



Kim Henriksson

Moottorin momenttisäädön toteutus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Sähkö- ja automaatiotekniikka

Insinöörityö

23.5.2025

Tiivistelmä

Tekijä: Kim Henriksson
Otsikko: Moottorin momenttisäädön toteutus
Sivumäärä: 37 sivua
Aika: 23.5.2025

Tutkinto: Insinööri (AMK)
Tutkinto-ohjelma: Sähkö- ja automaatiotekniikka
Ammatillinen pääaine: Automaatiotekniikka
Ohjaajat: Lehtori Tuomas Leppänen
Luotettavuusinsinööri Sami Häkkinen

Tässä opinnäytetyössä toteutettiin momenttisäädin nimeämättömän kohdeyrityksen sähkömoottorin ajopaikalle. Aiemmin moottorin momenttia on säädetty ajopaikassa invertterin omalla virtaan perustuvalla säätimellä. Sillä pystytään moottorin momenttia pitämään vakaana minuuttien aikavälillä, mutta tätä pidemmän aikavälin yli momentti poikkeaa sen asetusarvostaan.

Momenttisäädin kehitettiin PLC:llä, joka on liitetty invertteriin ja momenttianturiin. Momenttisäätö toteutettiin PID-säätimellä, jolla säädetään invertterin momentin asetusarvoa moottorista mitatun momentin perusteella. PLC:n koodin muutokset tehtiin väliaikaiseen testi-PLC:hen, jotta koodimuutosten toiminnan testaus häiritsisi ajopaikan käyttöä mahdollisimman vähän. Lisäksi automaatiojärjestelmän käyttöliittymää muokattiin siten, että näppäimistöllä voi syöttää moottorin momentin asetusarvon.

Opinnäytetyön jälkeen on vielä paljon kehitettävää, kuten momenttisäätimien virittämistä ja PID-säätimen eri rakennemuokkausten kokeilua.

Momentin asetusarvon syöttö automaattiajosekvenssiin käyttöliittymästä ja automaatiojärjestelmän dokumentointi ovat työn tavoitteita, joita ei ehditty toteuttaa ja joita on tarkoitus toteuttaa työn päätyttyä.

Avainsanat: Säätötekniikka, PID-säädin

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Kim Henriksson
Title: Implementation of Torque Feedback Control for a Motor
Number of Pages: 37 pages
Date: 23 May 2025

Degree: Bachelor of Engineering
Degree Programme: Electrical and Automation Engineering
Professional Major: Automation Engineering
Supervisors: Tuomas Leppänen, Lecturer
Sami Häkkinen, Reliability Engineer

The objective of this thesis was the implementation of torque feedback control for a motor operating site for an unnamed commissioning company. Previously the torque control of the motor had been implemented using the inverter's own current-based control. This type of control is able to keep the torque of the motor stable over short time intervals, but eventually the torque will drift away from its setpoint.

The torque controller was implemented in a PLC that is connected to the inverter and the torque sensor. The torque control was implemented using a PID-controller, which controls the torque setpoint of the inverter based on the measured torque. The changes made to the PLC code were implemented on a temporary test PLC, so that functionality testing would minimally disrupt the usage of the motor operating site. In addition to this, the UI of the automation system was modified so that the torque setpoint can be entered via keyboard.

Further developments to be carried out after this thesis are, for example better tuning of the torque controller and potentially testing of PID-controller modifications that could be better suited for the task at hand. The possibility to input torque setpoints to the automated drive sequence from the UI of the automation system and the documentation of the system were originally part of the objective of this thesis but they were not implemented. These objectives will be implemented after the thesis has been completed.

Keywords: Control engineering, PID-controller

Sisällys

Lyhenteet

1	Johdanto	1
2	Säätötekniikan perusteoria	3
2.1	Säätötekniikan määritelmä	3
2.2	Ohjausjärjestelmän toiminnalliset yksiköt	3
2.3	Ohjausjärjestelmän suureet	6
2.4	Lähtösuureen perushallintastrategiat	8
2.5	Ohjausjärjestelmän suunnitteluprosessi	12
2.6	Ohjausjärjestelmän suorituskykyvaatimukset	12
3	PID-säädin	16
3.1	PID-säätimen peruseriaatteet	16
3.2	PID-säätimen eri rakenteet	19
4	Opinnäytetyön tavoitteet ja vaatimukset	23
4.1	PLC:n toiminnan vaatimukset	23
4.2	Automaatiojärjestelmän käyttöliittymän ja dokumentaation vaatimukset	24
5	Nykyisen automaatiojärjestelmän kuvaus	26
5.1	Automaatiojärjestelmän rakenne	26
5.2	Automaatiojärjestelmän laitteisto	27
5.3	Automaatiojärjestelmän viestintäprotokollat	27
5.4	Automaatiojärjestelmän ohjelmat	28
6	Moottorin ajopaikan automaatiojärjestelmän muutokset	31
6.1	Automaatiojärjestelmän laitteistojen muutokset	31
6.2	Automaatiojärjestelmän ohjelmistojen muutokset	32
7	Yhteenveto	36
	Lähteet	37

Lyhenteet

ADS:	<i>Automation Device Specification.</i> TwinCAT-laitteiden käyttämä kommunikointiprotokolla.
AmsNetID:	<i>Automation message specification Network IDentifier.</i> ADS-protokollan käyttävien laitteiden osoite.
BIBO:	<i>Bounded-Input Bounded-Output.</i> Järjestelmän stabiili- teetin määritelmä, jossa järjestelmän ulostulo pysyy määritetyissä rajoissa, jos myös sen sisääntulo pysyy määritetyissä rajoissa.
CANopen:	<i>Controller Area Network open.</i> CAN-kenttäväylään pohjautuva protokollapino.
EtherCAT:	<i>Ethernet for Control Automation Technology.</i> Beckhoffin kehittämä kenttäväylä.
IO:	<i>Input/Output.</i> Tiedonsiirtoon kykenevä komponentti.
MIMO:	<i>Multiple-Input and Multiple-Output.</i> Järjestelmä, jolla on monta sisääntuloa ja ulostuloa.
MISO:	<i>Multiple-Input and Single-Output.</i> Järjestelmä, jolla on monta sisääntuloa ja yksi ulostulo.
MV:	<i>Manipulated Variable.</i> Prosessin tulosuure eli toimi- suure.
OP:	<i>OutPut.</i> Kontrollerin ulostulo eli ohjaussuure.
PC:	<i>Personal Computer.</i> Yleiskäyttöön tarkoitettu tietokone.

PID:	<i>Proportional–Integral–Derivative</i> . Säädintyyppi, joka säätää prosessia sen verrannollisuuden, integraalin ja derivaatan perusteella.
PLC:	<i>Programmable logic controller</i> . Automaatiojärjestelmissä käytetty teollisuuden tietokone.
POU:	<i>Program Organization Unit</i> . IEC 61131-3 -standardin määrittämä ohjelmaobjekti.
PROFIBUS DP:	<i>PROcess Field BUS Decentralized Peripherals</i> . PROFIBUS & PROFINET Internationalin ylläpitämä PROFIBUS kenttäväylän versio.
PV:	<i>Process Variable</i> . Prosessin lähtösuure eli prosessisuure.
SIMO:	<i>Single-Input and Multiple-Output</i> . Järjestelmä, jolla on yksi sisääntulo ja monta ulostuloa.
SISO:	<i>Single-Input and Single-Output</i> . Järjestelmä, jolla on yksi sisääntulo ja yksi ulostulo.
SP:	<i>SetPoint</i> . Ohjausjärjestelmän kontrollerin asetusarvo.
ST:	<i>Structured Text</i> . Yksi IEC 61131-3 -standardin kielistä.
TwinCAT:	<i>The windows Control and Automation Technology</i> . Beckhoffin kehittämä automaatiojärjestelmien ajo- ja ohjelmointiympäristö.
VB.NET:	<i>Visual Basic .NET</i> . Microsoftin kehittämä olio-ohjelmointiparadigmaan kuuluva ohjelmointikieli.

1 Johdanto

Säätötekniikassa pyritään pitämään jokin prosessi jossain halutussa tilassa. Säätötekniikan perustana on negatiivinen takaisinkytkentä. Negatiivisella takaisinkytkennällä tarkoitetaan, että prosessin lähtösuuretta mitataan ja sen arvo vähennetään prosessille annetusta asetusarvosta. Tällä tavoin pystytään prosessin pysyvä poikkeama lopulta poistamaan. Teollisuuden käytetyin negatiivista takaisinkytkentää hyödyntävä kontrolleri on PID-säädin. PID-säädintä voi helposti virittää nyrkkisäännöillä ja se soveltuu kaikkienkokoisten prosessien säätämiseen.

Opinnäytetyön tavoitteena oli toteuttaa momenttisäädin nimeämättömän kohdeyrityksen sähkömoottorin ajopaikalle. Säätimen kehittämisen lisäksi tavoitteena oli myös muokata automaatiojärjestelmän käyttöliittymään ja dokumentoida luotua järjestelmää.

Opinnäytetyö hyödyttää kohdeyritystä antamalla heille mahdollisuuden pitää moottorin momentti vakaana. Aiemmin momenttia pystyttiin pitämään vakaana vain minuuttien aikavälillä, joten sitä piti säätää toisinaan käsin sen korjaukseksi. Automaatiojärjestelmän käyttöliittymän muutokset helpottavat moottorin ajopaikan ohjausta, kun taas sen dokumentaatio selventäisi järjestelmän rakennetta.

Moottorin momentin säätö toteutettiin PID-säätimellä, joka ohjelmoitiin väliaikaiseen testi-PLC:hen. PLC:n koodin muutokset tehtiin väliaikaiseen testi-PLC:hen, jotta muutosten tekeminen ja testaaminen haittaisi moottorin ajopaikan käyttöä mahdollisimman vähän. Automaatiojärjestelmän käyttöliittymä muokattiin VB.NET-ohjelmointikielellä. Käyttöliittymään lisättiin momentin syöttökenttä. Käyttöliittymän momenttiskaalausta muutettiin myös prosentin sadasosista Newtonmetreihin.

Opinnäytetyössä kerrotaan ensin säätötekniikan perusteoriasta. Tämän jälkeen esitellään opinnäytetyössä hyödynnetyn PID-säätimen perusteista. Opinnäytetyön teoriaosuuden jälkeen kuvataan työn tavoitteet ja vaatimukset sekä nykyinen automaatiojärjestelmä. Lopuksi selvitetään automaatiojärjestelmään tehdyt muutokset.

2 Säätötekniikan perusteoria

2.1 Säätötekniikan määritelmä

Tässä luvussa 2.1 kerrotaan lyhyesti säätötekniikan määritelmästä.

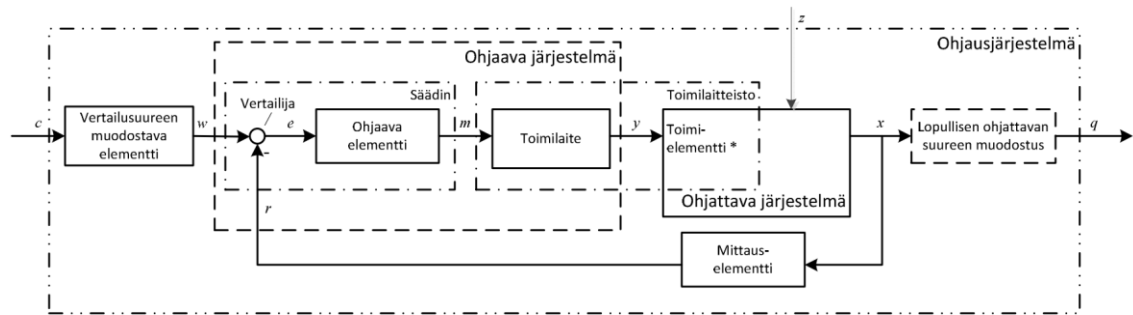
Säätötekniikka on monitieteinen tekniikan ala, jolla pyritään ymmärtämään järjestelmien toimintaa ja hallitsemaan niiden käytöstä. Säätötekniikassa tärkeää on järjestelmien matemaattinen mallintaminen. Mallien analysoinnin perusteella ohjausjärjestelmät suunnitellaan ja kehitetään. Ohjausjärjestelmät luodaan liittämällä yhteen niiden eri komponentteja sellaisella tavalla, että haluttu toiminto saavutetaan noudattamalla niille määritettyjä suorituskriteerejä. (1, s. 2–3.)

Säätötekniikan perustana on negatiivinen takaisinkytkentä, jossa prosessin lähtösuuretta mitataan ja vähennetään halutusta asetusarvosta. Tämän perusteella järjestelmän säätimen sisääntulo pienenee, mitä lähempänä prosessi on haluttua arvoa. Näin pystytään vähentämään prosessin korjaustoimia ja lopulta noudattaa ja pysyä halutussa asetusarvossa (1, s. 4).

2.2 Ohjausjärjestelmän toiminnalliset yksiköt

Tässä luvussa 2.2 määritellään ohjausjärjestelmän yleiset toiminnalliset yksiköt.

Ohjausjärjestelmän tyypilliset komponentit ja muuttujat ovat esitetty kuvan 1 toimintakaaviossa, joka on otettu standardista SFS-IEC 60050-351. Standardissa ohjausjärjestelmä on jaettu yksityiskohtaisemmin kuin tämän opinnäytetyön kaavioissa.



Kuva 1. Ohjausjärjestelmä ja sen sisältämät komponentit (2, s. 34).

Ohjausjärjestelmä

Ohjausjärjestelmä (*control system*) on koko mallinnetun järjestelmän rajaus, joka sisältää kontrollerin, prosessin, toimielimen ja mahdollisesti mittaelimen (2, s. 145).

Ohjaava elementti

Ohjaava elementti (*controlling element*) on se ohjausjärjestelmän elementti, joka sisältää jonkinlaisen algoritmin prosessin lähtösuureen hallitsemiseksi (2, s. 142).

Kontrolleri

Kontrolleri (*controller*) koostuu ohjaavasta elementistä ja siihen mahdollisesti liitetystä takaisinkytkennästä. Kun ohjausjärjestelmä on avoimessa silmukassa eli ilman takaisinkytkentää, kontrolleria kutsutaan ohjaimeksi (*open-loop controller*). Sen sijaan, jos ohjausjärjestelmä on suljetussa silmukassa eli silmukkaan on liitetty takaisinkytkentä, kontrolleria kutsutaan säätimeksi (*closed-loop controller*). (3; 4, s. 22 & 28.)

Toimilaitteisto

Toimilaitteisto (*final controlling equipment*) on toimilaitteen ja -elimen sisältävä toiminnallinen yksikkö (2, s. 151).

Toimilaite

Toimilaite (*actuator*) on se laite, joka muuttaa ohjaavan elementin lähtösuureen toimielimen tarvitsemaan muotoon prosessin vaikuttamiseksi. Toimilaitteiston toimilaite toimii ohjausjärjestelmän silmukan tehon vahvistajana. (2, s. 147; 5, s. 16.)

Toimielin

Toimielin (*final controlling element*) on se osa toimilaitteistosta, joka sisällytetään ohjattavaan järjestelmään eli prosessiin. Se manipuloi prosessin massa- tai energiavirtausta muuttaakseen prosessisuureen haluttuun arvoon. (2, s. 149.)

Ohjaava järjestelmä

Ohjaava järjestelmä (*controlling system*) on kontrollerista ja toimilaitteesta koostuva toiminnallinen yksikkö (2, s. 137).

Prosessi

Prosessi tai ohjattava järjestelmä (*controlled system*) on se toiminnallinen yksikkö, jonka lähtösuuretta hallitaan ohjausjärjestelmän kontrollerilla (2, s. 139).

Mittauselin

Mittauselin (*measuring element*) on ohjausjärjestelmän osa, jolla mitataan esimerkiksi prosessin lähtösuuretta tai häiriöitä (2, s. 144; 4, s. 24).

Vertailusuureen muodostava elementti

Vertailusuureen muodostava elementti (*reference-variable generating element*) muodostaa tulosuureen säätimen erosuureen laskentapisteelle jonkin funktion perusteella (2, s. 152).

2.3 Ohjausjärjestelmän suureet

Tässä luvussa 2.3 määritellään ohjausjärjestelmän yleisesti käyttämät suureet.

Opinnäytetyön kaavoissa ja kaavioissa käytetyt muuttujat eroavat SFS-IEC 60050-351 -ohjaustekniikan sanastossa käytetyistä muuttujista. Standardin muuttujat ovat merkitty "(IEC)" lyhenteellä epäselvyyksien välttämiseksi. Ne muuttujat mitkä eivät ole merkitty "(IEC)" lyhenteellä ovat opinnäytetyössä käytössä.

Prosessisuure

Prosessisuure y (*process variable, PV*) on ohjattavan järjestelmän eli prosessin lähtösuure. Tätä suuretta hallitaan ohjausjärjestelmän kontrollerilla. (2, s. 206; 6, s. 2; 4, s. 20 & 31.)

Prosessisuure y voidaan myös jakaa ohjattavaksi suureksi x (IEC) (*controlled variable*) ja lopulliseen ohjattavaksi suureksi q (IEC) (*final controlled variable*) (2, s. 120 & 133).

Asetusarvo

Asetusarvo r (*setpoint, SP*) määrää prosessin halutun lähtösuureen y arvon. Jos kontrollerin asetuservoa muutetaan harvoin, säädintä voidaan kutsua regulaattoriksi, kun taas jos kontrollerin asetuservoa muutetaan usein, kontrolleria voidaan kutsua joko seurantasäätimeksi tai servosäätimeksi, jos kyseessä on mekaanisen laitteen säädöstä. (5, s. 9; 6, s. 2.)

Säädössä asetusarvo r voidaan jakaa vertailusuureksi w (IEC) (*reference variable*) ja ohjesuureksi c (IEC) (*command variable*). Tällöin vertailusuureen muodostava elementti muodostaa vertailusuureen w (IEC) ja ohjesuureen c (IEC) jonkin funktion perusteella. (2, s. 121, 131 & 152.)

Erosuure

Erosuure e (*control difference variable*, vanhentunut: *error variable*) on asetusarvon r ja mittaussuureen y_{mit} ero. Erosuure r toimii säätimen tulosuureena. (5, s. 13; 2, s. 124.)

Mittaussuure

Mittaussuure y_{mit} tai takaisinkytkentäsuure r (IEC) (*feedback variable*) on prosessin lähtösuureen mitattu arvo (4, s. 31; 2, s. 123).

Ohjaussuure

Ohjaussuure u_1 tai ohjauksen lähtösuure m (IEC) (*controller output variable*, *OP*) on se suure, jolla kontrolleri ohjaa toimilaitetta (4, s. 31; 2, s. 126; 6, s. 2).

Toimisuure

Toimisuure u_2 tai y (IEC) (*manipulated variable*, *MV*) on ohjattavan järjestelmän eli prosessin tulosuure. Tällä suureella vaikutetaan suoraan prosessisuureeseen. (4, s. 31; 2, s. 127; 6, s. 2.)

Häiriösuure

Häiriösuure v tai z (IEC) (*disturbance variable*) on jokin ei-toivottu ja ennakoimaton ohjausjärjestelmän ulkopuolelta saapuva tulosuure. Häiriöt voivat esimerkiksi olla mittauskohinaa tai kuormitushäiriöitä. Häiriöitä voidaan mitata niiden kompensointia varten, jolloin kontrolleriin syötetään mitattu häiriösuure v_{mit} . (4, s. 31; 2, s. 129; 5, s. 13.)

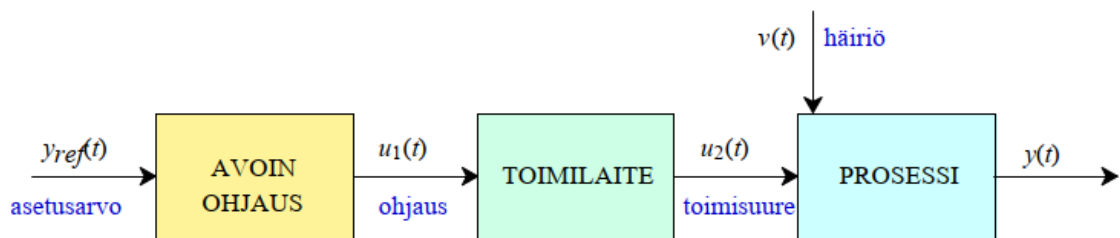
2.4 Lähtösuureen perushallintastrategiat

Tässä luvussa 2.4 lähtösuureen perushallintastrategian kolme tavanomaista muotoa esitetään. Tämän jälkeen luvussa käsitellään millä tavoin eri perushallintastrategioita voi yhdistellä. Lopuksi luvussa kerrotaan monimuuttujajärjestelmistä.

Prosessin hallintastrategiat voidaan tavanomaisesti jakaa kolmeen eri muotoon: ohjaus, kompensointi ja säätö (4, s. 20).

Lähtösuureen ohjaus

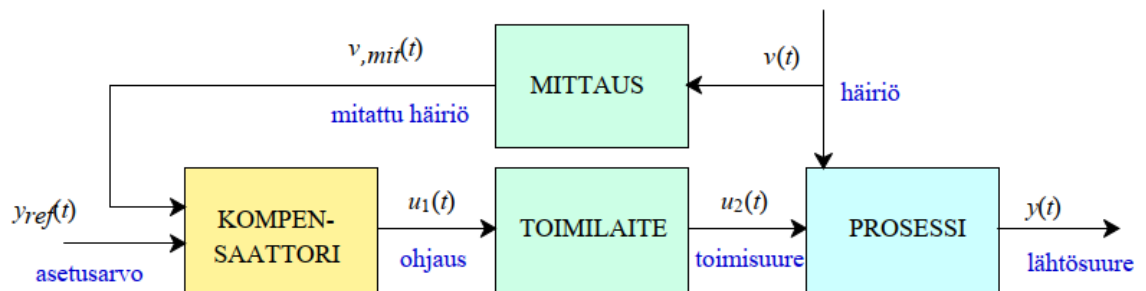
Lähtösuureen ohjauksessa tai lähtösuureen avoimessa ohjauksessa hallittavaa lähtösuuretta ei mitata ja käytetään kontrollerin tulosuurena. Ohjaussilmukka on avoin ja sen ainoa tulosuure on asetusarvo. Lähtösuureen ohjauksessa häiriöitä ei pystytä mitenkään kompensoimaan, joten niiden vaikutukset kumuloistuvat ajan myötä. Lähtösuureen ohjauspiirin lohkokaavio on esitetty kuvassa 2. (4, s. 22; 9, s. 7.)



Kuva 2. Lähtösuureen avoimen ohjauksen lohkokaavioesitys (4, s. 22).

Lähtösuureen kompensointi

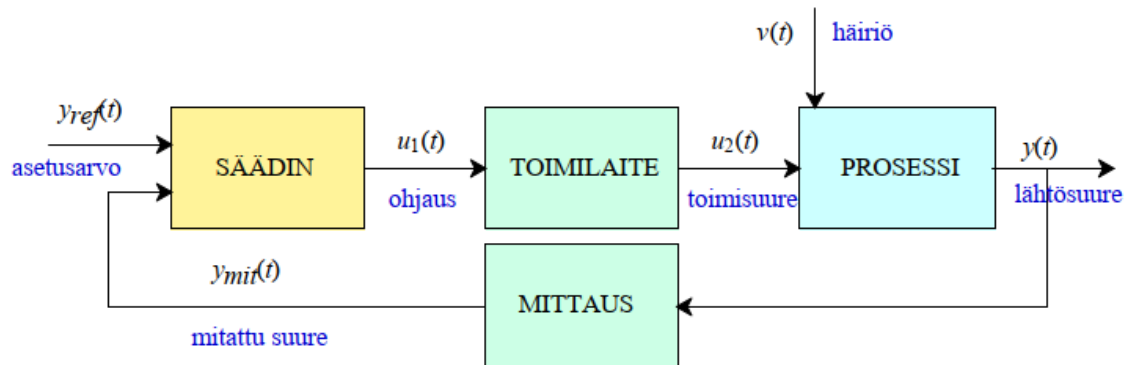
Lähtösuureen kompensoinnissa tai myötäkytkennässä pyritään esimerkiksi minimoimaan järjestelmän häiriöiden vaikutusta prosessiin. Häiriöitä mitataan ja mitatut häiriösuureet syötetään kompensoittorille tulosuureina asetusarvon lisäksi. Koska lähtösuureen kompensoittori on luonteeltaan ennakoiva, järjestelmän malli on tunnettava hyvin, jotta pystyttäisiin ennakoimaan mitä vaikutuksia häiriöillä on järjestelmään. Lähtösuureen kompensoittoriin lohkokaavio on esitetty kuvassa 3. (4, s. 24.)



Kuva 3. Lähtösuureen kompensoinnin lohkokaaavioesitys (4, s. 24).

Lähtösuureen säätö

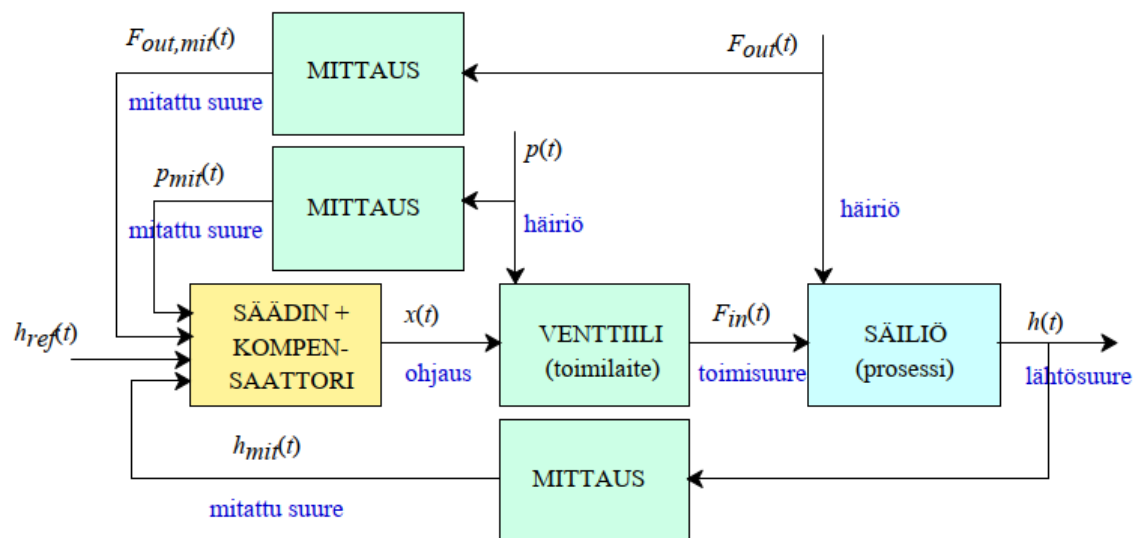
Lähtösuureen säädössä tai takaisinkytkennässä järjestelmän lähtösuuretta mitataan ja syötetään säätimelle tulosuureena asetusarvon lisäksi. Mitattu lähtösuure vähennetään asetusarvosta, jolloin saadaan erosuure, johon säädin reagoi. Koska säädin pystyy reagoimaan lähtösuureeseen, se on tarkempi ja sietää paremmin häiriöitä. Tällöin myös järjestelmän mallin tarkkaa tietämystä ei tarvita. Lähtösuureen säätöpiirin lohkokaavio on esitetty kuvassa 4. (4, s. 28; 9, s. 8.)



Kuva 4. Lähtösuureen säätimen lohkokaaavioesitys (4, s. 28).

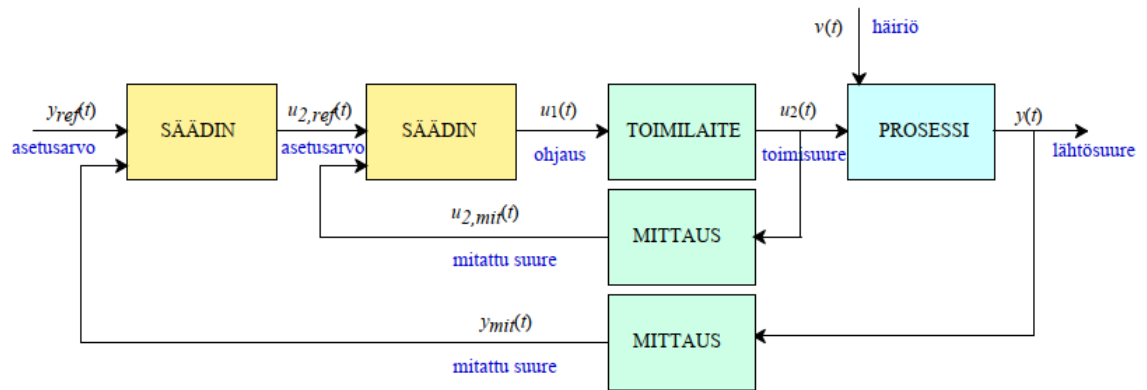
Edellä mainittujen prosessin perushallintastrategioita voi vapaasti yhdistellä tarpeen mukaisesti. Niitä voi yhdistellä samassa silmukkahierarkiassa eli rinnakkain tai eri silmukkahierarkiassa eli sisäkkäin. Hallintastrategioita voi myös yhdistellä sekvenssissä eli muuttamalla käytettyä strategiaa tilanteen mukaan. (4, s. 20.)

Esimerkki rinnakkaisesta hallintastrategian yhdistelmästä on kuvassa 5. Siinä säätimeen on lisätty toimilaitteen ja prosessin häiriöiden myötäkytkennät. (4, s. 20.)



Kuva 5. Rinnakkaisen hallintastrategian lohkokaaavioesitys (4, s. 20).

Esimerkki sisäkkäisestä hallintastrategiasta eli kaskadisäätimestä on kuvassa 6. Kaskadisäädön hyödyntämiseksi on sisäisen silmukan toiminta oltava huomattavasti nopeampi kuin ulomman silmukan. Sisäisen silmukan säädintä käytetään säätämään jotain toimitusuurta, jolla on suora vaikutus järjestelmän lähtösuureeseen. (4, s. 20.)



Kuva 6. Sisäkkäisen hallintastrategian lohkokaaavioesitys (4, s. 20).

Sekventiaalinen hallintastrategian muutos voi olla esimerkiksi valikoiva säätö, jossa toimilaitte, jota säädin ohjaa vaihtelee järjestelmän tilan perusteella. Se voi myös olla esimerkiksi vastakkainen tapaus eli kaksipuoleinen säätö, jossa säädin, joka säätää toimilaitetta vaihtelee järjestelmän tilan perusteella. (5, s. 41–42.)

Tähän mennessä on esitetty vain yhden muuttujan järjestelmiä eli SISO (*Single-Input and Single-Output*) järjestelmiä, mutta samoja hallintastrategioita voidaan myös käyttää monen muuttujan järjestelmissä. Monimuuttujakontrolleri voi olla tyyppiä MIMO (*Multiple-Input and Multiple-Output*), tyyppiä MISO (*Multiple-Input and Single-Output*) tai tyyppiä SIMO (*Single-Input and Multiple-Input*). (1, s. 4–5; 4, s. 62)

MIMO-järjestelmässä kontrollerilla on monta tulosuurta ja monta lähtösuurta. Kun halutaan säätää jotain kompleksista järjestelmää, jossa prosessin eri suureet vaikuttavat vahvasti toisiinsa, on järkevä käyttää yhtä MIMO-säädintä sitä varten, jossa prosessin dynamiikka on mallinnettu. (1, s. 4–5.)

MISO-järjestelmässä kontrollerilla on monta tulosuuretta mutta vain yksi lähtösuure. Tämä voi olla esimerkiksi lähtösuureen suhdesäätö, jos molemmat mitatut lähtösuureet syötetään kontrollerille tulosuureena asetusarvon mukana. Jos mitattujen lähtösuureiden suhde lasketaan ja se vähennetään asetusarvosta ennen kuin signaali syötetään kontrollerille tulosuureena, kyseessä olisi SISO-järjestelmä. (5, s. 39.)

SIMO-järjestelmässä kontrollerilla on yksi tulosuure ja monta lähtösuuretta. Tämä voi olla esimerkiksi tapaus, jossa kontrollerilla on useampi toimilaitte ja jossa se lähettää jokaiselle toimilaitteelle omansa ohjaussuureen. Jos kyseinen kontrolleri sen sijaan lähettäisi yhden ohjaussuureen usealle toimilaitteelle, kyseessä olisi SISO-järjestelmä. (5, s. 41.)

2.5 Ohjausjärjestelmän suunnitteluprosessi

Tässä luvussa 2.5 käsitellään lyhyesti ohjausjärjestelmän yleisestä suunnittelu-prosessista sen alusta loppuun.

Ohjausjärjestelmän suunnitteluprosessi alkaa vaatimusten määrittelyllä, jos niitä ei ole annettu jo ennen prosessin alkua. Silloin ohjausjärjestelmän tavoitteet muutetaan riittävän tarkoiksi spesifikaatioiksi, jotta valmis järjestelmä voidaan kokeellisesti todeta niitä täyttävän. Kun vaatimukset on määritelty, valitaan järjestelmälle sen tarvittavat kenttälaitteet kuten mm. mitta- ja toimilaitteet, jonka jälkeen ohjausjärjestelmä mallinnetaan matemaattisesti. Mallinnuksen perusteella valitaan järjestelmälle sopiva kontrolleri, jonka parametreja viritetään sopiviksi. Jos on tarvetta, prosessia iteroidaan sopivasta kohdasta, kunnes lopputuloksena on joko alkuperäiset tai muutettujen vaatimusten täyttävä ohjausjärjestelmä. (1, s. 20–21.)

2.6 Ohjausjärjestelmän suorituskykyvaatimukset

Tässä luvussa 2.6 esitetään joitakin yleisiä ohjausjärjestelmän suorituskykyvaatimuksia. Ohjausjärjestelmän suorituskykyvaatimukset esitellään ainoastaan laadullisesti, mainitsematta mistään kvantitatiivisista suorituskykyvaatimuksista.

Ohjausjärjestelmän suorituskykyvaatimukset voi olla annettu monella eri tavalla, kuten esimerkiksi aika- tai taajuusvasteena. Jotkut tavoitteet voivat jopa olla risiirittaisia ja edellyttää kompromisseja, kuten esimerkiksi kuormitushäiriön ja mittauskohinan vaimennuskyky. (10, s. 121.)

Joitakin yleisiä ohjausjärjestelmän suorituskykyvaatimuksia ovat:

- kuormitushäiriön vaimennuskyky,
- mittauskohinan vaimennuskyky,
- prosessimuutosten sietokyky,
- asetusravon seuranta,
- ohjauksen tasaisuus ja
- stabiliteetti (10, s. 122; 9, s. 10; 8, s. 223, 227 & 231).

Ohjaavan järjestelmän kuormitushäiriön vaimennuskyky

Ohjaavan järjestelmän kuormitushäiriön vaimennuskyvyllä tarkoitetaan sitä, miten hyvin ohjaava järjestelmä kykenee vaimentamaan kuormitushäiriöitä. Koska kuormitushäiriöt ovat yleensä matalataajuisia, ohjausjärjestelmän silmukkavahvistusta niillä taajuuksilla lisätään häiriöiden poistamiseksi. Silmukkavahvistus on ohjaavan järjestelmän ja ohjattavan prosessin yhteinen vahvistus. (10, s. 121–122 & 124; 1, s. 237.)

Ohjaavan järjestelmän mittauskohinan vaimennuskyky

Ohjaavan järjestelmän mittauskohinan vaimennuskyvyllä tarkoitetaan sitä, miten hyvin ohjaava järjestelmä kykenee vaimentamaan ohjausjärjestelmässä esiintyvää mittauskohinaa. Mittauskohina on yleensä korkeataajuuksista ja kuormitushäiriö matalataajuuksista. Tämän takia on yleensä mahdollista poistaa mittauskohinaa vaikuttamatta kuormitushäiriön vaimennuskykyyn. Mittauskohinaa voi vaimentaa joko suodattamalla tai vähentämällä silmukkavahvistusta korkeilla taajuuksilla. (10, s. 124; 1, s. 240.)

Ohjausjärjestelmän prosessimuutosten sietokyky

Ohjausjärjestelmän prosessimuutosten sietokyvyllä tarkoitetaan sitä, miten hyvin ohjausjärjestelmä kykenee pysymään vakaana, vaikka ohjattavan järjestelmän parametrit muuttuvat. Ohjattavan järjestelmän parametrit voivat muuttua ympäristön tai ikääntymisen vaikutuksesta tai voivat muuten erota mallinnetusta järjestelmästä. Ohjausjärjestelmän vankkuuden tai robustiuden määrittelee se, kuinka paljon ohjausjärjestelmä sietää sen parametrien muutosta ennen kuin siitä tulee epästabiili. (1, s. 233; 7, s. 254.)

Ohjaavan järjestelmän asetusarvon seuranta

Ohjaavan järjestelmän asetusarvon seurannalla tarkoitetaan sitä, miten hyvin ohjaava järjestelmä pystyy pitämään prosessisuureen sen halutussa arvossa. Sen määrittää esimerkiksi, miten nopeasti järjestelmä pystyy reagoimaan prosessi- tai asetusarvon muutokseen ja kuinka vähän prosessisuure poikkeaa sen halutusta arvosta järjestelmän tilapäisissä ja pysyvissä tiloissa. (10, s. 126–127.)

Ohjaavan järjestelmän ohjauksen tasaisuus

Ohjaavan järjestelmän ohjauksen tasaisuudella tarkoitetaan sitä, miten tasainen ohjaavan järjestelmän ulostulo eli toimitus on. Liiallinen toimitusvasteen värähtely kuluttaa sekä suuremman määrän energiaa että myös itse toimilaitetta. Toimitusvaste voi värähdellä joko ohjaussuureen värähtelyn myötä tai toimilaitteen vian seurauksena. Ohjaussuureen värähtely voi syntyä kontrollerin liian aggressiivisesta virstäytymisestä tai mittausvasteen värähtelyn takia. Värähtelyä aiheuttavan toimilaitteen vika voi olla esimerkiksi venttiilin liiallinen lepokitka. (8, s. 223 & 225–226.)

Ohjausjärjestelmän stabiliteetti

Ohjausjärjestelmän stabiliteetilla tarkoitetaan sitä, että ohjausjärjestelmän luontainen vaste ei kasva äärettömästi pakotetun vasteen vaikutuksesta. Ohjausjärjestelmän kokonaisvaste on pakotetun ja luontaisen vasteen summa. Jos luontainen vaste ei kasva mutta ei myöskään vaimene koskaan, kun aika lähestyy ääretöntä, kyseessä on marginaalisesti stabiili ohjausjärjestelmä. (9, s. 300.)

Yleensä ohjausjärjestelmien analysoinnissa käytetään BIBO (*Bounded-Input Bounded-Output*) stabiliteetin määritelmää, jossa ohjausjärjestelmän ulostulo pitää pysyä joidenkin raja-arvojen sisällä, jos myös sisääntulo pysyy sille määrättyjen raja-arvojen sisällä. (9, s. 300.)

3 PID-säädin

3.1 PID-säätimen peruseriaatteet

Tässä luvussa 3.1 kerrotaan PID-säätimen peruseriaatteista. PID-säätimestä kerrotaan ensin yleisesti, jonka jälkeen esitellään kolme osaa, josta PID-säädin koostuu. Lopuksi listataan yleisesti käytettyjä PID-säätimen osien yhdistelmiä.

PID-säädin on teollisuuden käytetyin säädintyyppi, jolla prosesseja säädetään. Se on helposti nyrkkisäännöillä viritettävä säädin, mutta se soveltuu myös ympäristöihin, joissa säätösilmukalla on useita häiriötekijöitä. Se koostuu kolmesta eri osasta: proportionaalisesta eli verrannollisesta (P), integroivasta (I) ja derivoivasta (D) osasta. (5, s. 67; 10, s. 59.)

PID-säätimen verrannollinen osa

Verrannollinen säätö suhteuttaa ohjaussignaalin erosuureen suuruuden perusteella. Ohjaussignaali voidaan laskea kaavasta 1. (8, PID s. 3.)

$$u(t) = K_p e(t) = K_p (y_{ref}(t) - y(t)) \quad (1)$$

$u(t)$ on ohjaussuure

$e(t)$ on erosuure

$y_{ref}(t)$ on asetusero

$y(t)$ on lähtösuure

t on aika

K_p on verrannollisuuskerroin

Jos käytetään ainoastaan P-säädintä, se ei kykene poistamaan prosessin pysyvää poikkeamaa. Siksi kaavaan 1 lisätään yleensä biasointitermi u_b , jota käytetään poikkeaman poistamiseksi. P-säätimen ohjaussignaali, johon on lisätty korjausermi u_b voidaan laskea kaavasta 2. (8, PID s. 4–5.)

$$u(t) = K_p e(t) + u_b \quad (2)$$

$u(t)$ on ohjaussuure

$e(t)$ on erosuure

t on aika

K_p on verrannollisuuskerroin

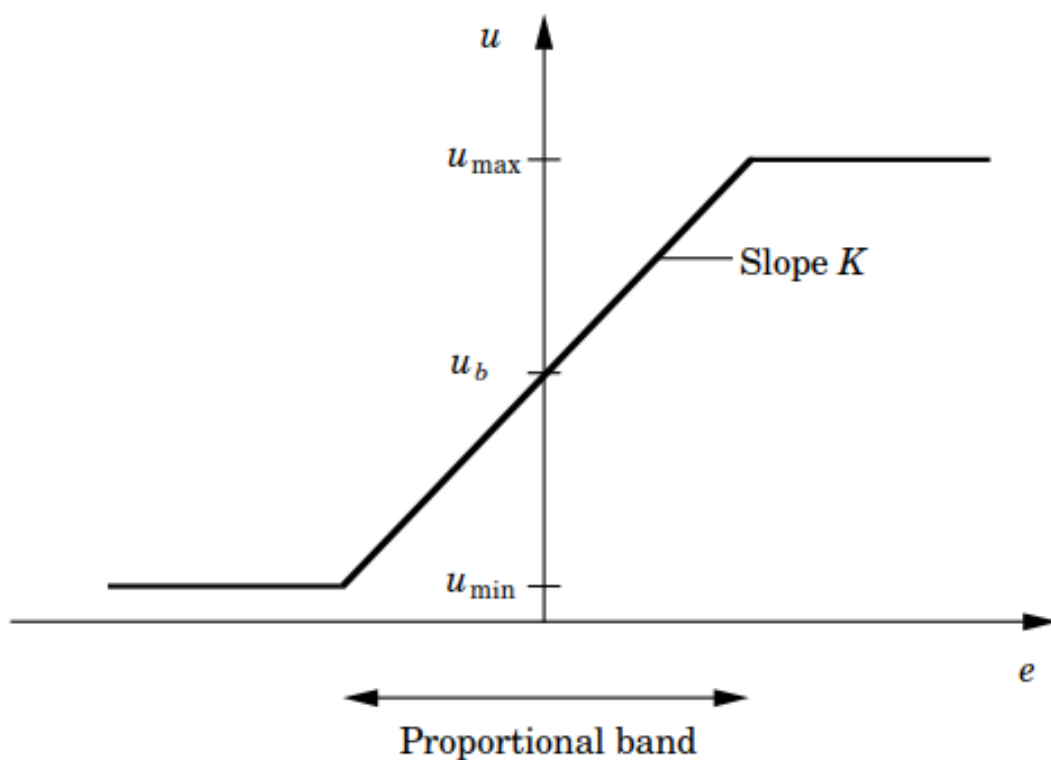
u_b on biasointitermi

Kaupallisissa P-säätimissä verrannollisuuskerroin K_p korvataan usein kertoalueella PB . Kertoalue on se alue, jossa P-säädin ei kyllästy. Se on esitetty kuvassa 7 ja se voidaan laskea kaavasta 3. (8, s. 5; 11, s. 20.)

$$PB = \frac{100}{K_p} \quad (3)$$

PB on kertoalue

K_p on verrannollisuuskerroin



Kuva 7. P-säätimen kertoalue (10, s. 62).

PID-säätimen integroiva osa

Integroiva säätö suhteuttaa ohjaussignaalin erosuureen integraalin perusteella. Tämän voi tulkita ajan suhteen siten, että integraalin kumuloitu arvo ottaa huomioon säätimen historialliset erosuureet. Integroiva säätö kykenee kokonaan poistamaan vakaan tilan erosuureen. Ohjaussignaali voidaan laskea kaavasta 4. (8, s. 5.)

$$u(t) = K_i \int_0^t e(\tau) d\tau \quad (4)$$

$u(t)$ on ohjaussuure

$e(\tau)$ on erosuure

t on aika

τ on integrointimuuttuja

K_i on integrointikerroin

PID-säätimen derivoiva säätö

Derivoiva säätö suhteuttaa ohjaussignaalin erosuureen derivaatan perusteella. Tämän voi tulkita ajan suhteen siten, että derivaatta ennustaa seuraavaa erosuureen arvoa eli tulevaisuutta. Ohjaussignaali voidaan laskea kaavasta 5. (8, s. 6.)

$$u(t) = K_d \frac{de(t)}{dt} \quad (5)$$

$u(t)$ on ohjaussuure

$e(t)$ on erosuure

t on aika

K_d on derivointikerroin

Yleisesti käytössä olevat P, I ja D osan kombinaatioita ovat:

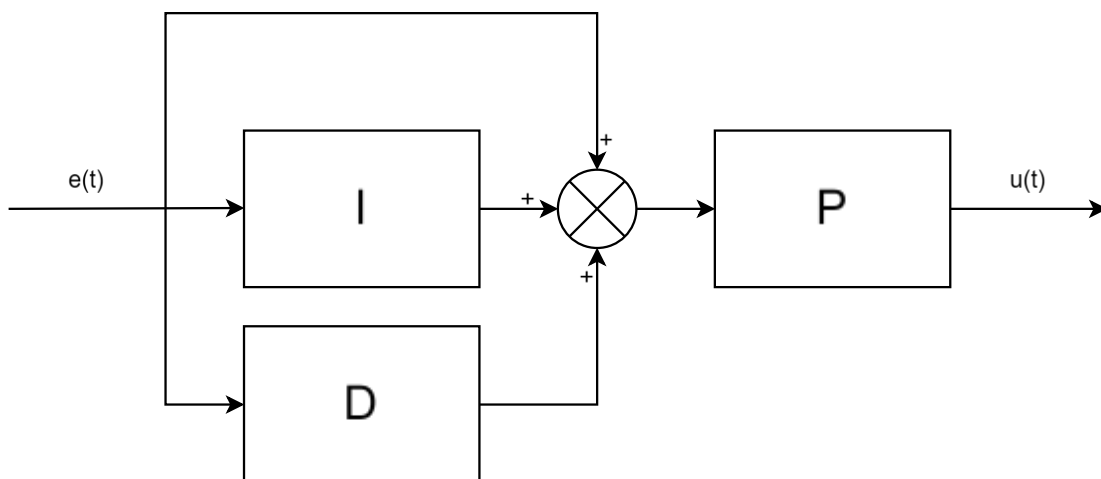
- P käytetään perussäätimenä,
- PI käytetään, kun halutaan eliminoida P-säätimen poikkeamaa,
- PID käytetään, kun halutaan eliminoida PI-säätimen epävakautta,
- PD käytetään kaskadisäädössä ja
- I käytetään kaskadisäätimen primäärisäätimenä (6, s. 16).

3.2 PID-säätimen eri rakenteet

Tässä luvussa 3.2 kerrotaan PID-säätimen eri rakenteista. PID-säädin voidaan jakaa kolmeen eri rakenteeseen, josta tässä luvussa kerrotaan. Nämä kolme eri rakennetta ovat ideaalirakenne, sarjarakenne ja rinnakkaisrakenne.

PID-säätimen ideaalimuotoinen rakenne

Ideaalirakenne on nykyisin käytetyin toteutustapa. Säätimen vahvistuskerroin vaikuttaa erosuureeseen sekä sen integraaliin että derivaattaan. Ideaalimuotoisen PID-säätimen lohkokaaavio näkyy kuvassa 8. (5, s. 90.)



Kuva 8. Ideaalirakenteinen PID-säätimen lohkokaaavioesitys (mukautettu 5, s. 90).

Ideaalirakenteisen PID-säätimen ohjaussignaali voidaan laskea kaavasta 6.

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] \quad (6)$$

$u(t)$ on ohjaussuure

$e(t)$ on erosuure

t on aika

τ on integrointimuuttuja

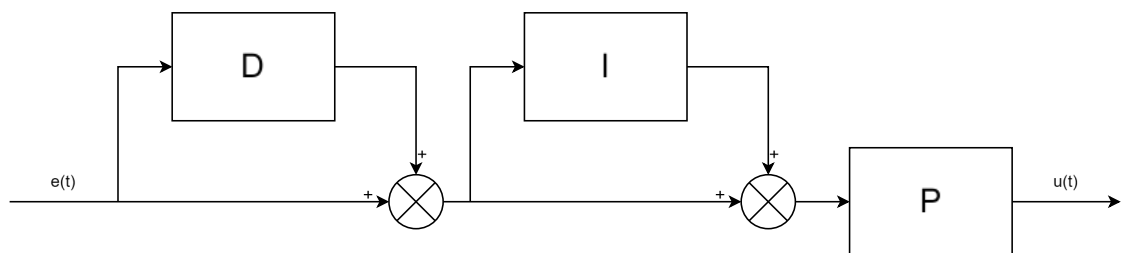
K_p on vahvistuskerroin

T_i on integrointiaika

T_d on derivointiaika

PID-säätimen sarjamuotoinen rakenne

Sarjamuotoisessa rakenteessa derivointiaika vaikuttaa integrointiosaan ja rakennetta kutsutaankin myös vuorovaikutteiseksi rakenteeksi. Sarjamuotoisen PID-säätimen lohkokaavio näkyy kuvassa 9. (5, s. 91.)



Kuva 9. Sarjarakenteisen PID-säätimen lohkokaavioesitys (mukautettu 5, s. 91).

Sarjarakenteisen PID-säätimen ohjaussignaali voidaan laskea kaavasta

7.

$$u(t) = K_p \left[e(t) \left(1 + \frac{T_d}{T_i} \right) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] \quad (7)$$

$u(t)$ on ohjaussuure

$e(T)$ on erosuure

t on aika

τ on integrointimuuttuja

K_p on vahvistuskerroin

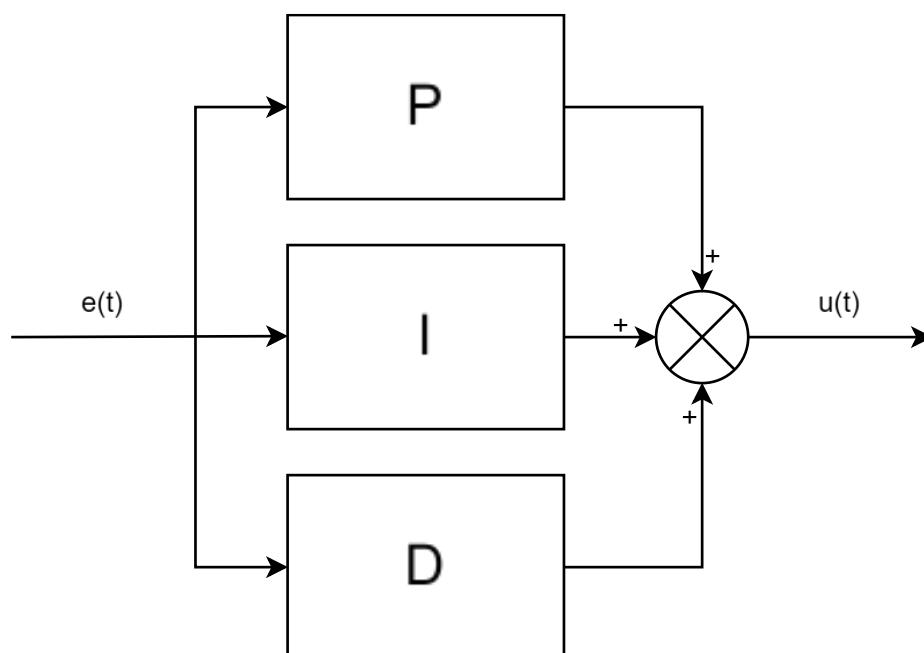
T_i on integrointiaika

T_d on derivointiaika

Kuten kaavasta voidaan havaita, jos derivoititermiä ei ole lainkaan, PID-säätimen sarjarakenne vastaa PID-säätimen ideaalirakennetta ja jos $\frac{T_d}{T_i}$ on hyvin pieni, rakenne lähestyy ideaalimuotoa. (5, s. 91.)

PID-säätimen rinnakkaismuotoinen rakenne

Rinnakkaismuodossa jokaisella haaralla on yleensä oma vahvistuskerroin. Tätä muotoa kutsutaan myös vuorovaikutuksettomaksi rakenteeksi, sillä säätimen eri osat eivät vaikuta toisiinsa. Rinnakkaismuotoisen PID-säätimen lohkokaavio näkyy kuvassa 10. (5, s. 91.)



Kuva 10. Rinnakkaisrakenteisen PID-säätimen lohkokaavioesitys (mukautettu 5, s. 90).

Rinnakkaisrakenteisen PID-säätimen ohjaussignaali voidaan laskea kaavasta 8.

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{de(t)}{dt} \quad (8)$$

$u(t)$ on ohjaussuure

$e(t)$ on erosuure

t on aika

τ on integrointimuuttuja

k_p on verrannollisuuskerroin

k_i on integrointikerroin

k_d on derivointikerroin

4 Opinnäytetyön tavoitteet ja vaatimukset

Tässä luvussa 4 opinnäytetyön kehittämisprosessin tavoitteet ja vaatimukset selvitetään. Aluksi luvussa kerrotaan hiukan yleisesti opinnäytetyön tavoitteista ja vaatimuksista. Tämän jälkeen PLC:n toiminnan vaatimuksista ja automaatiojärjestelmän käyttöliittymän ja dokumentaation vaatimuksista kerrotaan niiden alaluvuissa.

Opinnäytetyön kehittämisprosessissa pyrittiin modernisoimaan moottorin ajopaikan automaatiojärjestelmää. Siihen kuului sekä PLC:n koodin, että käyttöliittymän koodin muutoksia. Modernisoinnin avulla helpotetaan ja nopeutetaan moottorin ajopaikan operaattorin työtä.

Opinnäytetyön tärkein tavoite oli moottorin ajopaikan momenttisäätimen toteutus. Toisena tärkeänä tavoitteena oli moottorin ohjaukseen ja seurantaan tarkoitettun käyttöliittymän päivittäminen. Lopuksi oli tarkoitus dokumentoida koko automaatiojärjestelmä, kun kaikki tavoitteet on saatu toteutettua.

Opinnäytetyön vaatimuksia oli tarkoitus tarkentaa tarpeen mukaan työn edessä, eikä niille annettu tarkkoja kriteerejä. Tarkoituksena oli myös ajan salliessa lisätä toimintoja, funktioita ym. parannuksia automaatiojärjestelmän toimintaan mitä työn teon aikana tulee mieleen.

4.1 PLC:n toiminnan vaatimukset

Tässä luvussa 4.1 opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän PLC:n toiminnan vaatimukset selvitetään.

PLC:n toiminnan vaatimuksiin kuuluu:

- momenttisäädön toteutus moottorin ajopaikalle
- momenttisäädön yhteensopivuus momenttirampin kanssa
- koodin siistiminen ja organisointi
- muut tarvittavat koodinmuutokset käyttöliittymän vaatimusten toteuttamiseen.

Moottorin momenttisäätimen kriteerinä oli, että se pitää pystyä korjaamaan vaakan tilan virhe järkevässä ajassa ja se, että se ei saa itse aiheuttaa epästabiiliisuutta järjestelmään. Sen vaste asetusarvon muutokselle tulisi myös olla tarpeeksi maltillinen, jottei asetusarvoa ylitetä hirmuisesti.

Nykyisessä automaatiojärjestelmässä on käytössä momenttiramppifunktio, jolla voidaan asettaa tiettyjä momentin ramppiaikoja moottorille. Momenttisäätimen tulisi pystyä reagoimaan asetusarvon muutokseen tarpeeksi nopeasti ja tarkasti, jottei valitun rampin aika muutu oleellisesti.

Koska PLC:n koodi oli hyvin vaikeasti luettava ja sen eri komponentit eivät olleet organisoitu järkeviin kokonaisuuksiin, opinnäytetyön vaatimuksena oli myös koodin siistiminen. Siistimisellä pyritään helpottamaan tulevaisuuden muutoksien tekemistä ja saada näkyviksi turhaa tai huonosti kirjoitettua koodia, jotta korjaustoimenpiteitä voidaan sille suorittaa.

Käyttöliittymällä on myös useampi vaatimus, joita tulisi täyttää. Käyttöliittymä lähettää PLC:lle tehtävien toimenpiteiden käskyt, eikä itsestään aja mitään toimenpiteiden suorittamiseen tarvittavaa koodia. Tämä tarkoittaa, että PLC:n koodiin on tehtävä muutoksia, jotta käyttöliittymän vaatimukset saadaan toteutettua.

4.2 Automaatiojärjestelmän käyttöliittymän ja dokumentaation vaatimukset

Tässä luvussa 4.2 opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän käyttöliittymän ja dokumentaation vaatimukset selvitetään.

Käyttöliittymän toiminnan vaatimuksiin kuuluu:

- momenttiskaalauksen muutos Newtonmetreihin
- mahdollisuus syöttää momentti näppäimistöllä
- momentin lisäysmahdollisuus automaattiajosekvenssiin
- koodin siistiminen ja organisointi.

Nykyinen käyttöliittymän momenttiskaalaus on annettu prosentin sadasosissa (‰), jossa arvo 25000 ‰ vastaa 250 %, joka on invertterin maksimaalinen momentin asetusarvo suhteutettuna siihen parametroituun moottorin kilpiarvoon. Moottorin ajopaikan operaattorin on pakko käsin säätää momentti sen haluttuun Newtonmetriarvoon lukemalla momentin oloarvoa. Käyttöliittymän momenttiskaalauksen muuttaminen Newtonmetreiksi helpottaisi huomattavasti moottorin ajopaikan operaattorin työtä, koska silloin näytetty asetusarvo olisi samassa yksikössä kuin mitattu momentti.

Aikaisemmin ainoastaan ohjauskapuloista pystyi muuttamaan momentin asetusarvoa nupin kääntämisellä. Kun sen sijaan näppäimistöllä voisi syöttää käyttöliittymään momenttiarvo Newtonmetreissä, moottorin ajopaikan operaattori välttyisi nupin käsisäädöstä tarkan arvon saavuttamiseksi. Koska ohjauskapuloiden nupeilla on absoluuttiset asennot, ne ylikirjoittaisivat momentin syöttökentän arvon, jos niitä käytettäisiin yhtä aikaa kuin momentin syöttökenttää. Tämän ansiosta on jouduttava valitsemaan käyttöliittymästä jommankumman momentin syöttötavan ratkaisun toimiakseen.

Nykyisessä automaattiajosekvenssin luonnissa pystytään ainoastaan määrittämään sekvenssin osien pituudet ja moottorin päällä olon ja pyörimissuunnan. Moottorin nopeuden ja momentin asetusarvot luetaan tällöin moottorin ohjauskapulasta. Momentin lisäysmahdollisuus automaattiajosekvenssiin mahdollistaisi monimutkaisempien sekvenssien luonnin.

Dokumentoinnin vaatimukseen kuuluu automaatiojärjestelmän yleiskuvan laatiminen. Automaatiojärjestelmä piirustus/piirustukset tulisi tehdä kohdeyrityksen hyviä käytäntöjä noudattaen.

5 Nykyisen automaatiojärjestelmän kuvaus

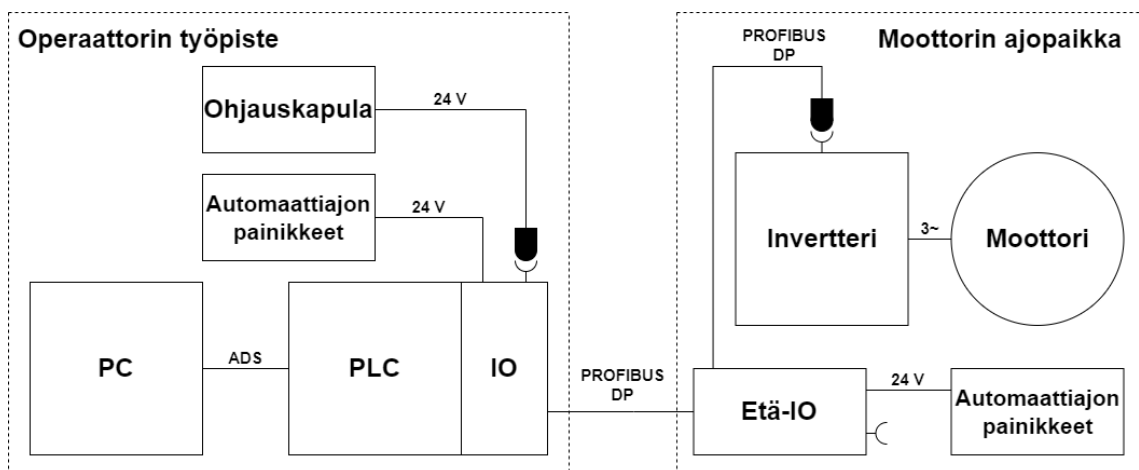
5.1 Automaatiojärjestelmän rakenne

Tässä luvussa 5.1 opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän rakenne käsitellään. Nykyisestä automaatiojärjestelmästä annetaan myös yleiskaavio.

Nykyinen automaatiojärjestelmä koostuu PLC:stä ja sen oheislaitteista, jolla ohjataan moottorin ajopaikkaa. Järjestelmään kuuluu anturit moottorin nopeuden ja momentin mittaukseen ja invertteri, jolla moottoria syötetään.

Moottorin ajopaikan invertterin tilaa ja joitakin asetuksia voidaan ohjata ja seurata käyttöliittymästä. Moottorin ajokäskyt, nopeuden ja momentin asetusarvot asetetaan moottorin ohjauskapulalla. Moottorin ajopaikalle on myös mahdollista ohjelmoida automaattiajosykli, jonka voi käynnistää ja pysäyttää ajopaikan läheisyydeltä tai operaattorin työpisteeltä.

Yleiskaavio automaatiojärjestelmän rakenteesta on kuvassa 11.



Kuva 11. Nykyisen automaatiojärjestelmän yleiskaavio.

5.2 Automaatiojärjestelmän laitteisto

Tässä luvussa 5.2 käsitellään opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän laitteistot. Automaatiojärjestelmän operaattorin työpisteellä sijaitsevat sekä moottorin ajopaikalla sijaitsevista laitteista kerrotaan.

Automaatiojärjestelmän PLC:nä toimii Beckhoffin CX9000-0001. PLC:ssä on suoraan liitettyjä IO-moduuleita, johon on liitetty operaattorin työpisteellä sijaitseva moottorin ohjauskapula ja automaattiajon ohjaavia painikkeita.

Automaattiajon painikkeet koostuvat käynnistys- ja pysäytyspainikkeesta sekä automaattiajon tilaa osoittavasta merkkivalosta. Niitä on sekä moottorin ajopaikan läheisyydessä että operaattorin työpisteellä.

Moottorin ohjauskapula koostuu nupeista, jolla muutetaan moottorin nopeuden ja momentin asetusarvoa sekä kytkimistä, jolla annetaan moottorille sen ajokäsky ja -suunta.

Moottorin ajopaikassa olevan sähkömoottorin akselissa on kiinni inkrementaalinen enkooderi, joka mittaa moottorin pyörimisnopeutta ja suuntaa. Enkooderi on suoraan liitetty moottoria syöttävään invertteriin nopeuden säätöä varten. Moottorin akselissa on myös kiinni momenttianturi, jolla momenttia mitataan.

Moottorin ajopaikan läheisyydessä sijaitsee PLC:n etä-IO, johon on liitetty automaattiajon painikkeet ja merkkivalo, potentiaalivapaat relelähdöt ja moottorin ohjauskapulan lähiliitännämahdollisuus.

5.3 Automaatiojärjestelmän viestintäprotokollat

Tässä luvussa 5.3 käsitellään opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän käyttämiä viestintäprotokollia.

PLC on yhdistetty etä-IO:hon ja invertteriin PROFIBUS DP -kenttäväylällä.

Moottorin ajopaikan ohjaava PLC on liitetty paikalliseen sisäverkkoon. Samassa sisäverkossa on myös PC liitettynä, jossa käyttöliittymä ajetaan. Käyttöliittymä kommunikoi PLC:n kanssa ADS protokollaa käyttäen.

5.4 Automaatiojärjestelmän ohjelmat

Tässä luvussa 5.4 esitellään opinnäytetyön kehittämisen kohteena olevan automaatiojärjestelmän käyttämät ohjelmat. Automaatiojärjestelmän ohjelmat koostuvat PLC:n ohjelmasta ja käyttöliittymästä. Tässä luvussa esitellään ensin PLC:n eri ohjelmaobjektit ja niiden toiminnot, jonka jälkeen esitetään automaatiojärjestelmän käyttöliittymän rakenteesta ja toiminnasta.

PLC:n ohjelma

PLC:n ohjelma on tehty TwinCAT 2 -ohjelmistoympäristöllä IEC 61131-3 -standardin ST-kieltä käyttäen. Ohjelma ajetaan PLC:ssä TwinCAT 2 -ajoympäristössä. PLC:n ohjelma on jaettu ohjelmaobjekteihin eli POUihin. POUihin kuuluu ohjelmat, funktiolohkot, ja funktiot.

PLC:n ohjelmassa on seuraavat ohjelma-POUt:

- kontaktorienhallintaohjelma,
- toimenpiteenkäsittelijäohjelma,
- sisääntulojen lukuohjelma ja
- lähtöjenkirjoitusohjelma.

Kaikki PLC:n ohjelma-POU:t ajetaan joka 20. millisekunti, paitsi toimenpiteenkäsittelijäohjelma, joka ajetaan joka 50. millisekunti. PLC:n funktiolohko-POU:t ja funktio-POU:t kutsutaan PLC:n ohjelma-POU:ista.

PLC:n kontaktorienhallintaohjelma hallitsee automaatiojärjestelmän kontaktorien, kuten esimerkiksi moottorin syöttökontaktoria vetämisestä ja päästämisestä.

PLC:n toimenpiteenkäsittelijäohjelma käsittelee automaatiojärjestelmän käyttöliittymästä saapuvat toimenpidepyynnöt, kuten esimerkiksi automaattiajosekvenssin luku.

PLC:n sisääntulojen lukuohjelma lukee esimerkiksi moottorin ohjauskapulan kytkimien ja nappien tilat. PLC:n lähtöjenkirjoitusohjelma kirjoittaa PLC:n ulostulojen tilat. PLC:n lähtöjenkirjoitusohjelma kirjoittaa esimerkiksi automaattimerkivaloja ohjaavien PLC:n ulostulojen tiloja.

PLC:n ohjelmassa on seuraavat funktiolohko-POUt:

- automaattiajon funktiolohko,
- automaattiajon laskurin funktiolohko ja
- signaalisuodatin funktiolohko.

PLC:n automaattiajon funktiolohkossa automaatiojärjestelmän automaattiajosekvenssi ajetaan. PLC:n automaattiajon laskurin funktiolohko laskee esimerkiksi automaattiajon käynnissäoloajan. PLC:n signaalisuodatin funktiolohko suodattaa moottorin ohjauskapulan nupeista luetut moottorin nopeuden ja momentin asetusarvot.

PLC:n ohjelmassa on seuraavat funktio-POUt:

- tiedontyyppien muunnosfunktiot,
- skaalausfunktiot ja
- momenttirampin funktio.

PLC:n tiedontyyppien muunnosfunktiot muuntavat erilaisia tietotyyppejä moottorin ajopaikan invertterin haluamaan muotoon. PLC:n skaalausfunktioissa moottorin ohjauskapulan nupeilta luetut nopeuden ja momentit skaalataan vastamaan prosentinsadasosia. PLC:n momenttirampin funktiossa moottorin ajopaikalle lähetetty momentin asetusarvo nostetaan askelittain haluttuun arvoon valitun ramppiajan mukaisesti.

Automaatiojärjestelmän käyttöliittymä

Automaatiojärjestelmän käyttöliittymä on tehty Visual Studiolla -ohjelmalla VB.NET-kielillä. Se ajetaan nykyisessä järjestelmässä vain yhdestä Windows-pohjaisesta PC:stä, johon otetaan etäyhteyttä, kun sitä halutaan käyttää.

Automaatiojärjestelmän käyttöliittymän visuaalinen rakenne koostuu paneelistä, jossa näkyy muun muassa invertterin tila, nopeus- ja momenttiohjeiden asetusarvot, moottorin ajosuunnat ja automaattiajon laskurien tietoja. Automaatiojärjestelmän käyttöliittymästä voidaan muuttaa muun muassa invertterin tilaa, momenttirampin aikaa, nopeus- tai momenttisäädön valintaa ja automaattiajon sekvenssiä.

6 Moottorin ajopaikan automaatiojärjestelmän muutokset

6.1 Automaatiojärjestelmän laitteistojen muutokset

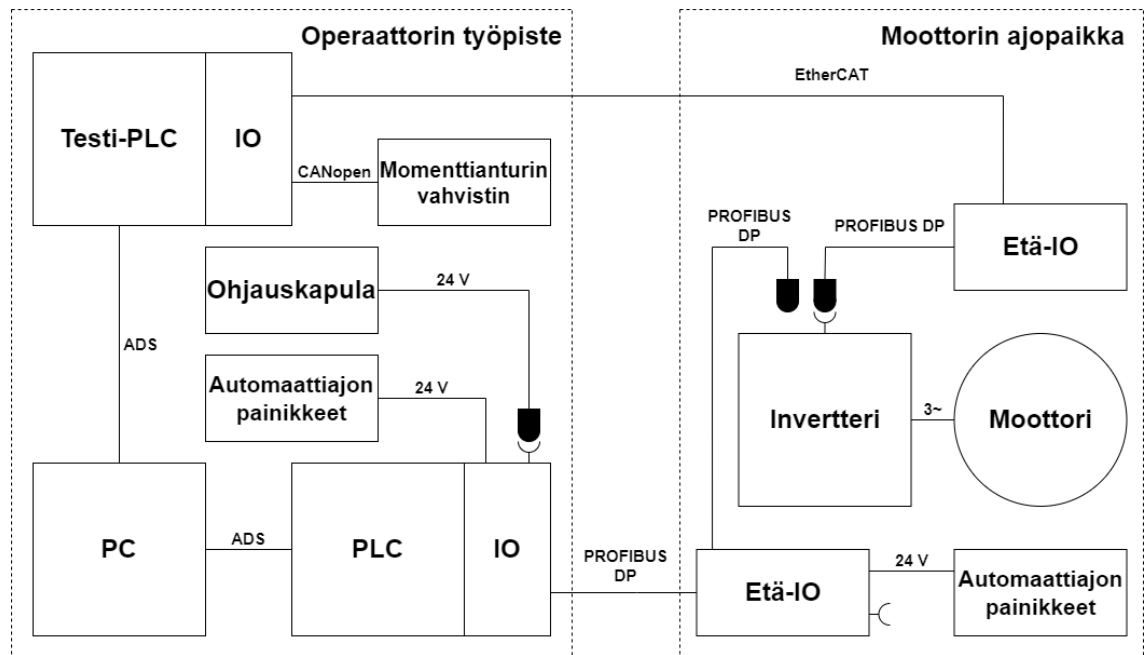
Tässä luvussa 6.1 kerrotaan opinnäytetyön kehittämisprosessin aikana tehtyjen automaatiojärjestelmän laitteistojen muutoksista. Automaatiojärjestelmään tehdyistä muutoksista annetaan myös yleiskaavio.

Automaatiojärjestelmän kehittämisen vuoksi otettiin käyttöön väliaikainen testi-PLC, jotteivat muutokset vaikuttaisi nykyisen järjestelmän käyttöön. Testi-PLC:n malli on CX9020. Se eroaa hiukan nykyisen järjestelmän PLC:stä.

Testi-PLC liitettiin invertteriin etä-IO-moduulilla, jota otettiin käyttöön muutosten testaamisen ajaksi. Väliaikainen etä-IO-moduuli liitettiin testi-PLC:hen EtherCAT-väylää käyttäen, jotta pystyttiin hyödyntämään olemassa olevia verkkokaapeleita. Väliaikaisessa etä-IO-moduulissa on myös PROFIBUS DP -väyläliitäntä, jota käytetään sen liittämiseen invertteriin. Kun testi-PLC:tä halutaan käyttää ohjaamaan moottorin ajopaikkaa, invertteri irrotetaan nykyisen automaatiojärjestelmän PROFIBUS PD -väylästä ja liitetään väliaikaisen etä-IO:n PROFIBUS DP -väylään.

Moottorin ajopaikan momenttianturin vahvistimeen lisättiin CANopen-väylämoduuli, jotta momenttiarvoa pystyttäisiin lähettämään testi-PLC:lle. Testi-PLC:hen siten lisättiin CANopen-väyläliitäntä, jotta se pystyisi lukemaan momenttianturin vahvistimen dataa.

Yleiskaavio automaatiojärjestelmän rakenteesta muutosten jälkeen on annettu kuvassa 12.



Kuva 12. Automaatiojärjestelmän yleiskaavio muutosten jälkeen.

6.2 Automaatiojärjestelmän ohjelmistojen muutokset

Tässä luvussa 6.2 kuvataan opinnäytetyön kehittämisprosessin aikana tehtyjen automaatiojärjestelmän ohjelmistojen muutoksista. Opinnäytetyön aikana automaatiojärjestelmän PLC:n ohjelmaa ja käyttöliittymää muokattiin. Tässä luvussa kuvataan ensin PLC-ohjelmaan tehdyt muutokset, jonka jälkeen kuvataan automaatiojärjestelmän käyttöliittymään tehdyt muutokset.

PLC-ohjelman muutokset

PLC-ohjelman muutoksen tekemiseen nykyisen automaatiojärjestelmän ohjelma kopioitiin testi-PLC:hen. Seuraavaksi kaikki testi-PLC:ssä kiinni olevat laitteet lisättiin TwinCAT-ohjelmistoympäristöön ja laitteiden sisään- ja ulostulot linkitettiin niiden muuttujiin.

PLC:n ohjelmaan luotiin momenttisäätöön tarvittavat uudet muuttujat. Moottorin ajopaikan momenttianturin mittausvahvistimen ulostulo linkitettiin sille luotuun muuttujaan. Sillä PLC:n lukemalla momenttiarvolla on eri etumerkki kuin momenttiarvon näytöllä, sen etumerkki vaihdetaan PLC:n koodissa. Momenttiarvon muuttuja on tyypiltään kokonaisluku, ja se on annettu kymmenen milliNewton-metrin yksikössä.

PID-säädin lisättiin testi-PLC:hen käyttämällä TwinCAT 2 -ohjelmistoympäristön valmista "FB_BasicPID"-funktio lohkoa. PID-säätimestä käytettiin ainoastaan verrannollista ja integroivaa osaa. PID-säädintä viritettiin ainoastaan kokeellisin menetelmin. Suunnitelman mukaan PID-säädintä piti virittää käyttämällä jotain tiettyä PID-säätimen viritysmenetelmää, mutta tähän kehittämissaika ei riittänyt.

PID-säätimen momentin asetusarvon lähde riippuu automaatiojärjestelmän käyttöliittymän momentin asetusarvon syöttökentän valintaruudun tilasta. Jos valintaruutu on painettuna, PID-säätimen asetusarvo luetaan käyttöliittymän momentin asetusarvon syöttökentästä. Jos sen sijaan valintaruutu ei ole painettuna, PID-säätimen asetusarvo luetaan moottorin ohjauskapulasta.

Jotta PID-säädin saataisiin toimimaan valmiin momenttiramppifunktion kanssa, momenttirampin käyttöä osoittavan muuttujan tilaa on ensin luettava. Automaatiojärjestelmän käyttöliittymässä on pudotusvalikko, jolla momenttiramppia voi ottaa käyttöön ja valita rampin pituutta. Jos momenttiramppia ei ole käytössä, PID-säätimen asetusarvoksi syötetään joko moottorin ohjauskapulasta luettu momentin asetusarvo tai käyttöliittymästä syötetty asetusarvo. Jos jokin momenttiramppi on otettu käyttöön, moottorin ohjauskapulasta tai käyttöliittymästä luettu asetusarvo syötetään ramppifunktiolle. Momenttiramppifunktion ulostulo sitten syötetään PID-säätimen asetusarvoksi.

Moottorin ajopaikalta mitattu momenttiarvo on etumerkillinen, eli mitattu momentti on joko positiivinen tai negatiivinen riippuen moottorin pyörimissuunnasta. PID-säätimelle syötetty momentin asetusarvo on aina positiivinen. Tämän takia PID-säätimelle asetusarvolle ohjelmoitiin ehdollinen etumerkin vaihto. Jos moottori pyörii positiivisen momentin suuntaan, PID-säätimen asetusarvon

jätetään positiiviseksi. Jos moottori pyörii negatiivisen momentin suuntaan, PID-säätimen asetusarvon etumerkki muutetaan negatiiviseksi. Jos PID-säätimen asetusarvo ei olisi negatiivinen, kun moottori pyörii negatiivisen momentin suuntaan, PID-säädin vain suurentaisi momentin virhettä. Tämä tapahtuisi koska PID-säädin ohjaisi moottoria pyörimään nopeammin momenttivirheen pienentämiseksi, mutta sillä pyörimissuunta antaa negatiivisen mitatun momentin, momenttivirhe vain suurentuisi.

PLC:n koodia korjattiin poistamalla siitä turhia osia, lisäämällä joitakin kommentteja ja asettamalla käyttäjän määrittämät tietotyypit sopiviin kansiohierarkiaan. Alkuperäinen PLC:n koodi oli hiukan epäselvä, koska sitä on muokattu useita kertoja. Osa koodista ei ollut enää käytössä, ja käytössä oleva koodi oli välillä vaikeasti luettava.

Automaatiojärjestelmän käyttöliittymän muutokset

Automaatiojärjestelmän käyttöliittymän muutokset tehtiin kopioimalla nykyisen käyttöliittymän Visual Studio -projekti ja muokkaamalla sitä. Tämän opinnäytetyön kehittämisprosessin aikana suunniteltu käyttöliittymä kehitettiin käytettäväksi testi-PLC:n kanssa. Tämä tehtiin muuttamalla AmsNetID-osoitetta johon käyttöliittymä ottaa yhteyttä. AmsNetID-osoite on ADS-protokollan käyttämä osoite, jota käytetään ADS-viestien reitityksessä.

Automaatiojärjestelmän käyttöliittymään lisättiin tarvittavat muuttujat ja luotiin PLC:n arvoja kirjoittaville muuttujille ADS-protokollan tarvittavat käsittelijät. PLC:lle kirjoitetaan momentin syöttökentän asetusarvo ja momentin syöttökentän valintaruudun tila. ADS-protokollan käsittelijät kirjoittavat muuttujien arvot PLC:n muuttujiin, kun niihin tehdään muutoksia.

Automaatiojärjestelmän käyttöliittymään lisättiin mahdollisuus syöttää haluttu momentti näppäimistöllä valintaruudun tilan perusteella. Kun valittuna on näppäimistöllä syöttö, moottorin ohjauskapulalta luetun momentin asetusarvoa esittävän palkin tilalle ilmestyy momentin syöttökenttä. Silloin ohjauskapulalta luettu momentin asetusarvo näytetään numeerisena arvona palkin sijasta.

Ohjauskapulalta luettu momentin asetusarvo näytetään myös silloin kun momentin syöttökenttä on käytössä, jotta sen arvo olisi aina moottorin ajopaikan operaattorin tiedossa. Tällä tavalla on mahdollista muuttaa molempien momentin syöttötapojen asetusarvot samaksi ennen momentin syöttötavan muutosta. Kun momentin syöttötapojen asetusarvot ovat samat, ei yllättäviä momentin asetusarvojen muutoksia tapahdu momentin syöttötavan vaihtaessa.

Kun automaatiojärjestelmän käyttöliittymä käynnistetään, momentin syöttökentän arvo ja sen valintaruudun tila luetaan PLC:ltä. Tämä tehdään koska käyttöliittymän momentin syöttökentällä ja sen valintaruudulla on oletusarvot. Jos käyttöliittymä ei lukisi momentin syöttökentän arvoa ja sen valintaruudun tilaa, niiden arvot muuttuisivat käyttöliittymän oletusarvoiksi aina käyttöliittymän käynnistyessä. Käyttöliittymän käynnistyksen jälkeen käyttöliittymä ainoastaan kirjoittaa momentin syöttökentän ja sen valintaruudun tilan PLC:hen.

Moottorin ohjauskapulan nupin tilaa näyttävän palkin skaalaa muutettiin prosentin sadasosasta Newtonmetreihin. Vaikka momentin syöttökenttään syötetään momenttiarvo Newtonmetreinä, sen arvo on sen muuttujassa kymmenen Newtonmetrin yksikössä, kuten sen vastaavassa PLC:n muuttujassa. Muuttujan arvoa vain skaalataan visuaalisesti kertomalla se sadalla.

Automaatiojärjestelmän käyttöliittymästä korjattiin myös joitakin virheitä. Kun käyttöliittymä avattiin, ADS-symbolien lukuvirheen ponnahtusikkuna ilmestyi joka kerta. Tämä johtui siitä, että käyttöliittymä luki PLC:n muuttujia, joita ei ollut. Koodista korjattiin myös automaattiajon laskurin arvon näyttämistä. Aiemmin automaattiajon käyntiaikaa laskettiin PLC:llä, mutta käyttöliittymä ei koskaan päivittänyt näytettyä automaattiajon käyntiaikaa.

Automaatiojärjestelmän käyttöliittymän visuaalisia elementtejä myös päivitettiin. Automaattiajosekvenssin luonnin ponnahtusikkunaan lisättiin syöttökenttä momentille. Momentin lisäysmahdollisuutta ei kehitetty opinnäytetyön aikana, vaan automaattiajosekvenssin ponnahtusikkunan momentin syöttökenttä on ainoastaan visuaalinen elementti.

7 Yhteenveto

Opinnäytetyön tavoitteena oli momenttisäätimen toteutus moottorin ajopaikalle. Lisäksi tavoitteena oli myös muokata automaatiojärjestelmän käyttöliittymää, jotta näppäimistöllä pystytään syöttämään momentin asetusarvoa ja lisäämään momentin asetusarvo automaattiajosekvenssiin. Automaatiojärjestelmä oli myös tarkoitus dokumentoida kaikkien tehtyjen muutosten jälkeen.

Opinnäytetyön kehittämisprosessissa tuotettiin momenttisäädin moottorille. Momenttisäädin sijaitsee automaatiojärjestelmän muutosten tekemistä varten käyttöön otetussa testi-PLC:ssä. Käyttöliittymään luotiin momentin asetusarvon syöttökenttä. Sillä on mahdollista muuttaa testi-PLC:n momentin asetusarvoa. Käyttöliittymän momentin skaala korjattiin myös vastaamaan Newtonmetrejä.

Momentin asetusarvon lisäys automaattiajosekvenssiin, momenttisäätimen parempi viritys ja automaatiojärjestelmän dokumentointi ei ehditty tehdä. Opinnäytetyön aikana tarkentui miten näitä muutoksia kannattaa tehdä, joten opinnäytetyön jälkeen tätä kehittämistä aiotaan jatkaa.

Myös muun opitun säätötekniikan teoriaa ei ehditty hyödyntää opinnäytetyössä, kuten oman PID-säädin lohkon luominen valmiin ratkaisun käyttämisen sijasta. Tietyt säätimen muokkaukset toimisivat paremmin tilanteissa, joissa halutaan optimoida sekä asetusarvon seuranta että kuormitushäiriöiden vaimennuskykyä. Automaatiojärjestelmässä on reilusti viivettä, joten viiveen vaikutusta pienentävää ohjauselementtiä olisi hyvä lisätä momenttisäätimeen.

Vaikka kaikkia opinnäytetyön tavoitteita ei saavutettu, työtä voi jo osittain hyödyntää moottorin ajopaikalla. Sen hyödyntäminen toki vaatii testi-PLC:n käyttöä. Opinnäytetyön jälkeen automaatiojärjestelmän nykyinen PLC kenties korvataan testi-PLC:llä.

Lähteet

- 1 Dorf, Richard & Bishop, Robert. 2016. Modern Control Systems. 13., painos. Boston: Pearson.
- 2 SFS-IEC 60050-351. 2015. Sähköteknillinen sanasto. Osa 351: Ohjaustekniikka. Suomen Standardisoimisliitto.
- 3 European Union terminology. Versio 2.34.1. 2025. Tietokanta. Interactive Terminology for Europe.
- 4 Zenger, Kai. 2022. ELEC-C1230 Sääntötekniikka Johdanto luku 1 ja 2. Opintomateriaali. Aalto yliopisto.
- 5 Harju, Timo & Marttinen Arto. 2000. Sääntötekniikan koulutusmateriaali. Koulutusmateriaali. Suomen Automaatioseura ry.
- 6 Altmann, Wolfgang. 2005. Practical Process Control for Engineers and Technicians. Oxford: Elsevier.
- 7 Åström, Karl. 2002. Control System Design. Opintomateriaali. Lund Institute of Technology.
- 8 Visioli, Antoni. 2006. Practical PID Control. Brescia: Springer.
- 9 Nise Norman. 2014. Control systems engineering. 7., painos. E-kirja. Wiley.
- 10 Åström, Karl & Hägglund, Tore. 1995. PID Controllers: Theory, Design, and Tuning. 2., painos. Research Triangle Park: Instrument Society of America.
- 11 Bernhardsson, Bo & Åström, Karl. 2016. Control System Design – PID Control. Opintomateriaali. Lund University.
- 12 Åström, Karl & Murray Richard. 2012. Feedback Systems. 2., painos. E-kirja. Princeton University Press.