

Teemu Pyykkönen

# OLOSUHDEKAAPIN KÄYTTÖÖNOTTO JA TESTAUSJÄRJESTELMÄN TOTEUTUS

Insinööri (AMK),

tietotekniikka

Kevät 2015



KAJAANIN  
AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES

## TIIVISTELMÄ

**Tekijä(t):** Pyykkönen Teemu

**Työn nimi:** Olosuhdekaapin käyttöönotto ja testausjärjestelmän toteutus

**Tutkintonimike:** Insinööri (AMK), tietotekniikka

**Asiasanat:** Olosuhdetestaus, Olosuhdekaappi, Testausjärjestelmä, C++

Tämän insinöörityön tavoitteena oli toteuttaa testausjärjestelmä Tieto-Oskari Oy:n valmistamalle tuotteelle. Testausjärjestelmän tuli pystyä käynnistämään testattavan tuotteen itsediagnostiikkatoiminto, tallentaa sen tulokset ja esittää ne graafisesti käyttäjälle. Järjestelmän käyttöalustana toimi Windows, ja se toteutettiin C++:lla. Tietoliikenne tapahtui sarjaliikenteenä pc:n ja testattavan tuotteen välillä käyttäen USB-RS232-muunninta ja valmista sarjaliikennekirjastoa. Ohjelman kehitysympäristönä käytettiin Borland C++Builderia.

Testausjärjestelmän toteuttamisen lisäksi työssä perehdyttiin olosuhdekaapin käyttämiseen ja olosuhdetestaukseen. Olosuhdekaapin käyttöön perehtymisen tarkoituksena oli madaltaa käyttöönottamisen kynnystä. Perehtymisen aikana kerätystä tiedosta laadittiin lyhytmuotoiset käyttöohjeet.

Työn lopputuloksena oli valmis testausjärjestelmä, jonka toimivuus testattiin olosuhdekaapilla toteutettujen olosuhdetestien yhteydessä. Testausjärjestelmää voidaan hyödyntää tuotannon testaamisessa jatkossa.

## ABSTRACT

**Author(s):** Pyykkönen Teemu

**Title of the Publication:** Commissioning of an Environmental Chamber and Developing Testing Software

**Degree Title:** Engineer, Information technology

**Keywords:** Environmental testing, Environmental chamber, Testing software, C++

The aim of this thesis was to develop testing software for a product which is manufactured by Tieto-Oskari Oy. The testing software had to be able to start a diagnosis feature of the product, save its results and present them graphically to the user. The platform on which the software was developed was Windows and it was programmed in C++. The serial communication between the product and pc was implemented by using USB-RS232-converter and serial communication library. The integrated development environment used in this thesis was Borland C++Builder.

In addition to developing testing software for Tieto-Oskari's product, further work was carried out by commissioning an environmental chamber and introducing it for environmental testing. This is disclosed in the second part of this thesis in detail. The aim of the commissioning was to make it easier for Tieto-Oskari to use their environmental chamber in carrying out any required testing. These tests take place in the final stages of the production. A short manual about environmental chamber was produced from the information which was gathered during the commissioning.

The final result of this thesis was a finished testing software. The functionality of this software was confirmed during environmental tests which were produced by the environmental chamber. The software can be used in tests which take place in production stages in the future.

## ALKUSANAT

Haluan kiittää Tieto-Oskari Oy:ltä Taito Heikkistä insinööriyöni aiheesta, Marko Kemppaista työni valvomisesta, sekä Sami Pyykköstä, Jukka Pöppöstä, Harri Hanhisaloa ja Ari-Antti Tervosta työn ohjeistuksesta. Lisäksi haluan kiittää Kajaanin ammattikorkeakoululta Asko Kinnusta työn valvomisesta ja Eero Soinista työn kielellisestä ohjauksesta.

## SISÄLLYS

1 JOHDANTO.....	1
2 LUOTETTAVUUS.....	2
3 TESTAUS.....	3
4 OLOSUHDETESTAUS.....	4
4.1 Olosuhteiden haittavaikutukset.....	4
4.2 Olosuhdetestien käyttötavat .....	6
4.3 IEC60068-2-1 Kylmätesti .....	7
4.4 IEC60068-2-14 Lämpötilan vaihtelu .....	9
5 OLOSUHDEKAAPPI .....	12
5.1 Olosuhdekaapin ominaisuudet .....	12
5.2 Winkratos-ohjelma.....	16
6 TESTAUSOHJELMAN SUUNNITTELU .....	18
6.1 Testausohjelman vaatimukset.....	18
6.2 Testausohjelman toiminnot ja ominaisuudet .....	19
6.3 Testin rakenteen määrittely .....	20
6.4 C++Builder ja TComPort .....	21
7 TYÖN SUORITUS.....	22
7.1 Käyttöönotto .....	22
7.2 Ohjelmointi .....	23
7.3 Testaus .....	27
8 TULOSTEN ANALYSOINTI.....	29
9 YHTEENVETO .....	30
LÄHTEET .....	31

## SYMBOLILUETTELO

ATE	Automatic Test Equipment
CSV	Comma-Separated Values
DFT	Design For Testability
PLC	Programmable Logic Controller
RAD	Rapid Application Development
RH	Relative Humidity
RS232	Recommended Standard 232
USB	Universal Serial Bus

## 1 JOHDANTO

Nykypäivänä elektroniset laitteet ovat osana arkista elämäämme ja niiden määrä kasvaa koko ajan. Vaikka käyttötarkoitukset vaihtelevat laajasti, on niillä kaikilla kuitenkin yhteinen piirre. Käyttäjä olettaa laitteen toimivan luotettavasti käyttöympäristöstä riippumatta.

Markkinoiden globalisoitumisen myötä elektroniikkalaitteita saatetaan käyttää sellaisissa ympäristöissä tai olosuhteissa, joihin niitä ei ole suunniteltu. Tämä aiheuttaa suuria haasteita laitesuunnitteluun, koska kaikkia laitteeseen vaikuttavia tekijöitä ei voida ottaa huomioon. Toisaalta, vaikka elektroniikkalaitteita olisikin huolella suunniteltu, ei tuotanto ole ikinä täysin virheetöntä. Tämän takia on kehitetty testausmenetelmiä, jotta voidaan varmistua elektroniikkalaitteen toimivuudesta ennen, kuin laite päätyy käyttäjälle [1, s. 3].

Tieto-Oskari Oy on kajaanilainen yritys, joka on perustettu vuonna 1991. Tieto-Oskari sekä suunnittelee että valmistaa elektroniikkaa ja ohjelmistoja. Tuotteita yritys valmistaa niin yksityishenkilöiden, yritysten kuin teollisuuden käyttöön. [2.]

Yhteistyökumppaneita ja asiakkaita Tieto-Oskarilla on 32:ssa eri maassa, joten laitteiden käyttöolosuhteet voivat vaihdella huomattavasti. Lisäksi monet Tieto-Oskarin asiakkaat toimivat teollisuudessa tai käyttävät yrityksen tuotteita vaikeissa olosuhteissa. Tästä syystä laadun ylläpito on Tieto-Oskarille tärkeää ja yritys on aiemmin hankkinut olosuhdekaapin tuotannon testausta varten.

Tämän insinööriyön tavoitteena oli kehittää testausjärjestelmä Tieto-Oskarin valmistamalle tuotteelle. Testausjärjestelmän tarkoituksena on kerätä tietoa olosuhdekaapissa testattavista tuotteista ja esittää tulokset käyttäjälle. Testausjärjestelmä toteutettiin pc:lle. Tämän lisäksi insinööriyössä oli tavoitteena perehtyä olosuhdekaapin toimintaan ja madaltaa kynnystä ottaa se käyttöön tuotantovaiheen testaukseen.

## 2 LUOTETTAVUUS

Kuten kaikilla muillakin laitteilla, elektroniikkalaitteilla on oma käyttöelinkaari. Käytöstä aiheutuva vanheneminen luonnollisesti lyhentää tätä elinkaarta, mutta toisin kuin monesti ajatellaan, kulumista tapahtuu myös silloin, kun laite ei ole käytössä.

Vanhenemisen nopeus on vahvasti sidoksissa elektroniikkalaitteeseen kohdistuviin olosuhteisiin ja rasituksiin. Onkin vain ajan kysymys, milloin elektroniikkalaitteeseen syntyy vikoja. Käyttäjän kannalta tämä on ongelmallista, koska viat eivät välttämättä ole silmin havaittavissa ja voivat paljastua käytössäkin vasta tiettyjen olosuhteiden vallitessa. Näin laitteen kunnosta ei monesti ole tietoa ja käyttäjä joutuu luottamaan valmistajaan tai jälleenmyyjään. Tästä syystä luotettavuus on ehdottomasti yksi tärkeimmistä arvoista elektroniikkalaitteen käyttäjälle. Se ilmaisee käyttäjälle olosuhteet, missä elektroniikkalaitte toimii, sekä ajan, miten kauan laitteen voidaan olettaa toimivan noissa olosuhteissa [3, s. 19].

Erilaiset olosuhteet ja käyttöympäristöt edellyttävät eritasoista luotettavuutta elektroniikkalaitteelta. Kuluttajan käyttämään elektroniikkalaitteeseen kohdistuu huomattavasti pienemmät oletukset luotettavuuden suhteen kuin teollisuudessa käytettävään. Näin ollen mitä vaativammaksi käyttöympäristöt muuttuvat, sen tärkeämmäksi arvoksi luotettavuus kasvaa.

Luotettavuus voi olla myös ratkaiseva tekijä päätettäessä, mitä elektroniikkalaitetta käytetään tai mikä ostetaan. Jos jonkin yrityksen myymässä tuotteessa ilmenee vikoja ennenaikaisesti, joutuu yritys yleensä korvaamaan ne uudella tuotteella tai tuotteen arvon rahallisesti. Viallisen tuotteen aiheuttamat vahingot eivät yleensä jää taloudellisiksi tai materiaalisiksi, vaan samalla voi kärsiä yrityksen imago [1, s. 3]. Vahinkojen välttämiseksi on tärkeää sekä parantaa että pitää yllä elektroniikkalaitteiden luotettavuutta. Testaamisessa pyritään tähän.



### 3 TESTAUS

Testauksen päätarkoituksena on varmistaa, että testattava laite täyttää oman laitespesifikaationsa. Yhtä tärkeänä tarkoituksena voidaan kuitenkin myös pitää mahdollisten piilovikojen ja toiminnallisten häiriöiden paikantamista. [3, s. 177.]

Suurin osa vioista saadaan karsittua pois yleensä suunnittelu- ja testausvaiheessa. Osa saattaa kuitenkin paljastua vasta tuotannon loppupäässä tai laitteen käytön aikana. Uusia vikoja voi myös syntyä tuotannon aikana, jos esim. laitteen valmistustapa muuttuu, komponenttien toleranssit ovat odotettua suurempia ja komponenttien tyypit tai valmistajat muuttuvat. [3, s. 177.]

Ei siis voida olettaa, että tuotantovaiheessa olevat laitteet toimisivat automaattisesti yhtä hyvin kuin suunnitteluvaiheen aikana kasattu laite. Testaavan henkilön tulisikin olla tietoinen mahdollisista muutoksista laitteen valmistuksessa, jotta syy toiminnallisuuden muuttumiseen voidaan jäljittää. Näin testattavuuden ylläpito helpottuu. [3, s. 177.]

#### Testattavuus

Jotta tuotannon ja huollon yhteydessä tapahtuva testaaminen olisi mahdollisimman tehokasta, tulee elektroniikkalaitteen olla suunniteltu sitä varten. Testattavuudella (design for testability, DFT) tarkoitetaan testauksen huomioon ottamista suunnitteluvaiheessa ja lisäämällä laitesuunnitelmaan testaamista tehostavia ominaisuuksia. Testattavuus toteutuu suunnittelussa esim. sijoittamalla mittauspisteet tärkeille signaali- ja jännitetasoille sekä komponenteille tai toteuttamalla laitteelle itsediagnostiikkatoimintoja. [3, s. 177.]

Testattavuudesta on tullut yhä tärkeämpää elektroniikkalaitteiden rakenteiden laajenemisen ja monimutkaistumisen myötä. Niille suunnatut automatisoidut testausjärjestelmät (automatic test equipment, ATE) lisäävät tarvetta testattavuuden huomioon ottamiselle entisestään. [3, s. 177.]

## 4 OLOSUHDETESTAUS

Olosuhdetestit ovat standardeissa määriteltäviä testausmenetelmiä, joilla voidaan todentaa elektroniikkalaitteen toiminta tiettyjen olosuhteiden vallitessa ja mallintaa olosuhteiden vaikutuksia laitteeseen. Testien tavoitteena ei siis ole matkia käyttöympäristön olosuhteita, vaan toistaa ympäristöolosuhteiden vaikutuksia [4, s. 7].

Olosuhdetestit voidaan jakaa kahteen pääryhmään testissä vaikuttavien olosuhteiden perusteella. Ensimmäiseen ryhmään kuuluvat testit, joissa elektroniikkalaitteeseen vaikuttavat tekijät ovat mekaanisia, kuten esim. iskut, värinät, kiihtyvyydet, korkeat äänet ja kovat tuulenpuuskat. Toiseen ryhmään kuuluvat puolestaan testit, joissa elektroniikkalaitteeseen vaikuttavat tekijät aiheutuvat ilmastosta. Näitä ovat esim. lämpötila, kosteus, paine, sade, suola, kaasu tai auringon säteily. Lisäksi on olemassa olosuhdetestejä, joissa esiintyy molempien pääryhmien olosuhteita. [1, s. 3.]

Olosuhdetestejä käytetään sekä tuotteen kehitysvaiheessa että tuotannon lopussa. Kehitysvaiheessa tapahtuvalla testauksella pyritään löytämään heikkoudet laitteen suunnittelussa sekä käyttörajat. Kehitysvaiheessa testataan myös monesti yksittäisiä komponentteja niiden soveltuvuuden selvittämiseksi. Tuotantovaiheessa tapahtuvalla testauksella pyritään varmistamaan tuotteen toimivuus ja paikantamaan laitteiden valmistus- ja satunnaisvikoja sekä karsimaan pois heikot yksilöt. [5, s. 38.]

### 4.1 Olosuhteiden haittavaikutukset

Eri olosuhteilla on myös erilaiset haittavaikutukset niin laitteiden sähköisiin ominaisuuksiin kuin rakenteisiin. Tästä syystä on tärkeää tuntee käyttöympäristössä esiintyvät olosuhteet ja niiden vaikutukset. Olosuhteet ja niiden aiheuttamat vaikutukset on esitelty taulukossa 1.

Taulukko 1. Olosuhteiden haittavaikutukset [1, s. 4] [6, s. 67]

Olosuhde	Vaikutukset
Tuulet ja turbulenssit	Rakenteiden rappioituminen ja hajoaminen, pintojen ja osien jäähtyminen pienillä tuulen nopeuksilla, lämmön syntyminen ilmanvastuksesta suurilla tuulen nopeuksilla, toiminnallisuuden pettäminen tuulen kuljettaman ja kiinnittämän lian takia
Yökaste, kuura, räntä, sade, lumi ja rakeet	Rakenteiden rappioituminen ja hajoaminen, hävittää lämpöä osista ja rakenteista, materiaalien korroosio, sähköviat ja suojaeroksien kuluminen
Hiekka ja pöly	Kuluttaa ja naarmuttaa pintoja, kasvattaa kitkan määrää, aiheuttaa epäpuhtauksia voiteluöljyihin ja tukkeumia putkiin, sekä materiaalin väsymistä ja halkeilua
Korroosiota aiheuttava ilmasto	Aiheuttaa kemiallisia reaktioita materiaaleissa, korroosiota, elektrolyysiä, pintojen rappeutumista, johtokyvyn kasvamista ja kontaktiresistanssin kasvamista
Kosteus	Korkea kosteudentaso aiheuttaa korroosiota ja hapettumista johtaviin materiaaleihin, imeytymistä huokosiin materiaaleihin ja materiaalien paisumista.  Matala kosteudentaso aiheuttaa materiaalien haurastumista ja rakeistumista
Auringon säteily	Synnyttää otsonia, kuluttaa värejä, kovettaa kumimateriaaleja, kasvattaa lämpötilaa laitekoteloiden sisällä ja aiheuttaa lämpövanhennusta
Korkeat lämpötilat	Muuttaa resistanssia, induktanssia, kapasitanssia, tehokertoimia ja dielektrisiä vakioita, vioittaa liikkuvia osia pehmentämällä ja turvottamalla lämpöeristeitä, turvottaa pintoja, vanhentaa laiteosia, nopeuttaa hapettumista ja kemiallisia reaktioita, haihduttaa voiteluöljyä ja muuttaa niiden viskositeettia, rasittaa rakenteita lämpölaajenemisen kautta
Matalat lämpötilat	Haurastuttaa kumia ja vähentää sen notkeutta, muuttaa sähköisiä vakioita, muodostaa jäätä kosteudesta, kasvattaa voiteluöljyjen viskositeettia ja aiheuttaa geelitymistä, kasvattaa lämpöhäviötä, aiheuttaa pintojen halkeamista ja rasitusta lämpölaajenemisen kautta
Lämpöshokki	Aiheuttaa pysyviä muutoksia laitteen sähköisessä suorituskyvyssä, sekä halkeamia ja mekaanisia vikoja
Suuri ja pieni paine	Aiheuttaa halkeamia ja muodonmuuttumisia rakenteissa, tiivisteiden vuotamista, rakenneaurioita ilmakuplien muodostumisen takia, mittavirheitä korkeusmittareissa ja muuttaa sähköisiä ominaisuuksia.
Kaasut	Korroosion nopeutuminen, dielektristen aineiden rappioituminen, muuttaa lämpösähköisen siirtymisen ominaisuuksia ja lisää hapettumista

## 4.2 Olosuhdetestien käyttötavat

Olosuhdetesteillä on monia eri käyttötarkoituksia, ja niillä pyritään todentamaan eri asioita. Tyyppitestauksessa pyritään todentamaan laitteen kykyä kestää tiettyjä ympäristöolosuhteita. Testi kohdistuu vain muutamiin yksilöihin, joten testi ei kerro mitään laitteen luotettavuudesta tai pitkäaikaisen rasituksen kestokyvystä. Testaus voidaan tehdä esim. prototyypivaiheessa, ennen sarjatuotannon aloittamista tai huollon jälkeen. Se ei siis ole ajallisesti sidonnainen. Supistettu testi voidaan kohdistaa yhteen kappaleeseen sen ominaisuuksien selvittämiseksi prototyypivaiheessa. [6, s. 78.]

Luotettavuustesteissä (IEC 60605: Equipment reliability testing) testaaminen kohdistuu moniin yksilöihin. Testeissä käytettävät ajat ovat pitkiä, mutta samalla testiolosuhteet ovat lievät. Ympäristöolosuhteet matkivat tyypillisiä käyttöolosuhteita ja ne vaihtelevat jaksoittain. Testissä pyritään löytämään satunnaisvikoja. [6, s. 78.]

Pitkäaikaisen rasituksen siedon testissä testataan muutamia yksilöitä. Testissä käytetään pitkiä testausaikoja ja ankaria olosuhteita. Näin testaamista voidaan kiihdyttää ja löytää ajasta riippuvaisia vikoja. Nämä viat ovat systemaattisia ja voivat olla tyybiltään esim. suunnittelu-, valmistus- tai kulumisvikoja. [6, s. 79.]

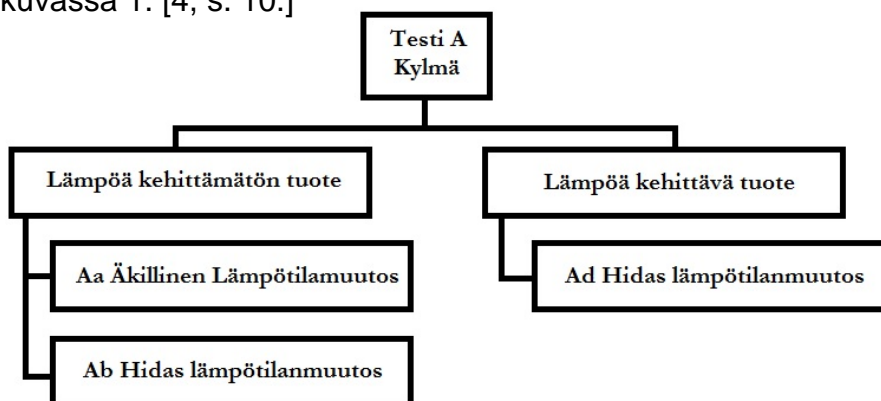
Rasituskarsinta (stress screening) on tuotelinjan loppupäässä tarkastuksia edeltävä ympäristökäsittely. Testissä pyritään löytämään ja karsimaan valmistuksessa syntyneet piilevät viat. Testissä esiintyvät rasitukset eivät kuitenkaan saa ylittää tuotteen tai materiaalien sähköisiä ja mekaanisia rasituskyynyksiä. Muuten seurauksena voi olla vaurioita tai väsymisilmiön kiihtyminen. Rasituskarsinnan tarkoituksena on nopeuttaa sellaisten vikojen syntymistä ja havaitsemista, jotka eivät paljastu muilla testaus ja tarkastusmenetelmillä. Rasituskarsintaa käytetään laadunvalvontaan. [6, s. 79.]

Vanhennusta (burn-in) käytetään parantamaan tuotteen luotettavuutta. Sen tarkoituksena on poistaa varhaisvikoja ja varmistaa, että tuotteen vikataajuus on vakiovikataajuusalueella käyttöönottovaiheessa. Vanhennusta käytetään komponenteille ja laitteille. [6, s. 79.]

### 4.3 IEC60068-2-1 Kylmätesti

IEC60068-2-1 on IEC-sarjaan kuuluva kylmätesti. Testin tarkoituksena on määrittää miten komponentti tai laite toimii ja kestää varastointia alhaisessa lämpötilassa. Testillä ei kuitenkaan voida määrittää laitteen toimivuutta tai varastoinnin kestävyyttä lämpötilan vaihteluiden aikana. [4, s. 10.]

Kylmätesti soveltuu sekä lämpöä kehittämättömille että lämpöä kehittäville tuotteille. Lämpöä kehittäviksi tuotteiksi määritellään ne tuotteet, joiden pinnan kuumin kohta on yli 5 °C korkeampi kuin ympäröivän ilman lämpötila. Lämpötila mitataan vapaassa ilmatilassa stabiloitumisen jälkeen. Testi on tarkoitettu normaalisti tuotteille, joiden lämpötila stabiloituu testin aikana. Kylmätestin lämpötilamuutos voidaan toteuttaa joko äkillisellä tai hitaalla muutosnopeudella. Äkillistä lämpötilamuutosta tulee käyttää vain silloin, kun voidaan olla varmoja, ettei äkillinen lämpötilan muutosnopeus vaurioita testattavaa tuotetta ja tuote on lämpöä kehittämätön. Kylmätestin valintaperiaate ja testityypit ovat esitetty kuvassa 1. [4, s. 10.]



Kuva 1. Kylmätestin testityypit

Kun kylmätesti aloitetaan, testattava näyte on huoneen lämpötilassa. Jos testissä käytetään äkillistä lämpötilan muutosta, olosuhdekaapissa on spesifikaation määrittelemä lämpötila. Testissä, jossa käytetään hidasta lämpötilan muutosta, olosuhdekaappi on aluksi huoneenlämpötilassa. Tämän jälkeen lämpötilaa muutetaan, kunnes se saavuttaa määritellyn testilämpötilan. Lämpötilan muutosnopeus ja taso on määritelty spesifikaatiossa. Kun näytteen lämpötila on stabiloitunut, sitä pidetään spesifikaatiossa määritelty aika

testiolosuhteissa. Lämpötila on stabiloitunut, kun kaikki näytteen osat poikkeavat enintään 3 °C tai spesifikaatiossa ilmoitetun määrän. [4, s. 11.]

Jos tuote on lämpöä kehittävä, tulee sille suorittaa lämpötilan mittauksia toistuvasti, jotta saadaan selville aikaväli, jolla lämpötila muuttuu 3 °C tai spesifikaatiossa ilmoitetun määrän. Lämpötila on stabiloitunut, kun kahden peräkkäisen aikavälin suhde ylittää arvon 1,7. Lämpöä kehittämättömät tuotteet testataan yleensä virrattomina, ja testin aikana käytetään koneellista ilmankiertoa. Lämpöä kehittävät tuotteet testataan virrallisina tai virrattomina riippuen spesifikaation määrityksistä. Koneellista ilmankiertoa ei suositella, mutta sitä voidaan käyttää, jos testiolosuhteita on hankalaa saavuttaa ilman sitä. Kylmätestin rasitusasteet on ilmoitettu taulukossa 2. [4, s. 11.]

Taulukko 2. Testin rasitusasteet

Testin lämpötila (°C)	-65	-55	-40	-25	-10	-5	5
Testin kesto-aika (h)	2	16	72	96			

Testistandardissa on määritelty lämpötilan toleranssiksi  $\pm 3$  °C. Käytettäessä hidasta lämpötilan muutosta (testityypit Ab ja Ad) muutosnopeus ei saa ylittää 1 °C:ta mitattuna alle viiden minuutin jakson keskiarvona. Rasituksen kesto voidaan rajoittaa aikaan, jolloin tuotteen lämpötila on stabiloitunut, jos testin ainoa tarkoitus on osoittaa, että testattava näyte toimii spesifikaatiossa määritellyssä lämpötilassa. [4, s. 11.]

Kun testin rasitus päättyy, toipuminen huoneenlämpötilaan tapahtuu joko olosuhdekaapissa tai huonetiloissa. Normaaleiksi toipumisolosuhteiksi on määritelty 15...35 °C ja 25...75 % RH. Jotta näytteen lämpötila stabiloituisi, on toipumisajaksi määritelty minimissään tunti ja maksimissaan kaksi tuntia. Tuona aikana tulee suorittaa kaikki ennalta määritellyt mittaukset loppuun. Mittauksissa, jotka suoritetaan testauksen aikana tai lopussa, ei välttämättä tarkisteta kaikkia tuotteen toimintaan liittyviä parametreja. [4, s. 12.]

## Testistä raportoitava tieto

Tuotespesifikaatiossa tulee määritellä testityyppi sekä alla esitetyistä tiedoista ne, jotka koskevat testityyppiä [4, s. 12]:

- esikäsittely
- edeltävät tarkastukset ja mittaukset
- testattavan näytteen kiinnitys tai tuenta
- näytteen tila mukaan lukien jäähdytys rasituksen aikana
- rasitusaste (lämpötila, lämpötilan muutosnopeus ja kestoaika)
- tarkastukset ja mittaukset ja/tai kuormitus rasituksen aikana
- toipumisolosuhteet, jos poikkeavat standardista
- lopputarkastukset ja mittaukset
- poikkeamat standardista tai asiakkaan ja toimittajan keskenään hyväksymästä menettelystä

### 4.4 IEC60068-2-14 Lämpötilan vaihtelu

IEC60068-2-14 on IEC-sarjan testi, ja sitä käytetään havainnollistamaan lämpötilan muutoksien vaikutuksia tuotteeseen. Testillä on kolme eri tyyppiä:

Testi Na: Testissä käytetään nopeaa lämpötilan vaihtelua, jonka muutos aika on määriteltä. Testin toteutetaan siirtämällä testinäyte olosuhdekaapista toiseen. Näin testinäytteeseen kohdistuvan lämpöshokin vaikutus on kohtalainen. [4, s. 25.]

Testi Nb: Testissä käytetään lämpötilan vaihtelua, jonka muutosnopeus on määritelty. Testi toteutetaan laittamalla näyte olosuhdekaappiin, joka on aluksi huoneen lämpötilassa. Lämpötilaa lähdetään muuttamaan asteittain spesifikaatiossa määritellyllä nopeudella. [4, s. 25.]

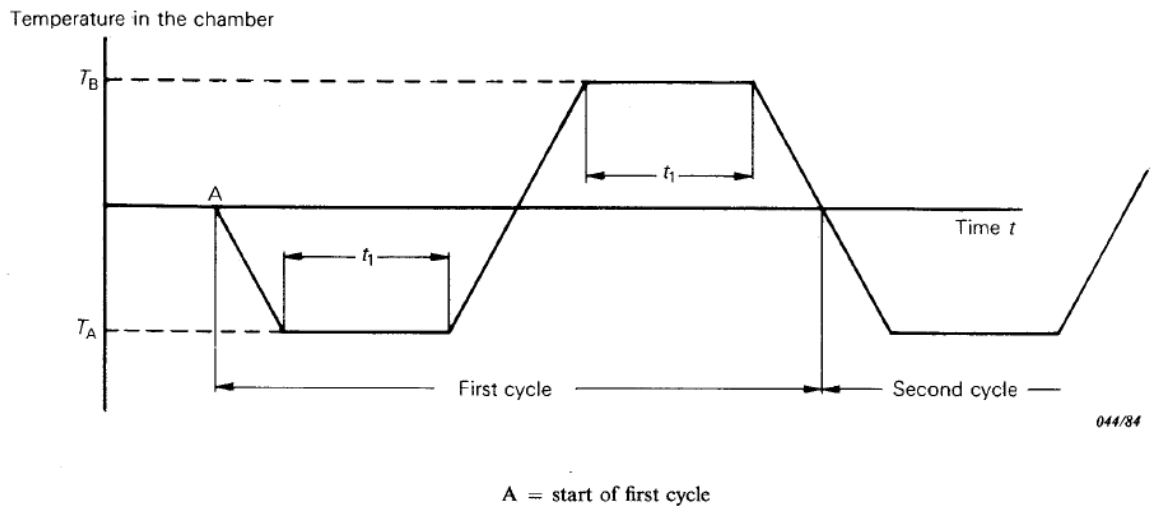
Testi Nc: Testissä käytetään nopeaa lämpötilan vaihtelua. Testi toteutetaan käyttämällä kahta nestekylpyä, joiden välillä näytettä siirretään. Testissä näytteeseen aiheutuva lämpöshokkivaikutus on huomattavan suuri. Tämä menetelmä sopii esim. lasi-metalli-tiivisteille, läpivienneille ja vastaaville näytteille. [4, s. 25.]

Testityypeistä Nb on käytetyin ja sen rasitusasteet on ilmoitettu taulukossa 3. Testintyyppin jakson rakenne on ilmaistu kuvassa 2.

Taulukko 3. Nb-testityypin rasitusasteet [4, s. 25]

Alalämpötila Ta (°C)	-65, -55, -40, -25, -10, -5, 5 (toleranssi $\pm 3$ °C) 30, 40, 55, 70, 85, 100, 125, 155, 175, 200 (toleranssi $\pm 2$ °C)
Ylälämpötila Tb (°C)	-65, -55, -40, -25, -10, -5, 5 (toleranssi $\pm 3$ °C) 30, 40, 55, 70, 85, 100, 125, 155, 175, 200 (toleranssi $\pm 2$ °C)
Testiaika t1	10 min, 30 min, 1 h, 2 h, 3 h
Lämpötilan muutosnopeus	1 $\pm$ 0,2 °C/min 3 $\pm$ 0,6 °C/min 5 $\pm$ 1,0 °C/min
Jaksoja	2 kpl, jos spesifikaatiossa ei toisin määritellä





Kuva 2. Lämpötilavaihtelutestin rakenne [7]

Testistä raportoitava tieto

Nb-tyyppisestä testistä tulee raportoida alla esitetyt tiedot [4, s. 26]:

- testikappaleen kiinnitys tai tuenta, jos poikkeaa standardista
- alalämpötila  $T_A$  ja ylälämpötila  $T_B$
- lämpötilan muutosnopeus
- jaksojen lukumäärä, jos muu kuin 2
- edeltävät tarkastukset
- testikappaleen tila kaappiin vietäessä
- testiaika  $t_1$ , jos muu kuin 3 h
- testinaikaiset tarkastukset
- toipumisolosuhteet
- lopputarkastukset.

## 5 OLOSUHDEKAAPPI

Jotta testistandardeissa määritellyt vaatimukset voitaisiin toteuttaa, on testaamista varten täytynyt kehittää erityisiä olosuhdekaappeja. Olosuhteet, joita näillä kaapeilla pystytään toteuttamaan, vaihtelevat tyyppikohtaisesti, mutta tyypillisesti ne pyrkivät täyttämään yleisimmät kuluttajalaitteille suunnatut standardit. Tässä insinööriyössä olosuhdetestien toteuttamiseen käytettiin Discovery DY110C -olosuhdekaappia, joka on esitetty kuvassa 3. Olosuhdekaapin käyttämisestä ja ominaisuuksista laadittiin insinööriyön aikana lyhyt dokumentti Tieto-Oskarille.



Kuva 3. Discovery DY110C -olosuhdekaappi suorittamassa olosuhdetestiä

### 5.1 Olosuhdekaapin ominaisuudet

Discovery DY110C -olosuhdekaappi on italialaisen yrityksen, Angelantoni Industrien, valmistama [8]. DY110C-olosuhdekaappi on tarkoitettu niitä ympäristötestejä varten, joissa vaikuttavina olosuhteina esiintyy lämpötila, kosteus tai molemmat [8]. Testien kohteena ovat yleensä elektroniikkalaitteet,

komponentit, koteloinnit tai mekaaniset osat. Discovery DY110C -olosuhdekaapin tukemat standardit on esitetty kuvassa 4.

MAIN STANDARDS	
TEMPERATURE TESTS	CLIMATIC TESTS
<b>COLD ONLY</b>	<b>CONSTANT CLIMATE</b>
DIN 40046 Page 3, Test A	DIN 40046
IEC 68-2-1, Test A	DIN 50014
BS 2011, Part 2, Test A	IEC 68-2-3, Test Ca
MIL-STD 810 D, Met. 502.2	MIL-STD 202 E, Met. 103 B
MIL-E 5272, Test 4.2	DIN/IEC 68-2-56
<b>HOT ONLY</b>	<b>VARIABLE CLIMATE</b>
DIN 40046, Page 4, Test 3	DIN/IEC 68-2-30 DB
IEC 68-2-2, Test B	IEC 68-2-38
BS 2011, Part 2, Test B	MIL-STD 202 E, Met. 106 D
MIL-STD 810 D, Met. 501.2	MIL-STD 883 C, Met. 1004.4
MIL-STD 883 C, Met. 1008.2	DIN 40046 Page 6 and 31
MIL-E 5272, Met. 4.1	IEC 68-2-4, Test D
MIL-STD 202 E, Met. 108 A	BS 2011, Part 2.1, Test Da
<b>HOT/COLD</b>	MIL-STD 750 B, Met. 1021.1
DIN 40046, Page 14, Test Nb	DIN 40046 Page 101
IEC 68-2-14 Nb	DIN 50016
MIL-STD 311 A, Part 112.1	MIL-STD 311 A, Part. 105.1
	MIL-STD 810, Met. 507 Proc. 1-2-3

Kuva 4. Discovery DY110C -olosuhdekaapin tukemat standardit [8]

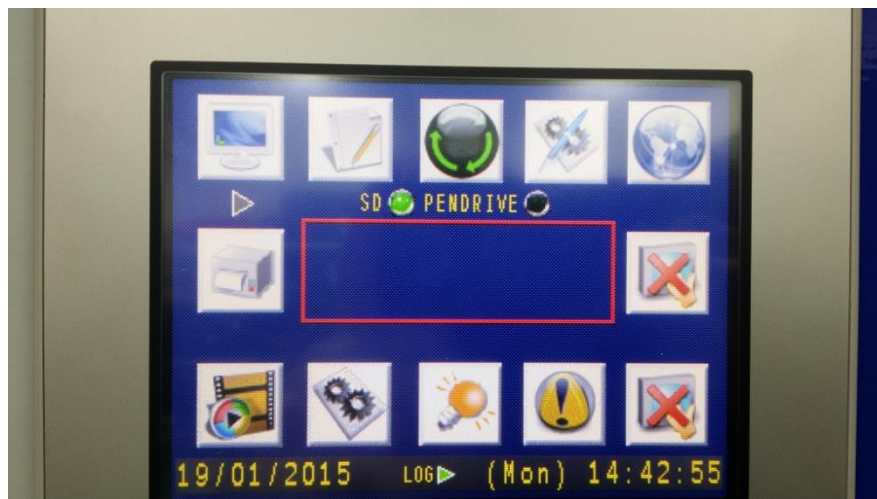
Jotta näytteiden testaaminen olisi mahdollista, löytyy olosuhdekaapin oikealta kyljeltä tiivistetty läpivienni. Läpiviennin kautta on mahdollista esim. kuljettaa johdotukset tai suorittaa mittauksia näytteelle testin aikana. Läpivienni on esitetty kuvassa 5.



Kuva 5. Olosuhdekaapin läpivienni

Käyttäjä voi ohjata olosuhdekaapin toimintaa joko etupaneelista löytyvästä keykratos-konsolista tai winkratos-ohjelmalla. Keykratos-konsoli on esitetty kuvassa 6. Konsoli on kosketusnäyttöllinen, ja siinä on seuraavat toiminnot:

- konsolin kieli
- lämpötilan ja kosteuden tason säätö
- testiohjelman luominen ja suorittaminen
- testikammion monitorointi
- hälytyksien monitorointi ja kuittaus
- mittaustuloksien kirjaaminen lokiin
- mittaustuloksien tallennus muistitikulle
- testikammion valaistuksen säätö
- kaapin ohjauslogiikan valvonta.



Kuva 6. Keykratos-konsoli

Olosuhdekaapin toimintaa ohjaa ohjelmoitava logiikkakontrolleri eli PLC (Programmable Logic Controller). PLC:n tehtävä on varmistaa, että olosuhdekaappi toimii oikein ja turvallisesti kaikissa tilanteissa. Jos virhetilanteita ilmenee, PLC katkaisee testin suorituksen ja antaa käyttäjälle virheilmoituksen.

Testin suorittaminen jatkuu, kun käyttäjä on korjannut syyn virheeseen ja kuitannut ilmoituksen. Näin suojataan sekä testinäytettä että olosuhdekaapin osia virhetilanteissa.

Suurin osa hälytyksien raja-arvoista on ennalta asetettu, mutta esim. lämpötilan raja-arvot tulee käyttäjän asettaa itse. Lämpötilalle on varmuudeksi toteutettu sekä raudalliset että ohjelmalliset hälytykset. Ohjelmalliset hälytykset tulee asettaa 5 °C korkeammiksi kuin testin aikana ilmenevä suurin ja pienin lämpötila. Raudalliset hälytykset tulee puolestaan asettaa 5 °C korkeammiksi kuin ohjelmalliset. Näin ohjelmalliset hälytykset laukeavat ensin ja raudalliset vasta jälkimmäisenä, jos lämpötila virheellisesti jatkaa nousuaan. Ohjelmalliset hälytykset asetetaan keykratos-konsolista ja raudalliset hälytykset olosuhdekaapin oikealta kyljeltä löytyvästä termostaatista. Termostaatti on esitetty kuvassa 7.



Kuva 7. Olosuhdekaapin termostaatti ja liitännät

Liitännöiden osalta DY110C-olosuhdekaapista löytyy RS232- ja Ethernet-liitännät verkkoliikennöintiä varten. Verkon kautta voidaan ohjata olosuhdekaapin toimintaa PC:llä, johon on asennettu winkratos-ohjelma. Verkkoliitännöiden lisäksi olosuhdekaapilla on analogisille ja digitaalisille signaaleille RS232 I/O-liitännät, joihin käyttäjä voi halutessaan kytkeä omia laitteita tai antureita. Jos I/O-liitännään on kytketty signaalia tuottava laite tai anturi, olosuhdekaappi kykenee

taltioimaan ja esittämään signaalin käyttäjälle. Discovery DY110C -olosuhdekaapin spesifikaatio on kokonaisuudessaan ilmoitettu taulukossa 4.

Taulukko 4. Discovery DY110C -olosuhdekaapin spesifikaatio

Kapasiteetti	112 l
Testitilan ulottuvuudet	548 mm x 460 mm x 447 mm
Olosuhdekaapin ulottuvuudet	880 mm x 1065 mm x 1380 mm
Lämpötilan säätöalue	-70°C...180 °C
Lämpötilan vaihtelu (max)	±0,1 °C...±0,3 °C
Lämmitysnopeus (max)	3,2°C/min
Jäähdytysnopeus	2,8°C/min
Ilmankosteuden säätöalue (10 °C...95 °C)	10 %...95 %
Kosteuden tarkkuus	±0,1 %...±3 %
Paino	435 Kg
Melutaso	60 dB
Käyttöjännite	230 V+6 % /-10 % /50 Hz /1+G
Liitännät	RS232, Ethernet

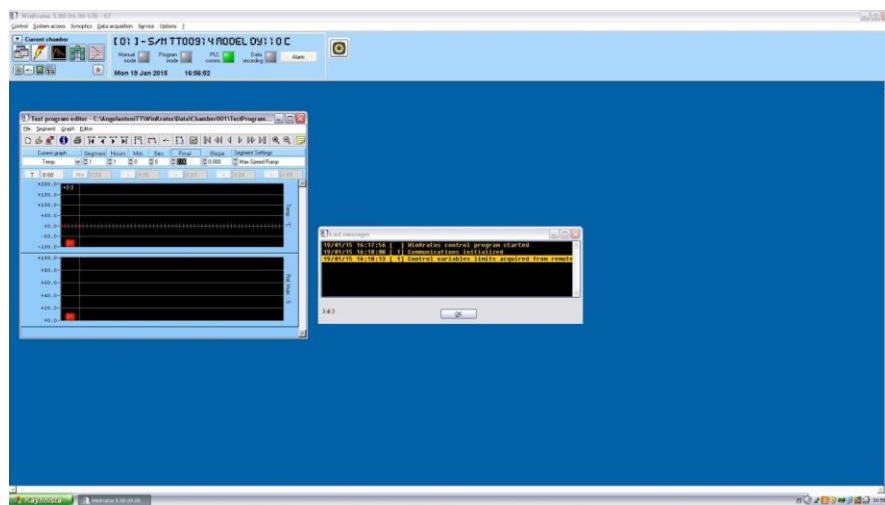
## 5.2 Winkratos-ohjelma

Winkratos on Angelantoni Industrien yrityksen kehittämä ohjaus- ja valvontaohjelmisto Discovery DY110C -olosuhdekaapille. Käyttöalustana toimii pc, jonka käyttöjärjestelmänä on Windowsin XP, Vista, 7 tai uudempi versio. [8.]

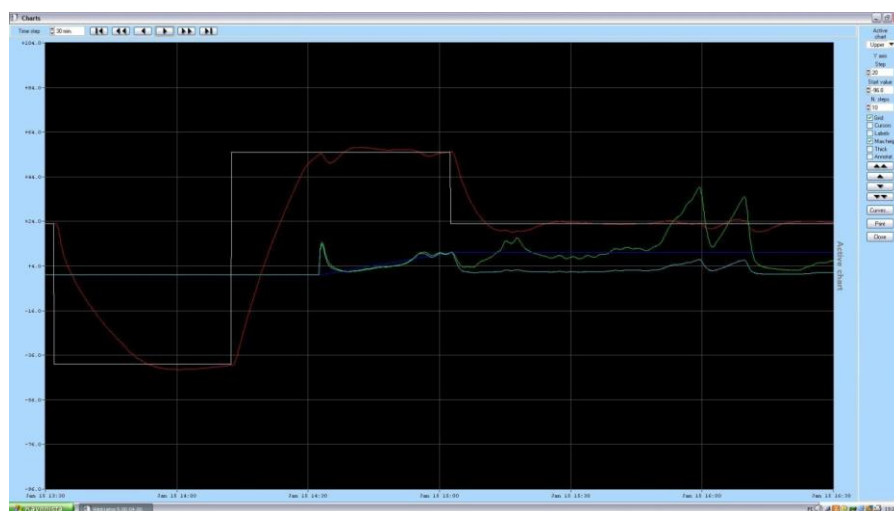
Olosuhdekaapin ohjaamiseen on kaksi tilaa, manuaalinen ja ohjelmallinen. Manuaalillassa käyttäjä asettaa itse halutun kosteuden ja lämpötilan tason sekä muutosnopeuden, jolla nämä tasot saavutetaan. Kun määritellyt tasot on

saavutettu, olosuhdekaappi pyrkii pitämään tilaa yllä niin kauan, kunnes käyttäjä määrittelee uudet tasot.

Ohjelmatilassa käyttäjä luo testiohjelman, jonka olosuhdekaapin PLC suorittaa. Ohjelma koostuu segmenteistä, joista kullekin on määritelty kestoaika, muutosnopeus sekä kosteuden ja lämpötilan taso. Halutessaan käyttäjä voi kytkeä tiedon tallentamisen päälle, jolloin testin tulokset kirjataan lokitiedostoihin ja esitetään reaaliajassa käyttäjälle. Tuloksia voidaan tarkastella numeerisessa tai graafisessa muodossa. Winkratos-ohjelman käyttöliittymä on esitetty kuvassa 8 ja sen tuottama graafi kuvassa 9.



Kuva 8. Winkratos-ohjelman päänäkymä



Kuva 9. Winkratos-ohjelman piirtämä graafi olosuhdetestistä

## 6 TESTAUSOHJELMAN SUUNNITTELU

Tämän insinööriyön pääpainona oli testausohjelman suunnittelu Tieto-Oskari Oy:n valmistamalle tuotteelle. Testausohjelman tavoitteena on sekä helpottaa että tehostaa tuotantovaiheen loppupäässä tapahtuvaa testausta. Lisäksi testausohjelman toivotaan helpottavan viallisten laitteiden paikantamista.

### 6.1 Testausohjelman vaatimukset

Testausohjelmalle määriteltiin vaatimukset ennen sen toteuttamista. Vaatimuksia kuitenkin lisättiin, karsittiin tai tarkennettiin insinööriyön edetessä aivan kuten muissakin ohjelmointiluontoisissa projekteissa. Vaatimusmäärittely muodostui keskustelemalla työn tilaajan ja tulevien käyttäjien kanssa siitä, mitä testausohjelmalta odotettiin. Ohjelman vaatimuksiksi määriteltiin seuraavat:

- Tietoliikenne pc:n ja testattavan tuotteen välillä tulee tapahtua muuntamalla USB-portin sarjaliikenne RS232-muotoiseksi. Muunnos toteutetaan käyttämällä yritykseltä löytyvää USB-RS232-muunninta.
- Ohjelman tulee olla yhteensopiva Tieto-Oskarilla yleisimpien käytössä olevien käyttöympäristöjen kanssa. Tästä syystä testausohjelman alustaksi valittiin Microsoft Windows. Testausohjelma tulee toimia ainakin Windowsin XP-, Vista- ja 7-versioissa.
- Ohjelman projekti tulee olla toteutettu Tieto-Oskarilla käytössä olevalla kehitysympäristöllä ja ohjelmointikielellä. Ohjelmointikieleksi valittiin C++ ja kehitysympäristöksi valittiin Borland C++Builder 6. Kehitysympäristöstä käytettiin C++Builderin vanhaa versiota, jotta projektissa voidaan hyödyntää jo olemassa olevaa sarjaliikennöintikirjastoa.
- Ohjelman tulee kestää mahdollisia virhetilanteita ja ilmoittaa niistä käyttäjälle. Jos virhetilanne ilmaantuu, on riskinä menettää kaikki testin aikana kerätty tieto. Tällöin mahdollisesti useamman vuorokauden mittainen testi jouduttaisiin uusimaan.



- Ohjelman tulee olla helppokäyttöinen ja selkeä ulkoasultaan.
- Projektin tulee olla helposti jatkettavissa.

## 6.2 Testausohjelman toiminnot ja ominaisuudet

Kuten ohjelman vaatimuksien kanssa, tarve ohjelman sisältämille ominaisuuksille ja toiminnoille kehittyi työn edistymisen myötä. Testausohjelmasta tulee löytyä seuraavat ominaisuudet ja toiminnot:

- Toiminto, joka käynnistää tuotteen itsediagnostiikan ja vastaanottaa sen tulokset. Tämän lisäksi ohjelman tulee pystyä muuntamaan vastaanotettu tieto oikeaan muotoon. Itsediagnostiikka käynnistetään lähettämällä käsky testattavalle tuotteelle sarjaliikenteen kautta.
- Toiminto, joka tallentaa kunkin diagnostiikan tulokset erillisiin tiedostoihin, ja mittauksen loputtua kasaa kaikki tulokset yhteen tiedostoon. Mittaustiedostojen formaatin tulee olla .CSV, jotta kerätty tieto voidaan tuoda helposti Exceliin.
- Laitelista, josta käy ilmi, mitkä laitteet on kytketty ja missä on esiintynyt poikkeamia verkkojen liikenteessä. Laitteiden tilat on ilmaistu väreillä, jotka vaihtuvat asetetun raja-arvon alittuessa. Vihreä ilmaisee laitteen verkkojen olevan kunnossa, keltainen ilmaisee verkon tilan käyvän matalana ja punainen ilmaisee, että verkossa on tapahtunut virhe. Laitteet, joita ei ole kytketty, näkyvät listassa harmaalla.
- Toiminto, joka esittää mittaustulokset graafisesti. Graafista tulee käydä ilmi diagnostiikkadatan arvo ja aika, milloin tieto on saatu. Lisäksi käyttäjän tulee pystyä valitsemaan, mitkä verkot piirretään graafiin. Myös aikaisempia mittaustuloksia tulee pystyä tarkastelemaan ohjelmalla.
- Virhetilanteiden tunnistaminen ja niistä ilmoittaminen käyttäjälle.
- Toiminto, joka tallentaa käyttäjän määrittelemät testiasetukset.

Lisäksi käyttäjän tulee pystyä tekemään seuraavat valinnat testin suorittamiseen liittyen:

- Valinta, joka aloittaa tai lopettaa testin.
- Valinta, joka asettaa mittaustiedoston nimen ja tallennuspolun.
- Valinta, joka asettaa USB- tai sarjaportin, mihin testattava laite on kytketty.
- Valinta, joka asettaa sarjaliikenteen nopeuden (Baud rate).
- Valinta, joka asettaa diagnostiikan kyselyn aikavälin.
- Valinta, joka asettaa raja-arvot laitelistan tilaa ilmaiseville väreille.
- Valinta, joka asettaa graafisesti reaaliajassa esitettävän laitteen.

### 6.3 Testin rakenteen määrittely

Kun testaus aloitetaan, ohjelma lähettää testattavalle tuotteelle käskyn ajaa itsediagnostiikan. Välittömästi käskyn lähettämisen jälkeen ohjelma vastaanottaa kuittauksen diagnostiikan aloittamisesta. Tämän jälkeen ohjelma jää odottamaan vastausta siksi aikaa, kunnes testattava tuote on suorittanut tiedonkeruun. Tiedonkeruun valmistuminen kestää noin minuutin ajan. Diagnostiikan valmistuttua ohjelma vastaanottaa tulokset ja muuntaa ne käyttäjän ymmärtämään muotoon. Muunnoksen jälkeen tulokset tallennetaan väliaikaisesti mittatiedostoihin ja esitetään käyttäjälle graafisesti. Kun testaus lopetetaan, kaikkien väliaikaisten mittaustiedostojen data kerätään yhteen ja tallennetaan käyttäjän aiemmin määrittelemään tiedostoon.

## 6.4 C++Builder ja TComPort

C++Builder on nopean kehityksen mallia noudattava (Rapid Application Development, RAD) ohjelmointiympäristö. Ohjelmointikielenään ympäristö käyttää C++:aa, jolle on toteutettu kattava valikoima valmiita kirjastoja. Tämän ansiosta C++Builderilla voidaan toteuttaa nopeasti ohjelmistoja, jotka sisältävät graafisen käyttöliittymän. Ohjelmointiympäristön ja sillä toteutettujen ohjelmien alustana toimii Windows. [9]

Jotta insinööriyön suorittamiseen varattu aika voitaisiin käyttää varsinaisen ohjelman ja toiminnallisuuden toteuttamiseen, käytettiin laitteiden kommunikointiin valmista TComPort-sarjaliikennekirjastoa. TComPort sisältää kaikki sarjaliikenteen edellyttämät perustoiminnot. Nykyisin TComPort-kirjasto on public domain -lisenssin alainen.

## 7 TYÖN SUORITUS

### 7.1 Käyttöönotto

Työprosessi aloitettiin käyttöönottamalla työn edellyttämät ohjelmistot ja kirjastot. Ensimmäisenä asennettiin C++Builderin 6 -versio. Ohjelman asentaminen eteni muuten normaalisti, mutta käytettäessä Windowsin 7 -versiota tulee ohjelma joko asentaa muualle kuin C-asemalle tai antaa asennuskansiolle kaikki käyttöoikeudet. Muuten ohjelma ei toimi puutteellisten käyttöoikeuksien takia.

Tämän jälkeen C++Builderiin lisättiin TComPort-sarjaliikennekirjasto. Kirjaston run-time package tulee ensin avata C++Builderilla ja kääntää. Kun käänös on tehty, tulee projektin library- ja include- polut asettaa osoittamaan kansioon, josta käännökset löytyvät. Lisäksi sama polku tulee lisätä ohjelman ympäristöasetuksista löytyvään library path -kohtaan. Tämän jälkeen avataan ja käännetään design-time package, joka lisää ohjelmaan uuden välilehden kirjaston komponenteille.

Kun kirjasto on asennettu, tulee siitä korjata kaksi virhettä. Ensimmäinen virhe korjaantuu, kun poistetaan sarjaliikennekirjaston CPORT.hpp-tiedostosta seuraavat määrittelyt:

```
/* Exception.CreateRes */ inline __fastcall EComPort(int Ident)/* overload */ : Sysutils::Exception(Ident) { }
```

```
/* Exception.CreateResHelp */ inline __fastcall EComPort(int Ident, int AHelpContext)/* overload */ : Sysutils::Exception(Ident, AHelpContext) { }
```

Toinen virhe korjaantuu, kun lisätään winspool.h-nimiseen tiedostoon kuvassa 10 punaisella alleviivatut rivit. Tiedosto löytyy ohjelman asennuskansion sisältä löytyvästä Include-nimisestä kansioista. Lisäksi projektin asetuksiin tulee määritellä conditional defines -kohtaan DONT\_USE\_WINSPOOL\_SETPORTA. C++Builderin ja sarjaliikennekirjaston asentamisen jälkeen asennettiin FTDI D2XX -ajurit, jotka mahdollistavat USB-laitteiden käytön. Työssä käytetty USB-RS232-muunnin tarvitsee FTDI-ajurit toimiakseen.

```

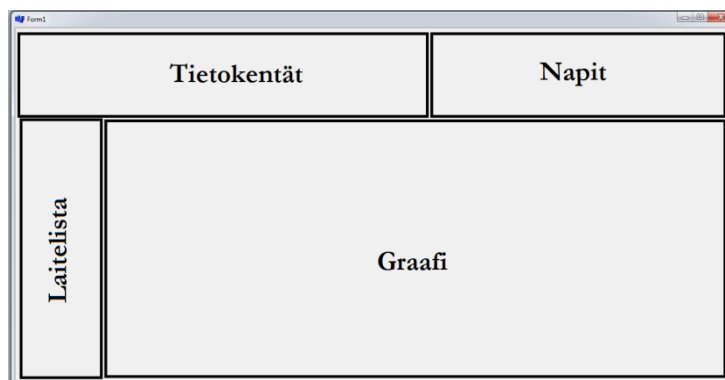
2515 #ifndef DONT_USE_WINSPOOL_SETPORTA
2516 BOOL
2517 WINAPI
2518 SetPortA(
2519     IN LPSTR     pName,
2520     IN LPSTR     pPortName,
2521     IN DWORD     dwLevel,
2522     IN LPBYTE    pPortInfo
2523 );
2524 BOOL
2525 WINAPI
2526 SetPortW(
2527     IN LPWSTR    pName,
2528     IN LPWSTR    pPortName,
2529     IN DWORD     dwLevel,
2530     IN LPBYTE    pPortInfo
2531 );
2532 #ifdef UNICODE
2533 #define SetPort  SetPortW
2534 #else
2535 #define SetPort  SetPortA
2536 #endif // !UNICODE
2537 #endif // !DONT_USE_WINSPOOL_SETPORTA

```

Kuva 10. Virheen korjaavat lisätyt rivit

## 7.2 Ohjelmointi

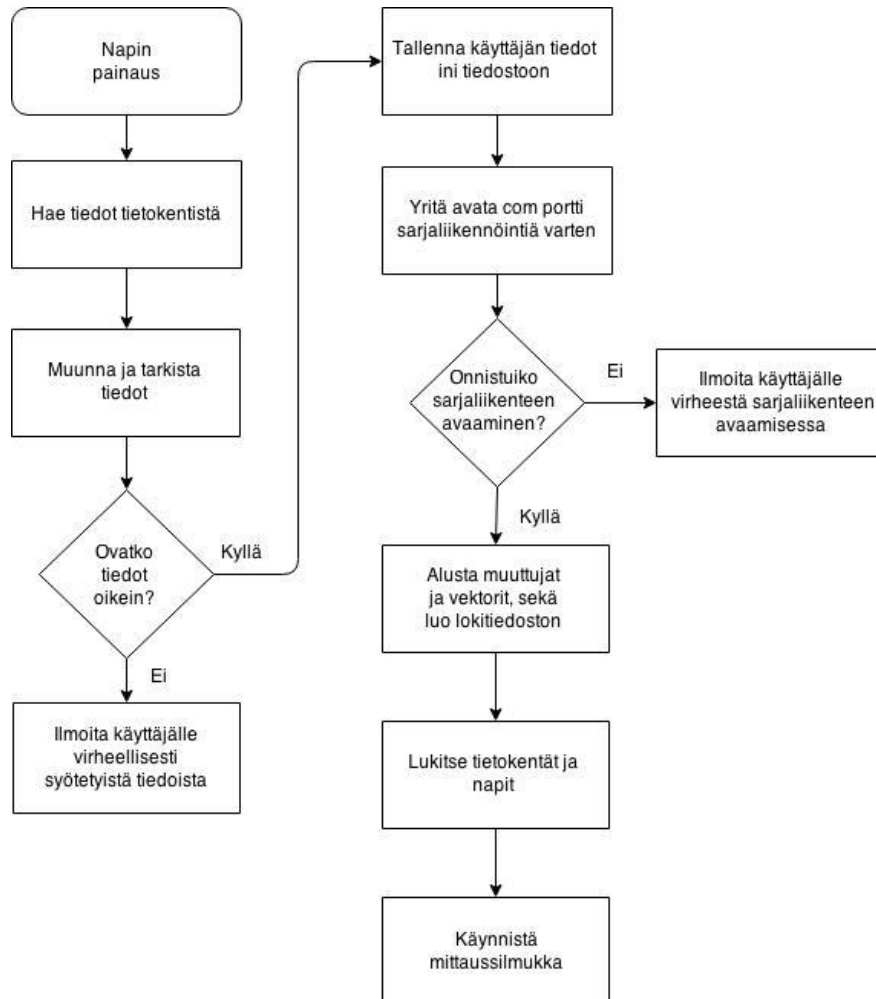
Testausohjelman tekeminen aloitettiin luomalla uusi projekti. Tällöin C++Builder luo automaattisesti projektipohjaksi tyhjän ikkunan. Ikkunaan ryhdyttiin hahmottelemaan ohjelman käyttöliittymän rakennetta. Hahmotelma käyttöliittymästä on esitetty kuvassa 11. Käyttöliittymän rakenne pyrittiin toteuttamaan mahdollisimman loogiseksi käyttäjän kannalta.



Kuva 11. Hahmotelma testausohjelman käyttöliittymän rakenteesta

Tämän jälkeen ryhdyttiin lisäämään tietokenttiä, jotka ovat oleellisia testin suorittamisen kannalta. Nämä kentät ovat valittu portti, baud rate, diagnoosien

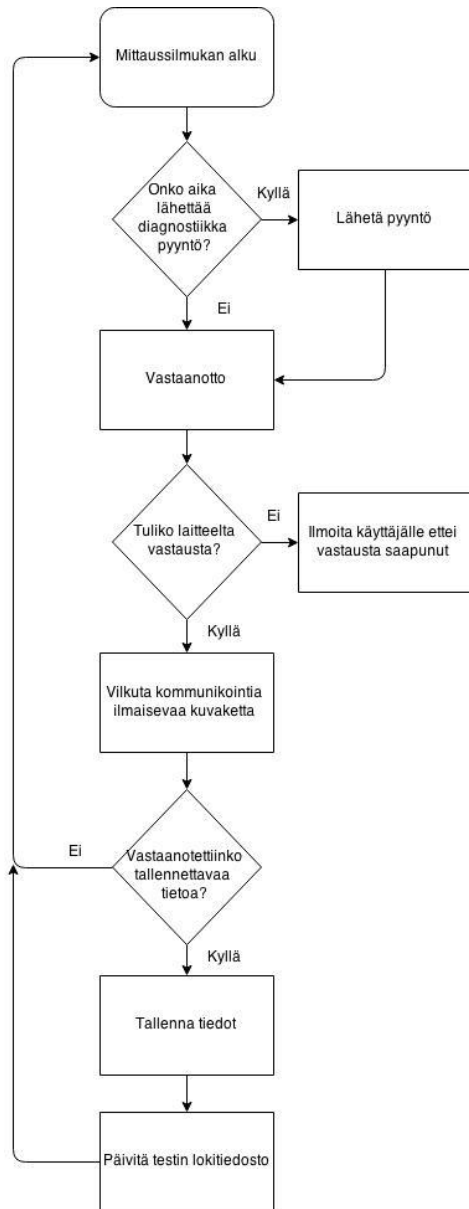
kyselyn aikaväli, mittaustiedoston nimi ja polku. Tietokenttien luomisen jälkeen ohjelmaan lisättiin napit tiedostopolun vaihtamiselle sekä mittauksen aloittamiselle ja lopettamiselle. Jotta ohjelman käyttö pysyisi yksinkertaisena, liitettiin suurin osa ohjelman toiminnasta Aloita mittaus -napin alle. Painettaessa Aloita mittaus -nappia suoritetaan kuvan 12 mukainen prosessi.



Kuva 12. Aloita mittaus -napin käynnistämä prosessi

Ohjelman mittaussilmukka toteutettiin luvun 6.3 määritelmän mukaisesti, ja se on esitetty kuvassa 13. Perinteisten toistorakenteiden sijasta mittaussilmukassa käytettiin C++Builderin ajastintoimintoa, jotta välttyttäisiin käyttöliittymän jäätymiseltä ja säikeiden luomiselta. Mittaussilmukkaan lisättiin myös toimintoja virheiden tunnistamista varten. Tärkeimmät näistä ovat ohjelman kaatumisen ja testattavan laitteen kommunikoimattomuuden tunnistaminen. Ohjelman kaatumisen tunnistaminen on toteutettu lokitiedostolla, jota päivitetään

mittaussilmukan yhteydessä. Jos ohjelma on kaatunut testin suorittamisen aikana, voidaan testiä jatkaa seuraavan käynnistyksen yhteydessä. Vaihtoehtoisesti testi voidaan päättää ja kerätä kaikki aiempi tieto yhteen mittautiedostoon. Jos testattava laite ei kommunikoi, ilmoitetaan ongelmasta käyttäjälle virhesanomalla.



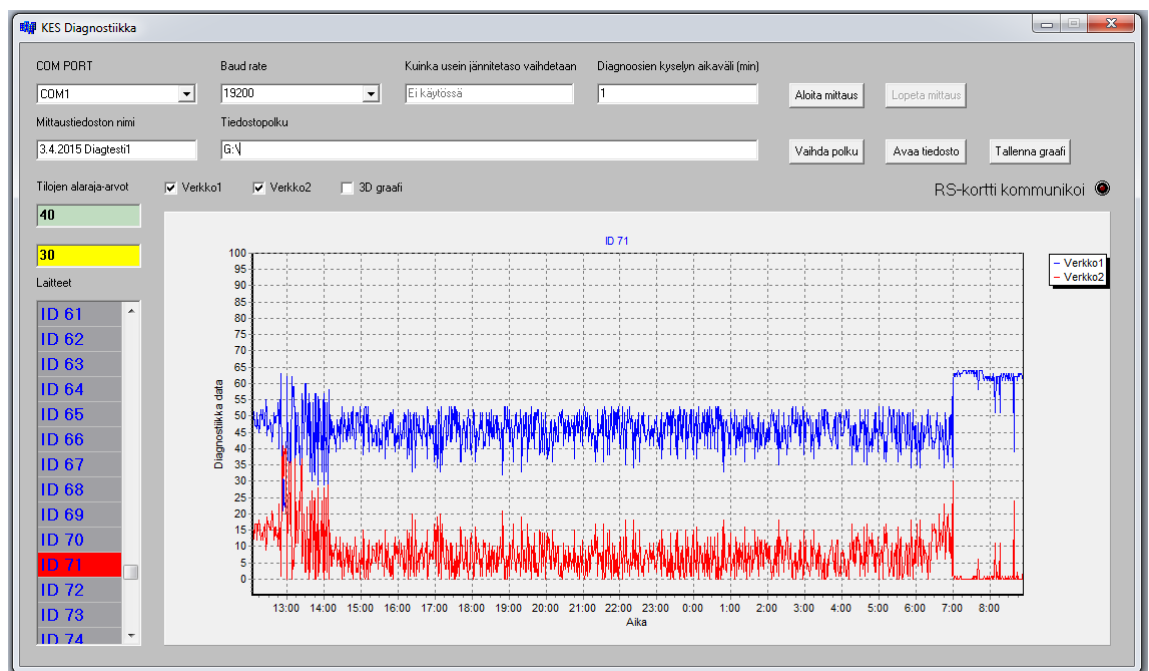
Kuva 13. Ohjelman mittaussilmukan rakenne

Lopeta mittaus -nappi sulkee kaikki testiä varten avatut yhteydet, kerää kaikki mittautulokset yhteen tiedostoon, poistaa väliaikaiset tiedostot sekä avaa tietokentät ja napit. Kun ohjelmaan oli toteutettu testin aloittaminen, sen käynnissä pitäminen ja lopettaminen, lisättiin ohjelmaan Avaa tiedosto -nappi.

Sitä painettaessa avautuu uusi ikkuna, josta käyttäjä voi valita aiemmin luodun mittaustiedoston. Kun mittaustiedosto on valittu, tuodaan mittauksen tulokset ohjelman muistiin.

Seuraavaksi ohjelmaan lisättiin laitelista ja tietokentät laitelistan värien raja-arvoille sekä graafi ja sen toiminnot. Laitelistassa esitetään kunkin laitteen tilat eri väreillä. Väreistä harmaa ilmaisee, että laitetta ei ole kytketty, vihreä ilmaisee laitteen tilan olevan kunnossa, keltainen ilmaisee tilan käyvän normaalia alempana ja punainen ilmaisee virhettä laitteen tilassa. Käyttäjällä voi asettaa tietokenttien avulla vihreän ja keltaisen tilan alarajat.

Kun listasta valitaan laite, ohjelma hakee sen tiedot ja piirtää niiden pohjalta graafin. Graafin osalta käyttäjä voi valita, mitkä verkot esitetään, onko se esitetty kaksi- vai kolmiulotteisena ja tallennetaanko graafi. Viimeiseksi mittaussilmukan loppuun lisättiin logiikka, joka tarvittaessa päivittää laitelistan tilat ja graafin. Testausohjelman lopullinen käyttöliittymä on esitetty kuvassa 14.



Kuva 14. Testausohjelmalle toteutettu käyttöliittymä

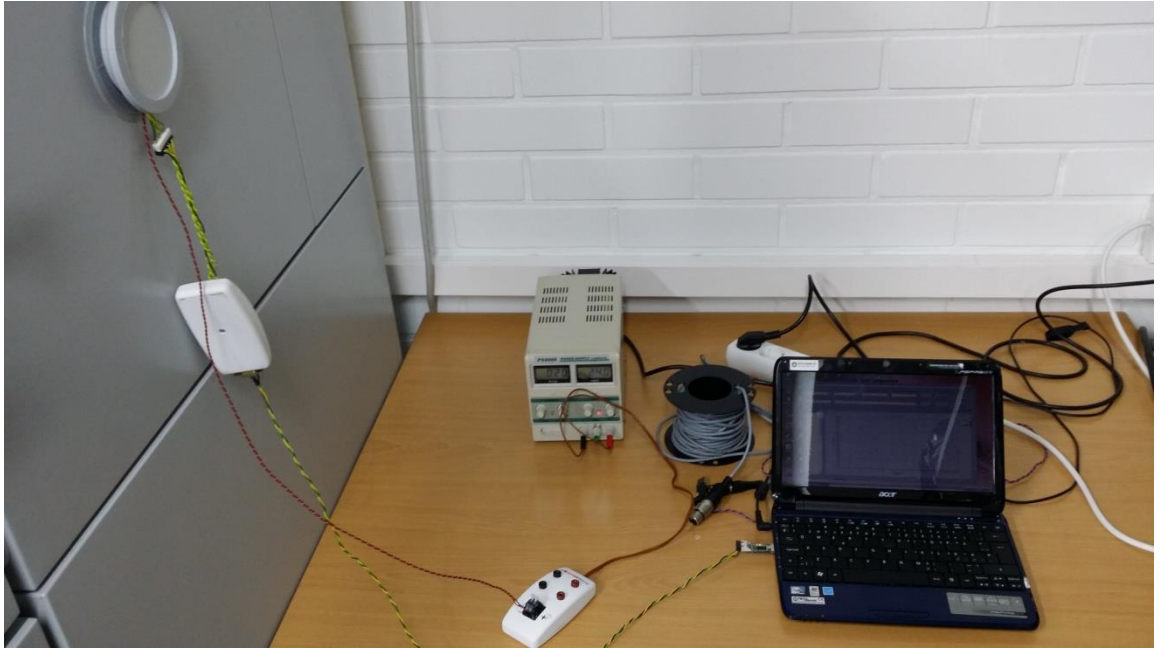


### 7.3 Testaus

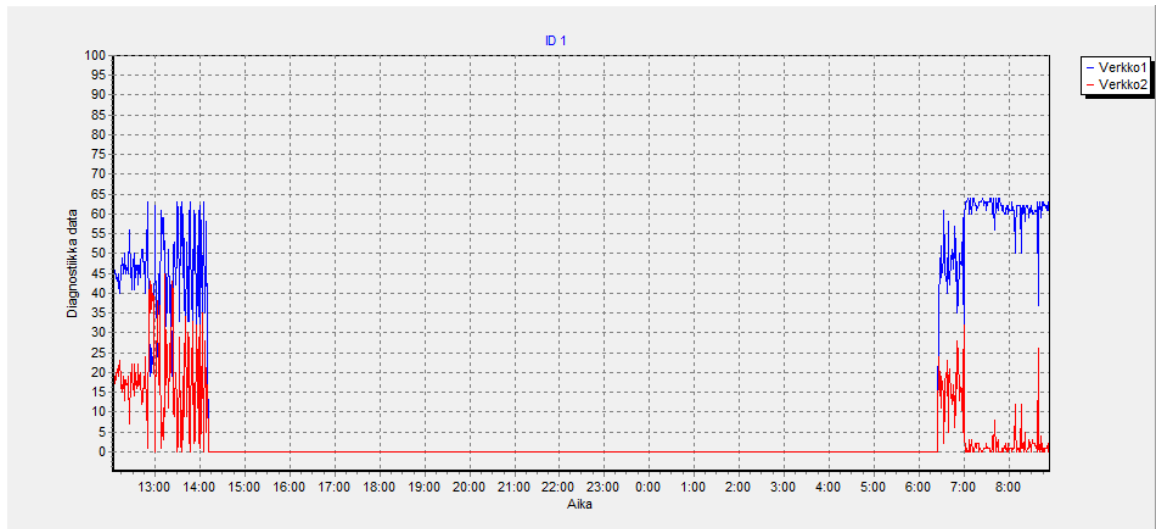
Kun testausohjelma oli saatu valmiiksi, sen toimivuutta testattiin olosuhdetestien yhteydessä. Ensimmäisenä määriteltiin kylmä- ja lämpötilan vaihtelu -testien rasisasteet. Kylmätestin tyypiksi valittiin Ab ja Lämpötilan vaihtelun tyypiksi Nb. Määrytyksien jälkeen testit luotiin winkratos-ohjelmalla ja olosuhdekaappi asetettiin valmiiksi testejä varten. Tämän jälkeen testikammioon sijoitettiin kolme testinäytettä, joista yksi oli yhdistetty USB-RS232-muuntimen kautta tietokoneeseen. Tietokoneella oleva testausohjelma asetettiin monitoroimaan testinäytteitä olosuhdetestien suorituksen ajaksi. Testin suorittaminen on esitetty kuvissa 3 ja 15. Kun testit olivat päättyneet, ohjelman todettiin toimivan halutulla tavalla. Testin tuloksista tuotettu graafi on esitetty kuvassa 16.

Taulukko 5. Olosuhdetestien määrytykset

Testi	60068-2-1 Kylmä	60068-2-14 Lämpötilan vaihtelu
Testin tyyppi	Ab	Nb
Tuenta	Metallinen ritilä, jonka päälle on aseteltu suojaava pahvi	Metallinen ritilä, jonka päälle on aseteltu suojaava pahvi
Alalämpötila	-40 °C	-40 °C
Ylälämpötila	-	70 °C
Muutosnopeus	1 °C/min	1 °C/min
Jaksojen lukumäärä	-	2 kpl
Testiaika t1	16 h	3 h
Edeltävät tarkistukset	Testinäytteiden toiminnallisuuden toteaminen huoneenlämpötilassa	Testinäytteiden toiminnallisuuden toteaminen huoneenlämpötilassa
Tila kaappiin vietäessä	Huoneen lämpötila ja kosteus	Huoneen lämpötila ja kosteus
Näytteen tila rasisuksen aikana	Näytteet ovat jännitteellisinä koko rasisuksen ajan	Näytteet ovat jännitteellisinä koko rasisuksen ajan
Testin aikaiset tarkastukset	Testiohjelma kirjaa ylös diagnostiikan tulokset	Testiohjelma kirjaa ylös diagnostiikan tulokset
Toipumisolosuhteet	23 °C ja 10% RH	23 °C ja 10% RH
Lopputarkastukset	Testinäytteiden toiminnallisuuden toteaminen huoneenlämpötilassa	Testinäytteiden toiminnallisuuden toteaminen huoneenlämpötilassa



Kuva 15. Testausohjelmaa suorittava tietokone



Kuva 16. Kylmätestin aikana kerätystä datasta tuotettu graafi

## 8 TULOSTEN ANALYSOINTI

Tässä luvussa arvioidaan työn aikana saatuja tuloksia. Aikaa työvaiheen suorittamiseen oli varattu hieman alle kolme kuukautta. Tuona aikana työssä perehdyttiin olosuhdekaapin käyttöön ja olosuhdetestaukseen. Olosuhdekaapilla testattiin Tieto-Oskarin valmistamaa tuotetta ja kaapin käytöstä laadittiin karkeat käyttöohjeet neljän eri manuaalin pohjalta.

Työn ohjelmointivaiheessa saatiin aikaiseksi toimiva testausohjelma ja ohjelmakoodia syntyi noin 1500 riviä. Ohjelman käyttöliittymä jäi ulkoasultaan karkeaksi, mutta kaikki sen toiminnot löytyvät samasta ikkunasta, mikä parantaa käytettävyyttä. Lisäksi käytettävyyttä pyrittiin parantamaan muistamalla osa käyttäjän valinnoista ja lukitsemalla käyttöliittymän elementit tilanteissa, joissa niitä ei saa tai tarvitse käyttää.

Väärinkäyttämisen mahdollisuudet pyrittiin minimoimaan rajoittamalla toimintojen käyttöä ja lisäämällä ohjelmaan logiikkoja, jotka tunnistavat virheelliset käyttötilanteet. Tässä onnistuttiin kuitenkin vain kohtalaisesti, koska jokainen lisätty toiminto lisää virheellisten käyttö- ja virhetilanteiden määrää moninkertaisesti ja usealle niistä ei keretty toteuttaa kunnollista käsittelyä. Etenkin ohjelman luomat tiedostot ovat virheellisiä ja käyttäjän toimesta tehdyt muutokset voivat rikkoa logiikan, jolla ohjelma käsittelee tiedostoja.

Toimivuuden osalta ohjelma suoriutui hyvin. Testausohjelma oli yhtäjaksoisessa testissä pisimmillään yli kolmen vuorokauden ajan. Tänä aikana ohjelmassa ei ilmennyt kaatumisia, muistivuotoja tai hidastelua. Ohjelman raskainkaan operaatio, joka suoritetaan, kun kaikki tieto kerätään samaan tiedostoon testin lopettamisen yhteydessä, ei aiheuttanut hidastelua. Tietoa kerääntyi kolmen vuorokauden ajalta noin kahden megatavun verran, kun diagnostiikkatietoja kyseltiin minuutin välein.

## 9 YHTEENVETO

Tämän insinööriyön tavoitteena oli madaltaa kynnystä olosuhdekaapin käyttöönottamisessa ja kehittää testausjärjestelmä. Olosuhdekaapin käyttöönottoon liittyvät tavoitteet saavutettiin, mutta testausjärjestelmän osalta alkuperäisiä tavoitteita jouduttiin karsimaan. Testausjärjestelmään oli alun perin suunnitelmassa toteuttaa toiminto, joka suorittaa käyttäjännitteen vaihdon olosuhdetestien aikana. Tästä jouduttiin kuitenkin luopumaan, kun työsioon varattu aika alkoi käydä vähiin. Muilta osin testausjärjestelmä on täysin toiminnallinen ja valmis käytettäväksi.

Omina henkilökohtaisina tavoitteina oli hankkia kokemusta pitkäkestoisista ja ohjelmointiluontoisista projekteista. Nämä tavoitteet toteutuivat hyvin, koska työn tilaajana toimi ennalta tuntematon yritys. Erityisesti ohjelmistokehittäjän ja tilaajan välinen vuorovaikutus korostui työn suorittamisen aikana. Kokemuksena insinööriyön tekeminen on ollut opettavainen, ja se toi esille henkilökohtaiset heikkoudet ja vahvuudet. Heikkouksista tärkein kehityksen kohde on oikean aikataulun laatiminen. Työhön laatimani aikataulu oli liian tiukka ja siinä pysyminen vaati paljon itseuria sekä lähes kaiken muun sivuuttamista. Tästä huolimatta suosittelen tuleville opinnäytetöiden tekijöille aiheen hakemista yritysmaailmasta, sillä sen antama kokemus on arvokas.

## LÄHTEET

- 1 What is environmental testing part 1. [PDF-dokumentti].  
<[http://www.espec.co.jp/english/tech-info/tech\\_info/pdf/a1/e\\_1.pdf](http://www.espec.co.jp/english/tech-info/tech_info/pdf/a1/e_1.pdf)> (luettu 2.1.2.2015)
- 2 Tietoja yrityksestä. [WWW-dokumentti].  
<<http://www.tieto-oskari.com/yritys/tietoja>> (Luettu 21.2.2015)
- 3 Loveday GC. Electronic testing and fault diagnosis. 3. ed. ed. Harlow: Longman; 1995.
- 4 Ympäristö olosuhdetestiivistelmät 248. 2004:1 cd-rom.
- 5 Nevalainen, Olavi 1998: Elektroniikkalaitteiden olosuhdetestit: Miten elektroniikka pannaan kestäväksi. Prosessori 3 (1998), s. 36 - 40.
- 6 Ympäristötestauskäsikirja. Espoo: KOTEL; 1989.
- 7 Environmental testing 2-14. Essais d'environnement. 2-14, Tests - Test N: Change of temperature =. Edition 6.0 ed. Geneva: International Electro technical Commission; 2009.
- 8 Discovery Chambers Temperature and Humidity tests. [PDF-dokumentti].  
<<http://www.logismarket.cz/ip/anamet-klimaticka-testovaci-komora-klimaticka-komora-z-rady-discovery-608661.pdf>> (Luettu 5.3.2015)
- 9 C++Builder. [WWW-dokumentti].  
<<http://en.wikipedia.org/wiki/C%2B%2BBuilder>>. (Luettu 18.3.2015)

