



TAMPEREEN
AMMATTIKORKEAKOULU

KONEAUTOMAATION ELEKTRONII- KAN OPETUKSEN TUKIMATERIAALIN KEHITTÄMINEN

Jouni Jurmu

Opinnäytetyö
Syksy 2017
Konetekniikka



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Konetekniikka
Koneautomaatio

JURMU, JOUNI:

Koneautomaation elektroniikan opetuksen tukimateriaalin kehittäminen

Opinnäytetyö 36 sivua, joista liitteitä 1 sivua
Syksy 2017

Opinnäytetyön tavoitteena oli koota tukimateriaali koneautomaation suuntautumispolun elektroniikan ja mikro-ohjaintekniikan opetukseen siten, että elektroniikan opetus antaa tarvittavat valmiudet mekatroniikan kursseille. Tukimateriaalin tuli sisältää analogia-elektroniikkaan, digitaalielektroniikkaan ja mikro-ohjaimen ohjelmointiin liittyvää teoriaa, esimerkkejä sekä kirjallisuus- ja tietolähtelinkkejä. Tukimateriaalin kohderyhmäksi valittiin koneautomaation suuntautumispolun elektroniikan opettajat.

Työn tuloksena syntyi yli 100-sivuinen tukimateriaali, jossa on käsitelty koneautomaation näkökulmasta tärkeimpiä analogia- ja digitaalielektroniikan komponentteja ja kytkentöjä, sekä mikro-ohjaimen ohjelmointiin ja hyödyntämiseen liittyviä asioita. Materiaaliin kerättiin esimerkkikytkentöjä, joita voidaan rakentaa ja mitata oppituntien yhteydessä. Tukimateriaali liitteineen on poistettu julkisesta raportista.

Tukimateriaalin sisältö määräytyi aloitushetken tietojen perusteella. Kun kurssijako ja kurssien sisällöt kehittyvät, tukimateriaalin sisältöäkin on syytä muokata muuttuneiden tarpeiden mukaan. Mikro-ohjaimen ohjelmointi -osiossa käsitellään Arduino Mega 2560 R3 -alustan käyttöä. Monet ohjelmaesimerkit ovat sovellettavissa useille muillekin Arduino-alustoille lähes sellaisenaan. Tukimateriaalia tai elektroniikan opetusta kehitettäessä pitää ottaa huomioon, että kurssi sisältää teorian lisäksi käytännön harjoituksia, joissa yksinkertaisia kytkentöjä simuloidaan, rakennetaan ja mitataan. Analogia- ja digitaalielektroniikan sekä mikro-ohjainten opetus yhden kurssin aikana siten, että opetetaan teoria ja tehdään teoriaan liittyviä käytännön harjoituksia, on todennäköisesti haastavaa.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Mechanical Engineering
Machine Automation

JURMU, JOUNI:

Development of support material for teaching of electronics in machine automation curriculum

Bachelor's thesis 36 pages, appendices 1 pages
August 2017

The purpose of this thesis was to collect material for electronics and microcontroller course to support teaching in machine automation curriculum. The main emphasis of the support material is to ensure that students will have good knowledge of electronics and microcontrollers before taking part into mechatronics courses. The support material aimed to serve teachers of electronics in machine automation curriculum.

The contents of the support material have been defined in the beginning based on the earlier course feedback and future objectives for the course. Courses and course contents may change, in course of time, and the support material should be updated regarding the changed needs and the requirements. The main emphasis in the microcontroller programming section is to give some examples how to utilize Arduino Mega 2560 R3 development platform. However, the examples can be used in other Arduino development platforms with or without minor modifications.

Support material contains the information and examples of the most important analog and digital electronics components. There are also applications as well as programming examples and usage of a microcontroller. Support material is not the public report.

Key words: electronics, arduino, digital technology, machine automation, microcontroller

SISÄLLYS

1	JOHDANTO.....	6
2	ELEKTRONIIKKA, MEKATRONIIKKA JA SULAUTETUT JÄRJESTELMÄT	7
2.1	Elektroniikka, mekatroniikka ja sulautetut järjestelmät	7
2.2	Elektroniikan opetus koneautomaation suuntautumispolulla	12
3	TUKIMATERIAALIN TEKEMINEN	14
3.1	Prosessi	14
3.2	Kirjallisuusselvitys.....	15
3.3	Analogiaelektroniikka.....	16
3.3.1	Operaatiovahvistimet	17
3.3.2	Aktiiviset suotimet	18
3.3.3	Virtalähde, tasasuuntaus.....	21
3.3.4	Transistori kytkimenä.....	21
3.4	Digitaalitekniikka.....	24
3.4.1	Perusportti- ja sekvenssipiirit.....	24
3.4.2	Karnaugh'n kartta.....	24
3.5	Mikro-ohjain, Arduino Mega 2560.....	26
3.5.1	Tavoitteet Arduino-ohjeistukselle.....	26
3.5.2	I2C-väylä ja LCD-näyttö.....	26
3.5.3	Ajastimet ja keskeytykset.....	27
3.5.4	DC-moottorin ohjaus.....	29
3.5.5	Anturit	30
3.5.6	Digitaali-analogia muunnin.....	30
4	POHDINTA.....	33
	LÄHTEET.....	34
	LIITTEET	36
	Liite 1. Ohjelmakoodin mallipohja	36

LYHENTEET JA TERMIT

ADC	Analog to digital converter, analogia-digitaali muunnin, AD-muunnin
DAC	Digital to analog converter, digitaali-analogia muunnin, DA-muunnin
I2C	I ² C-väylä, mikropiirien välisen tiedonsiirron väylä
IC	Integrated Circuit, mikropiiri
IIC	I ² C-väylä, Inter-IC
PWM	Pulse width modulation, pulssinleveysmodulaatio
TWI	Two Wire serial Interface, Atmelin käyttämä nimitys I ² C-väylästä

1 JOHDANTO

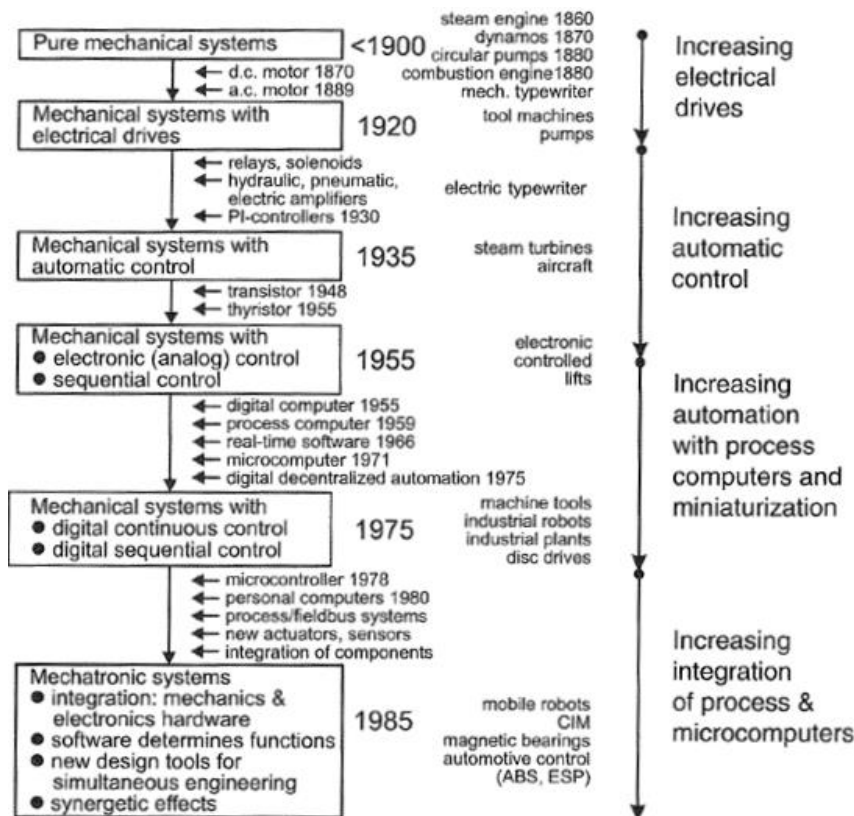
Elektroniikan ja mikro-ohjaintekniikan opintojakso on tärkeä osa koneautomaation suuntautumispolun ammatillisia opintoja. Mekatroniset laitteet, kuten esimerkiksi nykyaikaiset autot, sisältävät elektroniikkaa ja mikro-ohjainjärjestelmiä. Moniin elektroniikan sovelluksiin on saatavilla valmiita elektroniikkamoduuleita, jotka on suunniteltu tiettyä tarkoitusta varten. Aina ei kuitenkaan sopivaa valmista elektroniikkamoduulia ole olemassa, jolloin on suunniteltava vaatimukset täyttävä elektroniikkakytkentä ja testattava sen toimivuus. Elektroniikan sopimusvalmistaja voi tehdyn prototyypin ja mahdollisten lisävaatimusten pohjalta valmistaa elektroniikkamoduulin laitteen massatuotantoa varten. Analogia- ja digitaalelektroniikan osaamisen tärkeys korostuu, mikäli ohjauselektroniikkaa on tarkoitus suunnitella itse tai tilata elektroniikan sopimusvalmistajalta.

Opinnäytetyön tavoitteena oli tuottaa tukimateriaali konetekniikan opintosuunnan elektroniikan ja mikro-ohjaintekniikan opetukseen. Tukimateriaalin kohderyhmänä on elektroniikan ja mikro-ohjain tekniikan opettajat. Tukimateriaalin avulla opettajat voivat suunnitella kurssin opetusmateriaalin ja kurssilla tehtävät elektronisten kytkentöjen mitaukset. Mikro-ohjaintekniikan osalta tukimateriaali sisältää ohjelmointiesimerkkejä, joita voidaan soveltaa opetuksessa ja harjoitustöissä.

2 ELEKTRONIIKKA, MEKATRONIIKKA JA SULAUTETUT JÄRJESTELMÄT

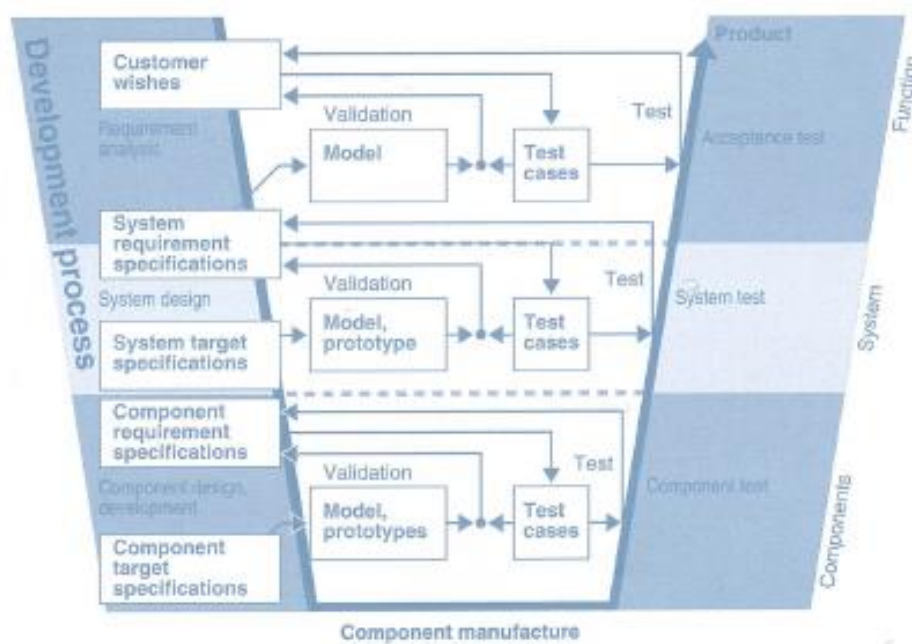
2.1 Elektroniikka, mekatronikka ja sulautetut järjestelmät

Nykyisin elektroniikkaa on melkein laitteessa kuin laitteessa. Kuvassa 1 on esitetty mekaanisten, sähköisten ja elektronisten järjestelmien historiallinen kehitys 1980-luvun puoliväliin saakka. Ensin oli vain pelkkiä mekaanisia laitteita ja asteittain laitteisiin on lisätty sähköistä tai elektronista ohjausta. Kehitys ei ole pysähtynyt 1980-luvun puolivälin tasolle. Euroopan unioni on asettanut rajoitteita vaarallisten aineiden, kuten esimerkiksi lyijyn ja elohopean, maksimimäärään sähkö- ja elektroniikkalaitteissa joita myydään EU-alueelle. Euroopan unionin RoHS direktiivissä on määriteltä tiettyjen vaaralliseksi luokiteltujen aineiden maksimipitoisuudet elektroniikkalaitteissa (European Commission. 2017.) RoHS direktiivi on osaltaan vaikuttanut elektroniikan komponenttien ja laitteiden valmistukseen. Mikroprosessorien ja mikrokontrollerien tehokkuus on kasvanut valmistusteknologian kehityksen myötä.



KUVA 1. Mekaanisten, sähköisten ja elektronisten järjestelmien historiallinen kehitys (Bishop. 2006)

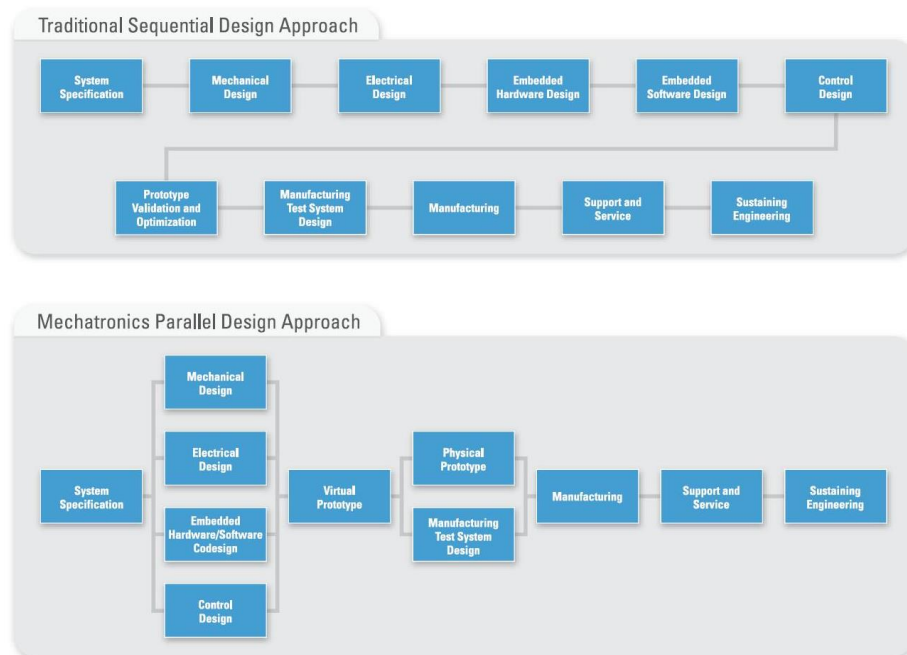
1980-luvun jälkeen myös tuotekehitysprosessit ja erilaiset mallinnustyökalut ovat kehittyneet edelleen. Kuvassa 2 on Boschin näkemys V-mallin mukaisesta tuotekehityksen prosessista. V-malli on yksi järjestelmäsuunnittelun (systems engineering) malleista (Incose. 2016). Boschin prosessi on jaettu kolmeen eri tasoon: toiminto (Function), järjestelmä (System) ja komponentti (Component). Jokaisella prosessin tasolla, tehdään rinnakkaista suunnittelua mm. mekaniikan, elektroniikan ja ohjausjärjestelmien osa-alueilla (Bosch. 2007).



KUVA 2. Tuotekehityksen vaiheet V-mallin mukaisesti (Bosch. 2007)

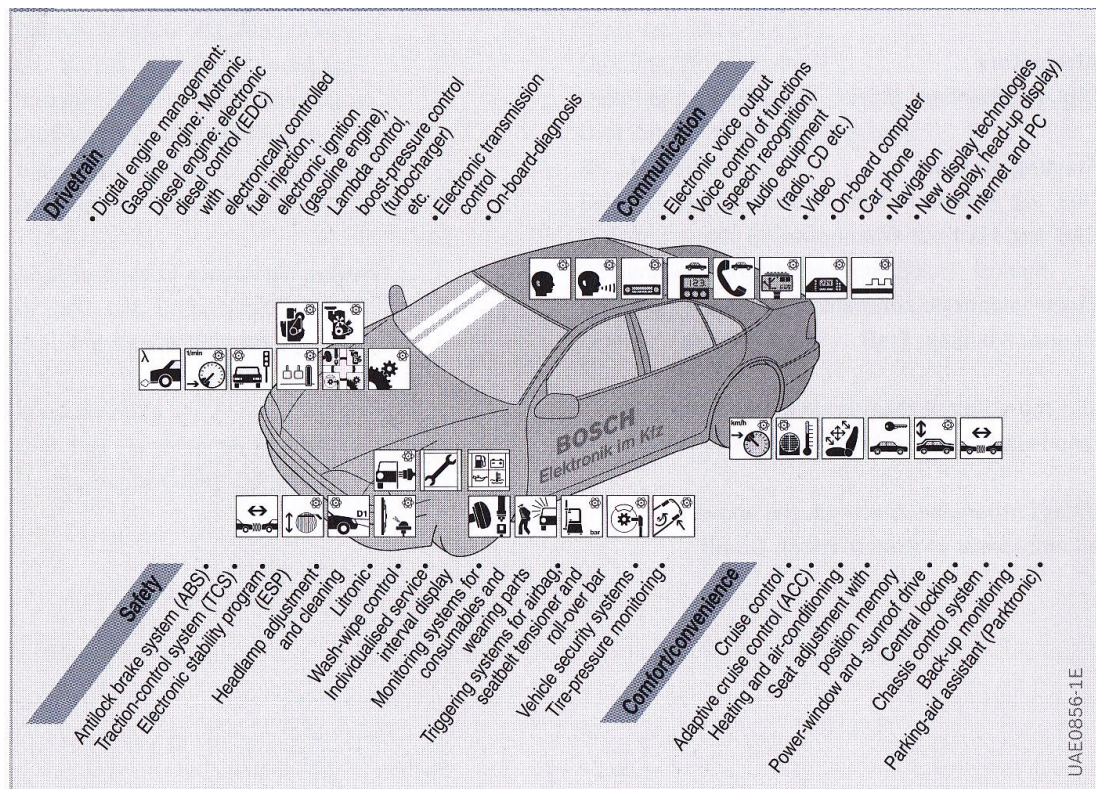
Kuvassa 3 on National Instrumentsin näkemys mekatronisten järjestelmien tuotekehitysprosessien kehityksestä. Perinteinen, vesiputousmallinen, lähestymistapa on tehdä tuotteeseen liittyvien osa-alueiden suunnittelu peräkkäin, tietyssä ennalta määrättyssä järjestyksessä. Prosessi toimii hyvin silloin, jos tuotteen määrittely tai vaatimukset eivät tuotekehityksen aikana muutu. Tuotteen määrittelyn tai vaatimusten muuttuessa, tai jos tuotteen valmistuksen jälkeen tuotteessa havaitaan virhe, pahimmassa tapauksessa koko tuotekehitysprosessi joudutaan käymään uudelleen läpi alusta alkaen. Rinnakkaisella suunnitteluprosessilla pyritään nopeuttamaan tuotekehitystä ja hallitsemaan tuotekehityksen aikaiset muutokset tuotteeseen siten, ettei koko suunnittelu prosessia tarvitse käynnistää uudelleen alusta ja mahdolliset ongelmakohdat huomataan nopeasti.

National Instrumentsin esittämässä suunnitteluprosessissa on mainittu virtuaalinen prototyyppi. Virtuaalinen prototyyppi tehdään mallinnustyökaluja hyödyntäen. Mallinnustyökaluja ovat esimerkiksi MATLAB/Simulink tai LabView. Näiden työkalujen lisäksi tarvitaan laitteesta tai laitteen osista 3D-mallit, jotka tehdään sopivalla 3D-mallinnustyökalulla.

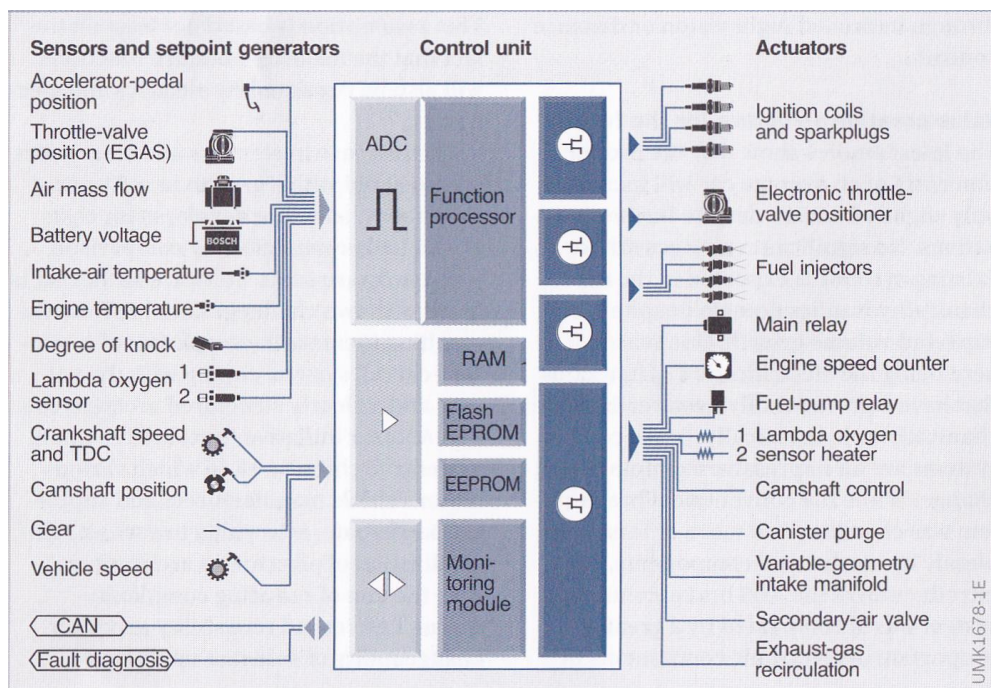


KUVA 3. Perinteinen suunnitteluprosessi ja mekatroniikan rinnakkainen kehitysprosessi (National Instruments. 2012)

Uusissa autoissa on paljon elektroniikkaa ja mikro-ohjaimilla ohjattuja toimintoja. Kuvassa 4 on listattu useimmissa nykypäivän autoissa olevaa elektroniikkaa neljään eri luokkaan jaoteltuna. Autoissa elektroninen ohjausjärjestelmä on jaettu pienempiin osajärjestelmiin. Jokainen osajärjestelmä ohjaa jotain tiettyä toimintoa tai toimintokokonaisuutta. Kuvassa 5 on esitetty tarkemmin Bosch Motronic -moottorinhallintajärjestelmän järjestelmälohkot. Kuvassa ohjausyksikkö (control unit) sisältää sekä mikro-ohjaimen että toimilaitteiden ohjaamiseen tarvittavan elektroniikan. (Bosch. 2007)



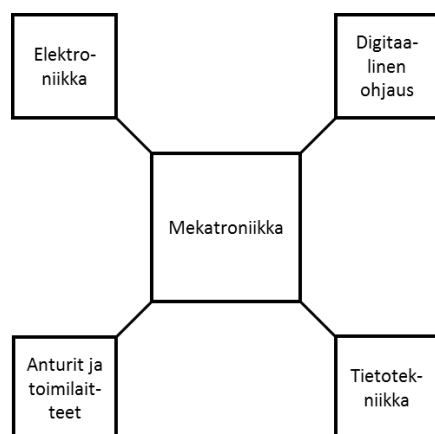
KUVA 4. Moottoroidun ajoneuvon elektroniikkaa (Bosch. 2007)



KUVA 5. Bosch Motronic -moottorinhallintajärjestelmän järjestelmälohkot (Bosch. 2007)

Onwubolun ja Airilan mukaan mekatroniikalle on olemassa useita erilaisia määritelmiä ja koulukuntia (Onwubolu. 2005, Airila. 1999). Useimmiten mekatroniikka määritellään mekaniikan, elektroniikan, ohjausjärjestelmien ja tietotekniikan yhdistämiseksi siten, että voidaan suunnitella ja valmistaa laadukkaita lopputuotteita (Onwubolu. 2005). Airilan mukaan termit koneautomaatio ja mekatroniikka olisivat lähellä toisiaan (Airila. 1999). Bolton määrittelee mekatronisen järjestelmän siten, että se on elektronisten-, mekaanisten- ja ohjausjärjestelmien täydellinen yhdistäminen. (Bolton. W. 2003, Bishop, R. 2006)

Mikä sitten olisi absoluuttisen oikea määritelmä mekatroniikalle? Onwubolun ja Boltonin määritelmät, joko yhdessä tai erikseen, kuvaavat hyvin mitä mekatronisten järjestelmien kehitys on. Pitäisikö mekatroniikan rinnalla puhua järjestelmäsuunnittelusta, systems engineering? Mekatroniikan projekteissa kuitenkin suunnitellaan järjestelmiä. Kuvassa 6 on esitetty mekatronisen järjestelmän pääkomponentit.



KUVA 6. Mekatronisen järjestelmän pääkomponentit (Onwubolu 2005, muokattu)

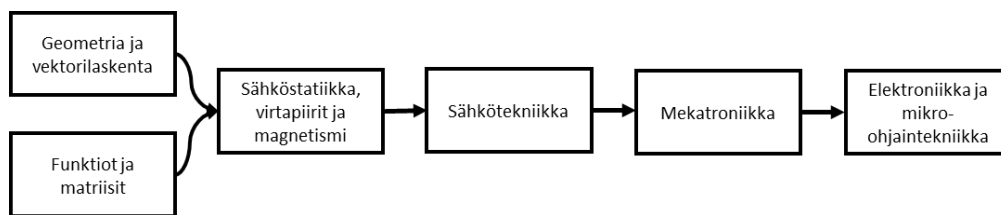
Onwubolu listaa kirjassaan mekatroniikan esimerkkeinä mm. lentosimulaattorit, manipulaattorit, mukautuva jousitus, videonauhurit ja ilman kuljettajaa kulkevat ajoneuvot. Esimerkkilistaan voi lisätä vielä esimerkiksi pyykinpesu- ja kuivauskoneet, autot sekä metsä- ja muut työkoneet. Kaikissa edellä mainituissa on antureita, ohjaustekniikkaa ja -elektroniikkaa ja mekaniikkaa.

Sulautetuista järjestelmistä on niin ikään monia määritelmiä. Jari Koskinen määrittelee sulautetun järjestelmän laitteiksi, joita käytetään johonkin erityistarkoitukseen ja joissa mikrotietokone on osana laitteen elektroniikkajärjestelmää (Koskinen, J. 2006).

2.2 Elektroniikan opetus koneautomaation suuntautumispolulla

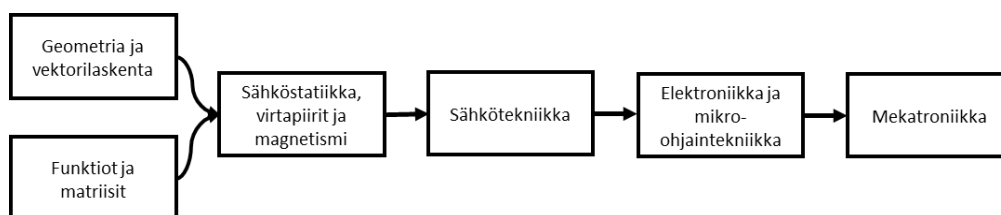
Opiskelu ammattikorkeakoulussa on kuin projekti, jossa opintojaksot ovat projektin tehtäviä ja joilla on riippuvuus toisistaan. Eri ammattiaineiden opintojaksot ovat riippuvaisia eri tavalla aiemmista opintojaksoista ja niiden sisällöistä. Elektroniikan opintojakso edellyttää, että opiskelijalla on riittävät pohjatiedot matematiikasta, fysiikasta ja sähkötekniikasta. Digitaalielektroniikkaan kuuluvia asioita loogisten toimintojen osalta käsitellään, ainakin osittain, ohjelmoitavia logiikkoja ja -releitä käsittelevillä opintojaksoilla. Mikro-ohjaintekniikka edellyttää perustason ohjelmointiosaamista, jota ei koneautomaation opintopolkuun tarkasteluhetkellä sisältynyt omana opintojaksonaan.

Kuvassa 7 on esitetty opintojaksojen välinen riippuvuus vuonna 2013 tai aiemmin aloittaneiden koneautomaation opintopolun opintosuunnitelman mukaan (Tampereen ammattikorkeakoulu). Kuvassa ei oteta kantaa matematiikan opintojaksojen keskinäiseen järjestykseen.



KUVA 7. Elektroniikan ja mikro-ohjain tekniikka -opintojakson sijoittuminen vanhan opintosuunnitelman mukaan (Tampereen ammattikorkeakoulu)

Kuvassa 8 on elektroniikka ja mikro-ohjaintekniikka -opintojakson ehdotettu parempi sijoittuminen kurssien järjestyksen osalta. Tässä ehdotuksessa mekatroniikan ja elektroniikan opintojaksojen keskinäinen järjestys on muutettu päinvastaiseksi. Ehdotus pohjautuu ajatukseen, että elektroniikan ja mikro-ohjaintekniikan opintojakso säilyisi opintosuunnitelmassa sellaisenaan tai sisällöllisesti muokattuna, ainoastaan ajoitus muuttuisi.



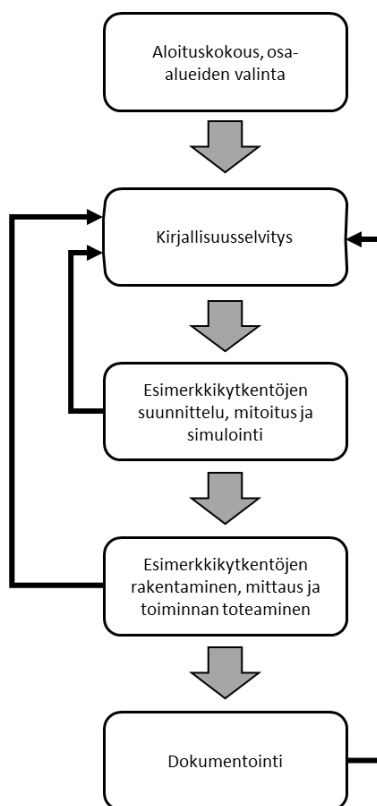
KUVA 8. Elektroniikka ja mikro-ohjain tekniikka -opintojakson parempi sijoittuminen opintopolussa

Jouko Heikkinen on kehityshankkeessaan ”Soveltavan elektroniikan opetussisällön kehittäminen – Muutostarpeet työelämän näkökulmasta” selvittänyt Tampereen teknillisen yliopiston elektroniikan kurssien kehittämistarpeita (Heikkinen. 2014). Kehityshankkeen raportissa on mainittu tärkeinä elektroniikan opetuksen osa-alueina transistorin toiminta kytkimenä, vahvistinkytkennät, suotimet ja komparaattorit. Raportin yhteenveto-osiossa on mainittu useiden elektroniikan opintojaksojen osalta, että tarkasteltavat kytkennät jatkossa simuloidaan, rakennetaan ja mitataan. Simuloinnin, rakentamisen ja mittaamisen tarkoitus on helpottaa teorian omaksumista itse tekemällä. (Heikkinen. 2014)

3 TUKIMATERIAALIN TEKEMINEN

3.1 Prosessi

Tukimateriaalin tekoprosessi on kuvattu kuvassa 9. Aloituskokouksessa sovittiin käsiteltävät osa-alueet, alustava aikataulu ja muut työn tekemisen liittyvät käytännön asiat. Käytännön prosessissa oli palattava usein kirjallisuusselvityksen aktiviteetteihin. Kirjallisuusselvitys ja dokumentointi olivat työmäärällisesti yllättävän suuria.



KUVA 9. Tukimateriaalin tekoprosessi

3.2 Kirjallisuusselvitys

Kirjallisuusselvitysvaiheessa oli tarkoitus kartoittaa lähdemateriaali, niin perinteinen painettu kuin sähköinenkin, jonka voi listata tukimateriaalin lähdemateriaaliksi. Työn edetessä kirjallisuusselvitysvaiheeseen oli aika ajoin palattava, kun havaittiin että jo kartoitetussa materiaalissa ei ollut riittävästi tai riittävän selkeästi kerrottu elektroniikkaan liittyvää aihealuetta.

Kirjallisuusselvitys tehtiin Tampereen ammattikorkeakoulun kirjaston kokoelmista ja saatavilla olevasta e-aineistosta sekä komponenttivalmistajien Internet-sivuilta. Kirjallisuusselvitys ei rajoittunut pelkästään suomenkielisiin teoksiin, vaan mukana oli englanninkieliset teokset sekä komponenttivalmistajien datalehdet ja muut julkaisut.

Analogiaelektroniikan osalta selkeimpiä lähdeoksia ovat Aaltosen (Aaltonen 2004) ja Silvosen (Silvonen 2009) kirjat. Englanninkielisiä analogiaelektroniikan lähdeoksia on paljon ja sen vuoksi on haastavaa suositella niistä vain yhtä tai muutamaa. Hyvän englanninkielisen lähdeoksen, joskin jo vanhan, on kirjoittanut Millmann (Millmann 1987). Elektroniikan opettajan onkin syytä valita sopivat lähdeokset opiskelijoille kuhunkin kurssitoteutukseen parhaiten sopivaksi.

Digitaalelektroniikan osalta lähdekirjoina voi käyttää Aaltosen ja Silvosen kirjoja tai Haikon kirjaa (Haiko 2012). Englanninkielisenä hyvänä lähteenä voi käyttää Floydin kirjaa (Floyd, 2009).

Analogiaelektroniikasta ja digitaalelektroniikasta löytyy paljon kirjallisuutta sekä perinteisessä painetussa muodossa että sähköisessä muodossa.

Mikro-ohjaimen osalta tärkeitä lähteitä ovat Arduino-kotisivut (Arduino. 2017), aiemmin tehty opinnäytetyö aiheesta (Seipijärvi. 2015) sekä Atmel 2560 -mikrokontrollerin datalehti (Atmel. 2014). Arduino-kotisivuilla on tietoa Arduino-laitteista, Arduinon ohjelmointikielestä sekä ohjelmointiesimerkkejä. Arduino mikrokontrollerin ohjelmointi tehdään C-kielellä, ja suositeltavaa on, että ennen Arduinon ohjelmointia on olemassa perusteet C-kielellä tehtävästä ohjelmoinnista. Gaddisin kirja (Gaddis, 2015) on hyvä lähde C ja C++ kielen alkeisiin.

Suomenkieliseltä Internet-sivulta, www.ohjelmointiputka.fi, löytyy ohjeita ja opastusta C-kielellä tehtävään ohjelmointiin. Joissain tapauksissa tarvitaan Arduinoon liitettävistä lisälaitteista, kuten esimerkiksi antureista, enemmän tietoa, jolloin on turvauduttava valmistajien datalehtiin. Arduinosta on saatavilla e-kirjoja Tampereen ammattikorkeakoulun kirjaston Finna-hakuliittymän kautta. Valitettavasti e-kirjojen saatavuus vaihtelee eikä ole takeita kirjan saatavuudesta jatkossa.

3.3 Analogiaelektroniikka

Analogiaelektroniikassa käsitellään komponentteja ja piirejä, joiden jännitetasot voivat vaihdella vapaasti käyttöjännitteen ja tehonkeston rajoissa (Aaltonen, 2004). Analogiaelektroniikan komponentteja ovat vastukset, kondensaattorit, kelat, diodit, tyristorit, triakit, optokomponentit, kuten esimerkiksi LEDit ja optoerottimet, bipolaaritransistorit, kanavatransistorit ja operaatiovahvistimet.

Analogiaelektroniikan osalta tukimateriaaliin valittiin esimerkkejä seuraavista sovelluksista:

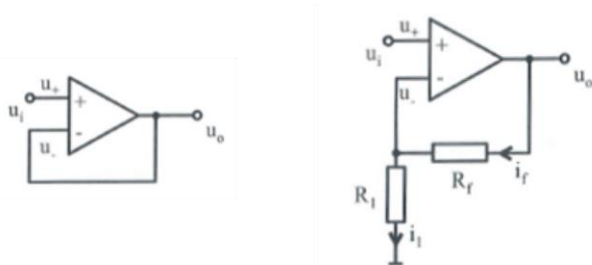
- Operaatiovahvistin
 - Puskurivahvistin
 - Ei-kääntävä vahvistin
- Aktiivisuotimet
 - Alipäästösuodin
 - Ylipäästösuodin
 - Kaistanpäästösuodin
 - Kaistanestosuodin
- Tasasuuntaus, hurinajännite ja jännitteen vakavointi
 - Virtalähde
- Optokomponentit
 - Optoerotin, moottorin ohjaus mikro-ohjaimella, 24V anturin kytkeminen mikro-ohjaimeen
- Transistorit
 - Bipolaaritransistori kytkimenä, H-silta tasavirtamoottorin ohjauksessa

Esimerkkikytkennät suunniteltiin EAGLE-ohjelmalla. EAGLE valittiin piirikaavion suunnitteluohjelmaksi, koska mm. Arduino-kehitysalustojen piirikaaviot oli suunniteltu sillä ja EAGLE oli ilmainen työkalu.

3.3.1 Operaatiovahvistimet

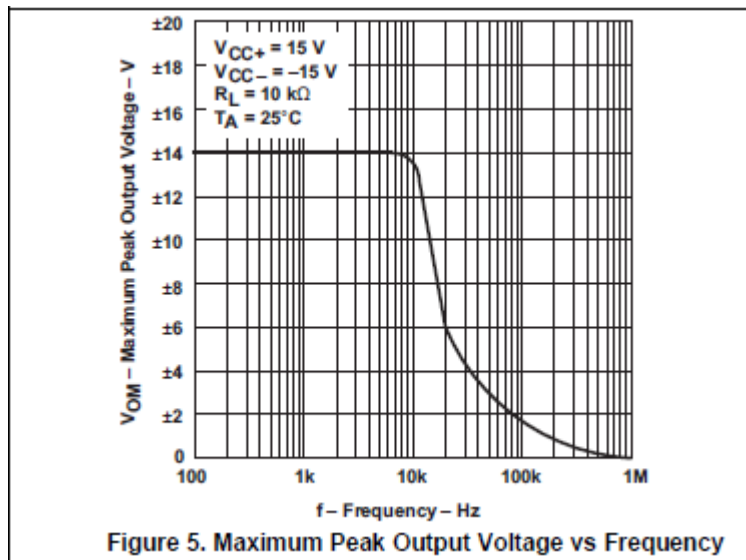
Operaatiovahvistinkytkennät toteutettiin edullisella yleiskäyttöisellä $\mu A741$ operaatiovahvistimella. Operaatiovahvistinpiiri 741 tuli markkinoille 1966 ja nykyisin monet elektroniikkavalmistajat valmistavat 741-operatiovahvistinta (Millmann. 1987). Koekytkentäalustalle rakennettiin kuvan 10 mukaiset operaatiovahvistinkytkennät. Kuvassa vasemmanpuoleinen kytkentä on jännitteenseuraaja. Jännitteen seuraajaa kutsutaan usein myös puskurivahvistimeksi, jolloin sitä käytetään erottamaan kaksi järjestelmän eri osaa toisistaan siten, etteivät ne kuormita toisiaan (Aaltonen. 2004). Kuvan oikeanpuoleinen kytkentä on ei-invertoiva vahvistin, jonka vahvistusta voidaan säätää vastuksien R_1 ja R_f avulla. Vahvistimen vahvistus saadaan laskettua kaavasta 1.

$$A_V = 1 + \frac{R_f}{R_1} \quad (1)$$



KUVA 10. Operaatiovahvistinkytkennät (Aaltonen, 2004. Muokattu.)

Jännitteen seuraajalla mitattiin tasajännitevaste, miten vahvistimen lähtöjännite muuttuu tulojännitteen funktiona. Mittaustulokset tallennettiin ja tuloksia vertailtiin kuvan 11 käyrään ja todettiin, että mittaustulos on linjassa käyrän kanssa.



KUVA 11. Operaatiovahvistimen lähtöjännitteen maksimi taajuuden funktiona (Texas Instruments)

Ei-invertoivalla vahvistimella mitattiin vaihtojännitevaste ja ennalta määritellyn vahvistuksen paikkansapitävyys sinimuotoisella signaalilla. Mittaus tehtiin oskilloskooppia ja funktiogeneraattoria käyttäen muutamalla eri taajuisella sisäänmenosignaaliilla. Mittauksessa verrattiin myös vahvistimen sisäänmenoon syötetyn signaalin ja ulostulosta mitatun signaalin muotoja keskenään. Signaalien vertailulla varmistettiin, ettei signaali vääristy tai säröydy vahvistettaessa.

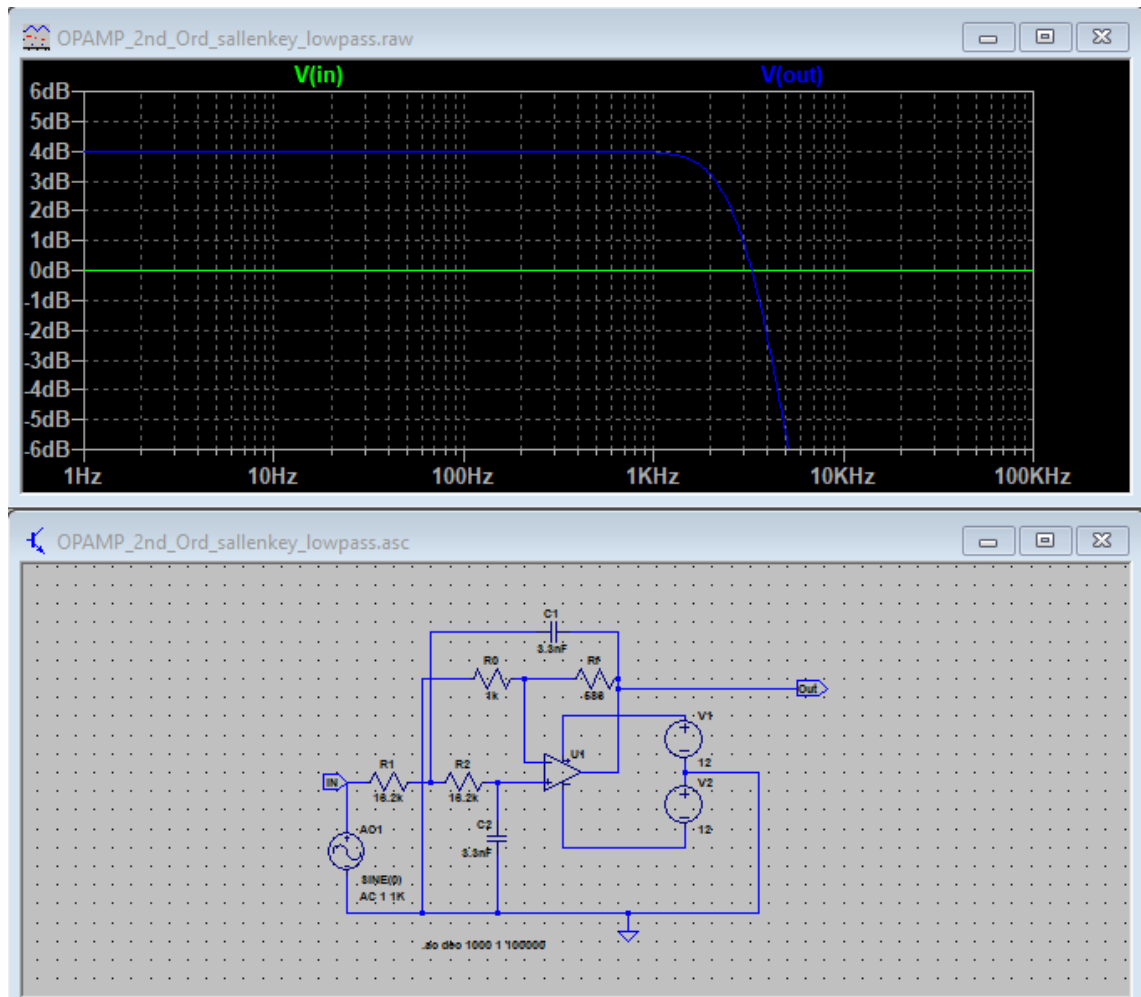
3.3.2 Aktiiviset suotimet

Aktiivisten suotimien osalta oli vapaus valita sopivat kytkentäesimerkit mittauksia varten. Ali- ja ylipäästösuotimet mitoitettiin ja rakennettiin Sallen-Keyn toisen kertaluvun suodattimiksi. Kaistanpäästösuodin mitoitettiin ja rakennettiin Sallen-Keyn kaistanpäästösuotimeksi. Kaistanestosuodin mitoitettiin ja rakennettiin kaksois-T kytkennän avulla. Mitoitusohjeina käytettiin Aaltosen kirjan ohjeistusta.

Ennen suodinkytkeiden rakentelua ja mittausta kytkentä simuloitiin ilmaisella LTSpice-ohjelmalla, jotta voitiin varmistua siitä, että laskettu rajataajuus tai keskitaajuus voidaan saavuttaa valituilla komponenttiarvoilla. LTSpice-ohjelmaa käytettiin, koska se on ilmainen ja suhteellisen yksinkertainen käyttää. Vaihtoehtoisesti olisi voinut käyttää simulointiin Micro-Cap-ohjelmaa, josta saa myös ilmaisen kokeiluversion käyttöön.

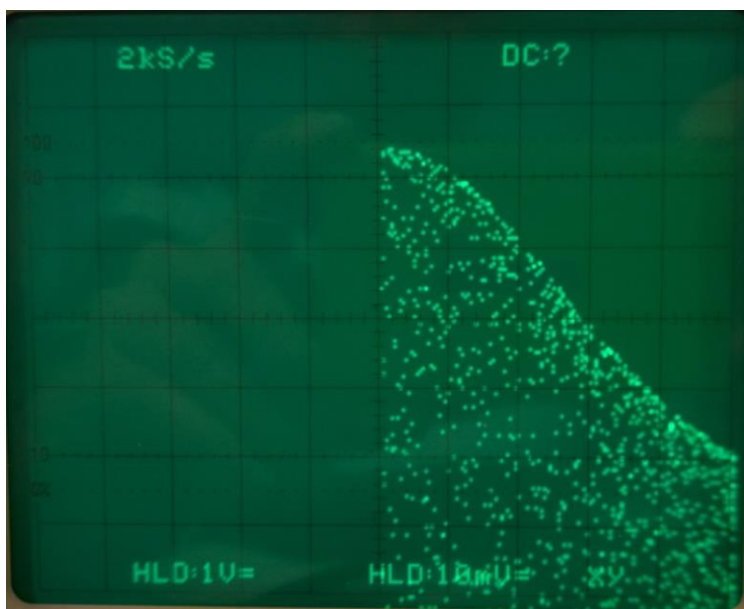
Simulointia tarvittiin erityisesti kaistanpäästösuotimen vahvistuksen ja mittauksen suunnitteluun. Simuloinnilla tarkistettiin eri vahvistusvaihtoehtojen vaikutus kaistanpäästösuotimen keskitaajuuden vahvistukseen. Simulointikierroksilla kokeiltiin erilaisia vahvistuksia ja laskettiin keskitaajuuden vahvistuksen perusteella kaistanpäästösuotimen ulostulon jännite. Mikäli laskettu ulostulon jännite ylitti kuvan 11 maksimi lähtöjännitteen, piti joko säätää vahvistusta pienemmäksi tai pienentää sisään menevän signaalin amplitudia.

Kuvassa 12 on Sallen-Keyn suotimella toteutetun toisen kertaluvun alipäästösuotimen simulointitulokset, kun suotimen vahvistus on mitoitettu Butterworth-suotimen mukaan. Alipäästösuotimen rajataajuus, -3 dB taajuus, on kuvan esimerkissä 3 kHz.



KUVA 12. Alipäästösuodinkytken simulointi LTSpice-ohjelmalla

Kun simuloinnin avulla kytkennät todettiin toimiviksi, rakennettiin kytkennät koekytken-
tällevylle. Suotimien mittausta funktiogeneraattoria ja oskilloskooppia käyttäen. Kun sisään
menevän signaalin taajuutta muutettiin funktiogeneraattorista, varmistettiin että sisään
menevän signaalin taso pysyy samana koko mittauksen ajan. Mittaustulokset taulukoitiin
ja niistä piirrettiin vastekäyrä. Alipäästösuodin mitattiin lisäksi käyttäen pyyh-
käisygeneraattoria ja X-Y-mittaukseen soveltuvaa oskilloskooppia, jolloin alipäästösuoti-
men vaste voitiin suoraan lukea oskilloskoopin näytöltä kuvan 13 mukaisesti. Pyyh-
käisygeneraattorin taajuus vaihteli 0 Hz – 5 kHz välillä. Kuva on rakeinen siksi, että os-
killoskoopin näytteenottotaajuus oli matala ja näytemuisti oli kooltaan pieni. Mittauk-
sessa käytettiin vanhaa analogia-digitaali oskilloskooppia. Kuvan epätarkkuudesta huoli-
matta näytöltä voidaan lukea alipäästösuotimen taajuusvaste.



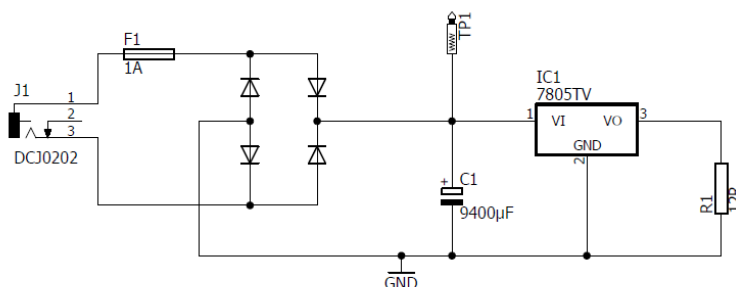
KUVA 13. Alipäästösuotimen taajuusvaste X-Y-mittauksena oskilloskoopilla

Alipäästösuotimen X-Y-mittausta varten piti varmistaa, että pyyhkäisygeneraattorista
saadaan oikeanlainen signaali X-Y-mittausta varten. Ideaalitapauksessa signaali olisi
saha-aalto, joka muuttuu lineaarisesti esimerkiksi -5 V tasosta +5 V tasolle.

3.3.3 Virtalähde, tasasuuntaus

Tavoitteena oli suunnitella ja rakentaa virtalähde, jossa on jännitelähdöt 24 V ja 5 V. Suunnittelu sisälsi myös tarvittavien komponenttien valinnan. Jännitelähteenä käytettiin tehdasvalmisteista säädettävää AC/AC-adapteria, joka kytkettiin rakennettuun virtalähteeseen liittimen kautta. Tukimateriaalissa on virtalähteen vaihtoehtoiset kytkentäkaaviot.

Jänniteregulaattorien datalehdillä oli esimerkkikytkennöissä korkeataajuisen värähtelyn estäviä kondensaattoreita. Käytännön kytkennässä nämä kondensaattorit kuitenkin toimivat korkeataajuisen värähtelyn aiheuttajana. Korkeataajuinen värähtely aiheutti regulaattorikomponenttien voimakkaan lämpenemisen ja regulaattorin sisäisen lämpötilasuojauksen kytkeytymisen päälle, joka asteittain laski regulaattorin lähtöjännitteen nolllaksi. Kun korkeataajuinen värähtely saatiin poistettua, niin regulaattori toimi kuten pitääkin. Kuvan 14 mukaisella kytkennällä mitattiin virtalähteen hurinajännite. Hurinajännite mitattiin testipisteestä TP1. Laskennallinen kuormitusvirta 12Ω kuormitusvastuksen kautta on 0,42 A. Jotta hurinajännite saatiin oskilloskoopilla mitattua, niin AC/AC-adapterilta saatavan vaihtojännitteen huippuarvoksi asetettiin noin 10 V.

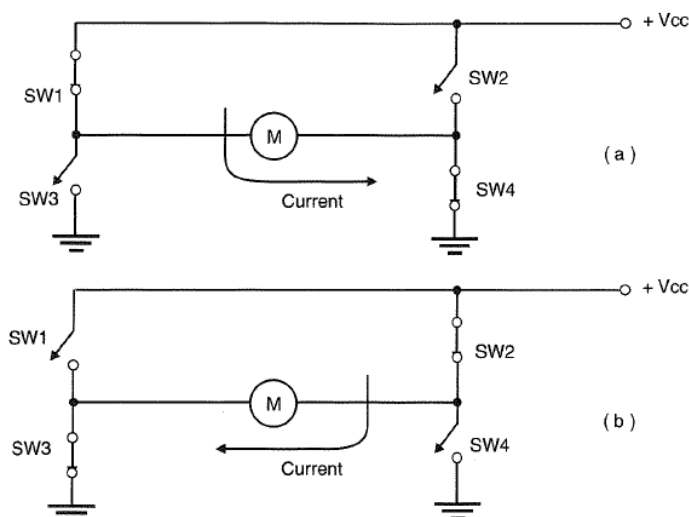


KUVA 14. Virtalähteen hurinajännitteen mittaus 12Ω keinokuormalla

3.3.4 Transistori kytkimenä

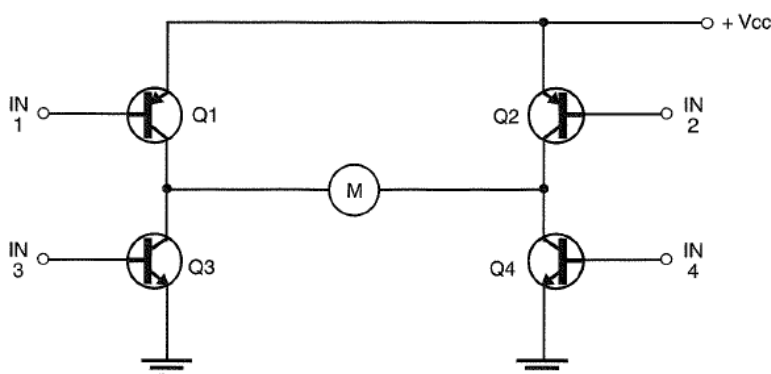
Transistoria voidaan käyttää joko vahvistimena tai kytkimenä. Työssä keskityttiin transistorin käyttöön kytkimenä ja esimerkkinä transistorin kytkin käytöstä oli tasavirtamootorin ohjaus H-sillan avulla. Transistorin käyttö kytkimenä on nostettu esiin eräänä nyky-suunnittelun kannalta tärkeimmistä aiheista Jouko Heikkisen kehittämishankkeen raportissa (Heikkinen. 2014). H-silta kytkennällä voidaan moottori ohjata pyörimään molempiin suuntiin.

Kuvassa 15 on periaatekuva H-sillasta kytkimillä toteutettuna. Kun kytkimet SW1 ja SW4 ovat kiinni ja kytkimet SW2 ja SW3 ovat auki moottori pyörii pyörimissuuntaan 1. Kun kytkimet SW1 ja SW4 ovat auki ja kytkimet SW2 ja SW3 ovat kiinni moottori pyörii pyörimissuuntaan 2.



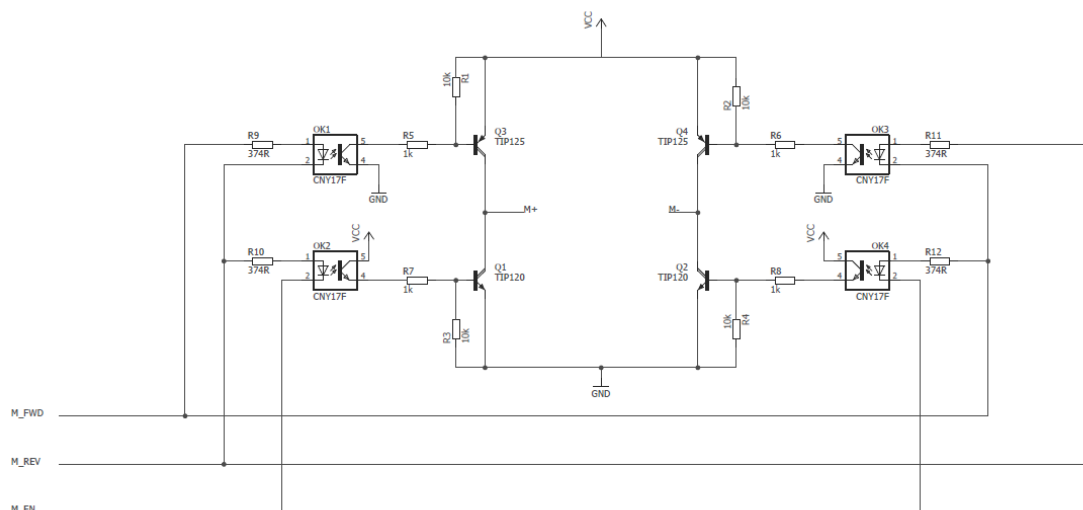
KUVA 15. H-siltakytkennän periaate kytkimiä käyttäen (Braga 2002)

Kuvassa 16 on H-siltakytkentä bipolaaritransistoreita käyttäen. Kuvassa on kytkimet korvattu NPN ja PNP transistoreilla. Kuvan mukaisessa kytkennässä moottorin ohjaukseen tarvitaan neljä ohjaussignaalia. Kuvasta puuttuvat transistorien kantavirtaa rajoittavat vastukset sekä suojadiodit, jotka käytännön kytkennässä pitää olla, jottei transistori tuhoudu.



KUVA 16. H-siltakytkentä bipolaaritransistoreita käyttäen (Braga 2002)

Mikro-ohjaimella ohjattavissa oleva H-silta kytkentäesimerkki löytyi McManisin (McManis C. 2003.) Internet-sivuilta. Kytkennässä olevat optoerottimet mahdollistavat H-sillan ohjauksen mikro-ohjaimella ja toisaalta suojaavat mikro-ohjainta, jos transistorit vaurioituvat siten, että käyttöjännite pääsee suoraan mikro-ohjaimen I/O-liitäntään. Kuvassa 17 on esitetty muokattu H-silta kytkentä. Tasavirtamoottori kytketään M+ ja M- liittimiin. Ohjattavan tasavirtamoottorin maksimivirta moottorin datalehden mukaan on 2,2 A. Kytkentään valittiin Darlinton-transistorit, TIP120 ja TIP125, joiden maksimi virrankesto on 5A, joissa on sisäänrakennettu suojadiodi (On Semiconductor. 2014). Sisäänrakennettu diodi suojaa transistoria induktiivisen kuorman, tässä tapauksessa tasavirtamoottorin, aiheuttamalta transienttipiikiltä.



KUVA 17. H-siltakytkentä optokytkimillä (McManis C. 2003. Muokattu)

Taulukossa 1 on esitetty transistorikytkinten tilat eri ohjaussignaaleilla. Mikäli moottorin pyörimisnopeutta halutaan säätää PWM signaalilla, kytketään PWM signaali M_EN liitäntään. PWM ohjaussignaalin perustaajuus pitää ottaa huomioon, kun optoerottimia tai transistoreita valitaan, jotta voidaan varmistaa, että kytkentä toimii riittävän nopeasti. Kun kaikkien kolmen ohjaustulon signaali on 0 V, niin moottori on vapaa-tilassa ja moottorin akseli pyörii vapaasti. Kun M_FWD ja M_REV ohjaussignaalit kytketään + 5 V ja M_EN ohjaussignaali kytketään 0 V, niin silloin moottorille on laitettu ”jarru” päälle ja moottorin navat ovat kytketty yhteen.

TAULUKKO 1 Moottorin ohjaus H-sillalla

M_FWD	M_REV	M_EN	Q1	Q2	Q3	Q4	M+	M-
0 V	0 V	0 V	auki	auki	auki	auki	-	-
+ 5 V	0 V	0 V	auki	kiinni	kiinni	auki	$V_{CC} - V_{EC_Q3}$	$GND + V_{CE_Q2}$
0 V	+ 5 V	0 V	kiinni	auki	auki	kiinni	$GND + V_{CE_Q1}$	$V_{CC} - V_{EC_Q4}$
+ 5 V	+ 5 V	0 V	kiinni	kiinni	auki	auki	0 V	0 V

Kuvan 17 mukainen kytkentä rakennettiin koekytkentäalustalle, kytkentään liitettiin moottori ja kytkennän toimivuus testattiin. Toimivuus testattiin ensin ilman mikro-ohjainta ja myöhemmässä vaiheessa myös mikro-ohjaimen kanssa.

3.4 Digitaalitekniikka

3.4.1 Perusportti- ja sekvenssiipiirit

Perusporttipiirien yhteydessä on käsitelty ideaalisen ja ei ideaalisen digitaalisen pulssin ominaisuuksia, logiikkatasot digitaalisissa piireissä ja häiriöiden vaikutus tilasignaaliin. Tukimateriaaliin on perusporttipiirien EI (NOT), JA (AND), TAI (OR), EI-JA (NAND), EI-TAI (NOR), poissulkeva TAI (XOR) sekä poissulkeva EI-TAI (XNOR) osalta esitetty piirrosmerkit, totuustaulukko, esimerkkipiiriin jalkajärjestys sekä kuinka perusportti voidaan rakentaa EI-JA perusportteja käyttäen. Sekvenssiipiireistä materiaaliin on otettu mukaan D- ja SR-kiikut. Lisäksi Schmitt-triggerin toiminta on lyhyesti käyty läpi operaatiovahvistin esimerkin ja hystereesikäyrän avulla.

Ajastinpiiri 555 luokitellaan kuuluvaksi digitaalitekniikkaan. Tukimateriaalissa on muutama ajastinpiiriin esimerkkikytkentä sekä niihin liittyvät mitoitusohjeet. Ajastimia käytetään usein ohjelmoitavissa logiikoissa esimerkiksi vikatilanteen havaitsemiseen.

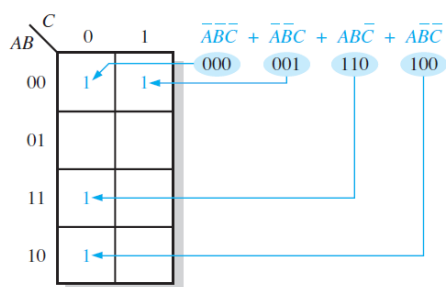
3.4.2 Karnaugh'n kartta

Karnaugh'n karttaa käytetään työkaluna digitaalitekniikan kombinaatiologiikan yksinkertaistamisessa. Karnaugh'n karttaa voidaan soveltaa myös ohjelmoitavien logiikoiden ja releiden logiikka-ohjelman yksinkertaistamiseen. Karnaugh'n kartta ja sen käyttö on tukimateriaalissa selostettu pääsääntöisesti Aaltosen kirjan pohjalta (Aaltonen, 2014).

Taulukossa 2 on totuustaulukko eräästä kolmituloisesta kombinaatiologiikasta. Totuustaulukkoa vastaava Boolean lauseke on $Y = \overline{A} \overline{B} \overline{C} + \overline{A} \overline{B} C + A \overline{B} \overline{C} + A \overline{B} C$. Kuvassa 18 on esitetty, kuinka Boolean lauseke sijoitetaan Karnaugh'n karttaan. Tämän Boolean lausekkeen toteuttamiseen tarvitaan esimerkiksi 4 3-tuloista JA-porttia, 3 2-tuloista TAI-porttia ja 8 EI-porttia.

TAULUKKO 2. Totuustaulukko

A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



KUVA 18. Totuustaulukon sijoittaminen Karnaugh'n karttaan (Floyd. 2009)

Boolean lausekkeen sievennys Karnaugh'n kartan avulla tapahtuu tässä esimerkissä etsimällä mahdollisimman suuria ykkösten muodostamia ryhmiä. Ryhmässä voi olla ykkösiä kahden potenssien mukaan, eli 1, 2, 4, 8 jne. Kuvassa 19 on ryhmät muodostettu ja sievennetyksi Boolean lausekkeeksi saadaan $Y = \overline{A} \overline{B} + A \overline{C}$. Tämän sievennetyn Boolean lausekkeen toteuttamiseksi tarvitaan 2 2-tuloista JA-porttia, 1 2-tuloinen TAI-portti ja 3 EI porttia.

		0	1
AB	00	1	1
	01	0	0
	11	1	0
	10	1	0

KUVA 19. Sievennys Karnaugh'n kartan avulla

Karnaugh'n kartan yhteydessä on lyhyesti käsitelty kombinaatiologiikan mahdollinen viiveistä johtuva virhetilanne, glitch, ja esimerkki, kuinka se voidaan Karnaugh'n kartan avulla havaita ja miten se voidaan eliminoida.

3.5 Mikro-ohjain, Arduino Mega 2560

3.5.1 Tavoitteet Arduino-ohjeistukselle

Arduino Mega 2560 R3 -mikro-ohjainkortin ohjelmointiesimerkkien ainoana ohjeistuksena oli, ettei se saa olla samanlainen, kuin mitä aiemmin on jo opinnäytetyönä Tampereen ammattikorkeakoululle tehty (Seipijärvi. 2015). Päällekkäisyyttä aiemmin tehtyyn opinnäytetyöhön ei voinut täysin välttää ja siksi joitain samoja asioita onkin tarkasteltu hieman eri näkökulmasta.

Ohjelmoinnissa on hyvä noudattaa hyvää ohjelmointitapaa, joka tarkoittaa mm. sitä, että ohjelmakoodirivit on kommentoitu ja muuttujien nimet ovat selkeitä. Ohjelmointiharjoituksia varten luotiin ohjelmakoodille mallipohja (Liite 1), joka ohjaa selkeän ohjelmakoodin tekemiseen. Ohjelmakoodin mallipohja sekä esimerkkiohjelmat ovat englanninkielisiä, jotta niitä voidaan käyttää englanninkielisillä kursseilla, joihin osallistuu vaihtopilaita.

3.5.2 I2C-väylä ja LCD-näyttö

I2C- tai IIC-väylä on Philipsin 1980-luvulla kehittämä kaksijohtiminen kaksisuuntainen väylä mikropiirien väliseen tiedonsiirtoon. I2C-väylän määrittelydokumentin ensimmäi-

nen versio julkaistiin vuonna 1992. I2C-väylästä käytetään usein merkintöjä IIC, I2C tai I²C. (NXP Semiconductors.) I2C-väylään saatetaan viitata myös lyhenteellä TWI (Two Wire serial Interface), joka viittaa I2C-väylän kaksijohtimiseen väyläkokoon. Esimerkiksi Atmelin mikrokontrollereissa ja Arduinon kirjastoissa I2C-tyyppiseen väylään viitataan TWI-lyhenteellä.

I2C-väylän etuna on, että tiedon siirtoon tarvitaan vain kaksi johdinta. Toinen johdin on sarjamuotoisen tiedon (data) siirtoon ja toinen sarjamuotoisen kellosignaalin siirtoon. Tiedonsiirtojohdin merkitään lyhenteellä SDA (serial data line) ja kellosignaalijohdin SCL (serial clock line). (NXP Semiconductors.)

Arduino -kehitysympäristölle on saatavilla monia LCD-näyttöjä, joita ohjataan eri tavoin. Arduino-kortilla on rajallinen määrä digitaalisia I/O liitäntöjä ja siksi on tärkeää, ettei LCD-näyttö, mikäli sovelluksessa sellaista tarvitaan, tarvitse montaa I/O liitäntää.

I2C-liitännällä varustettu LCD-näyttö on ratkaisu edellä kerrottuun ongelmaan. Tukimateriaalissa kerrotaan kuinka LCD-näyttö liitetään Arduino Mega 2560 mikro-ohjainkorttiin esimerkkejä LCD-näytön käyttöön liittyen.

3.5.3 Ajastimet ja keskeytykset

Arduino Mega 2560 -mikro-ohjainkortin prosessorissa on kuusi ohjelmoitavaa ajastinta, ajastimet Timer0 – Timer5. Ajastimista kaksi on 8-bittisiä ja neljä on 16-bittisiä. Ajastimilla toteutetaan vakioaikavälein tapahtuva keskeytys ja ajastimen keskeytysohjelmassa suoritetaan vakioaikavälein määritellyt tehtävät. Ajastimen aikaa säädetään esijakajalla, prescaler, jolla jaetaan mikroprosessorin kellotaajuus ennalta määriteltujen vakioiden avulla. Mikroprosessorin valmistaja on valinnut vakioiden arvoiksi 1, 8, 64, 256 ja 1024. Toinen ajastimen ajan säätöön vaikuttava asia on ajastimen bittimäärä. Bittimäärä määrää ajastimen laskurin arvoalueen. Esimerkiksi 8-bittinen laskuri voi saada arvot 0 – 255. Mikäli ajastinten laskurien arvoja ei alustusohjelmassa muuteta, on laskureiden arvo 0. (Atmel. 2014)

Ensimmäinen ajastinten käyttöön liittyvä tehtävä oli selvittää, miten PWM signaalin perustaajuutta 50% pulssisuhteella voidaan ajastinten avulla säätää. Ajastinten asetukseen ja asetusten palautusta varten tehtiin omat ohjelmat. Ennen kuin ajastinasetuksia muutetaan, on hyvä selvittää mihin muuhun tarkoitukseen kyseistä ajastinta käytetään Arduino ympäristössä. Esimerkiksi ajastinta 0 käytetään esimerkiksi Arduinon funktioiden `millis()` ja `mikros()` kanssa. Ajastimen 0 arvojen muutos vaikuttaa siis siihen miten esimerkiksi `millis()` funktion palauttama arvo kasvaa, oletusasetuksilla arvo kasvaa 1 ms välein. (Atmel. 2014)

Kun PWM perustaajuuden muuttamiseen tehdyt ohjelmat oli tehty, muutokset testattiin mittaamalla PWM signaalit oskilloskoopilla. Mittaamalla saadut perustaajuudet taulukoitettiin. Eri ajastimia ja esijakajia käyttämällä PWM perustaajuudeksi on mahdollista valita joko noin 30 Hz tai maksimissaan noin 62,5 kHz taajuus. Oletusarvoisesti PWM perustaajuus on noin 500 Hz tai noin 1 kHz, riippuen siitä, mitä Arduino Mega -mikro-ohjainkortin PWM lähtöä käytetään.

Arduinon oletusarvona on, että PWM pulssisuhdetta säädetään 8-bittisesti, jolloin pulssisuhde voi saada 256 eri arvoa. Arduino Mega 2560 -mikro-ohjaimella on mahdollista säätää pulssisuhdetta myös jopa 16-bittisesti (Atmel. 2014). PWM perustaajuuden muuttamisen lisäksi, tehtiin testiohjelma, jolla voidaan PWM-signaalin pulssisuhdetta muuttaa 10-bittisesti, jolloin pulssinleveys voi saada 1024 erilaista arvoa. PWM-signaalin perustaajuuden ollessa 1 kHz, pulssinleveyttä voidaan muuttaa noin 1 μ s askelin.

Toinen vakioaika keskeytyksiin liittyvä tehtävä oli vakioaikakeskeytyksen käyttäminen Arduinon digitaalisen lähdön tilan muuttamiseen. Jos digitaalisen lähdön tila oli LOW, muutettiin lähdön tilaksi HIGH ja päinvastoin. Samalla periaatteella tehtiin LCD-näytöllä tekstin rullaaminen oikealta vasemmalle.

Useissa esimerkeissä on digitaalisen lähdön tilan muuttamiseen tilasta toiseen vakioaika välein käytetty `delay()`-funktioita. `Delay()`-funktion käyttö on sikäli ongelmallinen, että se keskeyttää pääohjelasilmukan suorituksen määritellyn viiveen ajaksi. Viiveen demonstroimiseksi tehtiin esimerkkiohjelma, jossa näytölle tulostetaan aika mikro-ohjainkortin käynnistymisestä lähtien. Ilman viivettä aika päivittyy näytölle sekunnin välein, kun pää-

ohjelmasilmukkaan lisätään viive, huomataan että aika ei päivitykään enää sekunnin välein, vaan ajan päivityksessä on tauko. Esimerkkiohjelmassa ajan muodostamisen apuna käytettiin `millis()` –funktiota ja koska funktio käyttää vakioaikakeskeytystä laskuriarvon kasvattamiseksi, LCD-näytölle päivitettävä aika kasvaa viiveen mukaisin portain.

Monissa sovelluksissa käytetään antureita tai kytkimiä, joiden aktivoinnin jälkeen ohjelman tulee heti suorittaa anturiin tai kytkimeen liittyvää toimintoa. Jos pääohjelmasilmukassa on paljon ohjelmakoodia, voi käydä niin, ettei lyhykestoista kytkimen painallusta tai anturilta saatavaa pulssia havaita.

Arduino Mega 2560 -kortille on mahdollista asettaa 11 erilaista digitaalisen I/O-liitännän muutoksesta aiheutuvaa keskeytystä. Keskeytys voi aiheutua digitaalisen I/O-liitännän LOW-tilasta, mistä tahansa I/O liitännän tilan muutoksesta, I/O-signaalin muutoksesta HIGH → LOW tai I/O-signaalin muutoksesta LOW → HIGH. (Atmel. 2014)

Arduino Mega 2560 -kortissa on 54 digitaalista I/O liitäntää, joista 30 voidaan käyttää I/O liitännän tilamuutoksen aiheuttamaan keskeytykseen. On kuitenkin huomioitava, että useilla digitaalisilla I/O liitännöillä on yksi tai useampi erikoistoiminto, kuten esimerkiksi PWM ulostulo.

3.5.4 DC-moottorin ohjaus

DC-moottorin ohjaus Arduinolla toteutettiin käyttäen kuvan 17 mukaista H-siltaa, jolloin tavoitteena oli suunnitella ohjelma, joka ohjaa DC-moottoria pulssinleveysmodulaatiolla, moottorin pyörimisnopeutta on mahdollista säätää ja moottorin pyörimissuunta voidaan valita. Testiohjelmassa moottorin pyörimissuuntaa vaihdettiin määrävälein. Moottorin pyörimisnopeutta säädettiin muuttamalla pulssinleveysmodulaation pulssisuhdetta. Pulssinleveysmodulaation pulssisuhdetta säädettiin Arduinon analogia-digitaali muunnin tuloon kytketyllä säädettävällä jännitteellä. Analogia-digitaalimuuntimelta saatavan arvon perusteella pulssisuhdetta voitiin säätää lineaarisesti 0-100% välillä. Koska pulssisuhdetta ohjattiin 8-bittisesti ja analogia-digitaali muuntimelta saatiin 10-bittinen lukuarvo, piti analogia-digitaali muuntimelta saatu arvo skaalata 8-bittiseksi. H-siltakytkennästä johtuen käytettiin negatiivista logiikkaa pulssisuhteen asetukseen.

3.5.5 Anturit

Arduinoon on saatavilla useita erilaisia anturimoduuleita, joiden käyttöjännite voidaan ottaa suoraan Arduino mikro-ohjainkortilta. Tukimateriaalissa käydään läpi ultraäänianturin HR-SR04, lämpötila- ja suhteellisen kosteuden anturin DHT11 sekä OMRON E3FA-DP21 valoanturin liittäminen Arduino mikro-ohjainkorttiin.

Ultraääni ja lämpötila anturit valittiin sillä perusteella, että ne olivat saatavilla ja antureilta saadaan tieto eri tavalla. Molempiin antureihin löytyy omat kirjastot ja niiden avulla antureiden käyttö on helppoa. Kuitenkin, jos suunnitellaan aikakriittistä järjestelmää, jossa antureita on useita, on tarkistettava, miten ja missä muodossa tieto saadaan anturilta ja kuinka kauan tiedonsiirtoon tai tiedon saamiseen anturilta mahdollisesti kuluu aikaa. Tukimateriaalissa on kerrottu, miten ja missä muodossa antureilta saadaan tieto mittaustuloksesta sekä esimerkkiohjelmat antureiden käyttöön.

OMRON valoanturi toimii kuten kytkin ja kytketään Arduinoon optoerottimen kautta digitaaliseen tuloon. Arduinon digitaaliseen tuloon voidaan kytkeä anturi, jonka lähtöjännite on maksimissaan 5 V. OMRON valoanturi toimii 24 V jännitteellä ja anturin lähtöjännite on 24 V ja siksi tarvitaan optoerotin. Tukimateriaalissa on esimerkkikytkennät optoerottimen kytkemiseksi Arduinon digitaaliseen tuloon. Ohjelmallisesti valoanturia käsitellään kuten mitä tahansa digitaalisen tuloon liitettyä kytkintä.

3.5.6 Digitaali-analogia muunnin

Digitaali-analogia muunnin, DAC, otettiin mukaan siksi, että opinnäytetyön edetessä tuli tarve selvittää, miten Arduino mikro-ohjainkortilla voisi ohjata analogisella ohjaussignaalilla esimerkiksi servomootoria. Arduino mikro-ohjainkortista ei ole analogialähtöä, jota voitaisiin säätää esimerkiksi 1 mV välein jännitealueella 0 – 5 V. Tätä tarkoitusta varten tarvitaan digitaali-analogia muunnin. Jännitealueella 0 – 5 V noin 1 mV välein tarkoittaa sitä, että digitaali-analogia muuntimen pitää olla vähintään 12-bittinen. Tarvitava bittimäärä saadaan laskettua kaavasta 2, jossa U on lähtöjännitteen maksimiarvo, n on tarvittavien bittien lukumäärä ja U_{Δ} jännitteen minimimuutos.

$$U_{\Delta} = \frac{U}{2^n} \quad (2)$$

Kun kaava 2 ratkaistaan n suhteen ja sijoitetaan tarvittavat lukuarvot, niin saadaan tarvittava bittimäärä kaavan 3 mukaan. Laskusta saatu arvo pyöristetään lähimpään kokonaislukuun, jolloin saadaan tarvittavaksi bittimääräksi 12.

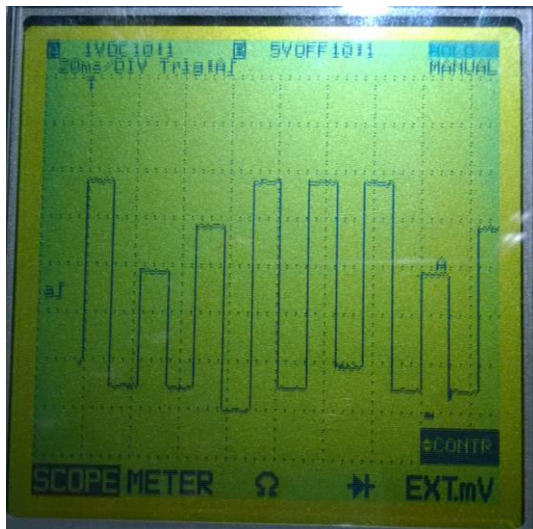
$$n = \log_2 \left(\frac{U}{U_{\Delta}} \right) = \log_2 \left(\frac{5 \text{ V}}{0,001 \text{ V}} \right) = 12,29 \approx 12 \quad (3)$$

Kun kaavaan 2 sijoitetaan lukuarvot, voidaan varmistaa, että jännitteen muutos on noin 1 mV.

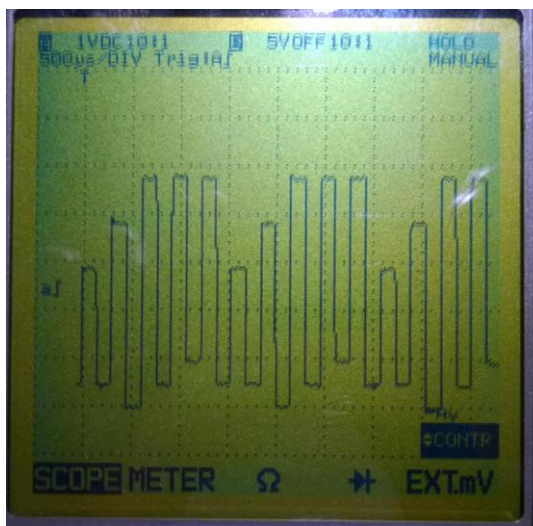
$$U_{\Delta} = \frac{U}{2^n} = \frac{5 \text{ V}}{2^{12}} = 0,001 \text{ V} = 1 \text{ mV}$$

Nopea digitaali-analogia muunnos saadaan tehtyä, jos data DA-muuntimelle saadaan rinnakkaismuodossa. Arduino Mega 2560 käsittelee vain 8-bittistä dataa ja maksimissaan kahdeksan digitaalilähdön tilaa voidaan muuttaa samanaikaisesti yhdellä ohjelmarivillä. Kaavan 2 mukaan laskettaessa 8-bitillä voidaan jännitettä muuttaa noin 20 mV portaissa, joka tässä tapauksessa on liian karkea säätö.

Arduinon löytyy 12-bittisiä DA-muuntimia, joihin digitaalinen lähtöarvo saadaan syötettyä I2C-väylän kautta. I2C-väylään kytketyn DA-muuntimen etuna on, että se kuluttaa vain muutaman digitaalisen lähdön Arduino-kortilta. Haittapuolena on I2C-väylän hitaus, joka johtuu siitä, että tietoa siirretään sarjamuodossa ja I2C-väylällä voi olla useita laitteita. Kuvassa 20 on tilanne, jossa LCD näyttöä päivitetään samalla kun DAC-kortille syötetään dataa. Kuvassa 21 on tilanne, jossa LCD näytön päivitys on poistettu käytöstä. Kun LCD näyttöä päivitetään, DAC-kortin lähtö muuttuu noin 12 ms välein. Ilman LCD näytön päivitystä DAC-kortin lähtö muuttuu noin 100 μ s välein, eli noin 100 kertaa nopeammin.



KUVA 20. DAC-kortin nopeuden testaus, LCD-näytön päivitys on käytössä



KUVA 21. DAC-kortin nopeuden testaus, LCD-näytön päivitys ei ole käytössä

Servomootorin ohjaukseen vaihtoehtoinen keino DA-muuntimen sijaan on käyttää PWM-signaalia, säätää sen perustaajuus sopivaksi ja suodattaa alipäästösuotimella PWM-signaali, jolloin saadaan servomoottoriohjaimelle tasajänniteohjaus.

4 POHDINTA

Opinnäytetyön tavoitteena oli tuottaa tukimateriaali Tampereen ammattikorkeakoulun koneautomaation opintopolun elektroniikan ja mikro-ohjaintekniikan opetukseen. Työn tuloksena syntyi yli 100-sivuinen tukimateriaali, joka sisältää teoriaa ja käytännön esimerkkejä analogia- ja digitaalelektroniikasta sekä mikro-ohjaimen ohjelmoinnista. Esimerkkien yhteydessä on mainittu, kuinka paljon aikaa kului kytkennän rakentamiseen ja mittaukseen.

Analogia- ja digitaalelektroniikan osalta tukimateriaali kokoaa esimerkit ja lähteet opintojakson suunnittelua varten. Mikro-ohjaintekniikan osalta materiaali täydentää aiemmissa opinnäytetöissä käsiteltyjä asioita ja tarjoaa syventävää tietoa keskeytyksistä, erilaisten antureiden kytkemisestä sekä I²C-väylän hyödyntämisestä. I²C-väylän käyttöesimerkit liittyvät LCD-näytön ja DA-muuntimen ohjaukseen. Erityisesti kun sekä I²C-väylälle kytketyt LCD-näyttö ja DA-muunnin ovat samanaikaisesti käytössä, väylän ja mikro-ohjaimen ominaisuudet tulevat hyvin esille.

Kun opintojaksot ja niiden sisällöt muuttuvat, on tukimateriaalin sisältöäkin tarvittaessa päivitettävä. Tukimateriaaliin voisi jatkossa lisätä ohjeita elektroniikan simulointiohjelmien, kuten LTSpice, Micro-Cap tai NI Multisim ja jonkin piirikaavion suunnitteluohjelman, esimerkiksi EAGLE, käyttöön sekä analogia- että digitaalelektroniikan näkökulmasta. Mikro-ohjaintekniikan osalta tukimateriaaliin voisi lisätä esimerkiksi MATLAB/Simulinkin hyödyntämisen mikro-ohjaimen ohjelmoinnissa.

Elektroniikan opintojaksossa olevan digitaalelektroniikan osalta voisi jatkokehityksenä miettiä, voidaanko ainakin osaa digitaalelektroniikan asioista käsitellä esimerkiksi ohjelmoitavia logiikkoja käsittelevien opintojaksojen yhteydessä. Kombinaatiologiikan harjoitustöitä suunniteltaessa voisi joissain tapauksissa olla järkevää käyttää ohjelmoitavia logiikkapiirejä perusporttipiirien sijaan.

LÄHTEET

Aaltonen, J., Kousa, S., Stor-Pellinen, J. 2004. Elektroniikan perusteet. 4. korj. p. Helsinki: Limes.

Airila, M. 1999. Mekatroniikka. 5. korj. p. Espoo: Otatieto Oy.

Arduino. 2017. Arduino homepage. Luettu 7.9.2017
<http://arduino.cc/>

Atmel. 2014. ATmega640/1280/1281/2560/2561 Complete datasheet.
http://www.atmel.com/Images/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf

Bishop, R. H. 2006. Mechatronics: An introduction. Boca Raton: Taylor & Francis.

Bosch. 2007. Automotive electrics and automotive electronics. 5th ed. Chichester: Wiley.

Braga, N. C. 2002. Robotics, mechatronics and artificial intelligence: Experimental circuit blocks for designers. Boston: Newnes.

Bolton, W. 2003. Mechatronics: Electronic control systems in mechanical and electrical engineering. 3rd ed. Harlow: Pearson.

European Commission. 2017. Restriction of Hazardous Substances. WWW-sivu. Luettu 3.8.2017. http://ec.europa.eu/environment/waste/rohs_eee/index_en.htm

Floyd, T. L. 2009. Digital fundamentals. 10th ed. Upper Saddle River, (NJ): Pearson Prentice Hall.

Gaddis, T. 2015. Starting out with C++: From control structures through objects. 8th ed., global ed. Harlow: Pearson Education.

Haiko, T., Salonen, K. & Harju, T. 2012. Digitaalielektroniikka: Peruspiireistä sulautetuihin järjestelmiin. Helsinki: Sanoma Pro.

Heikkinen, J. 2014. Soveltavan elektroniikan opetussisällön kehittäminen. Ammatillinen opettajakorkeakoulu. Tampereen ammattikorkeakoulu. Kehittämishanke.

Incose. 2016. Guide to the Systems Engineering Body of Knowledge (SEBoK) v1.7. Luettu 28.11.2016.
http://sebokwiki.org/wiki/Download_SEBoK_PDF

Koskinen, J. 2006. Mikrotietokonetekniikka: Sulautetut järjestelmät. Helsinki: Otava.

McManis C. 2003. The complete BJT Circuit. Building the H-bridge. WWW-sivu. Luettu 20.6.2016.
<http://www.mcmanis.com/chuck/Robotics/tutorial/h-bridge/bjt-circuit.html>

Millman, J., Grabel, A. 1987. Microelectronics. 2.ed. New York: McGraw-Hill.

National Instruments. 2012. Mechatronics – Five Design Challenges and Solutions for Machine Builders. WWW-sivu. Luettu 4.9.2017.

<http://www.ni.com/newsletter/50145/en/>

NXP Semiconductors. 2014. I2C-bus specification and user manual. Rev. 6.

http://www.nxp.com/documents/user_manual/UM10204.pdf

On Semiconductor. 2014. TIP120, TIP121, TIP122 (NPN); TIP125, TIP126, TIP127 (PNP) Plastic Medium-Power Complementary Silicon Transistor. Datasheet. Luettu 10.1.2017.

<https://www.onsemi.com/pub/Collateral/TIP120-D.PDF>

Onwubolu, G. C. 2005. Mechatronics: Principles and applications. Oxford: Elsevier/Butterworth-Heinemann.

Seipijärvi, A. 2015. Arduinon käyttö sulautettujen järjestelmien opetuksessa. Oppimateriaalin kehittäminen. Kone- ja tuotantotekniikka. Tampereen ammattikorkeakoulu. Opinnäytetyö.

Silvonen, K. 2009. Elektroniikka ja puolijohdekomponentit: Analogia- ja digitaalitekniikka, vahvistimet, teholähteet, suodattimet, teho- ja tietoliikenne-elektroniikka, piirimulointi. Helsinki: Otatieto.

Tampereen ammattikorkeakoulu. Opetussuunnitelma. Kone- ja tuotantotekniikan ohjelma, Koneautomaatio. Vuonna 2013 tai aiemmin aloittaneet. Luettu 3.7.2017.

<http://opinto-opas-ops.tamk.fi/index.php/fi/167/fi/57/13I140/9/year/2013>

Texas Instruments. μ A741 General-Purpose Operational Amplifiers. Datalehti. Luettu 1.7.2016.

<http://www.ti.com/lit/ds/symlink/ua741.pdf>

LIITEET

Liite 1. Ohjelmakoodin mallipohja

```
//-----
// TITLE:           Template for Arduino projects.
//
// DESCRIPTION:      Write here short description of your code.
//
// AUTHOR:           YOUR NAME
//
// COURSE ID:        xxxxxxx-xxxx
// COURSE NAME:      Course name
// GROUP NAME:       GROUP 1
// GROUP MEMBERS:    MEMBER 1, MEMBER 2, ETC.
//
// DATE:            YYYY.MM.DD
//-----

// Needed libraries here
//#include <library.h>

// Define needed constants here.
// If a certain constant is not needed in the final code, please use #define,
// otherwise const int
//#define cursor_pos_temp 0
//#define cursor_pos_temp 10
//#define ROW1 0
//#define ROW3 2
#define OUTPUTPIN 13 //This is only an example.

//const datatype CONST_NAME=VALUE;

// If you use any global variables, define variables here. Remember to
// initialize variables to desired value.
//datatype variable_name=initialValue;

// If you need to use your own functions. Put function prototypes here.
// Example:
//datatype your_function_name(datatype parameter1, ... datatype parameterX);
//
// Example: void UpdateValues2LCD(int temperature, int humidity);

void setup() {
    // put your setup code here, to run once:
}

//END SETUP

void loop() {
    // put your main code here, to run repeatedly:
}

//END LOOP

// Write your own functions here.
// Example:
//
// void UpdateValues2LCD(int temperature, int humidity){
//     // Write your own code here.
//     lcd.setCursor ( cursor_pos_temp, ROW1 );
//     lcd.print(temperature);
//     lcd.setCursor ( cursor_pos_humid, ROW3 );
//     lcd.print(humidity);
// }

// }//END OWN FUNCTION
```