

TAMPEREEN AMMATTIKORKEAKOULU
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio

Opinnäytetyö

Teemu Mauro

RFID-JÄRJESTELMÄN KÄYTTÖNOTTO

Työn ohjaaja
Työn teettäjä
Tampere 6/2009

Laboratorioinsinööri Seppo Mäkelä
Tampereen ammattikorkeakoulu

Tekijä	Teemu Mauro
Opinnäytetyön nimi	RFID- järjestelmän käyttöönotto
Sivumäärä	34
Valmistumisaika	30.6.2009
Työn ohjaaja	Seppo Mäkelä
Työn teettäjä	Tampereen ammattikorkeakoulu

TIIVISTELMÄ

RFID on yleisnimitys radiotaajuuksilla toimivalle etätunnistusteknologialle RFID- järjestelmissä RFID- lukija lukee ja kirjoittaa RFID- tunnisteeseen tietoa radioaaltojen avulla.

Opinnäytetyön tavoitteena on käyttöönottaa ja ohjelmoida RFID- järjestelmä harjoitustyökäyttöön Tampereen ammattikorkeakoulun kone- ja tuotantotekniikan koulutusohjelman opiskelijoita varten.

Työn tarkoituksena on perehdyttää opiskelijat passiivisen RFID- järjestelmän käyttöön, joka on merkittävästi kasvattanut suosiotaan kappaletavarateollisuuden tuotannonohjausmenetelmänä.

RFID- järjestelmä helpottaa selvästi tuotannon seurantaa, sillä tuotteista saadaan sen avulla runsaasti informaatiota. Tietomäärä, jota sen avulla voidaan siirtää on huomattavasti suurempi kuin esimerkiksi vaihtoehtoisilla viivakoodi- järjestelmillä.

Työssä käytettiin Siemensin Moby D passiivista RFID- järjestelmää ja sille tehtyä FB45-logiikkaohjelmaa. RFID- järjestelmä liitettiin Siemens Simatic 300 -logiikkaan.

Tulokseksi saatiin yksinkertainen ja helposti muokattava ohjelma, jolla voidaan ohjata ja lajitella kappaletavarateollisuuden tuotteita.

Writer	Teemu Mauro
Thesis	Commissioning of RFID- system
Pages	34
Graduation time	30.3.2009
Thesis Supervisor	Seppo Mäkelä
Co-operating Company	Tampere University of applied science

ABSTRACT

Main goal of this thesis is commissioning and programming of RFID- system for practice use to mechanical engineering students of Tampere university of applied science.

Meaning of this thesis is to familiarize students for use of passive RFID- system, which have significantly increased it's popularity as a production management method of piece goods industry.

RFID-system makes it easier to control production because it can give more information of products than competing bar code- systems.

The RFID-system used in this thesis was passive Siemens Moby D- system with FB45- program .RFID- system was connected to Siemens Simatic 300- logic.

An outcome of this thesis was simple and easily modifying program, which can control and classify products of piece goods- industry

ALKUSANAT

Tämä työ on tehty Tampereen ammattikorkeakoulun konelaboratoriolle. Haluan esittää kiitokseni Laboratorioinsinööri Seppo Mäkelälle opinnäytetyön aiheen antamisesta ja työn ohjaamisesta. Kiitän myös työn valvojana toiminutta Yliopettaja Olavi Kopposta.

Tampereella 8.6.2009

Teemu Mauro

SISÄLLYSLUETTELO

TIIVISTELMÄ	I
ABSTRACT.....	II
ALKUSANAT	III
SISÄLLYSLUETTELO.....	1
1 JOHDANTO	3
2 RFID-TEKNIikka	4
2.1 RFID- tekniikan taajuusalueet	5
2.1.1 LF- taajuusalue.....	5
2.1.2 HF- taajuusalue	6
2.1.3 UHF- taajuusalue	6
2.1.4 Mikroaaltoalue	7
2.2 RFID- tekniikan standardit.....	7
2.2.1 ISO 14443 -standardi	8
2.2.2 ISO 15693 -standardi	10
2.2.3 ISO 18000 -standardi	12
2.3 RFID- tunnistet.....	14
2.3.1 Aktiiviset RFID- tunnistet.....	15
2.3.2 Passiiviset RFID- tunnistet.....	16
2.4 RFID- lukijat	17
3 RFID- JÄRJESTELMÄN KÄYTTÖÖNOTTO JA OHJELMOINTI	18
3.1 RFID- järjestelmän laitteisto	19
3.1.1 Tunniste.....	19
3.1.2 Lukija	20
3.1.3 Liitäntämoduuli	20
3.2 RFID- ohjelman ominaisuudet.....	21
3.2.1 DB45- Parametrilohko	22
3.2.3 DB 47 -komentolohko.....	24
3.2.4 Luettavan ja kirjoitettavan tiedon lohkot	24
3.3 Ohjelman luominen.....	25
3.3.1 Ohjattava sovellus	25

3.3.2 Laitteiston konfigurointi	26
3.3.2 Parametrien asettaminen	27
3.3.3 Muuttujien tilataulukon muodostaminen	28
3.3.4 Kirjoitus- ja lukukomennon muodostaminen.....	29
3.3.5 Ohjelman muodostaminen	32
4 YHTEENVETO	33
LÄHTEET.....	34
LIITTEET	

1 Step 7 logiikkaohjelma 6 sivua

1 JOHDANTO

Tässä opinnäytetyössä käsitellään RFID- järjestelmän käyttöönottoa ja ohjelmointia. Työ jakaantuu kahteen osaan, joista ensimmäisessä käsitellään yleisesti erilaisia RFID- järjestelmiä, niiden taajuusalueita, standardeja sekä laitteistojen ominaisuuksia

Toisessa osassa käsitellään yhdenlaisen passiivisen RFID- järjestelmän käyttöönottoa ja ohjelmointia. Käyttöönottoon liittyviin asioihin kuuluvat RFID laitteiston ominaisuuksien esittely, sekä RFID- ohjelman ominaisuuksien esittely. Ohjelmointi osioissa käsitellään RFID- ohjelmaan tehtävää lisäystä ohjattavaa sovellusta varten.

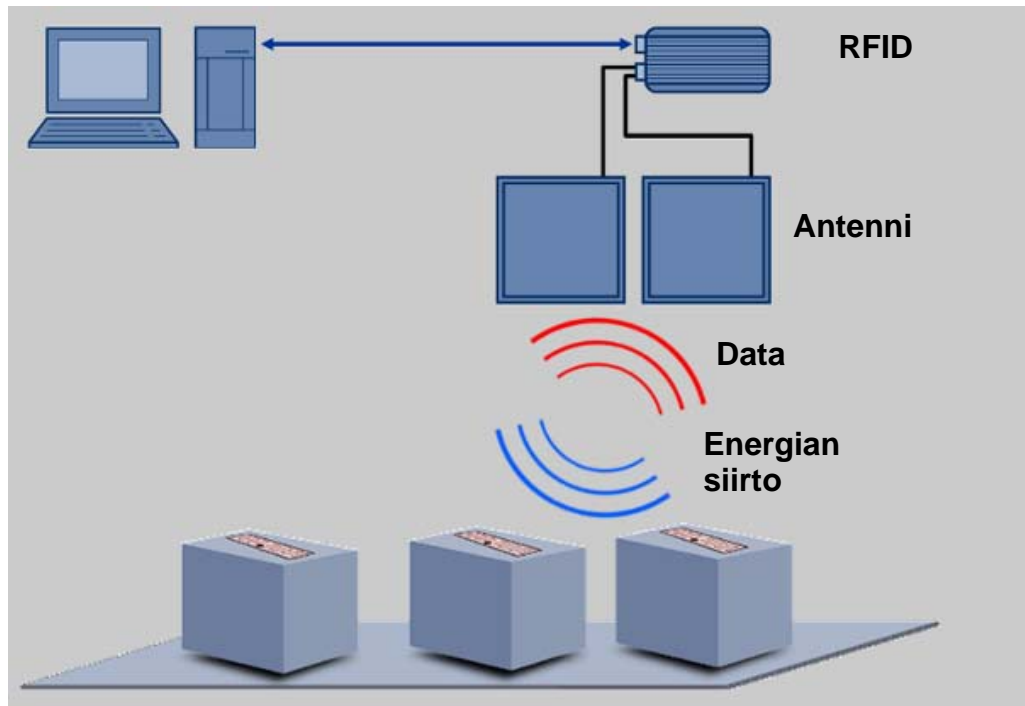
2 RFID-TEKNIikka

RFID(Radio Frequency Identification) on yleisnimitys radiotaajuuksilla toimivalle etätunnistusteknologialle. RFID(- järjestelmissä RFID-lukija lukee ja kirjoittaa RFID-tunnisteeseen tietoa radioaaltojen avulla.

RFID-teknologiaa voidaan verrata viivakoodiin. Kohteeseen kiinnitetään tunniste, joka kertoo kohteesta tietoja. RFID eroaa viivakoodista pääosin niin, että tunnistus voi tapahtua ilman suoraa visuaalista kontaktia tunnisteeseen. Lisäksi RFID- tunnisten sisältöä voidaan muuttaa matkan varrella kun taas viivakoodi on tulostuksen jälkeen muuttumaton. RFID- tunnisteet kestävät myös paremmin likaisia teollisuusolosuhteita kuin tavanomaiset viivakoodit.

RFID- järjestelmien idea on yksinkertainen: RFID- tunniste kiinnitetään haluttuun kohteeseen. Tunnisteeseen kirjoitetaan ja siitä luetaan tietoa RFID- lukijalla. Tietoa käytetään hyväksi taustajärjestelmän avulla

RFID on yleisnimitys kaikille radiotaajuuksilla tapahtuvalle tunnistamiselle, joten RFID-termin alle kuuluu monta erilaista teknologiaa. Oikean teknologian valinta tiettyyn sovelluskohteeseen ja oikeanlainen toteutus vaatii asiantuntemusta. Kuviossa 1. on esitetty periaatekuva RFID- järjestelmästä. [1]



Kuvio 1. Periaatekuva RFID- järjestelmästä

2.1 RFID- tekniikan taajuusalueet

RFID- järjestelmiin liittyvät olennaisena osana niiden käyttämät taajuusalueet. RFID-tunnisteet ja lukijat on suunniteltu keskustelemaan keskenään tietyllä taajuus alueella. Eri taajuusalueilla keskusteluun suunnitellut fysikaaliset mekanismit voivat olla erilaisia. Fysikaalisia mekanismeja on kahta eri tyyppiä: induktiivinen kytkentä ja radioaallot. Induktiivista kytkentää käytetään LF- ja HF- taajuusalueilla ja radioaaltoja UHF- taajuusalueella. [1]

2.1.1 LF- taajuusalue

LF (low frequency)- taajuusalueella RFID- järjestelmät toimivat 125 Hz taajuudella. LF- taajuusalueella toimivia järjestelmiä ei käytetä enää paljon uusissa sovelluksissa vaan käyttö rajautuu tiettyihin kulunvalvonnan ja eläintunnistuksen sovelluksiin. [1]

2.1.2 HF- taajuusalue

HF(high frequency)- taajuusalueella käytännön standarditaajuus on 13,56 MHz, joka on kansainvälisesti vapaa taajuus. HF- järjestelmiä käytetään yleensä lähietäisyydellä tunnistamisessa, kuten kulunvalvonnassa ja nykyisin myös teollisten tuotantolaitosten eri sovelluksissa. HF- taajuusalueella suurin mahdollinen lukuetaisyys lukijan ja tunnisteen välillä on noin 1,5 metriä optimiolosuhteissa, mutta käytännössä lukuetaisyys vaihtelevat samalle tunnisteele 0,05 m ja 1,0 m välillä sovelluksesta riippuen.

HF- taajuusalueen etuja verrattuna UHF- taajuusalueeseen, ovat kentän parempi läpäisykyky vettä sisältäviin aineisiin, häiriösietoisuus teollisuusympäristössä, ongelmattomuus heijastusten suhteen ja helppo lukialueen raja. [1]

2.1.3 UHF- taajuusalue

UHF (Ultra High Frequency) -taajuusalueet ovat hieman erilaisia ympäri maailmaa. Esimerkiksi Yhdysvalloissa RFID- järjestelmien käyttämä UHF- taajuusalue on 902-928 MHz, kun taajuus Euroopassa taajuusalue on hieman alempi ja pyörii 869 MHz:n ympärillä. UHF- tekniikka on saanut eniten kiinnostusta sen lupaavasta tulevaisuudesta logistiikan sovelluksissa. UHF- tekniikkaa sovelletaan maailmalla tunnetuiksi tulleissa logistiikan toimitusketjuihin liitetyissä RFID- järjestelmissä, joita muun muassa Wall-Mart, Tesco ja Metro Group ovat pystyttäneet.

Fysikaalisesti UHF- taajuusalueella toimiva RFID- tekniikka toimii eri tavalla kuin LF- ja HF- taajuusalueella toimivat. UHF- taajuusalueen RFID- tunnistuksessa on kysymys ns. far field -tekniikasta, jossa tunniste ja lukija kommunikoivat sähkömagneettista säteilyä, tässä tapauksessa radioaaltoja lähettämällä, kuten esimerkiksi radiossa ja matkapuhelimessa. HF ja LF- taajuusalueen RFID- tunnistuksessa on taas kysymys "near-field" induktiivisesta kytkennästä, jossa tunniste reagoi lukijan luomaan oskilloivaan magneettikenttään. Jos UHF- tekniikka voidaan verrata radioon, LF- ja HF- tekniikka toimii lähinnä kuten muuntaja. [1]

2.1.4 Mikroaaltoalue

Mikroaaltoalueella yleisin taajuus on 2,4 GHz. Mikroaaltoja käytetään enimmäkseen aktiivituunnistuksessa, jossa tunnistin sisältää oman virtalähteen. Mikroaaltojen tunnetuimpia sovelluksia on esimerkiksi automaattinen tunnistus tietulleissa. [1]

2.2 RFID- tekniikan standardit

RFID -tekniikan standardit määrittävät tunnisteen ja lukijan välisen tiedonvälityksen, tunnisteen tietosisällön, modulaatiotavat, ohjelmointikielen ja siirtonopeudet sekä muita huomioon otettavia asioita.

RFID- tekniikan standardit ovat tärkeitä etenkin logistiikan sovelluksissa, joissa rakennetaan avoimia kuljetusketjuja, koska tällöin useamman eri toimittajan järjestelmien tulee pystyä lukemaan samoja tunnisteita. Toinen standardien tärkeä tehtävä on valmistajariippumattomuus. Isoa järjestelmää rakennettaessa on hyvä varmistua siitä että järjestelmään sopivia laitteita ja tunnisteita voi myöhemmin ostaa vapaasti sitoutumatta johonkin tiettyyn toimittajaan. Standardi ei sinänsä takaa valmistajariippumattomuutta, mutta osa standardeista on ns. vapaita standardeja, joiden mukaisia laitteistoja kuka tahansa voi alkaa valmistaa.

RFID-tekniikan huomattavimpia standardoijia ovat ISO (International Organisation for Standardization) ja GS1-organisaatioon kuuluva EPCglobal Inc. Ensimmäisiä ISO:n määrittelemiä RFID-standardeja olivat kotieläinten ja karjan tunnistamiseen tarkoitettut ISO 11784 - ja ISO 11785 -standardit. ISO 11784 määrittää tunnisteen tietosisällön ja ISO 11785 määrittää tiedonsiirto-protokollan.

Maksujärjestelmien ja kontaktittomien älykorttien ISO 14443 -standardi ja etälukukorttien ISO 15693 -standardi luotiin tunnisteen ja lukijan välistä ilmarajapintaa varten. [1]

2.2.1 ISO 14443 -standardi

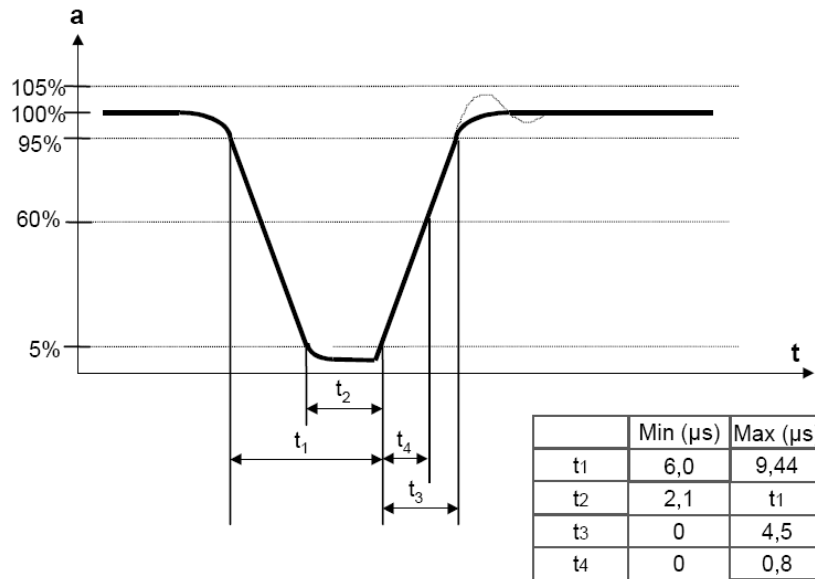
ISO 14443 on neliosainen kansainvälinen standardi kontaktivapaille älykorteille, jotka toimivat HF- taajuusalueella ja pienillä lukuetaisyysyksillä. Tämän standardin RFID-järjestelmien lukuetaisyys on korkeintaan 0,1 m ja oletusarvoinen lukunopeus 106 kb/s. Suurempia lukunopeuksia kuten 212 kb/s on myös saatavilla.

ISO 14443 -standardin neljä osaa asettavat seuraavia vaatimuksia RFID- järjestelmälle.

ISO 14443-1 -standardi asettaa vaatimukset fyysisille ominaisuuksille kuten kortin mitoille, jotka on määritetty standardissa ISO 7810, mekaaniselle kestävyydelle ja herkkyydelle ympäröiviä magneettikenttiä kohtaan. Standardi määrittää myös muutamia ympäristörasitteita, joita kortin tulee kestää ilman, että ne aiheuttavat pysyviä vaurioita sen toiminnalle. Näitä ovat ultraviolettisäteily, röntgensäteily, dynaaminen taivutus ja vääntöpaine, vaihtelevat magneetti- ja sähkökentät ja staattiset sähkö- ja magneettikentät. Standardissa määritetty toiminta lämpötila-alue on 0 - 50 celsiusta.

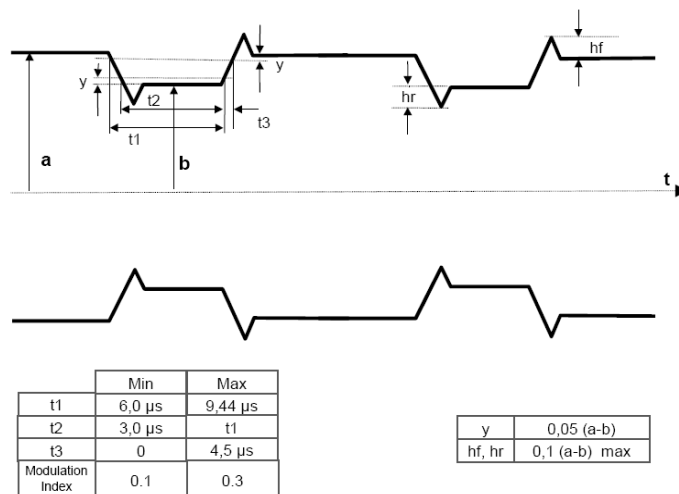
ISO 14443-2 -standardissa määritetään ominaisuudet tehon siirrolle ja kommunikoinnille tunnisteen ja lukijan välillä. Standardi määrittää tehon siirron kortille tapahtuvan taajuusalueella 13,56 MHz +/- 7kHz. Standardi määrittää järjestelmälle kaksi tiedonsiirto menetelmää TYPE A ja TYPE B. Molemmat kommunikaatio tavat toimivat half duplex -menetelmällä, joka tarkoittaa tiedon lähettämistä ja vastaanottamista vuorotellen. Näiden kahden tiedonsiirtomenetelmän erot muodostuvat tiedonsiirtoon tarvittavien magneettikenttien modulointitavoista, bitti- ja tavumuodoista sekä tunnisteen erottelutavoista.

TYPE A käyttää 100 %:n amplitudimodulaatiomenetelmää, mikä tarkoittaa, että tiedonsiirron aikana tieto koodataan käyttäen lyhyitä taukoja. Taukojen aikana tehoa ei siirry lukijasta tunnisteseen. Tämä asettaa erikoisvaatimuksia tunnisteele kortin sisällä. 100 %:n amplitudimodulaatiomenetelmän aaltomuoto on esitetty kuviossa 2.



Kuvio 2. 100 %:n amplitudimodulaation aaltomuoto

TYPE B käyttää 10 %:n amplitudimodulaatiomenetelmää, mikä tarkoittaa, että tieto koodautuu ainoastaan vähäisellä normaalin amplituditason alenemisella. Tällä saavutetaan tehon läpipääsy koko kommunikaatioprosessin ajan tunnisteiden ja lukijan välillä. Tällä saavutetaan suuria etuja verrattuna tyyppin A menetelmään.



Kuvio 3. 10 %:n amplitudimodulaation aaltomuoto

ISO 14443-3 -standardissa määritetään järjestelmän käyttöönottoon ja tunnisteen erotteluun liittyviä vaatimuksia. Standardi määrittää järjestelmän ohjelmallisia vaatimuksia, kuten tavumuodot, komentokentät ja ajoitukset. Standardi määrittää erottelutavat, joilla se havaitsee yksittäisen tunnisteen ja kommunikoi sen kanssa, kun useampia tunnisteita on lukuetaisytydellä. Standardin määrittelemät erottelutavat nojaavat siihen, että jokaisella tunnistella on oma yksilöllinen tunnistekoodi.

ISO 14443-4 -standardi määrittää korkean tason tiedonsiirto protokollia järjestelmälle. Nämä protokollat ovat valinnaisia elementtejä ISO 14443 -standardissa eli järjestelmää ei tarvitse suunnitella tätä protokollaa tukevaksi. Protokollassa käsitellään lähinnä tiedonsiirron nopeuteen liittyvää keskustelua tunnisteen ja lukijan välillä sekä häiriöiden käsittelyä ja järjestelmän palautumista. [2]

2.2.2 ISO 15693 -standardi

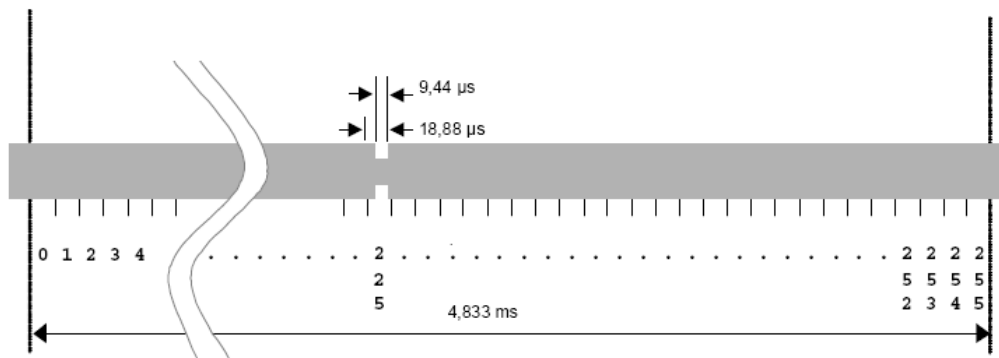
ISO 15693 on kolmeosainen kansainvälinen standardi kontaktivapaille älykortteille, jotka toimivat HF- taajuusalueella. Tämän standardin RFID- järjestelmien lukuetaisytyys on korkeampi kuin ISO 14443 -järjestelmien ja on korkeintaan 50 cm.

ISO 15693-1 standardi asettaa älykortille samat vaatimukset kortin fyysisille ominaisuuksille ja ympäristörasitteiden kestolle kuin ISO 14443-1 standardi.

ISO 15693-2 määrittää ominaisuudet tehonsiirrolle ja kommunikaatiolle tunnisteen ja lukijan välillä. Tämän standardin RFID- järjestelmissä käytetään samoja TYPE A- ja TYPE B -tiedonsiirtomenetelmiä kuin ISO 14443-2 standardin järjestelmissä. Standardi määrittää tiedon koodaamiselle kaksi tapaa 1-256 ja 1-4.

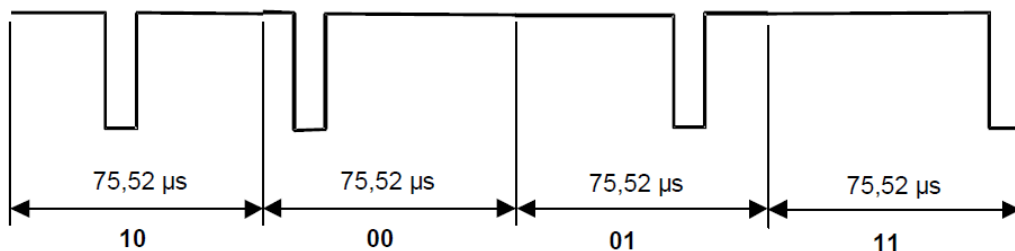
1-256 tavassa siirretään tietoa tavu kerrallaan. Tavun siirtoon käytettävä aika on tässä menetelmässä 4,88 ms. 4.88 ms aikajana jaetaan 256:een osaan. Tavun arvo

määräytyy sen perusteella, mille aikajanan osalle tiedonsiirrossa käytettävän amplitudimodulaation tauko osuu. Kuvassa 3 on esitetty 1-256 tyyppisen tiedon koodaus tavan aikajana, jossa tauko osuu kohdalle 225 dec. Tämän tiedon lukija siirtää tunnisteikortille.



Kuvio 4. 1-256 tyyppinen tiedonkoodaustapa

1-4 tyyppisessä tiedonkoodaus tavassa siirretään 2 bittiä kerrallaan. 4 bittiparia muodostavat tavun, ja pareista vähiten merkitsevä lähetetään ensin. Bittiparin lähetymisen kesto on 75 mikrosekuntia. Kuvassa 4 on esitetty 1-4 tyyppinen tiedonkoodaustapa, jossa tieto 225 dec siirretään lukijasta tunnisteikortille.



Kuvio 5. 1-4 tyyppinen tiedonkoodaustapa.

ISO 15693-3 määrittää järjestelmälle tunnisteiden erottelu tavat ja tiedonsiirto protokollat. Tunnisteiden erottelu ja lukijan ja tunnistekortin välinen tiedonsiirto tapahtuu niin, että lukija kutsuu tiettyä tunnistekorttia, minkä jälkeen tunnistekortti vastaa. Tunniste ei siis lähetä tietoa ennen kuin lukija on lähettänyt kutsun. Standardin toimivuus perustuu siihen, että jokaisella tunnisteella on oma yksilöllinen tunnistekoodi. [3]

2.2.3 ISO 18000 -standardi

ISO 18000 –standardiperhe on tehty ilmarajapinnan kehitykseen mm. tavaroiden seuranta varten varastokirjanpidossa.

ISO 18000-1 määrittää parametrit joita kaikkien ISO 18000 -standardiperheen osien tulee noudattaa. Standardiperheen myöhemmät osat määräävät tarkemmin arvot ilmarajapintojen parametreille, kuten taajuudelle ja ilmarajapinnan tyypeille.

ISO 18000-2 -standardi määrittää ilmarajapinnan RFID- järjestelmille, jotka toimivat alle 135 kHz:n taajuusalueella. Tämän standardin järjestelmiä käytetään erilaisissa kappaleenkäsittelysovelluksissa. Standardi määrittelee kaksi erilaista kommunikointi tapaa TYPE A ja TYPE B. Molemmat noudattavat kuitenkin samoja tunnisteiden erottelutapoja ja tiedonsiirtoprotokollia.

TYPE A käyttää full duplex – menetelmää, joka tarkoittaa lukijan ja tunnisteen välistä tiedon lähettämistä ja vastaanottamista yhtä aikaa. TYPE A järjestelmät toimivat 125 kHz: taajuudella.

TYPE B käyttää half duplex -menetelmää, joka tarkoittaa jo aikaisemmin mainittua tiedon lähettämistä ja vastaanottamista vuorotellen lukijan ja tunnisteen välillä. TYPE B järjestelmät toimivat 134,2 kHz:n taajuusalueella.

ISO 18000-3 -standardi määrittää ilmarajapinnan RFID- järjestelmille, jotka toimivat 13,56 MHz:n taajuusalueella. Tämän standardin laitteistojen tulee noudattaa ISO 15693-2 standardissa määritettyjä tiedonsiirto -ja tiedonkoodaustapoja sekä ISO 15693-3 standardissa määritettyjä tunnisteiden erottelutapoja, sekä protokollia.

ISO 18000-4 -standardi määrittää ilmarajapinnan RFID-järjestelmille, jotka toimivat 2,45 GHz:n taajuusalueella. Tämän standardin laitteistot on tarkoitettu teollisuuden, tieteen ja lääketieteen sovelluksiin Lukuetaisyydet ovat yleensä yli metrin suuruisia. Standardi määrittelee kaksi järjestelmää MODE 1 ja MODE 2. MODE 1 on tarkoitettu passiivitunnisteille ja MODE 2 aktiivitunnisteille.

ISO 18000-5 -standardi määrittää ilmarajapinnan RFID- järjestelmille, jotka toimivat 5,8-5,9 GHz:n taajuusalueella. Standardi on kuitenkin myöhemmin poistettu käytöstä, koska sen mukaisia laitteistoja kohtaan ei ole kansainvälistä kiinnostusta.

ISO 18000-6 -standardi määrittää ilmarajapinnan RFID- järjestelmille, jotka toimivat 860-960 MHz:n taajuusalueella. Standardi määrittelee lukijan ja tunnisteiden välisen fyysisen kommunikoinnin, protokollat ja komennot sekä tunnisteiden erottelutapoihin liittyvät asiat. Standardi jaetaan kolmeen eri tyyppiin: A, B ja C. Tyypit eroavat toisistaan niiden erilaisten koodaustapojensa. [4][5]

2.3 RFID- tunnistet

RFID- tunnistet voi olla kohteeseen kiinnitettävä tarra, kortti, nappi tai implantti; tämä sisältää antennin ja sirun, johon tietoa tallennetaan tai josta tietoa luetaan. Kaikissa tunnisteeissa on kiinteä sarjanumero ja tietty määrä vapaata kirjoitus tilaa riippuen tunnisteen standardista.

RFID- tunnisteeita voi olla joko aktiivisia tai passiivisia. Aktiivisilla RFID- tunnisteeilla on oma virtalähde, joka voi olla esimerkiksi paristo. Aktiivitunnisteeet poikkeavat passiivitunnisteeista kahdella merkittävällä tavalla. Aktiivitunnisteeden lukuetaisyys on huomattavasti pidempi, jopa 100 metriä. Toiseksi aktiivitunnisteeisiin voidaan sisällyttää tiedonkeruuta, esimerkiksi GPS- sijaintitietoa tai lämpötilan seurantaa.

Passiivinen RFID- tunnistet saa käyttövirtansa RFID- lukijasta ilmaitse.

Passiivitunnisteeet on suunniteltu kertakäyttöisiksi, ja niiden valmistuskustannukset pyritään pitämään mahdollisimman pieninä. Passiivitunnisteeden lukuetaisyys voi enimmillään olla n. 10 metriä. Passiivitunnisteeet ovat olennaisimpia tulevaisuudessa logistiikan eri sovelluksissa. Kuviossa 3. on esitetty aktiivisia ja passiivisia RFID- tunnisteeita. [1][6]



Kuvio 6. Aktiiviset ja passiiviset RFID- tunnistet.

2.3.1 Aktiiviset RFID- tunnistet

Aktiivisia RFID- tunnistetta käytetään suurissa logistisissa operatioissa, kuten rahtikonttien, junanvaunujen ja suurien uudelleenkäytettävien pakkauksien tunnistamisessa laajoilla tavarankäyttöalueilla. Aktiivitunnisteen tyypillinen lukualue on 20 metristä 100 metriin.

Karkeasti jaotellen aktiivisia RFID- tunnistetta on kahta eri tyyppiä: tunnistimia ja suuntalähettimeitä. Tunnistimet herätetään lähettämällä signaalia RFID- lukijasta. Niitä käytetään esimerkiksi tarkastuspistevalvonnassa yms. Kun auto tai vaunu lähestyy tarkastusasemaa, asemalla oleva lukija lähettää signaalia, joka herättää autoon tai vaunuun kiinnitetyn tunnistimen. Signaalin saatuaan tunnistin lähettää yksilöidyn tunnistesignaalin lukijalle. Koska tunnistin lähettää signaalia ainoastaan silloin kun se saapuu lukijan lähettämän signaalin alueelle, on sen virran kulutus olematon.

Suuntalähettimeitä käytetään eniten reaaliaikaisissa paikannusjärjestelmissä, joissa kohteen sijainti pitää tarkasti jäljittää. Näissä järjestelmissä suuntalähetin lähettää yksilöityä tunnistus signaalia määrättyin aikaväleihin. Aikavälit voivat vaihdella kerrasta kolmessa sekunnissa yhteen kertaan päivässä, sen mukaan, kuinka tärkeää kohteen sijainnin tietäminen on minäkin ajanjaksoina. Suuntalähetin lähettää signaalia, jonka vastaanottaa ainakin kolme lukijaa sillä alueella, jossa kohdetta liikutellaan. Reaaliaikaisia paikannusjärjestelmiä käytetään yleensä ulkona, kuten jakelualueilla. Autonvalmistajat käyttävät järjestelmiä suurissa tuotantolaitoksissa jäljittääkseen osalaatikoita.

Aktiivitunnisteen hinta on yleensä 8 - 40 euroa riippuen tarvittavan muistin määrästä, energian kulutuksesta, eri tunnistusominaisuuksien määrästä (esim. lämpötilan tunnistus) tai tunnistelle asetetuista käyttöolosuhdevaatimuksista.[1] [6]

2.3.2 Passiiviset RFID- tunnistet

Passiivitunnisteilla ei ole virtalähdettä, kuten paristoa, tai lähetintä, kuten aktiivitunnisteissa. Passiivitunnistet ovat huomattavasti halvempia kuin aktiivitunnistet. Hintahaarukka on n. 20 - 40 senttiä. Ne eivät vaadi minkäänlaista ylläpitoa, kuten pariston vaihtoa. Tästä syystä valmistajat käyttävät passiivitunnisteita toimitusketjuissaan. Passiivitunnisteilla on paljon pienempi lukuetaisyys kuin aktiivitunnisteilla, vai muutamasta sentistä 30 senttiin.

Passiivitunniste sisältää mikrosirun, joka on yhdistetty antenniin. Tunniste voi olla paketoitu monella eri tavalla. Se voi olla tarra, lappu, implantti, kortti yms. Sille on kehitetty erikoispaketoititapoja, jotta se kestäisi raskaita olosuhteita, kuten kylmyyttä, kuumuutta tai syövyttäviä kemikaaleja.

Passiivitunniste voi toimia LF- , HF- ja UHF- taajuuksilla. Radioaallot käyttäytyvät eritavalla kaikilla näillä taajuusalueilla. LF- ja HF -taajuusalueilla toimivia passiivitunnisteita käytetään silloin, kun halutaan lukea tietoa materiaalin tai veden läpi, lyhyeltä kantomatkalta.

UHF- taajuudella toimivia passiivitunnisteita ei voi lukea materiaalin läpi. Syy miksi valmistajat käyttävät UHF- taajuudella toimivia passiivitunnisteita ovat niiden halpahinta verrattuna pitkään lukuetaisyyteen. Valmistajat käyttävät niitä tilanteissa, joissa tunnistetia ei ole mahdollisuutta lukea lähietäisyydeltä. Tämän taajuusalueen passiivitunnisteita käytetään yleensä varastoissa, joissa seurataan tulevan ja lähtevän tavarän liikennettä.

Tapa, jolla passiivitunniste kommunikoi lukijan kanssa, on erilainen kuin aktiivitunnisteella. LF- ja HF -taajuusalueilla toimivat passiivitunnistet ja lukijat muodostavat sähkömagneettisen kentän. Kun tunniste saapuu lukuetaisyydelle, lukija huomaa muutoksen magneettikentässä ja muuntaa sen tiedoksi, jonka tietokone ymmärtää. Koska tunnisteen ja lukijan magneettikentät eivät ole kovinkaan suuret,

täytyy tunnisteen olla melko lähellä lukijaa, että lukuprosessi niiden välillä onnistuu. Tästä syystä lukuetaisyys näissä järjestelmissä ei ole kovinkaan suuri.

UHF- taajuusalueilla toimivat passiivitunnisteet eivät muodosta magneetikenttää. Tunniste kerää energiaa lukijan lähettämistä radioaalloista, ja tunnisteen mikrosiru käyttää energiaa muuttamalla kuormaa antennissa ja lähettää muuttuneen signaalin takaisin lukijalle. [1] [6]

2.4 RFID- lukijat

RFID- lukijalla on tehtävänä tuottaa radioaalloilla aktiiviselle tunnisteele tai sähkömagneettisella kentällään passiiviselle tai puolipassiiviselle tunnisteele tiedon lähettämiseen tarvittava energia ja ottaa vastaan tunnisteen lähettämä tieto sekä prosessoida se. Lukuetaisyyden määrittää sekä lukijan että tunnisteen antennien koko ja lukijan lähettämien radioaaltojen taajuuden tai sähkömagneettisen kentän voimakkuus. Antennin kokoon vaikuttaa ainoastaan se, miten pienenä tunniste halutaan pitää, mutta lukijan kentän voimakkuutta rajoitetaan maakohtaisilla rajoituksilla. Tämän vuoksi eri maissa käytetyt RFID- järjestelmät eivät välttämättä ole yhteensopivia. Vaikka lukijaa kutsutaankin lukijaksi, voi se myös muuttaa, lisätä tai poistaa radioaaltojen tai sähkömagneettisen kenttensä avulla tunnisteele olevaa tietoa.

Sekä tunniste että lukija sisältävät joko sisäisen tai ulkoisen antennin, jonka avulla tietoa siirretään tunnisteele lukijalle ja päinvastoin. Antennin rakenne ja sijoittelu vaikuttavat antennin lähetysalueen suuruuteen, ulottuvuuteen ja tiedonsiirron tarkkuuteen. Lukijan antenniominaisuudet vaihtelevat myös suuresti käyttötarkoituksen mukaan. Kädessä pidettävässä lukijassa itse lukija ja antenni on yhdistetty samaan laitteeseen, kun taas esimerkiksi isossa varastotilassa yhden lukijan antennia voi olla ympäri varastoa. Kuviossa 7 on esitetty aktiivisen RFID- järjestelmän lukija, sekä ulkoinen antenni ja kuviossa 8. passiivisen RFID- järjestelmän lukija, jossa on sisäinen antenni. [1] [7]



Kuvio 7. Aktiivisen RFID- järjestelmän lukija ja ulkoinen antenni.



Kuvio 8. Passiivisen RFID- järjestelmän lukija sisäisellä antennilla

3 RFID- JÄRJESTELMÄN KÄYTTÖÖNOTTO JA OHJELMOINTI

RFID- järjestelmää käyttöönottavan ja ohjelmoivan henkilön täytyy ymmärtää, minkälaista sovellusta RFID- järjestelmällä ohjataan. On tiedettävä onko kyseessä kappalevarateollisuuden sovellus, jossa tunnisteiden ja lukijan välinen lukuetaisyys on lyhyt, vai logistiikan sovellus, jossa lukuetaisyydet ovat suuria? Tämän tiedon perusteella voidaan päätellä, onko sovellukseen tuleva RFID- järjestelmä tyypiltään aktiivinen vai passiivinen.

Kun RFID- järjestelmän tyyppi on tiedossa, täytyy selvittää, minkälaista tietoa sovelluksesta tai sovelluksen käsittelemästä tuotteesta halutaan. Tämän perusteella voidaan valita tietyn standardin mukainen tunniste ja lukija, joilla kyseistä tietoa voidaan käsitellä.

Kun tietyn standardin laitteisto on tiedossa, täytyy valita taustajärjestelmä, kuten ohjelmoitava logiikka ja tietokone, joilla ohjataan tunnisteiden ja lukijan välistä kommunikointia sekä käytetään hyväksi luettua tietoa.

Jotta kaikki edellä mainitut saataisiin keskustelemaan keskenään, on tietokoneessa oltava ohjelma, jolla ohjelmoidaan logiikka ohjaamaan laitteistoa toimimaan halutulla tavalla.

Tässä osiossa tutustutaan opinnäytetyössä käytettyyn RFID- laitteistoon sekä RFID-ohjelman ominaisuuksiin ja sen luomiseen.

3.1 RFID- järjestelmän laitteisto

Tässä osiossa tutustutaan tutkintotyössä käytetyn RFID- laitteiston ominaisuuksiin.

3.1.1 Tunniste

RFID- tunnisteena käytettiin Siemens Moby D MDS D100 passiivista tunnistetta, joka on esitetty kuviossa 9. Tunniste on ISO 15963:n ja ISO 18000-3:n mukainen älykortti, jonne mahtuu tietoa 32 kilobittiä.



Kuvio 9. Siemens Moby D MDS D100 passiivinen RFID-tunniste

3.1.2 Lukija

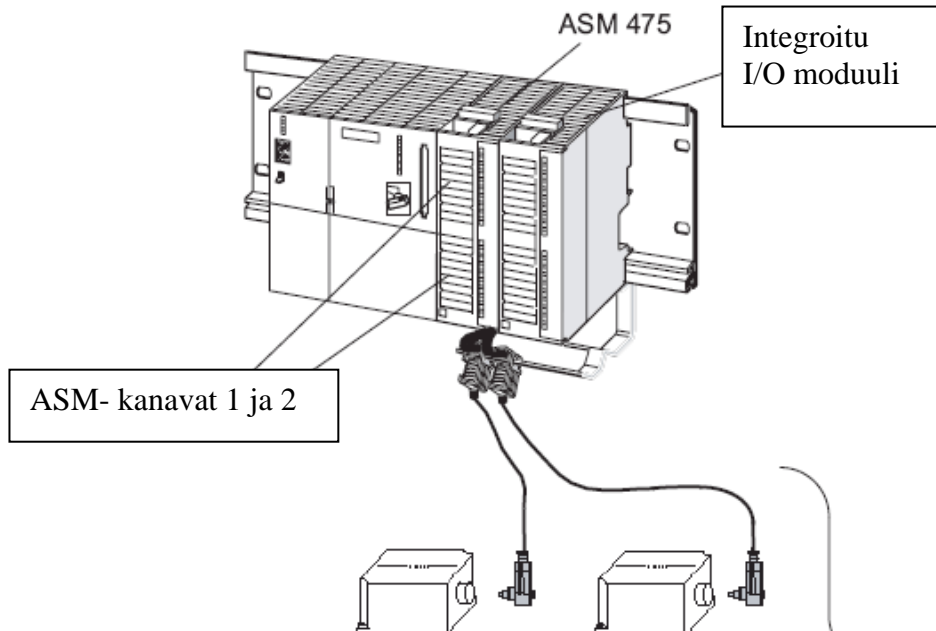
RFID- lukijana käytettiin Siemens Moby D SLG D12S -mallista passiivisen järjestelmän lukijaa, joka on esitetty kuviossa 10. Lukija on standardien ISO 15963 ja ISO 18000-3 mukainen. Tunnisteen luku- ja kirjoitusetäisyydeksi valmistaja on ilmoittanut 0,9 metriä. Tunnistusetäisyys on säädettävissä lukijan käyttämää tehoa muuttamalla.



Kuvio 10. Siemens Moby D SLG D12S RFID -lukija

3.1.3 Liitäntämoduuli

Edellä mainitulle RFID- lukijalle tarvitaan RFID- liitäntämoduuli, jonka avulla se kytketään ohjelmoitavaan logiikkaan. ASM 475 on Siemensin Moby D:n kanssa yhteensopiva integroitava liitäntämoduuli kahdelle RFID- lukijalle. Se on integroitavissa Siemens Simatic- 300 ja 400- mallisiin logiikkoihin. Liitäntämoduuli on esitetty kuviossa 11.



Kuvio 11. ASM475 -liitäntämoduuli integroituna logiikkaan.

3.2 RFID- ohjelman ominaisuudet

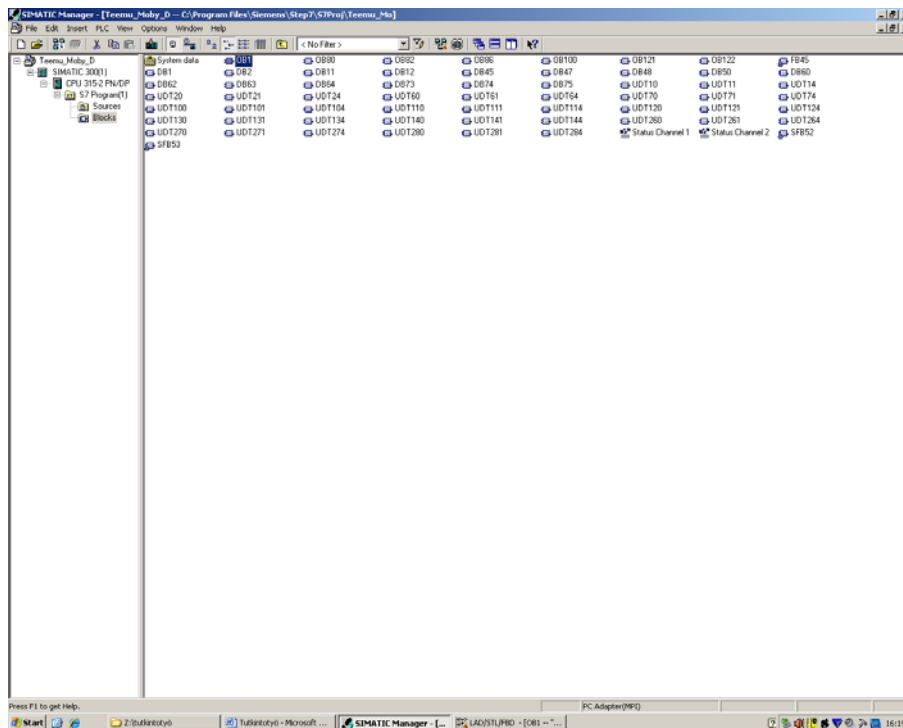
Tässä kappaleessa käsitellään RFID- ohjelman ominaisuuksia. Kappaleessa ei mennä tässä työssä käytettyjä ohjelman ominaisuuksia syvemmmälle vaan keskitytään ainoastaan olennaisiin Moby D -järjestelmälle tarkoitettuihin ominaisuuksiin.

Logiikan ohjelmoinnissa käytetään Siemens Step 7 -ohjelmistoa ja sille tehtyä FB 45 -lohkoa.

FB 45 on Siemens step 7 -lohko RFID-teknologialle. Sitä voidaan käyttää molempien SIMATIC S7-300- ja S7-400 logiikkaversioiden kanssa.

FB45- lohkon avulla voidaan käyttää kaikkia Siemensin MOBY -järjestelmiä ja siksi se sisältää näiden kaikkien järjestelmien ohjelmointiin tarvittavia ominaisuuksia.

FB45 jakaantuu useaan eri lohkoon. Kuviossa 12 on esitetty ohjelman lohkot, joista tässä työssä ohjelmoinnin kannalta merkittäviä ovat parametrilohko DB45, komentolohko DB47, organisointilohko OB1 sekä kirjoitettavan tiedon lohko DB48 ja luetun tiedon lohko DB50.



Kuvio 12. FB45 ohjelman lohkot

3.2.1 DB45- Parametrilohko

Ohjelman parametrisointi tehdään lohkoissa DB45. Lohko sisältää ohjelman input parametrit, control bitit, osoittimet sekä ohjelman sisäiset muuttujat; näitä on lohkoissa yhteensä viisikymmentä kappaletta jokaista lukijaa kohden.

Input parametrit: Nämä muuttujat asetetaan ohjelman konfiguroinnin yhteydessä, niitä ei ole tarpeen muuta ohjelman käytön aikana.

Control bitit: Käyttäjä aloittaa komentonsa näillä boolean muuttujilla.

Osoittimet: Osoittimet osoittavat komentojen edistymisen käyttäjälle. Niiden avulla voidaan tehdä helposti esimerkiksi virheanalyysyjä.

Taulukossa x on esitetty tärkeimpien parametrien merkitykset.

Nimi	Kommentti
ASM_address	RFID- liitäntämoduulin osoite. Osoitteen täytyy olla sama kuin Hardware osiossa oleva RFID- liitäntämoduulin aloitusosoite.
ASM_channel	RFID- liitäntämoduulissa olevan MOBY -kanavan numero.
command_DB_number	Komentolohkon numero.
command_DB_address	Komentolohkon osoite, josta seuraava komento alkaa.
Moby_mode	Tällä muuttujalla valitaan käytettävä lukija ja asetetaan käytettävä toimintamoodi. Tila 5 on varattu Moby D -lukijalle ilman monitunniste käsittelyä. Tila 7 on varattu Moby D -lukijalle monitunnistekäsittelyllä.
ANZ_MDS_present	Asettuu päälle, jos lukijaa havaitsee tunnisteiden läsnäolon.
cancel	Asetettaessa päälle keskeyttää luku- tai kirjoitusprosessin. Tämän jälkeen FB45 asettaa päälle ready muuttujan
command_start	Asetettaessa päälle aloittaa komennon tai komentoketjun, eli lukija on valmis luku- tai kirjoituskomentoon
init_run	Asetettaessa päälle suorittaa reset toiminnon, jolloin kaikki tiedot ja asetetut komennot häviävät lukijan muistista.
error	Tämä muuttuja asettuu päälle, jos komento keskeytyy epänormaalisti.
ready	Tämä muuttuja asettuu päälle, jos komento on suoritettu. Tämän jälkeen on kuitenkin varmistettava, että error muuttuja ei ole asettunut päälle. Tällä varmistetaan, että haluttu komento on suoritettu normaalisti.
ASM_busy error_Moby	Tämä osoitin osoittaa, että lukija on suorittamassa luku- tai kirjoituskomentoa. Tämä osoitin kertoo lukijan virheestä. Tämän virheen ilmetessä syttyy ERR_1 tai ERR_2- valo ASM-liitäntämoduulissa.

Taulukko 1. DB45- lohkon parametrit

3.2.3 DB 47 -komentolohko

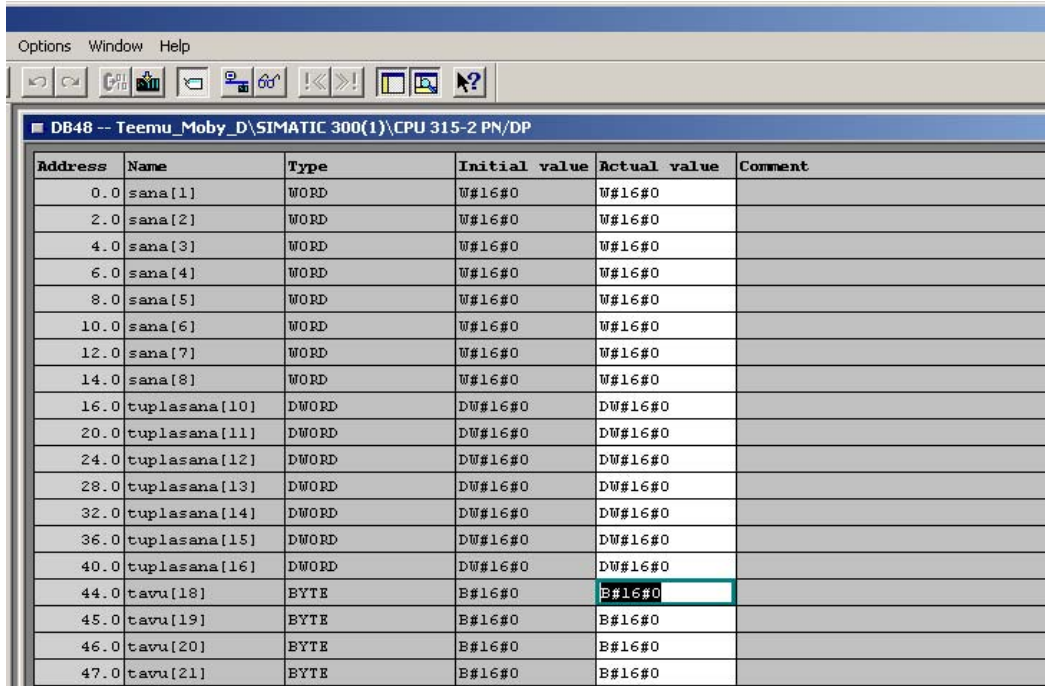
Tässä lohkossa määritellään komento, jonka lukija suorittaa. Määritykset on tehtävä ennen kuin Command start -muuttujan voi asettaa päälle luku- tai kirjoituskomentoa varten. Komentojen määrittystavut ja -sanat on esitetty taulukossa 2.

Taulukko. 2 DB 47- lohkon komennot

Nimi	Kommentti
command	Käyttäjä voi määrittää suorittaako lukija luku- vai kirjoituskomennon. 1 = lukukomento, 2 = kirjoituskomento
length	Käyttäjä voi määrittää luettavan- tai kirjoitettavan tiedon pituuden tavuina
DAT DB number	Käyttäjä voi määrittää luettavan- tai kirjoitettavan tiedon lohkon numeron
DAT DB address	Käyttäjä voi määrittää osoitteen lohkoista, josta tietoa lähdetään lukemaan- tai kirjoittamaan.

3.2.4 Luettavan ja kirjoitettavan tiedon lohkot

Jotta lukijalla pystyttäisiin lukemaan - ja kirjoittamaan tietoa, on tiedolla oltava paikka jonne se tallennetaan.. Tässä työssä kirjoitettava tieto tallennetaan lohkoon DB48 ja luettu tieto lohkoon DB50. Tieto voi olla muotoa tavu, sana tai tuplasana. Kuviossa 13 on esitetty eri muodossa olevia tietoja.



Address	Name	Type	Initial value	Actual value	Comment
0.0	sana[1]	WORD	W#16#0	W#16#0	
2.0	sana[2]	WORD	W#16#0	W#16#0	
4.0	sana[3]	WORD	W#16#0	W#16#0	
6.0	sana[4]	WORD	W#16#0	W#16#0	
8.0	sana[5]	WORD	W#16#0	W#16#0	
10.0	sana[6]	WORD	W#16#0	W#16#0	
12.0	sana[7]	WORD	W#16#0	W#16#0	
14.0	sana[8]	WORD	W#16#0	W#16#0	
16.0	tuplasana[10]	DWORD	DW#16#0	DW#16#0	
20.0	tuplasana[11]	DWORD	DW#16#0	DW#16#0	
24.0	tuplasana[12]	DWORD	DW#16#0	DW#16#0	
28.0	tuplasana[13]	DWORD	DW#16#0	DW#16#0	
32.0	tuplasana[14]	DWORD	DW#16#0	DW#16#0	
36.0	tuplasana[15]	DWORD	DW#16#0	DW#16#0	
40.0	tuplasana[16]	DWORD	DW#16#0	DW#16#0	
44.0	tavu[18]	BYTE	B#16#0	B#16#0	
45.0	tavu[19]	BYTE	B#16#0	B#16#0	
46.0	tavu[20]	BYTE	B#16#0	B#16#0	
47.0	tavu[21]	BYTE	B#16#0	B#16#0	

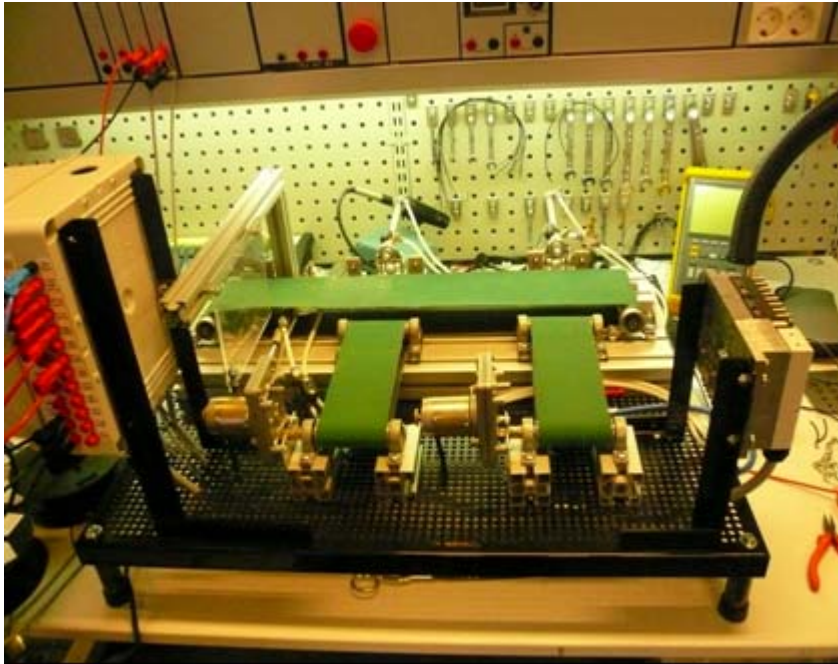
Kuvio 13. DB 48- kirjoitettavan tiedon lohko

3.3 Ohjelman luominen

RFID- ohjelman luonti vaatii hyvää logiikkaohjelmoinnin sekä ohjelmoitavan laitteiston tuntemusta. Ohjelmoijan on tiedettävä tarkkaan prosessin vaiheet, jossa RFID- laitteistoa käytetään hyväksi. Tässä osiossa käsitellään kolmen eri tietoryhmän luku- ja kirjoitusprosessia ja sen liittämistä työssä käytetyn sovelluksen logiikkaohjelmaan.

3.3.1 Ohjattava sovellus

Tässä työssä ohjattavana sovelluksena toimi kuljetinjärjestelmä, jossa on pääkuljetin ja kaksi sivukuljetinta. Pääkuljettimen varrella on kolme anturia sekä pääkuljettimen päässä että molempien sivukuljettimien kohdalla. RFID- järjestelmällä on tarkoitus ohjata kuljetinjärjestelmää niin, että pääkuljettimella kulkeva tuote ohjautuu siinä kiinni olevan tunnisteen tietojen perusteella oikeaan paikkaan. Kuviossa 14 on esitetty ohjattavana sovelluksena toimiva kuljetinjärjestelmä.



Kuvio 14. Kuljetinjärjestelmä

3.3.2 Laitteiston konfigurointi

Jotta Step 7 -ohjelmistolla pystyttäisiin ohjelmoimaan logiikkaa, on sille tehtävä laitteiston konfigurointi. Tämä tehdään ohjelman hardware-osiossa. Hardware-osiossa on lista Siemensin eri laitteista, josta on etsittävä sopivat tyyppimerkinnät logiikan virtalähteelle, prosessorille, RFID- liitäntämoduulille sekä I/O-moduulille, jolla tieto välitetään ohjattavan sovelluksen laitteille tai sovellukselta logiikalle. Kuviossa 15 on esitetty tässä työssä käytetyn laitteiston konfigurointi ja aloitusosoitteet.

Slot	Module	Order number	Firmware	MPI ad...	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU 315-2 PN/DP	6ES7 315-2EH13-0AB0	V2.3	2			
X1	MPI/DP			2	2047*		
X2	FN-IO				2046*		
3							
4	MOBY ASM475 - PARAM	6GT2002-0GA10			256...259	256...259	
5	DI16/DO16x24V/0.5A	6ES7 323-1BL00-0AA0			4...5	4...5	
6							
7							
8							

Kuvio 15. Laitteiston konfigurointi

3.3.2 Parametrien asettaminen

Konfiguroinnin jälkeen asetetaan parametrit DB45- parametrilohkoon. Asetettavia parametrejä ovat ASM- liitäntämoduulin aloitusosoite, RFID- lukijoiden kanavat ASM- liitäntämoduulissa, komentolohkon numero, komennon aloitusosoite sekä lukijoiden toimintamoodi. Kuviossa 16 on esitetty tässä työssä asetetut parametrit.

The image shows two screenshots of the SIMATIC Manager parameter table. The top screenshot shows parameters for SLC1 (addresses 0.0 to 15.0) and the bottom screenshot shows parameters for SLC2 (addresses 50.0 to 61.0). Arrows point from text boxes to specific parameter values in the tables.

Address	Name	Type	Initial	Actual	Value	Comment
0.0	SLC1.ASM_address	INT	256	256		Input: address of ASM (cycle word)
2.0	SLC1.ASM_channel	INT	1	1		Input: number of channel (1..4)
4.0	SLC1.command_DB_number	INT	47	47		Input: number of command DB
6.0	SLC1.command_DB_address	INT	0	0		Input: first address of commands in the command DB
8.0	SLC1.MDS_control	BYTE	B#16#1	B#16#1		Input: setup the MDS controlling (0,1,2)
9.0	SLC1.ECC_mode	BOOL	FALSE	FALSE		Input: working with ECC check
9.1	SLC1.RESET_long	BOOL	TRUE	TRUE		Input: true: long RESET-telegram, only used for MOBY mode 5, 6
10.0	SLC1.MOBY_mode	BYTE	B#16#5	B#16#5		Input: MOBY working mode
11.0	SLC1.scanning_time	BYTE	B#16#0	B#16#0		Input: scan time for long-range MOBY I/U
12.0	SLC1.option_1	BYTE	B#16#0	B#16#2		Input: reset-command option 1
13.0	SLC1.distance_limiting	BYTE	B#16#F	B#16#F		Input: range limit
14.0	SLC1.multitag	BYTE	B#16#1	B#16#1		Input: max. no. of MDS in field
15.0	SLC1.field_ON_control	BYTE	B#16#0	B#16#0		Input: working mode of zero

Address	Name	Type	Initial	Actual	Value	Comment
50.0	SLC2.ASM_address	INT	256	256		Input: address of ASM (cycle word)
52.0	SLC2.ASM_channel	INT	1	2		Input: number of channel (1..4)
54.0	SLC2.command_DB_number	INT	47	47		Input: number of command DB
56.0	SLC2.command_DB_address	WORD	W#16#0	W#16#50		Input: first address of commands in the command DB
58.0	SLC2.MDS_control	BYTE	B#16#1	B#16#1		Input: setup the MDS controlling (0,1,2)
59.0	SLC2.ECC_mode	BOOL	FALSE	FALSE		Input: working with ECC check
59.1	SLC2.RESET_long	BOOL	TRUE	TRUE		Input: true: long RESET-telegram, only used for MOBY mode 5, 6
60.0	SLC2.MOBY_mode	BYTE	B#16#5	B#16#5		Input: MOBY working mode
61.0	SLC2.scanning_time	BYTE	B#16#0	B#16#0		Input: scan time for long-range MOBY I/U

Kuvio 16. Parametrien asettaminen

3.3.3 Muuttujien tilataulukon muodostaminen

Luku- ja kirjoitusprosessin toteuttamiseksi on sille määritettävä komennot, joita prosessin halutaan noudattavan, sekä tiedot, joita halutaan kirjoitettavaksi tunnisteelle ja luettavaksi tunnisteelta. Tätä varten kannattaa muodostaa muuttujien tilataulukot molemmille, sekä lukevalle että kirjoittavalle lukijalle. Tähän taulukkoon listataan kaikki prosessin kannalta merkittäviä tietoja omaavat bitit, tavut ja sanat. Taulukon avulla on helppo seurata prosessia yhdestä ohjelman osiosta käsin.

Muuttujien tilataulukosta nähdään sinne listattujen bittien, tavujen ja sanojen osoitteet, nimet, niiden nykytila sekä kenttä, josta tavujen ja sanojen tilaa voidaan muokata.

Kuviossa 17 on esitetty tätä työtä varten muodostetut muuttujien tilataulukot.

The image shows two overlapping windows from the SIMATIC Manager software, displaying variable status tables for two different channels. The top window is titled 'Var - Status Channel 2' and the bottom window is 'Status Channel 2 -- Teemu_Moby_D\SIMATIC 300(1)\CPU 315-2 PN\DP\S7 Program(1)'. Both windows show a table with columns for Address, Symbol, Status value, and Modify value. The tables list various variables such as M (bits), DB45.DBX (data bytes), DB45.DBB (data blocks), and DB48.DBW (data words), along with their symbolic names and descriptions. The status values are mostly greyed out, indicating they are not currently active or are in a default state.

Address	Symbol	Status value	Modify value
1	M 1.0		
2	M 1.2		
3			
4	// Cancel		
5	DB45.DBX 19.0 "MOBY DB" SLG1.cancel		
6	// Command Start		
7	DB45.DBX 19.1 "MOBY DB" SLG1.command_start		
8	// System Start Up		
9	DB45.DBX 19.3 "MOBY DB" SLG1.init_run		
10	// Ready		
11	DB45.DBX 18.7 "MOBY DB" SLG1.ready		
12	// Presence of a MDS		
13	DB45.DBX 18.0 "MOBY DB" SLG1.ANZ_MDS_present		
14	DB45.DBX 19.2 "MOBY DB" SLG1.repeat_command		
15	// Error		
16	DB45.DBX 18.6 "MOBY DB" SLG1.error		
17	// Errors		
18	DB45.DBB 22 "MOBY DB" SLG1.error_MOBY		
19			
20	// MOBY Command		
21	DB47.DBB 0 "Command" Kanal_1_Befehl[1].command		
22	DB47.DBW 2 "Command" Kanal_1_Befehl[1].length		
23	DB47.DBW 6 "Command" Kanal_1_Befehl[1].DAT_DB_number		
24	DB47.DBW 8 "Command" Kanal_1_Befehl[1].DAT_DB_address		
25			
26	//Tulosanat kirjoitustilassa 1		
27	DB48.DBW 0		
28	DB48.DBW 2		
29			
30	//Tulosanat kirjoitustilassa 2		
31	DB48.DBW 4		
32	DB48.DBW 6		
33			
34	//Tulosanat kirjoitustilassa 3		
35	DB48.DBW 8		
36	DB48.DBW 10		
37			
38			
39			
40			
41			
42			
43			
44			
45			
46			
47			
48			
49			
50			
51			
52			
53			
54			
55			
56			
57			
58			
59			
60			
61			
62			
63			
64			
65			
66			
67			
68			
69			
70			
71			
72			
73			
74			
75			
76			
77			
78			
79			
80			
81			
82			
83			
84			
85			
86			
87			
88			
89			
90			
91			
92			
93			
94			
95			
96			
97			
98			
99			
100			

Kuvio 17. Muuttujien tilataulukot luku- ja kirjoitusprosessille.

3.3.4 Kirjoitus- ja lukukomennon muodostaminen

Tilataulukon muodostamisen jälkeen aletaan määrittää komentoa luku- ja kirjoitusprosessille. Taulukkoon on listattu kaikki komennon määrittämiseen vaadittavat tavut ja sanat, joiden arvoja muuttamalla saadaan suoritettua erilaisia komentoja.

Tämän työn luku- ja kirjoitusprosessin mukaisesti kirjoitettavan tiedon lohkoista on listattu taulukkoon kuusi sanaa, ja ne on jaettu kolmeen eri ryhmään. Luettavan tiedon puolelle on listattu kaksi sanaa, joihin luetut tiedot tallennetaan.

Näille sanoille annetaan arvot, joiden perusteella sovelluksen käsittelemät eri tuotteet tunnistetaan.

Komennon määrittäminen lukijalle aloitetaan command- tavusta. Sille annetaan arvo sen mukaan, suorittaako se luku- vai kirjoitusprosessin. Arvo on yksi, jos kyseessä on kirjoittava lukija ja kaksi, jos se on lukeva. Tämän jälkeen määritetään paikka, jonne luettu tai kirjoitettu tieto tallennetaan. Tässä työssä ne tallennetaan kirjoitettavan tiedon lohkoon DB48 ja luetun tiedon lohkoon DB50. Kyseisten lohkojen numero asetetaan DAT_DB_number-sanalle. Seuraavaksi rajataan tunnisteelle kirjoitettavan tai siltä luettavan tiedon määrä. Tämä tapahtuu asettamalla arvo tavuina length - sanalle. Esimerkiksi, jos halutaan lukea- tai kirjoittaa ainoastaan kaksi sanaa, asetetaan length-sanan arvoksi neljä. Tämän jälkeen asetetaan aloitusosoite, josta tietoa aletaan lukea tai kirjoittaa. Tämä tapahtuu asettamalla arvo DAT_DB_address-sanalle. Tässä työssä päätettiin asettaa aloitusosoite ohjelmallisesti, joten sen arvoksi jätettiin luku 0.

Jotta tuotteet, joita kuljetinjärjestelmä käsittelee, saataisiin eroteltua toisistaan, on niille annettava oma yksilöllinen tieto. Tässä työssä edellä mainittua tietoa kuvastaa kaksi sanaa, joille annetaan kummallekin arvo. Lukija kirjoittaa nämä kaksi komennossa määriteltyä sanaa tunnisteelle. Kuvioissa 18 ja 19 on esitetty tämän työn luku- ja kirjoitusprosessille määritetyt komennot ja kirjoitettavat tiedot.

	Address	Symbol	Status value	Modify value
1	M 1.0			
2	M 1.2			
3				
4	// Cancel			
5	DB45.DBX 19.0	"MOBY DB".SLG1.cancel		
6	// Command Start			
7	DB45.DBX 19.1	"MOBY DB".SLG1.command_start		
8	// System Start Up			
9	DB45.DBX 19.3	"MOBY DB".SLG1.init_run		
10	// Ready			
11	DB45.DBX 18.7	"MOBY DB".SLG1.ready		
12	// Presence of a MDS			
13	DB45.DBX 18.0	"MOBY DB".SLG1.ANZ_MDS_present		
14	DB45.DBX 19.2	"MOBY DB".SLG1.repeat_command		
15	// Error			
16	DB45.DBX 18.6	"MOBY DB".SLG1.error		
17	// Errors			
18	DB45.DBB 22	"MOBY DB".SLG1.error_MOBY		
19				
20	// MOBY Command			
21	DB47.DBB 0	"Command".Kanal_1_Befeh[1].command		B#16#01
22	DB47.DBW 2	"Command".Kanal_1_Befeh[1].length		4
23	DB47.DBW 6	"Command".Kanal_1_Befeh[1].DAT_DB_number		48
24	DB47.DBW 8	"Command".Kanal_1_Befeh[1].DAT_DB_address		0
25				
26	//Tulosanat kirjoitustilassa 1			
27	DB48.DBW 0			11
28	DB48.DBW 2			12
29				
30	//Tulosanat kirjoitustilassa 2			
31	DB48.DBW 4			21
32	DB48.DBW 6			22
33				
34	//Tulosanat kirjoitustilassa 3			
35	DB48.DBW 8			31
36	DB48.DBW 10			32
37				

Kuvio 18. Komennon määrittämiset kirjoitettavalle tiedolle.

	Address	Symbol	Status value	Modify value
1	M 1.1			
2	M 1.3			
3				
4		// Cancel		
5	DB45.DBX 69.0	"MOBY DB".SLG2.cancel		
6		// Command Start		
7	DB45.DBX 69.1	"MOBY DB".SLG2.command_start		
8		// System Start Up		
9	DB45.DBX 69.3	"MOBY DB".SLG2.init_run		
10		// Ready		
11	DB45.DBX 68.7	"MOBY DB".SLG2.ready		
12		// Presence of a MDS		
13	DB45.DBX 68.0	"MOBY DB".SLG2.ANZ_MDS_present		
14				
15		// Error		
16	DB45.DBX 68.6	"MOBY DB".SLG2.error		
17		// Errors		
18	DB45.DBB 72	"MOBY DB".SLG2.error_MOBY		
19				
20		// MOBY Command		
21	DB47.DBB 50	"Command".Kanal_2_Befehl[1].command		B#16#02
22	DB47.DBW 52	"Command".Kanal_2_Befehl[1].length		4
23	DB47.DBW 56	"Command".Kanal_2_Befehl[1].DAT_DB_number		50
24	DB47.DBW 58	"Command".Kanal_2_Befehl[1].DAT_DB_address		0
25				
26				
27				
28				
29		//Lahtosanat		
30	DB50.DBW 0			
31	DB50.DBW 2			
32				
33				
34				
35				
36				
37				
38				

Kuvio 19. Komennon määrittelyt luettavalle tiedolle.

3.3.5 Ohjelman muodostaminen

Kun kaikki logiikan ohjelmointiin vaadittava osoitteisto on selvitetty, voidaan aloittaa ohjelman muodostaminen. Kuten aikaisemmin mainittiin, ohjelmoinnissa käytettiin Siemensin Step 7- logiikkaohjelmaa ja sille tehtyä FB45- lohkoa.

Tehtävänä oli muodostaa ohjelma, joka mahdollistaa luku- ja kirjoitusprosessin suorittamisen ohjattavasta sovelluksesta käsin samalla, kun se ohjaa itse sovellusta.

Tämä mahdollistettiin käyttämällä painonappia, jota painamalla käynnistettiin kirjoitus prosessi. Painonappia painettaessa syttyi valo, joka kuvasti kirjoituskomentoa. Sovelluksessa oli siis valo jokaista kirjoituskomentoa varten. Kun ohjelma ilmoitti kirjoitusprosessin päättymisestä, käynnistettiin välittömästi lukukomento.

Kun lukukomento oli suoritettu, käynnistyi pääkuljetin, joka vei tunnisteelta luetun tiedon mukaisesti tuotteen haluttuun paikkaan. Tämä mahdollistettiin vertailemalla tunnisteelta olevaa tietoa ohjelmaan asetettuihin vakioarvoihin. Liitteessä 1 on esitetty ohjelma RFID:llä ohjatusta kuljetinjärjestelmästä.

4 YHTEENVETO

Tämän opinnäytetyön tarkoituksena oli käyttöönottaa RFID- järjestelmä ja tehdä sille opetuskäyttöön avuksi soveltuva ohjelma.

Kaikkia työssä käytetyn RFID- järjestelmän ominaisuuksia ei saatu hyödynnettyä, koska se olisi vaatinut asiantuntijoiden apua, mutta tavoitteena pidettyyn yksinkertaisen luku- ja kirjoitusprosessin muodostamiseen päästiin

Lopputulokseksi saatiin ohje työssä käytetyn laitteiston käyttöönotosta ja sen perusteella luodusta helposti muokattavasta ohjelmasta. Asiaa opiskelevat saavat työstä selkeän kuvan RFID- järjestelmän avulla toteutetusta kappalevarateollisuuden tuotannonohjausjärjestelmästä.

LÄHTEET

1. RFIDLab Finland – RFID- tekniikan perusteet. [www-sivu][viitattu 4.6.2009]
saatavissa: <http://www.rfidlab.fi>
<http://www.rfidlab.fi/default.asp?t=1&f=2&p=800&subp=800&subp0=800&did=61>
2. Otiglobal - Object [www-sivu][viitattu 4.6.2009] Saatavissa:
<http://www.otiglobal.com/objects/ISO%2014443%20WP%204.11.pdf>
3. Waazaa – Vicinity cards [www-sivu][viitattu 4.6.2009] Saatavissa:
<http://www.waazaa.org/15963.org>
4. ISO – ISO catalogue [www-sivu][viitattu 17.4.2009] Saatavissa: <http://iso.org>
[/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=34112](http://iso.org/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=34112)
5. High Tech Aid – Standards[www_sivu][viitattu 4.6.2009] Saatavissa:
<http://www.hightechaid.com/standards/18000.htm>
6. Savonia – wirelessplatform[www-sivu][viitattu 4.6.2009] saatavissa:
http://wirelessplatform.savonia-amk.fi/index.php?option=com_content&task=view&id=23&Itemid=50
7. Techweb – encyclopedia[www-sivu][viitattu 4.6.2009] saatavissa:
<http://www.techweb.com/encyclopedia/defineterm.jhtml?term=RFIDreader>

SIMATIC Teemu Moby D\ 06/18/2009 11:29:45 AM
 SIMATIC 300(1)\CPU 315-2 PN/DP\...\OB1 - <offline>

OB1 - <offline>
 Cycle Execution Cycle Execution
 Name: MOBY Family: MOBY
 Author: SR2206 Version: 1.3
 Block version: 2
 Time stamp Code: 06/02/2009 06:47:06 PM
 Interface: 02/15/1996 04:51:13 PM
 Lengths (block/logic/data): 00782 00630 00026

Name	Data Type	Address	Comment
TEMP		0.0	
OB1_System	Array [1..20] Of Byte	0.0	

Block: OB1 "Main Program Sweep (Cycle)"

Network: 1 call FB45

memory bit 1.0 is set: Start MOBY command for 1st SLG
 memory bit 1.1 is set: Start MOBY command for 2nd SLG
 memory bit 1.2 is set: Start init_run for 1st SLG
 memory bit 1.3 is set: Start init_run for 2nd SLG
 memory byte 2: OB1 used as edge triggered memory

```

CALL "MOBY FB" , "Instanz-DB K1" // Call FB 45 for each SLG in each c FB45 / DB11
    Params_DB :=45
    Params_ADDR:=0

CALL "MOBY FB" , "Instanz-DB K2" FB45 / DB12
    Params_DB :=45
    Params_ADDR:=50

A M 1.2 // memory bit init_run for 1st SLG
FP M 2.2
S "MOBY DB".SLG1.init_run DB45.DBX19.3 -- Set: Reset A
JC x01 SM and parameterize again

AN "MOBY DB".SLG1.ready DB45.DBX18.7 -- Command chain
JC x01 has been finished
A "MOBY DB".SLG1.error DB45.DBX18.6 -- Error during
JC x01 command processing has appeared

A M 1.0 //memory bit command_start for 1st SLG
FP M 2.0
S "MOBY DB".SLG1.command_start DB45.DBX19.1 -- Set: startup
in signal for command or command chain

x01: A M 1.3 //memory bit init_run for 2nd SLG
FP M 2.3
S "MOBY DB".SLG2.init_run DB45.DBX69.1 -- Set: Reset A
JC x02 SM and parameterize again

AN "MOBY DB".SLG2.ready DB45.DBX68.7 -- Command chain
JC x02 has been finished
A "MOBY DB".SLG2.error DB45.DBX68.6 -- Error during
command processing has appeared
    
```

SIMATIC Teemu_Moby_D\ 06/18/2009 11:29:45 AM
 SIMATIC 300(1)\CPU 315-2 PN/DP\...\OB1 - <offline>

```

JC      x02
A      M      1.1      //memory bit command_start for 2nd SLG
FP     M      2.1
S      "MOBY DB".SLG2.command_start      DB45.DBX69.1      -- Set: startup
                                                signal for command or command cha
                                                in
x02:   NOP      0
    
```

Network: 2

Painonapilla I4.0 kasvatetaan laskurin arvoa. Arvo tallentuu sanaan DB1.DBW0. Laskurin saavuttaessa arvon neljä tai muistipaikan M4.1 olessa päällä se resetoituu.

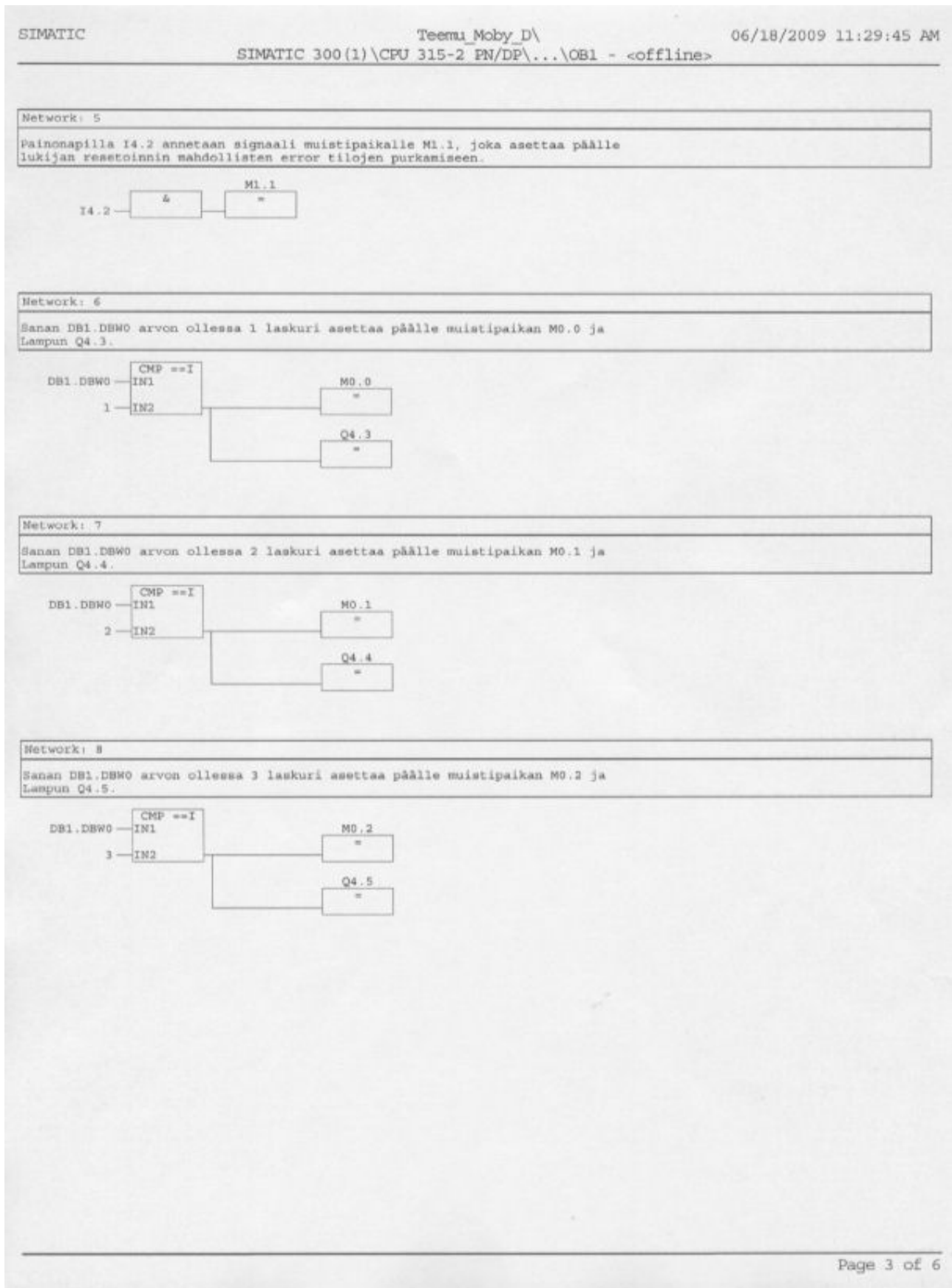
Network: 3

Tieto komennon suorittamisesta ja lukijan virhetilasta tallentuu muistipaikkaan M4.1

Network: 4

painonappi I4.0 asetetaan päälle muistipaikka M1.0, joka antaa käynnistys signaalin RFID-lukijalle SLG1, jonka jälkeen se on valmiudessa kirjoittaa tietoa tunnistelle.

Page 2 of 6



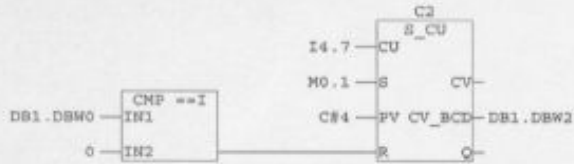
SIMATIC

Teemu_Moby_D\
 SIMATIC 300(1)\CPU 315-2 PN/DP\...\OB1 - <offline>

06/18/2009 11:29:45 AM

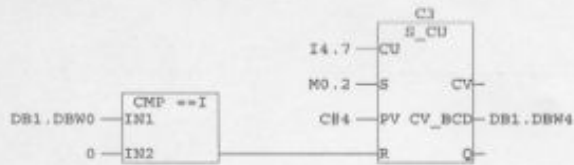
Network: 9

M0.1 asettaa arvon 4 sanalle DB1.DBW2.



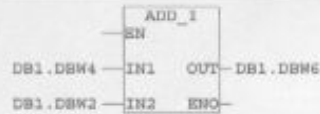
Network: 10

M0.2 asettaa arvon 4 sanalle DB1.DBW4.



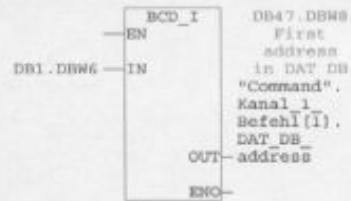
Network: 11

Sanojen DB1.DBW2 ja DB1.DBW4 yhteen laskettu summa tallentuu sanaan DB1.DBW6



Network: 12

DB1.DBW6 arvo muunnetaan oikeaan muotoon sanalle DB47.DBW8, joka ilmoittaa ensimmäisen tavun, josta tietoa aletaan lukea.



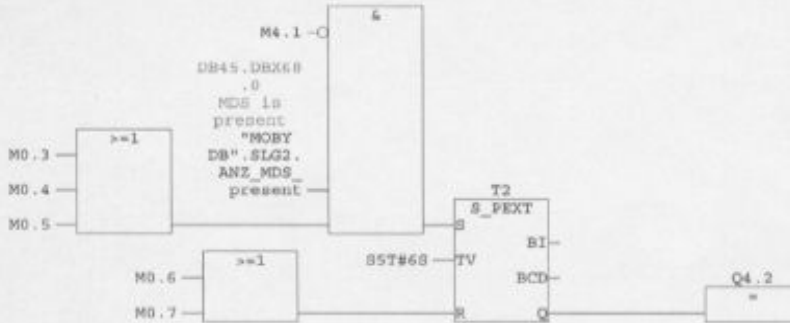
SIMATIC

Teemu_Moby_D\
 SIMATIC 300(1)\CPU 315-2 PN/DP\...\OB1 - <offline>

06/18/2009 11:29:45 AM

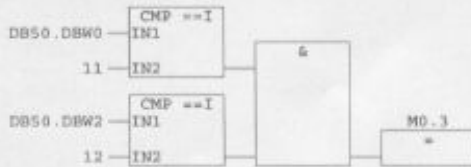
Network: 13

Kun RFID-lukija havaitsee tunnisteen läsnäolon ja on lukenut sillä olevan tiedon kuljetin menee ajastin T2 päälle ja pyörittää kuljetinta Q4.2 6 sekuntia.

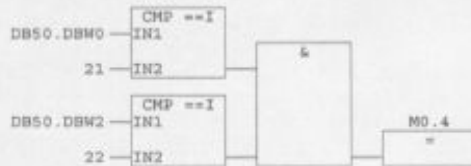


Network: 14

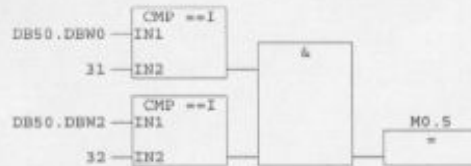
Kun luetun tiedon sanojen DB50.DBW0 ja DB50.DBW2 arvot ovat yhtäsuuret vertailuarvojen kanssa, tieto tallentuu muistipaikkoihin M0.3, M0.4 tai M0.5



Network: 15



Network: 16



SIMATIC

Teemu_Moby_D\
SIMATIC 300(1)\CPU 315-2 PN/DP\...\OB1 - <offline>

06/18/2009 11:29:45 AM

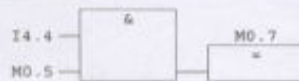
Network: 17

Kun anturi I4.3 tunnistaa kappaleen kuljettimessa 1, se antaa signaalin muistipaikalle M0.6.



Network: 18

Kun anturi I4.4 tunnistaa kappaleen kuljettimessa 1, se antaa signaalin muistipaikalle M0.7.



Network: 19

Muistipaikan M0.6 ollessa tilassa 1, menee ajastin T3 päälle ja pyörittää kuljetinta Q4.1 3 sekuntia



Network: 20

Muistipaikan M0.7 ollessa tilassa 1, menee ajastin T4 päälle ja pyörittää kuljetinta Q4.0 3 sekuntia

