



LUX 34 valoisuus- ja lämpötilalähettimen tuotannon testilaitteen automatisointi

Janne Suomi

Opinnäytetyö, AMK

Toukokuu 2023

Insinööri (AMK), sähkö- ja automaatiotekniikka

Suomi, Janne

LUX 34 valoisuus- ja lämpötilälähtetimen tuotannon testilaitteen automatisointi

Jyväskylä: Jyväskylän ammattikorkeakoulu. **Toukokuu 2023**, 52 sivua.

Tekniikan ala. Sähkö- ja automaatiotekniikan tutkinto-ohjelma. Opinnäytetyö AMK.

Julkaisun kieli: suomi

Julkaisulupa avoimessa verkossa: kyllä

Tiivistelmä

Rakennusautomaatiolaitteita valmistava yritys Pro dual Oy tarvitsi tuotantoon uuden automaattisen kalibrointi- ja testilaitteen yrityksen valmistamaa LUX 34 lämpötila- ja valoisuuslähetintä varten. Toimeksiantajan toive oli testilaitte, joka ei tarvitse ollenkaan tuotantotyöntekijän huomiota kalibrointi- ja testivaiheen aikana. Automaattisen kalibrointi- ja testilaitteen toivottiin käyttävän LabVIEW-ohjelmointikielellä luotua logiikkaa.

Tutkimustyössä käytettiin kehittämistutkimukselle ominaisia tutkimusmenetelmiä. Tutkimus aloitettiin kirjoittamalla testilaitteessa käytettävien sovelluksien ja komponenttien teknisiä tietoja. LabVIEW-ohjelman ominaisuuksiin ja toiminnallisuuksiin perehdyttiin hyödyntämällä verkkolähteitä ja ohjelmasta kertovaa kirjallisuutta. Automaattisen kalibrointi- ja testilaitteen tuomaa lopputulosta tarkasteltiin suorittamalla aikahyötylaskentaa, jossa muutoksen jälkeistä valmistusprosessia verrattiin edelliseen valmistusprosessiin.

Automaattista kalibrointi- ja testilaitetta varten rakennettiin siirreltävä Pro dual Oy:n universaali NG tester tuotantotestilaitte. Tuotantotestilaitteen toimintaa varten siihen asennettiin monitoimi- I/O laite, ohjelmointilaitte, lämpötila-anturi ja sekvenssitetieto, jonka tehtävä oli suorittaa luotuja LabVIEW-ohjelmia peräkkäisenä sarjana. LabVIEW-ohjelmat luotiin sisältämään tuotteen kaikki vaadittavat kalibrointi- ja testivaiheet ja ne rakennettiin luotettavaksi ja mahdollisimman nopeiksi suoritusajoiltaan.

Tutkimustyön toimeksiantajan palautteen ja suoritettujen aikahyötylaskentojen perusteella automaattinen kalibrointi- ja testilaitte todettiin olevan tuotantotyötä nopeuttava muutos, jota voidaan käyttää jatkossa LUX 34 tuotteen valmistuksessa.

Avainsanat (asiasanat)

LabVIEW, testauslaitteet, automaatio, tehostaminen, Pro dual Oy

Muut tiedot (salassa pidettävät liitteet)

-

Suomi, Janne

Automation of the LUX 34 light and temperature transmitters production testing device

Jyväskylä: JAMK University of Applied Sciences, May 2023, 52 pages.

Engineering and technology. Degree programme in electrical and automation technology. Bachelor's thesis.

Permission for open access publication: Yes

Language of publication: Finnish

Abstract

Produal Oy, a company that manufactures building automation equipment, had a need for a new automatic calibration and testing device for their LUX 34 temperature and lightness transmitter product that is manufactured by the company. The client's wish was a testing device that does not need any attention from a production worker during the calibration and testing phase. Client's wish was that the automatic calibration and testing device would use code created by LabVIEW as its logic.

Research methods specific to development research were used in the research work. The research started by mapping out all technical data of the applications and components needed in the testing device. The features and functionalities of the LabVIEW program were familiarized by using online sources and reliable literature about the program. The result brought by the automatic calibration and testing device was examined by performing a time benefit calculation, where the manufacturing process after the change was compared to the previous manufacturing process.

A moveable Produal's universal test platform NG Tester was built for the automatic calibration and testing device. For the operation of the test device, a multifunctional I/O device, a programming device, a temperature sensor and a sequence file of which task was to execute the created LabVIEW programs in a sequential series were installed into the device. The LabVIEW programs were created to include all of the required calibration and test phases of the product, and the programs were built to be reliable and as fast as possible in their execution times.

Based on the feedback from the client of the research work and the performed time benefit calculations, the automatic calibration and testing device was found to be a good change in the manufacturing process that speeds up the production work, which can be used in the future in the manufacture of the LUX 34 product.

Keywords/tags (subjects)

LabVIEW, testing equipment, automation, improving efficiency, Produal Oy

Miscellaneous (Confidential information)

-

Sisältö

Käsitteet	6
1 Johdanto	7
1.1 Opinnäytetyön tausta	7
1.2 Työn rajaus ja tavoitteet	7
1.3 Toimeksiantaja	8
2 Tutkimusasetelma	8
2.1 Kehittämistutkimus	8
3 Automatisoitu tuotantotestaus	9
3.1 Automaattinen testilaite	10
4 Laitteisto	11
4.1 LUX 34 valoisuus- ja lämpötilalähetin	11
4.2 LabVIEW	12
4.2.1 LabVIEW historia	12
4.2.2 LabVIEW ohjelmointikieli	12
4.2.3 LabVIEW käyttöliittymä ja työkalut	13
4.2.4 LabVIEW laitteisto ja ohjaimet	14
4.2.5 LabVIEW VI	15
4.2.6 LabVIEW koodin ohjelmoiminen	19
4.2.7 LabVIEW lohkoakaavion tietovirta	21
4.3 NI Measurement & Automation explorer (NI MAX)	22
4.4 MSP-FET	24
4.4.1 Laitteen kuvaus ja ominaisuudet	24
4.5 NI USB-6363	25
5 Kalibrointi- ja testilaitteen vaatimukset	26
5.1.1 Testilaitteen käytettävyys	27
5.1.2 Testilaitteen toiminnan luotettavuus	27
5.1.3 Testilaitteen LabVIEW ohjelma	28
6 Testilaitteen toteutus	29
6.1 NG tester	29
6.2 Testisekvenssi	31
6.3 LUX 34 piirikortin ohjelmoiminen	33
6.4 LabVIEW ohjelmakokonaisuus	35
6.5 Ohjelmallisuuden vaatimukset	36

6.5.1	LUX 34 mikroprosessorin erasointi ja ohjelmointi	36
6.5.2	LUX 34 lämpötilan kalibrointi	37
6.5.3	LUX 34 valoisuuden kalibrointi	40
6.5.4	LUX 34 laadunvarmistus	41
7	Tulokset	45
8	Pohdinta	47
	Lähteet	49
	Liitteet	51
	Liite 1. LUX 34 data sheet.....	51
	Liite 2. NG tester osarakenne.....	52

Kuviot

Kuvio 1	LUX 34 (Ulkovaloisuuslähetin LUX 34 n.d.)	11
Kuvio 2	LabVIEW VI Manager, Project Explorer näkymä (Using the Project Explorer Window (Unit Test Framework Toolkit) 2023).....	14
Kuvio 3	Front Panel (LabVIEW Front Panel Explained 2023)	16
Kuvio 4	Yksinkertainen lohkokaavio (LabVIEW Block Diagram Explained 2023).....	17
Kuvio 5	Lohkokaaavion liittimien heijastuminen front paneliin (LabVIEW Block Diagram Explained 2023)	18
Kuvio 6	Lohkokaaavion funktiopaletti (LabVIEW Block Diagram Explained 2023).....	18
Kuvio 7	Esimerkki ohjelmointielementeistä (What is LabVIEW? n.d.)	20
Kuvio 8	LabVIEW johtimien tietotyypit (What is LabVIEW? n.d.).....	21
Kuvio 9	Tietovirran havainnollistaminen (What is LabVIEW? n.d. muokattu)	22
Kuvio 10	NI MAX laitevalikko (Create Tasks for NI-DAQmx Devices in NI MAX 2023)	23
Kuvio 11	MSP-FET ja liitäntäkaapelit (MSP-FET n.d.).....	25
Kuvio 12	Monitoiminen I/O laite (USB-6363 n.d.)	26
Kuvio 13	Suunnittelukuva NG testerin rakenteesta LUX 34 fixtuurilla.	29
Kuvio 14	LUX 34 testisekvenssin esimerkkirakenne	31
Kuvio 15	Havainnoiva esimerkki NG tester pääkäyttöliittymästä	32
Kuvio 16	Järjestelmätason havainnointi sekvenssin ja pääkäyttöliittymän toiminnasta valmistusprosessissa	33
Kuvio 17	MSP-FET kytkentäohje laitteen kääntöpuolella (MSP-FET n.d.)	34
Kuvio 18	LUX 34 piirikortti kytkettynä prototyyppi fixtuuriin.....	34
Kuvio 19	MSP430Flasher in käyttäminen erasointiin LabVIEW:ssä.....	37
Kuvio 20	LUX 34 lämpötila jänniteviestin skaalaaminen celsius asteiksi.....	39

Kuvio 21 Lämpötilojen vertailulaskenta esimerkki LabVIEW:ssä.....	39
Kuvio 22 Loop-lohkojen looginen suoritus kalibrointi VI:n sisällä	40
Kuvio 23 Lämpötilan laadunvarmistuksen vertaustoiminto LabVIEW:ssä	42
Kuvio 24 Laadunvarmistuksen esimerkkirakenne sekvenssitiedostossa.....	43
Kuvio 25 LUX 34:n skaalauksen vaikutus jännitteenmäärään	44
Kuvio 26 Aikahyötylaskenta uudelle valmistusprosessille.....	46

Taulukot

Taulukko 1 Uuden testiprosessin vaikutukset tuotantoaikaan.	46
--	----

Käsitteet

LabVIEW on lyhenne englanninkielisistä sanoista: Laboratory Virtual Instrument Engineering Workbench. LabVIEW on National Instruments yrityksen luoma graafinen ohjelmointikielen käyttöliittymä (What is LabVIEW? n.d.).

NG tester on Pro dual Oy:n luoma universaali tuotantotestilaitte rakenne.

Fixtuuri on NG testeriin liitettävä tuotekohtainen osa, joka hyväksyy tai hylkää testattavan piirikortin.

VI on LabVIEW-ohjelmointikielessä keskeisin elementti, joka sisältää luodun ohjelman koodin (What is LabVIEW n.d.).

Tuotantotestilaitte on käsite, jolla kuvataan toimeksiantajan tuotantotiloissa sijaitsevaa, valmistetun tuotteen toimintaa koestavaa laitetta.

1 Johdanto

1.1 Opinnäytetyön tausta

Tämän työn tarkoitus on olla toimeksiantajalle yksi askel eteenpäin siirryttäessä tehokkaampaan tuotantomalliin automatisoinnin avulla. Yrityksessä on lähivuosina siirrytty muutamien uusien tuotteiden valmistuksessa automaattisiin tuotannotestauslaitteisiin. Edellisten automatisointikohteiden on todistettu tuovan suurta etua tuotannon tehokkuuteen. Yritys tarjoaa todella suuren valikoiman erilaisia rakennusautomaatiolaitteita, joten tuotantoon on päätetty kehittää suuri määrä uusia automaattitestilaitteita parantamaan myös vanhempien laitteiden valmistustehokkuutta.

Tässä opinnäytetyössä tutkittiin automaattisen kalibrointi- ja testauslaitteen tuomaa vaikutusta tuotannon tehokkuuteen. Toimeksiantona oli uuden kalibrointi- ja testilaitteen rakentaminen Pro-dual Oy:n valmistamalle LUX 34 valoisuus ja lämpötilalähettimelle. Tuotteen nykyinen tuotantotestausmenetelmä vaatii tuotantotyöntekijältä paljon aikaa ja manuaalista työtä testiprosessissa ja tästä syystä uuden testilaitteen suunnittelu oli tärkeää toimeksiantajalle.

1.2 Työn rajaus ja tavoitteet

Tuotannon tehokkuuden nostaminen kalibrointi- ja testilaitteen automatisoinnilla edellyttää testilaitteen prosessin mahdollisemman tarkkaa suunnittelua ja toteutusta. Tutkimus keskittyi LabVIEW-ohjelmointikielen hyödyntämiseen automaattisen kalibrointi- ja testausmenetelmän luomisessa. Yhden tuotantotyöntekijän käyttämää aikaa saadaan vapautettua valmistettavan laitteen muuhun kokoonpanoon, kun kalibrointi- ja testilaitte on täysin automaattinen. Edellisen tuotantotestilaitteen huono puoli oli, että tuotantotyöntekijän oli itse säädeltävä oikeat kalibrointi-arvot tuotteelle ja sen jälkeen todettava tuote toimivaksi. Tuotteita valmistetaan usein suuria määriä kerrallaan, joten testaus vie suuren ajan pois itse laitteen kokoonpanolta.

Toimeksiantona opinnäytetyölle oli luoda uusi automaattinen kalibrointi- ja testauslaite, jonka tehtävänä on suorittaa tuotteen ohjelmointi, kalibrointi ja tarkistus ilman tuotantotyöntekijän vaatimaa huomiota. Testilaitteen automatiikka oli tarkoitus luoda käyttämällä LabVIEW:n graafista ohjelmointikieltä ja suorittaa LabVIEW:llä luotuja koodeja yhtenäisenä sekvenssinä. Työ rajattiin

keskittymään pääsääntöisesti automaattisen testilaitteen LabVIEW logiikan valmistamiseen. Pääta-
voitteena oli saavuttaa virheetön kalibrointi ja testaus käyttämällä LabVIEW-ohjelmointikielellä
luotua sekvenssiä. Lisäksi tavoitteena oli luoda valmis tuotantotestilaitte ja saavuttaa LUX 34:n val-
mistukseen ajallinen hyöty uudella testilaitteella.

1.3 Toimeksiantaja

Opinnäytetyön toimeksiantajana oli Pro dual Oy. Pro dual Oy on vuonna 1988 perustettu rakennus-
automaatiolaitteita ja asiantuntijapalveluita tarjoava yritys. Yrityksellä on yli 30 vuoden kokemus
rakennusautomaatioalalta, jonka aikana yrityksen tuotevalikoima on kasvanut tarjoamaan yli 1000
eri tuotetta, kuten säätimiä, mittausantureita, lähettimiä ja järjestelmäkomponentteja. Pro dual
pyrkii viemään mittaus- ja säätölaitteiden kehitystä jatkuvasti eteenpäin rakennusautomaatio-
alalla. (Tietoa meistä 2021)

2 Tutkimusasetelma

2.1 Kehittämistutkimus

Työssä tavoitteet olivat saavutettavissa käyttämällä tutkimusmuotona kehittämistutkimusta. Ke-
hittämistutkimus noudattaa muiden tutkimusmuotojen mukaisesti samaa tutkimusprosessillista
kaavaa. Kehittämistutkimus alkaa tutkimusongelmasta, jolle on toivottu löytyvän ratkaisu. Tutki-
musongelma muodostuu aina käytännön ilmiöstä. Ilmiöön liitetään aina jokin ongelma, jota yrite-
tään ratkaista selvittämällä ongelman konteksti ja syy. Kehittämistutkimus tutkimusmuotona pyrkii
tutkimusongelman poistamiseen tai muuten parantamaan ongelmallista olotilaa. Tutkimusongel-
mat pyritään aina muuttamaan tutkimuskysymyksiksi, joihin tutkimuksessa pyritään vastaamaan.
(Kananen 2015, 11–12.)

Tutkimuskysymykset tässä opinnäytetyössä olivat:

- Miten LabVIEW ohjelmaa voidaan käyttää automaattisen testilaitteen logiikan luomisessa?
- Miten paljon on mahdollista tehostaa tuotantoa automatiikan avulla?
- Miten rakennetaan luotettavuutta automaattisen testilaitteen mittauksiin?

Kehittämistutkimus on monimenetelmäinen tutkimusstrategia. Kehittämistutkimusta ei luokitella omaksi erilliseksi tutkimusmenetelmäksi, vaan se sisältää joukon erilaisia tutkimusmenetelmiä. Tutkimusmenetelmiä yhdistetään ja käytetään kehittämiskohteen ja tutkimusongelman mukaisesti. Tarpeen mukaan kehittämistutkimus käyttää tutkimusotteessaan kvalitatiivisia ja kvantitatiivisia menetelmiä tutkimuksen ratkaisemiseksi. (Kananen 2015, 33.)

Tutkimustulosten luotettavuusarviointi on todettu haasteelliseksi kehittämistutkimuksessa kuten muissakin laadullisissa tutkimusmenetelmissä. Kehittämistutkimuksessa onnistumisen mittarina toimii muutoksen toteutuminen tahdotussa käytännön työelämän ongelmassa. Tieteellisesti tarkastellaan aina, onko tutkimuksen muutos seurausta tutkimuksen alussa mainituista asioista eikä ulkopuolisista vaikutteista. Muutosta arvioidaan asetetuilla mittareilla, joiden pitää olla valideja, eli mittaavan ongelman kannalta olennaisia asioita. (Kananen 2015, 59.)

Tässä kehittämistutkimuksessa tietoperustaa on rakennettu käyttämällä verkkojulkaisuja, ammattikirjallisuutta ja toimeksiantajan materiaalia. Tutkimus on tehty eettisiä menetelmiä noudattaen. Aineistonkeruussa on valittu lähdekriittisesti aineistoja, jotka voidaan käsittää täyttävän kriteerit esitettyjen faktojen oikeellisuudesta. Testilaitteen suunnittelu- ja toteutusprosesseissa on huomioitu toimeksiantajan esittämät toiveet koskien laiterakennetta ja käytettäviä materiaaleja. Tutkimuksen lopputuloksia arvioidaan uuden testilaitteen aikahyötylaskennan ja toimeksiantajan palautteen perusteella.

3 Automatisoitu tuotantotestaus

Tuotantotiloissa suoritettava tuotantotestaus tarkoittaa laadullista tuotteen tarkastelua, jossa varmistetaan, että valmistettu tuote on vaadituilta ominaisuuksiltaan toimiva. Tuotteiden laatu saadaan tasaiseksi, kun hyödynnetään tuotantotestausta tuotantoprosessin aikana. Ilman tuotantotestauksen suorittamista voidaan odottaa kalliita takaisinkutsuja, takuuvaatimuksia ja tuoteturvallisuusongelmia. Tuotantotestauksella saadaan myös varmistettua, että tuotemerkki säilyttää arvonsa, koska tiedetään, että asiakkaalle toimitettu tuote on varmasti toimiva. (Production line testing and quality assurance n.d.)

Tuotantoprosessia nopeuttaakseen tuotantotestauksesta on nykyaikana pyritty tekemään mahdollisimman automaattista. Automatisointi on monesta näkökulmasta hyvä vaihtoehto manuaaliselle

testaukselle, koska sillä saadaan lyhennettyä valmistusprosessin sisältämää testiaikaa, vältetään inhimilliset virheet ja optimoidaan tuotantotyöntekijän työvoima. (Test Automation Best Practices n.d.)

Automaattisuuden lisääminen tuotteen valmistusprosessiin on tärkeää ajallisen säästön kautta. Ennen automatisoinnin lisäämistä testiprosessiin täytyy suunnittelijan ottaa selvää mitä vaiheita valmistusprosessi sisältää ja kuinka kauan tuotteen tuotantotestaussykli kestää. Robotiikkaa on järkevintä hyödyntää tuotteissa joiden valmistusprosesseissa osien sijoittaminen tai kokoonpanovaiheet vievät suurimman osan ajasta. Perusautomaation lisääminen testiprosessiin on järkevintä niissä valmistusprosesseissa, joissa valmistusaika on pitkä testitekniikan takia. Testitekniikat, jotka mittaavat eri pisteitä tai asettavat erilaisia kalibrointeja tuotteille ovat merkittävästi nopeutettavissa perustestiautomaatiolla. (Test Automation Best Practices n.d.)

3.1 Automaattinen testilaitte

Tietokoneistettuja testilaitteita, jotka suorittavat elektronisten laitteiden laatu-, toiminnallisuus- ja suorituskykytestejä kutsutaan automaattisiksi testilaitteiksi. Automaattisella testilaitteella pyritään vähentämään ihmiseltä vaadittua vuorovaikutusta testiprosessissa yleisesti korvaamalla perinteisiä manuaalisia testilaitteita. Automaattisia testilaitteita käytetään yleisesti testaamaan integroituja piirejä, painettuja piirilevyjä, kiintolevyasemia tai autoteollisuuden käyttämiä järjestelmiä. (Daniel, B 2021.)

Yleinen automaattisen testilaitteen rakenne sisältää viisi pääkomponenttia: laitteiston, testauslaitteiston, ohjelmiston, testiyhteyden ja signaalilähteen. Laitteistolla tarkoitetaan testilaitteeseen asennettavia palvelimia, työasemia, virtalähteitä, digitaalisia- ja analogisia portteja ja liitäntämoduuleita. Ohjelmistona käytetään käytössä olevaa sovellusta, jolla suoritetaan testilaitteen testiprosessin kehittäminen. Testauslaitteistona ja signaalilähteenä voidaan käyttää esimerkiksi monitoimi- I/O laitetta. Testiyhteys luodaan testattavan tuotteen ja testilaitteen välille käyttämällä tarvittavia yhteyden muodostavia kontakteja, kuten metallikontakteja. (Daniel, B 2021.)

Nämä komponentit yhdistetään yhdeksi testiasema kokonaisuudeksi, joka voi vaihdella suuresti kooltaan ja kannettavuudeltaan. Automatisoidut testiratkaisut käyttävät kuitenkin aina käyttökohteessa tarvittavia komponentteja, jotka täytyy valita testattavien suureiden ja tuotteiden perusteella. (Daniel, B 2021.)

4 Laitteisto

4.1 LUX 34 valoisuus- ja lämpötilalähetin

LUX 34 mittaa kahta eri suuretta, valoisuuden voimakkuutta ja lämpötilaa. LUX 34 mittaa valoisuuden voimakkuutta välillä 0–10 000 Luxia ja lämpötilaa astevälillä -50 ...+50 °C. Mitatut arvot laite muuttaa lineaariseksi jännitelähdöksi 0–10 voltin alueella. Laite sisältää piirikortilla koodausliittimen S1, jonka avulla saadaan säädettyä halutessaan LUX 34 laitteen valoisuuden mittausasteikko joko 0–10 000 Lux tai 0–1000 Lux. (LUX 34 Ja LUX 34-100 Valoisuus- ja lämpötilalähetin 2010.)

Laitteen jännitelähtöjen avulla saadaan ohjattua haluttuja säätö- tai valvontajärjestelmiä. Valmis LUX 34 tuote (ks. kuvio 1.) mahdollistaa säätämisen järjestelmissä lämpötilan ja valoisuuden mukaan. LUX 34 tuotteen asennus tehdään käyttökohteessa ulkoseinälle siten, että tuote on mahdollisimman suojassa suoralta auringonpaisteelta tai mittaustietoa häiritsevältä ulkopuoliselta valolta. Lisätietoja LUX 34 laitteesta löytyy liitteestä 1. (LUX 34 Ja LUX 34-100 Valoisuus- ja lämpötilalähetin 2010.)



Kuvio 1 LUX 34 (Ulkovaloisuusläheter LUX 34 n.d.)

4.2 LabVIEW

LabVIEW on lyhenne englanninkielisistä sanoista: Laboratory Virtual Instrument Engineering Workbench. LabVIEW on National Instruments yrityksen luoma graafinen ohjelmointikielen käyttöliittymä, joka on otettu ensimmäisen kerran käyttöön vuonna 1986. Silloin LabVIEW:n tarkoitus oli toimia insinöörien työkaluna automaattisten mittauksien tulkinna. Graafinen ohjelmointikieli suunniteltiin olemaan yhtä suuri kehitysaskel insinööreille ja tutkijoille kuin finanssianalytikoille kehitetty laskentataulukko oli aikanaan. (What is LabVIEW? n.d.)

4.2.1 LabVIEW historia

Jeff Kodowsky nähdään LabVIEW-sovelluksen kehittäjänä. Kodowsky käytti 80-luvulla Applen valmistamaa tietokonetta, joka mahdollisti parhaimman graafisen ulostulon verrattuna mihinkään muuhun tietokoneeseen siihen aikaan. Kodowskyn idea oli hyödyntää tätä graafista ulostuloa nopeuttaakseen laitteistojen automatiikan ohjausta ja tästä kehittyi LabVIEW-sovelluksen idea ja kehittämisen alku. (What is LabVIEW? n.d.)

4.2.2 LabVIEW ohjelmointikieli

LabVIEW on sovellus, joka antaa mahdollisuuden ohjelmoida G-kielillä olevia ohjelmia. G-kieli on National Instruments yrityksen kehittämä graafinen ohjelmointikieli, joka oli alun perin tarkoitettu hyödynnettäväksi pelkästään GPIB- sarjaliikenteen kautta kommunikointiin. Nykyään G-kieli on kehittynyt mahdollistamaan sen käytön myös mm. automaattisissa testisovelluksissa ja tiedonkeruun suorittamisessa. (What is LabVIEW? n.d.)

LabVIEW on kehittynyt ajan myötä suuresti ja se ymmärretään nykyään G-kielen synonyymina, vaikka sovellus on pohjimmiltaan graafinen käyttöliittymä itse G-kielille. LabVIEW:n G-kielinen koodi ei kuitenkaan ole sama asia, kun G-koodi, jota käytetään CNC-koneissa. LabVIEW-sovellus tarjoaa G-kielen ohjelmoinnin lisäksi nykyään myös virheenkorjauksen, monisäikeen, laitteistohallintaa ja tehdyn sovelluksen käyttöliittymän. Nämä lisätoiminnot mahdollistavat LabVIEW-ohjelman käytön siten, että yhdistetään monia eri toiminnallisuuksia ja laitteita yhden helposti käytettävän sovelluksen sisään. (What is LabVIEW? n.d.)

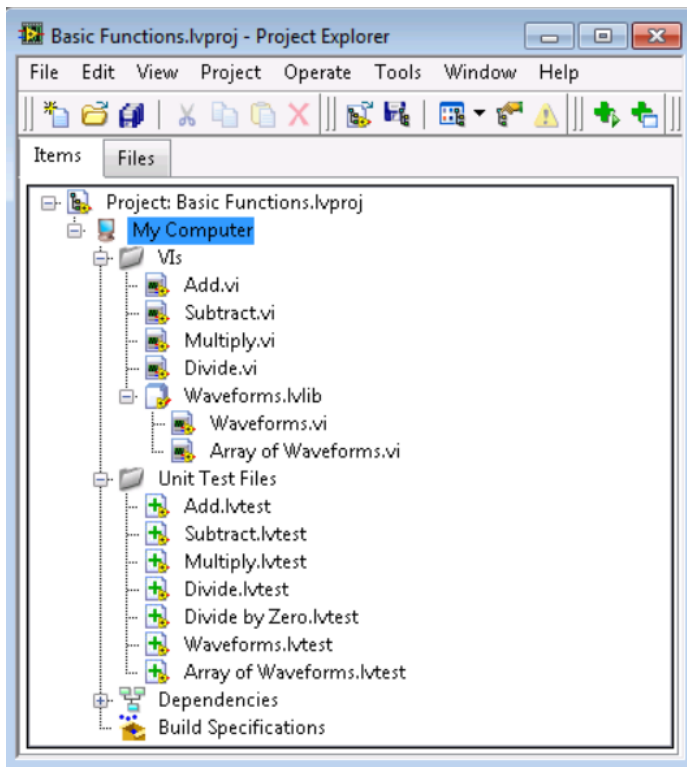
4.2.3 LabVIEW käyttöliittymä ja työkalut

LabVIEW-käyttöliittymä voidaan asentaa useimpiin tietokonejärjestelmiin mm. Windows, Linux ja OS X (Apple). LabVIEW-käyttöliittymää voidaan ajatella kokonaisuutena, joka sisältää kaikki ohjelmien luontia varten olevat työkalut. LabVIEW-ohjelman käyttämä G-koodi on yksi osa tästä kokonaisuudesta, G-koodi vastaa logiikan suorituksesta LabVIEW-ohjelmassa. Käyttöliittymä sisältää paljon toimintoja ja työkaluja, jotka eivät suoraan liity itse G-koodiin vaan helpottavat projektin luontia, ylläpitoa sekä virheenkorjausta. Näitä työkaluja ovat LabVIEW VI Manager, LabVIEW Tools Network, Templates & Sample projects, Example finder ja Debugging tools. (What is LabVIEW? n.d.)

Templates & Sample projects työkalu on tarkoitettu avuksi projektin alussa. Työkalu antaa valikoiman malleja ja valmiita esimerkkiprojekteja, joita sovelluksen kehittäjä voi käyttää hyödykseen työn aloituksessa. Malleista nähdään esimerkki ratkaisuja, joita voidaan suoraan muokata omaan käyttötarkoitukseen tai hyödyntää oman projektin ratkaisuisissa. Example finder toimii samalla tavalla, mutta on tarkoitettu työn edetessä ongelmatilanteiden ratkaisemiseen. Esimerkkiratkaisuja saadaan haettua satojen eri tilanteiden ratkaisemiseksi ja tuloksena haku antaa pieniä kuvaleikkeitä ratkaisuisista. (What is LabVIEW? n.d.)

Ongelmatilanteiden lisäksi jokaiselle projektille virhetilanteet ovat osa ohjelman kehitysvaihetta. LabVIEW:in toiminto Debugging tools tarjoaa monia nopeita virheenkorjaus menetelmiä. Menetelmiä, joilla virhetilanne voidaan paikantaa ovat indikaattorit, raja-arvot, pysäytyspisteet ja yksivaiheinen suoritus. Yksivaiheisella suorituksella voidaan hidastaa koodin suoritusnopeutta ja seurata ohjelman ajoa. Joskus monimutkaisiin ja vaativiin ohjelmakokonaisuuksiin on ladattava lisätyökaluja, joilla saadaan suoritettua ohjelmassa haluttu toiminnallisuus. LabVIEW Tools Network tarjoaa ladattavaksi lisätyökaluja, joista osa on National Instrumentsin ja osa National Instruments kumppaneiden tarjoamia. (What is LabVIEW? n.d.)

LabVIEW VI Manager, eli Project Explorer on keskeinen osa jokaiselle LabVIEW projektille. Project Explorer näkymää voidaan kuvata, projektin tutkimusnäyttönä (ks. kuvio 2). Kun projekti aloitetaan, tallennetaan Project Exploreriin kaikki kirjastot, koodiosastot, datatiedostot ja laitteistoon käytettävä ohjelmisto ylläpitoa ja käyttöä varten. (What is LabVIEW? n.d.)



Kuvio 2 LabVIEW VI Manager, Project Explorer näkymä (Using the Project Explorer Window (Unit Test Framework Toolkit) 2023)

4.2.4 LabVIEW laitteisto ja ohjaimet

LabVIEW ohjelmalla on kyky ohjata ja vastaanottaa viestejä yhteensopivilta elektronisilta laitteilta. Kyky vuorovaikutukseen laitteiden kanssa vaatii rajapinnan, jota LabVIEW-ohjelma käyttää muodostaessaan yhteyden järjestelmään kytkettyyn laitteeseen. Nämä rajapinnat on nimetty LabVIEW-ajureiksi ja LabVIEW-käyttöliittymä sisältää suuren määrän eri ajureita, jotka ovat valmiina käyttöä varten. Yksittäinen ajuri tarjoaa yhteyden LabVIEW-ohjelman ja käytettävissä olevan laitteen välille. Kun yhteys on luotu voi LabVIEW-ohjelmalla ohjata ja saada tietoja reaaliajassa järjestelmään yhdistetyltä laitteelta. (What is LabVIEW? n.d.)

LabVIEW-laiteajuri koostuu ohjelmistorutiineista. Ohjelmistorutiinit antavat mahdollisuuden laitteille lähettää LabVIEW:n järjestelmään tietoa sekä vastaanottaa sitä. Kokonaisajuri sisältää erilaisia alirutiineja, jotka antavat LabVIEW:lle mahdollisuuden suorittaa yksittäisiä tehtäviä. Alirutiinien suorittamia tehtäviä ovat: konfigurointi, tiedon kirjoittaminen, lukeminen ja välineen

käynnistäminen. LabVIEW antaa ajurille komennot, jotka muunnetaan laitteelle laitekohtaisiksi ohjeiksi. Ohjeet saadaan siirrettyä laitteelle käyttämällä USB-, Wi-Fi, Ethernet-, tai jotain muuta liitäntää, jota käytössä oleva laite hyödyntää. (What is LabVIEW? n.d.)

National Instruments on tuottanut suuren määrän LabVIEW:n ajureista, mutta ajuri kirjastoa on valmistettu myös kolmansien osapuolien, kuten laitevalmistajien toimesta. Osa LabVIEW-ajureista on toiminnaltaan yksinkertaisia ja sisältää rajoitetun määrän ohjeita ja ominaisuuksia. Toiminnallisuuksiltaan suuremmat ajurit voivat myös sisältää esimerkiksi etäyhteydellä toimivia säätimiä ja mahdollistaa useat eri laitekohtaiset liityntätavat. LabVIEW tarjoaa käyttöliittymässä monia ajureita ilmaiseksi käyttäjälleen. (What is LabVIEW? n.d.)

LabVIEW:n kautta on mahdollista ladata tuhansia National Instrumentsin tarjoamia ajureita ja yritys on laatinut mahdollisuuden ladata ajurit myös käyttämällä ID- tai National Instruments Driver Network palvelua. National Instruments kustantaa myös useiden laitevalmistajien kehittämät ajurit, jotka valmistetaan suoraan ladattaviksi heidän omien palveluidensa kautta. Monessa eri organisaatiossa käytetään LabVIEW:tä, joten laitevalmistajat voivat käyttää ajureiden kehittämistä itselleen hyödyllisenä kilpailukyvyyn ylläpitämisenä. Osa kehittyneistä ajureista voi kuitenkin vaatia käyttäjältä kuluja. Ajureiden kustannus riippuu laitevalmistajasta sekä ajurin kehittäjästä. (What is LabVIEW? n.d.)

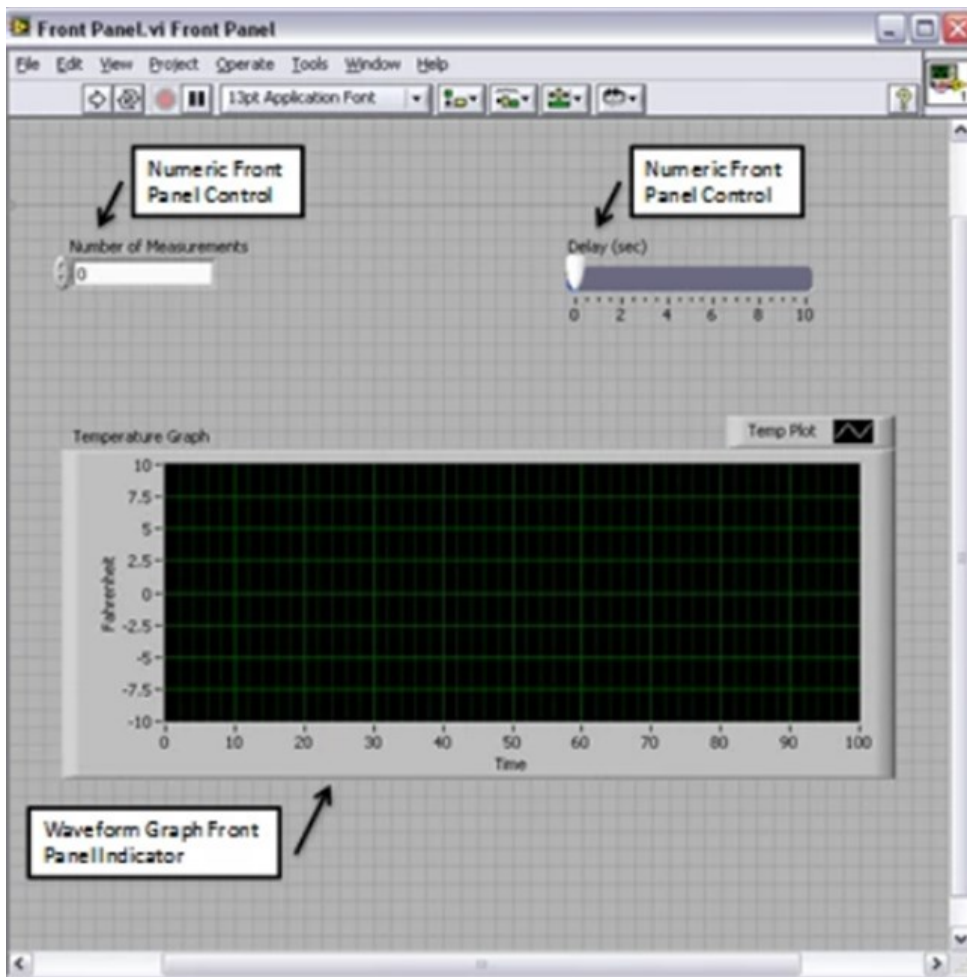
LabVIEW:n ja käytettävän välineen ajuri muodostuu Virtual Instrumentsin tai virtuaalivälineiden ryhmästä. LabVIEW Virtual Instrument vastaa aina tietystä toiminnosta, kuten konfiguroinnista, kirjoittamisesta tai laitteen lukemisesta. LabVIEW:n lähestymistavan etuutena on ohjelmoinnin helpottuminen käytettävissä laitteissa. Lähestymistapa tuottaa ratkaisun matalan asteen kielellä ohjelmoimiseen, joka helpottaa laitteiden käyttöönottoa. (What is LabVIEW? n.d.)

4.2.5 LabVIEW VI

LabVIEW virtual instrument, lyhenteenä VI, on kaikista keskeisin elementti LabVIEW-sovelluksen käyttöliittymässä. LabVIEW VI sisältää kaksi tärkeää komponenttia, etupaneelin ja lohkoakaavion. Etupaneelin ja lohkoakaavion avulla VI antaa mahdollisuuden käyttäjälleen toteuttaa visuaalisia algoritmeja ja toimintoja, joita käytetään suuremmassa ohjelmassa tai pelkästään yksittäisessä

VI:ssä. Kuten muutkin tunnetut ohjelmakielet, myös LabVIEW sisältää alirutiineja, joita sovelluksessa toteutetaan VI:den avulla. (What is LabVIEW? n.d.)

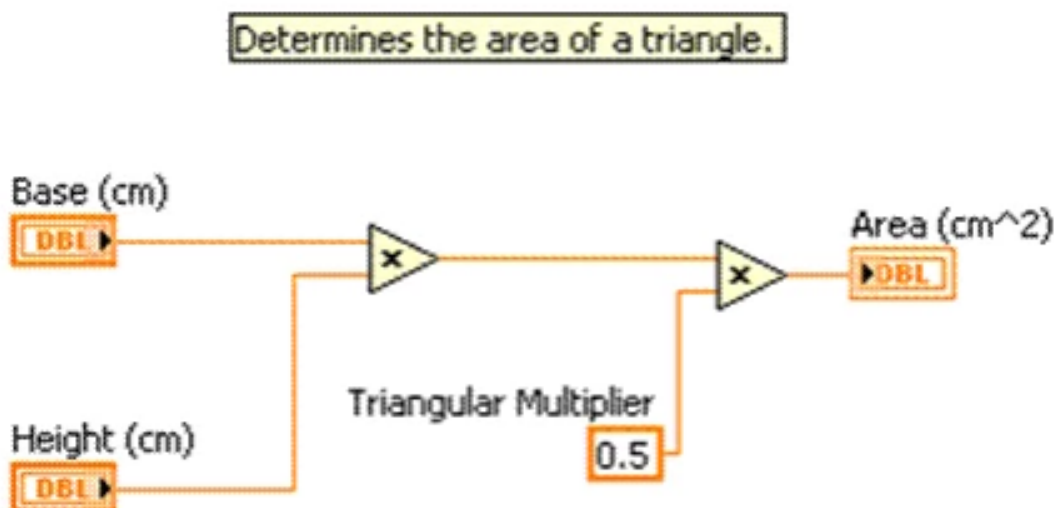
Etupaneeli on VI:n käyttöliittymä ja etupaneelin näkymää voidaan havainnollistaa esimerkkikuvan avulla (ks. Kuvio 3) (Front Panel Explained 2023). Etupaneelin ikkunasta käyttäjä voi nähdä erilaisia indikaattoreita, säätimiä ja mittareita. Sovelluksen kehittäjä on asettanut tarvittavat indikaattorit ja halutut ilmaisimet ohjelmaa luodessa etupaneeliin kuvastamaan ohjelmassa tapahtuvia ilmiöitä tai mahdollistaakseen käyttäjälleen ohjelman kontrolloimisen. Käyttäjä saa mahdollisuuden olla vuorovaikutuksessa VI:n kanssa säätämällä toiminnassa olevan VI:n säätimiä ja lukemalla takaisin-kytkentä tietoja. Etupaneeli voidaan rakentaa ohjelman alussa sijoittamalla heti kaikki tarvittavat elementit suunnitelluille paikoilleen. (What is LabVIEW? n.d.)



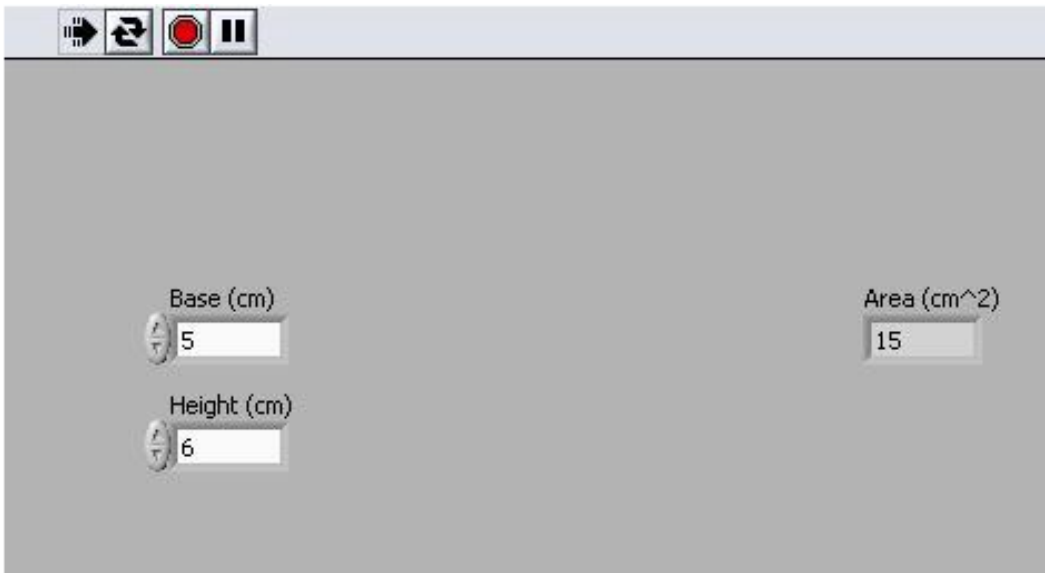
Kuvio 3 Front Panel (LabVIEW Front Panel Explained 2023)

National Instruments tuo esille paljon etupaneelin käyttöä helpottavia ohjeita verkkosivuillaan (ks. LabVIEW Front Panel Explained 2023). VI:n etupaneeli tulee näkyville aina kun käyttäjä avaa uuden tai valmiin VI:n. Etupaneelin yläosassa sijaitsee työkalurivi, joka sisältää monia eri toimintoja muokkausta varten. Controls paletti on etupaneelin nopea ja hyödyllinen aputyökalu. Controls paletin avaaminen onnistuu painamalla hiiren toista painiketta missä vain kohdassa paneelia. Paletti sisältää mittarit, säätimet ja muut tarvittavat indikaattorit, jotka voidaan sijoittaa paneeliin. (LabVIEW Front Panel Explained 2023.)

Lohkokaavio eli Block Diagram on toinen VI:n komponentti ja sen sisällä sijaitsee ohjelman graafinen esitys (ks. Kuvio 4). Lohkokaavio on tarkoitettu erotella lähdekoodi käyttöliittymästä käyttämällä mahdollisimman yksinkertaista ja loogista tapaa. Yksinkertaisuuden takaamiseksi lohkokaaaviossa on myös ohjelman rakenteeseen kuuluvia liittimiä, jotka peilaavat identtiset tiedot niiden vastakappaleisiin etupaneeliobjekteissa. Toiminnon etuutena on, että lohkokaaavioon suoritettavat muutokset heijastavat etupaneelin vastinkappaleisiin ja myös etupaneelin muutokset suoraan ohjelmaan. LabVIEW ohjelman sisältämä graafinen lähdekoodi on aina sijoitettuna myös lohkokaaavioon. Kuvion 4. mukaiset input ja output liittimet heijastuvat etupaneeliin kuvion 5. mukaisella tavalla, antaen käyttäjälle mahdollisuuden syöttää arvoja ja vastaanottaa niiden mukainen lopputulos. (LabVIEW Block Diagram Explained 2023.)

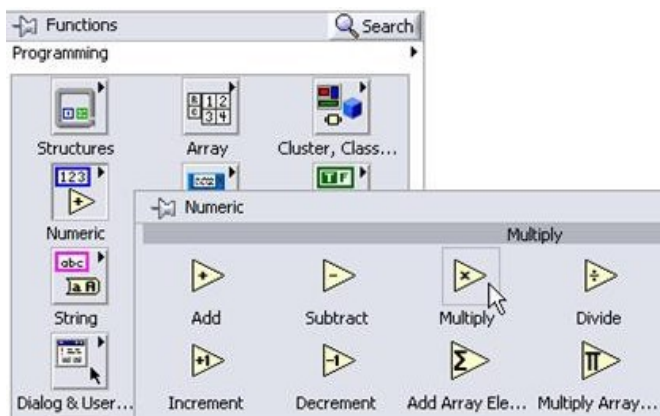


Kuvio 4 Yksinkertainen lohkokaaavio (LabVIEW Block Diagram Explained 2023)



Kuvio 5 Lohkokaavion liittimien heijastuminen front paneliin (LabVIEW Block Diagram Explained 2023)

Lohkokaavion voidaan katsoa esittävän ohjelman toimintaa visuaalisesti. Tämän esitystyylin avulla saadaan myös samalla ymmärrettyä tieto ohjelman toiminnallisuuksista. Ohjelman sisäinen vuokaavio voidaan olettaa olevan identtinen lohkokaaviossa olevan koodin kanssa. Lohkokaavion käyttöön on luotu monia helpottavia toimintoja, joiden avulla käyttäjän on nopeampi työskennellä lohkokaaavion kanssa. Lohkokaaviossa rakennettavaan ohjelmaan on olemassa LabVIEW:n funktiopaletti, jonka on tarkoitus tarjota kaikki ohjelmaan tarvittavat elementit ja toiminnot (ks. kuvio 6.). Funktiopaletin käyttäminen tekee ohjelmointityöstä tarkkaa ja nopeaa tarjoamalla ohjelmoijalle kaikki halutut toiminnot yhden valikon sisään. (What is LabVIEW? n.d.)



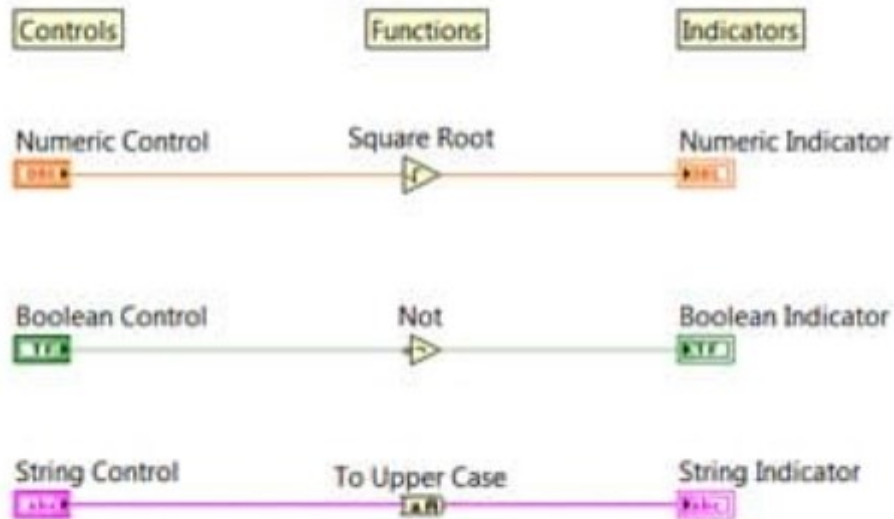
Kuvio 6 Lohkokaavion funktiopaletti (LabVIEW Block Diagram Explained 2023)

4.2.6 LabVIEW koodin ohjelmoiminen

LabVIEW:ssä ohjelmointi eroaa yleisesti käytössä olevista tekstipohjaisista ohjelmointikielistä. Ohjelmoiminen LabVIEW:ssä vaatii käytettävän erilaisia graafisia tekniikoita, jotka helpottavat esityksien rakentamista. Ohjelman graafista esitystä tarkastellessa voidaan ohjelman kulkua verrata varsinaisen prosessin kulkuun. (What is LabVIEW? n.d.) Ohjelmoiminen LabVIEW:ssä on nopeaa ja ohjelmointikieli mahdollistaa myös kaikista monimutkaisempien ohjelmien luomisen erittäin yksinkertaisella tavalla (Can LabVIEW C? n.d.). Toisaalta artikkelissa (What is LabVIEW? n.d.) mainitaan, että LabVIEW ohjelmointi vaatii käyttäjältään harjoittelua ja suunnitelmallista oppimisprosessia, jotta maksimoidaan nopeus ja tehokkuus ohjelmien luomisessa. Tiedonkulun ja lohkoakaavioelementtien tunteminen mahdollistaa nopeisiin tuloksiin pääsemiseen jo alussa. (What is LabVIEW? n.d.)

Jokainen ohjelmointisovellus sisältää omia termejä sekä tilan, jossa ohjelma luodaan. Kaikki LabVIEW:n ohjelmallisuus, joka tarvitaan, toteutetaan lohkoakaavion sisällä. Lohkoakaavion toimintaa käsitellään tarkemmin luvussa 4.2.5. LabVIEW:n ohjelmallisuus sisältää monien muiden ohjelmointikielien käyttämiä termejä. Muita ohjelmointikieliä yhdistävät termit, tulo, toiminto ja lähtö ovat osana LabVIEW ohjelmointia, mutta tunnetaan nimillä, ohjain, funktio ja indikaattori. (What is LabVIEW? n.d.)

Ohjaimet eli controllerit ovat ohjelmoinnissa tietoa vastaanottavat elementit. LabVIEW:ssä ohjaimet ovat elementtejä, jotka ovat sijoitettuna etupaneeliin ja sisältävät samalla vastaavan komponentin myös lohkoakaaviossa. Ohjaimen avulla käyttäjä voi syöttää tietoja ohjatakseen koko ohjelmakokonaisuutta sisäisesti yhden VI:n kautta. Ohjaimen ulkoasu on esitetty kuviossa 7. Vasemmalla ja ohjain on mahdollista liittää myös funktioon, jossa tietty toiminto suoritetaan tietyssä ohjaimen antamassa tilassa. Ohjaimen tilan määrittämät ohjaustiedot voidaan käyttää, jos ohjaustietojen tietotyypit ovat muista ohjelmointitietotyypeistä tutut, single, double, string tai muu tunnettu tietotyyppi. (What is LabVIEW? n.d.)



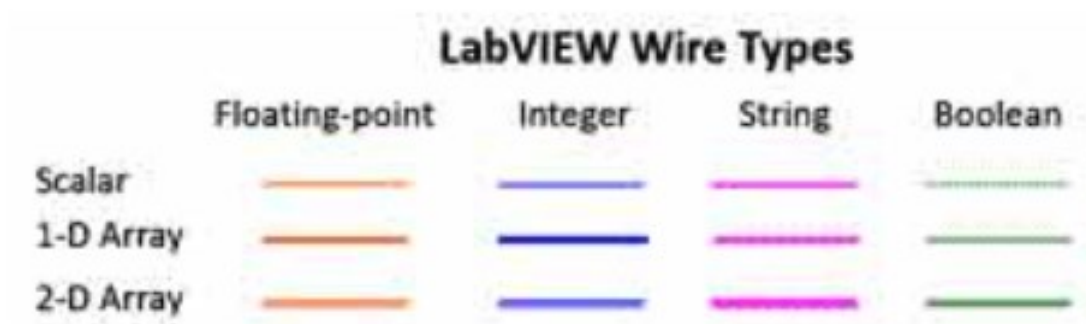
Kuvio 7 Esimerkki ohjelmointielementeistä (What is LabVIEW? n.d.)

Kaikki laskennallinen toiminnallisuus ohjelmoidaan LabVIEW:ssä käyttämällä funktioita. Funktiot ovat käytettävissä LabVIEW VI:n lohkoavaio ikkunasta avautuvasta Functions paletista (LabVIEW Block Diagram Explained 2023). Funktio edellyttää toimiakseen aina sille annettavan syötteen. Saadusta syötteestä funktio suorittaa toiminnon, joka ohjelmassa tarvitaan ja muuttaa sisään saadun syötteen toiminnon jälkeiseen tilaan. Erilaisia funktioita on saatavilla LabVIEW:ssä suuri valikoima, josta esimerkkejä on esitetty kuviossa 7. keskellä. Valikoimassa on saatavilla lisäksi esimerkiksi yksinkertaisia matemaattisia operaattoreita, spektrianalysejä tai kuvankäsittely funktioita. (What is LabVIEW? n.d.)

Useissa sovelluksissa loppukäyttäjää varten on luotava tietoa ilmaisevat näytöt. LabVIEW ohjelmoinnissa käytettävät indikaattorit ovat samankaltaisia kuin ohjaimet. Indikaattorin tehtävänä on ilmaista käyttäjälle ohjelmallisuuden tuloste. Kuviossa 7. oikealla on esitetty indikaattorin ulkoasu, joka vastaa ohjaimen ulkoasua. Indikaattori sisältää visuaalisen vastinkappaleen myös etupaneelissa, joka toimii kuten ohjain, heijastaen muutoksia molempiin suuntiin. (What is LabVIEW? n.d.)

LabVIEW:n ohjelmoinnissa suoritusjärjestys määrätään johdinkytkennöillä. Kuviossa 7. esitettyssä ohjelmointi esimerkissä on kuvattuna, miten ohjaimet, funktiot ja indikaattorit ovat yhdistettyinä

yhteen. Elementtien välinen yhteys on toteutettu käyttämällä tiedonsiirtojohtimia. Johtimien tarkoitus on kuvastaa ohjelman virtausta, jossa tieto liikkuu johdinta pitkin eteenpäin aina seuraavaan elementtiin. LabVIEW johtimien värit ja paksuus ilmaisevat johtimessa kulkevan tietotyypin. Kuviossa 8. on ilmaistuna johdon paksuuteen sidotut tietotyypit ja miten ne määritetään. Johdon paksuus ilmaisee tiedon johtimessa kulkevan tiedon tyypistä, joka on paksuuden mukaan joko skalaari, 1D- tai 2D taulukko. (What is LabVIEW? n.d.)



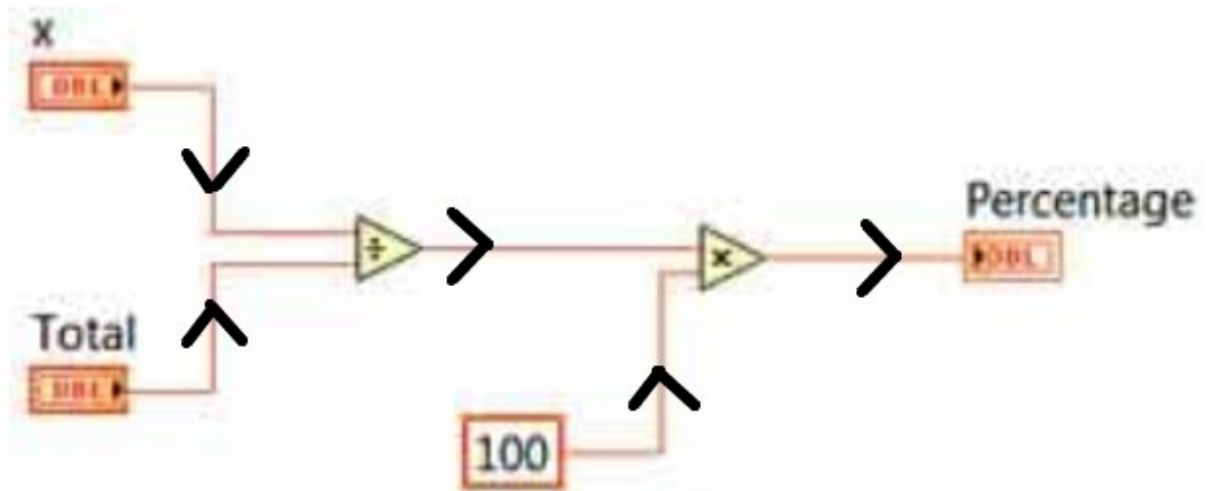
Kuvio 8 LabVIEW johtimien tietotyypit (What is LabVIEW? n.d.)

4.2.7 LabVIEW lohkokaaavion tietovirta

LabVIEW:n graafisen ohjelmakielen suoritusjärjestys asetetaan automaattisesti ohjelman sisällä tietovirran mukaan. LabVIEW-ohjelmakielen suoritusjärjestys poikkeaa yleisistä tekstipohjaisista ohjelmointikielistä, joissa suoritusjärjestys määräytyy rivien mukaan. Automaattinen suoritusjärjestys rakentuu funktion vastaanottamien syöttöjen kautta. Lohkokaaavion funktion suoritus tapahtuu vasta, kun funktio on vastaanottanut kaikki tarvittavat syöttötiedot. Funktion suorituksen jälkeen toimintojen jälkeinen tieto välitetään seuraavaan tietosolmuun, jossa sama suoritusjärjestyksen ehto toteutuu myös. (What is LabVIEW? n.d.)

Tämänkaltaisen tietovirta voi aiheuttaa tekstipohjaiseen ohjelmoimiseen tottuneelta henkilöltä aikaa tottua. Tiedonkulkemisen ymmärtäminen on todettu helpottavan ohjelmointikielen oppimista varsinkin alussa. Tietovirta pyrkii toimimaan vasemmalta oikealle ja tietovirran kulkua on kuvattu esimerkillä (ks. Kuvio 9), jossa suoritetaan kertolaskutoiminto. Suoritusjärjestys on esitetty

kuvion 9. esimerkissä mustilla nuolilla. Esimerkin suoritusjärjestys toteutuu siten, että kertolasku-funktio toteutuu vasta, kun jakolaskulta tuleva syöttö on valmis. (What is LabVIEW n.d.)

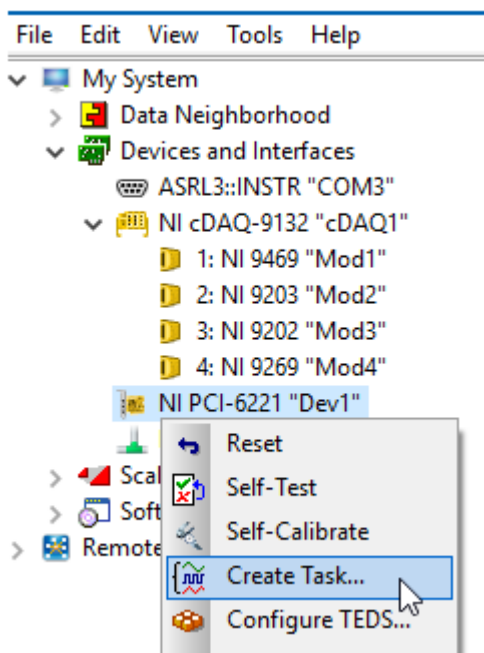


Kuvio 9 Tietovirran havainnollistaminen (What is LabVIEW? n.d. muokattu)

4.3 NI Measurement & Automation explorer (NI MAX)

NI MAX on ilmainen lisäosa, joka asentuu aina National Instruments ajureiden mukana tietokoneelle. NI MAX sovelluksen tehtävänä on määrittää asetukset kaikkiin National Instrumentsin valmistamiin sovelluksiin ja laitteistoihin. Kun määrittäykset on tehty, sovellus auttaa käyttäjää luomaan laitteistoon haluttuja tehtäviä, kanavia, asteikkoja ja rajapintoja. (What is NI Measurement & Automation Explorer (NI MAX) 2022.)

Kun fyysisiä laitteita on kytkettyä järjestelmään, voidaan NI MAX sovelluksessa luoda niille tehtäviä, jos käytössä olevalle laitteellesi on asennettuna N-DAQmx ohjain. Tehtävää luodessa on tarkastettava, että fyysinen laite näkyy NI MAX valikossa, laitteen kuvakkeen väri voi olla joko vihreä, harmaa tai musta. NI MAX laitevalikon kohdalla painettaessa hiiren kakkospainiketta halutun laitteen päällä, avautuu valikko, josta uusi tehtävä voidaan luoda painamalla "Create Task" (ks. Kuvio 10.). (Create Tasks for NI-DAQmx Devices in NI MAX 2023.)



Kuvio 10 NI MAX laitevalikko (Create Tasks for NI-DAQmx Devices in NI MAX 2023)

NI MAX tarjoaa käyttäjälle mahdollisuuden tehtävän luonnin yhteydessä päättää lähdön- tai mitaustyyppin ja ehdottaa laitteeltasi sopivaa porttia tämän toiminnon toteuttamiseen. Sovellustarpeitten mukaan NI MAX tarjoaa toiminnolle tarkentavia asetuksia, kuten skaalauksen, ajoituksen ja synkronointitiedon. (Create Tasks for NI-DAQmx Devices in NI MAX, 2023.)

NI MAX:issa luotua tehtävää voidaan käyttää myös LabVIEW:n ohjelmissa painamalla NI MAX:issa tehtävä aktiiviseksi ja raahaamalla se LabVIEW:n lohkokaavioon. Tämän avulla LabVIEW koodilla voidaan kontrolloida koodin ulkopuolella olevia lähtöjä tai sisäänuloja, josta voi olla suuri hyöty suuressa ja sekalaisessa koodissa. Kun tehtävä on lohkokaaviossa, LabVIEW luo siitä automaattisesti tehtävävakion, joka vastaa NI MAX:issa luotuja tehtäväkonfiguraatioita. Tehtävää voidaan tämän jälkeen käyttää muissakin ohjelmissa, koska asetukset tehtävästä pysyvät LabVIEW tehtävä kirjastossa. Käyttäjä voi lisätä tehtävän useisiin LabVIEW lohkokaavioihin samanaikaisesti. Tehtävää voidaan myös muokata erillisissä Virtual Instrumenteissa, joissa joitain ominaisuuksia tehtävästä voidaan muokata ilman, että itse tehtävää tarvitsisi luoda uudestaan eri ominaisuuksilla. (Create Tasks for NI-DAQmx Devices in LabVIEW 2022.)

4.4 MSP-FET

4.4.1 Laitteen kuvaus ja ominaisuudet

LUX 34:n piirikortti sisältää Texas Instruments valmisteisen MSP430F1101A mikropiirin, jonka ohjelmointi onnistuu MSP-FET laitteella. MSP-FET eli MSP MCU Programmer and Debugger (ks. kuvio 11.) on kompakti ja tehokas emulointi- ja ohjelmistokehitystyökalu, jota kutsutaan useasti myös nimellä virheenpaikannuslaite (debugger probe). MSP-FET tarjoaa mahdollisuuden käyttäjälleen aloittaa nopeasti ohjelmistojen kehittämistyön kaikilla pienitehoisilla MSP mikrokontrollereilla. (MSP-FET n.d.)

MSP-FET tukee reaaliaikaista virheentarkistusta ja ohjelmointia hyödyntämällä mm. JTAG liitäntää (MSP-FET n.d.). JTAG on integroitu testimenetelmä painettujen piirilevyjen liitäntöjen testaamiseen (What is JTAG? n.d.). Laite tarjoaa myös mahdollisuuden käyttää Texas Instrumentsin Backchannel UART yhteyttä MSP UART:n ja tietokoneen USB-liitännän välillä. MSP ohjelmoijalle tämä yhteys antaa hyvän menetelmän sarjaviestintään, joka tapahtuu MSP:n sekä tietokoneessa käytettävän päätteen välillä. Ohjelmien lataaminen kohteeseen tehdään käyttämällä bootstrap loader työkalua hyödyntämällä UART- ja I2C kommunikaatioprotokollia. (MSP-FET n.d.)

MSP-FET yhdistetään tietokoneeseen kaapelilla, USB-liitännän kautta, joka on kuvion 11. yläosassa. Laitteen 14-nastainen liitin osa kuviossa 11. keskellä, tarjoaa pääsyn MSP:n debug-emulointiporttiin. Emulointi portti rakennetaan JTAG-liitännällä tai käytetään vaihtoehtoisesti nastoja säästävää Spy-Bi-Wire (2-johdin JTAG) protokollaa. MSP:n emulointiportin ja 14-nastaisen kaapelin välinen yhteys luodaan yleisesti asettamalla standardoitu 14-nastainen liitin kohdelevylle ja reititetään tarvittavat emulointisignaalit niitä vastaaviin nastoihin. Tällä menetelmällä ohjelmistokehittäjät saavat yksinkertaisen ja järjestelmän sisäisen virheenkorjausmallin. (MSP-FET n.d.)

Ohjelmistokehityksen alkuvaiheessa MSP-FET voidaan liittää yhteen MSP Target Socket korttien kanssa. Liitäntä 14-nastaisen liittimen ja MSP Target Socket korttien välillä tarjoaa pääsyn MSP-laitteen nastoihin, joka helpottaa ohjelmistokehityksen aloittamista. Ohjelmistojen kehittämistyö voidaan aloittaa jo ennen oman kohdelevyn suunnittelua ja rakentamista. (MSP-FET n.d.)



Kuvio 11 MSP-FET ja liitäntäkaapelit (MSP-FET n.d.)

4.5 NI USB-6363

NI USB-6363 on yksi National Instrumentsin valmistamista monitoimisista I/O-laitteista (ks. kuvio 12.), joka tarjoaa mahdollisuudet käyttää digitaalista I/O:ta, analogista I/O:ta, ajastimia ja bittilas-kureita. USB-6363 soveltuu hyvin useisiin eri sovelluksiin ohjaus- ja testausautomaatiosta perustie-tojen ylös kirjaamiseen. (USB-6363 n.d.)

Laite sisältää 4kpl 32 bittistä laskuria ja ajastinta, 4kpl analogista ulostuloa, 48kpl digitaalista ulos- ja sisääntuloa ja 32kpl 16-bittistä analogista sisääntuloa. Ajastimia ja laskureita voidaan käyttää esimerkiksi taajuuksille, instanssien laskennalle tai enkooderille. Laite käyttää ajoituksessa ja synkronoinnissaan sisäänrakennettua National Instrumentsin STC3-tekniikkaa. Tekniikka tarjoaa käytet-täväksi itsenäisiä digitaalisia sekä analogisia ajoitusmoottoreita ja automaattisesti uudelleenkäyn-nistettäviä mittaustehtäviä. Laitteen mukana toimitetaan NI DAQmx ohjain ja konfigurointiin luotu apuohjelma. Apuohjelman ja ohjaimen tarkoitus on tuoda käyttäjälle yksinkertaisuutta laitteen konfigurointiin ja mittauksien toteuttamiseen. (USB-6363 n.d.)



Kuvio 12 Monitoiminen I/O laite (USB-6363 n.d.)

5 Kalibrointi- ja testilaitteen vaatimukset

Suunnittelutyön alussa tuotantotestilaitteeseen määriteltiin tarvittavat vaatimukset. Vaatimusten määrittelystä järjestettiin kokous, jossa keskustelimme yhdessä testilaitteen suunnitteluun osallistuvien kesken. Yrityksellä oli omalla kehitystyölistallaan tarve uudelle ja nopeammalle LUX 34 tuotantotestilaitteelle. Tarve uudelle tuotantotestilaitteelle oli syntynyt LUX 34:n tuotannon valmistusmenetelmän monimutkaisuuden ja ajallisen keston takia.

Nykyinen LUX 34:n valmistusmenetelmä sisältää yksittäisen LUX 34 piirikortin ohjelmoimisen erikseen erillisellä ohjelmointilaitteella. Ohjelmoitu piirikortti sijoitetaan sen jälkeen mekaaniseen testilaitteeseen, jossa LUX 34:n piirikortille kalibroidaan oikeat valoisuus- ja lämpötila-arvot. Tämän jälkeen LUX 34 testataan samalla testilaitteella, jotta sen kalibrointi on onnistunut ja LUX 34 toimii luvutulla tavalla. Tuotantotyöntekijä suorittaa tämän jälkeen LUX 34:lle laitteen loppukokoonpanon.

Tuotantotestilaitteen kehitystyön alussa tuotantotestilaitteen vaatimukset olivat automaattisuus ja tuotannon tehokkuuden lisääminen. Työn alussa tarkennettiin myös automaattisen tuotantotestilaitteessa vaadittavat ja LUX 34 tuotteeseen tarvittavat työtehtävät. Määrittelyn lopputulos oli

testisykliltään mahdollisimman nopea, helppokäyttöinen ja siirreltävä tuotantotestilaitte, jossa automaattisuus muodostuu LabVIEW-ohjelmointikielellä rakennetusta sekvenssistä. Tuotantotestilaitte tuli rakentaa NG tester alustalle, jota on hyödynnetty yrityksen ensimmäisissä automaattisissa tuotantotestilaitteissa. Tuotantotestilaitteen toiminta suunniteltiin jo olemassa olevalle LUX 34 tuotteen piirikortille toimivaksi. Suunnittelussa laadittiin myös aikataulus projektille, jotta tuotantotestilaitteen valmistuminen olisi johdonmukaisempaa ja helpommin seurattavissa.

5.1.1 Testilaitteen käytettävyys

Tuotantotestilaitteen vaatimuksena oli olla käytettävyydeltään yksinkertainen ja nopea. Yksinkertainen käytettävyys on tärkeää tuotantotilassa, jotta tuotannon työntekijöiden on helppo oppia uuden testilaitteen toiminta. Käytettävyyttä pyrittiin nostamaan rakentamalla tuotekohtainen NG tester fixtuuri nimenomaan Pro dual Oy:n omalle NG tester pohjalle (ks. Liite 2.). NG tester pohjalle rakennettu testifixeureita on tuotannossa käytetty jo yli kolmen vuoden ajan ja moni tuotantotyöntekijä on tullut tutuksi niiden toimintaan. NG tester tuo universaalina tuotantotestilaitteen rakenteena ominaisuuden, että kerran käytön opeteltuaan, tuotantotyöntekijä osaa käyttää tuotantotestilaitetta muidenkin tuotteiden kohdalla. NG testeristä on kerrottu lisää luvussa 6.1.

Nopean testiprosessin luominen oli vaatimuksista suurin. Tuotantotestilaitteen hyödyntämä LabVIEW VI:stä koostuva sekvenssi täytyi suunnitella ja rakentaa kokonaisuudessaan LUX 34 tuotteelle sopivaksi. Sekvenssin oli oltava suoritusnopeudeltaan mahdollisimman tuotannollisesti tehokas. Myös tuotteen ohjelmoinnin suunnittelu oli tärkeää. Ohjelmoinnin täytyi onnistua automaattisesti, jotta tuotantotyöntekijä voi tehdä muuta kokoonpanoa koko testiprosessin ajan.

5.1.2 Testilaitteen toiminnan luotettavuus

Testilaitteen tekemän ohjelmoinnin, kalibroinnin ja lopputestauksen tuli olla luotettavaa. Lopputuloksen luotettavuutta oli testattava automaattisen testilaitteen hyväksymille tuotteille fyysisillä mittauksilla, kun ensimmäiset tuotannossa tehdyt tuotteet olivat valmiita. Testauslaitte oli suunniteltava kestävä pitkäaikaisesti ja olemaan erittäin vikasietoinen. Vikasietoisuutta parannettiin työn edetessä varmistamalla jatkuvasti LabVIEW VI:den sekä sekvenssin toimivuus.

Testilaitteen valmistuttua sillä suoritettiin tuotannossa ensimmäisiä aikavertailuja, joissa verrattiin testilaitteiden suoritusnopeuksia uuden ja vanhan testilaitteen välillä. Ensimmäisten suurien erien valmistuksen yhteydessä tulee tulevaisuudessa selvittää myös mahdolliset ohjelmalliset viat, joita testiprosessissa alkuun voi esiintyä.

5.1.3 Testilaitteen LabVIEW ohjelma

Testilaitteen automaatiikka rakennettiin LabVIEW-ohjelmointikielellä. LabVIEW-ohjelmointikielellä luodaan useita eri toimintoja toteuttavia VI ohjelmatiedostoja. Ohjelmien sekvenssillinen suorittaminen toteutettiin käyttämällä Notepad++ tekstinkäsittelyohjelmaa. LabVIEW-ohjelmointikielellä luodut VI:t suoritettiin peräkkäisenä sarjana eli sekvenssinä tekstitiedoston mukaisesti.

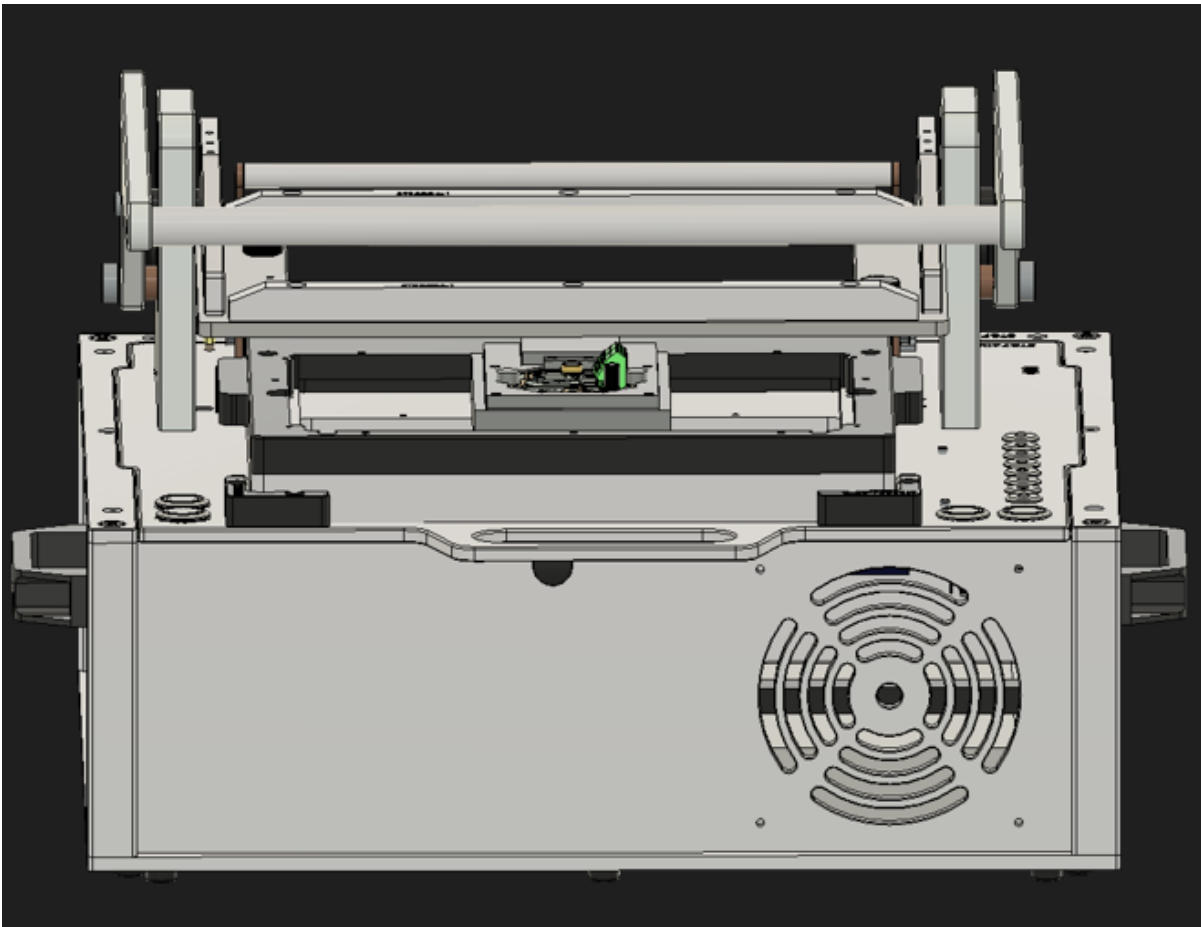
Geneerisiä LabVIEW-ohjelmointikielellä luotuja ohjelmia löytyi valmiina yrityksen verkkopalvelimilta ensimmäisistä automaatti testilaitteprojekteista ja näitä hyödynnettiin myös tässä projektissa. Edellisistä LabVIEW-ohjelmointikielellä luoduista ohjelmista oli tarkoitus muokata tarvittavat parametrit ja toiminnot, jotta ohjelmat saataisiin toimimaan myös LUX 34 tuotantotestilaitteessa. Uusia LabVIEW-ohjelmointikielellä luotuja VI:tä oli tarkoitus luoda paljon ja näiden ohjelmien tuli olla tarkennettuna pelkästään LUX 34 tuotantotestilaitteessa käytettyihin toimintoihin. Uusien LabVIEW:illä luotujen ohjelmien tuli pitää sisällään mikropiirin ohjelmointi, ulostulojen kalibrointi, kalibroinnin tarkastus ja hyväksytyin tuotteen testitarran tulostaminen.

Kokouksessa todettiin myös jälkeenpäin, että yhden LabVIEW VI:n toiminnallisuus täytyi olla lämpötilan lukeminen testilaitteessa käytetyltä lämpötilareferenssiltä. Lämpötilan lukeminen oli suoritettava käyttämällä referenssilämpötilalähteen käyttämää Modbus-sarjaliikenneprotokollaa. Tuotteen ja lämpötilareferenssin ilmoittamien jänniteviestien erotusta oli myös tarkoitus hyödyntää nopeuttamaan tuotteen kalibrointia.

6 Testilaitteen toteutus

6.1 NG tester

NG tester on universaali nimitys kaikille Pro dual Oy:n tuotannon uusille automaattituotantotestilaitteille. NG tester koostuu testilaitteen rungosta, sen sisältämästä laiterakenteesta ja LabVIEW pohjaisesta ohjelmallisuudesta. NG testeriin kiinnitetään myös tuotekohtainen fixtuuri, joka on kyseessä olevan laitteen piirikortin muotoinen ja sisältää metallikontaktit tarvittaville laitteen koskettimille. NG tester hyväksyy testattavan laitteen piirikortin toiminnan. NG tester hyödyntää toiminnassaan kytkentälevyä, jossa sijaitsee suorat AI/AO, DI/DO kytkennät testilaitteen I/O laitteille. NG tester sisältää kahvasta laskettavan kannen, joka painaa tuotteen piirikortin kiinni fixtuuriin. Valmiista LUX 34 tuotantotestilaitteesta oli työn alussa suunniteltu esimerkkikuva, jonka oli laatinut Pro dual Oy:n tuotekehityksen insinööri (ks Kuvio 13). NG tester testiaseman kanteen täytyi suunnitella myös valonlähde kalibrointeja varten.



Kuvio 13 Suunnittelukuva NG testerin rakenteesta LUX 34 fixtuurilla.

NG tester sijoitetaan liikutettavalle alustalle, jotta sen kuljettaminen tuotannon eri alueille on mahdollisimman helppoa suuren koon takia. NG testerin lisäksi kuljetettavalle alustalle kuuluu ohjelmallisuuden sisältävä tietokone, tulostin ja mahdollisesti myös tuotteen valmistamiseen vaadittavia komponentteja. Tuotantotestilaitteen valmistaminen aloitettiin testilaitteen NG testerin mekaanisesta kasaamisesta. Melkein kaikki laitteen kasaamiseen vaadittavat komponentit löytyivät toimeksiantajan varastosta.

Produalin universaali NG tester rakenne kalustetaan aina J-Link ohjelmointilaitteilla, joita tässä projektissa ei voitu käyttää tuotteen ohjelmoimiseen, mutta J-Link ohjelmointilaitteet kalustettiin tulevaisuutta varten myös tähän tuotantotestilaitteeseen. NG testerin universaali rakenne sisältää myös USB-6501 digitaalisen I/O laitteen ja USB-6363 monitoimi- I/O laitteen. Laitteiden avulla saadaan toteutettua kaikkien toimintojen tarvittavat analogiset- ja digitaaliset I/O säädöt sekä mitaukset. NG tester sisältää myös kolme tiedonsiirtoon tarkoitettua USB-hubia, joihin kiinnitetään tarvittavat RS485 ja RS232 kytkennät mm. kytkentälevyltä ja testilaitteen takaseinän liitimiltä.

Tuotantotestilaitteen runkoon kiinnittyy myös merkkivalot, joilla voidaan ilmoittaa käyttäjälle tietoja testiprosessin etenemisestä. Valojen lisäksi laitteeseen kuuluu myös valaistuja painikkeita. Painikkeiden toimintaa voidaan tarvita myöhemmin tulevaisuudessa, jos testilaitetta käytetään toisen tuoteperheen testauksessa. LUX 34 tuotantotestilaitteessa käyttäjän ei ole tarkoitus hyväksyä tai hylätä testiprosessia painikkeilla, mutta tarvittaessa painikkeisiin saadaan asetettua myöhemmin toimintoja, jos testiprosessi sen vaatisikin.

Tuotantotestilaitteen kahvan laskeminen on tuotannollinen tapa aloittaa testiprosessi. Kahvan liikettä seuraa läheisyyskytkin, jonka antamaa binääristä signaalia hyödynnetään ohjelmallisuudessa laukaisimena. Tuotantotestilaitteen fixtuuriin rajapintaan tulee integroituna Vaisalan HMP110 lämpötila- ja kosteusanturi, jonka antamaa digitaalista jänniteviestiä käytetään tuotteen kalibroinnissa referenssiarvona. Lisäksi NG testerin fixtuurin sisälle on sijoitettava myös LUX 34:n piirikortille sopiva ohjelmointilaite. Alustalla oleva tulostin kytketään myös tuotantotestilaitteeseen. Tulostimen tarkoitus on tulostaa onnistuneen testauksen jälkeen tuotekohtainen testi- ja kytkentätarra. NG tester tuotantotestilaitteen universaalisesta rakenteesta on luotu laitteen pääkomponentteja selventävä kuva, joka osoittaa pääkomponenttien nimet ja paikat testilaitteessa (ks. Liite 2).

6.2 Testisekvenssi

Testisekvenssi sisältää kokonaisuudessaan automaattitestilaitteen testausoperaation, joka koostuu kaikista testiprosessiin vaadittavista LabVIEW VI:stä. Sekvenssitiedoston tarkoitus on suorittaa kaikki testiin vaadittavat LabVIEW VI:t yhtenä pitkänä sarjana. LabVIEW VI:tä kutsutaan tiedoston kautta käynnistymään ja yksittäisen LabVIEW VI:n suorituksen jälkeen kutsutaan seuraava haluttu VI käynnistymään.

Testisekvenssi on tehty Notepad ++ tekstinkäsittelyohjelman avulla, jossa sekvenssitiedoston tyyppinä käytetään ini tiedostomuotoa. Sekvenssitiedosto luotiin heti projektin alussa, jotta sitä on helppo rakentaa ja kehittää LabVIEW VI:den valmistuessa. Sekvenssi on jaettu testiprosessin eri osa-alueisiin, jotta jokaisen kalibrointi- ja testitoiminnon jälkeen voidaan koestaa siihen asti rakentuneen sekvenssin toiminnallisuus. Sekvenssitiedoston esimerkkirakenne ja osa-alueisiin jakaminen on esitetty kuviossa 14.

```
# Esimerkki LUX34 testin sekvenssistä                               Parametrit

# Luodaan tarvittavat National Instrumentsin taskit
1. Testiaskellus=NI-analog-output-task,                            #A01 Valoisuuslähde;
2. Testiaskellus=NI-analog-input-task,                            #AI1 Valoisuudenmäärä;
3. Testiaskellus=NI-analog-input-task,                            #AI2 Lämpötila-arvo;
4. Testiaskellus=NI-digital-output-task,                          #DI1 Span-control;
5. Testiaskellus=NI-digital-output-task,                          #DI2 Zero-control;
6. Testiaskellus=NI-digital-output-task,                          #DI3 Testilaitteen-käyttäjännite;

# Tuotteen ohjelmoiminen
7. Testiaskellus=MSP-prosessorin-erasointi,                       #Syötetään kaikki vi:n tarvitsemat parametrit;
8. Testiaskellus=MSP-prosessorin-ohjelmoiminen,                   #Syötetään kaikki vi:n tarvitsemat parametrit;

# Kalibrointitilaan siirtyminen
9. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Zero-control
10. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 1;#Testilaitteen-käyttäjännite
11. Testiaskellus=Wait,                                           #Viive 1000ms
12. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 0;#Portti Zero-control

# Lämpötilan Kalibrointi
13. Testiaskellus=LUX34-lämpötila-kalibrointi,                    #Vi:n parametrejä, esim. LSL -0.1;
14. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 1;#Portti Zero-control
15. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 0;#Portti Zero-control

# Valoisuuden Kalibrointi
16. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 1;#Portti Span-control
17. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvo 8;LSL 8
18. Testiaskellus=Digitaalinen-output-tilanohjaus,                #Tila 1;#Portti Zero-control
```

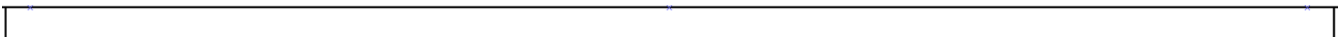
Kuvio 14 LUX 34 testisekvenssin esimerkkirakenne

Testisekvenssin suoritusjärjestys määräytyy testiaskelluksien järjestysnumeroiden mukaan, jotka ovat listattuna kuvion 14. vasemmassa sivussa. Vihreällä tekstillä kuviossa 14. on kuvattuna yksi testiprosessin osa-alue, jonka alapuolella on kaikki sen testiprosessin vaatimat LabVIEW-ohjelmat. Kun järjestyksessä seuraava LabVIEW-ohjelma aktivoituu, syötetään sen parametreihin sekvenssiedoston parametreissa olevat tiedot, kuten tilatiedot ja raja-arvot, joita ohjelma käyttää funktioissaan.

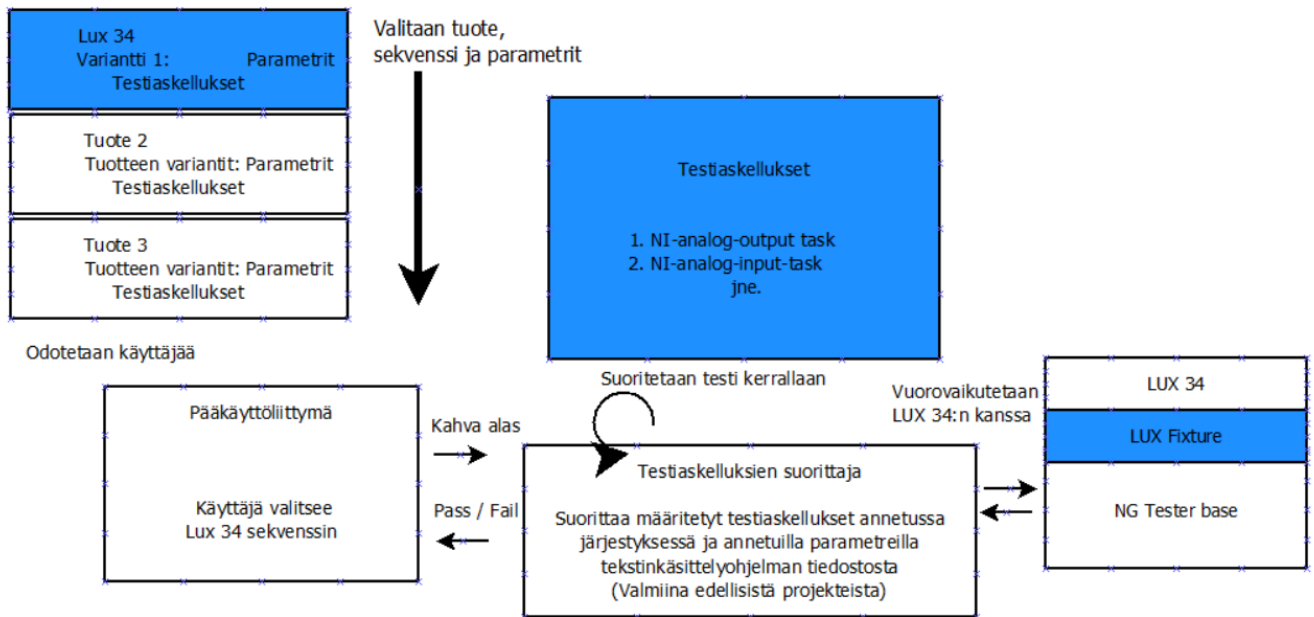
Sekvenssiedosto syötetään testilaitteen käyttämään pääkäyttöliittymä VI:hin. Pääkäyttöliittymä on olemassa toimeksiantajalla edellisestä projektista. Pääkäyttöliittymä näkyy tuotannossa testilaitteen käyttäjälle ja sisältää testin olennaiset etenemistä edustavat asiat, kuten suoritettujen testiaskellusten määrän graafisena kuviona. Pääkäyttöliittymä ilmoittaa myös käyttäjälle mahdollisen virheen testissä ja kertoo sekvenssiedoston testiaskelleen, jossa virhe on tullut. Kuviossa 15. on esitetty pääkäyttöliittymää havainnoiva esimerkki, jossa esiintyy kaikki oikean pääkäyttöliittymän oleellimmat objektit ja toiminnot. Kuviossa 15. näkyy testiprosessin edistyminen oranssilla edistymisindikaattorilla ja sekvenssin testiaskellukset ilmoitetaan vasemmalla ylhäällä olevassa ikkunassa, joka kertoo käyttäjälle tarvittavat tiedot. Kuviossa 16. kuvataan järjestelmätasolla koko tuotantotestilaitteen tuomaa valmistusprosessia, jossa sekvenssi ja pääkäyttöliittymä on mukana ja kuvioon on sinisellä merkattu osa-alueet, jotka olivat työn aikana luotava.

NG-TESTER							SULJE
Ladataan sekvenssiedostoa, odota hetki		Tässä valikossa valitaan haluttu sekvenssiedosto				▼	
		Tuotteen tuotekoodi ja tuotteen nimi				▼	
					Suorituksen kesto 00:00:00		● START
							■ STOP

Tässä esitetään sekvenssiedoston eteneminen ja sekvenssin suorituksessa mitatut arvot mm. Lämpötila-arvot. Saadaan myös tieto mahdollisesta testin hylkäysperusteesta.



Kuvio 15 Havainnoiva esimerkki NG tester pääkäyttöliittymästä



Kuvio 16 Järjestelmätason havainnointi sekvenssin ja pääkäyttöliittymän toiminnasta valmistusprosessissa

6.3 LUX 34 piirikortin ohjelmoiminen

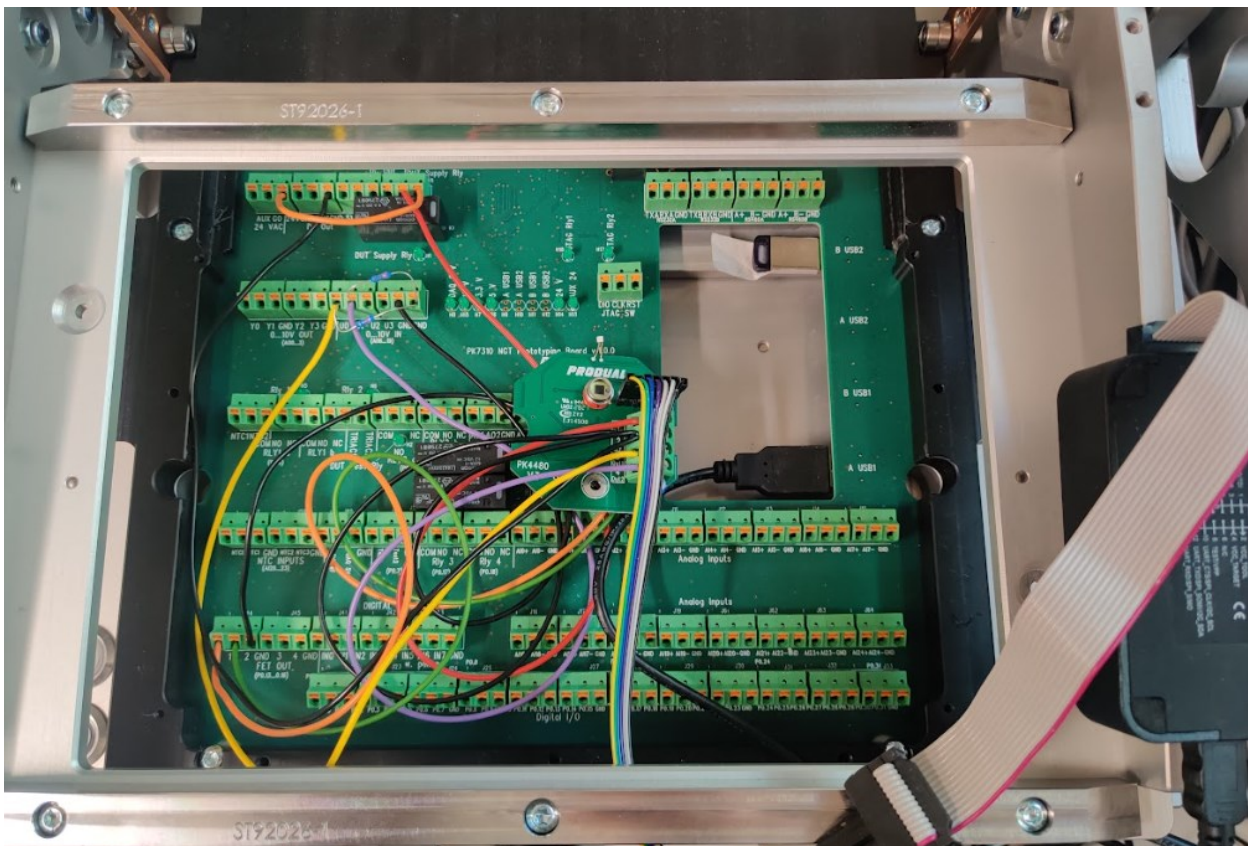
LUX 34 piirikortti sisältää Texas Instruments valmistaisen MSP430F1101A mikropiirin. Mikropiirin ohjelmoimiseen ei voitu käyttää universaalien NG tester rakenteen sisältämiä J-Link laitteita, joten oli tarve löytää tilalle tuotteen mikropiirille sopiva laite. Texas Instruments ilmoittaa verkkosivullaan MSP-FET ohjelmointityökalun toimivan kyseisen prosessorin ohjelmoimisessa (MSP-FET n.d.). Kun sopiva ohjelmointityökalu oli saatu, täytyi sen käyttämistä varten ladata MSP430Flasher tietokone sovellus. MSP430Flasher on avoimen lähdekoodin komentorivi käyttöliittymä, joka tarjoaa yleisimpiä ohjelmointikomentoja ja sitä voidaan käyttää MSP430 tyyppisten prosessoreiden ohjelmoimiseen (MSP430-Flasher n.d.).

Ohjelmointi oli suunniteltu toteutettavaksi automaattisesti aina ennen kalibrointia, joten tästä syystä ohjelmointilaitte tulee lopuksi kytkeä NG testerin fixtuuriin. Ennen lopullista kytkemistä täytyi kuitenkin ohjelmoinnin onnistuminen koestaa uudella ohjelmointilaitteella ja MSP430Flasher komentorivillä. LUX 34 piirikortti sisältää 7-napaisen ohjelmointiriman, josta levyä on ennen ohjelmoitu vanhalla ohjelmointilaitteella. LUX 34 piirikortti johdotettiin uuteen MSP-FET laitteeseen aluksi työpöydällä hyödyntämällä kuviossa 17. esitettyä MSP-FET laitteen kääntöpuolella olevaa

kytkentäkuvaa ja mukana tullutta 14-nastaista kaapelia. Piirikortin kytkentä tuotantotestilaitteen prototyyppi fixtuuriin on esitetty kuviossa 18.



Kuvio 17 MSP-FET kytkentäohje laitteen kääntöpuolella (MSP-FET n.d.)



Kuvio 18 LUX 34 piirikortti kytkettynä prototyyppi fixtuuriin

MSP-FET syöttää LUX 34 piirikortin mikroprosessorille käyttöjännitteen (VCC) ohjelmoinnin ajaksi kuviossa 18. näkyvän keltaisen liittimen kautta, joka mahdollistaa yhteyden MSP430Flasher komentorivin ja tuotteen mikropiirin välille. Ohjelmoinnissa oli myös tarkoitus saada aluksi pyyhittyä kaikki tieto mikä prosessorissa saattaisi jo olla.

MSP430Flasher komentoriville kirjoitettiin sovelluksen ohjeiden mukainen komento, jossa sovellukselle annetaan tieto ohjelmasta ja erilaisista argumenteista, joita voidaan käyttää pelkän ohjelmoinnin lisäksi. Ohjelmointiin käytettiin LUX 34 virallista ohjelmaa, joka oli ladattu tietokoneelle edellisestä ohjelmointilaitteesta. Sovelluksen ohjeiden avulla lisättiin myös onnistuneesti prosessorin tyhjennys käyttämällä komentorivissä argumenttia -e (erase).

6.4 LabVIEW ohjelmakokonaisuus

Suunnittelu vaiheessa päätettiin, että tuotantotestilaitteeseen käytettävä automaattisuus oli luotava LabVIEW-ohjelmointikielellä. LabVIEW VI:t täytyi suunnitella ja luoda tuotteen tarpeisiin. Aluksi listattiin vaadittavat VI:t, jotka täytyi luoda. Toimeksiantajan edellisistä tuotantotestilaitteista oli käytettävissä generisiä LabVIEW VI:tä, joita pystyttiin myös käyttämään tässä projektissa. Valmiina olevat VI:t sisälsivät toimintoinaan mm. virransyötön, modbus yhteyden, NG tester pääkäyttöliittymän, analogisen mittauksen ja taskin luomisen.

LUX 34 tuotteelle täytyi luoda täysin uusia VI:tä, jotka sisältävät toiminnoissaan ohjelmoinnin ja sen tarvittavat argumentit, tuotteen lämpö- ja valoisuusarvon kalibroinnin, valoisuuden ohjauksen analogisella jänniteviestillä kalibrointia varten ja tuotteen loppumittaukset. LabVIEW VI:t tuli rakentaa yksi kerrallaan testiprosessin määräämässä järjestyksessä, jotta saadaan todettua järjestelmällisesti missä vaiheessa projekti on. Jokainen LabVIEW VI ladattiin valmistumisen jälkeen Pro-dual Oy:n versionhallintaan, jossa se voidaan tarkistuttaa toimeksiantajan ohjelmointi-insinöörin toimesta.

6.5 Ohjelmallisuuden vaatimukset

Toimeksiannossa oli tarkennettu, että LUX 34:n lähdekoodiin ei saa tehdä muutoksia. LUX 34 on vanhaan tuoteperheeseen kuuluva tuote, jonka tuotantotestauksen käyttötapaukset oli alkuperäisesti suunniteltu toimimaan koneen ja ihmisen vuorovaikutuksen kautta. Tämän takia ainoa vaihtoehto kalibroinnin toteuttamiselle oli matkia vanhan kalibroitavan ohjeistusta.

Edellisen manuaalisen tuotantotestilaitteen testausohjeista selviää, että LUX 34:n lämpötila on tarkoitettu kalibroitavaksi kaksi celsius astetta vallitsevaa lämpötilaa alemmaksi, koska sillä saadaan kompensoitua piirikortin itselämpenemisestä aiheutuva vaikutus pois. LUX 34:n valoisuus oli kalibroitava myös matkimalla manuaalisen testausohjeen ilmaisemia keinoja.

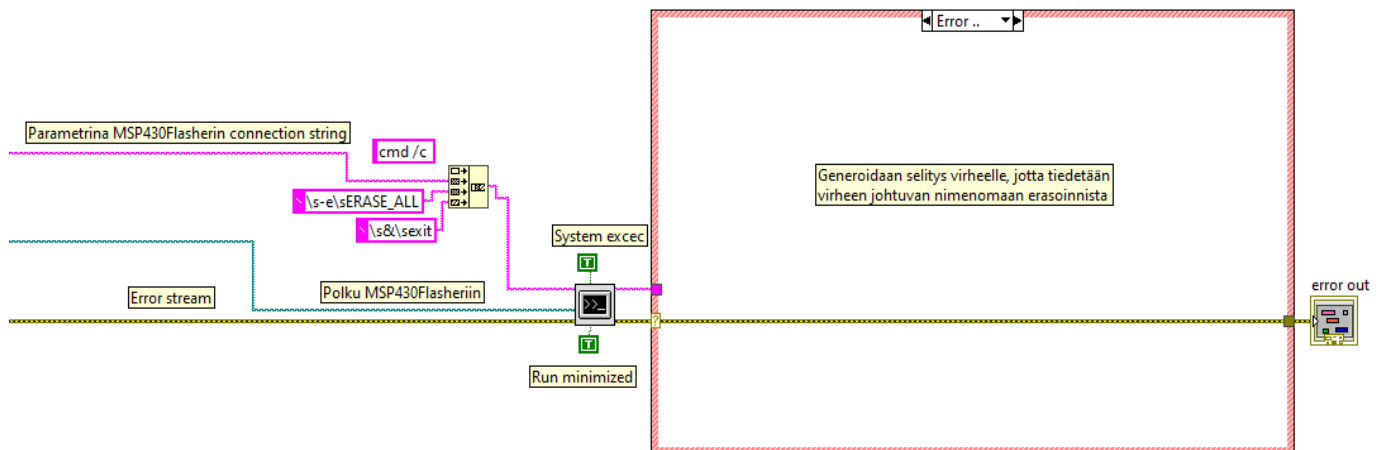
Laite ei sisällä mitään rakennusautomaatiolaitteille tyypillisiä sarjaliikenneprotokollia, kuten Modbus tai Bacnet sarjaliikenneprotokollia. Laitteen kanssa kommunikointiin ei pystytty tämän takia hyödyntämään muuta kuin jännitesignaaleita tuotantotestilaitteen kautta.

6.5.1 LUX 34 mikroprosessorin erasointi ja ohjelmointi

Ensimmäisenä luotavana VI:nä oli MSP-FET ohjelmointilaitetta hyödyntävä VI, jossa komentorivi työkalu MSP430Flasher aukeaa ja saadaan erasoitua eli tyhjennettyä kaikki prosessorin sisältämä tieto. LabVIEW VI:n luomisessa on tärkeää muistaa tietovirtaus, joten VI:n luominen aloitettiin vasemmalta oikealle ja ylhäältä alas periaatteella. Työn kaikkien VI:den sisäistä suoritusjärjestystä on varmistettu myös käyttämällä error-linjaa lohkojen välillä, joka varmistaa oikeanlaisen tietovuon suoritusjärjestyksen. Ehtolohko rakennetta käytettiin rakentamaan lohkoavioihin myös poikkeuskäsittelyä. Ehtolohkoissa on sisällettynä aina error-lohko, joka ilmoittaa käyttäjälle ongelmasta, jos VI:n parametreissa on virheellistä tietoa.

Ensimmäisenä lohkoavioon lisättiin parametrin, joihin tiedot voidaan myöhemmin syöttää sekvenssitiedostosta. Kaikissa LabVIEW lohkoavioissa on tärkeää muistaa sijoittaa oikeassa järjestyksessä tarvittavat toiminnot, joten ensimmäisenä suoritettava asia oli funktio, jossa MSP430Flasher sovellus tehtiin aukeamaan tietokoneen komentokehotteen kautta. MSP430Flasher aukeaa, kun tietokoneen komentokehoteeseen syötetään komento, jossa ilmoitetaan tarkasti sovelluksen sijainti tietokoneen resurssinhallinnassa.

VI:ssä toiminto luotiin käyttämällä komentokehotteen avaavaa toimintolohkoa, ja syöttämällä sen ”minimized” parametriin arvo true, jolloin komentokehote aukeaa taustalla häiritsemättä käyttäjää. Komentokehotteen ollessa auki sinne syötetään LabVIEW:n tekstiä yhdistävällä loholla halutut komennot järjestyksessä, MSP430Flasher sijainti parametrina ja sitten erasointiin vaaditut argumentit (ks. Kuvio 19).



Kuvio 19 MSP430Flasherin käyttäminen erasointiin LabVIEW:ssä

Kun erasointi VI oli valmis, saatiin siitä kopioitua LabVIEW:n projekti kansioon identtinen ohjelma, jota lähdettiin muokkaamaan. Muokkauksen tarkoitus oli saada edellisen VI:n rakenteella toteutettua myös tuotteen ohjelmointi. Erasointi VI:n rakenne vastasi todella hyvin myös ohjelmoinnin tarpeisiin, koska ohjelmoinnissa on myös käytössä MSP430Flasher komentorivi työkalu. Erasointi VI:n rakenteeseen ei tarvinnut tehdä mitään muutoksia vaan pelkästään argumenttien vaihdolla saatiin erasointi VI:n kopiosta toimiva LUX 34 ohjelmointi VI.

6.5.2 LUX 34 lämpötilan kalibrointi

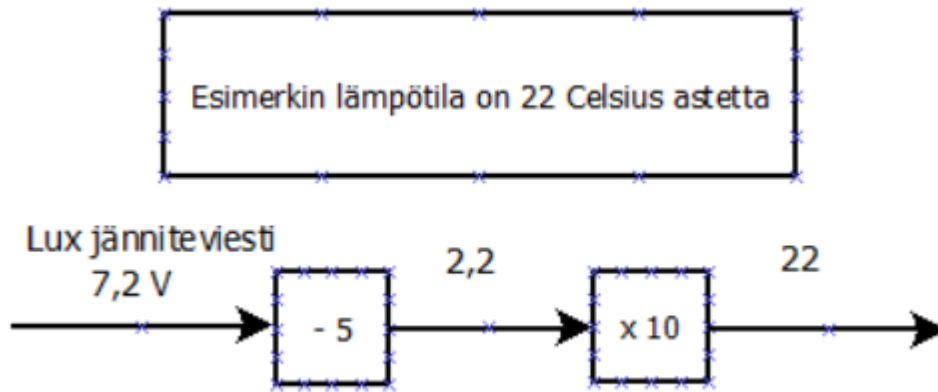
LUX 34 tuotteen lämpötilan kalibrointi suoritetaan, kun ohjelma on ladattu tuotteeseen onnistuneesti ja tuotteeseen kytketään käyttäjännite. Laitteen kalibrointiin on olemassa työohjeet tuotannon manuaaliselle tuotantotestilaitteelle. Työohjeista selviää, että tuotteen käyttäjännitteen kytkeytyessä päälle, täytyy tuotantotestilaitteessa oleva zero-painike olla painettuna samanaikaisesti.

Tuotteen työohjetta käytettiin apuna myös uuden tuotantotestilaitteen automaattisuuden luomiseen ohjelmallisesti. Tuotteen ohjelma sisältää tilan, jossa tuotteen kalibrointi pitää suorittaa. Tilaan pääsee, kun zero-painike on aktiivisena yli sekunnin ajan käyttöjännitteen kytkemisestä, jonka jälkeen zero-painike asetetaan pois päältä. Uudessa tuotantotestilaitteessa on tehty kytkennät vastaavasti zero- ja span-painikkeille, joiden tilaa voidaan kontrolloida ohjelmallisesti.

Ohjelmassa referenssilämpötilaa täytyy lukea Vaisalan HMP110 lämpö- ja kosteuslähettimeltä ja saman aikaisesti säätää tuotteen osoittama lämpötila kahden asteen verran referenssin alapuolelle. Lämpötilan säätäminen tapahtuu ohjelmallisesti käyttämällä tuotteelle tulevia Zero ja Span signaaleita. Kun lämpötilaa tahdotaan säätää ylöspäin, täytyy tuotteelle tuleva span-tulo olla aktiivisena, jonka jälkeen lyhyellä zero-tulon aktivoimisella lämpötila säätyy ylöspäin. Alaspäin säädettäessä aktivoidaan pelkästään lyhyesti zero-tulo.

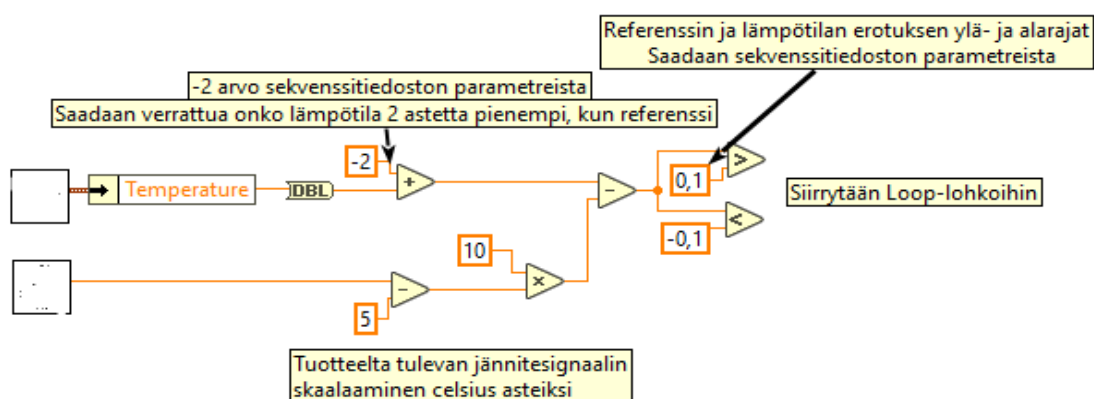
LabVIEW VI on luotu olemaan mahdollisimman nopea kalibroinnissaan. LabVIEW VI:n luominen alkoi rakentamalla lohkoakaavioon tarvittavat parametrit. Parametrien tarkoituksena on toimia lämpötila säädön käyttäminä muuttujina, tarjoten ohjelmalle mm. maksimi yritysten määrän ja hyväksytyyn säätöön riittävän raja-arvon. Parametrien jälkeen lohkoakaavioon lisättiin Loop-lohko, johon laskentaoperaatiot sijoitetaan. Loop-lohko toistaa sen sisällä olevia funktioita niin kauan kunnes sille asetettu hyväksymiskriteeri täyttyy.

Lämpötilan säätö toimii hyödyntämällä jänniteviestiä tuotteelta ja lämpötila-arvoa referenssiltä. Referenssin lämpötilatieto saadaan reaali muodossa modbusrekisteristä suoraan lohkoakaavioon, mutta tuotteen ilmoittama jänniteviesti oli muokattava esittämään reaalista lämpötilaa, jotta sitä voidaan verrata referenssi arvoon. LUX 34 tuotteen jänniteviesti sisältää 100 celsius asteen skaalan, joka ilmoitetaan ulostuloon 0-10V jänniteviestinä. LUX 34 lämpötilaskaala alkaa -50 celsiuksesta ja päättyy +50 celsiukseen, joten tämä täytyi huomioida reaaliarvoon skaalauksessa. LUX 34 tuotteen jänniteviesti muutettiin kuvion 20. mukaisilla laskuoperaatioilla vastaamaan reaalista lämpötilaa lohkoakaaviossa.

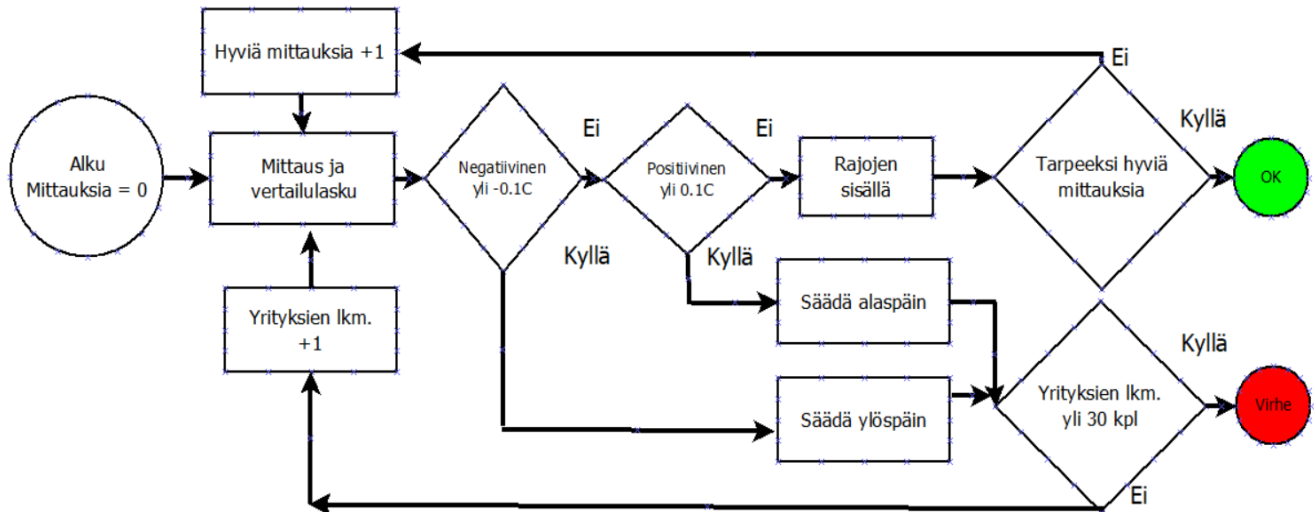


Kuvio 20 LUX 34 lämpötila jänniteviestin skaalaaminen celsius asteiksi

Kun skaalausoperaatio on suoritettuna Loop-lohkon sisällä, piti lämpötila-arvoille tehdä vertailu. Vertailun jälkeen oli luotava vielä kaksi Loop-lohkoa lisää, joissa kalibrointi suoritetaan. Lämpötilaerotuksen arvo määrää kumpaan Loop-lohkoon siirrytään erotuksen jälkeen. Jos tuotteen lämpötila-arvo on referenssiarvoa pienempi, siirrytään Loop-lohkoon, jossa toteutetaan yksi lämpötilan askellus ylöspäin. Jos lämpötila-arvo on suurempi, kun referenssin ilmoittama arvo, siirrytään vastaavasti Loop-lohkoon, jossa lämpötilaa askelletaan yksi kerta alaspäin. Vertailu ja seuraavien Loop-lohkojen valinta toteutuu lohkoaviossa kuvion 21. mukaisella logiikalla. Referenssin lämpötila luetaan kuvion 21. ylemmässä datavirrassa ja siitä poistetaan kaksi celsius astetta, jotta tuotteen lämpötila tulee kalibroiduksi kaksi astetta referenssiä pienemmäksi. Kuviossa 22. esitetään havainnoiva vertailun jälkeinen Loop-lohkojen suoritus ja kalibrointi VI:n toimintamalli.



Kuvio 21 Lämpötilojen vertailulaskenta esimerkki LabVIEW:ssä



Kuvio 22 Loop-lohkojen looginen suoritus kalibrointi VI:n sisällä

Kun kalibrointi on onnistunut saavuttamaan yhden desimaalin erotuksen, siirrytään valinnassa Loop-lohkoon, jossa sijaitsee laskuri. Laskuri laskee monta kertaa tuote saavuttaa raja-arvoissa olevan hyväksytyin kalibroinnin. Laskurin lukua verrataan samanaikaisesti sekvenssiedostosta syötettyyn arvoon, kun laskuri saavuttaa syötetyn parametrin arvon, siirrytään lämpötilan kalibroinnista seuraavaan kalibrointivaiheeseen.

6.5.3 LUX 34 valoisuuden kalibrointi

Valoisuuden kalibrointi tulee suorittaa LUX 34:lle lämpötilan kalibroinnin jälkeen seuraavana. Manuaalisen tuotantotestilaitteen työohjeessa on mainittuna kaksi pisteinen kalibrointi. Siinä tuotantotestilaitteen valolähde asetetaan antamaan 1000 luxia tuotteelle, jonka jälkeen kytketään piirikortin piikkirimaan S1 oikosulkupala, jolloin LUX 34:n skaalaus muuttuu 0–1000 lux asteikolle. Kun S1 oikosulkupala on kytkettynä tuotteeseen, alkaa tuotantotestilaitteessa valoisuusarvoa ilmoittavan näytön arvo suurentumaan. Arvon on tarkoitus nousta, kunnes saavutetaan vähintään 800 luxia, jonka jälkeen kuitataan yläasteikon pistekalibrointi pitkällä zero-painikkeen painalluksella. Tämän jälkeen oikosulkupala irroitetaan ja odotetaan, että näytön arvo on alle 200 luxia ja hyväksytään kaksi pisteinen kalibrointi pitkällä zero-painikkeen painalluksella.

Uudessa tuotantotestilaitteessa toiminnallisuus täytyi suorittaa siten, että käyttäjän ei tarvitse asettaa oikosulkupalaa piikkirimaan. Oikosulkupalan asettamista täytyi simuloida ohjelmallisesti,

joten täytyi selvittää piirikortin rakenteesta, miten tämä on mahdollista. Piirikortilla S1 piikkirima on liitoksissa identtisellä tavalla vanhan testilaitteen span-painikkeen kanssa mikroprosessorin jalkoihin. Span-tulon aktivoimista pystyttiin siis hyödyntää simuloimaan oikosulkupalan kytkentää NG testerissä.

Valoisuuden kalibrointia varten ei tarvittu uusia LabVIEW VI:tä. Toimeksiantajalla valmiina olevia ohjelmia käytettiin sekvenssitiedostossa toteuttamaan valoisuuden kalibrointi. Sekvenssitiedostoon lisättiin valmiina olevat digital outputteja ohjaavat ohjelmat, joilla simuloitiin oikosulkupalan kytkentä ja kalibroinnin hyväksyminen. Kappaleessa 6.5.4 on saatavilla lisätietoa valoisuuden kalibroinnista, kappaleessa on esitetty esimerkki sekvenssiratkaisusta, jossa valoisuuden kalibrointi on esillä.

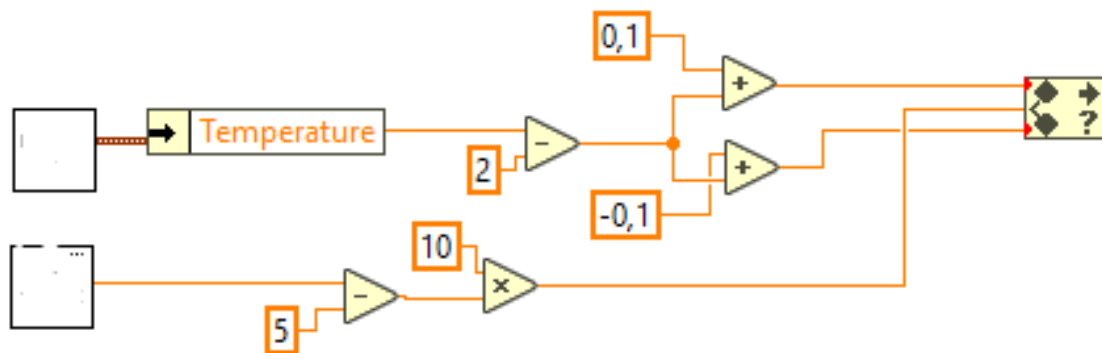
6.5.4 LUX 34 laadunvarmistus

Tuotteen kalibroinnin jälkeen täytyi tuotteelle suorittaa laadunvarmistus, jossa todetaan onnistuneet kalibroinnit ja tuotteen luotettava toiminta. Laadunvarmistuksessa on tarkoitus tarkastella tuotteen lämpötilaerotusta referenssiarvoon ja valoisuuden arvoa eri valoisuuksissa. Tuotteen lämpötilan täytyy olla kalibroinnin jälkeen kaksi celsius astetta pienempi kuin referenssin ja sallittu poikkeama tästä on ± 0.1 celsius astetta. Valoisuusmittauksen laatu todetaan kolmessa eri valoisuusasteessa, 1000-, 500- ja 0 luxissa. Valoisuudenskaala tuotteessa täytyy myös vaihtaa 500 luxin mittauksessa, jotta saadaan todennettua molempien alueiden mittausvarmuus.

Edellisen tuotantotestilaitteen työohjeessa oli kerrottu, että tuotteen kalibrointi tallentuu mikroprosessorille heti kahden pisteen valoisuuskalibroinnin jälkeen. Valoisuuskalibroinnin jälkeen LUX 34:lle tehdään jännitekatkos. LUX 34 käynnistyy jännitekatkoksen jälkeen lopulliseen toimintalaansa ja on valmiina laadunvarmistusta varten.

Tuotantotestilaitteen laadunvarmistusta varten luotiin uusi LabVIEW VI, jonka tarkoitus on suorittaa lämpötilan laadunvarmistus. Laadunvarmistukseen LabVIEW VI:n tulee lukea tuotteen lämpötilaa jänniteviestistä ja verrata sitä sekvenssitiedostosta syötettyihin parametreihin. Sekvenssitiedoston parametrit antavat lämpötilan ala- ja ylärajat, jonka välissä mittaustulos täytyy olla.

Lohkokaavioon oli sijoitettava samanlainen kuvion 20. mukainen tuotteen lämpötilan skaalaus, jonka jälkeen lämpötilalle suoritettiin raja-arvoinen vertaustoiminto. Referenssin arvosta täytyi vähentää kaksi celsius astetta ja lisätä sen jälkeen raja-arvo parametrit yhteenlasku operaattoreiden avulla lämpötila-arvoon. Kun raja-arvot ovat lisättyinä, voidaan yhteenlaskettuja referenssiarvoja käyttää "In range and coerce" funktiossa raja-arvoina tuotteen lämpötilalle (ks. Kuvio 23.).



Kuvio 23 Lämpötilan laadunvarmistuksen vertaustoiminto LabVIEW:ssä

Kuviossa 23. esitetyn toiminnallisuuden jälkeen täytyi lohkokaaavioon luoda kaksi Loop-lohkoa, jotka valitaan ohjelmassa sen perustella, onko lämpötila-arvo raja-arvojen sisällä vai sen ulkopuolella. Jos lämpötila ei ole raja-arvojen luokitteleman skaalan sisällä, siirrytään Loop-lohkoon, jossa toiminnallisuutena on LabVIEW:n wait-funktio. Funktion tarkoitus on odottaa sille syötetty aika ennen uuteen mittaukseen siirtymistä. Kun wait-funktion aika on kulunut, suoritetaan kuvion 23. esittämä vertaustoiminnallisuus uudestaan ja valitaan sen perusteella uudestaan mihin Loop-lohkoon siirrytään.

Lämpötilan saavuttaessa arvo, joka sijoittuu raja-arvojen määrittämälle skaalalle, suoritetaan Loop-lohko, jossa on laskuri toiminto. Laskuri toiminto laskee onnistuneiden mittausten määrän ja vertaa sitä sekvenssitiedostosta syötettyyn hyväksytyjen mittausten lukuun. Kun Loop-lohkossa sijaitseva laskuri saavuttaa sekvenssitiedostosta syötetyn hyväksyntä määrän, hyväksytään laitteen lämpötilan mittaus ja siirrytään seuraavaan ohjelmaan sekvenssitiedostossa. Laadunvarmistuksen

suorittavasta sekvenssitiedoston osasta on esitetty esimerkkirakenne kuviossa 24. Kuviossa 24. esitetään käytettyjen LabVIEW VI:den tarpeellisia toiminnallisuuksia, jotta laadunvarmistus onnistuu.

```
# Esimerkki LUX34 Laadunvarmistuksesta                               Parametrit

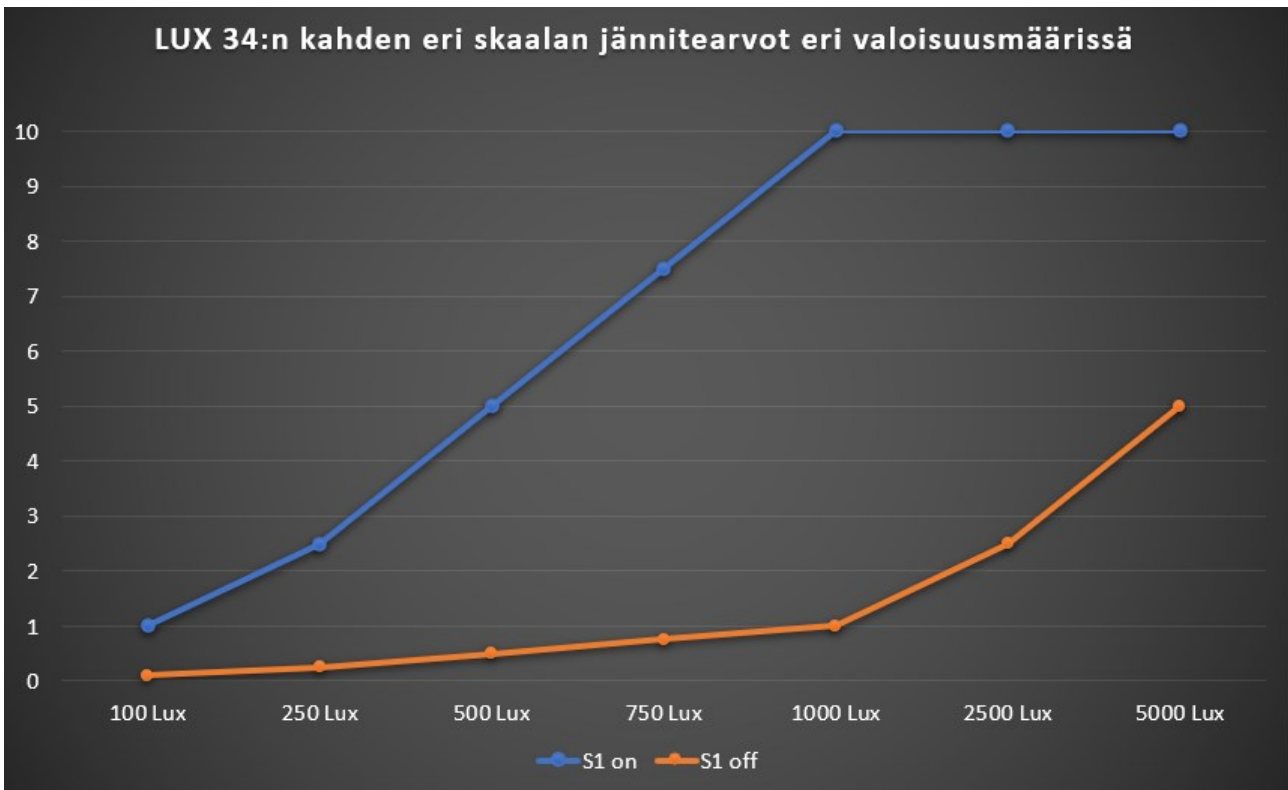
# Valoisuuden Kalibrointi
16. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Span-control
17. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvo 8; Yli 8 Volttia
18. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Zero-control
19. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 0;#Portti Zero-control
20. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 0;#Portti Span-control
21. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvo 2; Alle 2 Volttia
22. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Zero-control
23. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 0;#Portti Käyttäjännite

# Laadunvarmistus / Lämpötila & valoisuus
24. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Käyttäjännite
25. Testiaskellus=Lämpötilan-laadunvarmistus,                      #Raja-arvot 0.1 & -0.1; #Hyväksytty kun 3 x rajoissa.
26. Testiaskellus=Analoginen-output-tilanohjaus,                  #Tila 10V;#Portti Valoisuuslähde
27. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvot 0.95/1.05;#Portti Valoisuudenmäärä; Suorita 10x
28. Testiaskellus=Analoginen-output-tilanohjaus,                  #Tila 5V;#Portti Valoisuuslähde
29. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvot 0.45/0.55;#Portti Valoisuudenmäärä; Suorita 10x
30. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 1;#Portti Span-control
31. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvot 4.5/5.5;#Portti Valoisuudenmäärä; Suorita 10x
32. Testiaskellus=Analoginen-output-tilanohjaus,                  #Tila 0V;#Portti Valoisuuslähde
33. Testiaskellus=Jännitteen-evaluointi,                          #Raja-arvot 0.0/0.1;#Portti Valoisuudenmäärä; Suorita 10x
34. Testiaskellus=Digitaalinen-output-tilanohjaus,                 #Tila 0;#Portti Käyttäjännite
```

Kuvio 24 Laadunvarmistuksen esimerkkirakenne sekvenssitiedostossa

Valoisuuden mittausvarmuus täytyi todentaa sen jälkeen, kun lämpötila on hyväksytty laadunvarmistuksessa. Valoisuuden ohjausta on säädetty sekvenssitiedostossa käyttämällä valmiina olevia LabVIEW VI:tä kuvion 24. esittämän esimerkin tavoin. Ohjelmien avulla saadaan asetettua testilaitteen valoisuuslähettimelle jänniteviesti, jolla sen valotehoa voidaan säädellä. Ensimmäiseksi laadunvarmistuksessa säädetään valoisuuslähde 1000 lux:in valotehoon kuviossa 24. esitetyn 26. testiaskelluksen tavoin. Valotehon säädön jälkeen on käytetty lukuja arvioivaa ”evaluointi” VI:tä, jossa LabVIEW VI suorittaa laskennan, onko tuotteen antama jänniteviesti 0.95 V:n ja 1.05 V:n välillä ja toistaa laskentaa sekvenssitiedoston määrittelemän suorituskerran ja aikavälin ajan.

Tuotteen ollessa annettujen raja-arvojen sisällä, säädetään seuraavassa VI:ssä testilaitteen valoisuuslähde syöttämään 500 lux:in valoisuutta. Toiminnallisuus laadunvarmistuksessa on sama kuin 1000 lux:in laskennassa. Tuotteen valoisuuden jänniteviesti tulee olla lukuja arvioivassa VI:ssä tässä mitauksessa välillä 0.45 V–0.55 V. LUX 34:n valoisuuden arvoja eri skaaloilla ja eri valoisuusmäärissä on havainnollistettu kuviossa 25.



Kuvio 25 LUX 34:n skaalauksen vaikutus jännitteenmäärään

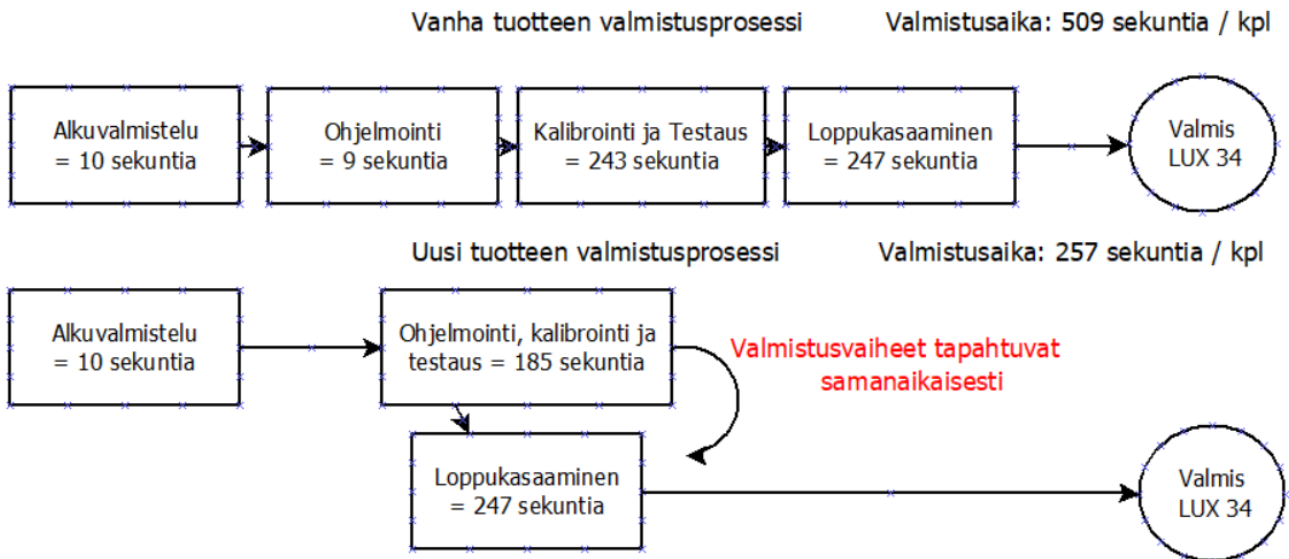
Seuraavaksi tuotteen span-tulo aktivoidaan käyttämällä digital outputia ohjaavaa VI:tä, jolloin tuotteen valoisuudenskaala muuttuu asteikolle 0–1000 lux:ia. Kuvion 25. mukaan, LUX 34:n skaalauksen muutos täytyy huomioida siirryttäessä arvioimaan jännitettä seuraavaksi span-tulon aktivoimisen jälkeen. Tämän jälkeen käytetään taas arvioivan VI:n laskentaa varmistamalla, onko tuotteen antama valoisuuden jänniteviesti välillä 4.5 V–5.5 V. Lopuksi valoisuustehoa säätelevään VI:hin syötetään sekvenssitiedostosta käsky, jolla valolähteen jännite lasketaan 0V:iin. Kun valotehoa ei ole, suoritetaan viimeinen arviointi, joka varmistaa tuotteen jänniteviestin olevan 0–0.01 V:ia. Tämän jälkeen laite hyväksytään laadunvarmistuksessa ja testiprosessi on suoritettu onnistuneesti.

7 Tulokset

Opinnäytetyön tarkoitus oli suunnitella ja rakentaa täysin automaattinen kalibrointi- ja testauslaite LUX 34 valoisuus- ja lämpötilälähtimelle, joka hyödyntää automaattisuudessaan LabVIEW-ohjelmointikielellä rakennettua sekvenssiä. Työn lopputuloksena oli automaattisuudeltaan valmis kalibrointi- ja testilaitte, joka suorittaa LUX 34 tuotteen ohjelmoinnin, kalibroinnin ja laadunvarmistuksen ilman tuotantotyöntekijää. Automaattitestilaitteen LabVIEW-sekvenssi on todettu toimivaksi ja testilaitteen hyväksymälle tuotteelle suoritettu erikseen mittaamalla laadunvarmistus, joka osoitti testilaitteen tuloksen olevan erinomaista. Valmistunut sekvenssi ja LabVIEW-ohjelmallisuus oli työn lopussa katselmoitu toimeksiantajan ohjelmointi-insinöörin kanssa. Tuotantotestilaitteen hyväksymiltä laitteilta mitattiin yleismittarilla lähtöjen jännitesignaaleita eri lämpötilassa ja valoisuudessa, joka todisti tuotteeseen suoritettujen kalibroinnin olevan toimiva. Virhetilanteita ei ilmennyt valmiilla sekvenssillä prototyyppi fixtuurin kanssa. Virheiden kartoittamisessa ei kuitenkaan voitu analysoida vielä tuotantokäytössä olevaa tuotantotestilaitetta.

Työ valmistui ajallaan, mutta testilaitteen tuotantovalmius jäi saavuttamatta, koska testilaitteeseen on vasta työn valmistuttua tilattu tuotekohtainen NG tester fixture, joka tulee lopulliseen tuotantovalmiiseen testilaitteeseen. Työn lopputulos on silti toimeksiantajan toivomuksien mukainen ja kriteerit täyttävä. Testilaitteen viimeisen osan saapuessa se voidaan kytkeä suoraan testilaitteeseen, koska testilaitteen automaattisuuden käyttämät I/O portit ovat identtiset tilatun fixtuurin I/O osoitteiden kanssa.

Lopputuloksen arvioinnissa käytettiin toimeksiantajan kanssa luotua aikahyötylaskentaa. Aikahyötylaskenta ottaa huomioon tuotteen valmistukseen kuuluvat osa-alueet ja edellisellä testilaitteella kuluvan testaus ajan. Uuden testilaitteen hyötynä on, että työntekijä vapautuu automaattisen testiprosessin takia kokoamaan tuotetta sen sijaan, että työntekijän tarvitsisi manuaalisesti kalibroida tai tarkistaa tuotteen oikea toiminta. Uuden testilaitteen sekvenssinsuoritus aika on keskiarvo, joka on saatu suorittamalla prototyyppi fixtuurilla ja valmiilla sekvenssillä useita testejä LUX 34:n piirikortille.



Kuvio 26 Aikahyötylaskenta uudelle valmistusprosessille

Aikahyötylaskenta kuviossa 26. esittää eroteltuna tuotteen jokaisen valmistuksen osa-alueen ja niiden keston. Automaattisen testilaitteen ansiosta, voidaan analysoida yhden osa-alueen vähenevän vanhaan testausmalliin verrattessa. Vanhassa testausmallissa loppukasaaminen tulee suorittaa vasta, kun työntekijä on suorittanut tuotteen testauksen. Uuden testilaitteen ansiosta tuotteen valmistusprosessiin kuluva aika on noin kaksi kertaa nopeampaa kuin vanhalla valmistusprosessillä. Taulukossa 1. on analysoitu valmistusprosessin vaikutusta tarkemmin.

Taulukko 1 Uuden testiprosessin vaikutukset tuotantoaikaan.

Valmistettu kappalemäärä	Vanhan testiprosessin valmistusaika (h)	Uuden testiprosessin valmistusaika (h)	Säästetty aika (h)
5000	706	355	351
4000	564	284	280
3000	423	213	210
2000	282	142	140
1000	141	71	70
500	70,5	35,5	35

LUX 34 tuotetta valmistetaan tuotannossa noin 3000 kappaletta vuodessa. Taulukko 1. osoittaa tämänhetkisellä valmistusmäärällä ajallisen hyödyn olevan 210 tuntia vuodessa. Mahdollisesti kasvavan LUX 34 valmistusmäärän myötä ajallinen hyöty tulee kasvamaan suhteessa aina vieläkin suuremmaksi.

8 Pohdinta

Tämä kehittämistutkimus pyrkii hyödyntämään oikeellisuuden ja luotettavuuden takia alkuperäislähteitä. Tekstissä on viitattu lähteen alkuperään oikeudenmukaisesti ja lähteen alkuperää kunnioitetaan selkeillä viittaustavoilla. Kehittämistutkimuksen tiedonkeruu on suoritettu eettisesti ja tutkimuksessa esitetyt tiedot on pyritty esittämään tarkasti ja huolellisesti. Tiedonhaussa oli ongelmana löytää ajankohtaisia lähteitä, joissa käsitellään tuotantotestausta. Työn kerronnassa on huolehdittu pitämään toimeksiantajan puolelta määrätyt asiat salaisina hyödyntämällä esimerkkikuvauksia tarkempien rakenteiden osoittamisen sijaan.

Työn oleellisin osuus oli LabVIEW-koodien ja sekvenssin luominen testilaitteelle. Ohjelmat ja sekvenssi olivat työn lopussa hyvin suunniteltu kokonaisuus, joka mahdollisti kaiken tarvittavan toiminnallisuuden. Työn aikana luodut LabVIEW VI:t, kuten MSP-430-ohjelmointi VI, ovat tallennettuna toimeksiantajan versionhallintaan ja ovat käytettävissä suoraan seuraavissa projekteissa. Tämä helpottaa jatkossa samankaltaisten projektien rakentamista ja tukee toimeksiantajan tulevia töitä.

Tuotantotestilaitetta ei työn aikana saatu tuotantovalmiiksi, joten luotettavuudeltaan työ perustuu vain prototyypillä onnistuneisiin testeihin ja toimeksiantajan suorittamaan koodin katselmointiin. Toiminnallisuus on katselmoitu projektiin osallistuvien kesken ja tuotantotestilaite on toiminut odotusten mukaisesti. Lopullisen fixtuurin saapuessa täytyy laite koestaa tuotantokäytössä, jossa voi alkuun ilmetä joitain ongelmia mitä ei muutamien tuotteiden valmistuksessa vielä ilmaantunut.

Testilaitteessa ilmeni työn edetessä silti mahdollisia jatkokehitystarpeita, joita olivat jäljitettävyyys ja LUX 34-100 tuotteen sekvenssi. LUX 34-100 tuotteen sekvenssin luominen on helppo tehdä saman testilaitteen käytettäväksi, koska tuotteen piirikortti on sama ja kalibrointi- ja testausprosessi on melkein identtinen LUX 34:n kanssa. LUX 34-100 tuote käyttää eri valoisuusasteikkoa, joten sekvenssin raja-arvojen muokkauksella saadaan myös kyky testata samalla laitteella molempia tuotteita. Toimeksiantajalla on olemassa kyky tuotannossa tehtävien tuotteiden jäljitettävyyteen. Kytkemistä ei tällä hetkellä toteuteta, sillä toimeksiantaja näkee vanhan tuoteperheen tuotteen muuttamisen jäljitettävyyteen liian haastavaksi. Mikäli jäljitettävyyys lisättäisi, tulisi tuote suunnitella sille yhteensopivaksi.

Toimeksiantajan palautteen mukaan työn tulos vastasi täysin odotuksia. Työn aikataulut oli onnistuneesti suunniteltu ja toteutunut. Omasta näkökulmasta tämä tutkimus oli osittain hankala toteuttaa, koska LabVIEW-ohjelmointikielen osaamistaso oli projektin alussa suppeaa. Tutkimuksen myötä osaaminen LabVIEW-ohjelman käytössä kasvoi huomattavasti. Tutkimustyö kokonaisuudessaan oli palkitsevaa ja opettavaa sen haastavuuden takia. Tutkimustyötä tehdessä saavutetun osaamistason ansiosta on helpompi jatkossa alkaa kehittää samankaltaista automaattisuutta myös muihin tuotannossa valmistettaviin laitteisiin.

Lähteet

Can LabVIEW C?. 2023. Informaatio julkaisu LabVIEW:n ohjelmallisuudesta National Instrumentsin verkkosivuilla. Viitattu 20.4.2023. <https://www.ni.com/fi-fi/shop/labview/can-labview-c-.html>.

Create Tasks for NI-DAQmx Devices in LabVIEW. 2022. National Instruments verkkosivuilla oleva artikkeli koskien task toimintojen luomisesta. Viitattu 17.4.2023. <https://knowledge.ni.com/KnowledgeArticleDetails?id=kA03q000000x15ZCAQ&l=fi-FI>.

Daniel, B. 2021. Trenton Systemsin verkkosivuilla oleva blogikirjoitus, jossa käsitellään automaattisia testilaitteita. Viitattu 10.5.2023. <https://www.trentonsystems.com/blog/automatic-test-equipment-overview>.

Kananen, J. 2015. Kehittämistutkimuksen kirjoittamisen käytännön opas: miten kirjoitan kehittämistutkimuksen vaihe vaiheelta. Jyväskylä: Jyväskylän ammattikorkeakoulu. Viitattu 5.4.2023. <https://janet.finna.fi/Record/jamk.992903484806251?sid=2928797835>.

LabVIEW Block Diagram Explained. 2023. Informaatio julkaisu lohkoakaaviosta National Instrumentsin verkkosivuilla. Viitattu 9.4.2023. <https://www.ni.com/fi-fi/support/documentation/supplemental/08/labview-block-diagram-explained.html>.

LabVIEW Front Panel Explained. 2023. Informaatio julkaisu etupaneelistä National Instrumentsin verkkosivuilla. Viitattu 9.4.2023. <https://www.ni.com/fi-fi/support/documentation/supplemental/08/labview-front-panel-explained.html>.

LUX 34 Ja LUX 34-100 Valoisuus- ja lämpötilalähetin. 2010. Kotka. LUX 34 tekninen esite Pro dual Oy:n verkkosivuilla olevassa linkissä. Viitattu 1.4.2023. <https://produal-pim.rockon.io/rockon/api/v1/int/extmedia/openFile/01TGWJBKBR2SJBBLFHGZG2D5IE7FIVPJW5>.

MSP430-FLASHER. N.d. Texas Instruments verkkosivuilla oleva informaatio sivu koskien MSP430-Flasherin ominaisuuksia. Viitattu 16.4.2023. <https://www.ti.com/tool/MSP430-FLASHER>.

MSP-FET. N.d. Texas Instruments verkkosivulla sijaitseva informaatiojulkaisu ohjelmointilaitte MSP-FET:stä. Viitattu 28.4.2023. <https://www.ti.com/tool/MSP-FET#overview>.

Production line testing and quality assurance. N.d. Informaatio julkaisu BRÜEL & KJÆR yrityksen verkkosivuilla. Viitattu 10.5.2023. <https://www.bksv.com/en/knowledge/applications/production-testing>.

Test Automation Best Practices. N.d. Testiautomaatiota käsittelevä informaatiojulkaisu Viewpoint Systemsin verkkosivuilla. Viitattu 11.5.2023. <https://www.viewpointusa.com/TM/wp/test-automation-best-practices/>.

Tietoa meistä. 2021. Pro dual Oy:n verkkosivuilla kerrottu tieto-osuus. Viitattu 25.4.2023. <https://www.produal.com/fi/about-us>.

Ulkovaloisuuslähetin LUX 34. N.d. Produal Oy:n verkkosivuilla esitetyssä tuotevalikoimassa esiintyvä informaatio sivu LUX 34 valoisuus- ja lämpötilälähtimestä. Viitattu 14.2.2023. <https://www.produal.com/fi/lux-34.html>.

USB-6363. N.d. National Instruments verkkosivuilla oleva infromaatiojulkaisu monitoimi-I/O laitteen ominaisuuksista. Viitattu 12.4.2023. <https://www.ni.com/fi-fi/support/model.usb-6363.html>.

Using the Project Explorer Window (Unit Test Framework Toolkit). 2023. Verkkojulkaisu National Instrumentsin verkkosivuilla. Viitattu 2.4.2023. https://www.ni.com/docs/en-US/bundle/labview-unit-test-framework-toolkit-api-ref/page/lvutfconcepts/utfc_project.html.

What is JTAG?. N.d. Artikkel Corelis verkkosivuilla. Viitattu 11.5.2023. <https://www.corelis.com/education/tutorials/jtag-tutorial/what-is-jtag/>.

What is LabVIEW?. N.d. Artikkel Electronics-notes verkkosivuilla. Viitattu 9.4.2023. <https://www.electronics-notes.com/articles/test-methods/labview/vis-virtual-instruments.php>.

Liitteet

Liite 1. LUX 34 data sheet



1133.31E
29.12.2010

LUX 34 LIGHT LEVEL AND TEMPERATURE TRANSMITTER

LUX 34 is a transmitter measuring the intensity of light and temperature and converting the results into linear 0-10V signals, which can be used for controlling lighting and heating via HVAC system.

The range of 0...1000 lx (S1=on; factory setting) or 0...10000 lx (S1=off) can be chosen by the installer. The range coding jumper switch S1 is located on the board inside the casing.

Also a model LUX 34-100 with the ranges of 0-100 lx and 0-500 lx is available. Factory setting is 0-100 lx.

The range of the temperature output is always -50 ... +50°C = 0...10V.

The location for the installation on the wall is recommended to select so, that the transmitter will not be exposed to the direct sunshine or external light sources disturbing the measuring.



Technical data:

Supply	24Vac/dc (22...30Vac/dc) < 0,1VA
Outputs	0-10V, < 1mA
Measuring ranges	
light level LUX 34	0...1000 lx* or 0...10000 lx
light level LUX 34-100	0...100 lx* or 0...500 lx
	* factory setting
temperature	-50 ...+50 °C
Sensor elements	
light level	photodiode
temperature	Pt1000 EN60751 B
Dimensions (w x h x d)	90 x 94 x 44 mm
Housing	IP 54, cable entry down
Materials	PBT, PC and PA

Wiring:

1	24 Vac/dc
2	0V
3	0-10Vdc output (= light level)
4	0-10Vdc output (= -50...+50 °C)

Ordering guide:

Model	Product number	Description
LUX 34	1133310	light level transmitter 1000 lx / 10000 lx
LUX 34-100	1133311	light level transmitter 100 lx / 500 lx

Products fulfil the requirements of directive 2004/108/EEY and are in accordance with the standards EN61000-6-3: 2001 (Emission) and EN61000-6-2: 2001 (Immunity).

Produal Oy

Keltakalliontie 18
48770 Koska
FINLAND
www.produal.com

Tel: +358-5-230 9200
Fax: +358-5-230 9210
info@produal.fi

Liite 2. NG tester osarakenne

