosina.

Testaussalkkua tullaan jatkokehittämään siten, että sitä pystytään hyödyntämään myös tulevaisuudessa esiteltävien uusien mittalaitetietokonesukupolvien testauksessa.

Asiasanat: metsäkoneet, mittaus, huolto, testaus, testausmenetelmät

ABSTRACT

Oulu University of Applied Sciences
Degree Programme in Information Technology

Author: Lappi Jussi-Matti Tuomas

Title of thesis: Testing of the Ponsse Opti6 Measuring System. Testing methods and design of the test case for a field service technician.

Supervisor: Niemi Eino

Term and year of completion: Autumn 2014

Number of pages: 90

The aim of this Bachelor's thesis was to design and productize a testing device of the Opti6 measuring device for Ponsse Plc's Global Service team. Over the past years the measuring and control systems of a forest machine have become more complicated and it was necessary to create a tool for workshop and field mechanics to be able to perform basic diagnostics for the measuring device. Based on this basic diagnostics the mechanic is better able to make the decision of taking the measuring computer to a workshop specializing in measuring device service.

The role of the computer in the measuring and machine control system is described in the beginning of the work. On this description both literature and my own experiences as a service trainer at Ponsse Global Service have been used as a source. In design of the measuring case and testing procedures, the most important sources were the cases and experience from my work.

As a result of this thesis a test case of the Opti6 measuring device was designed and productized as a special tool available through spare parts for workshops with a service contract to Ponsse Plc. With the help of the test case, the mechanic is able to perform diagnostics for the measuring system and also has the key components of the system available to use as spare parts, if needed.

The future development of the test case will include integration to the coming measuring systems so that also the newer generation systems can be diagnosed with the case.

Keywords: forestry machines, logging, measuring, service, testing, testing methods

Sisällys

ALKULAUSE	3
TIIVISTELMÄ	4
ABSTRACT	5
1 JOHDANTO	8
1.1 Opinnäytetyön aiheen tausta ja tavoite	8
1.2 Ponsse Oyj	8
2 PUUNKORJUU	11
2.1 Puunkorjuumenetelmät	11
2.2 Koneellisen puunkorjuun menetelmät	11
2.2 Tavaralajimenetelmässä käytettävät korjuukoneet	13
2.3 Puutavaran mittaus	14
2.3.1 Puutavaran mittauksen merkitys	14
2.3.2 Puutavaran mittauksen sääntely	15
2.3.3 Mittaustapahtuma hakkuukoneessa	16
2.4 Metsäkoneiden tietojärjestelmät	19
2.4.1 Tietojärjestelmän käyttö metsäkoneissa	19
2.4.2 Koneenohjaus	19
2.4.3 Puutavaran mittaus	20
2.4.4 Muut metsäkoneissa käytettävät ohjelmistot	22
3 MITTALAITTEIDEN HUOLTO	24
3.1 Mittalaitteiden huollon organisaatio	24
3.2 Mittalaittehuoltojen työkalut ja testausvälineet	25
3.2.1 Mittalaittehuollot korjaamoilla	25
3.2.2 Mittalaitteen testaus kentällä	26
3.3 Mittalaittehuollon työkalujen kehittämistarpeet	26
4 PONSSE-METSÄKONEEN OHJAUSJÄRJESTELMÄN RAKENNE	28
4.1 Opti6-tietokone	28
4.2 Koneenohjaus ja metsäkoneen tietoverkko	31
5 HUOLTOMIEHEN KENTTÄTESTAUSSALKUN VAATIMUSMÄÄRITTELY	34
5.1 Kenttätestaussalkun tarve	34
5.2 Kenttätestaussalkulle asetettavat vaatimukset	35

5.2.1 Eri testaustarpeet	35
5.2.2 Kenttäkelppoisuus	36
5.2.3 Taloudelliset seikat	37
5.2.4 Testaustulosten vertailtavuus	37
6 TOTEUTUS	39
6.1. Testaussalkun mekaaninen rakenne	39
6.2. Testaussalkun kytkentäkaavio sekä sähköinen toteutus	42
6.3. Testaussalkun ohjelmistot ja niiden käyttö sekä tulosten analysointi	46
6.3.1. Testaukseen käytettävät ohjelmistot	46
6.3.2 QA+WIN32 -ohjelmiston testien soveltaminen mittalaitetietokoneen testauksessa	47
6.3.3 Muut testaustarpeet	48
6.4. Testit ja niiden tulosten hyödyntäminen	49
6.4.1. Pikatesti	49
6.4.2. Rasitustesti	59
6.4.3. Arcnet-testi	59
6.4.4. GPS-testi	62
6.4.5. Stellaris-testi	63
6.4.6. HW-testi	64
6.4.7. Muut testausmahdollisuudet	65
6.5. Testaussalkun käyttöohjeet	66
6.6. Tuotteistus	66
7 TESTAUS	68
8 JATKOKEHITYSMAHDOLLISUUDET	69
9 YHTEENVETO	70
LÄHDELUETTELO	72
LIITTEET	74
Liite 1. Testaussalkun osaluettelo	74
LIITE 2. Testaussalkun johdinsarjakuva ja johdinsarjan osalista	76
LIITE 3. Testaussalkun kytkentäkaavio	78
LIITE 4. Esimerkki QA+WIN32-ohjelman testiraportista	80
LIITE 5. Pikatestin XML-koodi	86
LIITE 6. Rasitustestin XML-koodi	89

1 JOHDANTO

1.1 Opinnäytetyön aiheen tausta ja tavoite

Tämän opinnäytetyön tavoitteena on Ponsse Opti6 -mittalaitteen kenttätestausmahdollisuuksien kehittäminen. Ponsse Opti6 -mittalaite on tällä hetkellä viimeisin versio Ponssen Opti-mittalaitteperheestä ja se on otettu tuotantoon vuoden 2012 loppupuolella. Ensimmäisten mittalaitteiden takuu on näin ollen päättynyt vuoden 2013 loppupuolella, jolloin myös kentällä toimivien mittalaittehuoltojen on oltava valmiita suorittamaan laitteiden testausta sekä korjauksia. Koska vuosittain valmistetaan useita satoja metsäkoneita, jotka toimitetaan ympäri maailman ja joiden käyttöikä on varovaisestikin arvioiden vähintään 10 vuotta, tulee Opti6-mittalaitteen testaus ja huolto olemaan tärkeä osa mittalaittehuoltojen toimintaa tulevien vuosien aikana.

Tässä työssä keskitytään kartoittamaan ja kehittämään mittalaitteen testausmenetelmiä kehittämällä työkalu, jolla myös erityisesti mittalaittehuoltoon perehtymätönkin kenttähuoltomies pystyy tekemään perustestauksen metsäkoneen mittalaitteen eri komponenteille sekä hakemaan vikaa koneen ohjausjärjestelmästä. Toisaalta samalla työkalulla erikoiskoulutuksen saaneet mittalaittehuoltajat pystyvät tekemään pidemmälle vietyä diagnostiikkaa ongelmakoneiden ohjausjärjestelmälle. Kolmantena keskeisenä tavoitteena kenttätestaustyökalulle on, että sen mukana kenttähuoltomiehellä ovat ohjausjärjestelmän keskeisimmät osat, jotta niitä ei tarvitse erikseen tilata tai kuljettaa mukana huoltokäynnille mennessä, vaan huoltokäynnillä koneeseen mahdollisesti tarvittavat ohjausjärjestelmän varaosat löytyvät testaussalkusta.

1.2 Ponsse Oyj

Ponsse Oyj on vuonna 1970 perustettu teknologiateollisuuden yritys, jonka kotipaikka on Vieremällä. Ponsse on erikoistunut yksinomaan tavaralajimenetelmän puunkorjuukoneiden valmistukseen. Yrityksen perustaja, Einari Vidgrén rakensi ensimmäisen metsätraktorin oman puunkorjuuyrityksensä käyttöön vuonna 1969, koska silloin saatavilla olleet

metsäkoneet eivät olleet riittävän kestäviä. Koneen toimivuus, kestävyys sekä alhaiset ylläpitokustannukset kiinnostivat myös yrityksen ulkopuolisia asiakkaita ja tämän kysynnän vuoksi Ponsse aikanaan perustettiin. (Ponsse Oyj historia 2014.)

Alkuvuosina koneita valmistui muutaman koneen vuosivauhtia. Varsinainen valtakunnallinen tunnettavuus Ponselle syntyi 1980-luvun alkupuolella esitellyn S15-mallin ansiosta. 1980-luvun puolivälissä Ponsse aloitti puun kaatoon, karsintaan sekä katkontaan käytettävien harvesteripäiden valmistuksen ja vuonna 1986 asiakkaalle toimitettiin ensimmäinen läpimittaa mittaavalla mittalaitteella varustettu metsäkone. (Ponsse Oyj historia 2014.)

Vuonna 1988 Einari Vidgrén myi Ponssen koko osakekannan, jonka seurauksena Ponssesta tuli osa Norcar-metsäkoneyhtymää. Vuonna 1993 Ponsse siirtyi takaisin Einari Vidgrénin vetämän sijoittajaryhmän omistukseen ja samalla metsäkoneiden mittalaitteita valmistava Kajaanin Automatiikka Ky fuusioitiin Ponsseen. 1993 on myös mittalaitteiden kehityksen kannalta merkittävä vuosi, koska tällöin Ponsse toi markkinoille PC-tietokoneeseen pohjautuvan mittalaitteen ensimmäisenä valmistajana maailmassa. (Ponsse Oyj historia 2014.)

Ponssen kansainvälistyminen ja voimakas kasvuvaihe alkoivat 1990-luvun puolivälissä, jolloin perustettiin tytäryhtiöt ensin Ruotsiin ja sen jälkeen mm. Yhdysvaltoihin, Ranskaan ja Iso-Britanniaan. Kansainvälistyminen jatkui voimakkaana 2000-luvulla ja vuonna 2004 Ponselle myönnettiin Tasavallan Presidentin kansainvälistymispalkinto menestyksekkästä kansainvälisestä toiminnasta. Vuonna 2004 Ponssesta tuli myös seinäjokisen, työkoneiden ohjausjärjestelmiin erikoistuneen Epec Oy:n pääomistaja. (Ponsse Oyj historia 2014.)

2010-luvulla Ponssen kehitys on jatkunut vakaana ja myös uusille markkina-alueille on edetty harkiten. Yritys panostaa jatkuvasti ja vahvasti

tuotekehitykseen lähes 100 työntekijän voimin. Tuotekehityksen perustana on yrityksen perustajan Einari Vidgrénin periaate siitä, että koneiden käyttäjät ovat parhaita asiantuntijoita ja heitä pitää kuunnella koneita kehitettäessä. Vahvasta panostuksesta tuotekehitykseen kertoo vuoden 2013 toukokuussa ensiesittelynsä saanut Scorpion-harvesteri, joka sisältää monia alalla täysin ennen näkemättömiä ratkaisuja. (Ponsse Oyj historia 2014; Ponsse Oyj tuotekehitys 2014.)

Ponsse Oyj:n liiketoiminnassa on kolme tärkeää osa-aluetta. Metsäkoneiden ja niiden tietojärjestelmien valmistuksen lisäksi niihin liittyvien palveluiden tuottaminen asiakkaille koko metsäkoneen elinkaaren ajan on tärkeä osa yrityksen strategiaa. (Ponsse Oyj liiketoiminta-alueet 2014.)

2 PUUNKORJUU

2.1 Puunkorjuumenetelmät

Maailmassa on käytössä hyvin monenlaisia puunkorjuumenetelmiä erilaisiin olosuhteisiin, joissa korjuutyö suoritetaan. Niitä jaotellaan usealla eri tavalla, mutta pääjaottelut tehdään useimmiten joko korjuun koneellistamisasteen tai sen mukaan, missä muodossa puu kuljetetaan tienvarsivarastolle tehtaalle kuljettamista varten. (Uusitalo 2003, 53–57.)

Jaoteltaessa puunkorjuuta koneellistamisasteen mukaan jaetaan hakkuumenetelmät joko käsityönä ihmisvoimin tehtävään hakkuuseen tai koneelliseen hakkuuseen. Vähintään puolet maailmassa tehtävästä puunkorjuusta tehdään edelleen käsityönä, käyttäen puun kaatovälineenä joko moottorisahaa tai jopa käsityökaluja. Usein myös käsityönä kaadettavien puiden kuljetuksessa tienvarteen käytetään apuna erilaisia koneita, mutta kehitysmaissa myös eläinten avulla tehtävällä puun metsäkuljetuksella on edelleen merkittävä osuus. Koneellisessa puunkorjuussa kaikki tai ylivoimaisesti suurin osa korjuun työvaiheista on koneellistettu. Koneellisen puunkorjuun historiaan sisältyy monia erilaisia vaiheita ja kokeilujakin, sekä monista erilaisista koneista että eri korjuumenetelmistä. Länsimaissa puunkorjuun menetelmät ovat vakiintuneet muutamiin vakiomenetelmiin, mutta kehitysmaissa menetelmien kirjo on vielä hyvinkin laaja. (Uusitalo 2003, 56–57. Ponsse Oyj toimiala 2014.)

2.2 Koneellisen puunkorjuun menetelmät

Koneellistetussa puunkorjuussa on käyttöön vakiintunut kaksi eri menetelmää: kokorunkomenetelmä sekä tavaralajimenetelmä. Kokorunkomenetelmässä puut kaadetaan metsässä joko metsurityönä tai erityisellä kaato-kasauskoneella, minkä jälkeen ne kuljetetaan kokonaisina runkoina tienvarsivarastoon, jossa ne karsitaan. Tämän jälkeen rungot kuljetetaan kokonaisina tehtaalle jatkojalostukseen tai katkotaan varastopaikalla määrättyihin mittoihin ennen

jatkokuljetusta. Puutavaran mittaus suoritetaan tässä menetelmässä lähes aina puutavaraa käyttävällä tuotantolaitoksella. (Uusitalo 2003, 57.)

Tavaralajimenetelmässä puu kaadetaan, karsitaan ja katkotaan valmiiksi puutavaralajeiksi käyttötarkoituksen mukaan jo metsässä, josta eri tavaralajit siirretään useaan eri varastopinoon tien varteen jatkokuljetusta varten. Hakkuu tehdään useimmiten hakkuukoneella (harvesteri) ja metsäkuljetus sitä varten tehdyllä metsätraktorilla (ajokone, kuormatraktori). Puutavara voidaan mitata tavaralajimenetelmää käytettäessä koneellisesti heti korjattaessa tai puutavaraa käyttävillä tuotantolaitoksilla. Pohjoismaissa mittaus suoritetaan lähes aina metsäkoneen mittalaitteella jo katkonnan yhteydessä metsässä. (Uusitalo 2003, 56–57.)

Tällä hetkellä maailmanlaajuisesti kokorunkomenetelmän osuus on noin 60 % ja tavaralajimenetelmän noin 40 % koneellistetusta puunkorjuusta. Tavaralajimenetelmän osuus nousee hiljalleen koko ajan johtuen useista eri tekijöistä. (Ponsse Oyj toimiala 2014.)

Tavaralajimenetelmän etuina erityisesti alueilla, joilla osaavan työvoiman saanti on vaikeaa tai liian kallista, on pieni työvoiman tarve verrattuna kokorunko- tai käsityöhakkuuseen, koska koko hakkuuketjussa on vain kaksi konetta. Tällöin myös polttoaineenkulutus on pienempi tuotettua puukuutiota kohden. Lisäksi puun käyttö pystytään suunnittelemaan huomattavasti tarkemmin, koska hakkuukoneen mittalaite optimoi puun jaon eri tavaralajeihin tietokoneavusteisesti, jolloin kuljettajan tehtäväksi jää ainoastaan muuttaa mittoja, mikäli korjattavassa puussa on vikoja, jotka alentavat puun laatua ja aiheuttavat siten mittojen tai tavaralajin vaihdon. Tavaralajimenetelmää käytettäessä säästetään myös logistiikkakustannuksissa verrattuna kokorunkomenetelmään, koska tavaralajit voidaan kuljettaa suoraan niiden käyttöpaikkoihin ilman erillistä lajittelua joko ensimmäisessä käyttöpaikassa tai erillisessä terminaalissa. (Ponsse Oyj toimiala 2014.)

Koneellisen puunkorjuun osuus on Suomessa tällä hetkellä jo yli 90 % kaikesta puukaupasta ja teollisuuden pystykaupoista koneellisesti korjataan jo lähes kaikki puu. Tiettyjä maisemallisesti arvokkaita, maaston vuoksi erityisen hankalia tai esimerkiksi kaupunkiympäristössä sijaitsevia kohteita hakataan edelleen käsityönä. Samoin maanomistajien itse korjaamista hankintakaupapuista suuri osa on edelleen käsityönä hakattuja puita. (Uusitalo 2003, 63. Ponsse Oyj toimiala 2014.)

2.2 Tavaralajimenetelmässä käytettävät korjuukoneet

Pohjoismaista alkunsa saanut täysin koneellistettu tavaralajimenetelmä perustuu kahteen erilliseen koneeseen: hakkuukoneeseen eli harvesteriin, kuten konetta yleensä kutsutaan sekä kuormatraktoriin eli ajokoneeseen. (Uusitalo 2003, 57.)

Nykyisin käytettävät harvesterit ovat suurimmilta osin kuormainharvestereita. Tällä tarkoitetaan sitä, että kaikki puun käsittelyn vaiheet – kaato, karsinta, katkonta ja mittaus – suoritetaan nosturiin kytketyllä hakkuupäällä. Peruskoneen, nosturin sekä hakkuupään koko vaihtelee eri harvestereiden välillä suuresti, jolloin pienirunkoisia harvennuksia tekevä urakoitsija voi hankkia pienemmän ja sitä kautta yleensä edullisemmän koneen kuin ainoastaan suuripuustoista päätehakkuuta tekevä urakoitsija. Erityisesti Suomessa, mutta myös muualla Pohjoismaissa, ns. yleiskoneet, joilla pystytään tekemään lähes kaikki leimikot pienirunkoisista harvennuksista päätehakkuuseen, ovat viime vuosina muodostuneet käytetyimmäksi koneryhmäksi. (Uusitalo 2003, 69–71.)

Hakkuutyö harvesterilla jakautuu käytännössä kolmeen työvaiheeseen. Ensimmäinen vaihe on puun kaato, jolloin pystyssä olevaan puuhun tartutaan kiinni, se sahataan poikki ja kaadetaan hakkuupään ja nosturin avulla haluttuun suuntaan. Toinen vaihe on puun karsinta, jolloin puuta syötetään hakkupään läpi ja hakkuupäässä olevat karsintaterät karsivat rungosta oksat. Kolmannessa vaiheessa syöttö pysäytetään haluttuun mittaan ja puu katkotaan pituuden, läpimitan ja laadun perusteella eri tavaralajeiksi. Koko hakkuutyön taloudellisen

tuloksen kannalta tärkein työ eli puun laadun sekä eri tavaralajeiksi katkongan määrittäminen alkaa jo puuta kaadettaessa ja jatkuu varsinaiseen katkontasahaukseen saakka. Tätä puun laaduttamista ja mittaan jakamista kutsutaan apteraukseksi. Apterauksen yhteydessä puu mitataan ja mittalaite rekisteröi jatkuvasti puun tilavuuden sekä pölkkyjen pituusmitat muistiin. Apteraus sekä kuormaimen käsittelyn ja harvesteripään ohjaus vaativat kuljettajalta erinomaista koordinaatiota, hahmotuskykyä sekä jatkuvaa nopeaa päätöksentekoa hakkuun aikana. (Uusitalo 2003, 69–73, 151–154.)

Tavaralajeiksi katkottujen puiden kerääminen metsästä sekä kuljetus varastopaikalle tapahtuvat kuormaa kantavalla koneella. Puutavaran ajossa aiemmin käytetyt maataloustraktoriin perustuvat yhdistelmät ovat nykyisin väistyneet lähes täysin tarkoitusta varten erityisesti suunniteltujen kuormatraktoreiden tieltä. Metsäkuljetuksen työvaiheet ovat karkeasti seuraavat: ajo palstalle, kuorman keräys ja ajo keruun aikana, ajo varastopaikalle sekä kuorman purku varastopaikalle. Kuormatraktoreiden koko vaihtelee harvestereita enemmän sen mukaan, ajetaanko niillä puuta harvennus- vai päätehakkuuleimikoilta. Myös kuormatraktorit ovat erityisesti Suomessa kehittymässä kohti yleiskoneita, joilla pitää pystyä ajamaan kaikkia leimikkotyyppisiä. (Uusitalo 2003, 80–81.)

Harvesterin ja kuormatraktorin muodostamaa yksikköä kutsutaan metsäalalla useimmiten hakkuuketjuksi, missä yhdellä harvesterilla hakatut puut ajetaan yhdellä kuormatraktorilla varastopaikalle. Pääsääntöisesti yhdellä hakkuuketjulla on töissä neljä kuljettajaa, jolloin molemmat koneet toimivat kahdessa työvuorossa.

2.3 Puutavaran mittaus

2.3.1 Puutavaran mittauksen merkitys

Puutavaran mittauksella tarkoitetaan tietyn puutavaraerän tilavuuden ja laadun määrittämistä. Puutavaran mittaustietoja käytetään sekä metsänomistajalle puista maksettavan kauppahinnan määrittelyyn että puunkorjuun ja kuljetuksen

urakointimaksujen tai metsätyöpalkkojen määrittelyyn. Mittaustietoja voidaan tarvittaessa käyttää myös erilaisiin valvonta-, seuranta- ja suunnittelutarkoituksiin, esimerkiksi metsänomistajan metsäsuunnittelun apuna tai metsäyhtiön raaka-ainevirtojen ohjauksessa. (Metsätalouden kehittämiskeskus Tapio 2008, 383.)

Puutavaran mittaus voidaan suorittaa usealla eri tavalla ja mittauksen suorittamistavasta sovitaankin Suomessa aina puukauppaa tehdessä kirjallisesti. Puun mittaus voidaan suorittaa puun ostajan tai myyjän toimesta usealla eri menetelmällä. Ylivoimaisesti yleisin tapa puutavaraa mitatessa on nykyisin hakkuun yhteydessä harvesterin mittalaitteella suoritettava mittaus, jolloin kauppahinta määritellään metsäkoneesta saatavan mittaustodistuksen perusteella. Muita käytettäviä menetelmiä ovat tehdasmittaus, jossa mittaus suoritetaan puut vastaanottavalla tuotantolaitoksella, tienvarsimittaus, jossa mittaus suoritetaan käsityönä ostajan ja myyjän toimesta, tai painoon perustuva mittaus, jossa mittaus suoritetaan kuormatraktoriin asennetun kuormainvaan punnitustuloksen perusteella. (Metsätalouden kehittämiskeskus Tapio 2008, 383.)

Maailmanlaajuisesti ajatellen eri mittausmenetelmien käyttö on Suomeen ja pohjoismaihin verrattuna huomattavasti kirjavampaa. Pääsääntönä voidaan kuitenkin todeta, että yleisimmät mittausmenetelmät hakkuukoneilla hakattuja puita mitattaessa ovat joko tehdas- tai painoon perustuva mittaus. Hakkuukonemittausta ei vielä kovinkaan yleisesti käytetä ainakaan kauppahinnan määrittämiseen pohjoismaiden ulkopuolella, vaan metsäkoneen mittalaitetta käytetään pääasiassa eri tavaralajien pituus- ja läpimittavaatimusten seurantaan puuta katkottaessa.

2.3.2 Puutavaran mittauksen sääntely

Puutavaran mittauksen sääntely vaihtelee voimakkaasti eri puolella maailmaa. Suomessa puutavaran mittauksista on säädely lainsäädännöllä vuodesta 1939 saakka ja viimeisin voimassa oleva laki puutavaran mittauksesta on tullut

voimaan 14.6.2013. Eri mittausmenetelmistä ja niiden käytöstä säädellään asetuksella, jossa määrätään mittausmenetelmistä, niihin käytettävien laitteiden yleisistä vaatimuksista sekä mittauksen luotettavuuden varmistamisesta ja valvonnasta. (Metsätalouden kehittämiskeskus Tapio 2008, 284; L 14.06.2013/414, 1. luku 1§; AS 17.6.2013/12, 2.)

Puutavaran mittausta koskevan lainsäädännön tarkoituksena on turvata luotettavuus mittauksen eri osapuolien kannalta ja antaa yksityiskohtaiset määräykset mittausmenetelmistä, mittauksessa käytetyistä laitteista sekä toiminnasta mittauksen valvonnan, tarkastuksen sekä mahdollisten erimielisyyksien osalta (Metsätalouden kehittämiskeskus Tapio 2008, 384).

2.3.3 Mittaustapahtuma hakkuukoneessa

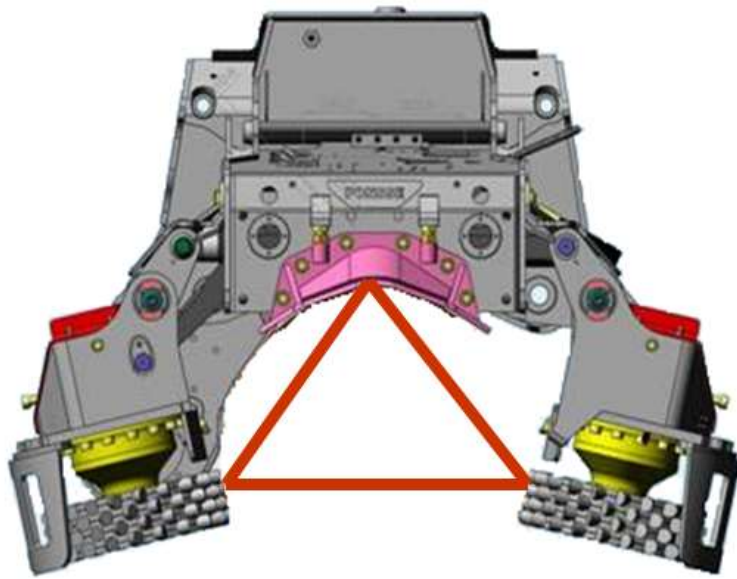
Puun mittaus on teknisesti hyvin vaativa toimenpide, sillä puun laatu ja sen fyysiset ominaisuudet, kuten kovuus, paino, kuoren rakenne, oksaisuus ja mutkaisuus, vaihtelevat jatkuvasti. Lisäksi mittaukseen vaikuttavat voimakkaasti ympäristön olosuhteet ja vuodenajat, esimerkiksi puun jäätyminen talviaikana tai puun kuoren irtoamisherkkyiden vaihtelut puun kasvuvaiheen mukaan. Tämän vuoksi mittauksen onnistumiseen vaikuttaa suuresti koneen ominaisuuksien lisäksi koneen kuljettajan ammattitaito sekä mittalaitetta käytettäessä että suoritettaessa sen kalibrointia työn aikana. (Metsätalouden kehittämiskeskus Tapio 2008, 384.)

Hakkuukonemittaus on hyväksytty Suomessa viralliseksi mittausmenetelmäksi vuonna 1992 ja se yleistyi erittäin nopeasti heti hyväksyntänsä jälkeen. Nykyisin yli 80 % Suomessa myytävästä yksityismetsien puutavarasta mitataan hakkuukoneen mittalaitteella. Hakkuukonemittauksella on monia suuria etuja. Hakkuukoneella mitattaessa saadaan puutavara mitattua reaaliaikaisesti puuta korjatessa sekä urakointimaksujen että puukauppatilitysten maksua varten. Lisäksi kuljettajalla on jatkuvasti käytössään puutavaran mittatiedot, jolloin puun jakaminen eri tavaralajeiksi on paitsi tehokasta myös oikeaa. Nykyisten korjuunohjauksen tietojärjestelmien avulla metsäkoneisiin pystytään

lähettämään puutavaralajien mittatietoja jatkuvasti, jolloin metsäyhtiö pystyy ohjaamaan reaaliaikaisesti metsästä korjattavan puutavaran mittoja ja laadutusta esimerkiksi sahalle tulleiden valmiin sahatavaran tilausten pituus- tai laatuvaatimusten perusteella. (Uusitalo 2003 151–153; Metsätalouden kehittämiskeskus Tapio 2008, 386–387.)

Hakkuukoneen mittalaite mittaa käsiteltävän puun pituutta ja läpimittaa jatkuvasti puun syötön, karsinnan ja katkonnan aikana. Pituus mitataan nykyisin lähes poikkeuksetta runkoa vasten pyörivän mittarullan avulla. Pituutta mitattaessa anturina käytetään yleensä pulssianturia, jolloin saavutetaan anturin resoluutiosta riippuen tasalaatuisen puutavaran mittauksessa teoreettiseksi tarkkuudeksi 2–3 mm. Asetuksessa määrätty minimiresoluutio pituusmitalle on 1 cm tai vähemmän. Käytännössä puun pinnan kovuuden ja laadun vaihtelu sekä viat aiheuttavat sen, että käytännön katkontatarkkuusvaatimus tuotantolaitoksilla vaatii useimmiten pölkkyjen pituusmitan olevan $-3...+3$ cm tavoitemitasta vähintään 80%:ssa pölkkyissä. (Metsätalouden kehittämiskeskus Tapio 2008, 386–387; AS 17.6.2013/12, 3.)

Puun läpimitta mitataan pölkyn pituusakselia vastaan kohtisuoran poikkileikkauspinnan halkaisijana. Mittaus suoritetaan yleisimmin mittaamalla hakkuupäästä joko karsimateriaalien tai syöttörullien asentoa. Läpimitan mittauksessa käytetään antureina joko potentiometriä tai pulssianturia, jolla saavutetaan terien tai rullien asennon, ja näin ollen puun halkaisijan mittauksessa useimmiten alle 0,5 mm:n tarkkuus. Asetuksessa määrätty minimitarkkuus läpimitan mittauksen resoluutiolle on 1 mm. Varsinainen mittaus perustuu yleensä kuvan 1 mukaiseen kolmiomittaukseen, joka käytännössä tarkkuudeltaan vastaa puusta ristiin mitatun läpimitan keskiarvoa. (Metsätalouden kehittämiskeskus Tapio 2008, 386–387; AS 17.6.2013/12, 3.)



KUVA 1. Kolmiomittauksen periaate

Hakkuukoneella mitattaessa mittalaite muodostaa läpimittahavainnoista ns. runkokäyrän, jossa rungon läpimitta on tallennettu 1–10 cm välein puuta syötettäessä. Rungon ja yksittäisen pölkyn kokonaistilavuus saadaan tällöin laskettua lisäämällä kyseisten kappaleiden tilavuudet yhteen. Koko puu mitataan samalla periaatteella, ja kuljettajan tehtävä on puuta käsitellessä huolehtia siitä että läpimitan ja laadun perusteella eri tavaralajeiksi jaettavat puun osat tulevat rekisteröityä oikeisiin tavaralajeihin. Rungon tyven tilavuus lasketaan erityisen tyvifunktion avulla, koska harvesteripään fyysisten rajoitteiden vuoksi rungosta ei voida saada mitattua läpimittatietoa ensimmäisten muutamien kymmenien senttimetrien matkalta. (Metsätalouden kehittämiskeskus Tapio 2008, 386–387; AS 17.6.2013/12, 3.)

Puun viat, puun laatu, eri puulajien ominaisuudet sekä korjuuolosuhteet vaikuttavat myös puun läpimitan mittauksen tarkkuuteen. Tämän vuoksi mittalaite on kalibroitava säännöllisesti. Kalibrointi suoritetaan useimmiten mittasaksilla, jolloin saksilla käsin mitattua tulosta verrataan hakkuukoneen mittalaitteen mittaustulokseen. Tämän jälkeen hakkuukoneen mittalaitteen kalibrointia voidaan muuttaa kalibrointimittauksen tulosten perusteella. Kalibrointimittaukseen käytettävien pölkkyjen määrä, kalibrointimittauksen

toistotiheys sekä mittalaitteen tarkkuuden suurimmat sallitut erot on säädelty mittausta koskevassa lainsäädännössä. Mittalaitteen tarkkuuden omavalvontaa suorittaa pääsääntöisesti koneen kuljettaja ja lisäksi urakanantajien on suoritettava säännöllisesti tarkastusmittauksia koneiden mittalaitteille. (Metsätalouden kehittämiskeskus Tapio 2008; 386–387. AS 17.6.2013/12, 11–14.)

2.4 Metsäkoneiden tietojärjestelmät

2.4.1 Tietojärjestelmän käyttö metsäkoneissa

Nykyaikaisessa metsäkoneessa tietojärjestelmää käytetään useaan eri tarkoitukseen. Varsinaisen puun mittauksen sekä mittaustietojen tallennuksen lisäksi saman tietojärjestelmän avulla huolehditaan koko metsäkoneen koneenohjauksesta, sekä kone- ja kuljettajakohtaisten säätöjen tekemisestä ja tallentamisesta. Lisäksi tietojärjestelmän avulla suoritetaan puun katkonnan optimointia, joka onkin yksi modernin metsäkoneteknologian ja siihen liittyvän tutkimuksen suurimmista ongelmista.

Käytännössä kaikkien suurten metsäkonevalmistajien tietojärjestelmät perustuvat tällä hetkellä Microsoftin Windows-käyttöjärjestelmää käyttävään PC-tietokoneeseen sekä tähän joko Arcnet- tai CAN-väylällä toteutettuun verkkoon, johon on liitetty koneen eri ohjainmoduulit. Joillakin pienemmillä valmistajilla on käytössä myös Linux-pohjaisia mittalaitteita, mutta näiden ongelmaksi muodostuu metsäyhtiöiden useimmiten käyttämien ohjelmistojen Windows-pohjaisuus, jolloin metsäyhtiöiden ohjelmistot saatetaan joutua asentamaan erilliseen tietokoneeseen.

2.4.2 Koneenohjaus

Koneenohjaus on metsäkoneessa toteutettu useimmiten useamman eri ohjainmoduulin avulla, joista kukin ohjaa yhtä metsäkoneen osaa tai vastaanottaa kuljettajan tekemiä ohjauslaitteiden liikkeitä. Useamman moduulin

käyttöön on käytännössä jouduttu sen vuoksi, että tällöin ei ajauduta reaaliaikaohjelmistossa liian usean rinnakkaisen prosessin käyttöön, mikä voisi aiheuttaa viiveitä koneen toiminnoille. Metsäkoneessa on käytössä lähes koko ajan useita rinnakkaisia toimintoja, joita on kyettävä säätämään lähes ilman viiveitä. Tällöin sekä prosessien ajoituksen että niiden priorisoinnin huolellinen suunnittelu ohjelmistossa on erittäin tärkeää. Esimerkiksi puuta syötettäessä puun nopeus kourassa voi nykyisissä tehokkaissa koneissa olla jopa 7 m/s. Tällöinkin puu on pystyttävä pysäyttämään hallitusti yhden senttimetrin tarkkuudella oikeaan kohtaan katkaisua varten, jolloin jo muutaman millisekunnin viivästys ohjauksessa aiheuttaa pysäytyksen väärään kohtaan ja näin ollen korjaavan syöttöliikkeen tarpeen.

Koneenohjauksessa voi olla käytössä esimerkiksi seuraavat moduulit: moottorinohjausyksikkö, nosturinojausyksikkö, voimansiirron ohjausyksikkö, penkin ohjausyksikkö sekä harvesteripään ohjausyksikkö. Lisäksi varsinainen tietokone sisältää käyttöliittymän kuljettajalle.

2.4.3 Puutavaran mittaus

Puutavaran mittaus voidaan toteuttaa joko hakkuukoneella harvesterikouralla tai ajokoneessa olevalla kuormainvaa'alla. Hakkuukoneella tehtävällä mittauksella on kaksi tarkoitusta. Ensimmäisenä mitataan ja tallennetaan hakatun puutavaran tilavuustiedot, jotka ovat useimmiten myös puusta maanomistajalle maksettavan puutavaratilityksen perusteena. Toisena ja sekä maanomistajan että teollisuuden kannalta vähintään yhtä tärkeänä käyttötarkoituksena mittaustiedoille on puun katkonnan ja tavaralajeiksi jaottelun ohjaus. (Uusitalo 2003, 155–157.)

Katkonnan ja eri tavaralajien optimointi on yksi modernin metsäkoneohjelmiston tekemisen ja siihen liittyvän tutkimuksen keskeisistä osista. Nykyään tavaralajeiksi katkontaa ohjataan niin sanotulla arvoapteerauksella, jonka tarkoituksena on optimoida yksittäisen rungon arvo. Optimointilaskenta

perustuu arvo- ja jakaumamatriiseihin, jotka ovat käytännössä kaksiulotteisia taulukoita. Kuvan 2 esimerkin mukaisessa arvomatriisissa määritellään tietyn tavaralajin eri pituus-läpimittayhdistelmien arvo suhteessa saman tavaralajin muihin mahdollisiin mittoihin sekä saman puulajin muihin tavaralajeihin. Jakaumamatriisi taas perustuu tuotantolaitoksen puun tarpeeseen ja siinä voidaan määritellä valmistettujen pölkkyjen jakauma eri pituus- ja läpimittaluokkien suhteen siten, että tuotantolaitos saa haluamansa mittaisia pölkkyjä tietyn osuuden kaikista valmistetuista pölkkyistä. (Uusitalo 2003, 155–159)

Pituus	Matriisit						
	370	400	430	460	490	520	550
150	1	1	260	200	1	230	210
160	1	1	250	200	1	220	210
170	1	1	250	200	1	220	210
180	1	1	250	200	1	220	210
190	1	1	250	200	1	220	210
200	1	1	250	200	1	220	210
220	1	1	250	200	1	220	210
250	1	1	250	200	1	220	210
260	1	1	250	200	1	220	210
280							
300							
350							
400							

KUVA 2. Esimerkki arvomatriisista.

Apteeraus voidaan suorittaa useammalla eri tavalla. Käytettäessä pelkkää arvoapteerausta, puun jakaminen tapahtuu ainoastaan arvomatriisin perusteella, eikä eri tavaralajien määrille ole erillistä tavoitetta. Pelkän jakaumamatriisin perusteella tehdessä kone taas jakaa puut siten, että eri mittojen jakauma vastaa täysin jakaumamatriisissa määriteltyjä tavoitteita. Useimmiten Suomessa käytetään näiden kahden yhdistelmää, jolloin kone pyrkii tavoitejakaumaan, mutta kuitenkin siten, että jakaumaan sallitaan

määritellyn toleranssin mukainen poikkeama, jotta puut saadaan jaettua mahdollisimman arvokkaiksi tavaralajeiksi. Käytettäessä automaattiapteerausta kuljettajan tehtävä on tarkkailla puutavaran laatua ja ohittaa automaatin tarjoamat mitat siinä tapauksessa että puun tekninen laatu ei täytä ostajan laatuvaatimuksia. (Uusitalo 2003, 155–159)

Koko apterauksen ja optimoinnin toiminta perustuu siihen, että puuta mitataan tarkasti sitä syötettäessä. Jakauma lasketaan ennustamalla rungon mitan kehittymistä syötön edetessä, eli koko runkoa ei mitata ennen katkontaa, vaan katkonta perustuu täysautomaattista apterausta käytettäessä ennusteeseen siitä, millainen vielä käsittelemätön loppuosa rungosta tulee olemaan. Kone tallentaa leimikon aiempien puiden mitoista muodostuneiden runkokäyrien tietoja jatkuvasti rengaspuskurin tavoin toimivaan runkopankkiin, joiden perusteella aina uutta puuta käsitellessä rungosta muotoa ennustetaan. Tämän ennustuksen perusteella ohjelma jakaa rungosta sitä käsitellessä eri tavaralajeihin ja muodostaa rungosta katkontaehdotuksen. Katkontaehdotus lasketaan uudestaan aina jos runkoa syötettäessä sen muoto muuttuu enemmän kuin asetetut toleranssit sallivat. Myös katkontaa arvioitaessa kuljettajan rooli on tärkeä, jotta kuljettaja pystyy arvioimaan katkontaehdotuksen oikeellisuutta ja ohittamaan esimerkiksi poikkeuksellisen muotoisten runkojen kohdalla automaattiapteerauksen tarjoaman katkontaehdotuksen. (Uusitalo 2003, 156–159)

2.4.4 Muut metsäkoneissa käytettävät ohjelmistot

Metsäkoneissa käytetään nykyisin varsinaisen mittalaitteohjelmiston lisäksi myös muita, lähinnä työnjohtoon ja korjuunohjaukseen liittyviä ohjelmistoja. Erityisesti pohjoismaissa korjuunohjaus, työmaatietojen lähetys ja vastaanotto sekä muut metsäkoneen päivittäiseen työhön liittyvät tiedot siirretään koneisiin täysin sähköisesti käyttäen metsäyhtiöiden omia korjuunohjausjärjestelmiä.

Edellä mainitut ohjelmistot ovat yleensä kolmansien osapuolien toimittamia ja urakanantajat asentavat ne yleensä metsäkoneen mittalaitetietokoneeseen.

Vaikka näitä ohjelmistoja kehittääkin kolmas osapuoli, annetaan ne testattavaksi metsäkonevalmistajille ennen julkaisua, jotta niiden toimivuus yhdessä metsäkoneohjelmiston kanssa voidaan varmistaa.

Huollon kannalta suurimman ongelman muodostavat ohjelmat, joita kuljettajat tai yrittäjät itse asentavat koneisiin ja joiden toimivuudesta ei näin voida olla varmoja. Koska tietokone itsessään on Windows-pohjainen sekä useimmiten nykyisin yhteydessä internetiin, on myös olemassa riski että kuljettajat käyttäessään internetiä tai sähköpostia saastuttavat koneen haittaohjelmilla, mikäli virussuojausta ei ole pidetty ajan tasalla.

3 MITTALAITTEIDEN HUOLTO

3.1 Mittalaitteiden huollon organisaatio

Ponssen mittalaittehuollon organisaatio jakautuu tällä hetkellä käytännössä kolmelle tasolle. Lähinnä käyttäjää oleva taso on maahantuojan paikallinen huolto-organisaatio, jossa joko maahantuoja itse tai maahantuojan jälleenmyyjät huolehtivat mittalaitteiden huollosta. Maahantuoja tai jälleenmyyjä on myös aina taho, johon asiakas ottaa yhteyttä ongelmatapauksissa.

Eri maahantuojien sekä jälleenmyyjien valmiudessa huolehtia mittalaittehuollosta on suuria eroja. Pitkäaikaisilla, kokeneilla ja ammattitaitoisilla jälleenmyyjillä on joillakin alueilla jopa lähes tehtaan mittalaittehuoltoa vastaavat valmiudet huolehtia mittalaitteiden huollosta. Toisaalta joidenkin maahantuojien ammattitaito riittää vain varalaitteen vaihtamiseen koneeseen vikatilanteessa, jonka jälkeen rikkoutunut laite joudutaan lähettämään tehtaan mittalaittehuoltoon korjattavaksi.

Toisella tasolla on tehtaan mittalaittehuolto, jonka tehtäviä hoidetaan tällä hetkellä pääasiassa Iisalmen huoltopalvelukeskuksessa sekä Kajaanin mittalaittehuollossa. Näissä molemmissa toimipisteissä sijaitsee lisäksi Suomen huoltopalveluiden mittalaittehuolto, joita suomessa on lisäksi 5 paikkakunnalla. Tehtaan mittalaittehuollot ovat toistaiseksi vastanneet myös kaikkien takuuajan rikkoutuneiden mittalaitteiden korjauksista. Takuukorjauksissa jatkossa jouduttaneen siirtymään siihen, että vähintäänkin omat tytäryhtiöt ympäri maailman tekevät myös takuukorjauksia. Takuukorjauksissa on kuitenkin tärkeää, että saadaan tietoa siitä, mitkä syyt takuukorjauksiin ovat johtaneet ja toisaalta se, että rikkoutuneet komponentit saadaan tuotekehityksen tutkittavaksi vähintäänkin uusien laitesukupolvien ensimmäisten rikkoontumisten osalta.

Tehtaan mittalaittehuolloissa keskitytään lähinnä komponenttien vaihtoihin sekä ohjelmistojen aiheuttamien toimintahäiriöiden ratkaisuihin. Komponenttien

korjaukseen ei tehtaankaan mittalaittehuoltojen kohdalla ole nähty tarvetta, koska komponenttien vaihtamisen kustannus asiakkaalle on useimmiten lopulta edullisempi, kun yksittäisen komponentin korjaukseen käytetyn työajan aiheuttama kustannus.

Korkeimmalla tasolla mittalaitteiden huollossa on valmistavan tehtaan, Epec Oy:n huolto, joka sijaitsee Seinäjoella. Epecin huollossa on mahdollisuus korjata myös komponentteja sekä tehdä yhteistyössä tuotekehityksen kanssa tarkempaa diagnostiikkaa rikkoontuneille komponenteille. Tämäntasoisien diagnostiikan hallitsevan henkilöstön rekrytointi sekä koulutus useampiin mittalaittehuoltoihin ei kuitenkaan ole tarkoituksenmukaista, koska komponenttien korjauksesta aiheutuu yleensä enemmän kuluja kuin uuden komponentin vaihdosta.

Lisäksi mittalaittehuollossa tärkeänä osana on varaosien toimitus sekä se, mitä varaosia mittalaitteisiin toimitetaan. Metsäkonealalla on useita eri toimintatapoja mittalaitteiden varaosatoimituksessa, lähtien siitä, että mittalaitte toimitetaan ainoastaan kokonaisena varaosana, päätyen siihen, että mittalaitteen komponentteja toimitetaan huollon käyttöön. Ponsella on päädytty siihen, että mittalaittehuoltojen käyttöön toimitetaan lähes kaikki helposti vaihdettavissa olevat komponentit, jotta asiakkaalle mittalaitteen rikkoontumisesta aiheutuvat kustannukset pysyvät kohtuullisena ja toisaalta korjaukseen kuluva aika on mahdollisimman pieni. Mittalaitteen varaosia ei kuitenkaan toimiteta suoraan asiakkaalle, eikä myöskään huolloille, jotka eivät ole saaneet koulutusta mittalaittehuoltoon.

3.2 Mittalaittehuoltojen työkalut ja testausvälineet

3.2.1 Mittalaittehuollot korjaamoilla

Tällä hetkellä mittalaittehuoltojen käytössä on melko vaihteleva työkaluvalikoima, joka koostuu huoltoihin rakennetuista testipöydistä, erilaisista testiohjelmistoista sekä vaihtelevasta varalaitte- sekä komponenttivalikoimasta. Ongelmana tällä hetkellä on se, että mittalaittehuolloillakaan työkalu-,

testiohjelmisto- tai varaosavalikoimaa ei ole vakioitu, jolloin ei myöskään ole mahdollista vakioida testien tuloksia ja tällöin tehtaan tukeen tulevien lisätietopyyntöjen taso on erittäin vaihteleva.

Korjaamoiden mittalaittehuollot tekevät mittalaitteiden vikakorjausten lisäksi päivityksiä koneenohjausohjelmistoihin sekä hieman markkina-alueesta riippuen myös metsäyhtiöiden ohjelmistojen asennuksia koneisiin. Lisäksi tärkeänä osana mittalaittehuoltojen toimintaa on yhteistyö muun huollon kanssa, jotta vikatilanteissa pystytään eristämään vian aiheuttaja mittalaitte-, sähkö- tai koneen mekaanisiin järjestelmiin. Tämän kaltaisen yhteistyön merkitys mittalaittehuollon ja muiden huoltopalveluiden välillä tulee jatkossa olemaan entistä korostetummassa asemassa, kun ohjausjärjestelmä edelleen kehittyy ja sen osuus koneen ohjauksessa ja toiminnassa entisestään kasvaa.

3.2.2 Mittalaitteen testaus kentällä

Mittalaitteen kenttätestauksen tilanne normaalin kentällä tehtävän huollon kohdalla on vähintäänkin yhtä kirjava kuin mittalaittehuolloissa. Mittalaitteiden kenttätestaus on suurelta osin kentällä liikkuvien huoltomiesten vastuulla. Huoltomiehet menevät kutsun saatuaan tutkimaan vikaantunutta metsäkonetta ja heidän on pystyttävä määrittelemään, johtuuko koneen vikaantuminen mittalaitteesta, koneen muusta sähköjärjestelmästä vai koneen mekaanisista järjestelmistä. Tällöin huoltomiehen henkilökohtainen osaaminen korostuu, koska koneiden luokse metsään menee harvoin useampi kuin yksi huoltomies kerralla. Käytännössä kenttätestaus oli työtä aloitettaessa huoltomiehen ammattitaidon sekä mittalaittehuoltojen puhelimen kautta annettavien neuvojen varassa, eikä varsinaisia testaustyökaluja ollut kentällä käytössä.

3.3 Mittalaittehuollon työkalujen kehittämistarpeet

Mittalaittehuollon työkalujen kehittämistarpeissa tärkeimpänä ja keskeisimpänä on tällä hetkellä sellaisen järjestelmän luominen, jolla voidaan varmistua kenttäolosuhteissa siitä, että mittalaitte sekä koneenohjausjärjestelmän eri komponentit ovat ainakin suurella todennäköisyydellä kunnossa tai toisaalta

vian aiheuttajina. Työkalun tulee olla sellainen, että kenttähuollossa työskentelevät huoltomiehet, jotka eivät käytännössä koskaan ole tietotekniikka-alan ammattilaisia, pystyvät tekemään testin ja päättelemään sen tuloksista riittäväällä varmuudella sen, onko tarvetta tuoda mittalaite tai jokin sen osa huoltoon tai vaihtaa jokin mittalaitteen tai koneen muun ohjausjärjestelmän kenttäolosuhteissa vaihdettavissa olevista komponenteista.

Tämän lisäksi työkalun kehittämisen tavoitteena tulee olla se, että kenttätestauksessa käytettävät menetelmät pystytään ainakin kohtuullisissa määrin vakioimaan, jolloin myös tehtaan mittalaitetukeen tulevien kysymysten taustatiedot ovat jo lähtötilanteessa paremmat. Nykyisen kaltainen kirjava tilanne testaustyökaluissa korostaa liikaa yksittäisen huoltoasentajan ammattitaidon merkitystä ja johtaa liian usein siihen että mittalaitteen tai koneenohjausjärjestelmän osia vaihdetaan varmuuden vuoksi ilman varmuutta siitä, että vaihdettava komponentti tosiasiallisesti on vian aiheuttaja.

Testaustyökalun kehittämisen yksi tavoite on myös varsinaisten ohjelmistovirheiden löytämisen helpottaminen, kun laitteista aiheuttavat viat on helpompi sulkea pois tilanteissa, jossa asiakas tai jälleenmyyjä raportoi tehtaalle koneen virheellisestä toiminnasta. Tällöin päästään nopeammin etsimään ja korjaamaan ohjelmistovirhettä ja toisaalta vähennetään aikaa jolloin joudutaan epäilemään sitä, onko jossakin koneessa tai koneissa esiintyvä virhe ohjelmiston vai laitteiston aiheuttama.

4 PONSSE-METSÄKONEEN OHJAUSJÄRJESTELMÄN RAKENNE

4.1 Opti6-tietokone

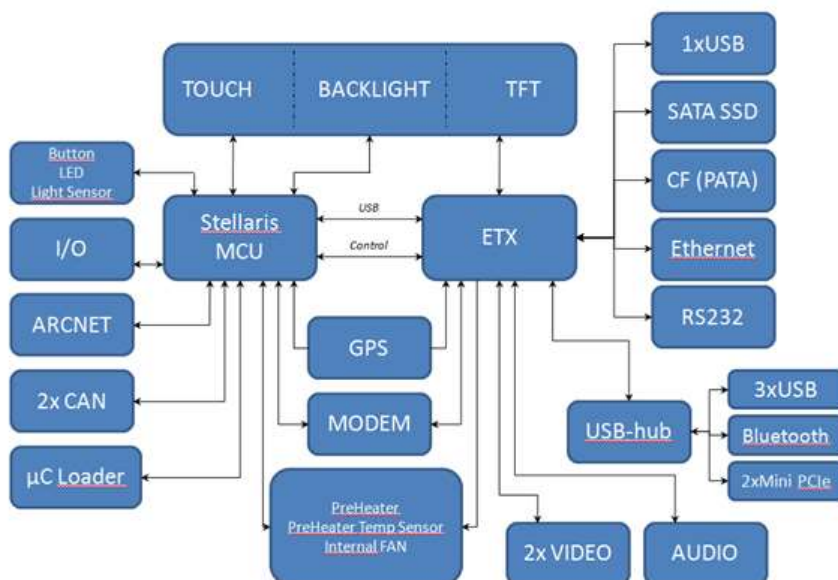
Ponsse Opti6 on erityisesti maastossa käytettäviin työkoneisiin suunniteltu tietokone, jonka suunnittelussa on otettu huomioon lämpötilan vaihtelu, värinä sekä käyttöympäristössä usein esiintyvät pöly ja epäpuhtaudet. Opti6 on ensimmäinen Ponssin koneissa käytettävä tietokone, jossa ei ole erillistä näyttöä, vaan koko kone on koottu yksien kuorien sisälle. Opti6:n suunnittelusta on vastannut Ponssin tytäryhtiö Epec Oy yhteistyössä Ponssin tietojärjestelmätuotekehityksen kanssa. Opti6:n tekniset perustiedot ovat taulukossa 1.

TAULUKKO 1. Opti6-tietokoneen perusominaisuudet

Proessori	Intel Core2 Duo
Työmuisti	2 Gt DDR2 SODIMM
Massamuisti	32 Gt SATA SSD
	2 Gt Flash-kortti
Näytönohjain	Intel Graphics Media Accelerator Integroitu näytönohjain
Näyttö	12,1” TFT, 1024 x 768 -resoluutio, kirkkaus 500 cd/m ² , kontrasti 700:1, automaattinen taustavalon säätö. Kosketusnäyttö.
Käyttöjännite	12 tai 24 V
Käyttölämpötila-alue	-40...+55 °C
Kotelointi	Alumiinikotelo, suojausluokka IP54

Opti6:n perusrakenteeseen kuuluu emolevy, joka on suunniteltu ja valmistetaan Epec Oy:n tuotannossa Seinäjoella. Emolevyyn liitetään erillinen Kontron ETX-CD-prosessorikortti, jolla sijaitsevat prosessori, RAM-muistikanta, northbridge-

piiri ja southbridge-piiri. Prosessorikortti perustuu Intelin 945GME Express -piirisarjaan. Prosessorikortin toimintoihin kuuluvat näytönohjaus, muistiväylän ohjaus, ulkoisten massamuistien ohjaus sekä Ethernet-portin, sarjaporttiliikenteen sekä USB-porttien ohjaus. (Kontron 2013, 7-13)



KUVA 3: Opti6-tietokoneen sisäinen rakenne

Emolevyllä sijaitsee jännitteenalennin, joka muuntaa laitteeseen syötettävän 12 tai 24 voltin jännitteen kolmelle eri jännitetasolle, jotka ovat 12, 5 tai 3,3 voltia. Kuvan 3 kaavion mukaisesti emolevyllä ovat myös GPS-piiri, micro-PCI-liitin esimerkiksi GPRS/3G-modeemia varten sekä muistikortin liitin. Lisäksi emolevyllä on Texas Instrumentsin valmistama Stellaris-mikrokontrolleri, joka perustuu ARM Cortex M3 -piiriin. Tällä kontrollerilla huolehditaan useista eri toiminnoista, jotka ovat mm. ne I/O-toiminnot, joita southbridge ei käsittele, eli esimerkiksi Arcnet- ja CAN-väyläliikenteet sekä GPS- ja modeemiliikenne tietyiltä osin. Lisäksi Stellarikseen on liitetty kosketusnäytön sekä näytön taustavalon ohjaus. Yksi maasto-olosuhteissa erittäin tärkeä ominaisuus on tietokoneen esilämmitys, joka toimii siten, että kytkettäessä metsäkoneen polttoainetoiminen esilämmitin päälle, annetaan myös Opti6:n Stellarikselle tieto siitä, että konetta ollaan esilämmittämässä käyttöä varten. Tällöin myös Opti6

lämmittää lämmitysvastuksilla näyttöä, massamuistia sekä prosessorikorttia, jotta mittalaitetietokoneen käynnistäminen on turvallista, kun varsinainen metsäkone käynnistetään.

Opti6:n jäädytys perustuu passiiviseen jäädytykseen. Kotelo on valmistettu alumiinista, ja sen takaosassa olevat jäädytysrivit johtavat lämpöä ohjaamon ilmaan. Prosessori, northbridge, southbridge, muistipiiri, massamuisti sekä jännitteenalentimen kondensaattorit yhdistetään lämmönjohtotahnalla suoraan alumiinikuoreen, jotta lämmönsiirto olisi mahdollisimman tehokasta. Lisäksi kotelon sisällä on pieni puhallin, joka kierrättää ilmaa kotelon sisällä.

Näyttönä Opti6:ssa on Chi Mei Optoelectronicin valmistama TFT-LCD-näyttö, jonka taustavalaistus perustuu LED-tekniikkaan. Näytön koko on 12,1 tuumaa ja resoluutio 1024 x 768, mikä on riittävä nykyisten metsäkoneohjelmistojen käyttöön. Metsäkonekäytössä tärkeät ominaisuudet ovat heijastamaton pinta sekä automaattinen kirkkauden säätö, jolloin kuljettajan ei tarvitse säätää kirkkautta siirryttäessä auringosta varjoon tai päivänvalosta yötyöskentelyyn. Näytölle sallittu käyttölämpötila on $-40 \dots +80$ °C. (Chi Mei Optoelectronics 2009, 4–6.)

Päämassamuistina koneessa on 32 gigatavun SSD-levy, jolle on asennettu käyttöjärjestelmä sekä koneenohjaus- sekä muut tarvittavat ohjelmistot. Lisäksi mittalain asettamien vaatimusten mukaisesti mittaustiedot on tallennettava kahdenkertaisesti, jotta mahdollisessa vikatilanteessa metsänomistajan mittaustiedot eivät pääse katoamaan. Tätä tarkoitusta varten Opti6:ssa on paikka muistikortille, jona tällä hetkellä käytetään 2 gigatavun CF-korttia.

Käyttöjärjestelmänä on Ponssen käytössä Microsoft Windows 7 Embedded, jota on käytetty vuoden 2011 syksystä alkaen. Ponsse on ensimmäinen metsäkonevalmistaja, joka siirtyi koneissaan käyttämään normaalia PC-tietokonetta Windows-käyttöjärjestelmällä. Tällä hetkellä PC-koneiden käyttö on yleisin tapa toteuttaa metsäkoneen mittalaite riippumatta valmistajasta, ja

Windows on yleisin käytetty käyttöjärjestelmä, vaikkakin jotkin valmistajat ovat siirtyneet viimeisten vuosien aikana myös Linux-sovelluksiin.

Mittalaitetietokone voi käyttää syöttöjännitteenä joko 12:ta tai 24:ää voltia. Metsäkoneen oma sähköjärjestelmä perustuu 24 voltin jännitteeseen, mutta lisäksi mittalaitteelle on rakennettu oma 12 voltin akkuvarmistusjärjestelmä, millä varmistetaan se, että mittalaitetietokone voidaan sammuttaa hallitusti myös tilanteessa, että metsäkoneen omaan sähköjärjestelmään tulee sen toiminnan täydellisesti estävä vika.

4.2 Koneenohjaus ja metsäkoneen tietoverkko

Mittalaitte toimii metsäkoneissa myös osana koneenohjausjärjestelmää. Koneenohjausjärjestelmässä on useita eri ohjainmoduuleita, joilla kullakin on oma tehtävänsä. Useiden moduuleiden käyttö on tarpeen sekä vianhallinnan, suorituskyvyn, että kustannuskysymysten kannalta.

Hakkuukoneessa on käytössä eri malleissa mittalaitteen lisäksi 5–8 erillistä moduulia, jotka ovat moottorinohjain, nosturimoduuli, penkkimoduuli, vaihteistomoduuli, aktiivivaimennusmoduuli, vaimennuksen anturimoduulit sekä harvesteripäämoduuli. Ajokoneissa näistä moduuleista jäävät pois harvesteripäämoduuli, aktiivivaimennusmoduuli sekä anturimoduulit.

Vianhallinnan kannalta usean moduulin käyttö helpottaa toimintaa vikatilanteessa, koska todennäköisyys sille, että yhden moduulin vikaantuminen estää koneen ajamisen pois metsästä korjausta varten, pienenee oleellisesti. Korjaukset ovat myös huomattavasti edullisempia, koska käytettävät moduulit voivat olla rakenteeltaan yksinkertaisempia ja niiden suorituskykyvaatimus ei ole niin suuri.

Käytännössä myös moduulien suorituskyky on erityisesti aiemmin aiheuttanut sen, että on ollut tarpeen käyttää useampia eri ohjaimia ohjaamaan koneen eri

toimintoja. Esimerkiksi harvesteripäässä puu saattaa liikkua syötettäessä 7 m/s, jolloin mitattaessa läpimittaa yhden sentin välein on läpimittahavaintoja ja siitä aiheutuvia toimintoja pystyttävä laskemaan ja ohjaamaan jopa satoja kertoja sekunnissa. Tämän vuoksi myöskään reaaliaikataskien pituus ei voi olla muutamaa millisekuntia pidempi, minkä vuoksi harvesteripäälle käytetään edelleen omaa ohjausmoduulia. Sama periaate pätee metsäkoneissa useisiin muihinkin toimintoihin, joiden on pystyttävä reagoimaan puun mittojen tai muotojen, maaston muotojen tai muiden olosuhteiden muuttuessa erittäin nopeasti.

Mittalaitteen tehtävä koneenohjauksessa on toimia käyttöliittymänä kuljettajan ja koneen välillä sekä tallentaa harvesteripäämoduulilta tulevat mittatiedot tietokantaan. Mittalaitteen lisäksi koneen asetukset tallennetaan nosturimoduulille, mikä mittalaitetietokoneen vikaantuessa mahdollistaa koneen liikuttamisen metsästä pois korjausta varten tai koneen siirtämisen lavetille ilman, että mittalaite on käynnistetty.

Mittalaitteella on myös mahdollista asettaa koneen käyttöliittymän asetukset siten, että kullakin kuljettajalla voi olla oma näppäinjärjestyksensä koneen hallintakahvoissa, sekä omat automatiikan, voimansiirron sekä nosturin nopeus- ja käyttäytymissäädöt. Mittalaitteohjelma sisältää myös erittäin monipuolisen ja laajan diagnostiikkaohjelmiston koneen toimintaan, minkä kautta sekä kuljettaja että erityisesti koulutettu huoltomies pystyy tekemään varsin perusteellista vikadiagnostiikkaa koneen eri toiminnoista. Lisäksi huoltomiehellä on järjestelmässä oma käyttäjätasonsa, joka mahdollistaa laajempien koneen säätöjen ja asetusten muokkaamisen.

Koneenohjausjärjestelmässä moduulien väliseen liikennöintiin käytetään tällä hetkellä Arcnet-verkkoa, lukuun ottamatta moottorinohjainta, joka on yhdistetty muuhun verkkoon CAN-väylän avulla. Lisäksi käytettäessä kuormainvaakaa tai tietyissä hakkuukoneissa käytössä olevia anturimoduuleita, ovat nämä liitettynä koneenohjaukseen CAN-väylän avulla.

Ohjainmoduulien lisäksi verkossa on keskitin eli hub, jonka avulla verkon topografia pystytään toteuttamaan tähtiverkkona. Aiemmin käytössä ollut rengasverkko on helpompi toteuttaa ilman hubia, mutta käytännön kokemukset vikasietoisuudesta ovat osoittaneet, että käyttämällä aktiivista hubia, joka sulkee vikaantuneen ja tämän vuoksi virheellisiä viestejä lähettävän moduulin pois verkosta, voidaan varmistua siitä että yhden moduulin tai yhden moduulin väyläkaapelin vikaantuminen ei aiheuta koko koneen täydellistä liikuntakyvyttömyyttä.

5 HUOLTOMIEHEN KENTTÄTESTAUSSALKUN VAATIMUSMÄÄRITTELY

5.1 Kenttätestaussalkun tarve

Ponssen huoltopalveluissa on kasvava tarve standardoida työkalu, jota mittalaittehuolto käyttää testaukseen vikatilanteissa. Tällä hetkellä eri mittalaittehuolloilla jopa saman markkina-alueen sisällä on käytössään hyvin eritasoisia sekä muutenkin erilaisia työkaluja ja testiohjelmistoja. Tämä aiheuttaa ongelmia erityisesti niissä tilanteissa, kun mittalaittehuolto joutuu ottamaan yhteyttä tehtaan tukipalveluihin.

Otettaessa yhteyttä tehtaan mittalaitetukeen olisi tärkeää, että saatavat taustatiedot sekä testaustulokset olisivat vakioituja siten, että lähtötilanne alettaessa käsittelemään tukipyyntöä olisi sama. Tämän hetkisessä tilanteessa ei voida varmistua siitä, mitä toimenpiteitä ja testauksia mittalaitteelle on ennen tukipyyntöä tehty. Tämä johtaa siihen että tukipyynnön saapuessa on useimmiten aloitettava testaukset lähes alusta, mikä lisää merkittävästi ratkaisun löytymiseen kuluvaa aikaa. Koska koneiden seisokkiajan kustannus on asiakkaille hyvin merkittävä, on tätä toimintoketjua pyrittävä nopeuttamaan.

Käytännössä yllä mainitun viiveen lyhentämiseen on kaksi mahdollisuutta: joko maahantuojalla on riittävä määrä varalaitteita varastossaan, jotta mittalaitteen rikkoontumistapauksissa voidaan koneeseen asentaa varalaitte korjausten ajaksi, tai testausvaihetta pyritään nopeuttamaan ja yksinkertaistamaan siten, että mittalaitteen korjaus tai sen lähetys korjattavaksi tehtaan huoltoon voidaan tehdä hyvin nopealla aikataululla.

Taloudellisesti ei ole mielekästä eikä aina mahdollistakaan pitää maahantuojien varastoissa useita varalaitteita. Tämän vuoksi testausmenetelmien kehittäminen ja standardointi on tärkeää. Mittalaitteen testaussalkun avulla voidaan vakioida sekä testausympäristö, missä testaukset tehdään, että testauksesta saatavat tulokset. Tällöin myös tehtaan tuen on huomattavasti helpompi antaa ohjeet

korjauksista myös niissä tapauksissa, joissa paikallinen mittalaitahuolto ei pysty tuloksia itse tulkitsemaan.

5.2 Kenttätestaussalkulle asetettavat vaatimukset

5.2.1 Eri testaustarpeet

Mittalaitteen ja koneenohjausjärjestelmän toimivuuden testauksessa kenttätestaussalkun kehitystyössä keskitytään väylän ja moduulien sähköiseen sekä ohjelmistojen testaukseen. Mittalaitteen kohdalla testaussalkussa keskitytään laitetestaukseen, jolloin testattaessa pystytään varmistamaan itse mittalaitetietokoneen toimivuus ja siten eristämään vika joko koneenohjausjärjestelmän puolelle tai ohjelmistoihin.

Varsinaisen ohjelmistotestauksen tekeminen on käytännössä lähes mahdotonta, koska mittalaitteohjelmistoa käytettäessä sekä mittauksessa että koneenohjauksessa on eri variaatioita siitä, miten koneen tulee käyttäytyä, erittäin paljon. Ohjelmistovikojen testauksessa tärkeimpänä toimintatapana säilyykin edelleen kuljettajien tekemien havaintojen testaaminen huollossa ja tuotekehityksessä joko koneella tai simulaattorilla. Samoin mittalaitetietokoneeseen asennettavien kolmansien osapuolien ohjelmistojen yhteensopivuuden testaus on pyrittävä mahdollisuuksien mukaan tekemään etukäteen, vaikkakin on selvää että kaikkia mahdollisia ohjelmistokokonaisuuksia ei pystytä käytännössä testaamaan.

Koneenohjausjärjestelmän testauksessa testaussalkun avulla on pystyttävä varmistamaan siitä, että testattavana olevan metsäkoneen fyysisen verkon sekä ohjausmoduulien väyläliikenne toimii. Tällöin vika pystytään eristämään joko metsäkoneen verkon fyysiseen osaan, eli johtosarjoihin tai liittimiin, moduuleihin, moduulien ohjelmistoihin tai moduulien ulkopuolisiin toimintoihin. Moduulien muiden I/O-toimintojen testaukseen on metsäkoneen ohjelmistossa itsessään olemassa kohtuullisen hyvät työkalut ja toisaalta täydellisen moduuli-I/O-testin tekeminen ei olisi tässä projektissa ajan ja resurssien puolesta mahdollista.

Lisäksi huoltotoiminnan kannalta olisi tärkeää, että testaussalkulla pystyttäisiin tarvittaessa ohjelmoimaan eri ohjainlaitteita, jolloin korjaamalla voitaisiin tehdä varaosaohjaimesta ohjelmoimalla valmis moduuli jopa suoraan asiakkaalle ilman että käytössä olisi valmista metsäkonetta. Pelkän valmiiksi ohjelmoidun moduulin vaihto on yksinkertainen toimenpide, jonka jopa asiakas pystyy tarvittaessa tekemään, joten tämä helpottaisi monissa tapauksissa varaosahuoltoa sekä vähentäisi turhia huoltokäyntejä, jotka sekä sitovat huollon resursseja että aiheuttavat asiakkaille turhia kustannuksia.

Testauksessa on tarvittaessa tekemään pikatesti, joka kertoo järjestelmän toiminnan nopeasti. Lisäksi tapauksissa, jossa epäillään laitteiden lämpenemisen vaikuttavan niiden toimintaan, on pystyttävä ajamaan rasiustestiä, joka kuormittaa laitteita riittävän pitkän ajan niiden lämpiämisen kannalta. Tällöin salkkuun on myös pystyttävä luomaan ympäristö, jossa laitteiden ulkopuolisen ilman lämpötila pystytään nostamaan ja pitämään riittävän korkealla, jolloin lämpiämisen aiheuttamat viat tulevat esille.

5.2.2 Kenttäkelpoisuus

Kenttätestaussalkun on nimensä mukaisesti oltava rakennettu niin, että se voidaan ottaa huoltoautoon ja viedä metsään koneen luokse. Tällöin testaussalkun on oltava riittävän kestävä, jotta se kestää fyysiset tärähdykset, pienet putoamiset sekä kantamisen metsään koneen luokse. Lisäksi salkun on oltava roiske- sekä sadeveden kestävä.

Salkun on oltava riittävän kevyt, jotta se pystytään tarvittaessa kantamaan koneen luokse maastoon. Tällöin rajoituksia asettavat salkussa käytettävien eri komponenttien painot, koska moduulien ja mittalaitetietokoneen keventäminen ei ole mahdollista.

Fyysisen kenttäkelpoisuuden lisäksi tärkeä osa kenttäkelpoisuutta on se, että salkku ja sen käyttö on tehty riittävän yksinkertaiseksi, jotta huoltomies, jolle ei ole annettu mittalaitesuunnittelun erikoiskoulutusta, pystyy tekemään

perustestauksen metsäkoneen ohjausjärjestelmälle ja siirtämään mittaustulokset mittalaitahuollolle arviointia varten tai selvissä tapauksissa päättämään diagnostiikan tulosten perusteella, onko tarvetta tuoda mittalaite huollettavaksi. Kentällä ei sinänsä tehdä mittalaitteen sisäisiä korjauksia tai komponenttien vaihtoja, vaan ne tulee suorittaa varsinaisten mittalaitehuoltojen ESD-suojatuissa työpisteissä.

5.2.3 Taloudelliset seikat

Testaussalkun hinta on oltava riittävän edullinen, jotta yksittäinen maahantuoja tai isompi jälleenmyyjä pystyy ostamaan salkun omaan käyttöönsä. Salkun rakenne ei saa olla liian monimutkainen eikä toisaalta vaatia liikaa erikoiskomponentteja, jotta sen hinta ei nouse liian suureksi.

Salkussa olevien moduulien sekä mittalaitteen on oltava suoraan käytettävissä varaosiksi tai vaihtolaitteiksi tarvittaessa, mikäli testausta tehdessä havaitaan että jokin yksittäinen osa mittalaitteesta tai koneenohjausjärjestelmästä vaatii huoltoa.

Mikäli maahantuojalla tai jälleenmyyjällä kuitenkin on jo valmiisi varastossaan varaosakomponentteja, tulee niitä voida käyttää testaussalkkuun, jolloin vältetään ylimääräisiltä kustannuksilta.

Hankkiessaan testaussalkun maahantuoja tai jälleenmyyjä saa toisaalta käyttöönsä varaosaohjaimet, jolloin niitä ei erikseen tarvitse ostaa varaosavarastoon. Testaussalkkua erikoistyökaluksi hinnoiteltaessa on otettava huomioon tämä seikka, jotta testaussalkun hinta on kohtuullinen verrattuna varaosina erikseen ostettavien komponenttien hintaan.

5.2.4 Testaustulosten vertailtavuus

Testaustulosten vertailtavuuden kannalta tärkeitä asioita on kaksi: testausympäristö on vakioitu ja testaukseen käytettävät työkalut on vakioitu. Vakioituun testausympäristöön päästään käyttämällä salkkua, koska siinä

käytettävät komponentit ovat aina samat. Toisaalta, mikäli esimerkiksi yksi koneenohjausjärjestelmän komponenteista ei toimi, voidaan se vaihtaa ja varmistaa että muu osa järjestelmästä toimii uuden vaihdetun komponentin kanssa.

Testaukseen käytettävien työkalujen vakioinnilla pystytään varmistamaan se, että työkalujen antamat tulokset ovat samassa muodossa ja helpommin tulkittavissa jokaisen eri testauskerran välillä. Markkinoilla on saatavilla useita valmiita laitetestausohjelmistoja, joten kokonaan uuden testausohjelman tekeminen ja käyttäminen ei ole järkevää vaan kannattaa käyttää jo valmiiksi tehtyjä ohjelmistoja.

Testaustulokset pitää pystyä tallentamaan testausohjelmasta muotoon, jossa ne on helppo siirtää tarvittaessa teknisen tuen kautta tehtaan mittalaitahuollon käyttöön. Tämä voi tulla kyseeseen sekä ongelmatapauksissa, joihin ei paikallisesti löydy ratkaisua, että esimerkiksi takuukampanjan tai tuotekehityksen esittämän pyynnön yhteydessä. Lisäksi testituloksista tulisi ilmetä mahdollisuuksien mukaan testattavan laitteen laitteistokonfiguraatio, jolloin olisi mahdollista myös vähintään kohtuullisesti seurata sitä, että jälleenmyyjien mittalaitahuolloissaan käyttämät osat ovat Ponssen alkuperäisvaraosia.

6 TOTEUTUS

6.1. Testaussalkun mekaaninen rakenne

Testaussalkun (kuva 4) mekaaniselle rakenteelle oli alussa asetettu seuraavat vaatimukset: salkun on oltava asentajan siirrettävissä metsäolosuhteissa koneen luokse, sen on kestävä tärinää sekä oltava pöly- ja roiskevesitiivis. Eri salkkuvaihtoehdoista päädyttiin sukellustarvikelaukkuun, joka modifioitaisiin sopivaksi testaussalkkukäyttöön. Sukellustarvikelaukun käytöstä oli aiempien siirrettävien mittausjärjestelmäsimulaattorien kautta hyviä kokemuksia, ja Ponsella oli valmis tavarantoimittaja tämän kaltaisille salkuille.



KUVA 4. Testaussalkku

Salkun sisälle tehtiin kiinnityspisteet virtalähteelle, josta tuodaan läpivienti salkun ulkopuolelle, johon edelleen liitetään virtajohto verkkovirtapistokkeeseen tai metsäkoneen apuvirtapistokkeeseen. Virtalähteeksi käy lähes mikä tahansa virtalähde. Ainoana perusvaatimuksena on se, että verkkovirran osalta virtalähteen on toimittava sekä 110 että 230 voltin sähköverkoissa.

Virtalähteeksi valittiin Yleiselektronikka Oy:n valikoimista malli Lambda LS150-24. Virtalähteen malli voi vaihtua toimittajan valikoimien mukaan, kunhan sen perusvaatimukset täyttyvät.

Lisäksi salkkuun rakennettiin teline ohjainmoduulille sekä varsinaiselle mittalaitetietokoneelle. Telineen tulee mahdollistaa moduulien sekä mittalaitteen kohtuullisen helpon ja nopean irrotuksen testaustilanteessa, mutta joka toisaalta kuitenkin takaa laitteiden pysymisen riittävän hyvin paikallaan kuljetuksen aikana. Lisäksi moduulit sekä mittalaitetietokone tulee saada riittävän hyvin kiinni, jotta kuljetuksen aikaiset tärähdykset eivät riko laitteita.



KUVA 5. Testaussalkku sisältä

Salkun johdinsarjan pituus suunniteltiin pituudeltaan sellaiseksi että salkun laitteita ei välttämättä tarvitse irrottaa, vaikka kenttätestiä tehdessä salkussa olevan moduulin tilalle haluttaisiin liittää koneessa kiinni oleva moduuli.

Lämpötilarasiustestsiä varten salkku toteutettiin niin, että se voidaan testauksen ajaksi sulkea ja jättää laitteet käyntiin salkun sisälle. Tällöin, kun mittalaitteessa ajetaan sitä voimakkaasti kuormittavaa testisykliä, sen oma lämmöntuotanto riittää lämmittämään salkun sisäilmaa niin että lämpötilan vaikutukset laitteiden toimintaan voivat selvitä. Salkkuun rakennettiin laitteiden jännitteensyötön sulkeva järjestelmä, joka sammuttaa laitteet noin 60 asteen lämpötilassa ja vastaavasti sallii laitteiden käytön lämpötilan taas alittaessa 50 astetta. Tällöin salkun lämpötila voidaan pitää noin 8 tuntia kestävän rasiustestisyklin aikana riittävän korkeana, jotta mittalaitteen osiin kohdistuu vastaava lämpötila kuin metsäkoneessa kuumalla säällä työskennellessä.

Lämpötila-anturia vaihtamalla voidaan rasiustestauksen lämpötila-aluetta tulevaisuudessa muuttaa, mikäli sille nähdään tarvetta. Lisäksi jatkokehitysmahdollisuutena on mahdollista harkita järjestelmää, jossa salkkua tuuletettaisiin esimerkiksi tietokoneen kotelotuulettimella lämpötila-anturin ohjaamana, jotta rasiustesti ei katkea välittömästi kun lämpötila nousee 60 asteeseen.

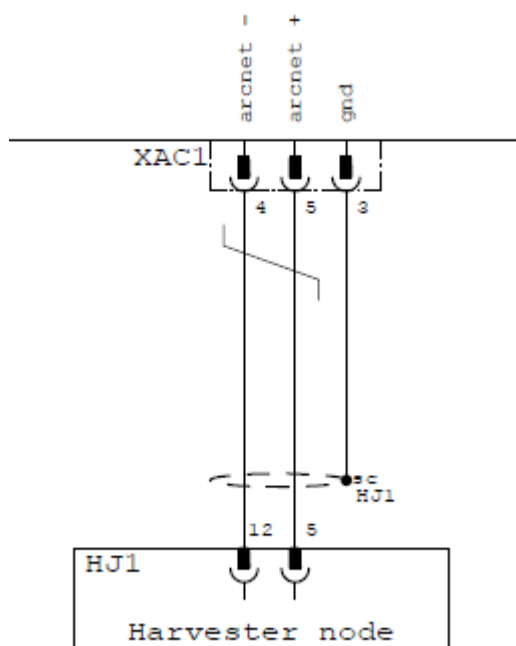
Testaussalkkuja varten laadittiin osalista, jotta salkkujen kokoamista varten on helppo tilata oikea määrä osia. Salkkujen varsinainen kokoamistyö teetetään alihankintana Ylä-Savon ammattiopistolla, jonka kanssa Ponsse on aiemminkin toteuttanut vastaavan kaltaisia projekteja. Salkun täydellinen osalista on liitteessä 1.

6.2. Testaussalkun kytkentäkaavio sekä sähköinen toteutus

Testaussalkun kytkennässä tarkoituksena oli rakentaa metsäkoneen ohjausjärjestelmää vastaava verkko, kuitenkin niin että verkon toiminta rajoittui ainoastaan moduulien verkkoyhteyksien testaukseen, ja moduulikohtaisten I/O-toimintojen testausta ei tässä projektissa lähdetty toteuttamaan.

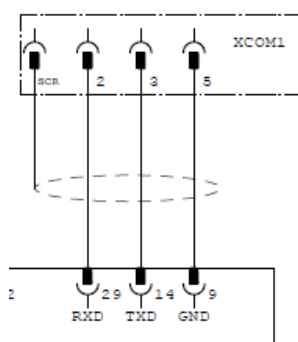
Verkkoa rakentaessa päädyttiin siihen, että kaapelien pituutta ei tarvitse toistaa täsmällisesti verrattuna metsäkoneen kaapelointiin. Kaapelin ominaisvastus on niin pieni, että sillä ei ole merkitystä Arcnet-verkon toiminnan kannalta, kun kaapelin pituuden muutos on käytännössä enintään 20 metriä. Kaapeli sekä verkon laitteet on myös impedanssisovitettu.

Väyläkaapelina käytettiin samaa kaapelia, jota käytetään myös metsäkoneen Arcnet-verkon kaapeloinnissa. Kaapeli on metalliverkkovaipalla häiriösuojattua kierrettyä parikaapelia LEONI L45551-C21-C8. Kaapelointi toteutettiin siten, että kaapelien metalliverkkovaipat yhdistettiin hubin kautta toisiinsa ja niitä ei yhdistetty verkon muihin laitteisiin. Kuvassa 6 esimerkkikytkentä hubista harvesteripäämoduulille.



KUVA 6: Harvesteripäämoduulin kytkentä hubiin

Lisäksi mittalaitetietokoneen XPC2-liittimeltä kaapeloidaan sarjaporttiliitäntä liittimeen XCOM1. Tällöin mittalaitetietokoneeseen pystytään myös liittämään metsäkoneessa olevat mittasakset, joilla työskennellessä suoritetaan mittalaitteen kalibroitimittaukset. Tämän ansiosta salkun avulla on tarvittaessa mahdollista suorittaa myös mittasaksien tiedonsiirron toiminnan testaus sekä saksien ohjelmointi. Mittasaksien kytkentä on esitetty kuvassa 7.



KUVA 7. Mittasaksien sarjaportin kytkentä.

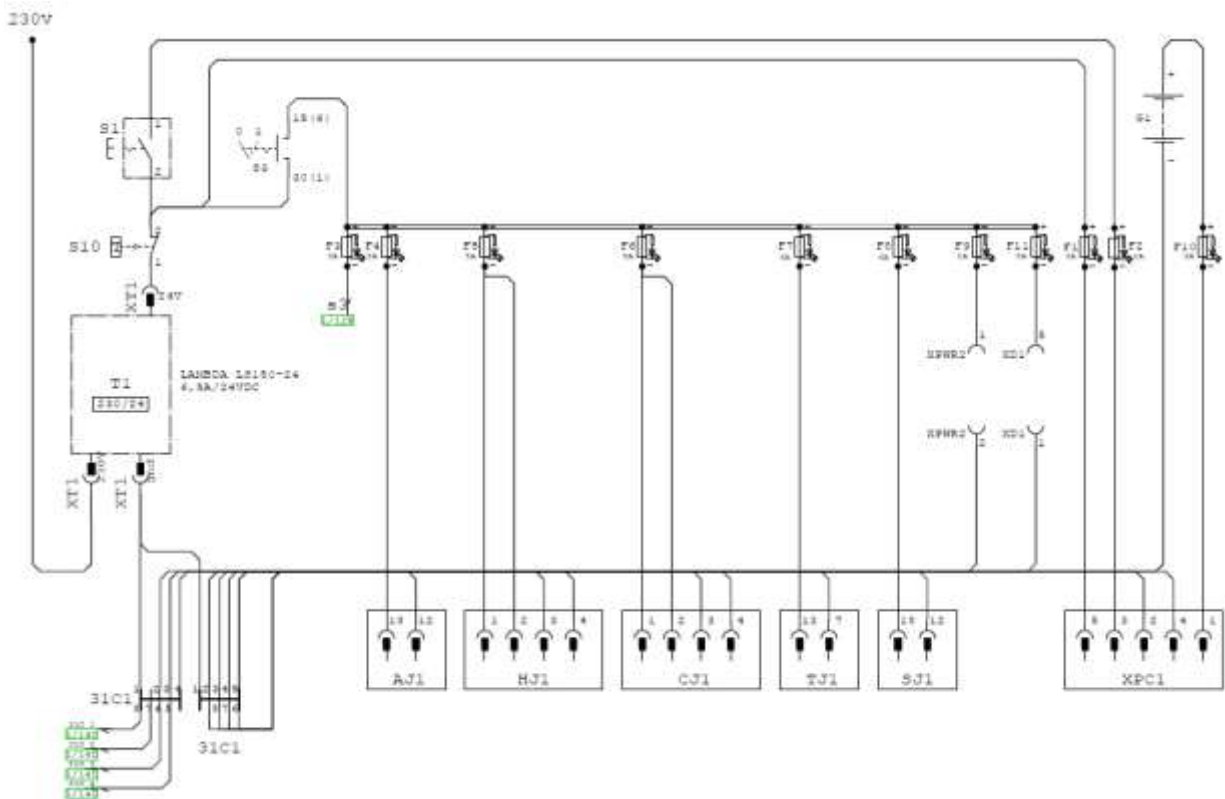
Moduulien virransyöttö toteutettiin virtalähteen kautta, josta jännite syötetään ensimmäisenä termostaatille S10. Tämän termostaatin tarkoituksena on katkaista laitteiden syöttöjännite, mikäli laitteiden ympäristön lämpötila nousee liian korkeaksi. Tällä varmistetaan, ettei itse salkun käytöllä pystytä vaurioittamaan laitteita esimerkiksi useita tunteja kestävästä raskasta rasituksesta aikana. Tämän jälkeen virtapiiri jaetaan kolmeen erilliseen piiriin.

Virtalähde on kytketty mittalaitetietokoneen virransyöttöön siten, että virtalähteen päälle kytkentä vastaa metsäkoneen päävirtakytkimen päälle asettamista. Kytkimellä S1 kytketään ns. lämmitysvirta. Kytkimen käytöllä voidaan simuloida koneen esilämmittimen antamaa lämmityssignaalia, jonka toiminta ja olemassaolo voidaan testata ohjelmallisesti. Kytkimellä S3 kytketään ns. virtalukkojännite, joka metsäkoneessa kytkee virrat myös ohjainmoduuleille.

Lisäksi virtapiiriin on rakennettu metsäkoneen ohjaamoakkaa vastaava akku G1, joka toimii vastaavasti kuin metsäkoneen 12 V:n piiri. Akkujännitteen

toimivuus voidaan testata katkaisemalla verkkovirtalähteeltä tuleva syöttö kokonaisuudessaan, jolloin mittalaitetietokoneen tulee edelleen pysyä käynnissä. Mittalaitetietokone on rakennettu siten, että 24 V:n jännitteellä toimiessaan se lataa 12 V:n piirissä olevaa akkua, joten akku ei tarvitse erillistä latausta. Kaikki syöttöjännitteet on suojattu erillisillä sulakkeilla, joista on luettelo taulukossa 2. Maadoituspisteet koottiin yhteen maadoituspisteeseen 31C1, mikä vastaa tilannetta metsäkoneessa jossa maadoitukset on kaikki koottu koneen rungossa oleviin maadoituspisteisiin.

Lisäksi virtapiiriin on valmiiksi tehty varaus anturimoduulien virransyötölle. Anturimoduulit itsessään käyttävät tiedonsiirtoon CAN-väylää ja ne tullaan jatkossa kytkemään aktiivivaimennusmoduuliin CAN-väylällä. Johtosarjassa ei kuitenkaan tässä vaiheessa huomioitu anturimoduuleita muutoin, kuin rakentamalla niiden syöttöjännitteelle varaus. Itse verkon johtosarjamuutos tulee olemaan pieni, ja helppo toteuttaa myös jälkikäteen.



KUVA 8: Jännitteensyötön kytkentä

TAULUKKO 2: Testaussalkun sulakkeet

Sulake	Koko	Toiminto
F1	3A	Mittalaitetietokone 24 V
F2	1A	Mittalaitetietokone esilämmitys
F3	3A	Hub
F4	5A	Aktiivivaimennusmoduuli
F5	5A	Harvesteripäämoduuli
F6	5A	Nosturimoduuli
F7	5A	Ajovoimansiirtomoduuli
F8	5A	Penkkimoduuli
F9	1A	Mittasakset
F10	3A	Mittalaitetietokone 12 V
F11	5A	Varaus anturimoduulit

Johdinsarjassa käytettiin myös liittiminä metsäkoneen johtosarjoissa käytettäviä osia, joista osaluettelo liitteessä 2. KytKentäkaavio piirrettiin ensin prototyypisalkun johdinsarjaa varten käsin, jonka jälkeen johdinsarja valmistettiin yksittäiskappaleena Ponssen huoltopalveluissa lisämessä. Prototyypivaiheen jälkeen kytKentäkaaviota täydennettiin sekä johdinsarjan mitoitus tarkennettiin ja kytKentäkaavio piirrettiin Ponssen tuotekehityksessä sähkösuunnittelussa käytettävällä E3-designer -ohjelmistolla puhtaaksi. Samalla johdinsarjasta piirrettiin myös johdinsarjatuotantoa varten johdinsarjapiirustus, jonka perusteella varsinainen sarjavalmistus tapahtuu. Testaussalkun täydellinen kytKentäkaavio sekä johdinsarjakuva ovat liitteissä 2 ja 3.

Johdinsarjojen varsinainen tuotanto myytäviä testaussalkkuja varten tullaan teettämään joko Ponssen oman tehtaan johdinsarjavalmistuksessa tai muitakin johdinsarjoja alihankintana toimittavalla SKS Connecto Oy:llä. Sarjatuotantosalkkujen mekaaninen kokoaminen tullaan myös teettämään yhteistyökumppanilla.

6.3. Testaussalkun ohjelmistot ja niiden käyttö sekä tulosten analysointi

6.3.1. Testaukseen käytettävät ohjelmistot

Testausohjelmistoa valittaessa oli ensimmäisenä määritettävä ohjelmistolle asetettavat perusvaatimukset. Ensimmäisenä vaatimuksena oli se, että ohjelmisto oli tarvittaessa pystyttävä suorittamaan itsenäisesti käynnistyvältä USB-tikulta, jotta mahdollisesti rikkoontunut massamuisti ei estäisi testiohjelman suorittamista. Toisena vaatimuksena oli se, että testi oli pystyttävä määrittelemään etukäteen siten, että testiä suorittavan asentajan, jolla ei välttämättä ole koulutusta mittalaitahuoltoon, ei tarvitse valita suoritettavia testejä, vaan prosessi on täysin automatisoitu. Kolmantena vaatimuksena on se, että ohjelmistosta on saatava raportti, joka voidaan tarvittaessa helposti lähettää sähköpostilla eteenpäin tehtaan tekniseen tukeen tai tuotekehitykseen.

Testaukseen käytettäviä ohjelmistoja valittaessa selvitettiin myös, mitä ohjelmistoja Ponssen tuotekehitys sekä tuotanto käyttävät. Yrityksen omista tuotekehitys- sekä tuotantotoiminnoissa Kajaanissa ja Seinäjoella oli varsin laajassa käytössä Eurosoftin QA+WIN32-testiohjelmisto, joka on suunniteltu nimenomaisesti PC-tietokoneiden laitetestaukseen.

Tämän jälkeen selvitettiin, onko kyseinen ohjelmisto sopiva myös huoltotoimintojen käyttöön ja onko olemassa muita vastaavia ohjelmistoja, joiden käyttöä voitaisiin harkita. Muina vastaavina vaihtoehtoina tukittiin kahta ohjelmistoa, jotka olivat Ultra-X Inc:n QuickCAST sekä PC-Doctor Inc:n Pc-Doctor. Selvittäessä näiden tuotteiden hintoja havaittiin, että eri ohjelmien lisenssien välillä ei ole käytännössä sellaista hintaeroa, joka vaikuttaisi merkittävästi koko salkun hintaan. Tämän vuoksi päädyttiin käyttämään Eurosoftin QA+WIN32-ohjelmaa, koska se oli yrityksessä jo ennestään käytössä ja toimivaksi havaittu. Testaukseen käytettävät skriptit jouduttiin suunnittelemaan uudelleen, koska tuotannon ja tuotekehityksen käyttämät testiskriptit eivät suoraan olleet sovellettavissa huollon käyttöön.

QA+WIN32-ohjelman testiskriptit ovat XML-kielisiä skriptejä. QA+WIN32 sisältää Windows-pohjaisen käyttöliittymän, jolla skriptien suunnittelua ja tekemistä on helpotettu. Käytettäessä käyttöliittymää skriptejä ei välttämättä tarvitse kirjoittaa suoraan, vaan eri testejä valittaessa ohjelma tekee niistä automaattisesti skriptin. Toimintaa voi verrata esimerkiksi webbisivujen tekoon käytettävään wysiwyg-editoriin, joka kääntää tehtävät sivut suoraan HTML-muotoon. QA+WIN32 ei itsessään vaadi valmiin skriptin tai testisalkun käyttöä, vaan ohjelma voidaan ajaa myös metsäkoneessa kiinni olevassa mittalaitetietokoneessa, kun ohjelman sisältävä USB-muisti sekä toiseen USB-porttiin liitettävä lisenssiavain kytketään metsäkoneen mittalaitetietokoneeseen.

6.3.2 QA+WIN32-ohjelmiston testien soveltaminen mittalaitetietokoneen testauksessa

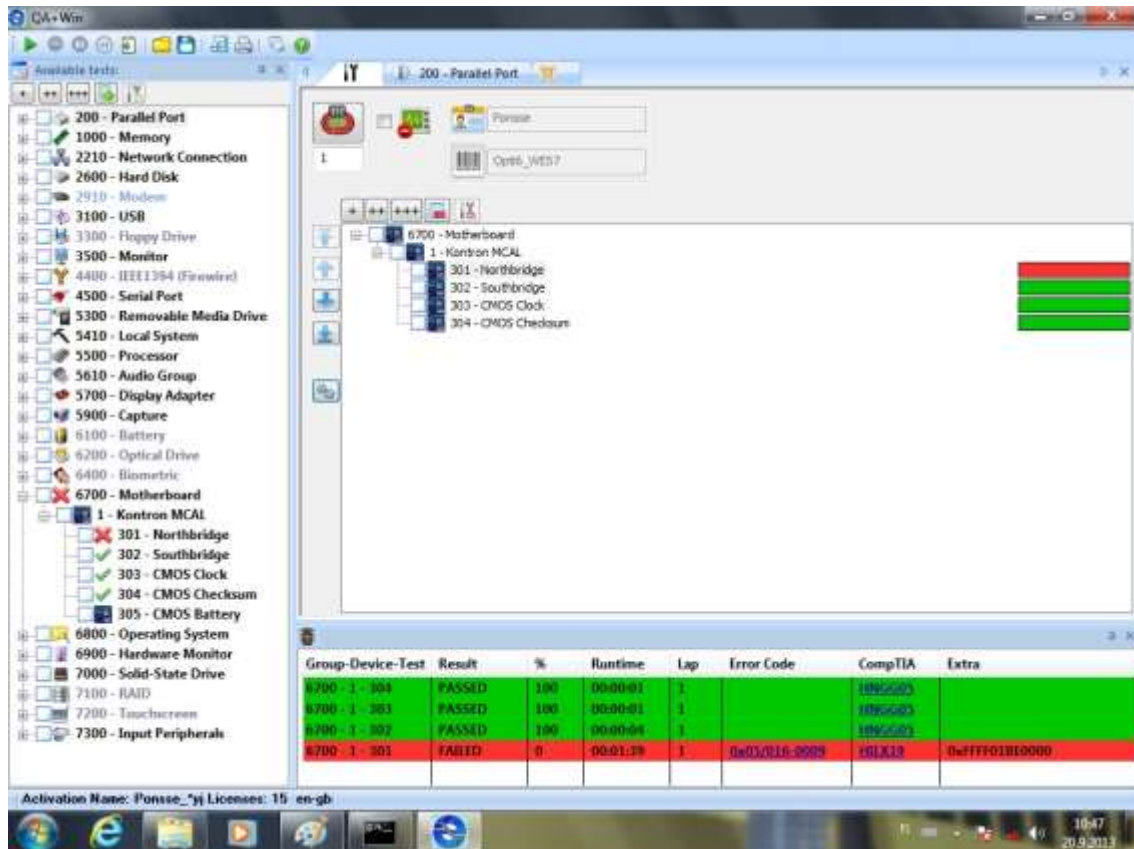
Mittalaitetietokoneen testauksen suunnittelussa tulee ensimmäisenä huomioida se, että tarkoituksena ei ole löytää vikaa yksittäisen piirin tasolla. Sen sijaan tarkoituksena on pyrkiä helpottamaan sitä, mikä varaosina saatavana oleva komponentti pitää vikaantuneeseen laitteeseen vaihtaa. Tällöin vaihdettavat komponentit ovat kerralla oikeita, mutta erityisesti ei myöskään varmuuden vuoksi vaihdeta komponentteja, jotka olisivat täysin toimivia.

QA+WIN32-ohjelmiston testeillä voidaan testata PC-laitetta hyvin monipuolisesti. Mittalaitetietokoneen rakenne ja esimerkiksi sarjaporttien kytkentä muutoin kuin standardeilla liittimillä aiheuttaa kuitenkin rajoitteita testauksen toteuttamismahdollisuuksiin, koska normaaleita sarjaporttitestauksessa käytettäviä porttiin liitettäviä plug-inejä eli lisäosia ei voida käyttää. Samoin USB-porttien testaaminen erillisen plugin avulla on tietokoneen sijainnin ja kytkennän vuoksi erittäin hankalaa.

Testeistä päätettiinkin soveltaa käytäntöön lähinnä laitteen perustoimintoja testaavia osuuksia, joissa ei normaalisti tarvitse käyttää portteihin kytkettäviä lisälaitteita. Tällöin pystytään riittävän suurella todennäköisyydellä joko

varmistamaan komponenttien viallisuus tai toisaalta rajaamaan ulos ehjiä komponentteja, mikä yhtä lailla johtaa oikean komponentin vaihtoon.

Testien suorittaminen antaa reaaliaikaisesti tietoa testien edistymisestä sekä niiden tuloksista testattavan tietokoneen näytöllä. Lisäksi testiskripti kirjoitetaan siten, että testiohjelman muistitikulle kirjoitetaan raportti testistä ja tämä on edelleen helppo lähettää tehtaan tekniseen tukeen, mikäli testitulosten tulkinta ei onnistu paikallisesti, tai mikäli testiraportteja erikseen pyydetään esimerkiksi uuden tuoteperheen lanseerauksen jälkeen. Esimerkki ohjelman ajon aikana näytettävästä reaaliaikaisesta tiedosta on kuvassa 9. Liitteessä 4 on esimerkki ohjelman muistitikulle tallentamasta testiraportista.



KUVA 9. QA+WIN32-ohjelman ikkuna testin suorituksen aikana

6.3.3 Muut testaustarpeet

Varsinaisen mittalaitetietokoneen laitetestauksen lisäksi metsäkoneen ohjausjärjestelmässä on osia, joita ei pystytä suoraan testaamaan

tehdastekoisilla ohjelmilla, vaan niiden testauksessa on käytettävä erityisesti sitä varten tehtyjä ohjelmistoja. Näiden toimintojen testaus on kuitenkin huollon toiminnan kannalta tärkeää, koska niiden merkitys koneen käytännön toiminnan kannalta on varsin suuri. Toimintoja, joita erillisillä ohjelmilla voidaan testata, ovat koneenohjausjärjestelmän Arcnet-verkko, GPS-paikannus, mittalaitetietokoneen Stellaris-piirin toiminta sekä järjestelmäinformaatiota näyttävä HW-testi. Lisäksi lähinnä koulutettujen mittalaitesuoltojen käyttöön päätettiin laatia ohjeet, joiden avulla mittalaitesuolot voivat tarvittaessa käyttää QA+WIN32-ohjelmaa yksittäisen komponentin tai osa-alueen testaamiseen.

6.4. Testit ja niiden tulosten hyödyntäminen

6.4.1. Pikatesti

Pikatestin suunnittelun tavoitteena oli se, että huoltomies pystyy ajamaan testin riittävän nopeasti metsäkoneessa huoltokäynnin aluksi varmistuakseen mittalaitetietokoneen toiminnasta ennen muun vianetsinnän aloittamista. Tällöin testin suorittamiseen kuluva aika ei saa ylittää 20–30:tä minuuttia. Pikatestissä testataan seuraavat tietokoneen osat: prosessori, prosessorikortti, muisti, näytönohjain, näyttö sekä massamuisti. Pikatesti voitiin sen lyhyen keston ansiosta tehdä niin, että se vaatii testaajalta palautetta.

Pikatestin ensimmäisessä osassa testataan mittalaitetietokoneen RAM-muistin toiminta. Testaukseen käytetään useita eri testejä seuraavassa järjestyksessä, koska muistipiirin toimintahäiriöissä yhden yksittäisen testikuvion käyttö ei useinkaan paljasta toimintavirhettä. Ensimmäisenä ajetaan pikatesti, joka kirjoittaa muistiin kuvion 0xC ja tarkistaa, että kirjoitettu kuvio tallentui muistiin oikein. Parametreillä on määritelty testin kestoksi 60 sekuntia, testin kattavuudeksi 100 % testattavasta muistista sekä cachen tila päälle. Pikatestin XML-koodi on esitetty koodissa 1.

```

6      <Test Test="301" TestName="Quick">
7          <TestParams>
8              <TestParam ID="1" Name="Duration" Value="60">
9                  </TestParam>
10             <TestParam ID="2" Name="Coverage" Value="100">
11                 </TestParam>
12             <TestParam ID="3" Name="EnableCaching" Value="1">
13                 </TestParam>
14             </TestParams>
15         </Test>

```

KOODI 1. Muistin pikatestin XML-koodi

Seuraavissa testeissä 1-bittiä siirretään bitti kerrallaan oikealle tai vasemmalle muutoin 0:ksi kirjoitettujen bittien läpi, ja tarkistetaan että bittien muutos tapahtuu oikein. Näiden testien parametrit on määritelty siten, että testien kesto on 60 sekuntia, kattavuus on 100% ja cache on päällä. Siirrettävän bitin testien XML-koodi on esitetty koodissa 2.

```

16     <Test Test="303" TestName="Walking Bit Left">
17         <TestParams>
18             <TestParam ID="1" Name="Duration" Value="60">
19                 </TestParam>
20             <TestParam ID="2" Name="Coverage" Value="100">
21                 </TestParam>
22             <TestParam ID="3" Name="EnableCaching" Value="1">
23                 </TestParam>
24             </TestParams>
25         </Test>
26     <Test Test="304" TestName="Walking Bit Right">
27         <TestParams>
28             <TestParam ID="1" Name="Duration" Value="60">
29                 </TestParam>
30             <TestParam ID="2" Name="Coverage" Value="100">
31                 </TestParam>
32             <TestParam ID="3" Name="EnableCaching" Value="1">
33                 </TestParam>
34             </TestParams>
35         </Test>

```

KOODI 2. Siirrettävän bitin testin XML-koodi

Neljäs muistille tehtävä testi muistuttaa toiminnaltaan hieman ensimmäistä testiä, mutta tällä kertaa muistiin kirjoitetaan ensin kuvio 0xA ja sen jälkeen kuvio 0x5. Molempien kirjoituskertojen jälkeen suoritetaan vertailu, jolla varmistetaan että kirjoitus on onnistunut. Tämän testin parametrit on määritelty vastaaviksi, kuin edellisissä testeissä. Testin XML-koodi on esitetty koodissa 3.

```

36     <Test Test="307" TestName="Checkerboard">
37         <TestParams>
38             <TestParam ID="1" Name="Duration" Value="60">
39                 </TestParam>
40             <TestParam ID="2" Name="Coverage" Value="100">
41                 </TestParam>
42             <TestParam ID="3" Name="EnableCaching" Value="1">
43                 </TestParam>
44         </TestParams>
45     </Test>

```

KOODI 3. Shakkilautatestin XML-koodi

Viimeisinä muistitesteinä suoritetaan juuttuneiden bittien testaus. Ensimmäisessä testissä kaikki muistin bitit kirjoitetaan ensin 1:ksi ja sen jälkeen 0:ksi, jonka jälkeen tarkistetaan että 0:ksi kirjoitus on onnistunut. Toisessa testissä toiminta on päinvastainen, jolloin lopuksi tarkistetaan että kirjoitus onnistuu 1:ksi. Myös tämän testin parametrit vastaavat aiempia muistitestejä. Testien XML-koodi on esitetty koodissa 4.

```

46     <Test Test="308" TestName="Bit Stuck High">
47         <TestParams>
48             <TestParam ID="1" Name="Duration" Value="60">
49                 </TestParam>
50             <TestParam ID="2" Name="Coverage" Value="100">
51                 </TestParam>
52             <TestParam ID="3" Name="EnableCaching" Value="1">
53                 </TestParam>
54         </TestParams>
55     </Test>
56     <Test Test="309" TestName="Bit Stuck Low">
57         <TestParams>
58             <TestParam ID="1" Name="Duration" Value="60">
59                 </TestParam>
60             <TestParam ID="2" Name="Coverage" Value="100">
61                 </TestParam>
62             <TestParam ID="3" Name="EnableCaching" Value="1">
63                 </TestParam>
64         </TestParams>
65     </Test>

```

KOODI 4. Juuttuneen bitin testin XML-koodi

Mikäli muistitestit eivät mene läpi, on lähes kaikkien virhekoodien seurauksena muistipiirien vaihto. On kuitenkin olemassa myös koodeja, jotka voivat aiheutua käyttöjärjestelmästä. Tällöin mittalaitetietokone on syytä toimittaa mittalaitahuoltoon, jossa järjestelmälle suoritetaan uudelleenasetus.

Muistitestien virhekoodit ja niiden seuraus mittalaitetietokoneen huollon kannalta on esitetty taulukossa 3.

TAULUKKO 3. Muistitestin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x00/3FF 0x02/3FF 0x03/004 0x04/004 0x05/004 0x05/005 0x06/004 0x07/001 0x0C/009 0x0D/3FF 0x0E/3FF 0x0F/3FF 0x10/3FF	Muistipiirin testaus ei mene läpi ja virhe johtuu muistipiirin sisäisestä viasta. Muistipiirit on vaihdettava.
0x08/3FF	Virhe testiskriptissä. Tämän virheen esiintyessä otettava yhteys tehtaan mittalaitahuoltojen tukeen
0x01/3FF 0x09/001 0x0B/3FF 0x0A/3FF	Käyttöjärjestelmän aiheuttama virhe. Suorita käyttöjärjestelmän ja ohjelmistojen uudelleen asennus.

Seuraavaksi pikatesti suorittaa näytön testauksen. Näyttötesti on interaktiivinen testi, jossa testin suorittaja tarkistaa näyttöön tulevan kuvan oikeellisuuden tarkistamalla, ettei näytössä ole sammuneita tai väärän värisiä pikseleitä. Testin tuloksien hyväksikäytettävyyden osalta on vielä määriteltävä taso, jolloin näyttöpaneeli esimerkiksi vaihdetaan takuukorjauksena. Näytön vaihtaminen ei normaalisti ole tarpeen yksittäisen pikselin ollessa viallinen, mutta esimerkiksi

kymmenen viallisen pikselin seurauksena. Luonnollisesti viallisen pikselin aiheuttamien näytön vaihtojen kriteerinä tulee myös käyttää sitä, miten häiritsevässä paikassa koneen käyttöä ajatellen viallinen pikseli sijaitsee.

Näyttötestien parametreissa määritellään koodissa 5 määritellyllä tavalla, näytetäänkö ennen testiä dialogi käyttäjälle. Dialogia ei käytetä tässä yhteydessä, vaan kuva näytetään suoraan, jonka jälkeen ohjelma pyytää käyttäjältä palautteen, oliko näytetty yksivärinen kuva näytössä oikein. Viimeisenä testissä olevassa sammuneen pikselin testissä määritellään näytettävän kuvan näyttöajaksi parametrilla 5 sekuntia. Testin antama tulos määrittyy käyttäjän antaman palautteen perusteella. Taulukon 4 mukaisen tuloksen perusteella tehdään ratkaisu joko näyttöpaneelin vaihtamisesta tai sen vaihtamatta jättämisestä.

```
68 <Group Group="3500" GroupName="Monitor">
69 <Device Device="1">
70   <Test Test="301" TestName="Red Purity">
71     <TestParams>
72       <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
73     </TestParams>
74   </Test>
75   <Test Test="302" TestName="Green Purity ">
76     <TestParams>
77       <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
78     </TestParams>
79   </Test>
80   <Test Test="303" TestName="Blue Purity ">
81     <TestParams>
82       <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
83     </TestParams>
84   </Test>
85   <Test Test="310" TestName="LCD Dead Pixel">
86     <TestParams>
87       <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
88       <TestParam ID="2" Name="ScreenPeriod" Value="5"></TestParam>
89     </TestParams>
90   </Test>
91 </Device>
92 </Group>
```

KOODI 5. Näyttötestit

TAULUKKO 4. Näyttötestin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x00/3FF	Käyttäjä antoi testistä palautteen, jonka mukaan näyttö ei ole kunnossa. Näyttöpaneeli vaihdettava, mikäli vikaantuneet pikselit häiritsevät käyttöä ja niitä on riittävän paljon.

Prossessorin testauksessa käytetään useita eri testejä, jotka testaavat prosessorin eri käskykantojen toimintaa koodin 6 mukaisesti. Kaikissa testeissä peruseriaatteena on, että prosessorilla teetetään eri käskykantojen mukaisia komentoja ja niiden tuloksia verrataan etukäteen laskettuihin oikeat tulokset sisältäviin taulukoihin. Mikäli tulokset eivät vastaa taulukkoarvoja, testi ei mene läpi.

```
93 <Group Group="5500" GroupName="Processor">
94 <Device Device="1">
95   <Test Test="301" TestName="Core Instruction Set">
96     <TestParams></TestParams>
97   </Test>
98   <Test Test="302" TestName="Floating Point Instruction Set">
99     <TestParams></TestParams>
100  </Test>
101  <Test Test="303" TestName="MMX Instruction Set">
102    <TestParams></TestParams>
103  </Test>
104  <Test Test="304" TestName="SSE Instruction Set">
105    <TestParams></TestParams>
106  </Test>
107  <Test Test="305" TestName="SSE2 Instruction Set">
108    <TestParams></TestParams>
109  </Test>
110  <Test Test="306" TestName="SSE3 Instruction Set">
111    <TestParams></TestParams>
112  </Test>
113 </Device>
114 </Group>
```

KOODI 6. Prossessorin testit

Käytännön toimenpiteenä taulukon 5 mukaisessa tilanteessa, jossa prosessorin testi ei mene läpi, on mittalaitahuollon tapauksessa prosessorikortin vaihto, koska pelkkää prosessoria ei tarjota varaosana mittalaitahuolloille.

TAULUKKO 5. Näyttötestin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x01/009 0x02...0x0F/3FF 0x10...0x19/3FF	Proessori ei läpäise testejä. Proessorikortti on vaihdettava.
0x00/3FF	Proessori ei tue testattavaa käskykantaa. Otettava yhteyttä tehtaan mittalaitahuollon tukeen.

Seuraavana testissä testataan näytönohjaimen sekä näyttömuistin toiminta koodin 7 mukaisesti. Testissä suoritetaan ensin kolme näyttömuistia testaavaa testiä. Testeissä näytölle piirretään ennalta määritellyt kuviot, ja näyttömuistin sisältöä verrataan ennalta piirrettyjen kuvien mukaiseen sisältöön. Näytönohjaimen testauksen aikana on tärkeää, että testin suorittaja ei käytä konetta millään tavoin, koska tämä voi johtaa näytön päivittämiseen mikä saattaa sekoittaa testin tuloksen. Viimeisessä testissä testataan näytönohjaimen 3d-kiihdytyksen toiminta, jossa käytännössä testataan eroavaisuutta näytönohjaimen 3d-toimintojen ja ohjelmallisesti lasketun 3d-kuvan välillä. Mikäli eroavaisuuksia on liikaa, testi ei mene läpi, jolloin on tehtävä taulukon 6 mukaiset toimenpiteet.

```
115 <Group Group="5700" GroupName="Display Adapter">
116 <Device Device="1">
117   <Test Test="301" TestName="Linear Memory ">
118     <TestParams></TestParams>
119   </Test>
120   <Test Test="302" TestName="Microtopology Memory">
121     <TestParams>
122       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
123     </TestParams>
124   </Test>
125   <Test Test="303" TestName="Chaotic Addressing Memory">
126     <TestParams>
127       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
128     </TestParams>
129   </Test>
130   <Test Test="304" TestName="Hardware Acceleration">
131     <TestParams>
132       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
133     </TestParams>
134   </Test>
135 </Device>
136 </Group>
```

KOODI 7. Näytönohjaimen testi.

TAULUKKO 6. Näytönohjaintestin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x00/001 0x01/001 0x02/001 0x03/3FF 0x04/001 0x05/3FF 0x06/001 0x18/3FF	Testit eivät mene läpi testiohjelman ikkunaan liittyvän virheen vuoksi. Varmista, että testien aikana ei ole käytössä muita ohjelmia eikä testaaaja ole käyttänyt laitetta testin aikana. Mikäli edellä mainitut asiat ovat kunnossa, on järjestelmä asennettava uudelleen.
0x08/3FF 0x0A/001 0x0B...0x0F/3FF 0x15/001 0x16/001 0x12/008 0x14/008 0x18/3FF	Varmista näytönohjaimen ajurien toiminta. Mikäli ajurit ovat kunnossa, näytönohjain on viallinen ja prosessorikortti on vaihdettava.
0x07/007 0x09...0x18/3FF	Näytönohjaimen toiminnassa on havaittu virhe. Näytönohjain on viallinen, prosessorikortti on vaihdettava.

Seuraavassa vaiheessa suoritetaan emolevyn ja prosessorikortin muiden piirien testaus koodin 8 mukaisesti. Näissä testeissä testataan northbridgen ja southbridgen toiminta, sekä CMOS-kellon ja tarkistussumman oikeellisuus. Northbridge-testissä määrätään parametrein testin kestoksi 30 sekuntia, käytettäväksi taajuudeksi 500 MHz sekä toleranssiksi 30 prosenttia. Muut tehtävät testit eivät vaadi parametointia. Southbridge-testissä tehdään sarjaportin sekä ethernet-portin testi. CMOS-reaaliaikakelloa verrataan järjestelmän kelloon ja CMOS:n tarkistussummaa verrataan laskettuun tarkistussummaan. Taulukossa 7 on esitetty emolevyn testin mahdolliset tulokset.


```

137 <Group Group="6700" GroupName="Motherboard">
138 <Device Device="1">
139   <Test Test="301" TestName="Northbridge">
140     <TestParams>
141       <TestParam ID="1" Name="Duration" Value="30"></TestParam>
142       <TestParam ID="2" Name="FSBSpeed" Value="500"></TestParam>
143       <TestParam ID="6" Name="FSBTolerance" Value="50"></TestParam>
144     </TestParams>
145   </Test>
146   <Test Test="302" TestName="Southbridge">
147     <TestParams></TestParams>
148   </Test>
149   <Test Test="303" TestName="CMOS Clock">
150     <TestParams></TestParams>
151   </Test>
152   <Test Test="304" TestName="CMOS Checksum">
153     <TestParams></TestParams>
154   </Test>
155 </Device>
156 </Group>

```

KOODI 8. Emolevyn testi

TAULUKKO 7. Emolevyn testin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x01/017 0x02/3FF 0x03/3FF 0x04/3FF	CMOS-testit eivät mene läpi. Tähän saattaa vaikuttaa emolevyllä olevan CMOS-patterin huono kunto. Vaihda ensin emolevyn patteri ja jos ei tämä auta, vaihda emolevy.
0x06...0x0f/3FF 0x0C/080 0x05/016	Emolevyn testit eivät mene läpi. Vaihda emolevy.

Viimeisenä pikatestissä testataan koodin 9 mukaisesti SSD-asema kahdella erillisellä lukutestillä. Ensimmäisenä on lineaarinen lukutesti, jossa lukuosoitteita kasvatetaan lineaarisesti ja toisena satunnaisten lukujen testi, jossa lukuosoitteet määritellään satunnaisalgoritmien avulla. Jokaisen lukukerran onnistuminen tarkistetaan. SSD-aseman testin tulokset on esitetty taulukossa 8.

```

157 <Group Group="7000" GroupName="Solid-State Drive">
158 <Device Device="1">
159   <Test Test="301" TestName="Linear Read">
160     <TestParams>
161       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
162       <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
163     </TestParams>
164   </Test>
165   <Test Test="302" TestName="Random Read">
166     <TestParams>
167       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
168     </TestParams>
169   </Test>
170 </Device>
171 </Group>

```

KOODI 9. SSD-aseman testi

TAULUKKO 8. SSD-aseman testin tulokset

Virhekoodi tai -koodit	Todennäköisin syy ja toimenpiteet
0x00/001 0x01/005 0x03/005 0x02/006 0x04/3FF	Lukutestit epäonnistuvat. Vaihda SSD-levy ja asenna järjestelmä uudelleen.

Pikatestin koko koodi on liitteessä 5. Pikatestin tuloksien tulkinnessa on pyritty tuottamaan lista, jonka perusteella huoltomies voi päätellä koneella tarvitseeko mittalaitetietokone toimittaa mittalaitahuoltoon. Lisäksi mikäli huoltotarve on olemassa, voi huoltomies jo tietokoneen mittalaitahuoltoon toimittaessaan antaa paremman vikakuvauksen mittalaitahuollolle ja säästää näin aikaa diagnostiikalta.

Esimerkki testausohjelman tulostamasta testiraportista on liitteessä 4. Raportista voi testitulosten lisäksi nähdä laitteiston konfiguraation, mistä voidaan myös takuuajana päätellä, onko laitteelle tehty takuumääräysten vastaisia muutoksia ja korjauksia. Tämä on myös tärkeää, koska monilla markkinoilla houkutus halvempien tietokonetarvikkeista ostettavien kuluttajatasen osien käyttämiseen korjauksissa ja korjausyrityksissä on suuri. Tällöin mittalaitetietokoneen takuu luonnollisesti raukeaa.

6.4.2. Rasitustesti

Luotaessa rasitustestiä oli tarkoitus luoda testi, jonka avulla laite voidaan jättää päälle ja kuormitetuksi useaksi tunniksi, jolloin mahdollisen kuormituksen sekä lämpiämisen tuoma epävakaas saadaan esille. Testi perustuu lähes kokonaisuudessaan pikatestiin sillä erotuksella, että rasitustestistä on jätetty pois käyttäjän palautetta vaativana osuutena näytön testaus. Tarkoituksena on tällöin ollut jättää testeihin ne osa-alueet, joiden toimintaa lämpiäminen todennäköisimmin haittaa.

Rasitustestissä ajetaan 40 kertaa testisilmukka, joka testaa muistin, prosessorin, näytönohjaimen ja näyttömuistin, emolevyn sekä SSD-levyn. Tämä testi vie aikaa noin 8 tuntia, mikäli se menee kaikilta osiltaan läpi. Testi on toteutettu siten, että se ei pysähdy vaikka jokin osa-alue ei läpäise testiä vaan jatkaa silmukkaa niin kauan kunnes 40 kierrosta on täynnä tai käyttäjä keskeyttää testin. Tällöin voidaan varmistua myös siitä, että jokaisella kierroksella sama komponentti ei läpäise testausta.

Vikakoodien sekä toimenpideohjeiden kohdalla pätevät samat ohjeet kuin pikatestin osalta. Tällöin vikakoodin sekä sen aiheuttaman testin perusteella voidaan päätellä, mikä komponentti mittalaitetietokoneeseen tulee vaihtaa.

6.4.3. Arcnet-testi

Arcnet-testi on erillinen ohjelma, jolla pystytään testaamaan metsäkoneen Arcnet-verkon tilaa. Mittalaitteeseen on sisällytetty verkon testaukseen kehitetty ohjelma. Tämän ohjelman käynnistävä komentojono sisällytettiin testausohjelman muistitikulle ja ohjelman antamista tiedoista laadittiin ohje, jonka perusteella huoltomiesten on helpompi tehdä vikadiagnostiikka Arcnet-verkolle.

Arcnet on verkko, joka perustuu tokenin eli lähetysvaltuutuksen käyttöön. Tällöin jokaisella verkkoon kytketyllä moduulilla on vuorollaan taattu pääsy verkkoon, koska jokainen laite saa vuorollaan varata valtuutuksen itselleen.

Tämä on metsäkonekäytössä yksi Arcnet-verkon etu CAN-väylään verrattuna, koska viestien priorisointia eri moduulien välillä ei tarvitse suunnitella yhtä tarkasti vaan kaikki moduulit pääsevät vuorollaan lähettämään viestejä verkkoon. Toisaalta, ongelmaksi saattaa jossakin tilanteessa muodostua se, että viestien lähettämisen väli voi olla hieman toivottua pidempi. Tämä on otettava huomioon moduulin ohjelmiston suunnittelussa sekä siinä, mitä kriittisiä tietoja välitetään verkon yli ja mitä toimintoja moduulin on kyettävä tekemään ilman verkon yli saatavaa tietoa.

Kuvassa 10 näkyvän Arcnet statistics-ohjelman tuloksista voidaan päätellä monia asioita. Transmitfails-sarakkeessa nähdään, mikäli jonkin viestin lähettäminen ei ole onnistunut. Tähän sarakkeeseen ei normaalisti tule lainkaan lukemia, ellei moduuleita ohjelmoida samalla kun Arcnet statistics -ohjelma on päällä. Mikäli tähän sarakkeeseen tulee lukemia, on moduulissa todennäköisesti sisäinen vika, koska moduulin I/O-piiri vastaanottaa tokenin, mutta moduuli ei sisäisesti lähetä verkkoviestin vastaanottokuittausta.

The screenshot shows a window titled "Arcnet statistics" with a table containing the following data:

Node	Transmitfails	Booted	Been suspect	Messages	State	My Recons	StartupCode
Opti (255)	0	0	0	15		0	0
HarvesterHead (254)	0	0	0	0		0	0
Seat (253)	0	0	0	0		0	0
Transmission (252)	0	0	0	0		0	0
Crane (251)	0	0	0	0		0	0
Active damping (250)							

At the bottom of the window, there are buttons for "Close", "-> 0", and "Raw", along with the text "Version 4.710.1".

KUVA 10. Arcnet statistics -ohjelma

Booted-sarakkeeseen lasketaan kerrat, jolloin kyseinen moduuli lähettää boottausviestin, eli osoitteen varauksen verkkoon. Vertaamalla tämän sarakkeen arvoa StartupCode-sarakkeen arvoon, missä kerrotaan verkossa moduulille lähetettyjen ohjelmallisten boottauspyyntöjen lukumäärä, voidaan nähdä moduulin virransyötössä tai sisäisessä virranhallinnassa mahdollisesti olevat ongelmat.

Been suspect -sarakeessa nähdään, jos moduuli ei ole ottanut vastaan tokenia. Tällöin moduulin verkkopiiri ei ole kunnossa. Käytännössä tämä tarkoittaa sitä, että moduuli on vaihdettava, koska myös Arcnet-piirit ovat moduulin sisäisiä piirejä. Messages-sarakeesta taas nähdään Arcnet-piirin moduulille välittämien viestien sekä niistä piirin verkkoon lähettämien vastaanottokuitausten määrä. Mikäli moduuli vastaanottaa viestejä mutta ei muutoin toimi, on mahdollista että sen ohjelmisto on jumiintunut. Tällöin moduulin saattaa saada toimimaan uudelleenohjelmoinnilla. Viestien määrää seuraamalla voidaan myös nähdä eri moduulien toimintaa, koska periaatteessa kaikkien moduulien viestilaskurin pitäisi nousta samaa vauhtia muutamien viestien tarkkuudella.

State-sarakeesta nähdään moduulin verkkoyhteyden tilan verkossa. Koska tämä tila saadaan Arcnet-ohjaimelta ilman että moduulin ohjelmisto vaikuttaa tähän millään tavalla, näyttää Status-näyttö verkossa olevat lyhytaikaisetkin ongelmat erittäin luotettavasti. Status-sarakeen arvon ollessa vihreä tokenin kierrätys moduulin kautta toimii normaalisti. Arvon ollessa punainen, verkko-osoitteeltaan alempi moduuli, joka lähettää tokenin moduulille, ei saa kyseiseltä, verkko-osoitteeltaan seuraavalta moduulilta tokenin vastaanottokuitausta. Jotta tila vaihtuu edelleen toiseksi, on tokenin kierron oltava normaali vähintään kolme sekuntia,

Mikäli tila on keltainen, toimii tokenin kierto normaalisti, mutta mittalaitetietokone ei pysty vastaanottamaan verkon diagnostiikkaviestejä. My recons -sarakeessa lasketaan kaikki verkossa olevat virheet, jonka jälkeen moduuli yhdistää itsensä verkkoon uudelleen. Recons-nimi saadaan englannin sanasta reconnect. Tällainen uudelleenyhdistäminen saattaa johtua siitä, että yksi moduuleista lähettää virheellisiä viestejä, jolloin tämän yhden moduulin My recons -lukeman ei pitäisi nousta, mutta kaikkien muiden moduulien lukema nousee.

Arcnet statistics -ohjelmalla pystytään tekemään hyvin monipuolista diagnostiikkaa verkon laitteille ja eristämään mahdollisia verkko-ongelmia joko yksittäisiin moduuleihin tai eri moduulien välille. Tällöin verkon tähtitopografia on erittäin hyödyllinen, koska yksittäisen moduulin tai sille johtavan kaapeloinnin vika ei aja koko verkkoa vikatilaan ja tämän kaltaisella ohjelmalla on sen jälkeen helppo nähdä vian aiheuttama moduuli.

6.4.4. GPS-testi

GPS-testi on yksinkertainen komentojono, jolla käytetään Teraterm Pro-ohjelmaa lukemaan NMEA-jono GPS-ohjaimelta. NMEA-jonosta voidaan päätellä, onko GPS-antennilla yhteys satelliitteihin ja tuottaako GPS-ohjain oikeaa sijaintidataa.

```
$GPGAA | 062454.000 | 6413.1077,N | 02746.7330,E | 1 | 07 | 1.7 | 166.3,M | 19.4,M | ,|0000*53|
```

KUVA 11. NMEA-jono

Kuvan 11 kaltaisesta NMEA-jonoista tarkastellaan metsäkonekäytössä normaalisti GPGAA-alkuista jonoa, koska muut NMEA-jonot näyttävät eri korjaussignaaleilla saatavien sijaintitietojen tilan. Koska metsäkonekäytössä ei käytetä korjaussignaalia, luetaan ainoastaan tätä jonoa. Jonosta voidaan nähdä vasemmalta oikealle lukien seuraavat asiat: UTC-aika, GPS-koordinaatit, GPS:n taso, satelliittien lukumäärä, horisontaalinen tarkkuus, korkeus merenpinnasta, geoidin korkeus merenpinnasta, sekä kahdesta viimeisestä kohdasta differentiaalikorjauksen tietoja, jotka eivät metsäkonekäytössä ole merkityksellisiä.

GPS-testin suurin merkitys on se, että pystytään eristämään mahdolliset paikannusongelmat joko fyysiseen GPS-järjestelmään tai paikannustietoa käyttäviin ohjelmistoihin. Mikäli fyysisessä GPS-järjestelmässä on vikaa, tulee ensimmäisenä tarkistaa antenni ja sinne johtava kaapeli liitoksineen ja mikäli GPS ei edelleenkään toimi, vaihtaa mittalaitetietokoneen emolevy. Ohjelmistongelmien ratkomisen on paikannussovellusten kyseessä ollessa hieman

ongelmallisempaa, koska ne ovat harvoin metsäkoneen mukana toimitettuja ohjelmia. Kolmansien osapuolien ohjelmistojen ongelmista vastaa pääsääntöisesti ohjelmistojen toimittaja itse.

6.4.5. Stellaris-testi

Kuvassa 12 esitettävällä Stellaris-testiohjelmalla voidaan tarkastella emolevyn Stellaris-piirin tietoja. Näistä käytännössä tärkeimmät ovat Stellariksen ohjelmistoversion tarkistus, ohjaamoakun latauksen tilan tarkistus, Stellariksen ulkoiset inputit sekä CAN-väylädiagnostiikka.

Näistä Stellariksen ohjelmistoversion tarkistus tarvitaan tietyissä tilanteissa päivitettäessä mittalaitteohjelmistoa, jolloin myös Stellariksen ohjelmisto on päivitettävä erillisellä työkalulla. Ohjaamoakun latauksen tilan tarkistuksesta taas voidaan varmistua, että mittalaitetietokone lataa ohjaamoakkuja normaalisti ja että ohjaamoakku on kunnossa.

Inputeista tärkein on wake1-input, josta nähdään Webaston antaman esilämmitysherätteen tila. CAN-väylädiagnostiikkaa tarvitaan kuormatraktoreissa vaakayhteyksiä käytettäessä, koska mittalaitte on kytkettynä vaakajärjestelmään muista järjestelmistä poiketen CAN-väylällä.



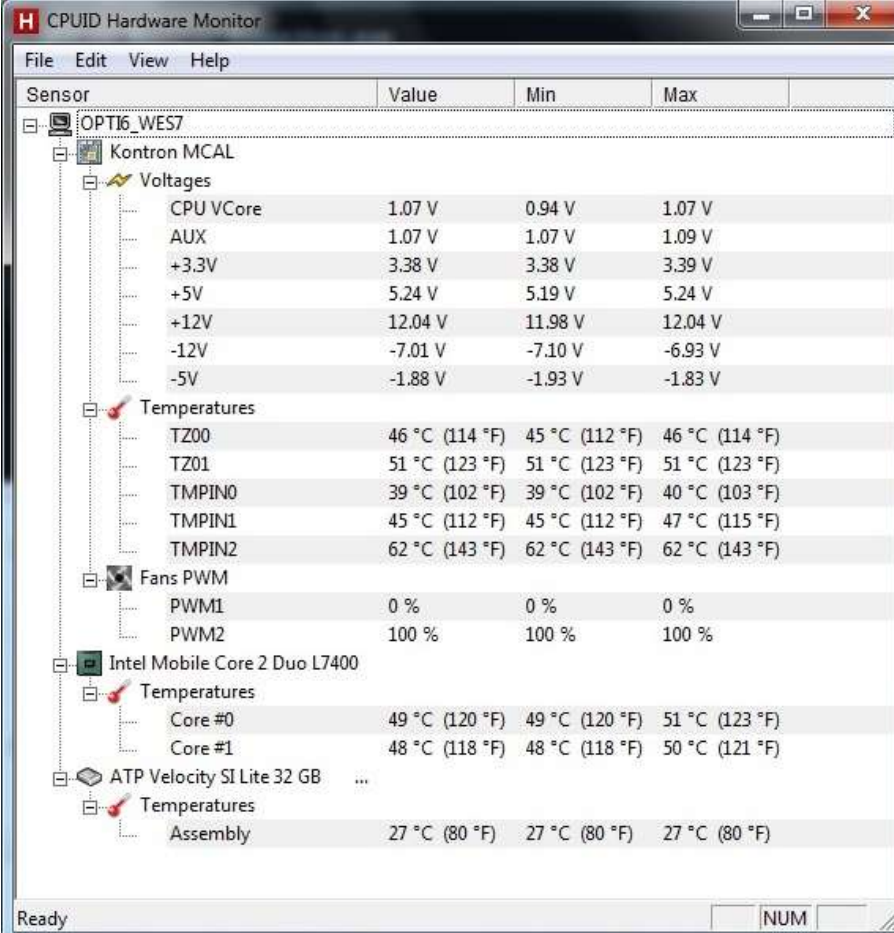
KUVA 12. Stellaris-testi

6.4.6. HW-testi

Kuvan 13 HW-testillä voidaan tarkkailla prosessorikortin lämpötiloja sekä jännitteitä tietokoneen käydessä. HW-testillä on mahdollista esimerkiksi tarkkailla, mikäli jonkin komponentin lämpötila nousee poikkeuksellisen korkeaksi metsäkoneella työskennellessä, tai mikäli prosessorikortin jännitteet heittelevät.

Lämpötilojen noususta voi päätellä komponentin viallisuuden lisäksi sen, onko syytä tarkistaa esimerkiksi korjauksen jälkeen jonkin piirin lämmönjohtotahnan asennuksen onnistuminen. Jännitteiden heilahtelusta voi nähdä mikä emolevyn

jännitteenalentimista ei toimi kunnolla, joskin jännitteenalentimien vikaantuminen johtaa aina koko emolevyn vaihtoon.



The screenshot shows the CPUID Hardware Monitor application window. The window title is 'CPUID Hardware Monitor'. The menu bar includes 'File', 'Edit', 'View', and 'Help'. The main content area displays a tree view of hardware sensors. The 'OPTI6_WES7' folder is expanded, showing 'Kontron MCAL' and 'Voltages'. The 'Voltages' section contains a table with columns for sensor name, current value, minimum, and maximum. Below this are 'Temperatures' for various sensors, 'Fans PWM' for two fans, 'Intel Mobile Core 2 Duo L7400' with its own temperature sensors, and 'ATP Velocity SI Lite 32 GB' with its assembly temperature.

Sensor	Value	Min	Max
OPTI6_WES7			
Kontron MCAL			
Voltages			
CPU VCore	1.07 V	0.94 V	1.07 V
AUX	1.07 V	1.07 V	1.09 V
+3.3V	3.38 V	3.38 V	3.39 V
+5V	5.24 V	5.19 V	5.24 V
+12V	12.04 V	11.98 V	12.04 V
-12V	-7.01 V	-7.10 V	-6.93 V
-5V	-1.88 V	-1.93 V	-1.83 V
Temperatures			
TZ00	46 °C (114 °F)	45 °C (112 °F)	46 °C (114 °F)
TZ01	51 °C (123 °F)	51 °C (123 °F)	51 °C (123 °F)
TMPIN0	39 °C (102 °F)	39 °C (102 °F)	40 °C (103 °F)
TMPIN1	45 °C (112 °F)	45 °C (112 °F)	47 °C (115 °F)
TMPIN2	62 °C (143 °F)	62 °C (143 °F)	62 °C (143 °F)
Fans PWM			
PWM1	0 %	0 %	0 %
PWM2	100 %	100 %	100 %
Intel Mobile Core 2 Duo L7400			
Temperatures			
Core #0	49 °C (120 °F)	49 °C (120 °F)	51 °C (123 °F)
Core #1	48 °C (118 °F)	48 °C (118 °F)	50 °C (121 °F)
ATP Velocity SI Lite 32 GB ...			
Temperatures			
Assembly	27 °C (80 °F)	27 °C (80 °F)	27 °C (80 °F)

KUVA 13. HW-monitor

6.4.7. Muut testausmahdollisuudet

QA+WIN32-ohjelmisto mahdollistaa myös yksittäisen testin ajamisen tietylle mittalaitteen komponentille suoraan ohjelmistosta. Tämä mahdollisuus on lähinnä mittalaitetuholloille käytännöllinen, koska tällöin on mahdollista varmistaa vaikkapa huoltomiehen antama metsässä tehty vikadiagnosi ennen varsinaisia korjaustoimenpiteitä.

Yksittäisen komponentin testaaminen tämän kaltaisessa tilanteessa säästää aikaa, mikäli samalla on käytettävissä riittävän läheltä testausajankohtaa oleva täydellinen testausraportti. Yksittäistä komponenttia testattaessa tulee kuitenkin

muistaa, että täydellisen testin ajaminen antaa aina laajemman kuvan koko järjestelmän toiminnasta.

Lisäksi testaussalkkuun sisällytettiin Stellaris-piirin ohjelmointiin tarvittavat työkalut, eli USB-sarjaporttiadapteri sekä ohjelmointikaapeli. Näiden käytöstä laadittiin myös tarkat ohjeet, koska usein mittalaitteohjelmisto päivitetään kentällä metsässä, jolloin myös huoltomiehen on pystyttävä tekemään Stellariksen päivitys.

6.5. Testaussalkun käyttöohjeet

Testaussalkulle laaditaan myös käyttöohjeet, jotka sisältävät selostuksen salkun sisäisestä rakenteesta, ohjelmistojen käyttöohjeet sekä tulosten tulkintaohjeet. Käyttöohjeen laadinta ei kuitenkaan kuulunut tähän insinööriyöhön ja siitä vastasivat Ponsen puolesta Lasse Tikkanen ja Petri Nousiainen. Käyttöohjeen laadinnassa oma roolini oli taustatietojen antaminen sekä tulosten tulkintaohjeiden siirtäminen käyttöohjeeseen.

6.6. Tuotteistus

Testaussalkusta tehtiin myytäväksi kaksi erilaista kokonaisuutta. Nämä testaussalkut ovat myynnissä Ponsen valtuutetuille huoltopisteille sekä maahantuojien valtuutetuille huolloille. Ensi vaiheessa testaussalkkuja tarjottiin maihin, joissa on myös oma valtuutettu mittalaittehuolto. Toisessa vaiheessa pitää harkita, onko testaussalkun tarjoamisesta myös muihin maihin riittävästi hyötyä, jotta tätä kannattaa harkita.

Full kit -nimellä myytävään täydelliseen sarjaan kuuluvat testaussalkku, täydellinen mittalaitetietokone sekä kaikki salkkuun kytkettävät ohjainmoduulit. Mittalaitetietokone sekä moduulit ovat suoraan hyödynnettävissä varaosina tai tarvittaessa vaihtolaitteina odottaessa rikkoontuneen komponentin korjausta. Basic kit -nimellä myytävään sarjaan ei sisälly mittalaitetietokonetta eikä ohjainmoduuleita. Basic kit sopiikin sellaiselle jälleenmyyjälle, jolla on jo

kyseiset laitteet omassa varaosavarastossaan ja joka haluaa hyödyntää niitä testauskassa kustannusten säästämiseksi.

7 TESTAUS

Testaussalkun testaamista suunnitellessa tuli suunnitella kaksi eri testausta. Ensinnä salkun toimivuus sen valmistuksen jälkeen ja sitä kautta salkkua toimitettaessa on kyettävä varmistamaan ja toisaalta on testattava salkun toiminnallisuus ennen sen sarjavalmistuksen aloittamista.

Salkun valmistusvaiheen testaukseen tulee sisältyä johdinsarjan testaus, joka tehdään yksinkertaisesti mittaamalla kytkentäkaavion mukaisesti johdinsarjan toimivuus siten, että eri liitinten välillä ei ole katkoksia yhteyksissä ja toisaalta johdinsarja ei ole oikosulussa mistään kohdasta. Tämä testaus suoritetaan johdinsarjan valmistuksen yhteydessä. Lisäksi koottaessa salkkua, mittalaitetietokoneen sekä sen ohjelmistojen, salkun Arcnet-verkon sekä virransyöttöjen toimivuus testataan kokeilemalla se käytännössä.

Salkun toiminnallista testausta suoritettaessa tehtiin prototyyppisalkulle ensin tuotantovaiheessa tehtäviä testauksia vastaavat testit. Sen lisäksi suoritettiin testausskriptien ajoja useille eri mittalaitetietokoneille, joissa pystyttiin näkemään se, että skriptit antavat toisaalta toimiviksi tiedetyissä mittalaitetietokoneissa halutut tulokset ja toisaalta viallisissa laitteissa ongelmat saadaan esille. Käytännössä viallisten laitteiden vähäinen määrä sarjatuotannon ollessa aikaisessa vaiheessa oli niin pieni, että osittain rikkoontuneiden laitteiden vianetsinnän onnistumisen todentaminen jää käytännössä tehtävän työn tulosten arviointiin.

Testausskriptien käytännön tulosten perusteella niiden koostumusta pystytään käytännössä säätämään, mikäli sellaista tarvetta ilmenee. Salkun toimivuudesta kerätään muutenkin palautetta eri maista, joissa sitä käytetään ja sitä kautta salkkua pyritään kehittämään eteenpäin, jotta se olisi mahdollisimman tuottava työkalu mittalaittehuolloille.

8 JATKOKEHITYSMAHDOLLISUUDET

Testaussalkun yhtenä ja ehkäpä tärkeimpänä jatkokehityskohteena on eri ohjainmoduulien I/O-toimintojen testaus. Sinänsä tarve tämän kaltaiselle testaukselle olisi olemassa jo nyt, mutta tästä projektista se jouduttiin jättämään pois sekä testauksen monimutkaisuuden että myös siitä aiheutuvien kustannusvaikutusten vuoksi. Lisäksi tämän kaltainen I/O-testaus olisi järkevää toteuttaa pienemmässä joukossa toimipisteitä, jolloin myös pystyttäisiin hallinnoimaan moduulien takuukäsittelyä ja -palautuksia paremmin.

Toinen jatkokehitysmahdollisuus on uusien päivitettyjen mittalaitteiden testauksen integrointi tämän työn tuloksena syntyneeseen testaussalkkuun. Jossakin määrin tulevaisuuden vaatimuksia on pyritty ottamaan salkun suunnittelussa huomioon, mutta esimerkiksi uuden Scorpion-harvesterin kaikkia anturimoduuleita ei vielä tässä vaiheessa integroitu salkkuun.

Tulevien mittalaitte- ja mittalaitetietokonemallien testausohjeita ja testauskriptejä laadittaessa pitääkin miettiä mahdollisuutta lisätä uudet anturimoduulit tähän salkkuun, koska koneenohjauksen perusrakenne on vielä seuraavan mallisarjan koneissa pääosin sama.

Tämän testaussalkun perusrakenne on kunnossa siihen saakka, kunnes koneenohjaukseen tehdään suurempi päivitys ja käytettävä väyläteknikka mahdollisesti muuttuu toiseksi. Perusajatus testausmenetelmistä ja niiden toteutuksesta säilyy samana myös siitä eteenpäin, mutta diagnostiikkatyökalujen kehittyessä testausta voidaan tällöin todennäköisesti muutenkin edelleen helpottaa ja tehostaa.

Seuraavan sukupolven Opti7-mittalaitteen testaukseen käytettäessä nykyiset testit toimivat lähes sellaisenaan. Ainoat muutostarpeet testiskripteissä ovat väylännopeusparametreissa emolevyn testiosuudessa, jolloin voidaan testata väylää korkeammilla nopeuksilla.

9 YHTEENVETO

Työn tarkoituksena oli laatia työkalu Ponssen mittalaittehuoltojen käyttöön. Työkalulla tuli pystyä helpottamaan mittalaitetietokoneen testausta siten, että mittalaittehuoltoon kouluttamatonkin huoltomies pystyisi kenttäolosuhteissa tekemään perustestauksen mittalaitetietokoneelle sekä ohjainmoduuleille varmistuakseen niiden toimivuudesta. Lisäksi salkkua oli mahdollista käyttää apuna koulutettaessa huoltomiehille metsäkoneen ohjausjärjestelmän rakennetta sekä sen toimintaa.

Työn tuloksena syntyi testaussalkku, joka keskeisiltä osiltaan täyttää työlle asetetut tavoitteet. Se on riittävän kompakti ja kenttäkelpoinen, jotta se voidaan siirtää huoltoauton mukana metsään, ja ajaa tarvittaessa testit joko salkun avulla tai suoraan metsäkoneessa salkun mukana olevalla testiohjelmalla. Lisäksi salkussa ovat kaikki ohjausjärjestelmän keskeiset komponentit, joita voidaan suoraan käyttää vaihto- tai varaosalaitteina, kun metsäkonetta korjataan. Tällöin turhat varaosien hakumatkat etäälläkin olevilta työkohteilta jäävät pois.

Testaussalkun QA+WIN32-testiohjelma tulostaa testeistä vakioidun raportin, josta mittalaittehuolto pystyy näkemään korjattavaksi tuotavan laitteen tilan ja kokoonpanon, ja samalla varmistamaan että laitteeseen ei ole tehty korjauksia tarvikelosilla. Raportit ovat myös helposti lähetettävissä sähköpostilla tehtaan tekniseen tukeen tai tuotekehitykseen, mikäli tämän kaltaiselle tiedonkeruulle on tarvetta uusien tuotteiden tai vikadiagnostiikan kannalta.

Testaussalkun rakenne ja ohjelmistot on valittu siten, että salkku on käytettävissä suoraan ainakin seuraavan mittalaitesukupolven testauksessa, jolloin sen elinkaari on vähintäänkin seuraavien muutaman vuoden mittainen. Lisäksi tulee huomioida, että tälläkin hetkellä vanhimmat käytössä olevat Ponssen metsäkoneet ovat yli 20 vuotta vanhoja, joten nykyisten koneiden testaustarve pikemminkin kasvaa vielä vähintään seuraavat 5-10 vuotta.

Tärkeänä osana työssä oli myös se, että testaussalkku voitiin tuotteistaa myytäväksi tuotteeksi. Tällöin salkusta on etua sekä sen ostajalle testauksen helpottuessa ja tehostuessa, että myös Ponselle, koska salkkuja myydessä saadaan sen kehitykseen sijoitetulle työajalle ja resursseille korvaus.

LÄHDELUETTELO

Chi Mei Optoelectronics corp. 2009. Model G121X1-L03 tentative specification 9.2.2009. Saatavissa: <http://www.azdisplays.com/PDF/G121X1-L03.pdf>.

Hakupäivä 3.5.2014

Kontron, 2013. ETX-CD User's Guide document revision 130. Saatavissa: <http://www.kontron.com/downloads/manual/mcalm130.pdf?product=86789>

Hakupäivä 3.5.2014

L 14.06.2013/414. Laki puutavaran mittauksesta. Saatavissa:

<http://www.finlex.fi/fi/laki/ajantasa/2013/20130414>. Hakupäivä 18.4.2014.

Metsätalouden kehittämiskeskus Tapio 2008. Tapion taskukirja. 25. painos. Hämeenlinna: Metsäkustannus Oy

AS 17.6.2013/12. Maa- ja metsätalousministeriön asetus puutavaran mittausmenetelmäryhmien ja mittausmenetelmien tarkemmasta sisällöstä sekä mittalaitteiden käytöstä. Saatavissa: <http://www.finlex.fi/data/normit/41198-13012fi.pdf>. Hakupäivä 18.4.2014.

Mueller, S. 2013. Upgrading and repairing PC's. 21. painos. Indianapolis, USA: Pearson Education Inc.

Ponsse Oyj historia. Ponsse Oyj:n kotisivut.

<http://www.ponsse.com/fi/ponsse/historia>. Hakupäivä 20.4.2014

Ponsse Oyj liiketoiminta-alueet. Ponsse Oyj:n kotisivut.

<http://www.ponsse.com/fi/ponsse/liiketoiminta-alueet>. Hakupäivä 20.4.2014

Ponsse Oyj toimiala. Ponsse Oyj:n kotisivut.

<http://www.ponsse.com/fi/ponsse/toimiala>. Hakupäivä 20.4.2014.

Ponsse Oyj tuotekehitys. Ponsse Oyj:n kotisivut.

<http://www.ponsse.com/fi/ponsse/tuotekehitys>. Hakupäivä 20.4.2014.

Ponsse Oyj vuosikertomus 2013. Ponsse Oyj:n kotisivut. Saatavissa:

http://www.ponsse.com/fi/content/download/12930/268189/file/Ponsse%20Oyj_Vuosikertomus%202013.pdf. Hakupäivä 10.4.2014

Uusitalo, J. 2003. Metsäteknologian perusteet. Helsinki: Kustannusosakeyhtiö Metsälehti

Wright, R. 2006. How computers work. 8. painos. Indianapolis, USA: Pearson Education Inc.

LIITTEET

Liite 1. Testaussalkun osaluettelo

Testaussalkku

Osa	Nimike	Määrä	Yks.	Tarvitaan
Kutistesukka 32/8 CPA100 4:1 WITH GLUE	0017297	0,12	m	1,68
Opti6 testcase kaapelisarja	PS1022	1		14
Virtalukko	0056129	1		14
Hub	0067483	1		14
Akku	0055999	1		14
Näppäimistö	0071289	1		14
Hiiri	0071313	1		14
USB-ulkoinen liitin	0071293	2		28
USB-kaapeli liitin-pc	0067850	2		28
Virtapistokkeen kiinnitys: mutterit M3 nyloc	0027098	2		28
Optin aluslevyn kiinnitys lyhyt M8x90	0008567	4		56
Optin aluslevyn kiinnitys pitkä M8x130	0065489	2		28
Moduulien kiinnityslevy/salkku: uppokantapultti M6x25	0043791	2		28
Moduulien kiinnityslevy/Optin aluslevy: pultti M8x20	0065039	2		28
Sulakekiskon kiinnitys: pultit M6x40	0065037	2		28
HUBin kiinnitys: pultit M6x40	0008730	2		28
Hiiren kiinnitysremmi: pultit M8x20	0065039	2		28
Mutteri M3 nyloc	0027098	2		56
Korialuslevy M8	007336	14		196
Korialuslevy M6	0007329	3		42
Aluslaatta M8	0063387	14		196
Mutteri M8	0007492	10		140
Mutteri M8 nyloc	0007484	6		84
Mutteri M6 nylock	0007500	6		84
USB-liittimen kiinnitys: uraruuvi M3x6	0061108	4		56
Optin tuen stopparipultit M4x10	0056334	2		28
Optin tuen stopparimutterit M4	0056247	2		28
Teholähteen kiinnitysruuvi WRONIC	007066	1		14
Salkku	ST038401	1		14
Näppäimistön teline	1630973	1		14
Äänieriste 1mx1,5m pala	0001057	0,32	m ²	4,48
Reunatiiviste	0061022	1,5	m	21
Tarranauha	0068909	0,4	m	5,6
Suojaspiraali 15/18MM	0013430	0,15	m	2,1
Sakkeli 6mm - 1/4"	0025318	1		14

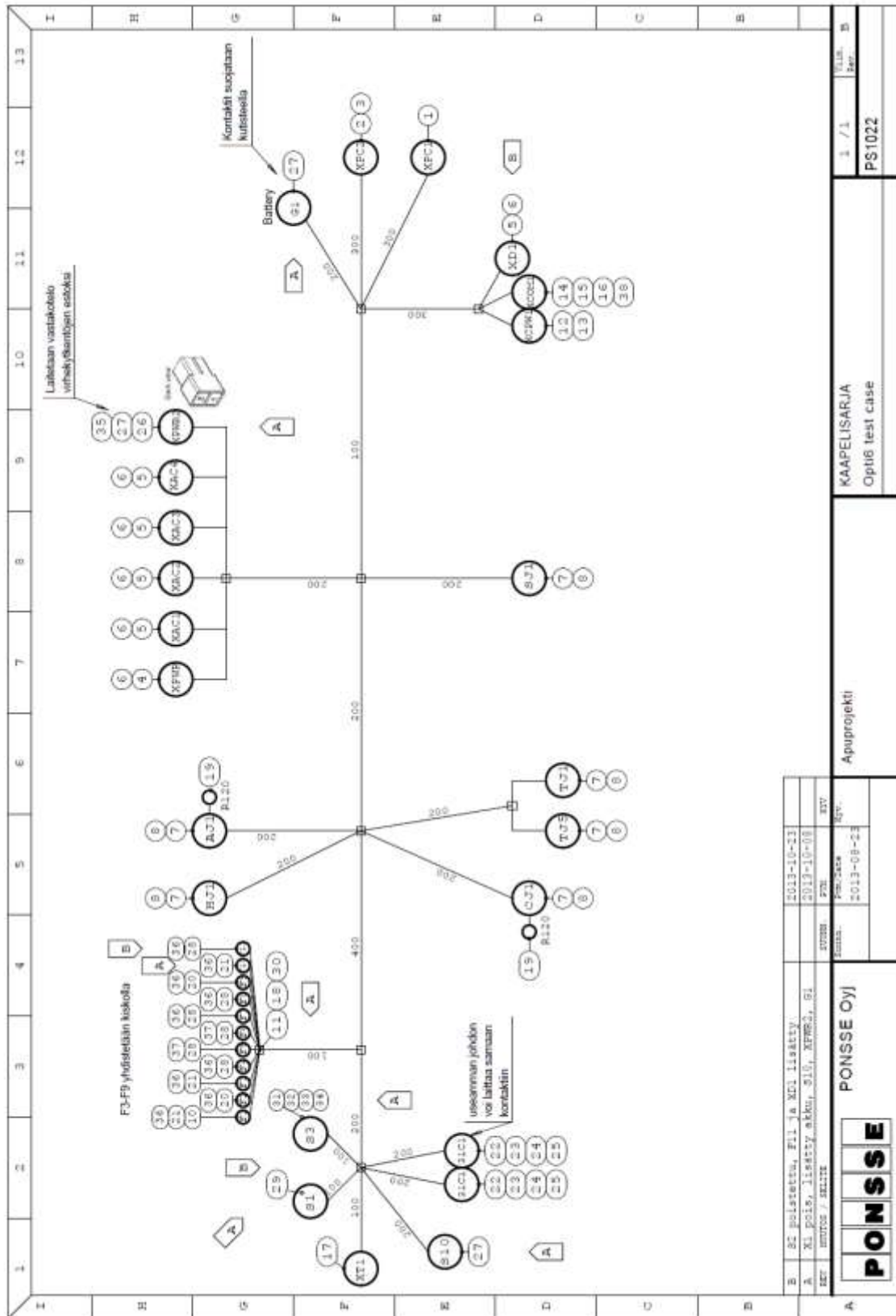
Jälkivarustelulistat: Full ST0384

Osa	Nimike	Määrä
Opti 6	0072002	1
Opti 6 kiinnityslevy	P43123	1
Opti 6 /kiinnityslevy: laippapultti M5x12	0064242	4
Harvesteripäämoduuli	1631723	1
Nosturimoduuli	1631687	1
AV-moduuli	1631729	1
Penkkimoduuli	1631700	1
Ajomoduuli	1631703	1
Masser mittasaksien liitin OPTI6	PS0967	1
Latausyksikkö Caliper 3	0072670	1
USB to serial adapteri	0072047	1
Opti6 programming cable	PS0988	1
Kynät	0072052	1
* Eurosoft -lisenssipaketti + USB-tikku	OK	1
Virtakaapeli 230V	OK	1

Jälkivarustelulistat: Basic ST0385

Osa	Nimike	Määrä
Masser mittasaksien liitin OPTI6	PS0967	1
Latausyksikkö Caliper 3	0072670	1
USB to serial adapteri	0072047	1
Opti6 programming cable	PS0988	1
Kynät	0072052	1
* Eurosoft -lisenssipaketti + USB-tikku	OK	1
Virtakaapeli 230V	OK	1

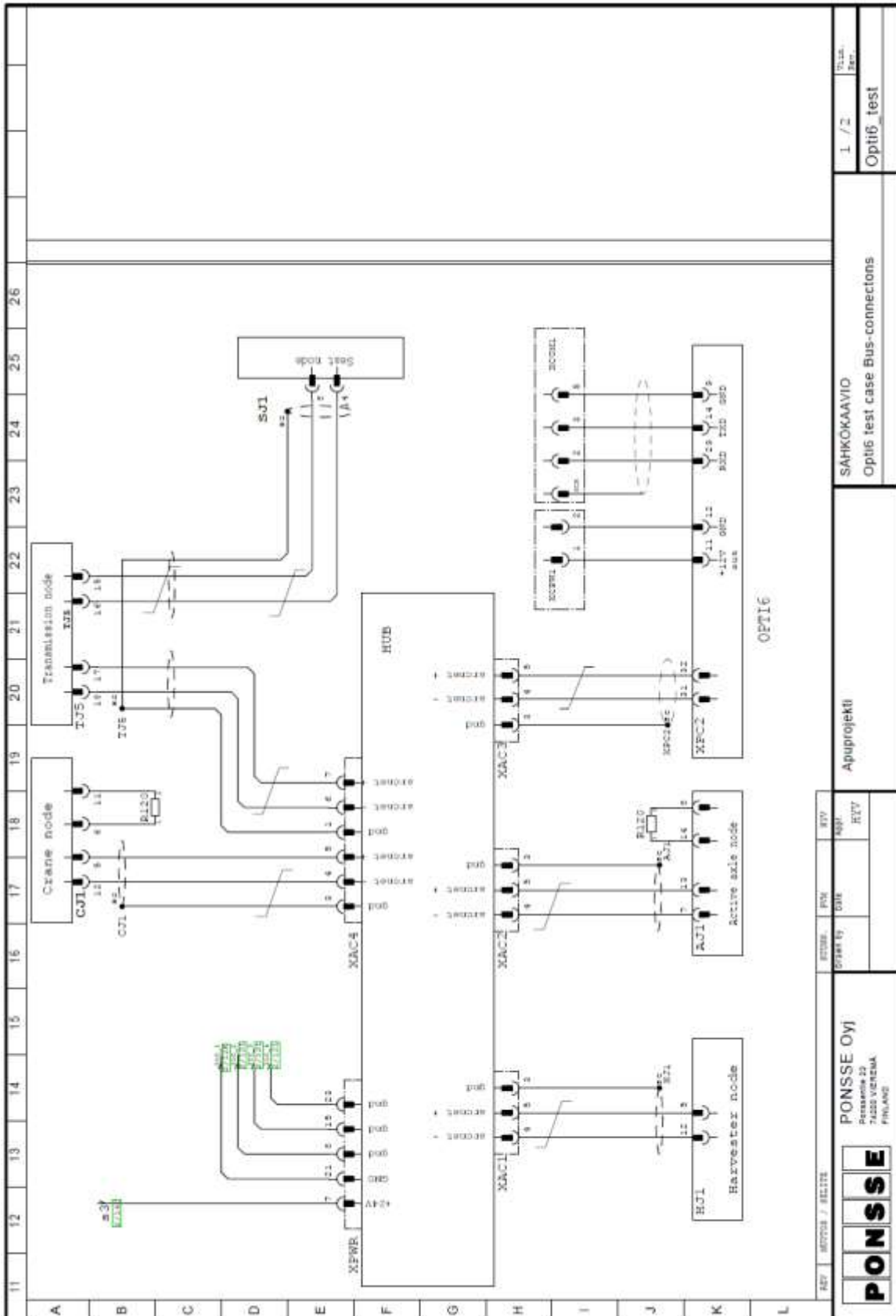
LIITE 2. Testaussalkun johdinsarjakuva ja johdinsarjan osalista



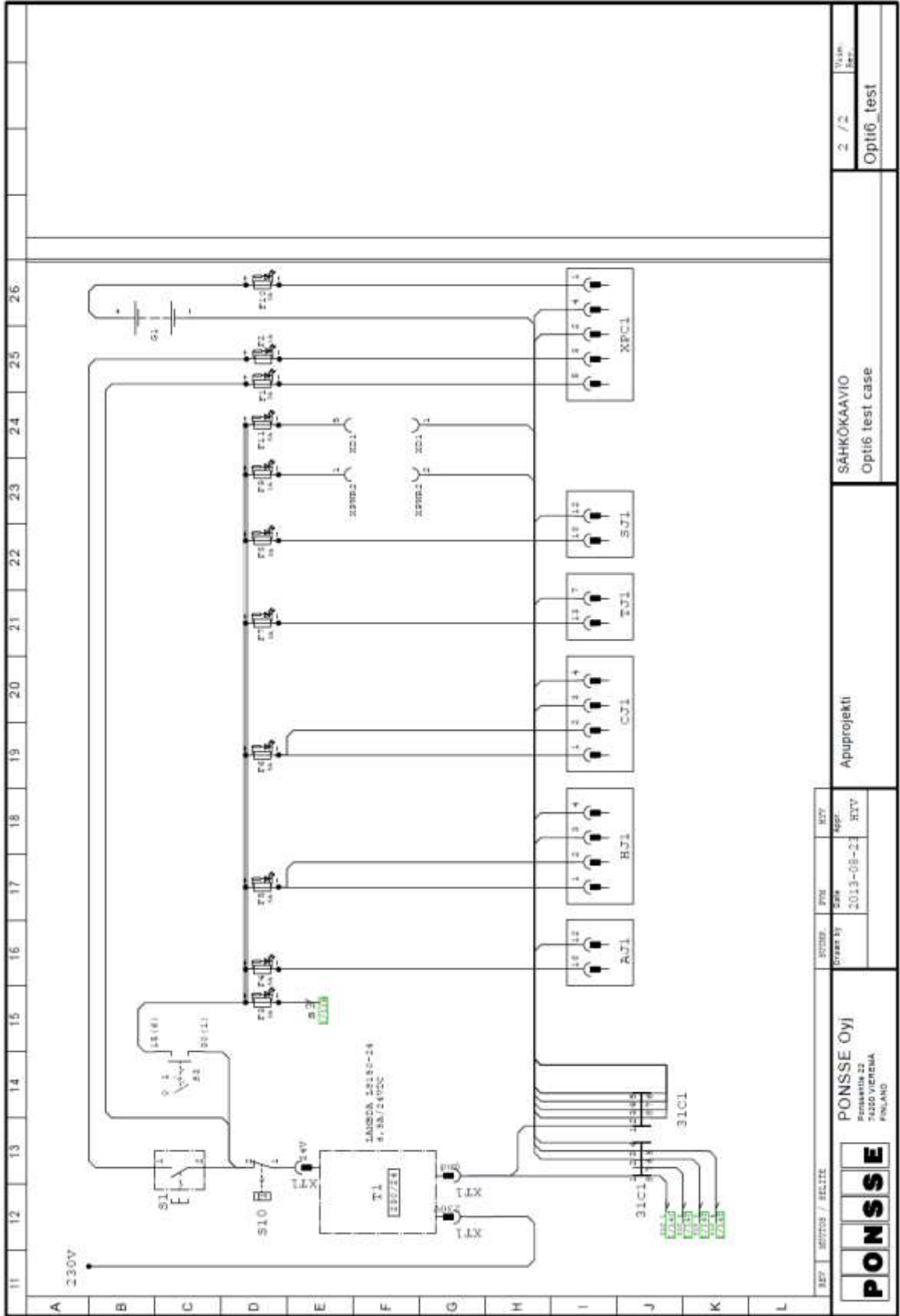
Osa	Nimikentto	Nimitys	Laatu	Kpl	Osa	Nimikentto	Nimitys	Laatu	Kpl
1	0071324	KAAPPELI	ERNI MAXIBRIDGE 254830 WITH 3M A	1	101	0013466	KAAPPELI	1X1,0 RUSKEA R2	16,5
2	0071269	MONINAPALIJTIN	IFD1-16-D-K SAMTEC 32-pole	1	102	0013526	KAAPPELI	1X1,0 YELLOW R2	11,6
3	0089945	KONTAKTI	CC79L-2024-01-L SAMTEC au fem 0.2-0	32	103	0055642	KAAPPELI	1X0,75 R2 RUSKEA	0,6
4	0064220	MONINAPALIJTIN	AMP 770680-4 GRAY23	1	104	0055645	KAAPPELI	1X0,75 R2 KELTAINEN	0,6
5	0064223	MONINAPALIJTIN	AMP 776286-1 BLACK8	5	105	0055715	KAAPPELI	3X0,25 Lvyk	0,6
6	0064224	KONTAKTI	AMP 770520-3 0.5-1,5mm ²	19	106	0071757	KAAPPELI	LEONI L45551-C21-C8	3,6
7	0056176	MONINAPALIJTIN	344106-1 AMP, 18N	6					
8	0056177	KONTAKTI	AMP 175104-2 NAARAS 0.5-2,0mm ²	24					
10	0081155	SULAKEPESÄ	30 01 93 8 UK6-FCV-LED24	11					
11	P33675	ASENNUSKISKO	35mm DIN rail, 150mm	1					
12	0063622	LIITIN	MOLEX 39-01-3042 4-pole	1					
13	0055939	KONTAKTI	MOLEX 39-00-0039 fem AWG18-24	2					
14	0061214	D-LITIN RUNKO	FKEI D-9 HOUSING EMC	1					
15	0055959	D-LITIN	9W MALE	1					
16	0061222	MUTTERI	UNC4-40X5,5 WURTH 453554	2					
17	0017310	PAATEHOLKKI	1,0 YELLOW	1					
18	0061157	KINTEASILTA	FBI10-8 Phoenix 0203263	1					
19	0056659	METALLIKALVOVASTUS	120R	2					
20	0056606	SULAKE	SULAKE 1A W 9999 993 830	2					
21	0025501	SULAKE	FUSE 3A PUDENZ	3					
22	0064737	MONINAPALIJTIN	DT06-08SB-CE05 DEUTSCH	2					
23	0064738	MONINAPALIJTIN	W8S-P012 LOCK DEUTSCH	2					
24	0064723	KONTAKTI	0482-201-16141 SOCKET DEUTSCH, 0,	16					
25	0064347	HAAROTUSELEMENTTI	DT04-8PB-P021 DEUTSCH	2					
26	0060220	LIITIN	055510021 WURTH, double case 6,3mm	621					
27	0060794	JOHTOLIITIN	GHW46040.123.211 FEM 6,3mm/0,5-1m	5					
28	0025492	SULAKE	FUSE 5A HELLA 8JS	6					
29	0056304	KATKAISUJA	MICRO SWITCH 2-POS SOLDERABLE	1					
30	0061159	PAATYPUKISTIN	Phoenix 3022218	2					
31	0056130	SAHKOLIITIN	MERIT 330290 pin nbr 1-4	1					
32	0056131	LIITIN (SAHKÖ)	MERIT 330291 pin nbr 5-10	1					
33	0056198	LIITIN	GHW 46566.201.011 FEM 9,5mm/4-6m	1					
34	0060794	JOHTOLIITIN	GHW46040.123.211 FEM 6,3mm/0,5-1m	1					
35	0062202	LIITIN	AMP 180924-0 / 5551002 double case 6,	1					
36	0017310	PAATEHOLKKI	1,0 YELLOW	13					
37	0017307	PAATEHOLKKI	1,5 RED ERISTETTY	2					
38	0061622	PAATEHOLKKI	6mm ²	1					

 PONSSSE Vuorokki FINLAND	PONSSSE Oyj	Puh.nro 0033-10-23	Apuprojekti	1 / 1 PS102201
		Osa OSALUETTELO PS1022	Puh.nro PS102201	

LIITE 3. Testaussalkun kytkentäkaavio



REV	REVISE / EILITE	REV	REV	1 / 2	1 / 2
DATE	DATE	DATE	DATE	DATE	DATE
DESIGNER	DESIGNER	DESIGNER	DESIGNER	DESIGNER	DESIGNER
CHECKER	CHECKER	CHECKER	CHECKER	CHECKER	CHECKER
APPROVED	APPROVED	APPROVED	APPROVED	APPROVED	APPROVED
PONSSE Oyj Penssola 22 74200 VIEREMA FINLAND			SAHKOKAAVIO Opti6 test case Bus-connections		
Apuprojekti			Opti6_test		



REF	MITTOS / SELITE	MITTOS	PIE	MITT	REF	Var.	Rev.
	PONSSE Oyj	Orasyy	2013-08-23	RVV		2	1/2
	PONSSE Ponsseita 22 70200 VIEREMÄ, FINLAND						
	SAHKÖKAAVIO		Apuprojekti				
	Opti6 test case						
					Opti6 test		

LIITE 4. Esimerkki QA+WIN32-ohjelman testiraportista



Detailed Test Report

Test Results

Tester: Ponsse Machine: Opti6_WES7

Test Item	Result
1000 - Memory	
1 - DDR2 SDRAM (112)	
301 - Quick	PASSED - HNGG05
303 - Walking Bit Left	PASSED - HNGG05
304 - Walking Bit Right	PASSED - HNGG05
307 - Checkerboard	PASSED - HNGG05
308 - Bit Stuck High	PASSED - HNGG05
309 - Bit Stuck Low	PASSED - HNGG05
3500 - Monitor	
1 - Generic Non-PnP Monitor	
301 - Red Purity	PASSED - HNGG05
302 - Green Purity	PASSED - HNGG05
303 - Blue Purity	PASSED - HNGG05
310 - LCD Dead Pixel	PASSED - HNGG05
5500 - Processor	
1 - Intel Mobile Core 2 Duo L7400	
301 - Core Instruction Set	PASSED - HNGG05
302 - Floating Point Instruction Set	PASSED - HNGG05
303 - MMX Instruction Set	PASSED - HNGG05
304 - SSE Instruction Set	PASSED - HNGG05
305 - SSE2 Instruction Set	PASSED - HNGG05
306 - SSE3 Instruction Set	PASSED - HNGG05
5700 - Display Adapter	
1 - Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)	
301 - Linear Memory	PASSED - HNGG05
302 - Microtopology Memory	PASSED - HNGG05
303 - Chaotic Addressing Memory	PASSED - HNGG05
304 - Hardware Acceleration	PASSED - HNGG05
2 - Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)	
6700 - Motherboard	
1 - Kontron MCAL	
301 - Northbridge	PASSED - HNGG05
302 - Southbridge	PASSED - HNGG05
303 - CMOS Clock	PASSED - HNGG05
304 - CMOS Checksum	PASSED - HNGG05
7000 - Solid-State Drive	
1 - ATP Velocity SI Lite 32 GB ATA Device	
301 - Linear Read	PASSED - HNGG05
302 - Random Read	PASSED - HNGG05

Device Information

Memory

DDR2 SDRAM (112)

0x00500001 - Device	1
0x00500002 - DeviceCode	102
0x00500007 - DPC	08 (Memory)
0x00000001 - InstalledMemorySize	1.99 GB

Component 1

0x00000000 - ComponentName	Memory Device
0x00000001 - DeviceCode	8
0x00000002 - MemoryDeviceInformationName	Physical Memory
0x00000003 - MemoryDeviceInformationDeviceLocator	M1
0x00000004 - MemoryDeviceInformationBankLocator	Bank 0
0x00000005 - MemoryDeviceInformationAssetTag	Physical Memory 0
0x00000006 - MemoryDeviceInformationSize	2 GB
0x00000007 - MemoryDeviceInformationFormFactor	12
0x00000008 - MemoryDeviceInformationMemoryType	21
0x00000009 - MemoryDeviceInformationTypeDetail	128
0x0000000A - MemoryDeviceInformationManufacturer	NULL
0x0000000B - MemoryDeviceInformationSerialNumber	NULL
0x0000000C - MemoryDeviceInformationPartNumber	75.A73AR.G04
0x0000000D - MemoryDeviceInformationSpeed	NULL
0x0000000E - SPDHostIndex	0
0x0000000F - SPDChannel	0
0x00000010 - SPDAddress	80
0x00000011 - SPDMemoryType	DDR2
0x00000012 - SPDManufacturerID	Apacer Technology
0x00000013 - SPDSpecification	PC2-5300
0x00000014 - SPDSize	2048 MBytes
0x00000015 - SPDMaxBandwidth	PC2-5300 (333 MHz)
0x00000016 - SPDPartNumber	75.A73AR.G04
0x00000017 - SPDManufacturingDate	Week 14/Year 12
0x00000018 - SPDNumberOfBanks	2
0x00000019 - SPDDataWidth	64 bits
0x0000001A - SPDCorrection	None
0x0000001B - SPDProfileSpec0	3-3-3-9-12 at 200.0 MHz, at 1.8 volts
0x0000001C - SPDProfileSpec1	4-4-4-12-16 at 266.7 MHz, at 1.8 volts
0x0000001D - SPDProfileSpec2	5-5-5-15-20 at 333.3 MHz, at 1.8 volts
0x0000010B - DGR	02012142

Monitor

Generic Non-PnP Monitor

0x00500001 - Device	1
0x00500002 - DeviceCode	101
0x00500003 - PNPDeviceID	NULL
0x00500004 - DeviceID	DesktopMonitor1
0x00000001 - Description	Default Monitor
0x00000002 - Manufacturer	NULL
0x00000003 - ScreenHeight	768
0x00000004 - ScreenWidth	1024

Processor

Intel Mobile Core 2 Duo L7400

Device Information

0x00500001 - Device	1
0x00500002 - DeviceCode	7
0x00500007 - DPC	07 (Processor)
0x00000001 - PhysicalID	0
0x00000002 - Manufacturer	Intel
0x00000003 - ProcessorType	Intel Mobile Core 2 Duo L7400
0x00000004 - Codename	Merom
0x00000005 - Package	Socket 479 mPGA
0x00000006 - Description	Intel(R) Core(TM)2 Duo CPU L7400 @ 1.50GHz
0x00000007 - Family	6
0x00000008 - Model	f
0x00000009 - Stepping	11
0x0000000A - ExtendedFamily	6
0x0000000B - ExtendedModel	f
0x0000000C - Revision	G0
0x0000000D - Technology	65 nm
0x0000000E - BusSpeed	-1.0 MHz
0x0000000F - RatedBusSpeed	-1.0 MHz
0x00000010 - MaxSpeed	1500 MHz
0x00000011 - StockBusSpeed	166 MHz
0x00000012 - CurrentSpeed	1496.3 MHz
0x00000013 - MMXSupported	1
0x00000014 - ExtendedMMXSupported	0
0x00000015 - CoresPerUnit	2
0x00000016 - 3DNowSupported	0
0x00000017 - Extended3DNowSupported	0
0x00000018 - SSESupported	1
0x00000019 - SSE2Supported	1
0x0000001A - SSE3Supported	1
0x0000001B - SSE3Supported	1
0x0000001C - SSE4_1Supported	0
0x0000001D - SSE4_2Supported	0
0x0000001E - SSE4A0Supported	0
0x0000001F - SSE5Supported	0
0x00000020 - X86_64Supported	1
0x00000021 - NXSupported	0
0x00000022 - VMXSupported	1
0x00000023 - HyperThreadingSupported	0
0x00000024 - HyperThreadingEnabled	0
0x00000025 - VirtualTechnologySupported	1
0x00000026 - VirtualTechnologyEnabled	1
0x00000027 - LogicalProcessors	2
0x00000028 - L1DataCache	2 x 32 KBytes
0x00000029 - L1DataCacheAssociativity	8
0x0000002A - L1DataCacheLineSize	64 Bytes
0x0000002B - L1InstructionsCache	2 x 32 KBytes
0x0000002C - L1InstructionsCacheAssociativity	8
0x0000002D - L1InstructionsCacheLineSize	64 Bytes
0x0000002E - L1TraceCache	NULL
0x0000002F - L1TraceCacheAssociativity	NULL
0x00000030 - L2UnifiedCache	4096 KBytes
0x00000031 - L2UnifiedCacheAssociativity	16
0x00000032 - L2UnifiedCacheLineSize	64 Bytes
0x00000033 - L3UnifiedCache	NULL

Device Information

Dx00000034 - L3UnifiedCacheAssociativity	NULL
Dx00000035 - L3UnifiedCacheLineSize	NULL
Dx00000036 - Core0Temperature	26 C (78 F)
Dx00000037 - Core1Temperature	26 C (76 F)

Display Adapter

Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)

Dx00500001 - Device	1
Dx00500002 - DeviceCode	1
Dx00500003 - PNPDeviceID	PCI\VEN_8086&DEV_27A6&SUBSYS_19998086&REV_03\3&42FFD25&0&11
Dx00500007 - DPC	01 (Video)
Dx00000001 - ModelNumber	Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)
Dx00000002 - Memory	NULL
Dx00000003 - BPP	32
Dx00000004 - HorizontalResolution	1024
Dx00000005 - VerticalResolution	768
Dx00000006 - RefreshRate	60
Dx00000007 - DriverDate	20090506000000.000000-000
Dx00000008 - DriverVersion	8.15.10.1749
Dx00000009 - Availability	Running or Full Power
Dx0000000A - CurrentResolution	1024 x 768 x 4294967296 colors
Dx0000000B - VideoProcessor	NULL
Dx0000000C - CodeName	NULL
Dx0000000D - GPUTemperature	NULL
Dx0000000E - MemoryBusWidth	NULL
Dx0000000F - DirectXVersion	11.0
Dx00000010 - ManufacturingProcess	NULL
Dx00000011 - GPUClock	NULL
Dx00000012 - VendorName	Intel Corporation
Dx00000013 - DeviceName	Mobile 945GM/GU Express Integrated Graphics Controller
Dx00000014 - SubVendorID	0x8086
Dx00000015 - SubDeviceID	0x1999
Dx00000016 - Revision	0x03
Dx00000017 - DXSHardwareSupported	0
Dx00000018 - DXSSoftwareSupported	1

Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)

Dx00500001 - Device	2
Dx00500002 - DeviceCode	1
Dx00500003 - PNPDeviceID	PCI\VEN_8086&DEV_27A6&SUBSYS_19998086&REV_03\3&42FFD25&0&10
Dx00500007 - DPC	01 (Video)
Dx00000001 - ModelNumber	Mobile Intel(R) 945 Express Chipset Family (Microsoft Corporation - WDDM 1.0)
Dx00000002 - Memory	256 MB
Dx00000003 - BPP	NULL
Dx00000004 - HorizontalResolution	NULL
Dx00000005 - VerticalResolution	NULL
Dx00000006 - RefreshRate	NULL
Dx00000007 - DriverDate	20090506000000.000000-000
Dx00000008 - DriverVersion	8.15.10.1749
Dx00000009 - Availability	Off Line
Dx0000000A - CurrentResolution	NULL
Dx0000000B - VideoProcessor	Intel(R) GMA 950
Dx0000000C - CodeName	NULL
Dx0000000D - GPUTemperature	NULL
Dx0000000E - MemoryBusWidth	NULL

Device Information

0x0000000F - DirectXVersion	NULL
0x00000010 - ManufacturingProcess	NULL
0x00000011 - GPUClock	NULL
0x00000012 - VendorName	Intel Corporation
0x00000013 - DeviceName	Mobile 945 Express Chipset
0x00000014 - SubVendorID	0x8086
0x00000015 - SubDeviceID	0x1999
0x00000016 - Revision	0x03
0x00000017 - DXSHardwareSupported	0
0x00000018 - DXSoftwareSupported	0

Motherboard

Kontron MCAL

0x00500001 - Device	1
0x00500002 - DeviceCode	5
0x00500006 - BaseboardSerialNum	ZMD790110
0x00500007 - DPC	05 (Motherboard)
0x00500008 - Name	Ver 1.00PARTTBL
0x00500009 - MajorVersion	2
0x0050000A - MinorVersion	4
0x0050000B - WMIVersion	PTLTD - 6040000
0x0050000C - Serial#	N/A
0x00000001 - SystemInformationManufacturer	N/A
0x00000002 - SystemInformationProductName	N/A
0x00000003 - SystemInformationSerialNumber	N/A
0x00000004 - SystemInformationVersion	N/A
0x00000005 - SystemInformationUuidInRegFormat	{00000000-0000-0000-0000-000000000000}
0x00000006 - SystemInformationUuid	0x00000000000000000000000000000000
0x00000007 - BaseboardInformationManufacturer	Kontron
0x00000008 - BaseboardInformationProduct	MCAL
0x00000009 - BaseboardInformationVersion	0114
0x0000000A - ChassisInformationType	NULL
0x0000000B - ChassisInformationDescription	System Enclosure
0x0000000C - ChassisInformationLock	0
0x0000000D - ChassisInformationManufacturer	No Enclosure
0x0000000E - ChassisInformationName	System Enclosure
0x0000000F - ChassisInformationSerialNumber	None
0x00000010 - ChassisInformationAssetTag	No Asset Tag
0x00000011 - ChassisInformationTag	System Enclosure 0
0x00000012 - ChassisInformationVersion	N/A
0x00000013 - BiosInformationVendor	Phoenix Technologies LTD
0x00000014 - BiosInformationVersion	MCALR118
0x00000015 - BiosInformationDate	20090928000000.000000+000
0x00000016 - PCIslotCount	-1
0x00000017 - DiskDriveCount	2
0x00000018 - VideoController	Intel Corporation Mobile 945GM/GU Express Integrated Graphics Controller
0x00000019 - VideoController	Intel Corporation Mobile 945 Express Chipset
0x0000001A - IDEController	Intel Corporation 82801GBM/GHM Serial ATA Storage Controller
0x00000024 - TPMPresent	0
0x00000025 - TPMActivated	0
0x00000026 - TPMEEnabled	0

Solid-State Drive

ATP Velocity SI Lite 32 GB ATA Device

Device Information

0x00500001 - Device	1
0x00500002 - DeviceCode	70
0x00500003 - PNPDeviceID	IDEIDISKATP_VELOCITY_S_LITE_32_GB_____L0412C___15874D41048080.0.0
0x00500004 - DeviceID	\\.\PHYSICALDRIVE1
0x00500007 - DPC	70 (Solid-State Drive)
0x00000001 - Description	Disk drive
0x00000002 - Capacity	32 GB
0x00000003 - InterfaceType	IDE
0x00000004 - Manufacturer	(Standard disk drives)
0x00000005 - SerialNumber	99026130413231500130
0x00000006 - Size	31658360832

This system has been tested using QA+Win™ test software from Eurosoft (UK) Ltd

© Copyright 1997-2012, Eurosoft US Inc. - All Rights Reserved

Phone - +44 (0)1202 297315, Fax - +44 (0)1202 558280

www.eurosoft-uk.com

Activation Name: Ponsse_*) Licenses: 15 en-gb

Page 6 - 03 February 2014 12:15:52

LIITE 5. Pikatestin XML-koodi

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Script RunType="LapCount" RunCount="1" StopOnFail="0">
3   <Group Group="1000" GroupName="Memory">
4     <Device Device="1">
5
6       <Test Test="301" TestName="Quick">
7         <TestParams>
8           <TestParam ID="1" Name="Duration" Value="60">
9             </TestParam>
10          <TestParam ID="2" Name="Coverage" Value="100">
11            </TestParam>
12          <TestParam ID="3" Name="EnableCaching" Value="1">
13            </TestParam>
14        </TestParams>
15      </Test>
16      <Test Test="303" TestName="Walking Bit Left">
17        <TestParams>
18          <TestParam ID="1" Name="Duration" Value="60">
19            </TestParam>
20          <TestParam ID="2" Name="Coverage" Value="100">
21            </TestParam>
22          <TestParam ID="3" Name="EnableCaching" Value="1">
23            </TestParam>
24        </TestParams>
25      </Test>
26      <Test Test="304" TestName="Walking Bit Right">
27        <TestParams>
28          <TestParam ID="1" Name="Duration" Value="60">
29            </TestParam>
30          <TestParam ID="2" Name="Coverage" Value="100">
31            </TestParam>
32          <TestParam ID="3" Name="EnableCaching" Value="1">
33            </TestParam>
34        </TestParams>
35      </Test>
36      <Test Test="307" TestName="Checkerboard">
37        <TestParams>
38          <TestParam ID="1" Name="Duration" Value="60">
39            </TestParam>
40          <TestParam ID="2" Name="Coverage" Value="100">
41            </TestParam>
42          <TestParam ID="3" Name="EnableCaching" Value="1">
43            </TestParam>
44        </TestParams>
45      </Test>
46      <Test Test="308" TestName="Bit Stuck High">
47        <TestParams>
48          <TestParam ID="1" Name="Duration" Value="60">
49            </TestParam>
50          <TestParam ID="2" Name="Coverage" Value="100">
51            </TestParam>
52          <TestParam ID="3" Name="EnableCaching" Value="1">
53            </TestParam>
54        </TestParams>
55      </Test>
56      <Test Test="309" TestName="Bit Stuck Low">
57        <TestParams>
58          <TestParam ID="1" Name="Duration" Value="60">
59            </TestParam>
60          <TestParam ID="2" Name="Coverage" Value="100">
61            </TestParam>
62          <TestParam ID="3" Name="EnableCaching" Value="1">
63            </TestParam>
64        </TestParams>
65      </Test>
66    </Device>

```

```

67 </Group>
68 <Group Group="3500" GroupName="Monitor">
69 <Device Device="1">
70 <Test Test="301" TestName="Red Purity">
71 <TestParams>
72 <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
73 </TestParams>
74 </Test>
75 <Test Test="302" TestName="Green Purity ">
76 <TestParams>
77 <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
78 </TestParams>
79 </Test>
80 <Test Test="303" TestName="Blue Purity ">
81 <TestParams>
82 <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
83 </TestParams>
84 </Test>
85 <Test Test="310" TestName="LCD Dead Pixel">
86 <TestParams>
87 <TestParam ID="1" Name="DisplayInitialTestMessage" Value="0"></TestParam>
88 <TestParam ID="2" Name="ScreenPeriod" Value="5"></TestParam>
89 </TestParams>
90 </Test>
91 </Device>
92 </Group>
93 <Group Group="5500" GroupName="Processor">
94 <Device Device="1">
95 <Test Test="301" TestName="Core Instruction Set">
96 <TestParams></TestParams>
97 </Test>
98 <Test Test="302" TestName="Floating Point Instruction Set">
99 <TestParams></TestParams>
100 </Test>
101 <Test Test="303" TestName="MMX Instruction Set">
102 <TestParams></TestParams>
103 </Test>
104 <Test Test="304" TestName="SSE Instruction Set">
105 <TestParams></TestParams>
106 </Test>
107 <Test Test="305" TestName="SSE2 Instruction Set">
108 <TestParams></TestParams>
109 </Test>
110 <Test Test="306" TestName="SSE3 Instruction Set">
111 <TestParams></TestParams>
112 </Test>
113 </Device>
114 </Group>
115 <Group Group="5700" GroupName="Display Adapter">
116 <Device Device="1">
117 <Test Test="301" TestName="Linear Memory ">
118 <TestParams></TestParams>
119 </Test>
120 <Test Test="302" TestName="Microtopology Memory">
121 <TestParams>
122 <TestParam ID="1" Name="Duration" Value="60"></TestParam>
123 </TestParams>
124 </Test>
125 <Test Test="303" TestName="Chaotic Addressing Memory">
126 <TestParams>
127 <TestParam ID="1" Name="Duration" Value="60"></TestParam>
128 </TestParams>
129 </Test>
130 <Test Test="304" TestName="Hardware Acceleration">
131 <TestParams>
132 <TestParam ID="1" Name="Duration" Value="60"></TestParam>

```

```

133     </TestParams>
134 </Test>
135 </Device>
136 </Group>
137 <Group Group="6700" GroupName="Motherboard">
138 <Device Device="1">
139   <Test Test="301" TestName="Northbridge">
140     <TestParams>
141       <TestParam ID="1" Name="Duration" Value="30"></TestParam>
142       <TestParam ID="2" Name="FSBSpeed" Value="500"></TestParam>
143       <TestParam ID="6" Name="FSBTolerance" Value="50"></TestParam>
144     </TestParams>
145   </Test>
146   <Test Test="302" TestName="Southbridge">
147     <TestParams></TestParams>
148   </Test>
149   <Test Test="303" TestName="CMOS Clock">
150     <TestParams></TestParams>
151   </Test>
152   <Test Test="304" TestName="CMOS Checksum">
153     <TestParams></TestParams>
154   </Test>
155 </Device>
156 </Group>
157 <Group Group="7000" GroupName="Solid-State Drive">
158 <Device Device="1">
159   <Test Test="301" TestName="Linear Read">
160     <TestParams>
161       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
162       <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
163     </TestParams>
164   </Test>
165   <Test Test="302" TestName="Random Read">
166     <TestParams>
167       <TestParam ID="1" Name="Duration" Value="60"></TestParam>
168     </TestParams>
169   </Test>
170 </Device>
171 </Group>
172 </Script>

```


LIITE 6. Rasiustestin XML-koodi

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Script RunType="LapCount" RunCount="40" StopOnFail="0">
3   <Group Group="1000" GroupName="Memory">
4     <Device Device="1">
5       <Test Test="301" TestName="Quick">
6         <TestParams>
7           <TestParam ID="1" Name="Duration" Value="60"></TestParam>
8           <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
9           <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
10        </TestParams>
11      </Test>
12      <Test Test="303" TestName="Walking Bit Left">
13        <TestParams>
14          <TestParam ID="1" Name="Duration" Value="60"></TestParam>
15          <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
16          <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
17        </TestParams>
18      </Test>
19      <Test Test="304" TestName="Walking Bit Right">
20        <TestParams>
21          <TestParam ID="1" Name="Duration" Value="60"></TestParam>
22          <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
23          <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
24        </TestParams>
25      </Test>
26      <Test Test="307" TestName="Checkerboard">
27        <TestParams>
28          <TestParam ID="1" Name="Duration" Value="60"></TestParam>
29          <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
30          <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
31        </TestParams>
32      </Test>
33      <Test Test="308" TestName="Bit Stuck High">
34        <TestParams>
35          <TestParam ID="1" Name="Duration" Value="60"></TestParam>
36          <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
37          <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
38        </TestParams>
39      </Test>
40      <Test Test="309" TestName="Bit Stuck Low">
41        <TestParams>
42          <TestParam ID="1" Name="Duration" Value="60"></TestParam>
43          <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
44          <TestParam ID="3" Name="EnableCaching" Value="1"></TestParam>
45        </TestParams>
46      </Test>
47    </Device>
48  </Group>
49  <Group Group="5500" GroupName="Processor">
50    <Device Device="1">
51      <Test Test="301" TestName="Core Instruction Set">
52        <TestParams></TestParams>
53      </Test>
54      <Test Test="302" TestName="Floating Point Instruction Set">
55        <TestParams></TestParams>
56      </Test>
57      <Test Test="303" TestName="MMX Instruction Set">
58        <TestParams></TestParams>
59      </Test>
60    </Device>
61  </Group>
62  <Group Group="5700" GroupName="Display Adapter">
63    <Device Device="1">
64      <Test Test="301" TestName="Linear Memory ">
65        <TestParams></TestParams>
66      </Test>
```

```

67     <Test Test="302" TestName="Microtopology Memory">
68         <TestParams>
69             <TestParam ID="1" Name="Duration" Value="60"></TestParam>
70         </TestParams>
71     </Test>
72     <Test Test="303" TestName="Chaotic Addressing Memory">
73         <TestParams>
74             <TestParam ID="1" Name="Duration" Value="60"></TestParam>
75         </TestParams>
76     </Test>
77     <Test Test="304" TestName="Hardware Acceleration">
78         <TestParams>
79             <TestParam ID="1" Name="Duration" Value="60"></TestParam>
80         </TestParams>
81     </Test>
82 </Device>
83 </Group>
84 <Group Group="6700" GroupName="Motherboard">
85     <Device Device="1">
86         <Test Test="301" TestName="Northbridge">
87             <TestParams>
88                 <TestParam ID="1" Name="Duration" Value="30"></TestParam>
89                 <TestParam ID="2" Name="FSBSpeed" Value="664"></TestParam>
90                 <TestParam ID="6" Name="FSBTolerance" Value="30"></TestParam>
91             </TestParams>
92         </Test>
93         <Test Test="302" TestName="Southbridge">
94             <TestParams></TestParams>
95         </Test>
96         <Test Test="303" TestName="CMOS Clock">
97             <TestParams></TestParams>
98         </Test>
99         <Test Test="304" TestName="CMOS Checksum">
100             <TestParams></TestParams>
101         </Test>
102     </Device>
103 </Group>
104 <Group Group="7000" GroupName="Solid-State Drive">
105     <Device Device="1">
106         <Test Test="301" TestName="Linear Read">
107             <TestParams>
108                 <TestParam ID="1" Name="Duration" Value="60"></TestParam>
109                 <TestParam ID="2" Name="Coverage" Value="100"></TestParam>
110             </TestParams>
111         </Test>
112         <Test Test="302" TestName="Random Read">
113             <TestParams>
114                 <TestParam ID="1" Name="Duration" Value="60"></TestParam>
115             </TestParams>
116         </Test>
117     </Device>
118 </Group>
119 </Script>

```