

## OFFLINE-MITTALAITTEISTON ANTURIPÄIVITYS



Ammattikorkeakoulututkinnon opinnäytetyö  
Sähkö- ja automaatiotekniikka, insinööri (AMK)

Syksy 2022

Anna Haarala

Sähkö- ja Automaatiotekniikka

Tekijä Anna Haarala

Työn nimi Offline mittalaitteiston anturipäivitys

Ohjaaja Jan Nowak

Tiivistelmä

Vuosi 2022

---

Työn tarkoituksena on suorittaa offline-mittalaitteistoon anturin vaihdos. Anturin vaihdos on kriittinen toimenpide, koska vanhaa anturia ei enää valmisteta. Anturin vaihdos suoritetaan liikuteltavaan laboratorio-mittaluokan laitteistoon. Saatuja tuloksia hyödynnetään laitteen tehdasversion toiminnan tukemiseen.

Työhön kuului anturin asentaminen, konfigurointi käyttöön, ohjelmointi olemassa olevaan ohjelmaan ja käyttöönotto. Mittalaitteiston vanha anturi suoritti tiedonsiirron milliampeeriviestinä, mutta uusi anturi käyttää tiedonsiirrossa Modbus-protokollaa. Laitteiston PLC on Beckhoff ja ohjelmisto on tehty TwinCatilla.

Anturin vaihto on mahdollista, mutta uusien antureiden kalibrointi on syytä todentaa ennen kentälle asentamista. Tulokset, joita saatiin anturien uudelleen kalibroinnin jälkeen ovat suoraan verrannollisia vanhaan anturiin ja anturin herkkyys on hyvä.

Avainsanat Modbus, anturi, tiedonsiirto, automaatioarkkitehtuuri, Beckhoff

Sivut 32 sivua ja liitteitä 1 sivua

Electrical and Automation Engineering

Author Anna Haarala

Subject Offline-equipment sensor update

Supervisors Jan Nowak

Abstract

Year 2022

---

The purpose of the work was to update a sensor to offline-equipment. Sensor update was critical because the old sensor was no longer in production. The sensor change was done in a portable laboratory scale equipment. The results received from this sensor change will be utilized as support to factory scale equipment.

The work included installing the sensor, configuration on how to use it and coding the sensor to the existing program and deployment. The old sensor that was used in the equipment used milliampere message. The new sensor uses the Modbus-protocol in data transfer. The PLC of the equipment was Beckhoff based and the software is made with TwinCat.

When installing new sensors the factory calibration of new sensors ought to be verified before the installation of the sensors. The results that were received after re-calibration were proportional to the old sensor and the sensitivity was good.

Keywords Modbus, sensor, data transfer, automation architecture, Beckhoff

Pages 32 pages and appendices 1 pages

## Sisällys

1	Johdanto .....	1
2	Automaatioarkkitehtuuri.....	2
3	Kenttälaitteet.....	3
3.1	Mittausanturi ja mittauslähetin .....	4
3.2	Anturilla mittaaminen .....	5
4	Tiedonsiirto.....	6
5	Tiedonsiirto, instrumentointi ja ohjausjärjestelmässä.....	7
6	Tiedonhallinta.....	8
6.1	Hajautettu ohjausjärjestelmä (DCS).....	8
6.2	Ohjelmoitava logiikka (PLC) .....	9
6.3	SCADA.....	11
7	Modbus.....	12
7.1	Modbus protokolla.....	13
7.2	Modbus toimintokodeista .....	14
7.3	Modbus ongelmat .....	15
8	Mittalaitteisto.....	15
9	Mittalaitteiston PLC.....	17
9.1	PLC- topologia .....	17
9.2	TwinCAT3 ja ohjelmistokieli .....	19
10	Työn toteutus .....	19
10.1	Laitteiston päivitys ja käyttäjä palaute .....	20
10.2	Anturin konfiguraatio.....	21
10.2.1	Modbus-funktio.....	22
10.2.2	Anturin näkyminen käyttöliittymässä .....	24
11	Anturin testaus .....	25
12	Loppupäätelmä.....	31

## **Liitteet**

Liite 1 Anturinkytkeä valmistajalta

## 1 Johdanto

Tämän opinnäytetyön tarkoituksena on tehdä liikuteltavaan offline-mittalaitteistoon anturinvaihdos. Teoreettisessa osuudessa tutkin automaatioarkkitehtuuria kenttälaitteelta, kenttäväylän kautta ohjelmoitavaan logiikkaan.

Automaation kehitys on mahdollistanut teollisen kehityksen, jolla kuluja ja tuotantoa saadaan tehostettua. Muihin automaation hyötyihin kuuluvat työntekijäturvallisuuden parantuminen ja laadulliset tekijät. Ohjelmoitavat logiikat ja smart sensorit mahdollistavat nopean analyysin prosessissa.

Työ toteutettiin yhteistyössä yrityksen kanssa, mutta koska kyseessä on yrityksen liikesalaisuudet, tullaan työssä käsittelemään hyvin yleisellä tasolla laitteistoa ja sen käyttötarkoitusta. Työn salaisen luonteen vuoksi käytän vanhasta anturista nimitystä vanha anturi ja uudesta anturista nimitystä uusi anturi.

Työ sai alkunsa todellisesta tarpeesta vaihtaa liikuteltavan offline-mittalaitteiston anturi uuteen. Vanhaa anturia ei enää valmisteta. Vanha anturi toimi analogisella liitännällä, kun uusi anturi toimii Modbus-väylällä. Offline-mittalaitteisto toimii tuotantojen Online-mittalaitteiden tukena. Anturin vaihdoksessa tärkeintä on, että uusi anturi tuottaa vertailukelpoista mittausaineistoa vanhaan anturiin verrattuna.

Työ tehtiin muutoksenhallintana, koska kyseessä oli olemassa oleva mittalaite. Samalla kartoitettiin mittalaitteiston muut kehityskohteet. Käyttöliittymä ja ohjaukoodi olivat valmiina ja itse fyysinen laitteisto kasattu. Muutostyönä tehtiin uuden anturin asennus Beckhoffin logiikan RS-458 sarjaporttiin Modbus-väylällä sekä anturin kytkentä. Ohjaukoodin päivitys oli keskeisessä osassa työtä. Dokumentaatio päivitettiin vastaamaan nykyistä tilannetta.

Uuden anturin toimivuus todennettiin tekemällä vertailumittauksia vanhan anturin kanssa rinnakkain. Mittauksissa käytettiin tarkistusliuosta ja todellisia näytteitä. Tämä mahdollisti antureiden vertailun ja tulosten vertailukelpoisuuden keskenään.

## 2 Automaatioarkkitehtuuri

Väylätekniikan kehittymisen ja standardoitumisen myötä automaatioarkkitehtuuri on muuttunut. Eri automaatioiden yhteenliittymät voidaan toteuttaa aiempaa helpommin. Automaatiojärjestelmä koostuu eri automaatioasemista, jotka suorittavat eri tehtäviä. Prosessi koostuu prosessiasemista (PCS eli Process Control Station), jotka hoitavat säätöjä, mittauksia ja ohjauksia. Kenttälaitteet ovat yhdistetty prosessiin binääristen ja analogisten liitäntöjen lisäksi myös väylien avulla. Sulautettu I/O voidaan liittää HART-kenttälaitteeseen (HART- Highway Addressable Remote Transducer). Kenttälaitteiden välisenä väylänä voidaan käyttää esimerkiksi Profibus PA:ta (Profibus DP:n kautta), Foundation Fieldbussia ja ASiA. Nämä väylät ovat suoraan kytkettävissä prosessiasemiin. Asemat on kytketty toisiinsa optisten tai sähköisten väylien avulla. (Kippo & Tikka, 2008 s. 44)

Käyttöliittymällä eli HMI:llä (Human Machine Interface) tarkoitetaan tietokoneen ja ihmisen väliseen vuorovaikutukseen tarvittavia laitteita ja ohjelmia. Käyttöliittymä suunnitellaan käyttötarpeiden mukaan. Prosessin ohjaus suoritetaan käyttöliittymän kautta. Käyttöliittymiä on eri tasoissa. Operaattori tarvitsee käyttöliittymän, josta hoidetaan prosessin ohjaus. Operaattorille näkyvät prosessilaitteet, mittaus- ja ohjauskohteet sekä näiden liittyminen toisiinsa. Operaattori pystyy muuttamaan asetusarvoja ja ohjaustapaa näytöltä, sekä ajamaan laitetta manuaalitulassa. Suunnittelijalla on oma käyttöliittymä, jossa voidaan tehdä prosessin sovellussuunnittelua, simuloida piirien toimintaa ja tehdä prosessinäyttöön muutoksia. Kunnossapitoa varten on oma käyttöliittymä. Kunnossapitokäyttöliittymällä voidaan tarkkailla ja tehdä kunnossapitoon liittyviä toimenpiteitä. (Kippo & Tikka, 2008, s. 46)

Automaatiojärjestelmissä on prosessiasemia. Prosessiaseman tehtävä on liittää järjestelmä prosessiin sekä hoitaa säätö-, mittaus-, logiikka- ja ohjaustoiminnot. Prosessiasemat toimivat tiedonsiirron solmupisteinä. Prosessiasema on joko suoraan liitetty prosessiin prosessiliitäntäyksiköiden kautta tai väylän avulla. (Kippo & Tikka, 2008, s. 49–50)

Tavallisimmat yksiköt Kipon (2008) mukaan ovat

- binäärinen lähtöyksikkö (BO) jolla toteutetaan ON/OFF- ohjauksia
- binäärinen tuloyksikkö (BI) jolla toteutetaan rajakytkimien tilan tutkintaa
- analoginen lähtöyksikkö (AO) jolla ohjataan esimerkiksi venttiilin asentoa standardiviestin avulla
- analoginen tuloyksikkö (AO) joka välittää mittausviestit automaatiojärjestelmään

Liitäntäyksiköitä on useita ja ne valitaan käyttöön erilaisten suojaustarpeiden sekä käytettävän virta- ja jännitetason mukaan. Kenttäväylät ovat mahdollistaneet älykkäät kenttälaitteet. Digitaalinen väylä siirtää tietoa molempiin suuntiin. Älykkäiden kenttälaitteiden ominaisuuksiin kuuluu muun muassa itsediagnostiikka. Väylän avulla voidaan muuttaa parametrejä helposti ja kenttälaitteet voivat suorittaa laskentaa kuten prosessiasemat. (Kippo & Tikka, 2008, s. 49–50)

### 3 Kenttälaitteet

1960-luvulla 4–20:n milliampeeriviesti kehitettiin de facto-standardiksi instrumenteille ja kenttälaitteille. Instrumenttivalmistajilla oli standardi viestintäraja pinta mihin pohjata tuotteensa. Tämä mahdollisti asiakkaalle valinnan vapauden eri instrumenttitoimittajien välillä. Mikroprosessorien tulon ja tekniikan kehityksen myötä tilanne on muuttunut. Sama instrumentti voi mitata useampaa arvoa prosessista. Instrumenteilla voi olla paikallisia ja etänäyttöjä. Instrumenttien luotettavuus, taloudellisuus, mahdollisuus sisäiseen viritykseen ja diagnostiikkaan on parantunut. Teollisuudessa tapahtuu asteittaista muutosta analogisesta tekniikasta digitaaliseen. (Mackay, 2004, s.4)

Mittausantureilla, lähettimillä ja rajakytkimillä seurataan prosessin tilaa. Prosessiin vaikutetaan pumpuilla, venttiileillä, lämmitys- ja jäähdytyslaitteilla. Näitä laitteita ohjataan automaatiolaitteilla. Liityntä digitaalisilla mittalaitteilla tehdään tavallisesti analogisesti. Milliampeeriviesti 4–20 mA on yleisin käytössä oleva viestimuo. Digitaalisissa

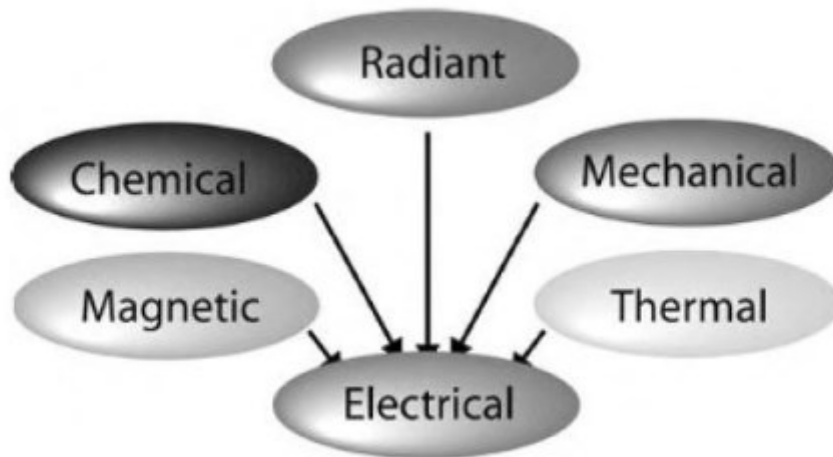


kenttälaitteissa on useita parametrejä, joiden tilaa on tarkistettava tai muutettava käytön aikana. Digitaalinen kaksisuuntainen tiedonsiirto kenttälaitteen ja automaation välillä mahdollistaa sen, että kenttälaitteen parametrit voidaan muokata suoraan suunnittelijan tai kunnossapitohenkilön työasemalta. (Kippo & Tikka, 2008, s. 59)

### 3.1 Mittausanturi ja mittauslähetin

Anturit muuttavat signaalin erilaisilta energia-alueilta sähköiseksi. Kuvassa 1 anturit on lajiteltu kuuteen luokkaan.

Kuva 1. Antureiden luokat (Meijer, 2008, s. 3)



Kuvassa 1 ylimpänä luokassa on optinen mittaus (Radiant), jossa optinen signaali muuttuu elektroniseen muotoon. Esimerkkinä voidaan käyttää optista anturia, joka kääntää signaalin elektroniseen muotoon. Mekaaniset signaalit, kuten lämpötilan mittaus, muuttavat mitatun arvon sähköiseksi signaaliksi. Jopa sähköantureita on olemassa. Sähköanturit muuttavat sähköisen signaalin toiseksi sähköiseksi signaaliksi. Magneettista signaalia pystyään muuttamaan sähköiseksi signaaliksi. Viimeisenä kemiallinen ja biokemiallinen anturi pysty kääntämään viestin sähköiseksi tästä esimerkkinä pH-anturi ja DNA anturit. (Meijer, 2008 s. 3)

Anturit voidaan edelleen jakaa aktiivisiin ja passiivisiin antureihin. Aktiivinen anturi voi saavuttaa suuren tehonlisäyksen lähtö- ja tulosignaalien välillä, kun taas passiivinen anturi saa lähtö energiansa tulosignaalista esimerkiksi paineesta. Aktiivisen ja passiivisen anturin suurin ero on mittaus resoluutio. Aktiivisella anturilla saavutetaan suurempi resoluutio, mutta sekään ei ole rajaton, koska ulkoinen energiansyöttö voi aiheuttaa sille rajoitteita. Passiiviset anturit ovat halvempia, joten hankintaan vaikuttaa resoluution tarve ja kustannukset. (Morris, 2001, s. 12–13)

Jatkoluokitteluna anturin voi määrittää avoimeksi järjestelmäksi (open system), jossa anturi ei anna palautetta tai suljetun silmukan järjestelmäksi (closed-loop system), jossa anturi antaa palautteen. (Meijer, 2008, s. 4)

Mittausanturi muuttaa halutun prosessi suureen sähköiseksi suureksi. Mittausanturilta saatutieto muutetaan standardiksi mittausviestiksi mittauslähettimellä. Mittauslähettimeltä viesti lähtee muun automaation käyttöön. Mittauslähettimen avulla voidaan mittaussignaali muuttaa lineaariseksi, valita sopiva mittausalue mittausanturin spesifikaatioiden mukaan ja suodattaa häiriöitä mittaussignaalista. Lähetin kommunikoi usein muun automaation kanssa digitaalisesti standardeja ja viestiprotokollia käyttäen. Analoginen viesti sisältää mittausarvon. Analogisen viestin suunta on automaatiojärjestelmään päin. Digitaalisessa tiedonsiirrossa viestit kulkevat kahteen suuntaan samassa kaapelissa. Mittausanturi ja lähetin on integroitu usein yhteen. (Kippo & Tikka, 2008, s. 59–60)

Kun anturin analoginen liitäntäpiiri, analogia-digitaalimuunnin (ADC tai A/D) ja väylän käyttöliittymä asennetaan samaan koteloon, saadaan smart -anturi eli älykäsanturi. Jos kaikki toiminnot siirretään yhdelle sirulle, saadaan integroitu älykäs sensori. (Meijer, 2008, s. 9)

### **3.2 Anturilla mittaaminen**

Automaation tavoite on saada ohjatut järjestelmät käyttäytymään halutulla tavalla. Tietoa järjestelmän toiminnasta saadaan mittaamalla eri suureita. Jotta järjestelmän ohjaus toimii tarpeeksi nopeasti, pitää mittauksen olla nopea ja tarkka. Mittauksella määritetään jonkin

ominaisuuden suuruus. Kansainvälisen SI-mittayksikön suureita käytetään mittauksissa. Mitattu suure ilmaistaan verrannollisena johonkin vertailupisteeseen verrattuna. Tällainen vertailupiste voi olla esimerkiksi veden kiehumispiste. Vertailupistettä kutsutaan normaaliksi ja mittalaite säädetään näyttämään mahdollisimman tarkasti oikeita mittoja tai arvoja verrattuna normaaliin. (Kippo & Tikka, 2008, s. 21–22)

Mittaukseen liittyy aina epätarkkuutta. Epätarkkuus ilmoitetaan mittayksiköissä tai prosenteissa. Epävarmuus voi olla virhe täydestä mitta-alueesta, mittauksen hetkellisestä arvosta tai mittauksen maksimiarvosta. Kalibroinnilla voidaan selvittää mittalaitteen lukeman virheen suuruus suhteessa referenssi- eli vertailumittalaitteeseen. Kalibroinnilla saadaan selville mittalaitteen virhe tietyssä olosuhteessa tietyllä hetkellä. Virityksellä havaitaan ja korjataan mittalaitteen virhe mahdollisimman lähelle referenssi- eli vertailumittalaitteen lukemaa. Mittausanturin ominaisuudet muuttuvat ajan kuluessa. (Kippo & Tikka, 2008, s. 21–23)

Epätarkkuus kasvaa ja aiheuttaa ajautumavirhettä, kun mittausanturi vanhenee. Todellisen ja mitatun arvon välinen erotus on mittausvirhe. Mittausvirhe voidaan ilmoittaa prosenteissa tai mittayksikössä mittausalueesta. Mittauksen tulee olla luotettava ja toistettava. Luotettavuudella tarkoitetaan laitteen kykyä suorittaa tarvittu mittaus määritetyissä olosuhteissa tietyn ajan tai tietyn mittausmäärän. Toistettavuus osoittaa, kuinka identtisiä mittauksia voidaan saada toistettavalla mittauksella. (Kippo & Tikka, 2008, s. 21–23)

Mittalaitteen herkkyys kertoo mittalaitteen lähtömuuttujan muutoksen suhdetta tulomuuttujaan eli mitattavan kohteen vastaavaan muutokseen. Herkkyys esimerkiksi ilmoitetaan yksiköillä V/mm tai mV/ °C. Resoluutio eli erotuskyky kuvaa pienintä havaittavaa muutosta mitattavassa suureessa. Suuri resoluutio tarkoittaa, että mittalaite voi havaita pienen muutoksen. (Kippo & Tikka, 2008, s. 21–23)

#### **4 Tiedonsiirto**

Tietoa siirretään kohteesta toiseen. Moni tieto siirtyy analogisesti, mutta nykyaikana suurin osa tiedosta siirtyy digitaalisena. Tiedonsiirrossa tieto siirtyy nollina ja ykkösinä. Kaikki tieto

tarvitsee lähettäjän ja vastaanottajan ja linkin näiden kahden toimijan välille. Linkkinä toimivat kuparikaapeli, optinen kaapeli sekä radio- ja mikroaallot. Jotkut lyhyet linkit käyttävät rinnakkaisia yhteyksiä, mikä tarkoittaa, että useita kaapeleita tarvitaan signaalin kuljettamiseen. Tiedon kuljettamiseen pitää olla yhteinen sopimus, jotta lähettäjä ja vastaanottaja ymmärtävät toisiaan. Tätä järjestelmää lähettäjän ja vastaanottajan välillä kutsutaan protokollaksi. (Mackay, 2004, s. 2)

Protokollaa käytetään kommunikaatiojärjestelmän sisällä. Protokollat olivat perinteisesti laitevalmistajien ja ohjelmistokehittäjien omia, jolloin laitteiden keskinäinen integrointi oli vaikeaa. Integroinnin helpottamiseksi tuli standardisointi. Standardi voi kehittyä valmistajan protokollasta (de facto standardiksi) tai se kehitetään tietyn yritysalan toimijoiden kesellä. Standardit mahdollistavat uusien tuotteiden integroinnin jo olemassa oleviin järjestelmiin. Teollisuusalalla on muutama vallitseva standardi. Modbus on ollut de facto standardi ja fyysisiä standardeja ovat olleet EIA-232 ja EIA-485. Instrumentoinnin ja logiikan yhteen sovittamiseen fieldbussin välityksellä. Tämän vuoksi on noussut kilpailevia fieldbus standardeja kuten Profibus, ASi, Foundation Fieldbus, DeviceNet ja Ethernet. (Mackay, 2004, s. 2–3)

## **5 Tiedonsiirto ja instrumentointi ohjausjärjestelmässä**

Tietoa hankitaan ohjausjärjestelmään erilaisilla mittalaitteilla eli instrumenteilla. Tieto siirretään kontrollerille. Kontrolleri välittää tiedon eli ohjaussignaalin ohjauslaitteelle, joka toimii prosessissa. Integroidut järjestelmät mahdollistavat tiedon nopean ja tehokkaan siirron järjestelmien välillä. Tämä vähentää kaapeloinnin ja päätepisteiden tarvetta. (Mackay, 2004, s. 2–3)

Instrumentoinnin ja ohjausjärjestelmän tarkoitus prosessissa on prosessin ohjaus ja hälytysten antaminen. Tyypillisesti prosessin ohjaus on tehty analogisilla ohjaimilla 4–20:n mA silmukkaviestillä. Milliampeeriviestin käyttöä hyödynnetään monissa laitteissa ja on tavallista, että eri laitteita yhdistetään samaan ohjausjärjestelmään. Erilliset ohjaimet ja instrumentit on nykyaikana korvattu integroiduilla järjestelmillä kuten hajautetuilla järjestelmillä. (Mackay, 2004, s. 2–3)

Instrumentointi ja ohjausjärjestelmä ohjaa sekvenssejä, lukituksia ja hälytyksiä. Ohjauksen sekvensointi, lukitus ja hälytykset on tyypillisesti tehty releillä, ajastimilla ja muilla ohjauskeskukseen tai moottorikeskukseen johdotetuilla komponenteilla. Sekvenssit määrittävät käyntijärjestyksen laitteille. Hälytykset, lukitukset ja sekvenssi ohjaus on pääsääntöisesti korvattu ohjelmoitavalla logiikalla. Hälytys ja lukitusohjaukset muodostavat ison osan turvalogiikkaa. (Mackay, 2004, s. 2–3)

Käyttöliittymän tarkoitus on toimia prosessin ohjauksessa ja näyttönä reaaliaikaisille prosessi arvoille. Aiemmin ohjaus suoritettiin paikallisilta näytöiltä usean operaattorin toimesta. Jokainen operaattori oli vastuussa jostakin prosessin osasta. Modernit järjestelmät käyttävät yhtä valvomoa koko laitoksen valvomiseen. Valvomo on varustettu tietokonetyöpisteillä, jotka keräävät tietoa kenttälaitteilta ja muuttavat sen graafiseksi näytöksi prosessinvalvontaan, hälytysten valvontaan, sekvenssiohjaukseen ja lukitukseen. (Mackay, 2004, s. 2–3)

## **6 Tiedonhallinta**

Aiemmin ohjausta varten tarvittavat tiedot kerättiin manuaalisesti eri näytöiltä, ohjauslaitteilta tai prosessinäytteiden analyysistä. Nykyään tiedonkeruu on integroitu ohjausjärjestelmään. Integroitujen elektroniikkalaitteiden, mikroprosessorien ja tietoliikenteen kehityksen myötä tiedonhallintajärjestelmät ovat kehittyneet. Prosessien hallinnan kannalta neljä keskeisintä kehitysaskelta ovat olleet hajautetut ohjausjärjestelmät (DCS), ohjelmoitavat logiikat (PLC), SCADA (supervisory control and data acquisition) järjestelmät ja Smart-instrumentit. (Mackay, 2004, s. 3)

### **6.1 Hajautettu ohjausjärjestelmä (DCS)**

Hajautettu ohjausjärjestelmä on laitteisto- ja ohjelmistopohjainen prosessinohjaus- ja tiedonhankintajärjestelmä. Hajautettu ohjausjärjestelmä pohjaa tiedonvalvontatieteen ja se kehitettiin korvaamaan monet erilliset PID-säätimet. Siinä on modulaarinen, hajautettu mutta integroitu arkkitehtuuri. Jokainen moduuli suorittaa annettua spesifiä tehtävää. Normaalisti tiedonvalvontatiellä on käyttöliittymäyksikkö, joka mahdollistaa helpon kytkennän

muihin laitteisiin, kuten ohjelmitavaan logiikkaan tai valvoviin tietokoneisiin. (Mac Kay, 2004, s. 3)

## 6.2 Ohjelmitava logiikka (PLC)

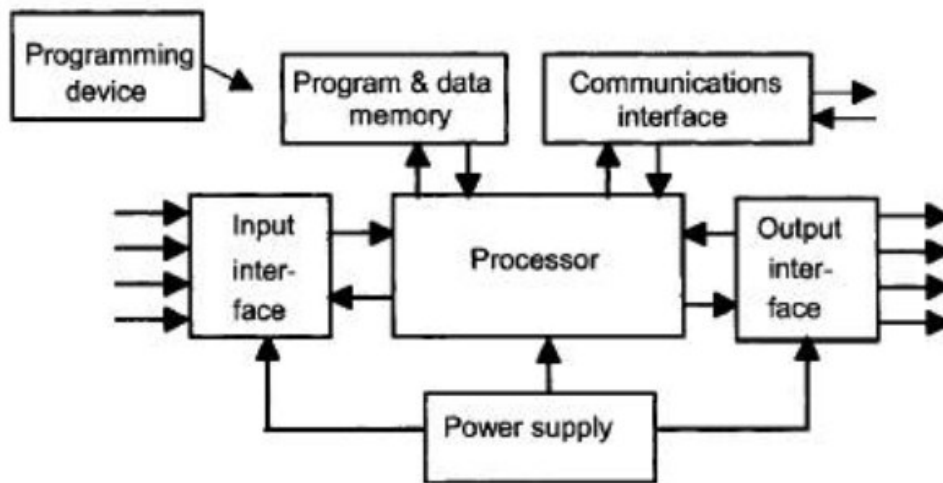
Ohjelmitavat logiikat (PLC eli Programmable Logic Controller) kehitettiin 60-luvun lopulla korvaamaan elektromagneettisia releitä. Releitä käytettiin ensisijaisesti sekvenssien ohjaukseen ja lukitukseen. Ohjelmitava logiikka sisältää analogiset ja digitaaliset I/O moduulit sekä mahdollisuuden hajautetun ohjausjärjestelmän käyttöön. (Mackay, 2004, s. 3–4)

Ohjelmitava logiikka on itsenäinen yksittäinen automaatiojärjestelmä. Tätä kutsutaan myös Stand alone -laitteeksi. Järjestelmä suorittaa tietyn prosessin osaa, joka toimii vaiheittaisesti. Ohjelmitava logiikka suorittaa omaa tehtäväänsä annettujen toimintaohjeiden eli logiikalle ladattujen sovellusohjelmien mukaisesti. Prosessiin PLC liittyy prosessiiliityntäyksiköiden kautta. Ohjelmitavassa logiikassa laskentaoperaatiot ovat pääosin loogisia AND, OR, NADN ja NOR- operaatioita. (Kippo & Tikka, 2008, s. 54–56)

PLC eli ohjelmitava logiikkaohjain on mikroprosessoripohjainen ohjain, joka käyttää ohjelmitavaa muistia ohjeiden säilyttämiseen ja funktioiden toteuttamiseen. Tällaisia funktioita ovat muun muassa sekvensointi, ajastaminen, laskeminen ja aritmeettiset toiminnot. Näillä toiminnoilla ohjataan koneita ja prosesseja. Ohjelmitava logiikkaohjain on esiohjelmoitu, jotta ohjausohjelma voidaan keskittää. Ohjelmitava logiikkaohjain toteuttaa logiikka ja kytkentätoimintoja. Tulo-laitteet, kuten anturit tai kytkimet ja lähtölaitteet, kuten moottorit ja venttiilit on kytketty ohjelmitavaan logiikkaan. Kirjoitettu ohjelma tallennetaan ohjelmitavan logiikan muistiin. Ohjain seuraa tulo- ja lähtösignaaleja ja ohjelman mukaan toimittaa sääntöjä, jotka ovat ohjelmaan luotu. (Bolton, 2006, s. 3)

PLC-järjestelmässä ovat normaalisti peruskomponentit prosessoriyksikkö, muisti, virtalähde, input/output-käyttöliittymäosio, viestintärajapinta ja ohjelmointilaite. Kuvassa 2 on esitetty PLC- järjestelmän perusjärjestys.

Kuva 2. PLC- järjestelmä (Bolton, 2006, s. 4)



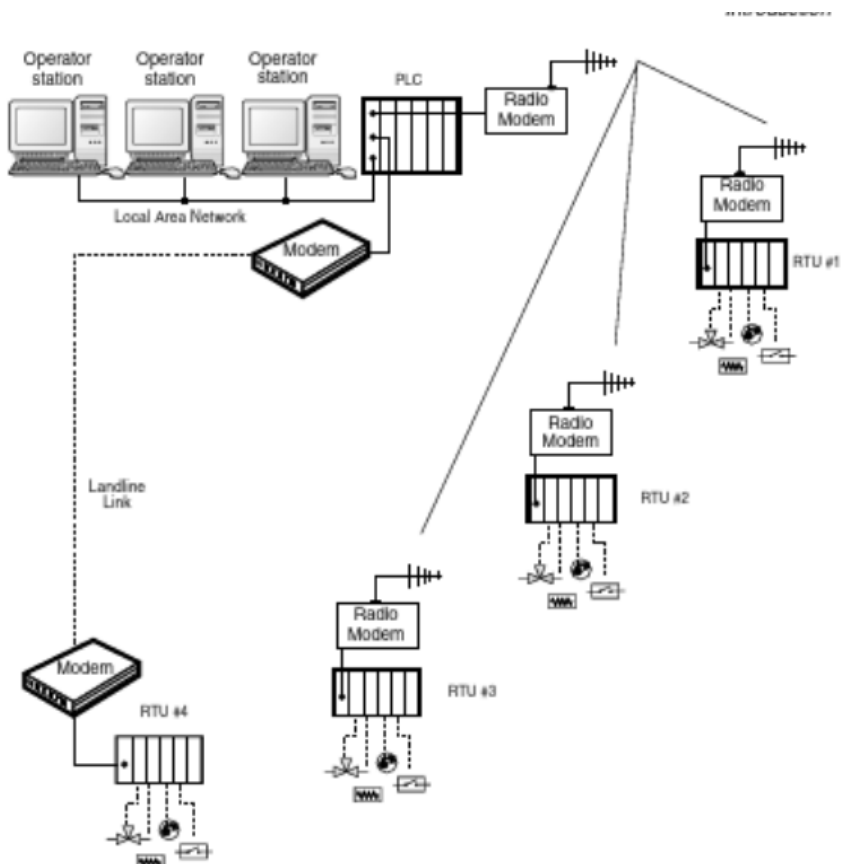
Prosessoriyksikkö eli CPU (central processing unit) sisältää mikroprosessorin. Mikroprosessori tulkitsee tulosignaalit ja suorittaa ohjaustoiminnot, jotka on tallennettu sen muistiin. Tämän jälkeen se viestittää ohjelman vaatimat toimintasiinaalit signaalilähtöön. Virtalähdeyksikköä (power supply unit) tarvitaan muuttamaan vaihtovirtajännite mataliksi tasajännitearvoiksi, jota prosessori ja tulo- sekä lähtöpiirit käyttävät. Ohjelmointilaitetta (programming device) tarvitaan käytettävän ohjelman siirtämiseen prosessorin muistiin. Ohjelma kirjoitetaan laitteelle usein tietokoneella ja siirretään PLC-muistikortille tai muistiin. Muisti (Memory unit) on paikka, jossa ohjelma on tallennettuna. Sinne voidaan myös tallentaa prosessin tiedot. Tulo- ja lähtöosioissa prosessori saa tiedon ulkoisilta laitteilta ja kommunikoi tietoa ulkoisille laitteille. Tulo- ja lähtölaitteet voidaan jakaa antavan signaalin mukaan diskreettisiinaaliksi, digitaaliseksi- tai analogiseksi signaaliksi. Diskreettiä tai digitaalista signaalia antavat laitteet ovat niitä, joissa signaali on joko päällä (ON) tai pois päältä (OFF). Analogiset laitteet antavat signaalia, jonka koko on suhteellinen seurattavan muuttujan kokoon. Esimerkiksi lämpötila-anturi antaa mA-arvon suhteessa lämpötilaan.

Viestintärajapintaa (communications interface) käytetään datan vastaanottamiseen ja lähettämiseen viestintäverkossa. (Bolton, 2006, s. 4–5)

### 6.3 SCADA

SCADA-järjestelmä eli Supervisory Control And Data Acquisition, koostuu etäpääteyksiköistä eli RTU:sta eli Remote terminal uniteista, jotka toimittavat tiedot päävalvomoon. SCADA etäpääteyksiköt keräävät kentältä tiedon ja yhdistävät ne takaisin pääasemalle tietoliikenneväyliä pitkin. Kuvassa 3 on esitetty SCADA järjestelmä.

Kuva 3. SCADA järjestelmä (Mackay ym., 2004, s. 5)



Kuvassa 3 nähdään, että kolme RTU-yksikköä on yhdistetty etäyhteydellä ohjelmoitavaan logiikkaan ja yksi RTU on kytketty kaapelilla. Jokaisessa RTU:ssa on erilaisia toimilaitteita, joita järjestelmä valvoo ja ohjaa. (Mackay, 2004, s. 4–5)



## 7 Modbus

Modicon eli nykyinen Schneider Electric kehitti Modbus protokollan vuonna 1979. Sitä käytetään luomaan älykkäiden laitteiden välille asiakas-palvelin -kommunikaatio. Modbus on standardoitu, avoin ja laajasti käytetty verkkoprotokolla teollisessa valmistusympäristössä. Sitä voidaan käyttää tuhansissa eri laitteissa diskreettien, analogisten I/O- ja rekisteritietojen siirtämiseksi ohjauslaitteiden välillä. Asiakas-palvelin sovelluksissa Modbusia käytetään laitteiden valvontaan ja ohjelmointiin. (Modbus, n.d)

Modbusia käytetään useissa asiakas-palvelinsovelluksissa laitteiden valvontaan ja ohjelmointiin esimerkiksi kommunikointiin älykkäiden laitteiden, antureiden ja instrumenttien välillä sekä kenttälaitteiden valvontaan PC:n ja käyttöliittymän avulla. Modbus on myös ihanteellinen protokolla RTU-sovelluksiin, joissa tarvitaan langatonta viestintää. Tästä syystä sitä käytetään lukemattomissa kaasu-, öljy- ja sähköasemasovelluksissa. Modbus ei ole vain teollinen protokolla. Myös rakentaminen, infrastruktuuri, liikenne ja energiasovellukset hyödyntävät sen etuja. (Modbus, n.d)

Modbus on yksinkertainen, joustava ja julkinen protokolla, joka mahdollistaa laitteiden erillisen ja analogisen tiedon siirron. Modbus järjestelmän haasteisiin kuuluu sarjaväylien hitaus 9600sta 115000 baudiin. Tämä tarkoittaa 0.010 Mbps- 0.155 Mbps nopeutta, kun verrataan Ethernetin jopa 10 Gbps nopeuteen. Käytettäessä EIA-232 tai EIA-485 laitteita, rajoitteena on integroitavien laitteiden määrä. Esimerkiksi 500:n laitteen kytkentä EIA-485 järjestelmällä vaatii monimutkaisen hierarkian asiakkaita ja palvelimia sisältävässä puutopologiassa. Modbusin sarjayhteydet ovat yhden asiakkaan malleja. Tämä tarkoittaa, että yksi laite ei voi viestiä usealle palvelimelle. Tämä seurauksena yksi asiakas on tietoinen kaikesta reaaliaikaisesta tiedosta. (Mackay, 2004, s.97–99)

Modbus-protokollassa on kiinteitä ominaisuuksia, kuten kehysmuoto, kehyssekvenssit, tietoliikennevirheiden käsittely ja poikkeusolosuhteet sekä suoritettavat toiminnot. Muut ominaisuudet voidaan valita. Muhin ominaisuuksiin kuuluvat lähetysväline, lähetysominaisuudet ja lähetystapa eli RTU tai ASCII. Käyttäjän ominaisuudet on asetettu jokaiseen laitteeseen. Käyttäjän ominaisuuksia ei voi muuttaa järjestelmän ollessa käynnissä.

ASCII on luettava tiedonsiirtotila ja sitä käytetään esimerkiksi testaamiseen. RTU on kompakti ja nopea ja sitä käytetään normaaliin toimintaan. RTU-tilaa kutsutaan joskus myös Modbus-B:ksi Modbus-Binary muodossa, joka on suositeltu Modbus-tila. ASCII-lähetystilassa sanoman pituus on noin kaksikertaa RTU-viestin pituus. Lähetys- ja viestintävirheille Modbus tarjoaa virhetarkistuksen. Viestintävirheiden havainnointiin käytetään merkkikehitystä, pariteettitarkistusta, redundanssitarkistusta tai kuudentoista bitin syklistä redundanssitarkistusta (CRC-16). Kapseloimalla Modbus-tiedot TCP/IP-pakettiin voidaan Modbus-paketteja lähettää lähi- ja suuralueverkkojen yli. (Mackay, 2004, s. 97–99)

Modbus Plus on teollisuuden ohjaussovelluksien lähiverkkojärjestelmä. Verkotetut laitteet voivat vaihtaa viestejä prosessien ohjauksesta ja valvonnasta. Modbus Plus-protokollatopografia mahdollistaa jopa 64:n laitteen verkkosegmentin. Yksilöllinen verkko-osoite 01-64 välillä on määritettävä jokaiselle verkkosegmentin laitteelle. Viesti välitetään laitteelta toiselle viestin sisällä olevan reitin kautta. Reitti sisältää kohdelaitteen verkko-osoitteen ja verkkojen väliset tarvittavat osoitteet. (Mackay, 2004, s. 97–99)

## 7.1 Modbus protokolla

Kuvassa 4 on esitetty tyypillinen Modbus-viestikehyksen muoto.

Kuva 4. Modbus viestikehyksen muoto (Mackay, 2004, s. 98)

Address field	Function field	Data field	Error check field
1 byte	1 byte	Variable	2 bytes

Kuvassa 4 oleva ensimmäinen kenttä on osoitekenttä (address field), joka koostuu yhdestä tietotavusta. Tavu identifioi pyyntökehyksissä ohjaimen, jolle pyyntö on osoitettu. Vastaava vastauskehys alkaa vastaavan laitteen osoitteella. Palvelimen osoitekenttä voi olla välillä 1–247. Käytännön rajoitukset rajoittavat enimmäismäärää palvelimissa. Viestin toinen kenttä on funktiokenttä, joka koostuu yhdestä tietotavusta. Asiakas pyynnössä tämä tavu identifioi toiminnon. Kohteena olevan ohjelmoitavan logiikan tulee suorittaa tämä toiminto. Pystyessään suorittamaan pyydetyn toiminnon, vastauksen toimintokenttä (function field)

toistaa alkuperäisen pyynnön. Tietokenttä (data field) on kolmas kenttä viestikehyksessä. Tietokentän pituus vaihtelee sen mukaan, mikä toiminto toimintokentässä on määritelty. Asiakaspyynnössä tietokenttä sisältää tiedot, joita ohjelmoitava logiikka tarvitsee suorittaakseen pyydetyn toiminnon. Ohjelmoitavan logiikan vastauksessa tietokenttä sisältää kaikki kyseisen asiakkaan pyytämät tiedot. Virheentarkistuskenttä (Error check field) sisältää kaksi viimeistä tavua sanomakehyksessä. Virhetarkistus varmistaa, että laitteet reagoivat viesteihin, jotka ovat vaurioituneet. (Mackay, 2004, s. 97–99)

## 7.2 Modbus toimintokodeista

Tietotyyppien osoitealueet ja poikkeamat sekä toimintokoodit on esitetty taulukossa 1. Toimintokoodi määrittää kohdeohjaimelle odotetun toiminnon. Pyyntötietokenttien merkitys riippuu määritetystä toimintokoodista. (Mackay, 2004, s. 98)

Taulukko 1 Modbustoimintokoodit (Mackay, 2004, s. 98)

Data type	Absolute addresses	Relative addresses	Function codes	Description
Coils	00001 to 09999	0 to 9998	01	Read coil status
Coils	00001 to 09999	0 to 9998	05	Force single coil
Coils	00001 to 09999	0 to 9998	15	Force multiple coils
Discrete inputs	10001 to 19999	0 to 9998	02	Read input status
Input registers	30001 to 39999	0 to 9998	04	Read input registers
Holding registers	40001 to 49999	0 to 9998	03	Read holding register
Holding registers	40001 to 49999	0 to 9998	06	Preset single register
Holding registers	40001 to 49999	0 to 9998	16	Preset multiple registers
–	–	–	07	Read exception status
–	–	–	08	Loopback diagnostic test

Read coil or digital outputstatus (code 01) toiminnolla asiakas voi saada kohdelaitteen yhden tai useamman logiikkakelan On/Off- tilaan. Pyyntön tietokenttä koostuu ensimmäisen kelan suhteellisesta osoitteesta ja sen jälkeen luettavien kelojen lukumäärästä. Vastauskehys koostuu tietokentän kelatavujen määrästä. Tätä seuraa useat tavut keladataa. (Mackay, 2004, s. 98)

Read digital input status (code 02) toiminnolla asiakas voi lukea yhden tai useamman erillisen tulon kohdelaitteessa. Read holding registers (code 03) toiminnolla asiakas voi hankkia kohdelaitteen yhden tai useamman pitorekisterin sisällön. Tämän toiminnon avulla isäntä voi hankkia kohdelaitteen yhden tai useamman pitorekisterin sisällön.

Hallintarekisterin suhteellisesta osoitteesta koostuu pyyntökehyksen tietokenttä.

Rekisteridatavujen määrästä koostuu vastausajan tietokenttä. Rekisteridatavuja seuraa pitorekisterin monet tavut. Pyydettyjen rekisterien sisältö (16 bittiä) palautetaan kahdessa peräkkäisessä datatavussa (merkittävin tavu ensin). (Mackay, 2004, s. 98–99)

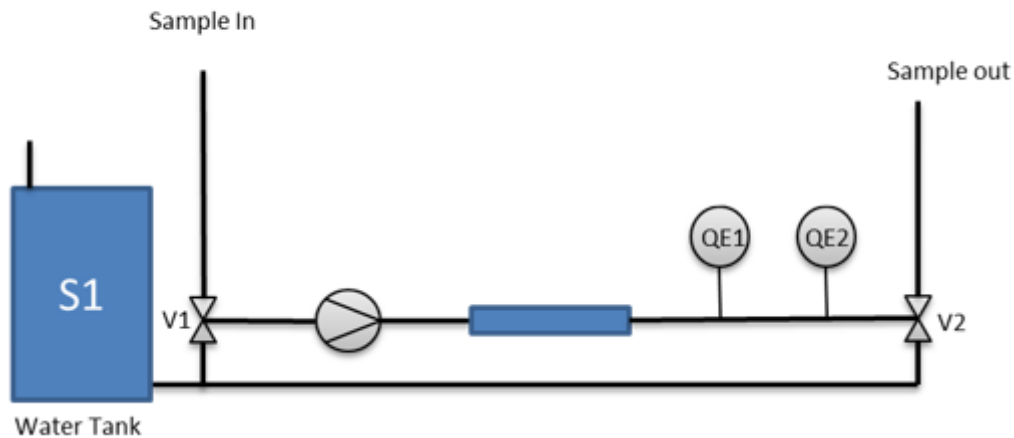
### **7.3 Modbus ongelmat**

Modbus-ongelmat jakaantuvat laitteisto- tai ohjelmisto ongelmiin. Laitteisto-ongelmiin kuuluvat kaapelointivirheet ja vialliset tiedonsiirtoliittymät. Ohjelmisto-ongelmiin kuuluvat muun muassa asiakkaan yritys päästä käsiksi olemattomiin solmuihin tai käyttää virheellisiä toimintokodeja, olemattomien muistipaikkojen osoitus palvelimella tai laittomien tietomuototyyppien määrittäminen. Jos isäntäsovellus yrittää päästä käsiksi olemattomiin solmuihin tai käyttää virheellisiä toimintokodeja, syntyy ohjelmistoprotokollaan liittyviä ongelmia. (Mackay, 2004, s. 109)

## **8 Mittalaitteisto**

Työssä käsiteltävä mittalaitteisto on liikuteltava offline-mittalaitteisto. Laitetta käytetään teollisuudessa olevan online-mittalaitteiston tulosten ratifiointiin ja teknisen kehityksen tukena. Laitteisto on esitetty PID-kaaviossa kuvassa 5.

Kuva 5. Mittalaitteiston PID-kaavio



Kuvassa 5 esitettyssä laitteistossa kierrätetään vesifaasissa olevaa näytettä kahden anturin lävitse. Tässä työssä tarkastelun alla on kuvassa 5 esitetty anturi QE2. Työssäni käytän nimitystä vanha anturi, jolla tarkoitan vanhaa myynnistä poistunutta anturia ja uusi anturi, joka on korvaava, uusi anturi. Muita toimilaitteita ovat pumppu, kolmitieventtiilit V1 ja V2 sekä anturi QE1. Laitteisto toimii esiohjatulla sekvenssillä tai manuaalisesti ajaen. Mittaukset ovat avoimia mittauksia, eli ne eivät ohjaa laitteiston muita toimilaitteita.

Uuden anturin kytkentäkaavio on esitetty liitteessä 1. Uuden anturin käyttöohjeesta saadaan tarvittavat arvot anturin arvojen lukemiseen Modbus-väylällä. Uuden anturin toimivuutta pystyttiin testaamaan anturin omalla verkosta löytyvällä konfiguraatio-ohjelmalla. Uuden anturin omasta ohjelmasta laitettiin anturi jatkuvaan mittaustilaan sekä määritettiin mittaussekvenssi anturille.

Jotta mittaukset voitiin tehdä luotettavasti samoilla näytteillä, asennettiin uusi ja vanha anturi sarjaan mittalaitteeseen. Mittauskyvetit, joissa anturit ovat, asennettiin siten, että kyvettien tilavuutta pystyttiin muuttamaan. Näytteitä pystyttiin tarvittaessa ajamaan muiden mittausten ja laitteiston linjojen ohitse.

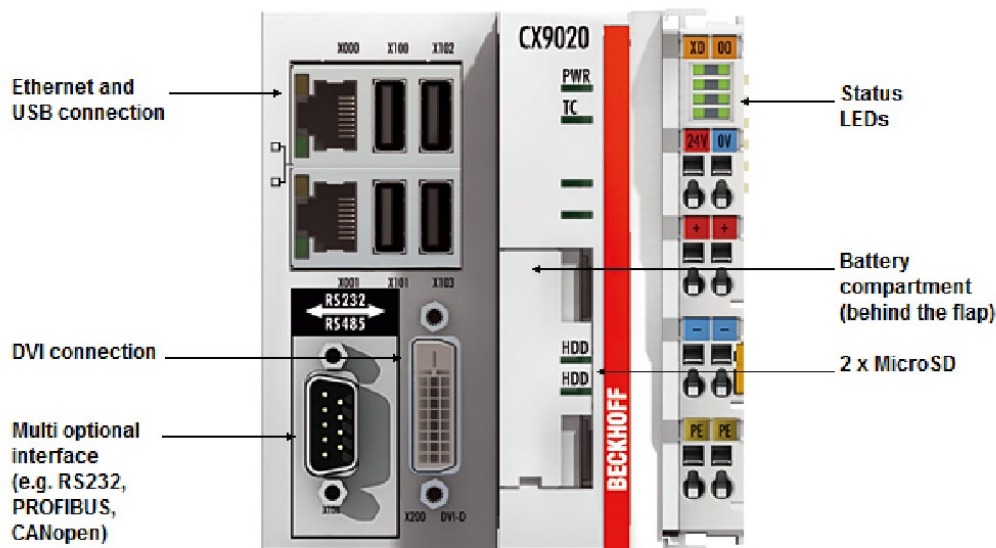
## 9 Mittalaitteiston PLC

Mittalaitteiston ohjelmointi ympäristö on TwinCat ja laitteisto koostuu Beckhoffin PLC-moduulista ja lisäkorteista.

### 9.1 PLC- topologia

Laitteiston ohjelmoitavana logiikkana on käytössä CX9020-N031 moduuli, josta on eri versioita. Moduuli on varustettu Ethernet-ohjausjärjestelmällä. I/O- järjestelmän integraatio tapahtuu suoraan CPU-moduuliin. CPU-moduulissa on paikat kahdelle MicroSD muistikortille, kahdelle Ethernet RJ45-käyttöliittymälle, neljälle USB 2.0:lle, DVI-D -paikalle ja yhdelle RS422/RS485- liittimelle. Kuvassa 6 on esitetty laitteistossa oleva CPU-moduuli.

Kuva 6. CPU- moduuli 9020 (Beckhoff, 2021)



CPU-moduuliin on linkitetty erilaisia moduuleita mittalaitteiston eri laitetarpeita varten. Alla on esitelty erilaiset moduulit, joita laitteessa on käytössä ja niiden toiminnot.

Moduulit EL9210 ja EL9200 mahdollistavat erilaisten potentiaaliryhmien asettamisen 24V tasavirran vakiojännitteellä. Moduuli EL9210 on varustettu sulakkeella ja diagnostiikalla, kun

moduuli EL9200 on varustettu pelkästään sulakkeella. Moduuleihin syöttöjännite otetaan vastaan etuliittimen kautta. Sisäisten tehokoskettimien kautta jännite johdetaan viereisiin liittimiin. (Beckhoff.fi, haettu 6.9.2022)

EL9187-moduuli on potentiaalinjakoliitin, joka tarjoaa 8 liitinkohtaa 0 voltin potentiaaliin. Moduuli EL9186 on potentiaalinjakoliitin, jossa on 8 liitinkohtaa 24 voltin potentiaaliin. EL9187- ja EL9186-moduuleilla voidaan vähentää ylimääräisten riviliittimien käyttöä riviliittimessä. (Beckhoff.fi, haettu 6.9.2022)

Moduuli EL4102 on analoginen lähtöliitin, joka tuottaa signaaleja 0–10 voltin alueella. Sähköisesti eristetty jännite syötetään prosessitasolle 16 bitin resoluutiolla. EtherCAT-terminaalin lähtökanavilla on yhteinen maadoitus. EL4102 moduulissa yhdistyy kaksi kanavaa yhden kotelon sisään. Signaalin tila EtherCAT-päätteessä ilmaistaan valodiodeilla. (Beckhoff.fi, haettu 6.9.2022)

Moduuli EL3161 on analoginen tuloliitin, joka käsittelee signaaleja 0–10 voltin alueella. Jännite digitoidaan 16:sta bitin resoluutioon ja siirretään korkeamman tason automaatiolaitteeseen. (Beckhoff.fi, haettu 6.9.2022)

Moduuli EL2004 on digitaalinen lähtöliitin. Moduuli yhdistää automaatiolaitteen binaariset 24 voltin tasajännite ohjaussignaalit toimilaitteille.

Analoginen tuloliitin EL3122 käsittelee signaaleja 4–20 mA:n alueella. Virta digitoidaan 16:sta bitin resoluutioon ja siirretään korkeamman tason automaatiolaitteeseen sähköisesti. Kaksi kanavaa on yhdistetty samaan koteloon. EtherCAT-väylän avulla päätteen tila välitetään säätimelle. Moduulissa on indikaattorit ylikuormitustilan ja katkenneen johdon osoittamiseen. (Beckhoff.fi, haettu 6.9.2022)

Moduuli EL2602 on lähtöliitin, jossa on kaksi relettä. Jokaisessa on yksi kosketin. Relekosketin kytketään tehokoskettimiin. Nämä soveltuvat käytettäväksi 230 voltin vaihtovirtaan asti ja joita voidaan käyttää verkkovirtaa vaativien laitteiden kytkemiseen. (Beckhoff.fi, haettu 6.9.2022)

## 9.2 TwinCAT3 ja ohjelmistokieli

Ohjelmisto on ohjelmoitu TwinCAT 3:lla. TwinCAT on avoin PC-pohjainen ohjaustekniikka. TwinCAT tarkoittaa The Windows Control and Automation Technologya ja automaattiosarja muodostaa ohjausjärjestelmän ytimen. TwinCAT- ohjelmistojärjestelmä muuttaa lähes kaikki PC-pohjaiset järjestelmät reaaliaikaiseksi ohjaukseksi. TwinCAT 3:n etuihin kuuluvat yksi ohjelma kehitykselle ja ohjelmoinnille, Visual Studio integraatio ja vapaus valita ohjelmointikieli. (Beckhoff.fi, n.d)

Ohjelmistossa valittavia kieliä ovat STD ja FBD. FBD (function block diagram) kieltä kuvataan PLC-ohjelmissa graafiseksi kieleksi. Siinä tieto kulkee laatikoiden läpi. Laatikoida voidaan käyttää uudelleen. Toimilohko, eli function block, on ohjelmointiyksikkö, joka toteutettuna antaa yhden tai useamman lähtöarvon. Standardi IEC 113-3 määrittää laatikon ulkomuodon. Toimilohko laatikossa tulosignaalit tulevat vasemmalta ja lähtösignaalit lähtevät oikealle. Itse laatikon toiminto on kirjoitettuna laatikkoon. (Bolton, 2006, s. 94)

Toinen TwinCAT-ohjelmassa käytetty kieli on jäsennetty teksti (ST eli Structured text). Se muistuttaa PASCAL-ohjelmointikieltä. Ohjelma kirjoitetaan lausuntojen sarjana, joita puolipiste erottaa. Lausunnot käyttävät ennalta määrättyjä lausuntoja ja aliohjelmiä muuttujia muuttaakseen. Nämä muuttujat ovat ennalta määritettyjä arvoja, sisäisen muistin arvoja tai tuloja tai lähtöjä. Toimeksiantolausuntoja käytetään osoittamaan miten muuttujan pitäisi muuttua. (Bolton, 2006, s. 121)

TwinCAT 3 mahdollistaa ohjelmoinnin STD tai FBD kielellä. Laitteiston koodia on kirjoitettu sekä STD:llä ja FBD:llä. Suurin osa koodista on STD:tä, joten valitsimme päivitykseen STD:n.

## 10 Työn toteutus

Työn tarkoituksena oli verrata nykyistä vanhaa anturia uuteen anturiin ja selvittää saadaanko antureilta sama vaste pienillä määritystasoilla. Työ koostui kolmesta osasta; Selvityksestä laitteiston muista päivitystarpeista käyttäjiltä, itse anturin konfiguroinnista ohjelmistoon ja laitteistoon ja anturin testaamisesta sekä tarvittavan dokumentaation päivityksestä. Anturin



arvot piti saada kirjoittamaan myös tallennettavaan tietokantaan ja anturin arvon piti näkyä reaaliaikaisena näytössä. Työn edetessä tutustuin alkuperäiseen ohjelmaan ja sen toimilohkoihin.

### **10.1 Laitteiston päivitys ja käyttäjä palaute**

Laitteiston päivitystarve johtuu entisen anturin valmistuksen lopetuksesta. Uuden anturin pitää tuottaa yhtä hyvää vertailukelpoista tietoa, jota vanha anturi tuottaa. Laitteiston päivitystä varten kartoitettiin myös muut tarpeet, jos laitteistoa olisi tarve kehittää vielä tämän työn ulkopuolelta.

Laitteisto on ollut operatiivisessa käytössä jo hetken aikaa ja laitteiston toimivuudesta saatiin kiitettävästi palautetta. Osa palautteesta oli lähinnä laitteiston fyysiseen käyttöön liittyviä, mutta palautteesta löytyi myös itse ohjelmaan liittyviä ongelmia. Laitteiston käyttöliittymässä ongelmaksi koettiin tallennuksen ylikirjoitus, eli jos ohjelmassa haluaa tallentaa ajetun näytteen tiedon ja unohtaa kirjoittaa uuden nimikkeen näytteelle, tallentaa ohjelma uuden ajon jo ajetun ajon päälle.

Toinen ongelma koettiin tiedon käsittelyssä, kun ajetun ohjelman tiedot pitää siirtää muistitikulla toiselle koneelle ja ajaa analyysiohjelmaan. Toiveena oli langatonta tiedonsiirtoa, mutta wifi-yhteys on käytössä laitteiston ohjelmoitavan logiikan ja ohjaustietokoneen välisessä yhteydessä. Tähän on olemassa optio liittää Ewon yksikkö väliin, jolloin kaikki tieto saataisiin siirrettyä pilvipalveluun suoraan.

Kolmas ongelma koski laitteiston magneettiventtiileitä ja letkuja. Letkujen materiaali ei ole optimaalinen, jos letkut jäävät venttiilien tai pumppupään sisään puristuksiin. Käytössä olevat kolmitieventtiilit myös kuumenevat käytössä. Jos laitteistoa ja vaihdettavia osia haluttaisiin vähentää voisi osan linjoista tehdä kiinteämmiksi sekä vaihtaa nykyiset kolmitieventtiilit toisenlaisiin venttiileihin.

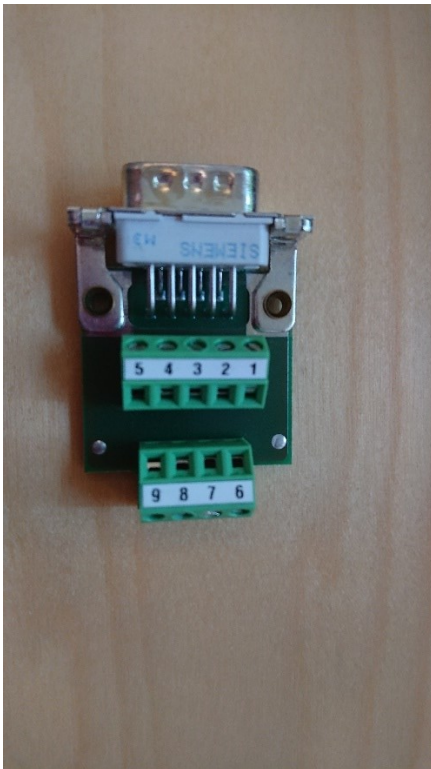
## 10.2 Anturin konfiguraatio

Uusi anturi toimii väyläliikenteellä, kun edellinen vanha anturi toimi analogiatulolla.

Käytettävä Beckhoff PLC CX9080 on hankittu RS232/485 sarjaportilla. Tämä mahdollistaa anturin kytkemisen suoraan väylään ilman erillistä RS232/485 moduulia.

Anturin kytkentäkaavio on esitetty liitteessä 1. Kuvassa 7 on esitetty D9-liitin, jolla anturin kaapelit kytkettiin Beckhoffin RS-485 porttiin.

Kuva 7. Liitin



Anturin kaapelit kytkettiin kuvassa 7 olevaan liittimeen liitteen 1 mukaan. Anturin virta otetaan Beckhoffin 24 V liittimistä moduuleilta EL9210 ja EL9200.

Sarjaportti haetaan laitetopologiaan manuaalisesti. Porttia ei pysty skannaamaan, koska se ei ole erillinen moduuli Beckhoff-topologiassa. Manuaalisessa lisäyksessä sarjaportti pitää hakea Miscellaneous -osion alta Serial Porttina. Portin asetusarvot tulee synkronoida anturin asetusarvojen kanssa.

Asetusarvojen synkronointi ei ollut odotetun yksinkertaista. Jos olisimme käyttäneet erillistä Beckhoffin sarjaporttimoduulia, olisimme mahdollisesti voineet tehdä automaattisen parametriskannauksen. Koska käytössä oli tuo ohjelmoitavan logiikan sarjaportti, piti haku tehdä manuaalisesti. Yhteyksien kartoittaminen osoittautui ongelmallisemmaksi kuin kukaan oli ennakoanut. Jouduimme poistamaan ja linkittämään useaan otteeseen arvot, jotta aloimme saada tietoliikennettä anturista.

### **10.2.1 Modbus-funktio**

Ohjelma kirjoitettiin jäsennettynä tekstinä eli ST:na. Ohjelmaan määritettiin muuttujat muuttujalistaan (VAR inputs / outputs).

Modbus-funktiona käytetään RTU Read funktiota eli lukufunktiota, koska tarve on anturin lukemiseen. Modbus-väylän toteutukseen Beckhoff järjestelmässä on haettava erillinen kirjasto. Tc2\_modbusRTU kirjaston avulla pystytään luomaan Modbus-funktiot, standardin mukaan. Funktiota voidaan käyttää joko FBD tai STD versiona, koska nykyinen koodi on pääosin tehty STD-kielellä. Read-funktio eli luku-funktio on esitetty kuvassa 8.

Kuva 8. Modbus RTU.readRegs- funktio

```

1 ModbusRTU.ReadRegs (
2     UnitID:= 5,
3     Quantity:= 8,
4     MAddr:=1500, // tätä arvoa muuttamalla voi lukea haluttua 998 temp, 1000 conser
5     cbLength:=sizeof (aRead),
6     pMemoryAddr:=ADR(aRead),
7     Execute:= Execute FALSE,
8     Timeout:= T#5S,
9     BUSY =>,
10    Error =>,
11    ErrorId =>,
12    cbRead => );
13
14    Execute FALSE R=ModbusRTU.BUSY TRUE OR ModbusRTU.Error FALSE;
15    Execute FALSE S=NOT ModbusRTU.BUSY TRUE AND NOT Execute FALSE;
16
17    //Linkitetään Modbussista luettu WORD-data.
18    //Paikat [1] ja [2] vaihdettava.
19
20    Float.aWord[2] 16661 :=aRead[1] 16661;
21    Float.aWord[1] 45466 :=aRead[2] 45466;
22
23    //Mitta-alue realina ja desimaalina
24
25    fSensor 9.36 :=Float.fReal 9.36;
26    RETURN

```

Funktion käyttämiseen tarvitsimme useampaa arvoa. Anturilla on tunnistenumero, joka löytyy anturin ohjeesta ja tämä tunnistenumero pitää laittaa unitID kohtaan. Quantity kohdassa määritetään lähtörekisterien lukumäärä. Arvo löytyy laitteen käyttöohjeesta, mutta arvon oltava isompi kuin 0. MAddr kohtaan kirjoitetaan luettavien lähtörekisterien osoite. Arvo löytyy käyttöohjeesta ja tähän koodiin on kommentoitu arvot, joita voidaan käyttää tässä arvona.

Anturi antaa arvot WORDinä, joten käytimme funktiota, jolla arvon saa lukuarvoksi. Ohjelmaan piti kirjoittaa rivi sille, että luettu 2 x 16 bittinen WORD- data käännetään reaalityyppiseksi, että tavujen paikat ovat oikeilla kohdilla. Tälle käänökselle piti tehdä DUT (Data type unit) eli yksikön käänös taulukoitavaan muotoon.

Laitelinkitykset tehdään TwinCAT ohjelman PLC mappings osiossa, jotta data array saadaan näkyviin pitää valita all types -valintaruutu aktiiviseksi. Laitelinkitysten jälkeen anturia voidaan alkaa lukea.

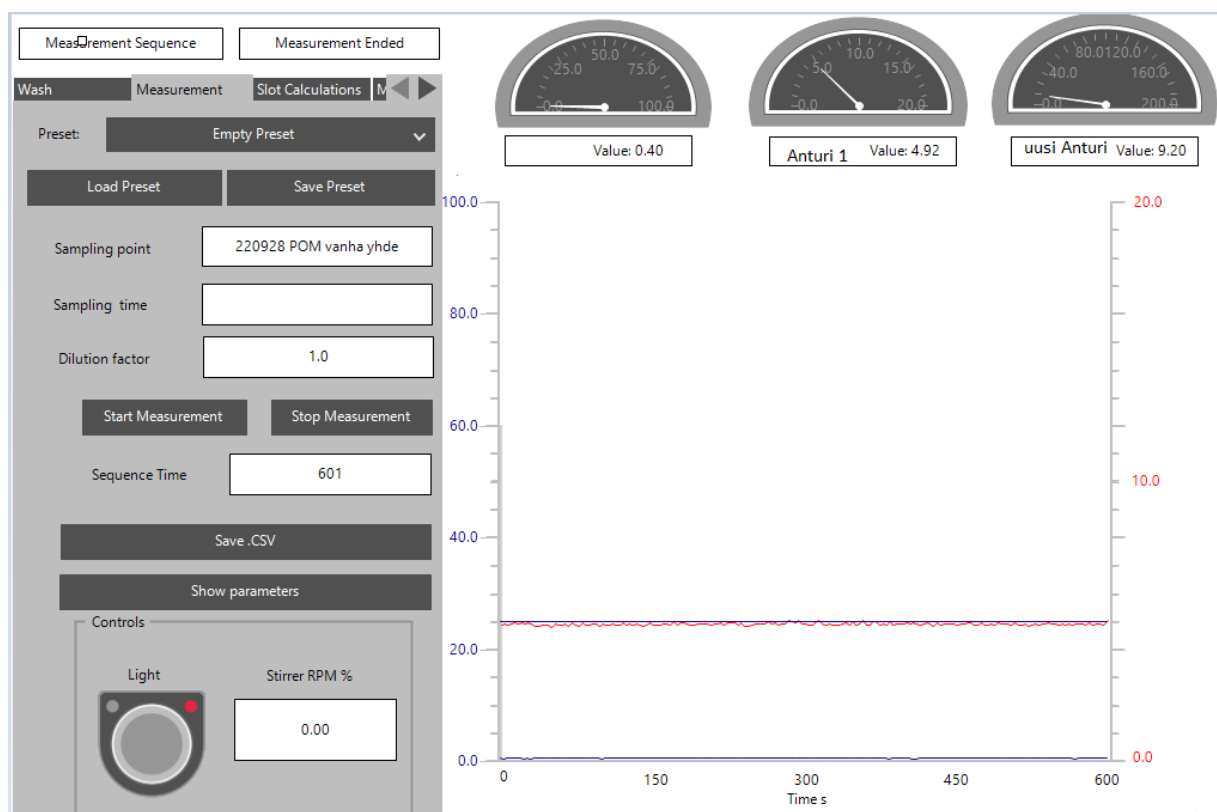
Anturin tuottama tieto piti lisätä taulukkoon, jota ohjelma luo. Tätä varten piti luoda muuttujat. Muuttujat luotiin globaaliin muuttujalistaan. Muuttujien nimeämisessä piti miettiä miten saadaan selkeästi eroteltua alkuperäinen vanha anturi uudesta anturista.

### 10.2.2 Anturin näkyminen käyttöliittymässä

Anturi laitettiin käyttöliittymään näkyviin, koska vertailuanalyysjä tehtäessä on helpompi suoraan nähdä arvo näytöltä. Näytteen läpimenoaika on noin puoli tuntia, joten arvon näkeminen näytöltä helpottaa heti näkemään eroavaisuutta vanhaan anturiin.

Käyttöliittymään piti lisätä indikaattori, joka kytkettiin muuttujiin. Kuvassa 9 näkyy päivitetty käyttöliittymä.

Kuva 9. Käyttöliittymän päivitys



Kuvassa 9 näkyy käyttöliittymään mittaristo, joka reagoi reaaliajassa näytteen muutoksiin. Anturin tuottamat arvot tallentuvat myös taulukkoon, joka tallentuu ohjelmoitavan logiikan omalle muistikortille.

Tallennetut arvot voidaan hakea ohjelmoitavan logiikan omalta kovalevyllä. Kovalevyn kytkentään piti muokata Windowsin omia SMB-jakoarvoja, jotta näkyvyys levyille oli mahdollista. Työssä myös opittiin, että yrityksen tietokonepäivitysprotokollan myötä SMB-jako aina tippuu pois päältä ja se pitää asentaa uudelleen.

## 11 Anturin testaus

Anturia testattiin anturin kalibrointiin ja tarkistukseen tarkoitetulla liuksella. Liuos tehtiin sekoittamalla reagenssia milli-q veteen. Varastoliuoksen väkevyys oli 80 mg/l. Antureita mitattiin kahdella eri pesäsyvyydellä. Nykyinen upotussyvyys vanhalla anturilla on 5 cm, mutta uuden anturin suositussyvyys on 10 cm. Vertailumittaukset suoritettiin kahdella eri pesäsyvyydellä. Varastoliuoksesta laimennettiin käyttöliuos tarkistusmittauksia varten.

Taulukossa 2 on esitetty käyttöliuosväkevyudet.

Taulukko 2 Tarkoitusliuoksen väkevyys (Haarala, 2022)

Väkevyys µg/l
1
2
5
10
15
20
100

Uusi ja vanha anturi asennettiin laitteiston linjaan sarjaan, jonka johdosta oli mahdollista ajaa sama näyte molempien antureiden lävitse. Tarkistusliuosta kierrätettiin järjestelmässä niin kauan, että anturin arvo vakiintui. Jokaisen koepisteen välissä linjasto ja anturipesät tyhjennettiin ja huuhdeltiin.

Taulukosta 3 näkee anturin vertailuarvot 5 cm syvyydellä

Taulukko 3 Vertailuarvot 5 cm upotuksella

Syvyys /cm	Liuos $\mu\text{g/l}$	Vanha anturi	Uusi anturi	stab aika	Kommentti
5	0	3,06	12,22		tyhjäpesä
5	100	20,1	200	2 min/heti	
5	0	3,5	0		pesu
5	50	20,09	99,87	2 min/heti	
5	0	3,51	0		pesu
5	10	20,13	9,52	2 min/heti	
5	0	3,09	0		pesu
5	1	4,23	0	2 min/heti	

Vertailu mittaus tehtiin 10 cm upotuksella. Tulokset on esitetty taulukossa 4.

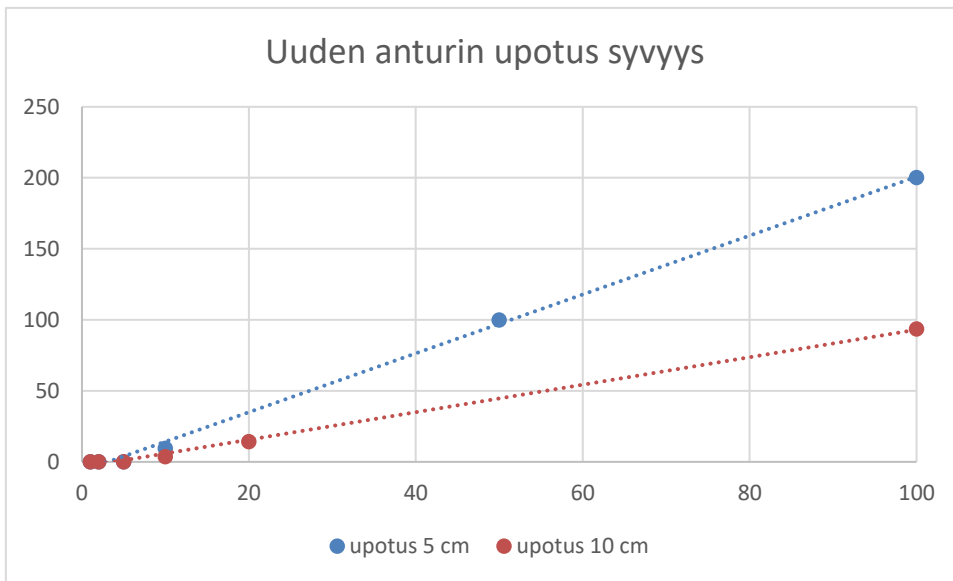
Taulukko 4 10 cm upotus

Syvyys /cm	Liuos $\mu\text{g/l}$	Vanha anturi	Uusi anturi	stab aika	Kommentti
10	0	3,06	21.81		tyhjäpesä
10	100	20,13	93,6	15 min	
10	0	3,24	0		pesu
10	20	20,10	14,07	2 min/heti	
10	0	3,12	0		pesu
10	15	20,1	8,05	2 min/heti	
10	0	3,17	0		pesu
10	10	15,05	3,43	2 min/heti	
10	0	3,2	0		pesu
10	5	8,9	0		
10	0	3,17	0		pesu
10	2	5,19	0	2 min/heti	
10	0	3,18	0		pesu
10	1	4,04	0		

Mittauksista on nähtävissä, että upotus vaikuttaa uuden anturin tarkistuskäyrän jyrkkyyteen.

Kuvassa 10 on esitetty kuvaajana uuden anturin eri syvyydellä mitatut arvot.

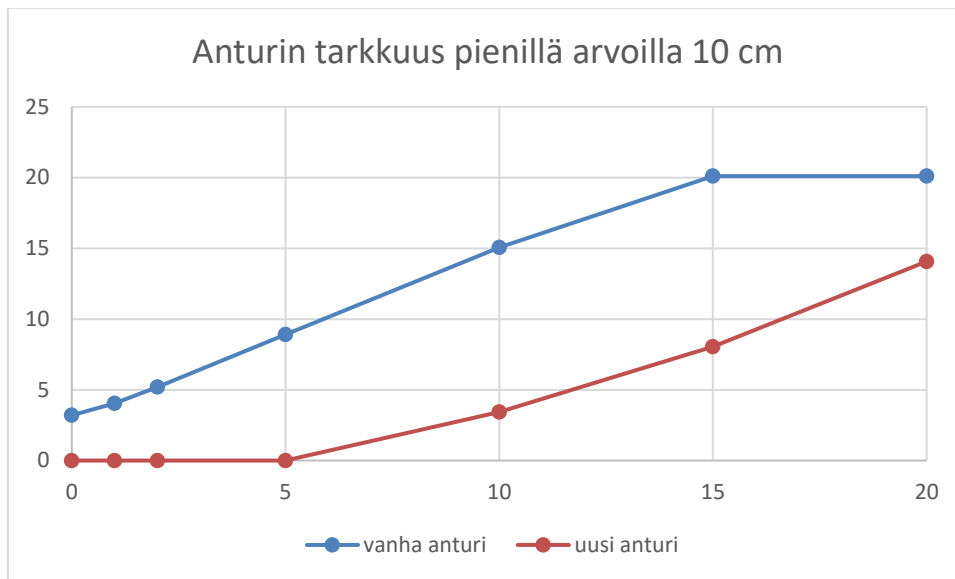
Kuva 10. Uuden anturin upotussyvyys tarkistusliuoksilla



Kuvan 10 kuvaajasta on selkeästi nähtävissä, että kalibrointi on lineaarinen molemmissa tapauksissa. Ongelmaksi muodostuu pienten arvojen tarkkuus. Kuvassa 11 on esitetty anturin syvyyden vaikutus pienissä pitoisuuksissa. Vertailussa on käytetty 10 cm:n upotusta vanhalla ja uudella anturilla, koska lineaarisuus oli tässä syvyydessä parempi tarkistusliuoksen pitoisuuteen verrattuna.



Kuva 11. Anturin tarkkuus pienillä arvoilla



Tuloksista on nähtävissä, että alle 5  $\mu\text{g/l}$  tarkkuudella uusi anturi ei anna arvoja. Kuvan 11 kuvaajasta on nähtävissä, että kalibrointi on lineaarinen. Ongelmaksi muodostuu, että millä alueella anturi on lineaarinen. Uuden anturin herkkyys on heikompi pienemmissä tarkistusliuoksen pitoisuuksissa verrattuna vanhaan anturiin. Valmistajan mukaan anturin pitäisi olla tarkempi pienillä mittausalueilla, joten tämän todettuamme uudet anturit lähetettiin uudelleen kalibrointiin tehtaalle.

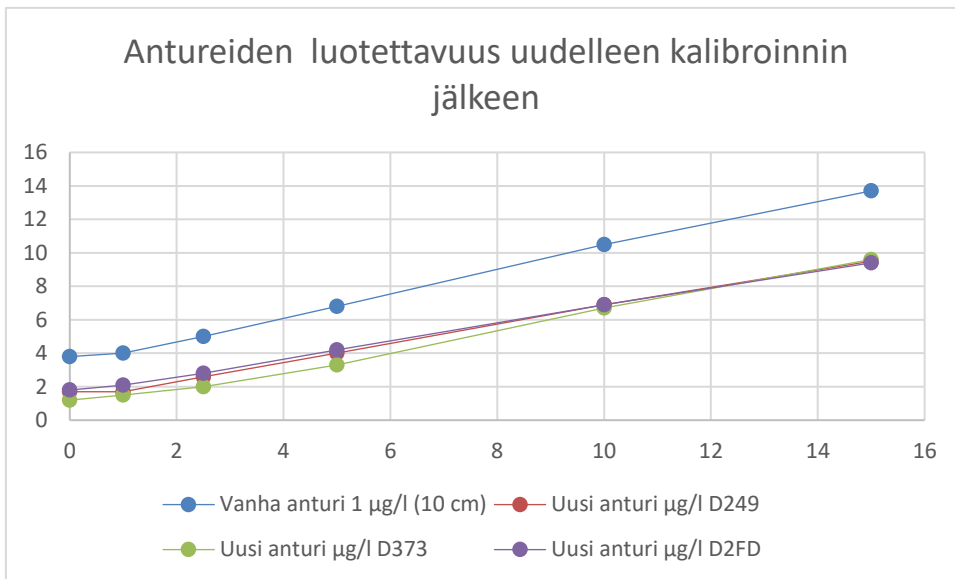
Kalibrointiin lähetettiin useampi anturi. Uudelleen kalibroinnin jälkeen tulokset on esitetty taulukossa 5.

Taulukko 5 Uudelleen kalibroinnin jälkeiset tulokset

Reagenssi $\mu\text{g/l}$	Vanha anturi 1 $\mu\text{g/l}$ (10 cm)	Uusi anturi $\mu\text{g/l}$ D249	Uusi anturi $\mu\text{g/l}$ D373	Uusi anturi $\mu\text{g/l}$ D2FD
0	3,8	1,7	1,2	1,8
1	4	1,7	1,5	2,1
2,5	5	2,6	2	2,8
5	6,8	4	3,3	4,2
10	10,5	6,9	6,7	6,9
15	13,7	9,5	9,6	9,4

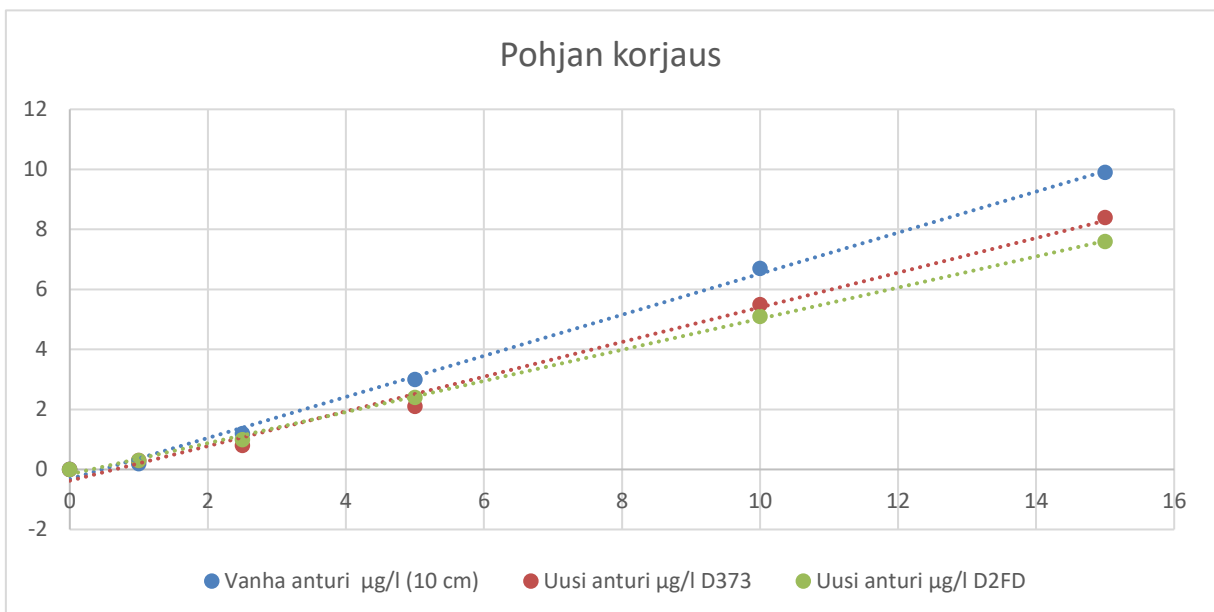
Taulukon 5 pohjalta on tehty kuvaaja, joka on esitetty kuvassa 12.

Kuva 11. Tulokset uudelleen kalibroinnin jälkeen



Kuvan 12 kuvaajasta on selkeästi nähtävissä, että uudelleen kalibrointi auttoi antureiden pienten pitoisuuksien tunnistamisherkyyteen. Anturit korreloivat hyvin vanhaan anturiin, kun antureille tehdään pohjatasokorjaus. Pohjatasokorjaus tehtynä suorat näyttävät lähes samalta ja antavat saman arvon. Kuvan 13 olevassa kuvaajassa voi nähdä miten pohjatasokorjaus vaikuttaa suoriin.

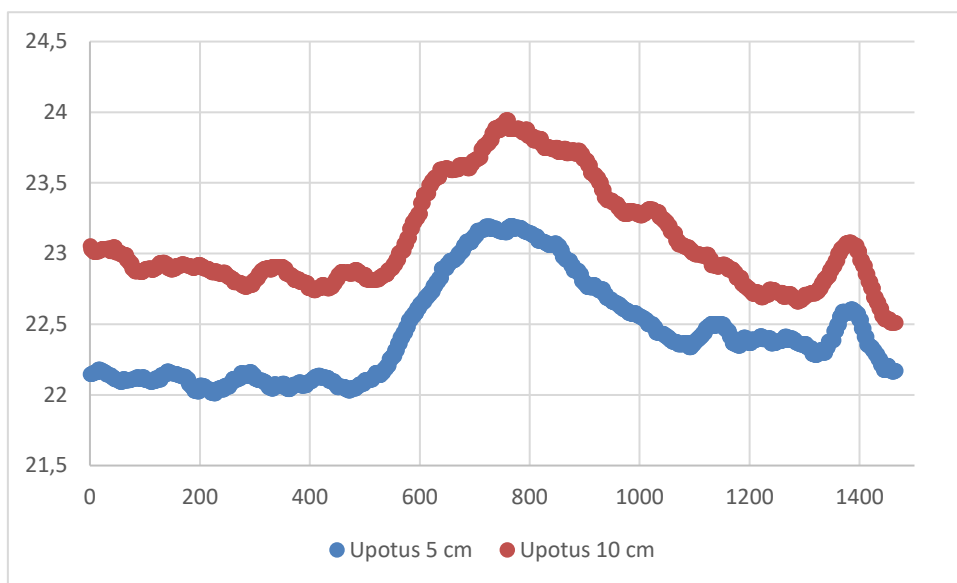
Kuva 12. Pohjatasokorjaus



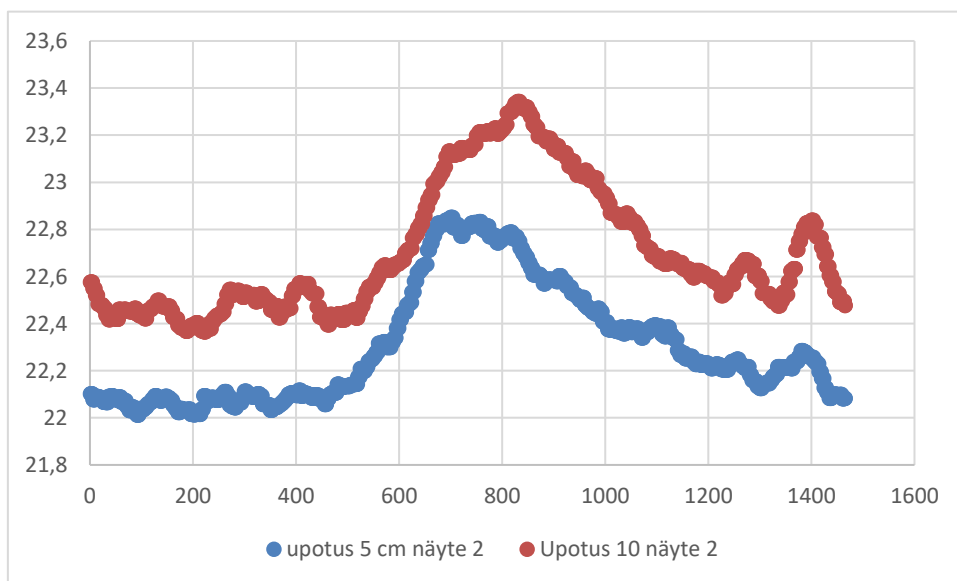
Kuvasta 13 nähdään, että pohjatasokorjaus siirtää arvot lähemmäs toisiaan. Tämän avulla pienemmillä arvoilla tehtävissä mittauksissa vanha ja uusi anturi korreloivat hyvin yhteen. On myös havaittavissa, että riippumatta käytettävästä anturista ovat suorat lähes samat samoissa pitoisuuksissa, kun pohjatasokorjaus on otettu huomioon.

Oikeilla näytteillä tehdyssä mittauksissa pystytään näkemään, että upotussyvyydellä on vaikutus anturin pohjatason korkeuteen. Kuvissa 14 ja 15 on esitetty kahdesta eri näytteestä tehdyt ajot kahdella eri upotussyvyydellä.

Kuva 13. Näyte 1 upotus syvyyden vaikutus

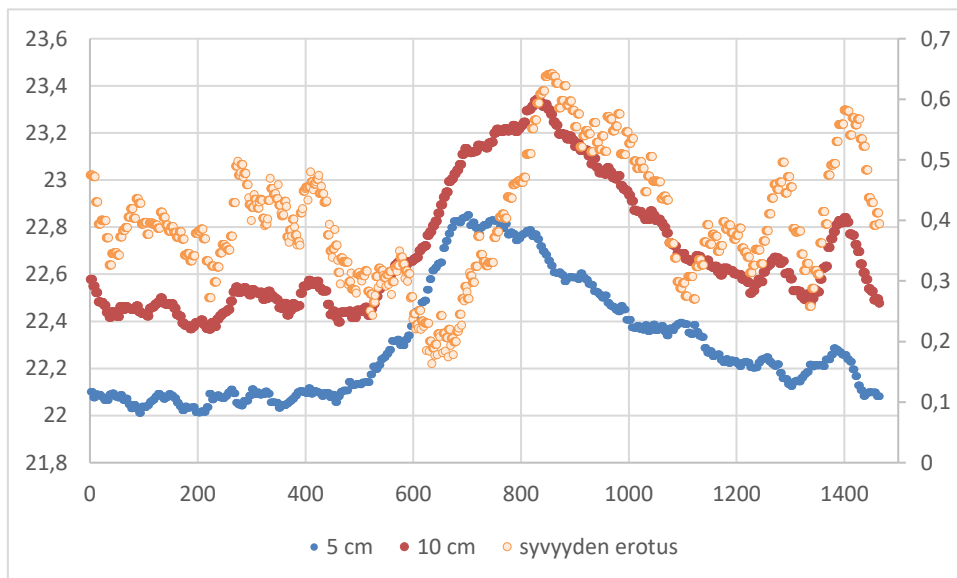


Kuva 14. Näyte 2 upotus syvyyden vaikutus



Kuvissa 14 ja 15 nähtävät oikeilla näytteillä tehdyt mittaukset kertovat sen, että uusi ja vanha anturi mittaavat samoja näytteitä samalla tavalla. Pohjan taso on vain erikohdalla riippuen anturikyvetin syvyydestä. Kuvassa 16 on esitetty anturimittausten erotus sekundaarisella akselilla.

Kuva 15. Anturikyvetin syvyyden erotus



Kuvasta 16 näkee, että anturien syvyyden aiheuttama tarkkuusero on keskiarvoltaan 0,4. Kun pitoisuus näytteessä alkaa kasvaa, myös tarkkuusero kasvaa maksimissaan 0,24 yksiköllä 0,64. Minimissään erotus on 0,16 yksikköä. Koska mitattu pohjataso on 22 yksikön luokassa, prosentuaalinen syvyyden aiheuttama virhe on keskiarvoisesti noin 1,8 %.

## 12 Loppupäätelmä

Työ oli onnistunut. Uusi anturi tuottaa vertailukelpoista tietoa verrattuna vanhaan anturiin. Tämän seurauksena vanhoja antureita voidaan korvata luotettavasti uudella anturimallilla.

Anturin konfigurointi oli haastavampaa kuin aluksi ajateltiin. Ongelmat alkoivat kastuneen koneen sisään jääneestä versionhallinnasta. Versionhallintaa tulee lisätä ja läpinäkyvyyttä siitä missä ohjelmaa säilytetään ja miten erilliset ohjelmapäivitykset pidetään ajan tasaisina. Oikeudet ohjelman muokkaukseen ja käyttöön on myös määritettävä ja ohjelman

kommentoinnin on oltava sillä tasolla, että kuka tahansa pystyy jatkamaan työtä. Tähän kuuluu tarvittavien dokumenttien päivitys ja paikkansa pitävyys.

Versionhallinnan ensimmäinen ongelma oli yhteyksien luominen twinCatin kautta ohjelmitavaan logiikkaan. Koska salasanat ja koodi olivat kadonneet, piti ohjelmitavan logiikan yksikölle tehdä tehdasasetusten palautus, jotta saatiin uusi ohjelma ajettua sisään ohjelmitavaan logiikkaan. Yhteyden muodostaminen langattomasti ei onnistunut, joten projekti suoritettiin Ethernet-kaapelia käyttäen. Kun yhteys ohjelmitavan logiikan ja Twincat ohjelman välille oli muodostettu, aloitettiin kirjoittamaan Modbus-komentoja.

Minulla oli tukena osaavia ammattilaisia, ei Beckhoff twinCatin Modbus-konfiguraatio ollut selkeä. Selkeästi useassa kohdassa on ajateltu, että CX-moduuleihin liitetään erillinen RS-kortti. Koska meidän tapauksessa ei ollut mahdollista liittää uutta korttia tilanpuutteen vuoksi, oli käytettävä CX-moduulin omaa RS-liitintä. Tämä aiheutti ongelmia laitteen linkityksiä tehtäessä.

Kun anturi saatiin tuottamaan arvoja, piti seuraavaksi saada ohjelman funktio toimimaan. Funktion toimintakuntoon saattamiseen tarvittiin teknisen tuen apua, sillä Beckhoffin oma ohjeistus käytettävistä muuttujista oli puutteellinen. Kun tämä saatiin ratkaistua, päästiin kirjoittamaan koodiin tiedon tallennus ja päivittämään käyttöliittymä uudelle anturille.

Ensimmäisten mittausten jälkeen oli selkeästi havaittavissa, että anturin herkkyys ei ollut samaa tasoa vanhan anturin kanssa ja anturin kalibrointisuora oli virheellinen. Anturi ei tuottanut mitään arvoa pienemmillä pitoisuuksilla verrattaessa tuloksia vanhaan anturiin. Uputussyvyydellä oli merkitys kalibrointikäyrän kulmakertoimeen. Anturi ei antanut haluttuja arvoja pienillä pitoisuuksilla, kuten valmistaja lupasi, joten anturit piti lähettää uudelleen kalibroitaviksi valmistajalle.

Kalibroinnin jälkeen mittauksia jatkettiin kahdella upotussyvyydellä. Anturit tuottivat vertailukelpoista tietoa pienilläkin pitoisuuksilla uudelleenkalibroinnin jälkeen. Tarkemmalle tarkastelulle jää, että tarvitseeko upotussyvyyteen palata vai riittääkö noin 1,8 % epätarkkuus mittauksissa.

Opin työssä paljon lisää Modbus-protokollasta, sensoreiden asennuksesta ja kalibroinnista ja tarkistuksesta. Löysimme työn edetessä myös virheitä ohjelmasta, jotka tulemme korjaamaan tulevaisuudessa.

## Lähteet

Beckhoff. (2021). *Control Drawing I/O, CX, CPX*. Beckhoff.

[https://download.beckhoff.com/download/document/io/ethercat-terminals/controldrawing\\_io\\_cx\\_cpx\\_en.pdf](https://download.beckhoff.com/download/document/io/ethercat-terminals/controldrawing_io_cx_cpx_en.pdf)

Bolton, W. (2006). *Programmable Logic controllers*. Elsevier Science & Technology.

Kippo, A. K. & Tikka, A. (2008). *Automaatiotekniikan perusteet*. Edita.

Mackay, S., & Wright E. & Reynders, D. & Park, J. (2004). *Practical Industrial Data Networks : Design, Installation and Troubleshooting*. Elsevier Science & Technology.

Meijer, Gerard C.M. (2008). *Smart Sensor systems*. Wiley United Kingdom.

Modbus Organisation, (n.d). *Modbus FAQ*. Modbus Organisation.

<https://modbus.org/faq.php>

Morris, Allan S. (2001). *Measurement and Instrumentation Principles*. Butterworth Heinmann.

**Liite 1: Anturin kytkentä valmistajalta**

