

Opinnäytetyö (AMK)

Kone- ja automaatiotekniikka

2019

Samuli Sormunen

OHJELMOITAVAN KÄSITTELYPÖYDÄN OHJELMISTON MÄÄRITTELY JA SUUNNITTELU

Samuli Sormunen

OHJELMOITAVAN KÄSITTELYPÖYDÄN OHJELMISTON MÄÄRITTELY JA SUUNNITTELU

Tämä opinnäytetyö on tehty Pemamek Oy:lle. Työn tavoitteena oli kehittää vakiolaitteisiin uusia ominaisuuksia, joilla lisätään yrityksen kilpailukykyä markkinoilla.

Tarkoituksena on luoda automaattinen ohjelmisto vakiolaitteisiin, joissa kaikki ohjaukset ovat aikaisemmin toimineet manuaalisesti. Ohjauksen pitää pystyä kommunikoimaan sille suunnitellun käyttöliittymän kanssa. Ohjelmiston pitää ylläpitää haluttujen liikkeiden paikkatietoja.

Opinnäytetyössä käydään läpi automaatiosuunnittelun prosessia ja sen vaiheita kehitystyön elinkaaren aikana. Tähän kuulu ohjelmistovaatimuksia, toimintojen määrittelyä, suunnittelua ja testauksia. Ohjelmistovaatimukseen ja toimintojen määrittelyyn kuuluvat asiakkaan ja toimeksiantajan vaatimusten ja tarpeiden selvittäminen. Testauksessa tutustutaan V-malliin ja sen toteuttamiseen eri vaiheissa kehitystyön elinkaarta.

Työssä noudatettiin käytäntöjä ja sääntöjä, jotka määriteltiin suunnitteluprosessissa. Työn lopputuotteena oli vakiokäsittelypöytä, joka sisälsi uuden käyttöliittymän, automaattisen ohjauksen ja liikkeiden paikkatiedon tallentamisen. Vakiokäyttöpöydän käyttöönotto ja siihen kuuluvat testit tehtiin ja hyväksyttiin asiakkaan toimipisteessä.

Ohjelmiston suunnittelun aikana huomattiin tärkeitä muutostarpeita käsittelypöytien kehityksen kannalta. Työllä saavutettiin sille asetetut tavoitteet.

ASIASANAT:

Automaatio, automaatiosuunnittelu, ohjelmisto, suunnitteluprosessi ja testaus.

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Mechanical Engineering

2019 | 25 pages

Samuli Sormunen

PROGRAMMABLE POSITIONER'S SOFTWARE DEFINITION AND DESIGN

This thesis was commissioned by Pemamek Oy. The purpose of this thesis was to develop new features for standard devices that will increase the company's competitiveness in the market.

The goal was to create automatic controls for standard devices that have previously operated all controls manually. The machine control must be in communication with the user interface and it must maintain the positional information of the desired motion.

This thesis examined the process of automated design and its various stages in the development life cycle. This includes software requirements, feature definitions, planning and testing. Software requirements and functional specifications include clarification of customer's and Pemamek's needs. The test part of thesis introduces the V model and its implementation at different stages of the development life cycle.

This work follows the practices and rules defined in the design process. The final product of the thesis is a standard positioner, which includes a new interface, automatic control and recording of position data. Positioner deployment and related tests were performed and approved at the customer working site.

During the design of the software, important needs for changes were identified for the development of positioners. All the goals were achieved.

KEYWORDS:

Automation, automation engineering, software, planning process and testing

SISÄLTÖ

KÄYTETYT LYHENTEET TAI SANASTO	6
1 JOHDANTO	7
1.1 Toimeksiantaja	7
1.2 Työn esittely ja tavoitteet	7
2 OHJELMISTON SUUNNITTELU	9
2.1 Ohjelmistovaatimukset	10
2.2 Toimintojen määrittely	11
2.3 V-malli ja testauksen suunnittelu	11
3 OHJELMOITAVA LOGIIKKA	15
3.1 Komponentit	15
3.1.1 TwinSAFE EL 6900	15
3.1.2 SICK AFM60A	16
3.1.3 B&R Power Panel T50 10.1"	17
3.1.4 Lenze i550	18
4 TOTEUTUS	19
4.1 SPS 1000 CA	19
4.2 Simulointimalli	20
4.3 Käyttöliittymä	21
4.4 FAT & SAT	22
5 POHDINTA	23
LÄHTEET	25

KUVAT

Kuva 1. Yleiskuva käsittelypöydästä ja sen ominaisuuksista.	9
Kuva 2. Beckhoff CX 9020 -ohjelmoitava logiikka (Beckhoff 2019).	15
Kuva 3. Beckhoff EL6900 -turvalogiikka (Beckhoff 2019).	16
Kuva 4. SICK AFM60A -absoluuttianturi (SICK 2019).	17
Kuva 5. B&R Power Panel T50 10.1 -kosketuspaneeli (B&R Industrial Automation 2019).	17
Kuva 6. Lenze i550 -taajuusmuuttaja (Lenze 2019).	18
Kuva 7. SPS 1000 CA -käsittelypöytä.	19
Kuva 8. SPS 1000 CA -käsittelypöydän simulointi -malli.	20
Kuva 9. HMI -paneeli käsittelypöydässä.	21
Kuva 10. Laitteen dynaaminen testi.	22

KUVIOT

Kuvio 1. Suunnittelun sisällön jakautuminen elinkaaren vaiheisiin (Hirvonen ym. 2010, 14).	10
Kuvio 2. Testauksen V-malli (Haikala & Märijärvi 2011).	12
Kuvio 3. Testauksen suunnitteluun vaikuttavia tekijöitä (Haikala & Märijärvi 2011).	14

KÄYTETYT LYHENTEET TAI SANASTO

Lyhenne	Lyhenteen selitys
PLC	Ohjelmoitava logiikka (engl. Programmable Logic Controller)
I/O	Tiedon siirräntä (engl. Input/Output)
HMI	Käyttöliittymä (engl. Human-Machine Interface)
FAT	Tehdashyväksyntä (engl. Factory Acceptance Test)
SAT	Työmaahyväksyntä (engl. Site Acceptance Test)
OPC	Avoin tiedonsiirto standardi (engl. Open Platform Communications)
UA	Yhdistetty arkkitehtuuri (engl. Unified Architecture)
ADS	Automaatiolaitteen määrittely (engl. Automation Device Specification)
MCD	Mekatroniikan konseptisuunnittelija (engl. Mechatronics Concept Designer)

1 JOHDANTO

1.1 Toimeksiantaja

Pemamek Oy on vuonna 1970 perustettu Loimaalla toimiva perheyriutus, joka keskittyy hitsaus- ja tuotantoautomaation sovelluksiin. Pemamek työllistää yli 230 henkilöä. Pemamek on toimittanut jo yli 15000 erilaista hitsauslaiteratkaisua asiakkailleen yli viiteenkymmeneen eri maahan. Pemamek on erikoistunut automatisoidun hitsaamisen ja tuotantojärjestelmien suunnitteluun sekä valmistamiseen.

Pemamek pyrkii luomaan hyviä luottamussuhteita asiakkaisiin, yhteistyökumppaneihin ja työnkeijöihin. Yrityksen toiminta on hyvin asiakaslähtöistä ja pyrkii aina parantamaan asiakkaan tuottavuutta, sekä kilpailukykyä markkinoilla. Innovaatio on Pemamekille hyvin tärkeä osa yrityksen imagoa ja se pyrkii olemaan markkinoilla automatisoidun hitsausteknologian edelläkävijä. (Pemamek 2019)

1.2 Työn esittely ja tavoitteet

Opinnäytetyön tavoitteena oli kehittää ja parantaa Pemamekin vakiolaitteisiin kuuluvien käsittelypöytien toiminnallisuutta ja niiden hyötyjä. Tässä opinnäytetyössä keskityttiin ohjelmoitavan logiikan (PLC) suunnitteluun ja toteutukseen. Aikaisemmin pöydät ovat toimineet käsiohjauksella, jolloin käyttäjä oli itse vastuussa siitä, että kappale ajetaan oikeaan asentoon. Ohjelmiston uudistuksella käyttäjä pystyy tallentamaan paikkatietoja ja ajamaan niitä uudestaan ohjelmiston muistista. Tämä helpottaa laitteen käyttö ja tehostaa työprosessia.

Ohjelmoitavien käsittelypöytien kehittäminen on tärkeää Pemamekin vakiolaitteiden tulevaisuudelle ja kilpailukyvyyn parantamiselle. IOT-alan kehityksen myötä on kasvanut tarve kerätä enemmän tietoa laitteistoista. Käsittelypöytien manuaaliset toiminnot eivät enää riitä täyttämään asiakkaiden tarpeita. Myös teollisuudessa tarve toimintojen ja prosessien automatisointiin on kasvanut. Tiedonkeruun ja automatisoinnin kysynnän takia näitä ominaisuuksia halutaan lisätä Pemamekin laitteisiin.

Opinnäytetyön alussa suurin osa lähdeaineistosta tuli suoraan Pemamekiltä ja asiakkaalta. Tämä lähdeainesto perustui aikaisempiin suunnitelmiin ja päätöksiin työn toteuttamisesta, mikä toimi pohjatyönä opinnäytetyölle. Ulkopuolista lähdeaineistoa käytin työssä suunnittelun eri vaiheiden ja tarvittavien komponenttien teknisten tietojen selvittämiseen. Saatua lähdeaineistoa projektin aloittamiseen oli hyvin puutteellista. Tässä työssä käsitelen näiden ongelmien selvittämistä. Työssä ensisijaisesti selvitän ohjelmiston modulaarisia mahdollisuuksia, paikkatiedon ylläpitämiseen tarvittavia toimenpiteitä ja sitä miten näitä pystytään testaamaan.

Työn alussa käsitellään ohjelmistosuunnittelun eri vaiheita. Tämä sisältää: Vaatimusten määrittelyn, toimintojen määrittelyn, arkkitehtuurin suunnittelun, ohjelmiston suunnittelun, ohjelmoinnin ja ohjelmiston testausta. Ohjelmiston testauksessa tutustutaan testauksen V -malliin. Lopussa kerrotaan opinnäytetyön tuloksia ja tuotoksia.

2 OHJELMISTON SUUNNITTELU

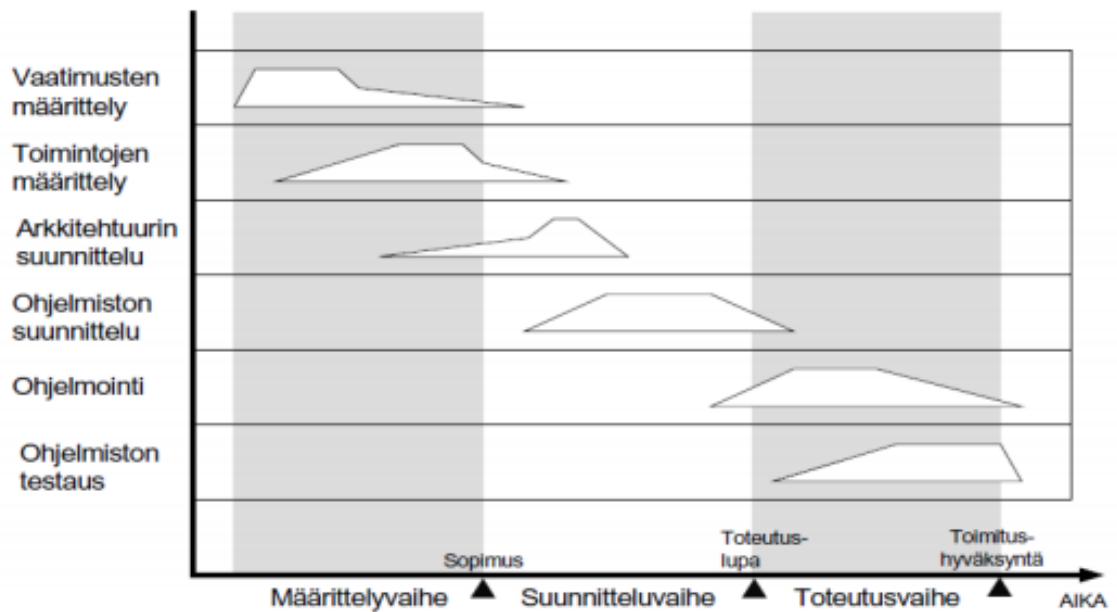
Automaatiosuunnitteluprojekti voi olla toimeksiantajan esittämä ongelma, johon yritys koittaa kehittää ratkaisua. Se voi olla vanhentuneen tekniikan parantamista, prosessin tai prosessin osan automatisointia. Automaatiosuunnittelussa on tärkeintä ylläpitää huolellisuutta koko suunnittelun elinkaaren aikana, jotta saavutetut ratkaisut ovat toimeksiantajan ja laitetoimittajan tarpeiden ja vaatimusten mukaisia. Laitetoimittaja voi jakaa automaatiosuunnittelun alustavaan suunnitteluun, perussuunnitteluun, toteutussuunnitteluun, korjauksiin ja täydennyksiin. (Vuorenmaa 2014, 5.)



Kuva 1. Yleiskuva käsittelypöydästä ja sen ominaisuuksista.

Onnistunut suunnitteluprosessi edellyttää, että toimeksiantajalta kerätään mahdollisimman laajat lähtötiedot. Lisäksi osapuolilla on oltava tarkka ja yhtenevä ajatus sekä suunnittelun kulusta että suunniteltavasta kohteesta ja sen ominaisuuksista (kuva 1).

Suunnitteluvaiheet voidaan jakaa kolmeen päävaiheeseen, joita ovat määrittelyvaihe, suunnitteluvaihe ja toteutusvaihe (Kuva 1).



Kuvio 1. Suunnittelun sisällön jakautuminen elinkaaren vaiheisiin (Hirvonen ym. 2010, 14).

Tässä työssä suunnittelun vaiheet näkyvät luvuissa 2 ja 4. Määrittelyvaihetta käydään läpi luvuissa 2.1 ja 2.2, suunnitteluvaihetta käsitellään luvussa 2.4 ja 2.5 ja toteutusvaihe on käsitelty luvussa 4. Luku 2.3, joka sisältää tietoa testauksen suunnittelusta ja toteuttamisesta, liittyy suunnittelun jokaiseen vaiheeseen. Luvut 2.4 ja 2.5 on poistettu julkaisusta salassapito syistä.

2.1 Ohjelmistovaatimukset

Suunnittelu aloitettiin määrittelemällä kaikki ohjelmiston vaatimukset. Ohjelmistovaatimukseen vaikuttaa aina asiakkaan ja toimittajan omat odotukset ja tarpeet laitteelle. Suurin osa asiakkaan vaatimuksista liittyy turvallisuuteen, kappaleen lastaus- ja purkusekvenssiin sekä laitteen mekaanisiin rajoihin. Toimittajan puolesta suunnittelussa keskityttiin tulevaisuuden näkymään ja siihen, miten suunnitellaan sellainen ohjelmisto, jolla olisi jatkossa useampia käyttökohteita.

Ohjelmiston suunnitteluvaiheessa oli suunniteltava kaikki ohjelmistoon liittyvät kohdat sillä periaatteella, että niiden käyttö olisi tulevaisuudessa mahdollisimman helppoa. Tämä vaati ohjelmiston jakamista moduuleihin ja niiden pitämistä toisistaan

riippumattomina. Vakiolaitteen perustoiminnoille oli jo olemassa ohjelmia, joiden rinnalle pystyttiin kehittämään haluttujen toimintojen moduuleja.

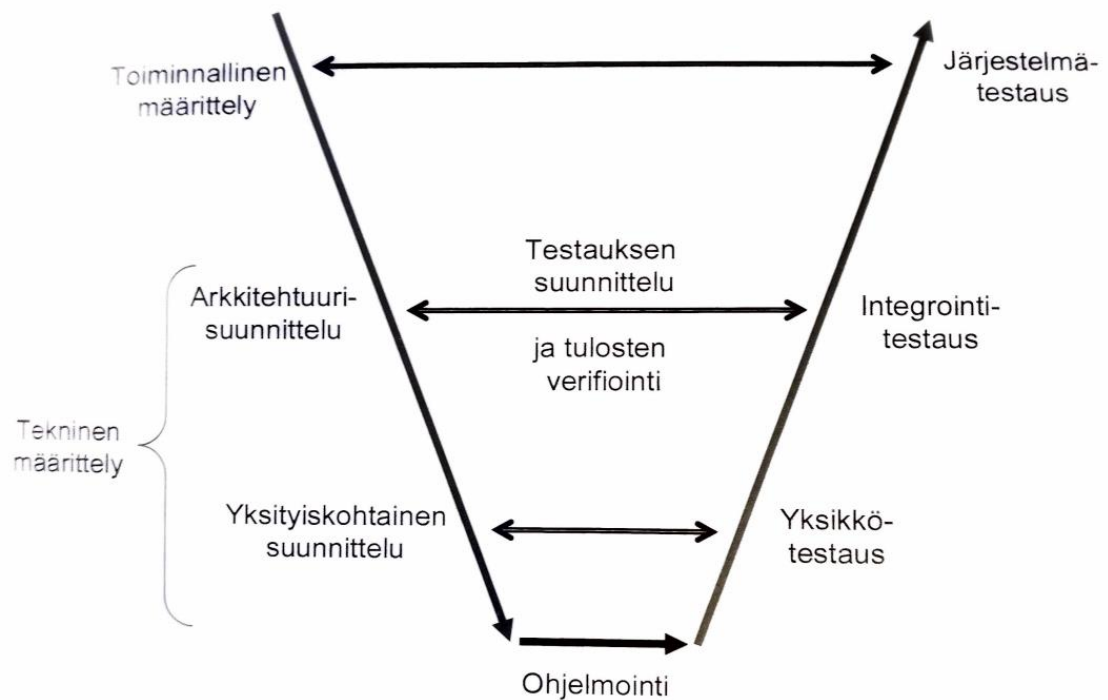
2.2 Toimintojen määrittely

Määrittelyvaiheessa otetaan huomioon automaatiojärjestelmän vaatimukset ja toiminnot, sen tarkempaa suunnittelua ja toteutusta varten. Määrittelyvaiheeseen kuuluvat myös esi- ja perussuunnittelu. Esisuunnittelussa asiakas laatii käyttäjävaatimukset ja alustavan kelpuutus suunnitelman. Perussuunnittelussa kuvataan automaatiojärjestelmän toiminnot toteutusta, tarkempaa suunnittelua ja sopimusta varten. (Hirvonen ym. 2010. 16)

Toimintojen määrittelyssä käytettiin hyödyksi yleiskuvaa käsittelypöydästä ja sen ominaisuuksista (Kuva 1). Toimintojen määrittelyyn vaikuttaa myös asiakkaan vaatimukset, joita myös käsiteltiin luvussa 2.1. Toimintojen määrittelyssä tärkeänä osana on myös testauksien suunnittelu.

2.3 V-malli ja testauksen suunnittelu

Tärkeä osa kehitystyön elinkaarta on testauksien suunnittelu ja niiden toteuttaminen. Suunnittelun ja kehitystyön elinkaaren suhdetta testaukseen voidaan havainnollistaa V-mallilla. V-mallin avulla seurataan ja suunnitellaan suoritettavia testauksia kehitystyön eri vaiheissa.



Kuvio 2. Testauksen V-malli (Haikala & Märijärvi 2011).

Haikala ja Märijärvi (2011, 206-208) jakavat V-mallin testaustasot kolmeen eri ryhmään, joita ovat yksikkötestaus, integrointitestaus ja järjestelmätestaus (Kuvio 2). **Yksikkötestauksessa** testattava kohde on yksittäinen moduuli tai luokka. Tällä tarkoitetaan usein ohjelman pienintä mahdollista osaa. Moduulin tekijä suorittaa pääsääntöisesti tämän testauksen. Testin tuloksia verrataan teknisen määrittelyn yksityiskohtaiseen suunnitteluun. Tätä testiä varten yleensä luodaan testipeti, jolla pystytään simuloimaan luokan tarvitsemaa ympäristöä. Testipeti saattaa sisältää testiajureita ja tynkämoduuleja, jotka mahdollistavat ympäristön simuloinnin.

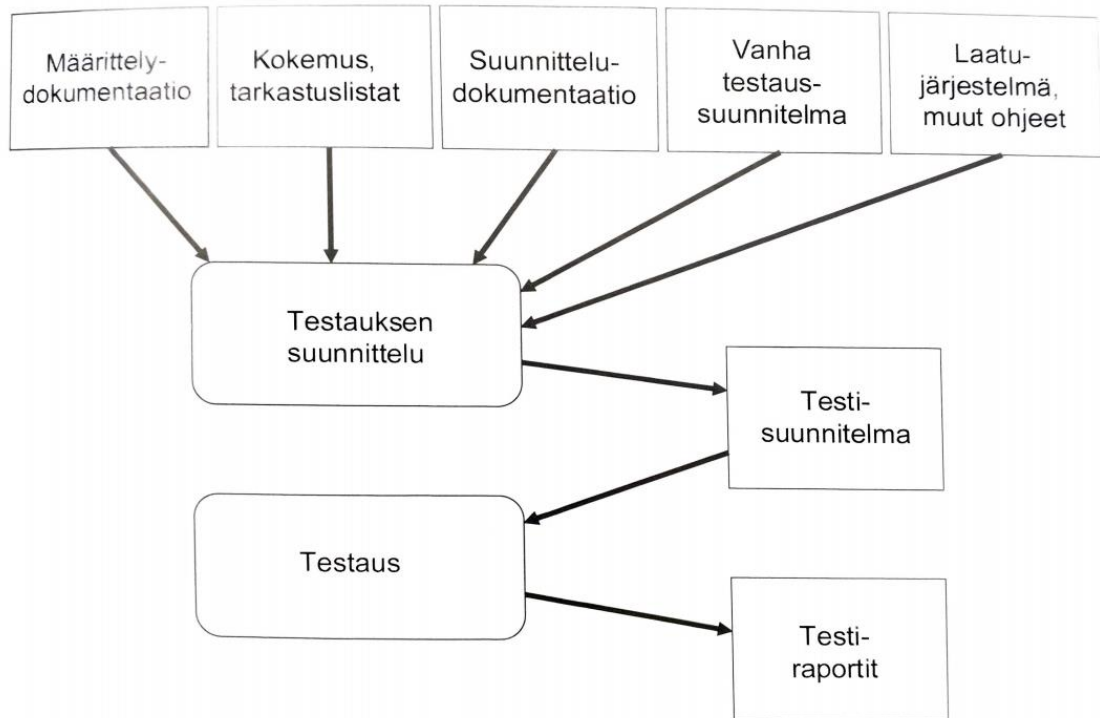
Integrointitestauksessa tarkastellaan useammasta moduulista koostuvaa järjestelmän osaa. Teknistä määrittelyä verrataan integraatiotestauksesta saatuihin tuloksiin. Integrointitestauksessa tarkkaillaan moduulien rajapintoja ja niiden toimivuutta. Yksikkötestauksen osia voidaan hyödyntää integrointitestauksessa.

Järjestelmätestauksessa tarkastellaan ja tutkitaan koko järjestelmän toimivuutta. Testauksesta saatuja tuloksia verrataan ohjelmiston vaatimusmäärittelyyn. Järjestelmätason testaajien pitää olla mahdollisimman riippumattomia järjestelmän suunnittelusta ja kehitystyöstä. Järjestelmätason testauksessa voidaan suorittaa myös mahdollisia hyväksymis- ja kenttätestauksia. Yleensä tässä yhteydessä suoritetaan

kuormitustestit. Kuormitustesteissä suoritetaan staattinen- ja dynaaminen testi. Kuormitustestien avulla pystytään varmistamaan, että laite kestää varmasti sille suunnitellun kuorman. Järjestelmätason testaukseen kuuluu myös luotettavuus-, asennus- ja käytettävyystestaus. Luotettavuustestaus kertoo järjestelmän toimintavarmuuden ja kyvyn palautua virhetilanteesta. Asennustestauksella selvitetään, tapahtuuko asennus odotusten mukaisesti. Käytettävyystestaus tarkastelee järjestelmän hyödynnettävyyttä sen toiminnassa.

Mahdollisten virheiden korjaus muodostuu kalliimmaksi, mitä korkeammalla V-mallia ollaan. Järjestelmätason korjaamisella voi aiheuttaa sen, että korjaustoimenpiteitä pitää tehdä myös alemmilla tasoilla. Mahdollisten korjausten jälkeen testausasojat joudutaan uusimaan, jotta varmistutaan järjestelmän toimivuudesta. Tällaisissa tilanteissa suositellaan kaikkien testien uusimista. Uusituissa testeissä voi ilmetä uusia virheitä, jotka ovat seurausta korjaustoimista. Tätä korjausprosessia kutsutaan regressiotestaukseksi.

Testaussuunnitelmasta selviää, milloin testejä tehdään, millaisia testejä tehdään ja mitä lopputuloksia odotetaan. Testaussuunnittelua tehdessä on tarkasteltava testeihin vaikuttavia tekijöitä ja niiden vaikutusta testausprosessiin (Kuvio 3). On myös tärkeää määrittää testeille niiden lopettamiskriteerit ja hyväksymiskriteerit. Lopettamiskriteerit on kuitenkin tärkeä pitää erillään hyväksymiskriteereistä, koska testaus ei aina onnistu hyväksytysti. (Haikala & Märijärvi 2011, 216.)



Kuvio 3. Testauksen suunnitteluun vaikuttavia tekijöitä (Haikala & Märijärvi 2011).

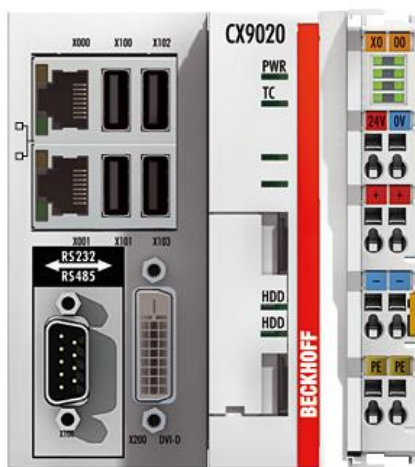
Yksikkötestauksessa päätettiin suorittaa vakiopöydän ohjelman uusien moduulien testaus. Paikkatietoa käsittelevän moduulin testaus sijoittui projektin testauksissa ensimmäisiin. Tämä johtuu siitä, että suurimpana haasteena oli paikkatietojen ylläpitäminen, josta kerrotaan luvussa 2.5.

Integroititestausta suoritettiin Siemens NX MCD -simuloinnin avulla. Integroititestausta tapahtui lähempänä projektin loppupuolta. Tästä löytyy lisää tietoa luvussa 4.2.

Järjestelmätestaus suoritettiin tuotannon puolella. Järjestelmätason testit, jotka suoritettiin, olivat kuormitustesti ja FAT -testi, joista on lisätietoa luvussa 4.4.

3 OHJELMOITAVA LOGIIKKA

Logiikkana käytettiin Beckhoffin CX 9020:tä (Kuva 2). Kyseinen logiikka on Microsoft Windows Embedded Compact 7 -pohjainen järjestelmä. Logiikkaa ohjelmoidaan Beckhoff:n TwinCat 2 PLC sovelluksella, joka on vanhempi versio uusimmasta TwinCat 3 PLC:stä. Logiikka käyttää CANopen -protokollaa. I/O -laitteiden kommunikointi tapahtuu E- tai K-linkillä ja väyläkommunikointi tapahtuu EtherCat pohjalta. (Beckhoff 2019).



Kuva 2. Beckhoff CX 9020 -ohjelmoitava logiikka (Beckhoff 2019).

Vakiopöydissä on aikaisemmin käytetty vakiona Beckhoff CX8050 -logiikkaa, mutta tässä projektissa huomasimme ongelmia kyseisen logiikan kanssa. CX 8050:ssä ei ollut tarpeeksi muistia pyörittämään ulkopuoliseen kommunikointiin tarvittavaa OPC-UA -palvelinta.

3.1 Komponentit

3.1.1 TwinSAFE EL 6900

Beckhoffin turva-automaatiojärjestelmää kutsutaan nimellä TwinSAFE. EL6900 -turvalogiikka ja turvapuolen I/O -kortit ovat kiinnitettynä CX9020 -logiikkaan. TwinSAFE turvlogiikka toimii erillisenä yksikkönä ja se ohjelmoidaan aina erikseen. Tällä varmistetaan ja ylläpidetään laitelta vaadittua turvallisuustasoa.



Kuva 3. Beckhoff EL6900 -turvalogiikka (Beckhoff 2019).

Beckhoffin turvalogiikan (Kuva 3) ja siihen kuuluvat I/O -kortit pystyy tunnistamaan niiden keltaisesta väristä.

Vakiopöydissä ei ole ollut erillistä turvalogiikkaa, koska ne ovat olleet manuaali toimisia ja turvatoimien taso ei ole vaatinut vielä erillistä turvalogiikkaa. Liikkeiden automatisoinnin ja turvatoimien lisäämisen takia, turvalogiikka oli pakollinen lisäys toteutukseen.

3.1.2 SICK AFM60A

Tässä työssä käytettiin SICK valmistajan AFM60A EtherCAT -absoluuttianturia. Absoluuttianturilla saadaan kerättyä tietoa sijainnista, kulmasta ja kierrosluvuista, laitteen eri positiosta. Yksittäinen koodi on kohdennettu jokaiselle positiolle. Erottelukyky kertoo koodien määrän kierroksen aikana. Absoluuttinen paikkatieto saadaan jokaisen koodin muodostamasta yksiselitteisestä referenssistä. Koska paikkatieto ei nollaannu käynnistyksen yhteydessä, referenssikäyttö ei ole tarpeellista. Yksikierros-absoluuttianturi pitää paikkatietoa vain yhden kierroksen sisällä. Monikierros-absoluuttianturi taas määrittää myös paikkatiedon useampien kierrosten sisällä.

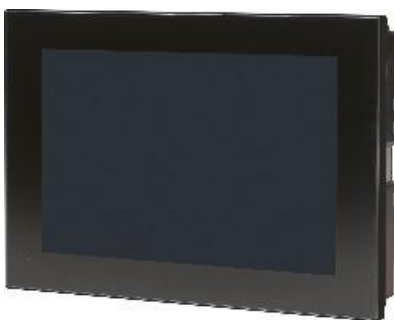


Kuva 4. SICK AFM60A -absoluuttianturi (SICK 2019).

AFM60A -absoluuttianturi (Kuva 4) sisältää 4096 kierrosta, jokaista kierrosta kohden on olemassa 262 144 asemamäärää ja sen resoluutio on 18 bit x 12 bit. Anturista kerätään koodiarvoa EtherCAT väylän avulla logiikalle. (SICK 2019)

3.1.3 B&R Power Panel T50 10.1"

HMI -paneeliksi valittiin B&R Power Panel T50 10.1" (Kuva 5), joka sisältää ARM Cortex-A9 -prosessorin lisäksi kosketusnäytön multi-touch -toiminnolla. Paneeli pystyy lukemaan selainpohjaisen käyttöliittymän CX 9020 -logiikalta. (B&R Industrial Automation 2019)



Kuva 5. B&R Power Panel T50 10.1 -kosketuspaneeli (B&R Industrial Automation 2019).

3.1.4 Lenze i550

Pöytien liikkeen ohjaus tapahtuu Lenze:n i550 -taajuusmuuttajilla (Kuva 6). Taajuusmuuttajat syöttävät muokattua jännitettä ja taajuutta, jolla pystytään säätämään oikosulkumootorin nopeutta portaattomasti.



Kuva 6. Lenze i550 -taajuusmuuttaja (Lenze 2019).

Kommunikointi logiikalta taajuusmuuttajille tapahtui EtherCat -väylän kautta ja niiden parametreja pystyi konfiguroimaan joko suoralla yhteydellä itse taajuusmuuttajaan tai logiikan parametrien kautta. Konfigurointi tapahtui tässä tilanteessa Lenze:n oman EASY Navigator -ohjelman kautta. Kyseisellä ohjelmalla pääsi parametroinnin lisäksi seuraamaan taajuusmuuttajan hetkellisiä arvoja, virheilmoituksia ja taajuusmuuttajan talteen keräämää dataa. (Lenze 2019).

4 TOTEUTUS

4.1 SPS 1000 CA

Opinnäytetyön lopputuotteena oli SPS 1000 CA eli skyhook positioning system (Kuva 7). Asiakkaan käyttämä lyhenne kyseisestä käsittelypöydästä on CA. Tuotenimeen sijoitettu 1000 -luku kertoo laitteelle määritellyn maksimimassan eli tässä tapauksessa 1000kg.



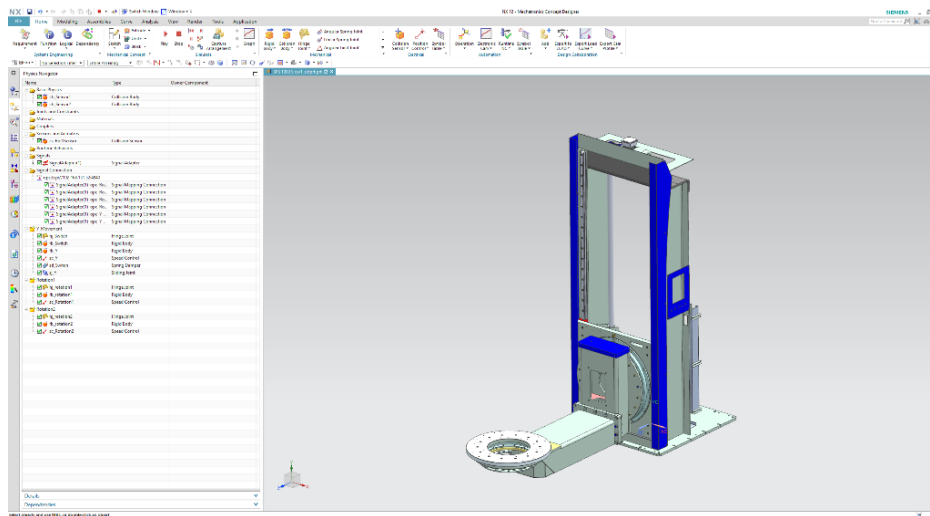
Kuva 7. SPS 1000 CA -käsittelypöytä.

Käsittelypöydällä on yleensä kaksi tai kolme liikeakselia, joita voidaan ohjata joko hydraulisesti tai sähköisesti. Kyseisellä pöydällä on kolme liikettä: nosto, kallistus ja pöydän pyörimys. Kaikkia näitä liikkeitä ohjataan taajuusmuuttajilla.

Vaikka kyseiset käsittelypöydät ovat vakiotuotteita Pemalla, oli tämä projektille ensimmäinen oikea testialusta ja ensimmäinen käsittelypöytä, joka sisälsi uudet ominaisuudet.

4.2 Simulointimalli

Simulointi on mahdollistanut virtuaalisen käyttöönoton, jossa mekaniikka-, sähkö- ja automaatio suunnittelu yhdistyy. Perinteisiin käyttöönottomenetelmiin verrattuna, virtuaalinen käyttöönotto mahdollistaa linjaston tai laitteen testaamisen ennen varsinaisen fyysisen laitteen tuottamista. Tämä helpottaa suunnitteluvaiheen testaamista ja helpottaa teknisten virheiden poistamista. (Tarvas 2018)



Kuva 8. SPS 1000 CA -käsittelypöydän simulointi -malli.

Simulointi tehtiin Siemens NX Mechatronics Concept Designer –ohjelmiston (MCD) avulla (Kuva 8). Simulointia ei alustavan suunnitelman mukaan pitänyt hyödyntää, mutta tuotannon aikataulu myöhästyi alustavasta suunnitelmasta ja täten jouduttiin tekemään ratkaisu sen käytöstä testien tekemistä varten. Simuloinnissa luotiin SOLIDWORKS -kokoontalon avulla liikkeet ja liikkeiden paikkatiedot.

Simulointi kommunikoi logiikan kanssa OPC-UA -palvelimen välityksellä, josta logiikka otti vastaan paikkatiedot ja lähetti ohjeet liikkeille. Liikkeiden ohjetunnus oli joko millimetriä sekunnissa tai astetta sekunnissa. Paikkatieto liikkeiltä tuli logiikalle millimetreinä tai asteina.

Jotta paikkatietoa voitiin hyödyntää testauksessa, jouduttiin se skaalamaan, jotta paikkatieto kuvasti pulssianturin antamaa pulssiarvoa. Pulssianturin maksimiresoluution takia jouduttiin myös luomaan keinotekoinen kierto pulssiarvolle, jotta saatiin testattua pulssialueen ylityksen vaikutus ohjelmassa.

4.3 Käyttöliittymä

Käyttöliittymän toteutus tehtiin Vue Javascript pohjaisella web -sovelluksella. Beckhoff:lla on Javascriptiä varten olemassa ADS -kommunikointikirjasto. Tällä saatiin käyttöliittymä lukemaan ja kirjoittamaan muuttujia PLC:lle.



Kuva 9. HMI -paneeli käsittelypöydässä.

Kyseinen käyttöliittymä (Kuva 9) luotiin tätä projektia varten, koska asiakkaalla oli tärkeitä vaatimuksia käyttöliittymän ulkoasulle ja sen toiminnoille. Käyttöliittymä itsessään ei pysty ohjaamaan käsittelypöytää, vaan sitä käytetään ilmoittamaan koneen paikkatiedot, anturien tilatiedot ja suorittamaan lastaus- sekä purkusekvenssi. Lastausta ja purkua varten asiakas oli dokumentoinut tarkkaan jokaisen sekvenssin askeleen ehdot ja niiden kulkujärjestyksen. Kyseinen käyttöliittymä ei sisällä eri työnkiertojen ja niiden vaiheiden

tallentamista eikä niiden muokkausta. Asiakas järjestää vastaavan toiminnon omalla ylemmän tason kommunikoinnilla OPC-UA -palvelimen avulla.

4.4 FAT & SAT

Laitteiden toimivuus ja hyväksyntätesti (FAT) tehtiin yhdessä asiakkaan kanssa. Testissä käytiin läpi kaikkien toimintojen, liikkeiden ja komponenttien toimivuus, sekä laitteille tehdyt rasitus testit.



Kuva 10. Laitteen dynaaminen testi.

Staattisessa testissä kiinnitettiin käsittelypöydän varteen kappale, jonka massa oli 1,22 kertainen laitteen maksimikuormasta. Kappaletta pidettiin pöydässä kiinni akseleita liikuttamatta määrätyn ajanjakson ajan ja seurataan käsittelypöydän kestävyyttä. Dynaaminen testi (Kuva 10) tehtiin vain 1,1 kertaisella massalla ja se suoritettiin ajamalla kaikkia käsittelypöydän liikkeitä ääriarajoille.

Työmaahyväksynnän testi (SAT) tehtiin asiakkaan tiloissa, kun laite oli asennettu. SAT -testillä asiakas hyväksyy laitteen käyttöönoton ja sen toimivuuden.

5 POHDINTA

Työssä saavutettiin sille asetetut tavoitteet ja toiveet erittäin onnistuneesti. Opinnäytetyön lopputuotteena tehtiin asiakkaalle toimiva tuote, mikä hyväksyttiin ensin tehdashyväksyntätestissä ja vielä asiakkaan toimipisteessä työmaahyväksyntätestissä. Kyseinen tuote sisälsi opinnäytetyön tuotoksia. Vaikka kehitysprojekti jatkuu, eikä asiakasyhteistyö ole vielä loppunut, olivat lopputyön tulokset sitä, mitä tavoiteltiin ja hyvä ensimmäinen etappi kokonaisprojektille. Projektin aikataulu oli hyvin kiireellinen, mutta onnistuimme silti saamaan aikaan toimivan kokonaisuuden tavoiteaikaan mennessä. Asiakkaalta saatu palaute oli positiivista ja pystyimme vastaamaan heidän asettamiin tavoitteisiin. Tämä oli hyvin tärkeä osa myös jatkuvaa asiakassuhdetta ajatellen, koska asiakkaalle on tulevaisuudessa vielä menossa useampi laite.

Seuraavana vaiheena asiakasyhteistyössä on asiakkaan palautteen kerääminen, jonka avulla kehitetään ja muokataan laitteen toimintoja. Peman kannalta keskitymme tulevaisuudessa jatkamaan kehitystyötä vakiolaitteiden parissa, jotta saamme luotua oman käyttöliittymän, mikä mahdollistaa työkiertojen ajon ja muokkaamisen.

Suunnittelun alkuvaiheilla tehtyjen kaavioiden ansiosta ohjelmoinnin aloittaminen ja siinä eteneminen oli helpompaa. Ilman kunnollista suunnitteluprosessia ja sen dokumentointia, olisi ohjelmointityössä ja testauksissa tullut paljon tietoaukkoja, joihin olisi joutunut aina erikseen selvittämään vastauksen.

Käsittelypöydän suunnittelun, tuotannon ja toimituksen aikataulu oli erittäin tiukka ja siihen jouduttiin projektin aikana tekemään pieniä muutoksia. Tästä aiheutui toimitusprosessin aikana virheitä, joihin jouduimme keksimään ratkaisut asiakkaan päässä käyttöönoton yhteydessä.

Isoin haaste projektissa oli mielestäni akselien paikkatiedon ylläpitäminen ja sen testaaminen. Oli vaikeaa luoda absoluuttianturien ylijäämälaskentaa ilman varsinaista pohjaa ja sen testaaminen oli vielä hankalampaa. Alkuperäisten suunnitelmien mukaan olisin voinut testata tätä toimintoa oikealla tuotteella, mutta tuotannon viivästyksien takia päätimme toteuttaa testit simuloinnissa. Tämä oli tärkeä päätös projektille, koska tuotannossa laitteen kasaaminen ja asennus viivästy paljon. Simuloinnilla sain kirittyä lyhennettyä tarvittavaa aikaa laitteen käyttöönotolle. Suurin osa toiminnoista lähti toimimaan heti ohjelmien latauksen jälkeen. Kaikkia järjestelmän toimia ei aina pysty

testaamaan tietokoneen ääressä, joten ohjelmassa joutuu tekemään pieniä muokkauksia.

Projektin aikana tuli myös paljon kehitysideoita yrityksen toimintatavoille ja kehitysprojektille. Yksi suurimmista oli simuloinnin hyödyntäminen automaatio suunnittelussa ja testauksessa. Automaatio suunnittelussa ei olla vielä aikaisemmin käytetty laitteen simulointia hyödyksi, joten tämä avasi uusia kysymyksiä simuloinnin tarpeille tulevaisuudessa. Projektissa tuli myös tärkeää näyttö simuloinnin sekä ajallisista että tehokkuuteen liittyvistä hyödyistä. Kehitysprojektin aikana huomattiin myös, että käyttöliittymäkommunikaatioon tarvitaan uudistuksia ja muokkauksia tulevaisuutta varten.

Itselleni kyseisen projektin parissa työskentely on ollut suuri etu. Uutena työntekijänä pääsin heti kehittämään Peman tuotteisiin jotain uutta, mitä ei ole aikaisemmin tehty. Tämän takia koen, että projekti on opettanut minulle paljon enemmän kuin se, että olisin työskennellyt jo valmiina olemassa olevan järjestelmän parissa. Projektin aikana oppimista tuki projektin laajuus ja kuinka suurin osa asioista oli itse opittava. Koska kyseessä oli paljon uutta asiaa Pemallekin, jouduin selvittämään paljon asioita itsenäisesti ja pohtimaan niitä.

LÄHTEET

Beckhoff 2019. Viitattu 19.6.2019

https://www.beckhoff.com/english.asp?embedded_pc/cx.htm?id=15987759973374

Boijer, M. 2014. Tietojärjestelmien dokumentointi. Opinnäytetyö. Informaatio- ja mediatekniikka. Helsinki: Arcada ammattikorkeakoulu. Viitattu 17.9.2019

https://www.theseus.fi/bitstream/handle/10024/74935/Examensarbete_MikaelaBoijer.pdf?sequence=1&isAllowed=y

B&R Industrial Automation 2019. Viitattu 16.9.2019 <https://www.br-automation.com/en/products/hmi/power-panel-t-c-series/power-panel-t-series/power-panel-t50/6ppt50101e-16b/>

Haikala, I. & Märijärvi, J. 2011, Ohjelmistotuotannon käytännöt, 12. uudistettu painos, Helsinki : Talentum.

Haikala, I. & Märijärvi, J. 2004, Ohjelmistotuotanto, 10. painos, Helsinki : Talentum.

Hirvonen, J.; Hukki, K.; Strömman, M. & Tommila, T. 2010. Automaatiosuunnittelun prosessimalli – Yhteiset käsitteet verkottuneen suunnittelun perustana. (Helsinki). Suomen Automaatioseura Ry, SAS julkaisusarja nro 35. Viitattu 17.9.2019.

https://www.automaatioseura.fi/site/assets/files/1367/automaatiosuunnittelun_prosessimalli.pdf

Lenze 2019. Viitattu 16.9.2019 <https://www.lenze.com/en-de/products/inverters/control-cabinet-installation/i550-frequency-inverters/>

Pemamek 2019. Viitattu 16.9.2019 <https://pemamek.com>

SICK 2019. Viitattu 19.6.2019 <https://www.sick.com/fi/fi/enkooderit/absoluuttianturit/c/g244395>

Tarvas, T. 2018. Virtuaalinen mallinnus mullistaa käyttöönottoa. Viitattu 17.9.2019

http://www.automaatiovayla.fi/wpcontent/uploads/2018/03/Automaatiovayla_2_2018.pdf

Vuorenmaa, E. 2014. Pk-yritysten automaatiosuunnittelu ja automaation käyttö. Opinnäytetyö. Tuotantotalouden koulutusohjelma. Ylivieska: Centria ammattikorkeakoulu. Viitattu 17.9.2019

https://www.theseus.fi/bitstream/handle/10024/75834/vuorenmaa_esa.pdf?sequence=1&isAllowed=y