



TAMPEREEN  
AMMATTIKORKEAKOULU

# FORMULA STUDENT KILPA-AUTON DRS-JÄRJESTELMÄ

Tuomas Pärnänen

Opinnäytetyö  
Elokuu 2018  
Sähkötekniikan ko.  
Automaatiotekniikka



## TIIVISTELMÄ

Tampereen ammattikorkeakoulu  
Sähkötekniikan ko.  
Automaatiotekniikka

PÄRNÄNEN, TUOMAS:  
Formula Student kilpa-auton DRS-järjestelmä

Opinnäytetyö 72 sivua, joista liitteitä 4 sivua  
Elokuu 2018

---

Tässä opinnäytetyössä tutkittiin Formula Student kilpa-auton ilmanvastusta vähentävän järjestelmän toimintaa, hyötyjä ja toteutusta. Tarkoituksena on parantaa kilpa-auton suorituskykyä radalla ja kilpailun staattisissa osioissa.

Opinnäytetyöhön kuuluu koko ilmanvastusta pienentävä järjestelmä, mutta pääpaino on sähköisessä toteutuksessa. Lähtötiedot järjestelmään tulivat aiempien vuosien autoista ja Solidworks Flow -virtaus-simulaatiolla saaduista arvoista.

Sähköiseen toteutukseen sisältyy anturisignaalien lukemiseen suunniteltu moduuli, siipi-profiilien asentoa kääntävien moottoreiden valinta, moottoreiden ohjauselektronikan suunnittelu, ARM-prosessoreiden C-kielellä kirjoitettu ohjelma ja kaikkia laitteita yhdistävä CAN-väylä.

Järjestelmän edut ja haitat huomioon ottaen järjestelmä päätettiin toteuttaa vain takasiiven profiileihin. Jatkossa enemmän testattua järjestelmää voidaan hyödyntää myös auton etusiivessä. Talviolosuhteiden vuoksi viimeisiä käytännön testauksia järjestelmälle ei pystynyt suorittamaan.

## ABSTRACT

Tampere University of Applied Sciences  
Degree Programme in Electrical Engineering  
Automation Engineering

PÄRNÄNEN, TUOMAS:  
Formula Student Racing Car's Drag Reduction System

Bachelor's thesis 72 pages, appendices 4 pages  
August 2018

---

The purpose of this thesis was to investigate the performance, benefits and implementation of the drag reduction system in a Formula Student racing car in order to improve the car's performance on the track and in the competition's static events.

The whole drag reduction system is discussed in this thesis, but the emphasis is on electronic implementation. Initial values and data were collected from the cars of previous years and values were obtained from Solidworks Flow Simulation.

The electronic implementation includes modules designed for sensor signals, servo motors rotating the wing profile positions, electronic design of the servo motor controller, a program written in the C language for ARM processors, and the communication between devices via CAN bus.

In view of the advantages and disadvantages of the system, it was decided to implement only the profiles of the rear wing. In the future, once the system has been tested further, it can also be utilized in the racing car's front wing. Due to the winter, the last practical tests was not been accomplished.

---

Keywords: formula student, drag reduction system, wings, electronic design

## SISÄLLYS

1	JOHDANTO.....	7
2	FORMULA STUDENT .....	8
2.1	Historia.....	8
2.2	Tampere UAS Motorsport .....	10
3	AKTIIVINEN AERODYNAMIIKKA .....	11
3.1	Laskennallinen hyöty .....	11
4	MOOTTORIT .....	19
4.1	CFD-simulaatio.....	19
4.1.1	Simulointiparametrit .....	19
4.1.2	Siipiprofiilin nivelpiste .....	20
4.1.3	Koko siiven simulaatiot .....	23
4.1.4	Etusiiven simulaatiot.....	24
4.2	Moottorin valinta .....	26
4.3	Moottorin muokkaus.....	27
5	CAN-väylä.....	28
5.1	CAN-väylän jännitetasot.....	29
5.2	Väylän ajoitus .....	30
5.3	CAN-viestin kehys.....	31
6	PIIRILEVYJEN VALMISTUS.....	33
6.1	Piirilevyn suunnittelusäännöt.....	33
6.1.1	Pattern Class.....	34
6.1.2	Drill Class .....	35
6.2	Piirilevyjen tilaus .....	35
6.3	Komponenttien juottaminen.....	36
7	PIIRILEVYJEN SUUNNITTELU .....	37
7.1	CAN-moduuli .....	37
7.1.1	Liittimet.....	37
7.1.2	Piirilevyn suojaus .....	40
7.1.3	Kanavan kytkentä.....	41
7.2	Servo-ohjain.....	45
7.2.1	Jännitteensyöttö.....	45
7.2.2	Ohjainpiirilevy .....	48
8	KOODI .....	50
8.1	STM32CubeMX .....	50
8.1.1	Graafinen prosessorin kytkentä.....	50
8.1.2	Kellotaajuuspuu.....	52

8.1.3	Parametrit .....	54
8.1.4	Generoitu koodi.....	56
8.2	CAN-moduuli C-Koodi .....	57
8.2.1	CAN-moduulin muuttujien alustus .....	58
8.2.2	Analogikanavien luku .....	59
8.2.3	Datan käsittely ja lähetys väylään .....	60
8.3	Servo-ohjain C-Koodi.....	61
8.3.1	CAN-viestin vastaanotto .....	62
8.3.2	Laskenta .....	62
8.3.3	Servon ohjaus .....	63
8.3.4	Prossessorin PWM-lähtö .....	64
9	KOKOONPANO .....	65
9.1	Ohjaussignaali.....	65
9.2	Servomootoreiden kiinnitys .....	67
9.3	Ohjainlaite.....	67
9.4	Profiilien kiinnittäminen .....	68
9.5	Muita kokoonpanoon liittyviä asioita .....	69
10	POHDINTA.....	70
	LÄHTEET .....	71
	LIITTEET .....	73
	Liite 1. CAN-moduuli piirilevy kytkentä. ....	73
	Liite 2. CAN-moduuli piirilevy kytkentä 2. ....	74
	Liite 3. Servon jännitteensyötön piirilevy. ....	75
	Liite 4. Servo-ohjaimen piirilevy. ....	76

**LYHENTEET JA TERMIT**

TAMK	Tampereen ammattikorkeakoulu
DRS	Drag reduction system, ilmanvastusta pienentävä järjestelmä
Formula SAE	Formula Society of Automotive Engineers
RC	Radio controlled, radio-ohjattava
CFD-simulaatio	Computational Fluid Dynamics, virtaus-simulaatio
Drag	Ilmanvastus
Downforce	Voima, jolla ilmavirtaus painaa siipeä alaspäin.
CAN-väylä	Kaksijohtiminen tiedonsiirtoväylä.
PCB	Printed Circuit Board, piirilevy
Soldermask	Piirilevyn pinnalla oleva lakkakerros, joka hylkii juotosta
Silkscreen	Piirilevyn pinnalla olevat tekstit ja muut painatukset
Potentiometri	Komponentti, jonka vastusta voidaan muuttaa pyörittämällä käsin tai työkalulla

## 1 JOHDANTO

Formula Student -sarjassa on jo vuosia sallittu aerodynamiikkaa parantavat siivet autoissa. Siipien myötä auton ajettavuus parantuu huomattavasti mutkissa, mutta kasvaneen ilmanvastuksen johdosta autolla häviää suoralla.

Opinnäytetyössä käsitellään ilmanvastusta pienentävän järjestelmän (eng. Drag Redution System) suunnittelua ja toimintaa. Lähtöarvoja on kerätty aiempien vuosien kilpailuista autojen tiedonkeruujärjestelmän avulla. Virtaus-simulaatiolla saatiin selvitettyä moottorin mitoittamiseen ja sijoitukseen tarvittavia lähtöarvoja ja myös optimoitua siipiprofiilien asentoa.

Siiven toiminta toteutettiin automaattisesti, jolloin kuljettajan ei tarvitse huolehtia järjestelmän toiminnasta. Siiven asentoon vaikuttavat kaasupolkimen, jarrupolkimen ja ratin asento. Näitä signaaleita varten kehitettiin CAN-moduuli mittauspiiri, joka lukee anturi-signaalit ja lähettää tiedon CAN-väylään.

Avautuvautuvan siiven toimilaitteelle suunniteltiin ohjainpiiri, joka ottaa vastaan väylässä lähetetyt tiedot ja ohjaa moottorien asentoa tarpeen mukaan.

## 2 FORMULA STUDENT

Formula Student on kansainvälinen Formula SAEn organisoima suunnittelukilpailu korkeakoulu- ja yliopisto-opiskelijoille. Opiskelijat suunnittelevat ja rakentavat vuosittain yksipaikkaisen formulatyylisen prototyypikilpa-auton, jolla he kilpailevat toisiaan vastaan erilaisissa staattisissa ja dynaamisissa kilpailun osa-alueissa.

Formula Student -kilpailun säännöt täyttävän auton suunnittelu sisältää tutkimustyötä, suunnittelua, valmistusta, testausta, kehitystyötä, markkinointia, organisointia, esittelyä ja taloudenhallintaa. (SAE International 2017)

### 2.1 Historia

Formula Student perustettiin 1980 Teksasin yliopistossa yhdessä kolmen suuren amerikkalaisen autovalmistajan kanssa (GM, Ford ja Chrysler). Alkuperäiset säännöt sallivat minkä tahansa 4-tahtisen moottorin käytön yhden tuuman (25.4 mm) imuilman kuristimella. Polttoaineena sallittiin bensiinit, dieselit, metanoli ja etanoli. (Case 1996)

Ensimmäinen kilpailu järjestettiin Teksasissa 1981, johon osallistui neljä yliopistoa. Kilpailun staattisina osa-alueina olivat esiintyminen (Appearance) ja suunnittelu (Engineering and Design). Dynaamisina osa-alueina kiihdytys (Acceleration), ohjattavuus (Maneuverability), 22.5 km kestävyysajo (Endurance) ja polttoainetaloudellisuus (Fuel Economy). Ensimmäisten kilpailuvuosien aikana yliopistot käyttivät laajasti erityyppisiä moottoreita, kuten 65 cc<sup>3</sup> Kawasakeja ja 11 hp Briggs & Strattonia hyvän polttoainetaloudellisuuden takia ja 600 cc<sup>3</sup> Hondan ja 750 cc<sup>3</sup> Triumphin moottoreita hyvän suorituskyvyn takia. Etenkin kiihdytyksessä autojen suorituskyky parani paljon ensimmäisten vuosien aikana. (Case 1996)

1984 staattisena osa-alueena mukaan tuli kustannuslaskelma (Cost). Kustannuslaskelmassa laskettiin auton materiaali ja valmistuskustannukset olettaen, että samanlaisia ajoneuvoja valmistetaan 1000 kappaletta. Tavoitekustannukset olivat 2000 \$ ja suurimmat sallitut kustannukset olivat 4500 \$. (Case 1996)

1985 Formula Student laajentui Yhdysvaltojen ulkopuolelle, kun Ecole Polytechnic Montreal osallistui Teksasin kilpailuun. (Case 1996)



1986 sarja laajeni Texasista Lawrenceen, kun Lawrence Technological University järjesti toisen kilpailun. (Case 1996)

1987 mukaan tuli viimeinen nykyisistä kilpailun osa-alueista, kitkaympyrä (Skid Pad). Jo ensimmäisessä kilpailussa kaksi yliopistoa saavuttivat yli 1 g sivuttaiskiihtyvyyden. (Case 1996)

1980-luvun lopulla remmi- ja pakokaasuahdetut moottorit yleistyivät kasvattaen suorituskykyä, jolloin sarjaan tuli 610 cc<sup>3</sup> kuutiotilavuusraja ja kuristimen kokoa pienennettiin 23 mm. (Case 1996) (SAE International n.d.)

1998 järjestettiin ensimmäinen näytösluontoinen Formula Student kilpailu Euroopassa, Englannissa. Kilpailuun osallistui kolme yhdysvaltaistiimiä ja neljä englantilaistiimiä. Onnistuneen kilpailun jälkeen The Institution of Mechanical Engineering otti hallintaan kilpailun organisoinnin yhteistyössä Formula SAEn kanssa. Vuonna 2007 kilpailu siirtyi tunnetulle Silverstone Circuitille. Formula Student UK on järjestetty vuosittain aina tähän päivään asti. (Institution of Mechanical Engineers 2016)

2000-luvulla kuristimen koko on pienentynyt 20 mm bensiiniä käytettäessä ja 19 mm etanolia käytettäessä. 2016 Formula Student Germany (FSG) teki kokonaan oman sääntökirjan aiemman muutaman sivun sääntölisäysten lisäksi. FSG pyrki tällä yksinkertaistamaan ja selkeyttämään Formula Student -sääntöjä. Euroopan kilpailut käyttävätkin nykyään yleisemmin FSG-sääntöjä alkuperäisten FSAE-sääntöjen sijaan.

Vuosien varrella tunnettuja henkilöitä moottoriurheilusta, kuten Ross Brawn, Pat Symonds, Daemon Hill, Bob Bell, James Allison, Leena Gade, Paddy Lowe ja David Brabham on ollut mukana Formula Student -tapahtumissa tai tukemassa tapahtumia. (Institution of Mechanical Engineers 2016)

Vuonna 2017 Euroopassa järjestettiin 9 virallista kilpailua:

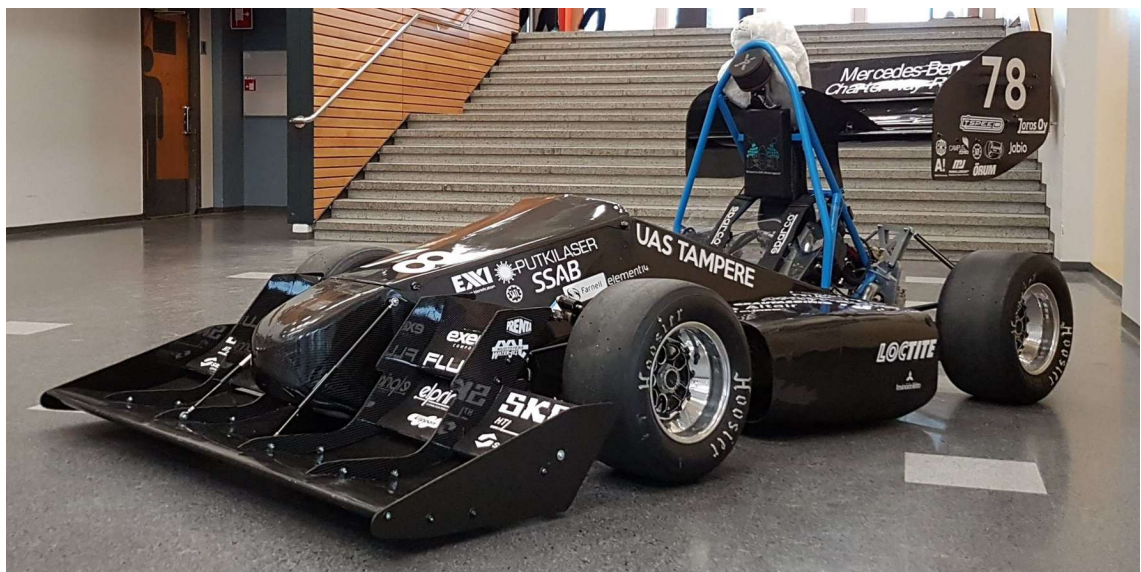
- Formula Student Germany
- Formula Student UK
- Formula Student Austria
- Formula Student Italy
- Formula Student Spain
- Formula Student Hungary
- Formula Student East

- Formula Student Czech
- Formula Student Netherlands

## 2.2 Tampere UAS Motorsport

Tampere UAS Motorsport aloitti toiminnan vuonna 2007. Formula Student -tiimin ensimmäinen auto valmistui vuonna 2008. Tämän jälkeen Tampereen ammattikorkeakoulussa on vuosittain suunniteltu ja rakennettu uusi auto. Vuodesta 2013 lähtien auton voimanlähteenä on ollut 550 cc<sup>3</sup> Aprilia RXV -moottori.

Ensimmäiset siivet olivat FS014-autossa, jonka jälkeen siivistä on tullut kaksi kehitysversiota. FS015 ja FS016 ajettiin samalla etu ja takasiivellä. FS017 tuli uusi paranneltu siipipaketti. Opinnäytetyössä käsiteltävä DRS-järjestelmä perustuu FS017-auton siipipakettiin (kuva 1).



Kuva 1. FS017 TAMKIn aulassa.

### 3 AKTIIVINEN AERODYNAMIIKKA

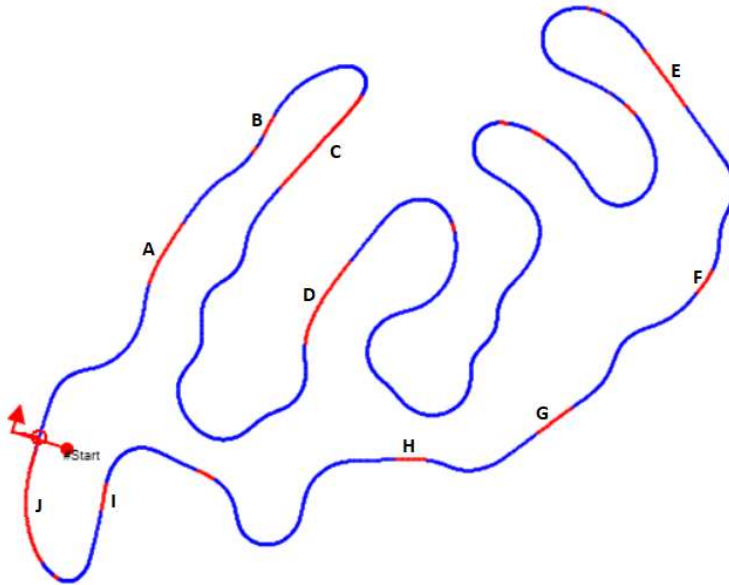
DRS (Drag Reduction System) eli ilmanvastusta pienentävä järjestelmä on tuttu 2010-luvun Formula 1 -autoista (kuva 2). Samoihin aikoihin aerodynamiikkaa parantavat siivet sallittiin ja myös avautuvat siivet ilmestyivät Formula Student -autoihin. (Formula One 2018)



Kuva 2. Red Bull Racing DRS-järjestelmä (Formula One 2018).

#### 3.1 Laskennallinen hyöty

Euroopan suurin kilpailu FSG käydään Hockenheimringillä. FSG Endurance/Autocross -rata on joka vuosi samanlainen. Formula Student -rata on rakennettu Formula 1 -radan pääsuoran jälkeiseen ”Nordkurve”-mutkaan. Kuvassa 3 Tampere UAS Motorsport FS016 -autolla ajettu rata kierros Race Technology DL1 tiedonkeruujärjestelmän GPS-signaalista.



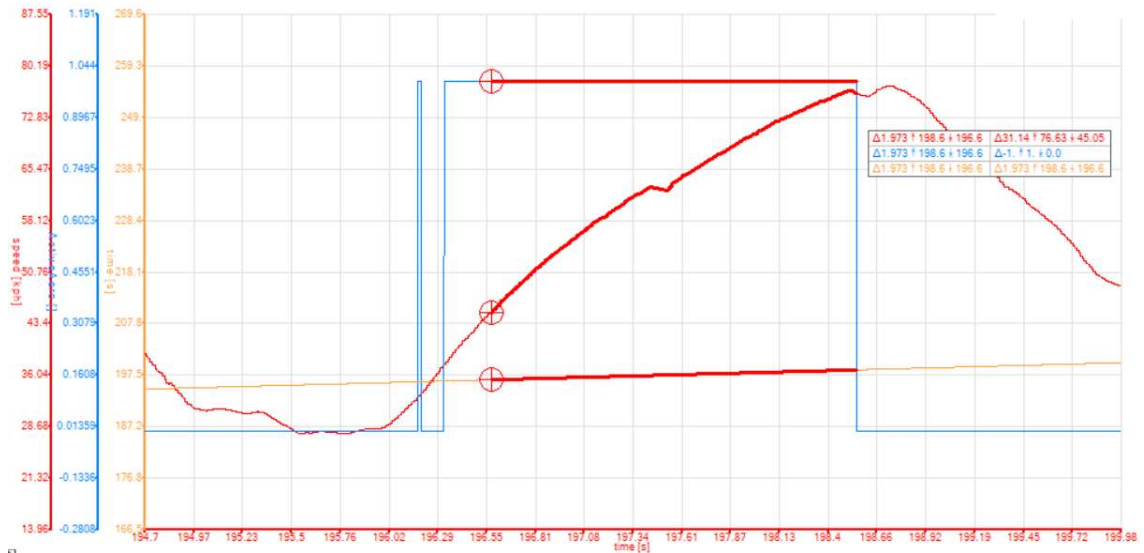
Kuva 3. Formula Student Germany Endurance -rata.

Kuvan 3 ratakarttaan on merkitty punaisella kohdat, joissa DRS-järjestelmästä saavutetaan etua. Nämä ”DRS-alueet” on eroteltu toisistaan kirjaimin A, B ja C jne.

DRS-alueiden ehtoina on käytetty seuraavia ohjauksen ehtoja:

1. Kaasupoljin on ollut pohjassa: kaasuläpän asentoanturin (Throttle Position Sensor, TPS) signaali yli 95 %.
2. Jarrupoljinta ei ole painettu: jarrupaineanturin näyttämä alle 5 bar.
3. Rattia käännetty alle 20 astetta.

Race Technologyn Analysis ohjelman rajoitteiden vuoksi 250 ms viivettä ei pystynyt toteuttamaan ratakartan kuviin. Tämä on otettu huomioon DRS-alueiden aikoja ja nopeuksia tutkittaessa (kuva 4). Myös lyhyet alle 250 ms alueet on jätetty kokonaan huomiomatta.



Kuva 4. DRS-alueen C nopeus ja kulunut aika.

2016 FSG Endurancen aikana ajetuissa kierrosajoissa siipi on jatkuvasti kiinni. Jokaisen DRS-alueen dataa tutkittiin kuvan 4 mukaisesti ja otettiin ylös alkunopeus, loppunopeus ja kulunut aika. Näiden perusteella laskettiin nopeuden kasvu kaavan 1 mukaan.

$$\Delta v = v_2 - v_1 \quad (1)$$

Nopeuden muutoksesta ja kuluneesta ajasta laskettiin kaavan 2 mukaan keskimääräinen kiihtyvyys.

$$a = \frac{\Delta v / 3.6}{t} \quad (2)$$

Arvot kirjattiin taulukkoon 1.

Taulukko 1. DRS-alueiden perustiedot

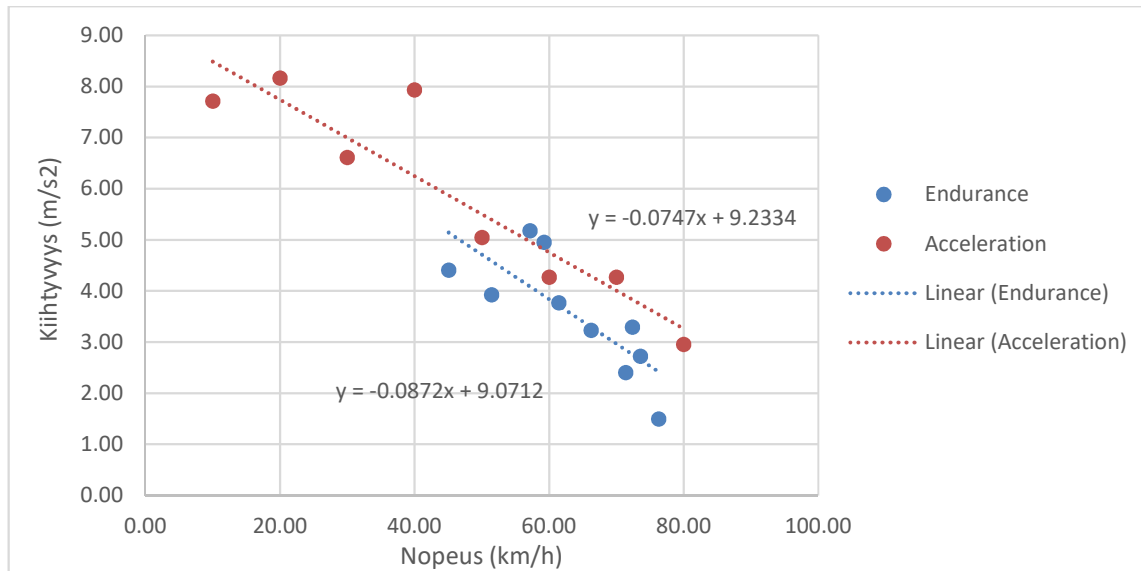
Alue	t (s)	v <sub>1</sub> (km/h)	v <sub>2</sub> (km/h)	Δv (km/h)	a (m/s <sup>2</sup> )
A	0.77	72.40	81.53	9.13	3.29
B	0.11	73.56	74.63	1.08	2.73
C	1.97	45.05	76.63	31.27	4.41
D	1.23	61.42	78.12	16.70	3.77
E	0.73	76.26	80.20	3.94	1.50
F	0.23	59.23	63.32	4.10	4.95
G	0.38	71.38	74.85	3.29	2.40
H	0.24	66.26	69.05	2.79	3.23
I	0.25	57.13	61.79	4.66	5.18
J	1.64	51.45	74.68	23.20	3.93

2016 FSG Acceleration osa-alueessa siipi oli auki. Muuten auto oli täysin vertailukelpoinen Enduranceen verrattuna. Accelerationin datoista (taulukossa 2) nähdään auton kiihtyvyys siiven ollessa auki.

Taulukko 2. Acceleration perustiedot

Nopeus (km/h)	t (s)	v <sub>1</sub> (km/h)	v <sub>2</sub> (km/h)	Δv (km/h)	a (m/s <sup>2</sup> )
10-20	0.36	10.00	20.00	10.00	7.72
20-30	0.34	20.00	30.00	10.00	8.17
30-40	0.42	30.00	40.00	10.00	6.61
40-50	0.35	40.00	50.00	10.00	7.94
50-60	0.55	50.00	60.00	10.00	5.05
60-70	0.65	60.00	70.00	10.00	4.27
70-80	0.65	70.00	80.00	10.00	4.27
80-90	0.94	80.00	90.00	10.00	2.96

Kiihtyvyyksiin syntyy virheitä kiihdytyksen aikana tapahtuvien vaihteen vaihtojen seurauksena. Virheet pyrittiin poistamaan piirtämällä kiihtyvyyden ja nopeuden suhteen pistekaavio ja lisäämällä kuvioon trendiviivat (kuvio 1).



Kuvio 1. Endurance- ja Acceleration-tapahtumien kiihtyvyydet.

Trendiviivojen antamalla kaavoilla 3 ja 4 laskettiin keskimääräiset kiihtyvyydet DRS-alueille siiven ollessa kiinni ja auki.

$$a_{DRSon} = -0.0747 \cdot v_1(\text{km/h}) + 9.2334 \quad (3)$$

$$a_{DRSon} = -0.0747 \cdot 72.40 + 9.2334$$

$$a_{DRSon} = 3.83$$

$$a_{DRSoff} = -0.0872 \cdot v_1(\text{km/h}) + 9.0712 \quad (4)$$

$$a_{DRSoff} = -0.0872 \cdot 72.40 + 9.0712$$

$$a_{DRSoff} = 2.76$$

Alkunopeus ( $v_1$ ) muutettiin m/s yksikköön kaavan 5 mukaisesti.

$$v_1(\text{m/s}) = \frac{v_1(\text{km/h})}{3.6} = \frac{72.40}{3.6} = 20.11 \quad (5)$$

Alkunopeudesta  $v_1(\text{m/s})$ , kuluneesta ajasta  $t$  ja aiemmin lasketusta kiihtyvyydestä  $a_{DRSoff}$  laskettiin alueen keskimääräinen loppunopeus

$$v_2(\text{m/s}) = v_1(\text{m/s}) + t \cdot a_{DRSoff} \quad (6)$$

$$v_2(\text{m/s}) = 20.11 + 0.77 \cdot 2.76$$

$$v_2(\text{m/s}) = 22.24$$

Alku ja loppunopeuden perusteella laskettiin keskinopeus ( $v$ ) alueelle kaavan 7 mukaisesti.

$$v_{m/s} = \frac{v_1(m/s) + v_2(m/s)}{2} \quad (7)$$

$$v_{m/s} = \frac{20.11 + 22.24}{2}$$

$$v_{m/s} = 21.17$$

Keskinopeuden ja kuluneen ajan perusteella laskettiin DRS-alueiden pituudet ( $l$ ) kaavan 8 mukaisesti.

$$l = v(m/s) \cdot t \quad (8)$$

$$l = 21.17 \cdot 0.77$$

$$l = 16.30$$

Taulukko 3. Laskelmia DRS-alueiden datoista.

Alue	t (sec)	$v_1$ (km/h)	$v_1$ (m/s)	$v_2$ (m/s)	$v$ (m/s)	l (m)	$a_{DRSon}$	$a_{DRSoft}$
A	0.77	72.40	20.11	22.24	21.17	16.30	3.83	2.76
B	0.11	73.56	20.43	20.73	20.58	2.26	3.74	2.66
C	1.97	45.05	12.51	22.65	17.58	34.63	5.87	5.14
D	1.23	61.42	17.06	21.63	19.35	23.80	4.65	3.72
E	0.73	76.26	21.18	22.95	22.07	16.11	3.54	2.42
F	0.23	59.23	16.45	17.35	16.90	3.89	4.81	3.91
G	0.38	71.38	19.83	20.91	20.37	7.74	3.90	2.85
H	0.24	66.26	18.41	19.20	18.80	4.51	4.28	3.29
I	0.25	57.13	15.87	16.89	16.38	4.10	4.97	4.09
J	1.64	51.45	14.29	21.81	18.05	29.60	5.39	4.58

Taulukon 3 arvoista voidaan muodostaa yhtälöryhmä, jolla ratkaistaan DRS-alueen aika siiven ollessa auki. Yhtälöryhmässä on huomioitu suuremman nopeuden seurauksena aikaisempi jarrutus mutkaan. Jarrutusmatkan kasvu otettiin huomioon laskemalla alku- ja loppunopeudeksi sama kuin siiven ollessa kiinni. Endurancen dataja tutkittaessa huomattiin jarrutuksissa hidastuvuuden olevan keskimäärin  $10 \text{ m/s}^2$ .



Yhtälöryhmän kaavalla 9 lasketaan kiihdytys huippunopeuteen siiven ollessa auki. Muuttuja  $t_1$  kuvaa kiihdytykseen kulunutta aikaa.

Kaavalla 10 lasketaan hidastuvuus jarrutustilanteessa maksiminopeudesta takaisin DRS-alueen haluttuun loppunopeuteen. Muuttuja  $t_2$  kuvaa jarrutukseen kulunutta aikaa.

Kaavalla 11 lasketaan DRS-alueelle kulunut kokonaisaika.

Yhtälössä 12 kiihdytetty matka ja jarrutettu matka lasketaan yhteen, josta saadaan jo aiemmin tunnettu DRS-alueen pituus. Pituudet lasketaan keskinopeudesta ja kuluneesta ajasta.

$$a_{DRSon} \cdot t_1 + v1 = v_{MAX} \quad (9)$$

$$10 \text{ m/s}^2 \cdot t_2 + v2 = v_{MAX} \quad (10)$$

$$t_{DRSon} = t_1 + t_2 \quad (11)$$

$$\frac{v1 + v_{MAX}}{2} \cdot t_1 + \frac{v2 + v_{MAX}}{2} \cdot t_2 = l \quad (12)$$

Sijoittamalla arvot muuttujiin ja ratkaisemalla yhtälöryhmän, saadaan ratkaistua DRS-alueeseen kulunut aika  $t$  ja alueen huippunopeus  $v_{MAX}$ . DRS-alueella säästetty aika saatiin laskemalla alkuperäisen ja lasketun arvon erotus. Lasketut arvot taulukossa 4.

Taulukko 4. DRS-alueiden lasketut ajat ja maksiminopeudet.

Alue	t (sec)	t <sub>DRSon</sub>	v <sub>MAX</sub>	Δt (sec)
A	0.77	0.757	22.794	0.013
B	0.11	0.110	20.812	0.000
C	1.97	1.913	23.336	0.057
D	1.23	1.201	22.322	0.029
E	0.73	0.718	23.524	0.012
F	0.23	0.229	17.486	0.001
G	0.38	0.377	21.188	0.003
H	0.24	0.239	19.359	0.001
I	0.25	0.249	17.034	0.001
J	1.64	1.594	22.509	0.046

Säästetystä ajasta  $\Delta t$  huomataan, että kohdissa B, F, G, H ja I siiven hyöty on merkityksetön. Yhteenlaskettu hyöty per kierros on 0.165 sekuntia.

0.165 sekuntia ei ole pitkä aika, mutta Formula Student -sarjassa erojen ollessa pieniä sillä on merkitystä. 2016 FSG Autocross osa-alueessa 0.165 sekunnin aikaparannuksella olisi noussut keskimäärin 1-3 sijaa ylöspäin tuloksissa.

## 4 MOOTTORIT

Järjestelmän ollessa kaksiasentoinen, toteutukseen on mahdollista käyttää solenoid-tyyppistä ratkaisua, servo- tai lineaarimoottoria. Kun koko auton massaa yritetään saada mahdollisimman pieneksi, on myös DRS-järjestelmän paino tärkeässä roolissa. Solenoidin massa kuparikäämeillä ja rautaytimellä nousee niin korkeaksi, että solenoidi jäi pois vaihtoehtoista. Järkevän hintainen ja pienikoinen lineaarimoottori on kahta muuta vaihtoehtoa hitaampi vaihtoehto. Saatavilla olevien lineaarimoottorien valikoima on myös pienempi kuin solenoidella ja servomoottoreilla. 1:4 ja 1:8 kokoisissa pienoismalli RC-autoissa käytetään paljon pieniä ja tehokkaita servomoottoreita. RC-autoihin tarkoitettuja servomoottoreita valmistetaan usean eri laitevalmistajan toimesta, joten saatavuus on hyvä ja vaihtoehtoja on paljon.

Servomoottori valikoitui sopivimmaksi ratkaisuksi. CFD-simulaatiolla ratkaistaan moottorin vaatima vääntömomentti, jonka perusteella valitaan sopivan kokoinen servomoottori järjestelmään.

### 4.1 CFD-simulaatio

CFD-virtaussimulaatiot tehtiin Solidworks 2016 Premium -ohjelmistolla. Opinnäytetyön tärkein osa on sähköinen toteutus, joten Solidworksiin päädyttiin aiemman kokemuksen ja virtaussimulaation helppokäyttöisyyden vuoksi.

#### 4.1.1 Simulointiparametrit

Simulointiprojekti luotiin Flow Simulation Wizardilla. Avustetussa ohjelmassa määritellään projektin perustiedot, käytetty yksikköjärjestelmä SI-järjestelmäksi, analyysin tyyppi ulkopuoliseksi virtaukseksi, kaasun tyyppi ilmaksi ja lopuksi liikkeestä syntyvä virtausnopeus Z-akselille.

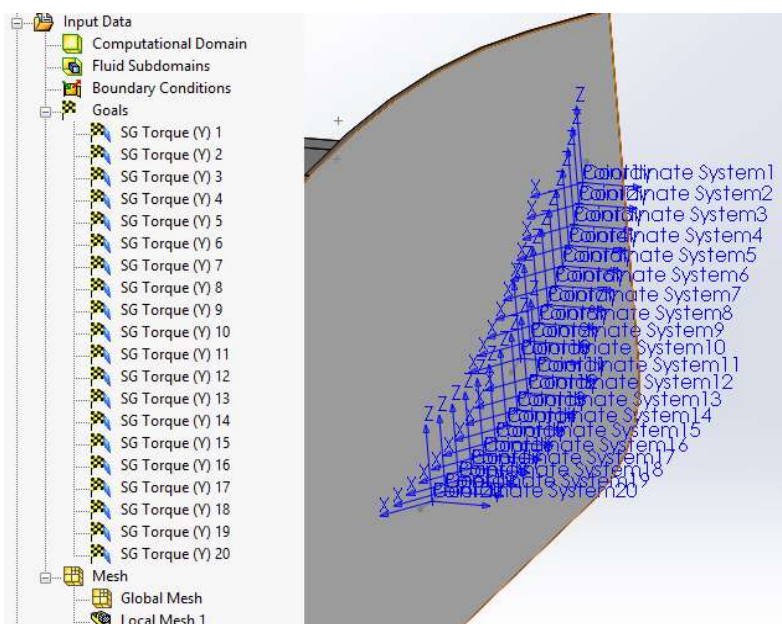
Solidworks Flow Simulaation simulaatioparametrit ovat hyvin rajoittuneet, mutta riittävät työn suorittamiseen. Esimerkiksi turbulenssimalli on ennalta määritetty, eikä sitä pysty muuttamaan.

#### 4.1.2 Siipiprofiilin nivelpiste

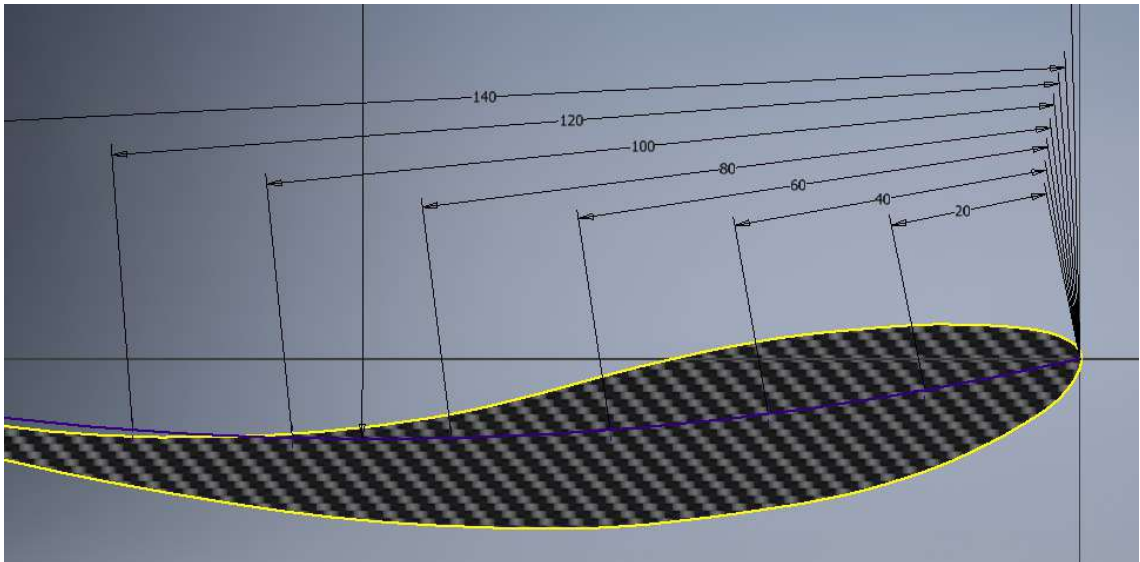
Nivelpiste on kääntyvässä profiilissa kohta, josta siipi on laakeroitu päätylevyihin. Nivel-piste vaikuttaa moottorin jatkuvaan momenttiin, jonka seurauksena pisteellä on suuri vaikutus moottorin mitoitukseen. Valittua pistettä käytetään lopuissa laskuissa.

On tärkeää optimoida myös järjestelmän virrankulutus. Yleisin tilanne radalla on ajotilanne, jossa siipi on kiinni. Mekaanisilla stoppareilla rajoitetaan siiven liike ala-asennossa ja samalla mitoitetaan saranapiste niin, että ilmavirta painaa profiileita kiinni stoppareita vasten. Stopparia vasten ollessa servomoottorin virrankulutus laskee huomattavasti, kun sen akselille ei kohdistu momenttia.

Takasiiven keski- ja yläprofiileita lähdettiin tutkimaan samanaikaisesti. Molempiin profiileihin tehtiin 10 simulointipistettä, joihin kohdistuvaa vääntömomenttia lähdettiin tutkimaan. Kuvassa 5 SG Torque (Y) 1-10 ovat ylempään profiilin pisteitä ja 11-20 alemman profiilin pisteitä.

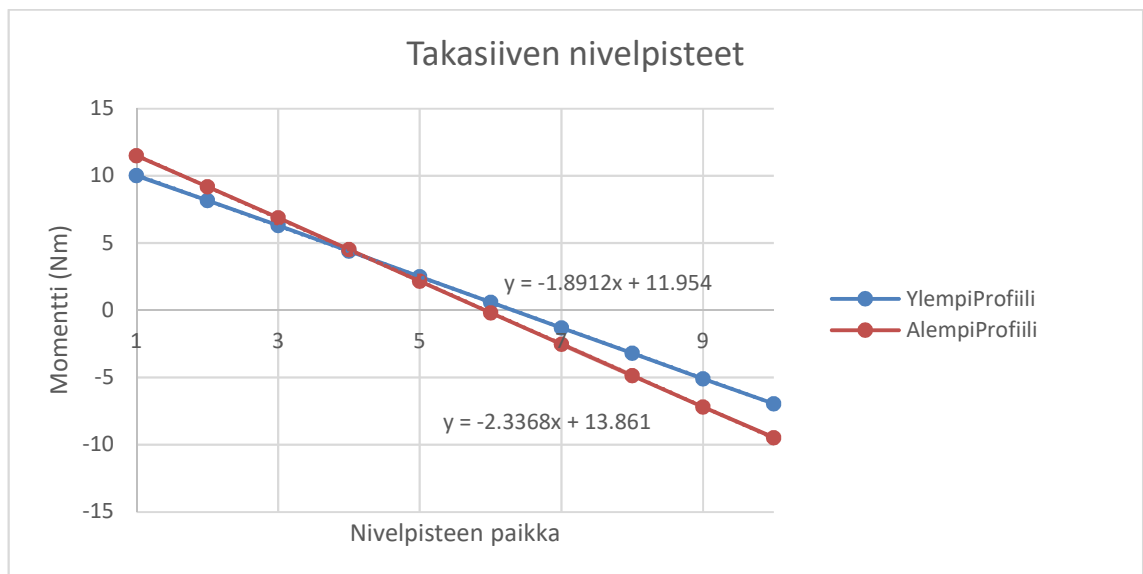


Kuva 5. Simulointipisteitä takasiiven profiileissa.



Kuva 6. Simulointipisteitä takasiiven profiilissa.

Kaikki pisteet mitoitettiin profiilin etureunan kohdalta (kuva 6). Ensimmäinen piste (piste 10) on 20 mm profiilin etureunasta ja loput pisteet 20 mm välein. Mitat ovat suoraa viivoja eivätkä kulje profiilin muodon mukaan. Simulointi suoritettiin 60 km/h nopeudella.



Kuvio 2. Nivelpisteen simulointitulokset.

Nivelpiste haluttiin paikkaan, jossa profiili vääntää 0.5 Nm väännöllä kiinni ilmapvirran vaikutuksesta (kuvio 2).

$$0.5 \text{ Nm} = -1.8912 * x + 11.954 \quad (13)$$

$$x = 6.0565$$

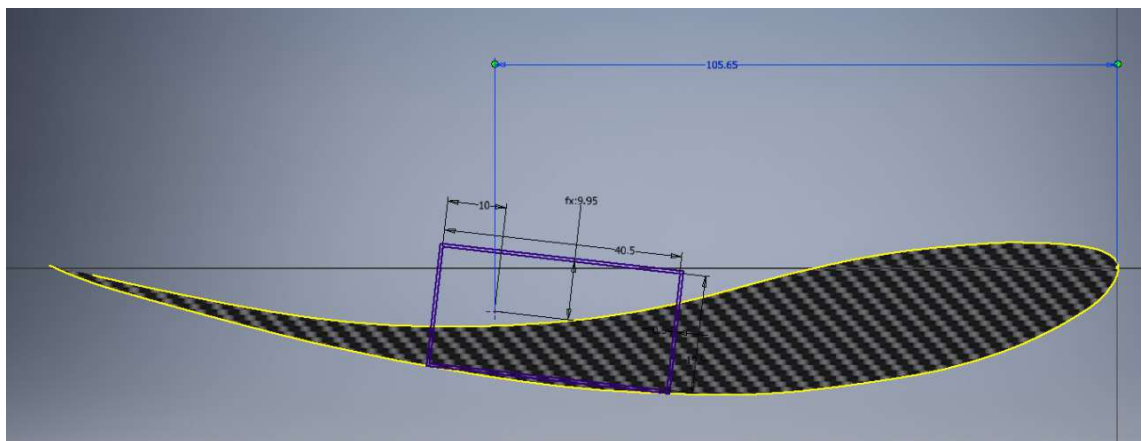
$$\text{Nivelpiste} = (10 - 6.0565) * 20 \text{ mm} + 20 \text{ mm} = 98.87 \text{ mm}$$

$$0.5 \text{ Nm} = -2.3368 * x + 13.861 \quad (14)$$

$$x = 5.7176$$

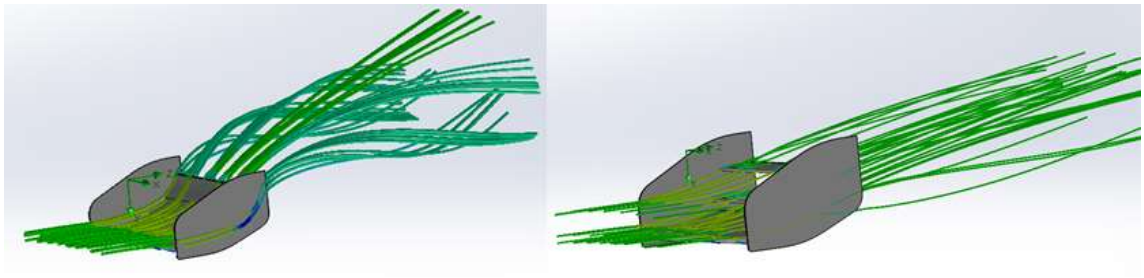
$$\text{Nivelpiste} = (10 - 5.7176) * 20 \text{ mm} + 20 \text{ mm} = 105.65 \text{ mm}$$

Trendiviivojen kaavoista 13 ja 14 laskettiin nivelpisteiden paikat. Molempien profiilien ollessa fyysisesti samanlaisia ja nivelpisteen eron ollessa pieni, päädyttiin asettamaan nivelpiste samaan kohtaan. Molemmat profiilit suunniteltiin 105.65 mm mukaan, jolla alempi profiili vääntää 0.5 Nm kiinnipäin ja ylempi profiili vähän enemmän (kuvio 4 ja kuva 7).



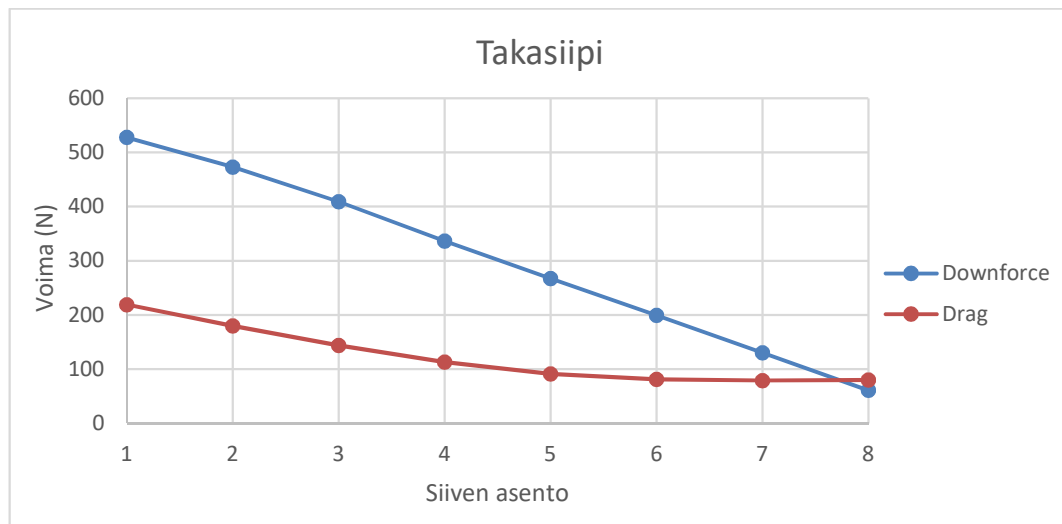
Kuva 7. Mitoitettu nivelpiste profiilissa.

### 4.1.3 Koko siiven simulaatiot



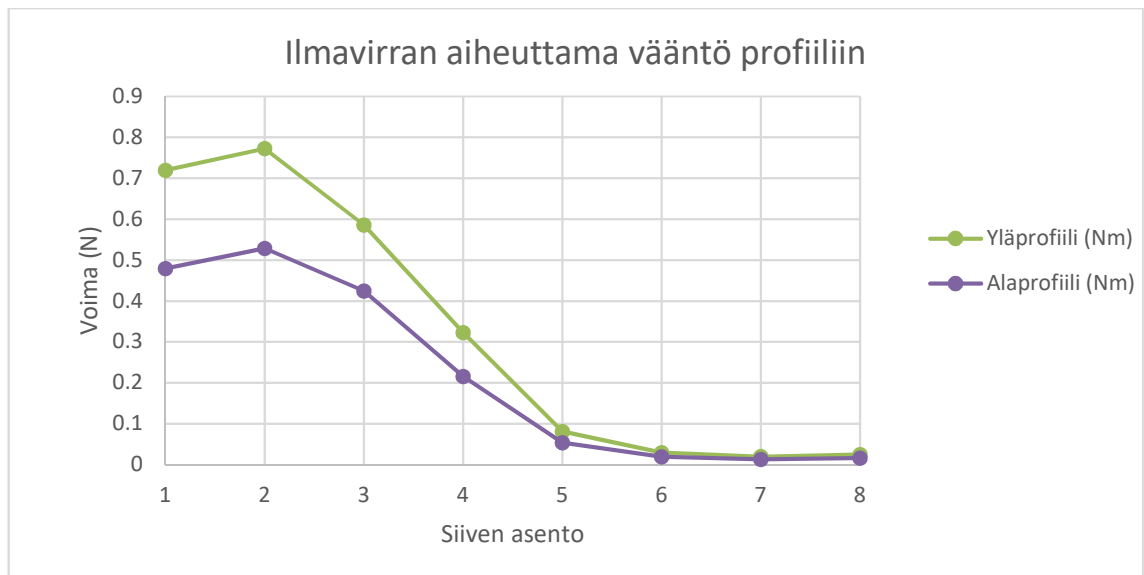
Kuva 8. CFD-simulaatio takasiivestä.

Kuviosta 3 huomataan, että siiven auetessa drag tippuu 219 N  $\rightarrow$  78 N ja downforce 525 N  $\rightarrow$  80 N.



Kuvio 3. Takasiiven drag ja downforce.

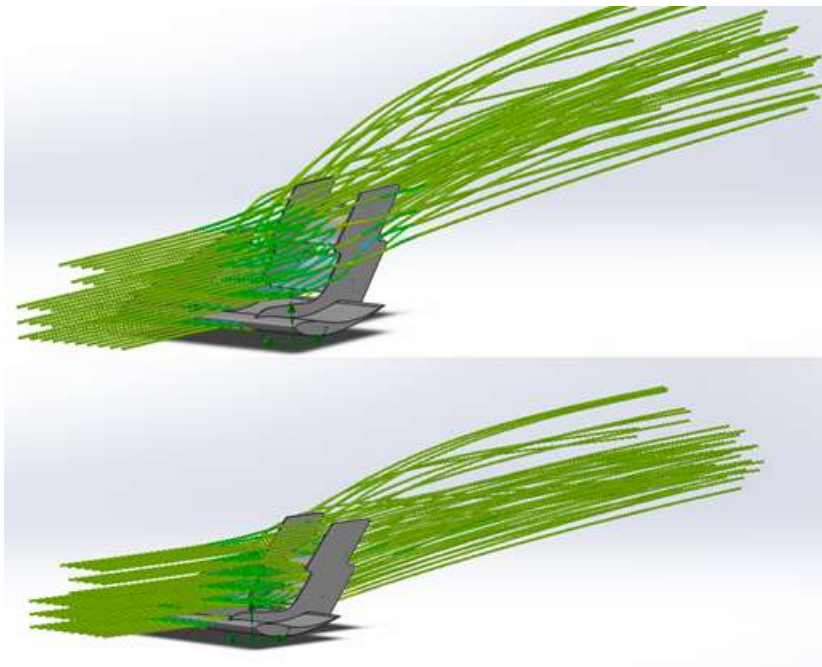
Siiven asennon muuttuessa profiileihin vaikuttava momentti on suurimmillaan siiven ollessa lähes kiinni (kuvio 4) ja lähestyy nollaa profiilin avautuessa. Liikettä mekaanisesti hidastavia tekijöitä ei ole otettu huomioon. Suuremmalla momentilla saavutetaan nopeampi profiilin liike. Tätä ei pystynyt Solidworksin simulaatiolla kokeilemaan.



Kuvio 4. Profiilien momentti (1 = kiinni, 8 auki).

#### 4.1.4 Etusiiven simulaatiot

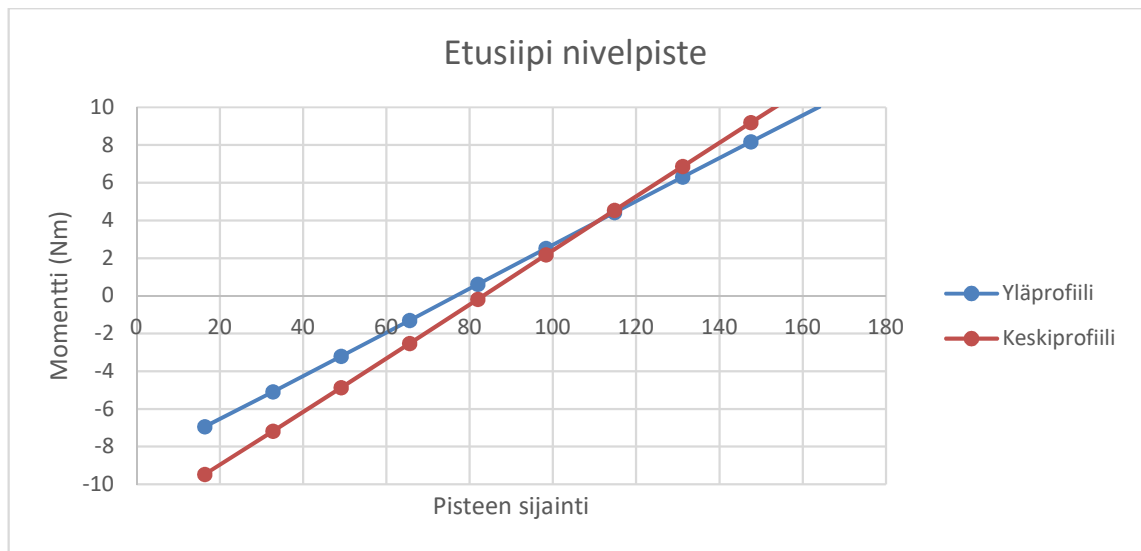
Etusiivelle tehtiin samanlaiset simulaatiot kuin takasiivelle (kuva 9).



Kuva 9. CFD-simulaatio etusiivestä.

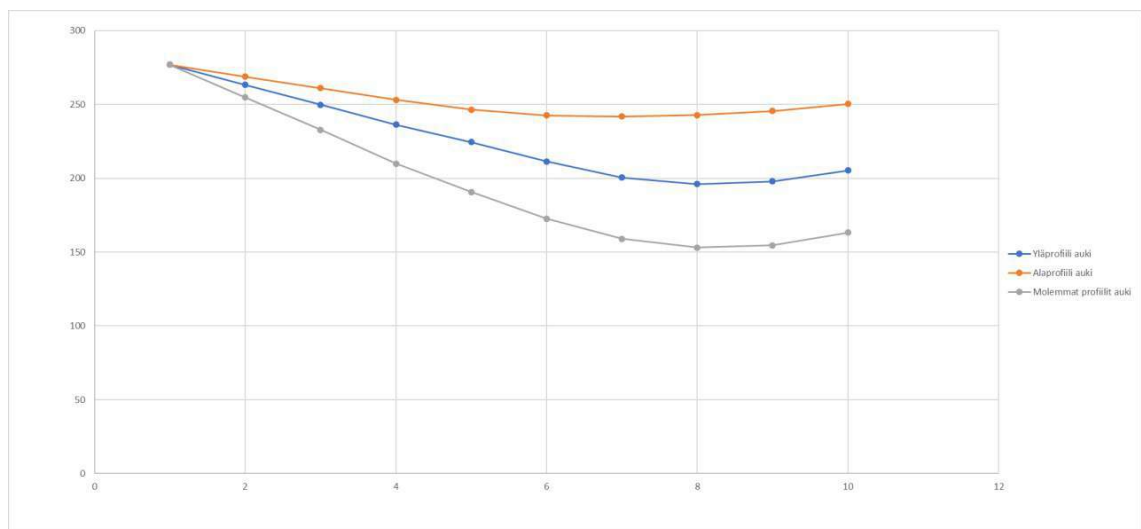
Eri pisteiden momentit simuloitiin. Kuviossa 5 pisteen sijainti millimetreinä. Etäisyys on mitattu profiilin etureunasta.





Kuvio 5. Etusiiven nivelpiste

0.5 Nm momentin sijainti laskettiin kuten takasiivellekin. Ylemmän profiilin nivelpisteeksi saatiin 81.1 mm ja alemman profiilin nivelpisteeksi 86.7 mm. Etusiiven profiilit ovat paljon pienemmät kuin takasiiven. Profiileita on 4 kpl, kaksi kummallakin puolella siipeä. Yhdelle pienelle profiilille riittää yksi moottori, jos profiilin toisen päädyn laakeroi päätylevyyn. Yhden moottorin käyttö helpottaa ohjauksen toteuttamista huomattavasti.



Kuvio 6. Etusiiven drag profiilien avautuessa.

Saaduilla nivelpisteillä simuloitiin siipiprofiilien asennot läpi, josta saatiin kuvion 6 mukaiset käyrät. Siiven ollessa kiinni drag oli noin 275 N. Kun pelkät alemmat profiilit avattiin, drag tippui 243 newtoniin. Kun pelkät ylemmät profiilit avattiin, drag tippui 198 newtoniin. Kaikkien profiilien avaamisella drag tippui 152 newtoniin.

Takasiivellä ilmanvastus pieneni huomattavasti enemmän profiileita käännettäessä. Tämä selittyy käännettävän profiilin koolla. Säännöt rajoittavat etusiiven profiilien kokoa. Pyörän edessä siiven korkeus saa olla maksimissaan 250 mm maasta, jättäen pyörän sisäpintojen väliin vain pienen tilan isommille siipiprofiileille

Takasiivessä järjestelmän hyöty on huomattavasti suurempi, joten järjestelmä päädyttiin toteuttamaan pelkästään taakse. Moottorien paikat ovat kuitenkin laskettu, joten DRS-järjestelmää voidaan hyödyntää myöhemmin myös etusiiven kanssa.

#### 4.2 Moottorin valinta

Servomoottorin tärkeimmät ominaisuudet olivat pieni koko ja suuresta vääntömomentista syntyvä nopea liike. Tarkkaan mitoitettun nivelpisteen johdosta profiili liikkuu pienelläkin väännöllä.

Pieniä servomoottoreita käytetään paljon 1:6 ja 1:8 radio-ohjattavissa pienoismalleissa. Hyvin yleinen moottorin koko näissä laitteissa on 40 mm x 40 mm x 21 mm. Tämän kokoluokan moottorin saisi vielä sovitettua kompaktiksi paketiksi siipiprofiilin sisälle.



Kuva 10. BlueBird BMS35A.

Tarjonnan ollessa hyvin suurta, kokoluokasta löytyy myös tehokkaita ja hyvin laadukkaita vaihtoehtoja.

Turnigy TGY-DS3509MG 40 kg/cm @ 8.4 V (3.92 Nm) oli vääntömomentiltaan suurin moottori, mikä kokoluokasta löytyi. Tuotetta ei enää valmisteta, joten sen saatavuus on hyvin huonoa.

BlueBird BMS35A 36 kg/cm @ 7.4 V (3.53 Nm) (kuva 10) oli yleinen, internetissä hyväksi valmistajaksi keuhuttu ja kohtuu hintainen. Vääntömomentti on ilmoitettu 7.4 voltin jännitteelle. Servomoottorin luvataan kestävän 8.5 voltia, jolloin myös vääntömomentti kasvaa.

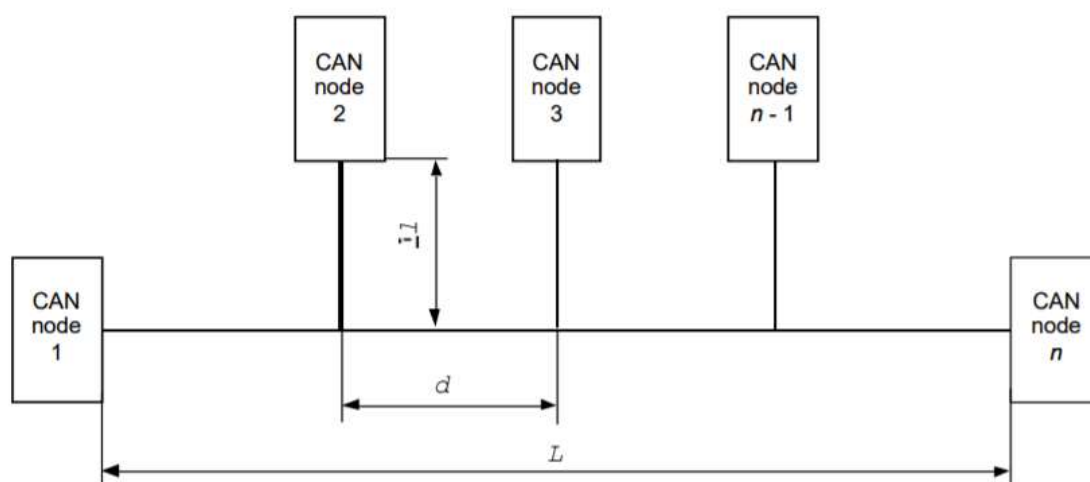
Kahdella moottorilla per profiili 7 Nm @ 7.4 V vääntö riittää nopeaan liikkeeseen ja titaanista valmistettu vaihteisto kestää perinteisiä teräsrattaita paremmin rasitusta. BlueBird BS35A hinta oli noin 60 €/kappale. (Blue Bird Technology n.d)

### **4.3 Moottorin muokkaus**

Moottorin sisäinen piirilevy hoitaa servon toiminnan ja takaisinkytkennän käsittelyn. Haluttiin kuitenkin takaisinkytkentätieto myös servo-ohjaimen käyttöön. Servomoottori purettiin ja takaisinkytkennän toteutusta tutkittiin. Servon sisällä on perinteinen analoginen potentiometri, jolle tuli 5 voltin käyttöjännite servon sisäiseltä piirilevyltä. Potentiometrin signaalijohtimesta haaroitettiin signaali servomoottorin ulkopuolelle, omavalmisteiselle ohjainpiirilevylle. Signaali haluttiin lukea häiritsemättä servon omaa ohjainta. Signaaliin lisättiin sarjavastus ja sen lukemiseen käytettiin myöhemmin esiteltyä operaatiovahvistipiiriä.

## 5 CAN-väylä

CAN eli Controller Area Network on ajoneuvoteollisuudessa laajasti käytetty väylä. CAN-väylä on suunniteltu luotettavaksi ja helposti laajennettavaksi. Väylässä ei ole erillistä isäntälaitetta (master), vaan kaikki laitteet (nodet) ovat saman arvoisia (kuva 11).



Kuva 11. CAN-väylän nodet (ISO 11898-2 1999, 8)

ISO 11898-2 Road vehicles – Controller area network (CAN) -standardi määrittelee autoteollisuudessa käytössä olevan CAN-väylän. CAN-väylässä on kaksi johdinta: CAN High (CANH) ja CAN Low (CANL). Johtimet ovat parikierrettyjä.

Väylän maksiminopeudella 1 Mbps väylä voi olla enintään 40 metriä pitkä (Kuvassa 11 mitta L). Ajoneuvokäytössä väylän pituus ei ole juuri koskaan rajoittava tekijä, mutta muita käyttökohteita varten voidaan käyttää laskukaavaa 15.

$$\text{Väylän maksimipituus (m)} = \frac{50}{\text{Väylän nopeus (Mbps)}} \quad (15)$$

(Texas Instruments 2008).

Runkoväylään tulee haaroja eri nodeille. Standardi suosittelee runkoväylästä haaroittuvalle väylälle (Kuvassa 11 mitta l) maksimipituudeksi 30 cm. Tätä pidemmällä haaroilla väylään syntyy heijastumia ja virheellisiä väylän kehyksiä.

CAN-väylä päätetään  $120\ \Omega$  päätevastuksilla molemmista ääripäistä, väylän kokonaisvastuksen ollessa  $60\ \Omega$ .

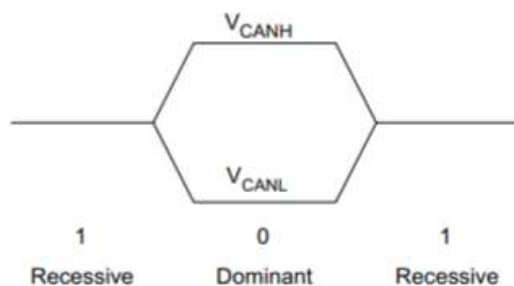
Lähetettävän viestin tunniste (identifier) määrää myös viestin prioriteetin. Kaikki viestit menevät väylän kaikille laitteille. Vastaanottava laite määrää, ottaako se vastaan kyseisellä tunnisteella olevan viestin.

### 5.1 CAN-väylän jännitetasot

Recessive-tilassa (1) CANH ja CANL välillä on alle 12 millivoltin jännite-ero, molempien linjojen ollessa 2.5 V potentiaalissa maatasoon verrattuna.

Väylän ollessa dominant-tilassa (0) CANH jännite on 3.5 V ja CANL jännite 1.5 V. Väylälaitteet tunnistavat dominant-tilan, kun väyläjohtinten jännite-ero on yli 1.2 V.

CAN-transeiver piirejä on 3.3 voltin käyttöjännitteellä toimiville ARM-prosessoreille soveltuvasti. Tämäntyyppinen piiri ei nosta CANH jännitettä standardiin 3.5 volttiin asti, mutta väyläjohtinten jännite-ero on riittävä 1.8 volttia. (ISO 11898-2 1999, 14)

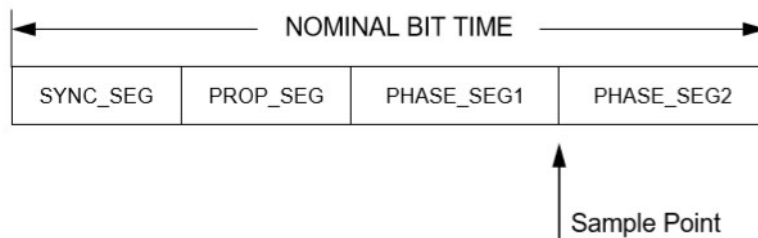


Kuva 12. CAN-väylän Recessive- ja Dominant- bitit.

## 5.2 Väylän ajoitus

Kaikki väylässä olevien laitteiden tulee käyttää samaa väylän nopeutta. Väylässä ei ole erillistä kellopulssia, joten väylän eri laitteet voivat ajautua keskenään eri tahtiin. Tämä voi johtua esimerkiksi pienestä kellotaajuuden toleranssista. Tästä syystä väylä onkin asynkroninen väylä. Väyläviestin ensimmäinen bitti SOF synkronoi väylän laitteet keskenään.

Väylän kellotaajuus muodostuu bittien ajasta (Nominal Bit Time), joka taas muodostuu 8-25 pienemmästä osasta  $t_q$  eng. ”Time Quanta”.



Kuva 13. CAN-väylän segmentit (Bosch 1991, 27).

Väylän yksi bitti jakautuu aika segmentteihin (kuva 13):

**Synchronization segmentissä** väylän muut laitteet tunnistavat lähettävän laitteen tahdin. SYNC\_SEG on aina  $1 t_q$  pitkä.

**Propagation time segment** kompensoi väylän fyysisen viiveen.

**Phase buffer segment 1&2.** Kahden viimeisen segmentin (Phase 1 ja Phase 2) välissä sijaitsee ”sample point” eli piste, jossa väylälaite lukee bitin tilan.

$$\text{Time Quanta } t_{BS1} = \text{PROP\_SEG} + \text{PHASE\_SEG1} \quad (16)$$

$$\text{Time Quanta } t_{BS2} = \text{PHASE\_SEG2} \quad (17)$$

$t_{BS1}$  ja  $t_{BS2}$  on määriteltävissä CAN-väyläohjaimen alustuksessa. ”Sample pointin” paikka määräytyy ”Time Quantien” määrästä ja jakaumasta. ”Sample pointin” paikka on tyypillisesti 75 % kohdalla koko bitistä. ”Sample Pointin” paikka saadaan laskettua kaavasta

$$\frac{1 + t_{BS2}}{1 + t_{BS1} + t_{BS2}} * 100 = \text{Sample Point} \quad (18)$$

75 % ”Sample point” kaavassa 18 toteutuu arvoilla

$$t_{BS1} = 2$$

$$t_{BS2} = 5$$

$$\frac{1 + 5}{1 + 2 + 5} * 100 = 75\%$$

(Bosch 1991)

### 5.3 CAN-viestin kehys

CAN-väylän viestin kehys on kuvan 14 mukainen.

#### Standard CAN



#### Extended CAN



Kuva 14. CAN-väylän viesti kehys (Texas Instruments 2008, 3-4).

**SOF (Start of Frame)**, kehyksen ensimmäinen bitti kertoo muille nodeille lähetettävästä viestistä. SOF-bitillä muut laitteet synkronoivat väylän ajoituksen.

**Identifier**, 11-bittinen tunniste, joka samalla määrää väylän prioriteetin. Kahden eri noden lähettäessä tunnistebittejä väylään ”0” on dominoiva. Jos toinen node huomaa väylän bitin muuttuvan dominantiksi (0), on pienemmällä ID:llä on lähtemässä viesti väylään. Jos taas node saa oman ID:nsä läpi, on viesti korkeimmalla prioriteetillä ja lähetystä voidaan jatkaa.

**RTR (Remote Transmission Request)**, CAN 2.0A -standardissa vastauspyyntö vastaanottavalle nodelle. CAN 2.0B -standardissa SRR-bitti, joka on merkityksetön.

**IDE (Identifier extension)**, jos bitti on dominant (0), käytetään CAN 2.0A -standardin mukaista 11 bittistä ID:tä. Jos bitti on recessive (1), käytetään 29-bittistä ID:tä ja 18 identifier bittiä on tulossa lisää.

**Identifier**, CAN 2.0B mukaista viestiä lähetettäessä 18-tunnistebittiä kuten aiemminkin. Tunnistebittien jälkeen RTR-bitti.

**r1** (vain 2.0B) ja **r0**, varabittejä standardin laajentamista varten, merkityksettömiä.

**DLC (Data Length Code)**, 4-bittinen luku, joka kertoo montako bittiä dataa on tulossa

**Data**, maksimissaan 64 bittiä dataa.

**CRC (Cyclic Redundancy Check)**, datan virheentarkastus summa.

**ACK (Acknowledges)**, jos data ja virheentarkastussumma täsmäävät, vastaanottava node kirjottaa bittiin dominantin.

**ACK delimiter ja EOF (End Of Frame)**, bitit ovat aina recessive (1), kertoo väylälle lähetyksen loppuvan.

**IFS** 7 bitin aikana väylä on tyhjänä antaen laitteille aikaa ottaa edellinen viesti talteen. (Texas Instruments 2008)

Pisimmillään CAN 2.0B mukainen viestikehys on 114 bittiä pitkä. 1 Mbps väylään mahtuu 8772 viestiä sekunnissa, lähetettävän viestin ollessa pisin mahdollinen (kaava 19).

$$\frac{1000000 \text{ bits/second}}{114 \text{ bits/frame}} = 8772 \text{ frames/second} \quad (19)$$



## 6 PIIRILEVYJEN VALMISTUS

DRS-järjestelmän ohjaus toteutettiin kahdella erilaisella piirilevyllä. Servo-ohjainpiirilevyllä ja anturisygnaaliluku CAN-moduuli piirilevyllä. Tiedonsiirto piirilevyjen välillä on toteutettu CAN-väylän välityksellä. Piirilevyjen suunnittelussa käytettiin Kicad-piirilevyn suunnitteluohjelmaa.

### 6.1 Piirilevyn suunnittelusäännöt

Piirilevyt tilattiin belgialaiselta Eurocircuitsilta. Suunnittelussa on tärkeää noudattaa piirilevyvalmistajan antamia suunnittelusääntöjä (Design rules). ”Design rules”-tiedosto Kicad-ohjelmalle on ladattavissa Eurocircuitsin sivuilta, jolloin mitoitus voi tarkistaa Kicadin omalla ”design rules check”-toiminnolla. On kuitenkin tärkeää ymmärtää mitä mikään suunnittelusääntö tarkoittaa.

Eurocircuits tarjoaa 4 erilaista pakettia valmistustoleransseja. Pienemmillä mitoilla valmistettu piirilevy on aina kalliimpi valmistaa. Eurocircuitsin tarjoamia paketteja ovat:

Naked proto	Piirilevy ilman soldermaskia ja silkscreeniä
PCB proto	2 tai 4 kerroksinen levy ilman silkscreeniä
Standard poll	2-16 kerroksinen levy soldermaskilla ja silkscreenillä
BINDI poll	Suurempia piirilevy eriä Intiasta
SEMI-FLEX poll	Piirilevyjä taipuisalla flex-kaapelilla
RF poll	Isola I-TERA materiaali korkeille taajuuksille
IMS poll	Alumiiniytiminen PCB hyvällä lämmönjohtavuudella

Piirilevypaketiksi sopivia olisivat PCB Proto ja Standard Poll. Naked proto ei olisi mahdollinen, koska piirilevyt juotetaan itse kasaan. Soldermask on välttämätön pienille komponenteille käsin juotettaessa. Valinta kohdistui Standardiin, koska Proton ja Standardin välillä ei ollut hintaeroa.

Kuparoinnin paksuutena käytettiin Standardin piirilevyn oletusta 18 µm sisemmissä kerroksissa ja 35 µm ulommissa kerroksissa. (Eurocircuits n.d.)

### 6.1.1 Pattern Class

Seuraavaksi Standard-piirilevylle valitaan kuvioluokka (Pattern Class) taulukossa 5.

Taulukko 5. Eurocircuitsin kuvioluokat.

Pattern Class	3	4	5	6	7	8	9	10
OTW	0,250	0,200	0,175	0,150	0,125	0,100	0,090	<0,090
OTT-OTP-OPP	0,250	0,200	0,175	0,150	0,125	0,100	0,090	<0,090
OAR	0,200	0,150	0,150	0,125	0,125	0,100	0,100	<0,100
ITW	0,250	0,200	0,175	0,150	0,125	0,100	0,090	<0,090
ITT-ITP-IPP	0,250	0,200	0,175	0,150	0,125	0,100	0,090	<0,090
IAR	0,200	0,150	0,150	0,125	0,125	0,125	0,125	<0,125
IPI	0,275	0,225	0,225	0,200	0,200	0,200	0,200	<0,200

Ensimmäinen kirjain O tai I määrittelee missä piirilevyn kerroksissa sääntö on voimassa.

O	Outer	Piirilevyn uloimmat kerrokset
I	Inner	Piirilevyn sisällä olevat kerrokset

Jälkimmäisten kahden merkin selitykset:

TW	Track Width	Vetojen leveys
TT	Track to Track	Vetojen välinen etäisyys
TP	Track to Pad	Vedon ja komponentin jalan välinen etäisyys
PP	Pad to Pad	Komponenttien jalkojen välinen etäisyys
AR	Annular Ring	Läpiviennin kuparoinnin leveys
PI	Pad Isolation	Läpiviennin reiän etäisyys kupariin

Pattern Class 7 alkaen piirilevyn valmistuskustannukset kasvoivat huomattavasti, joten suunnitteluun käytettiin Pattern Class 6:n mukaisia mittoja. Pattern Class 6 mukaisilla mitoilla on vielä mahdollista suunnitella kompaktin kokoinen piirilevy tähän tarkoitukseen. (Eurocircuits n.d.)

### 6.1.2 Drill Class

Porausluokka (Drill Class) määrittelee pienimmän valmistuksessa käytettävän porante-rän. Porauksia on kahden tyyppisiä: läpivientejä (VIA) ja läpijuotettavien komponenttien jalkoja (Through Hole). Eurocircuits määrittelee kaikki 0.45 mm ja pienemmät reiät läpi-vienneiksi, joiden valmistustoleranssi on suurempi kuin läpijuotettavien komponenttien. Kaikki reiät porataan millikokoisina 0.05 mm välein. Todellinen reiän koko on kuitenkin 0.1 mm suurempi kuin terä.

Taulukko 6. Eurocircuitsin porausluokat.

Drill Class	A	B	C	D	E	F
min PHD	0,60	0,45	0,35	0,25	0,20	<0,20

PHD            Production Hole Diameter

Todellisen reiän koko

Samoin kuin Pattern Class kanssa, Drill Class D alkaen piirilevyn valmistuskustannukset kasvoivat huomattavasti, joten suunnitteluun käytettiin Drill Class C:n mukaisia mittoja. (Eurocircuits n.d.)

## 6.2 Piirilevyjen tilaus

Lähes kaikki piirilevyvalmistajat käyttävät gerber-formaattia. Eurocircuits hyväksyy myös suosittujen piirilevysuunnitteluohjelmien Eaglen ja Kicadin kicad\_pcb tiedostofor-maatit. Ohjelmisto tarkistaa piirilevyn edellisen kappaleen suunnittelusääntöjen mukai- sesti ja kertoo piirilevyn virheellisistä kohdista.

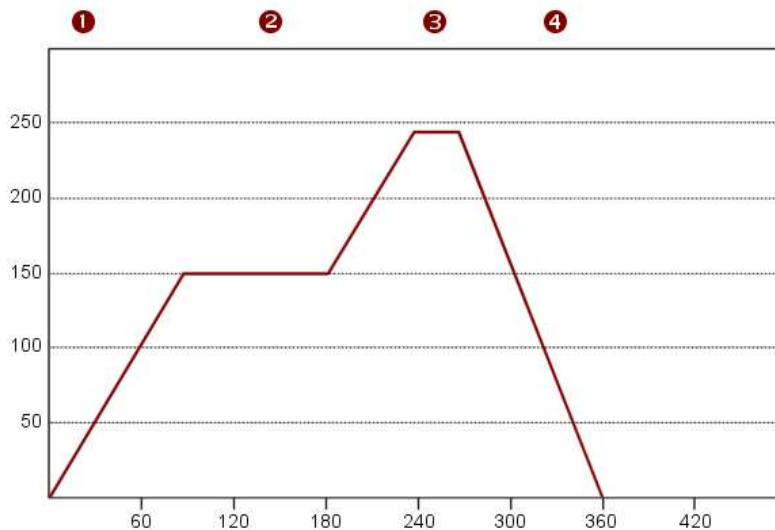
Eurocircuitsin web-pohjainen ohjelma näyttää kaikki kuparikerrokset, printtikerrokset, lakkakerrokset, läpivientien kuparoinnit yms. Tässä vaiheessa on vielä mahdollista pois- taa ylimääräisiä tekstejä lopullisesta piirilevystä.

Tilauksesta ja maksusta piirilevyt ovat nopeimmillaan toimitettu Suomeen viikossa.

### 6.3 Komponenttien juottaminen

Perinteisen kolvin ja tinan sijaan pienemmät pintaliitoskomponentit on helpompi juottaa juotospastan avulla. Pastaa levitetään piirilevylle komponenttien jalkojen kohdalle. Tässä on hyvä käyttää piirilevyn mukana tilattavaa stensiiliä. Stensiili on ohut alumiinilevy, jossa komponenttien jalkojen kohdalla on oikean kokoiset reiät. Juotospasta levitetään stensiilin päälle ja tasoitetaan levyn yli niin, että jokaisen komponentin jalkaan tulee juotospasta. Ylimääräiset juotospastat pyyhitään pois ja stensiili nostetaan piirilevyn päältä.

Tämän jälkeen komponentit ladotaan piirilevylle ja lämmitetään juotosuunissa piirilevyvalmistajan antaman lämpöprofiilin mukaisesti.



Kuvio 7. Juotosuunin yleinen lämpötilaprofiili (CompuPhase n.d.).

Kuvassa 7 on esimerkki juotosuunin lämpöprofiilista.

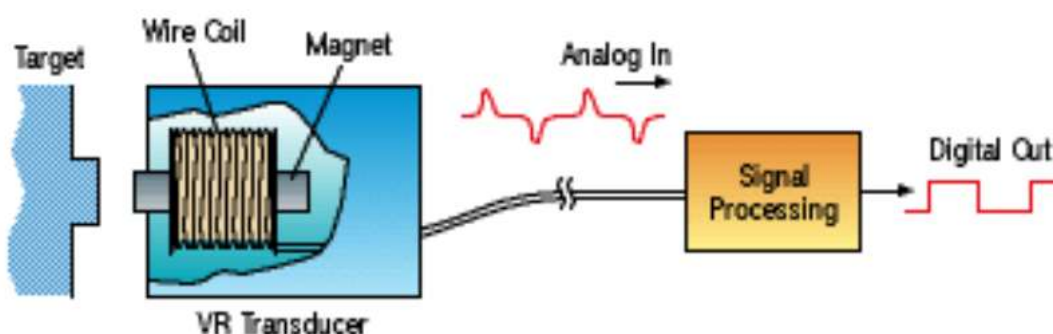
1. Piirilevy esilämmitetään ja juotospastan fluxi aktivoituu. Lämpötilan nousu maksimissaan 3 °C/s.
2. n. 150 °C odotetaan, että kaikki suurimmatkin kuparialueet saavuttavat tasaisen lämpötilan
3. Lämpötilaa nostetaan n. 2 °C/s juotospastan sulamispisteeseen ja ainakin 25 °C sen yli. Lämpötilaa pidetään yllä noin 10 sekunnin ajan.
4. Lämpötila lasketaan n. 4 °C/s takaisin huoneenlämpöön.

(CompuPhase n.d.)

## 7 PIIRILEVYJEN SUUNNITTELU

### 7.1 CAN-moduuli

CAN-moduulin tehtävä on lukea analoginen tai digitaalinen anturisygnäali ja lähettää tieto CAN-väylään. CAN-moduulista pyrittiin tekemään mahdollisimman monikäyttöinen muidenkin antureiden lukemiseen. Jokainen kanava suunniteltiin lukemaan sekä analogisia että digitaalisia signaaleita.



Kuva 15. VR-anturin toimintaperiaate (Ed Ramsden 2000).

Haasteen suunnitteluun toi VR (Variable Reluctance Sensor) pyörintänopeusanturi (kuva 15). Kun ferromagneettinen aine ohittaa anturin magneettikentän, magneettikenttä muuttuu ja käämiin indusoituu vaihtojännite. Kun mitattavan pulssin taajuus kasvaa myös signaalin amplitudi kasvaa. Mitattaessa kovalla vauhdilla pyörivää kappaletta, kuten ahdetun moottorin ahtimen pyörintänopeutta, signaalin amplitudi saattaa nousta jopa 50 Vpp. (Ed Ramsden 2000)

Ensimmäisenä piirilevyn suunnittelussa tuli päättää kanavien lukumäärä. Kanavien määrä vaikuttaa olennaisesti moduulin liittimen valintaan ja prosessorin kokoon.

#### 7.1.1 Liittimet

Liittimien valinnalla oli suuri vaikutus CAN-moduulin fyysiseen kokoon ja toteutukseen. Liitin on piirilevyn korkein komponentti, määräten samalla moduulin paksuuden. Liitinvalinnassa kriteereinä olivat liittimen kompakti koko, saatavuus, pinnien määrä ja hinta.

Moduulia on tarkoitus käyttää myös pyörännopeuksien (4), iskunvaimentimien asentojen (4), ratin asennon, jarrupaineiden (2) ja virrankulutuksen mittaukseen. Moduulista päätettiin tehdä 16-kanavainen, mikä on STMicroelectronics ARM-prosessorien portin koko rajoittaen monien ominaisuuksien määrän kuuteentoista. (STmicroelectronics 2018a, 176)

16 kanavan lisäksi tarvitaan pinnit käyttöjännitteelle (+12V ja GND), CAN-väylälle (CANL ja CANH) ja antureiden käyttöjännitteelle (+5V ja SGND). Liittimessä tulee olla vähintään 22 pinniä.

Vaihtoehtoja tutkittiin suurilta liitinvalmistajilta ja vaihtoehtoiksi jäivät Molex CMC ja CMX -sarja, Delphi Sicma, AMP 1.0 -sarja ja Deutsch Autosport -sarjat. Vaihtoehdot jakautuivat selvästi kahteen hintaluokkaan metallirunkoisten Autosport liittimien ollessa noin kolme kertaa kalliimpia kuin muovirunkoiset Molex, Delphi ja AMP. AMP 1.0 -sarjan 26 pinninen liitin on muovirunkoisista vaihtoehtoista kompakteimman kokoinen ja laajasti saatavissa.

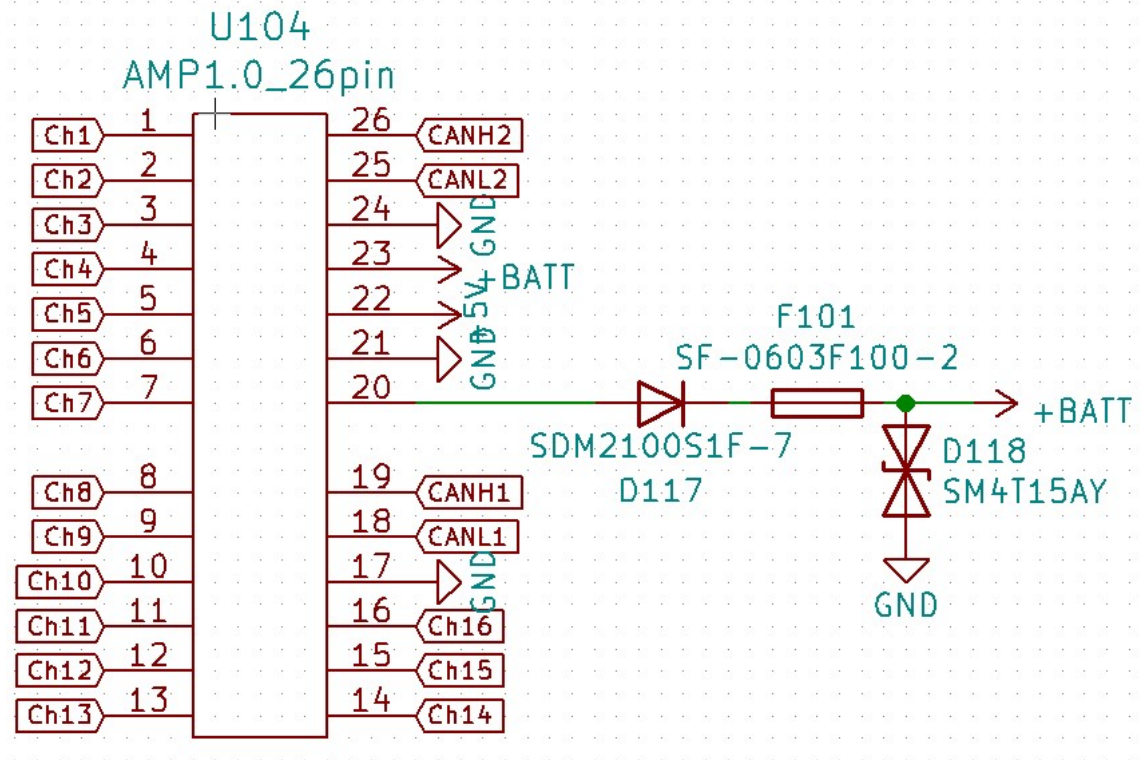
Autosport liitoksen kustannuksien kasvaessa lähemmäs 100 euroa, päädyttiin AMP 1.0 sarjan 26 -pinniseen liittimeen, joka on myös vesitiivis IP67 luokituksellaan. Ylimääräiset pinnit mahdollistivat myös toisen CAN-väylän lisäämisen moduuliin. (TEconnectivity 2010)

CAN-moduulin pinnijärjestys taulukossa 7.

Taulukko 7. CAN-moduulin pinnijärjestys

<i>Pinnin numero</i>	<i>Selite</i>
1	Mittauskanava 1
2	Mittauskanava 2
3	Mittauskanava 3
4	Mittauskanava 4
5	Mittauskanava 5
6	Mittauskanava 6
7	Mittauskanava 7
8	Mittauskanava 8
9	Mittauskanava 9
10	Mittauskanava 10
11	Mittauskanava 11
12	Mittauskanava 12
13	Mittauskanava 13
14	Mittauskanava 14
15	Mittauskanava 15
16	Mittauskanava 16
17	GND
18	CAN1L
19	CAN1H
20	+12V käyttöjännite
21	GND käyttöjännite
22	+5V ulostulo
23	+12V ulostulo
24	GND
25	CAN2L
26	CAN2H

### 7.1.2 Piirilevyn suojaus



Kuva 16. Piirilevyn liitin ja jännitteensyötön suojapiiri.

Käyttöjännite suojattiin väärältä napaisuudelta sarjadiodin avulla. SOD-123 paketissa oleva DIODES INC SDM2100S1F-7 diodi suojaa piiriä 100 V negatiiviselta jännitteeltä ja kestää 2A jatkuvan virran. Myötäsuunnassa diodin yli jää 830 millivoltin kynnyksjännite. (Diodes Incorporated 2016)

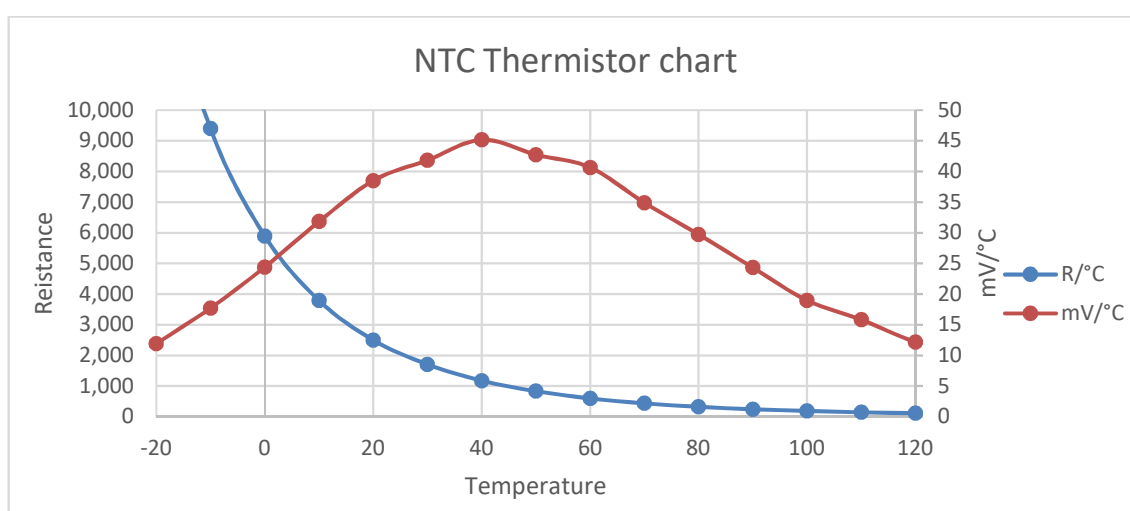
Sarjadiodin lisäksi käyttöjännitteessä on STMicroelectronicsin SM4T15AY TVS (transient voltage suppression) diodi. SM4T15AY on 15 voltin TVS diodi, joka pyrkii leikkaamaan ylimenevän jännitteen pois. Tarkoituksena on suojata piiriä korkeilta ESD jännitepiikeiltä. (STmicroelectronics 2015)

Käyttöjännitteessä on myös yhden ampeerin pintaliitos-sulake, Bourns SF-0603F100-2. Sulake suojaa piirilevyn vetoja ylivirralla. (Bourns 2017)



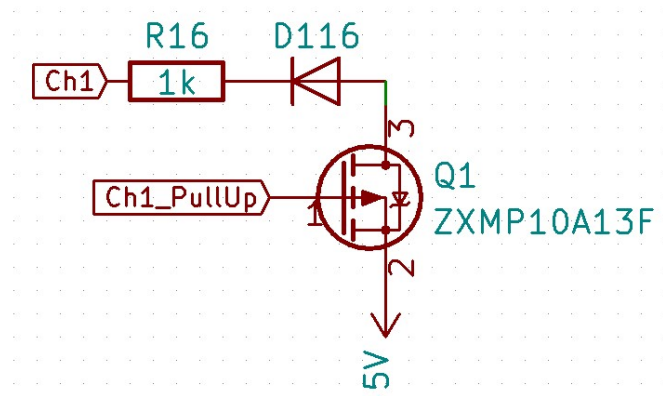
### 7.1.3 Kanavan kytkentä

Kanavissa on ohjelmallisesti kytkettävä 1 k $\Omega$  ylösvetovastus. Ajoneuvokäytössä lämpötila anturit ovat yleisimmin NTC-elementillä varustettuja. NTC elementin vastuksen suhde lämpötilaan ei ole lineaarinen, joten ylösvetovastuksen koolla voidaan vaikuttaa tarkimpaan mitta alueeseen. Mittaussignaalin mV/°C arvo on suurimmillaan, kun ylösvetovastuksen ja anturin vastuksen arvot ovat lähellä toisiaan. NTC-elementin 1 k $\Omega$  vastus on noin 45 °C asteen kohdalla. Tyypillisesti mitattavat kohteet ovat ilmanlämpötilasta 120 °C lämpötilaan. 1 k $\Omega$  vastuksella saavutetaan mahdollisimman laaja mitta-alue näille lämpötiloille (Kuvio 8).



Kuvio 8. NTC elementin vastusarvot.

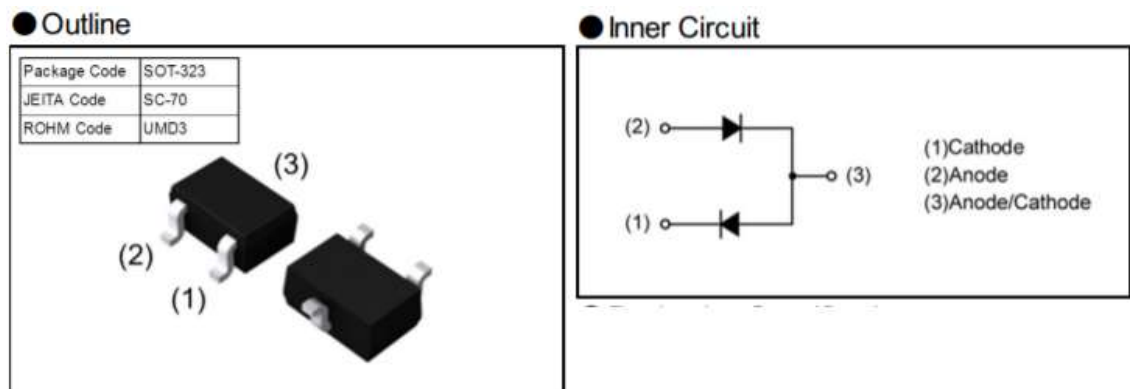
5 voltin ylösvetovastusta ohjaa P-kanavan ZXMP10A13F MOSFET (kuva 17). Muussa käytössä kanavan jännite voi kuitenkin olla enemmän kuin 5 voltia, jolloin MOSFETin nielu (Drain) on suurempi kuin lähde (Source). Tätä MOSFET ei kestä, joten ylösvetovastuksen kanssa sarjaan kytkettiin pienen kynnsjännitteen TCA9535 diodi. Diodi kuitenkin aiheuttaa jännitehäviön ylösvetoon, jonka voi kompensoida nostamalla ylösvetojännitettä, jolloin diodin kynnsjännitteen jälkeen ylösveoto olisi 5 voltia. Tavoite oli saada piirilevystä mahdollisimman kompaktin kokoinen, joten diodin jännitehäviö kompensoidaan ohjelmallisesti.



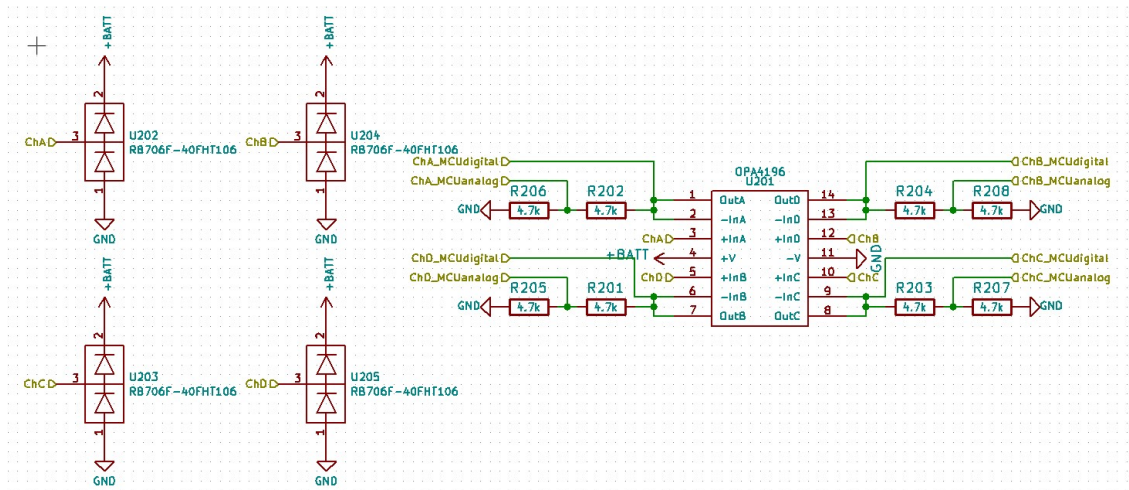
Kuva 17. Kanavan ylösveto.

Kanavaan lisättiin 20 k $\Omega$  sarjavastus, jolla minimoidaan kanavan vaikutus mitattavaan signaaliin ja rajoitetaan mahdollista vikatilanteen aiheuttamaa oikosulkuvirtaa.

Käyttöjännitteen lisäksi myös mittauskanavat on suojattu jännitepiikeiltä ROHM semiconductorin RB706F-40FH pienen kynnyksjännitteen schottky barrier diodisillalla (kuva 18). Diodit leikkaavat mittauskanavan negatiivisen jännitteen -0.2 volttiin ja positiivisen jännitteen  $V_{BATT}+0.2$  volttiin. Tällä suojataan kanavan operaatiovahvistinta, jonka kanavat kestävät -0.5 V  $\rightarrow$   $V_{BATT}+0.5$  V jännitteen. (ROHM Semiconductor 2016)



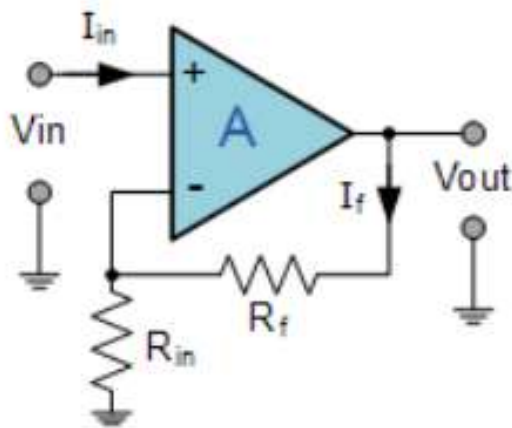
Kuva 18. ROHM Semiconductor RB706F-40H Schottky Barrier Diode (Semiconductor 2016).



Kuva 19. Kanavan schottky diodi, operaatiovahvistin ja jännitteenjako.

ARM-prosessorin maksimijännite on 3.3 V, joten 5 voltin jännite tulee jakaa pienemmäksi jännitteenjakovastuksilla (kuva 19). Jännitteenjakovastukset kuitenkin pienentävät kanavan impedanssia, joka saattaa vaikuttaa kytketyn anturin signaaliin. Sisään tuleva jännite vahvistetaan operaatiovahvistimella ennen jännitteenjakoa.

Operaatiovahvistin jänniteseuraajana (eng. Voltage Follower tai Buffer) vahvistaa mitatun signaalin.



Kuva 20. Ei-invertoiva operaatiovahvistin.

Jännitteenjako toimii kuvan 20 mukaisen ei-invertoivan operaatiovahvistin kytkennän mukaan. Kaavassa 20.

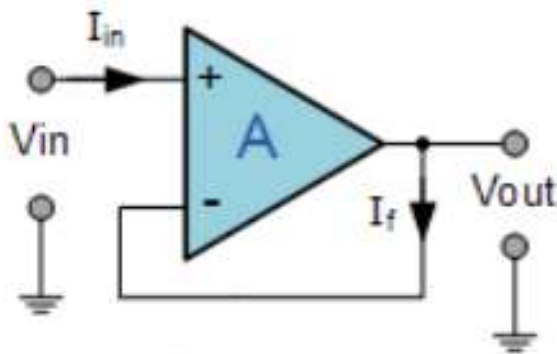
$$V_{OUT} = V_{IN} \cdot \left(1 + \frac{R_f}{R_{in}}\right) \quad (20)$$

Kun  $R_f$  on nolla ja  $R_{in}$  ääretön, saadaan kytkennäksi kuvan 21 mukainen jänniteseuraaja.

$$V_{OUT} = V_{IN} \cdot \left(1 + \frac{0\Omega}{\infty}\right)$$

$$V_{OUT} = V_{IN} \cdot 1$$

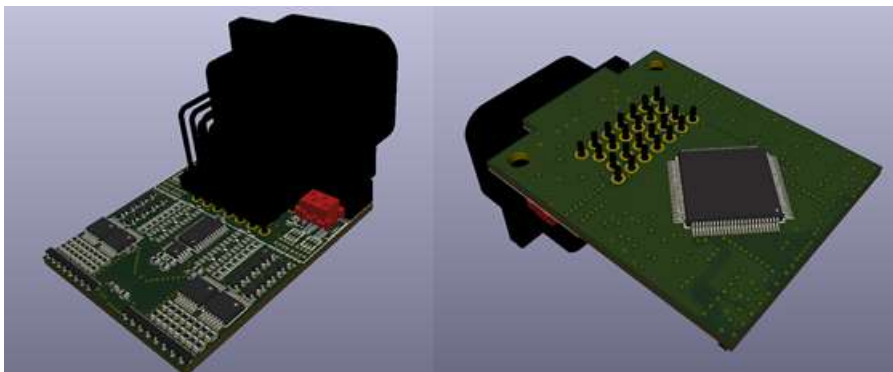
$$V_{OUT} = V_{IN} \quad (21)$$



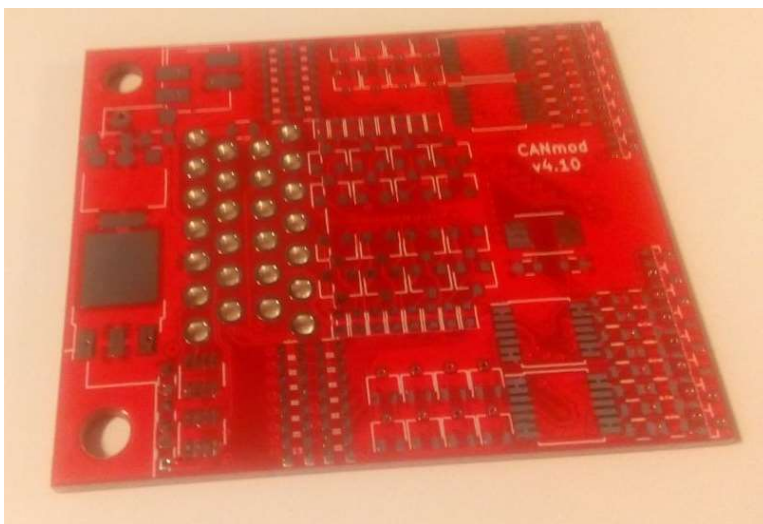
Kuva 21. Jänniteseuraaja.

(Electronics Tutorials. n.d.)

Operaatiovahvistimen lähtöön on kytketty pieni SC-75 paketissa oleva N-kanavan MOSEFET, joka vetää prosessorin pinniä maihin. Prosessorin sisäinen ylösvetovastus nostaa jännitteen takaisin ylös. Prosessorilla voidaan lukea suoraan kanavan digitaalista tietoa ja keskeytyksien avulla laskea korkeataajuisia PWM-signaalia.



Kuva 22. CAN-moduuli piirilevy Kicadissä.



Kuva 23. Valmis CAN-moduulin piirilevy.

Koko piirilevyn kytkentä liitteessä 1.

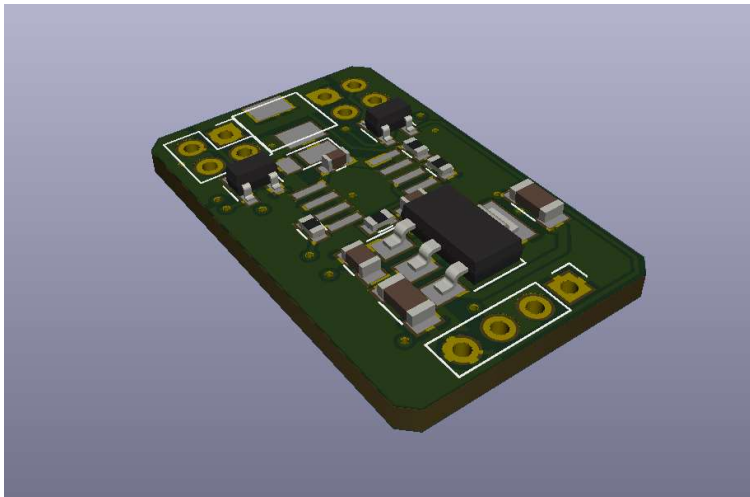
## 7.2 Servo-ohjain

Servoille rakennettiin CAN-väylällä kommunikoiva ohjainlevy. Ohjainlevyllä on mikrokontrolleri servon ohjaukseen, takaisinkytkentään luku ja CAN-väyläpiiri kommunikointiin. Jännitteensyöttö on erillisellä piirilevyllä saman kotelon sisällä.

### 7.2.1 Jännitteensyöttö

Moottorin maksimi virta 2.4 A @ 7.4 V käyttöjännitteellä. Piiri saa käyttöjännitteen 12 V akusta. Kahdelle servolle sopiva regulaattoriipiiri suunniteltiin 5 A virralle (kuva 24).

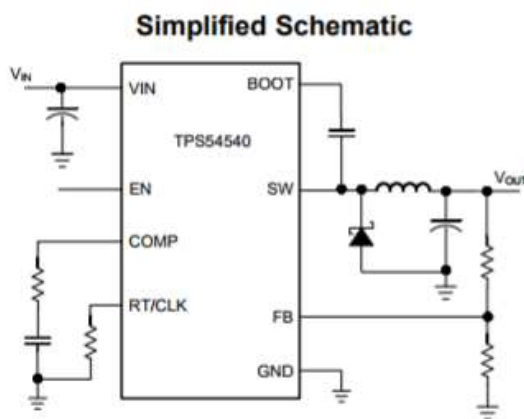
Regulaattoreita on kahden tyyppisiä, lineaarisia ja hakkureita. Lineaaristen regulaattoreiden sisäinen jännite muuttuu kuorman mukaan, pitäen ulostulojännitteen vakiona. Lineaarinen regulaattori hukkaa kaiken tiputetun jännitteen lämmöksi. Tasainen ulostulojännite on lineaarisen regulaattorin hyvä ominaisuus ja soveltuukin hyvin esim. pienivirtaisille mikrokontrollereille. Lämpenemisen takia se ei sovellu isovirtaisiin sovelluksiin.



Kuva 24. 5A 7.4V hakkurivirtalähde.

Hakkuriregulaattorit (Buck/Switching regulator) katkovat sisään tulevaa jännitettä korkealla taajuudella, jonka jälkeen jännite tasataan uudestaan tasajännitteeksi kelalla ja kondensaattoreilla. Hakkuriregulaattorin hyötysuhde onkin paljon parempi kuin perinteisen lineaarisen. (Renesas n.d)

Regulaattoriksi valittiin Texas Instrumentsin TPS54540 hakkuriregulaattori. 5 A regulaattorissa on takaisinkytkennällä säädettävä ulostulojännite. TPS54540 pieni fyysinen koko HSOP8 paketissa ja hyvä n. 90 % hyötysuhde vaikuttivat valintaan. Hakkuriregulaattorin kytkentä kuvassa 25.



Kuva 25. Texas Instruments TPS54540 ohjeellinen kytkentä (Texas Instruments 2013).

**VIN** – Regulaattorin sisääntulo jännite 4.5 V – 42 V. Tässä tapauksessa syötetään 12 voltin akkujännite

**EN** – Enable-pinniin voidaan tehdä sisäänmenevästä jännitesignaalista jännitteenjaolla alijännitesuoja. Tässä tapauksessa jätettiin kytkemättä, koska ei ole tarvetta kytkeä koko käyttöjännitettä pois.

**COMP** – PWM hakkurin sisäänmenon ja ulostulon vertailija. Pinniin kytketään kompensointiokondensaattori, jonka arvo saadaan laskettu kaavasta

**RT/CLK** – Hakkurin taajuus ja ajoituspinni. Hakkurin taajuudeksi voidaan asettaa 100kHz – 2500 kHz. Taajuutta nostettaessa komponentti arvot ja koko pienenevät, mutta samalla hyötysuhde heikkenee ja hakkurin pienin mahdollinen pulssin leveys nostaa minimi ulostulojännitettä.

$$\begin{aligned} V_{OUT(\min)} &= V_{IN} \cdot On\ time_{\min} \cdot f_{SW} & (22) \\ V_{OUT(\min)} &= 15V \cdot 135\ ns \cdot 2500kHz \\ V_{OUT(\min)} &= 5.06V \end{aligned}$$

Edes maksimitaajuus ei rajoita ulostulon jännitettä. Hyötysuhteen takia päädyttiin kuitenkin esimerkkiipiireissäkin olevaan 500 kHz hakkuritaajuuteen. (King B. & Nowakwski R. 2006)

Taajuus asetetaan  $R_T$  vastuksella, jonka arvo on laskettavissa kaavasta

$$\begin{aligned} R_T(k\Omega) &= \frac{101756}{f_{SW}(kHz)^{1.008}} & (23) \\ R_T(k\Omega) &= \frac{101756}{500kHz^{1.008}} \\ R_T &= 193.6\ k\Omega \end{aligned}$$

**BOOT** – Hakkurin MOSFETin toiminnan kannalta pinniin on sijoitettava ”bootstrap” kondensaattori. TPS54540 datakortin mukaan BOOT ja SW pinnien väliin tulee kytkeä 0.1μF keraaminen kondensaattori.

**SW** – Hakkurin ulostulo.

**FB** – Takaisinkytkennällä säädetään ulostulojännite. Jännitteenjakovastuksien  $R_{HS}$  ja  $R_{LS}$  arvot saadaan laskettua kaavasta

$$R_{HS} = R_{LS} \cdot \left( \frac{V_{OUT} - 0.8V}{0.8V} \right) \quad (24)$$

Vastuksien suhteen ollessa oikea, vastuksien arvolla ei ole suurta merkitystä.  $R_{LS}$  arvoksi valitaan 1 k $\Omega$ , jolloin

$$R_{HS} = 1000\Omega \cdot \left( \frac{7.4V - 0.8V}{0.8V} \right)$$

$$R_{HS} = 8250 \Omega$$

**GND** – Kaikki virta kulkee tämän maapinnan kautta, kytketään isoon maataloon. Hakkurin pohjassa oleva lämpölevy ”Thermal Pad” juotetaan myös maataloon, josta komponentin tuottama lämpö johtuu pois.

Piirilevyn kytkentä liitteessä 2.

### 7.2.2 Ohjainpiirilevy

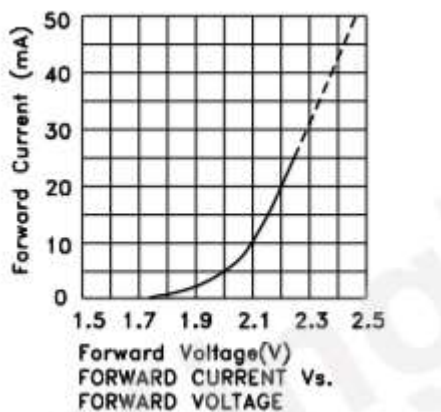
Ohjainpiirilevyssä pyrittiin käyttämään mahdollisimman samanlaista kytkentää CAN-moduulin kanssa, jolloin myös koodi olisi osittain yhteensopivaa levyjen välillä. Piirilevyllä käytettiin CAN-moduulin kanssa samaa regulaattoripiiriä, CAN-piiriä ja ohjelmointiliitintä.

Ohjainpiirilevyllä suunniteltiin neljä potentiometriä, joista voidaan hienosäätää profiilien nollakohtia ja saman profiilin servojen välistä asentoa. Nämä potentiometrit ei ole mitenkään kytketty piirilevyn ulkopuolelle, joten pystyttiin käyttämään suoraan prosessorin 3.3 V jännitettä ja kytkemään suoraan prosessorin analogikanavaan.

Piirilevyllä lisättiin painonappi, josta DRS-järjestelmää on helppo koekäyttää. Nappi kytkettiin maadoittamaan prosessorin kanavaa. Prosessorin sisäinen ylösvetovastus nostaa kanavan 3.3 volttiin, jos nappia ei ole painettu. 100 nF suotokondensaattori estää napin signaalia värähtelemästä nappia painettaessa.



Piirilevyllä on 4 lediä, joista yksi palaa siipeä ohjattaessa auki ja kolme muuta ovat ohjelmointia helpottavia merkkivaloja. Lediksi valittiin 0603 paketissa oleva vihreä Kingbrighth KP-1608SGC. Ledin tyypillinen virta on 20mA ja kynnysjännite 2.2 voltia.



Kuva 26. Pintaliitosledin jännite/virta käyrä (Kingbright 2012).

Lediä kuitenkin haluttiin käyttää virransäästösyistä himmeämpänä 5 mA virralla, jolle datasheet ilmoittaa 2.0 voltin kynnysjännitteen (kuva 26).

Ledin etuvastus saadaan laskettua ohmin lain kaavasta.

$$R_{LED} = \frac{(V_{SUPPLY} - V_{LED})}{I_{LED}} \quad (25)$$

$$R_{LED} = \frac{(3.3 V - 2.0 V)}{5 mA}$$

$$R_{LED} = 260 \Omega$$

Servomoottoreiden ohjaukseen riitti suoraan prosessorin 3.3 voltin jännite. Ohjauslinjaan kytkettiin sarjavastus suojaamaan prosessoria mahdollisilta oikosulkutilanteilta.

Piirilevyn kytkentä liitteessä 3.

## 8 KOODI

ARM-prosessoreiden koodi kirjoitettiin C-kielellä. Kehitysympäristönä toimi Keilin uVision. STMicroelectronics on kehittänyt STM32 -prosessoreille erilaisia kirjastoja, jotka helpottavat koodin kirjoittamista. Vanhempi ”Standard Peripheral Library” tarjoaa valmiita funktioita erilaisiin toimintoihin. STMicroelectronics kehitti STM32Cube pake- tin korvaamaan Standard Peripheral libraryn, tuoden mukanaan paljon käyttäjä ystäväl- lisemmän ympäristön koodin alustukseen ja kirjastoihin. STM32Cube paketti sisältää STM32CubeMX -ohjelmiston, HAL-kirjastot (hardware abstraction layer) ja LL-kirjastot. (The Low-Layer) (STmicroelectroncs 2018b)

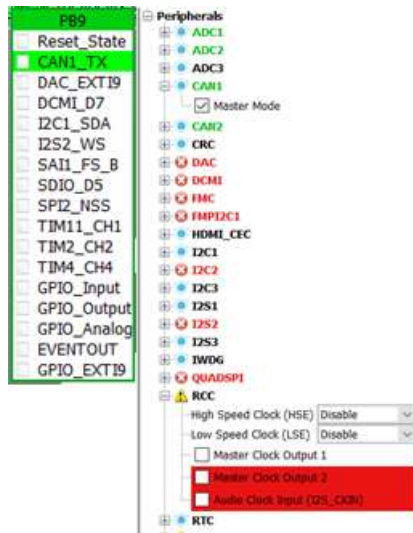
### 8.1 STM32CubeMX

Koodi on mahdollista kirjoittaa alusta asti käsin datasheetin ja reference manualin mu- kaan, mutta STMicrocontrollers on kehittänyt alustukseen ja parametreihin STM32Cu- beMX -ohjelmiston. CubeMX tarjoaa graafisen prosessorin kytkennän, kellotaajuus puun ja parametreja, joista generoidaan pohjakoodi prosessorille, säästäten satoja rivejä C-kie- len koodia.

#### 8.1.1 Graafinen prosessorin kytkentä

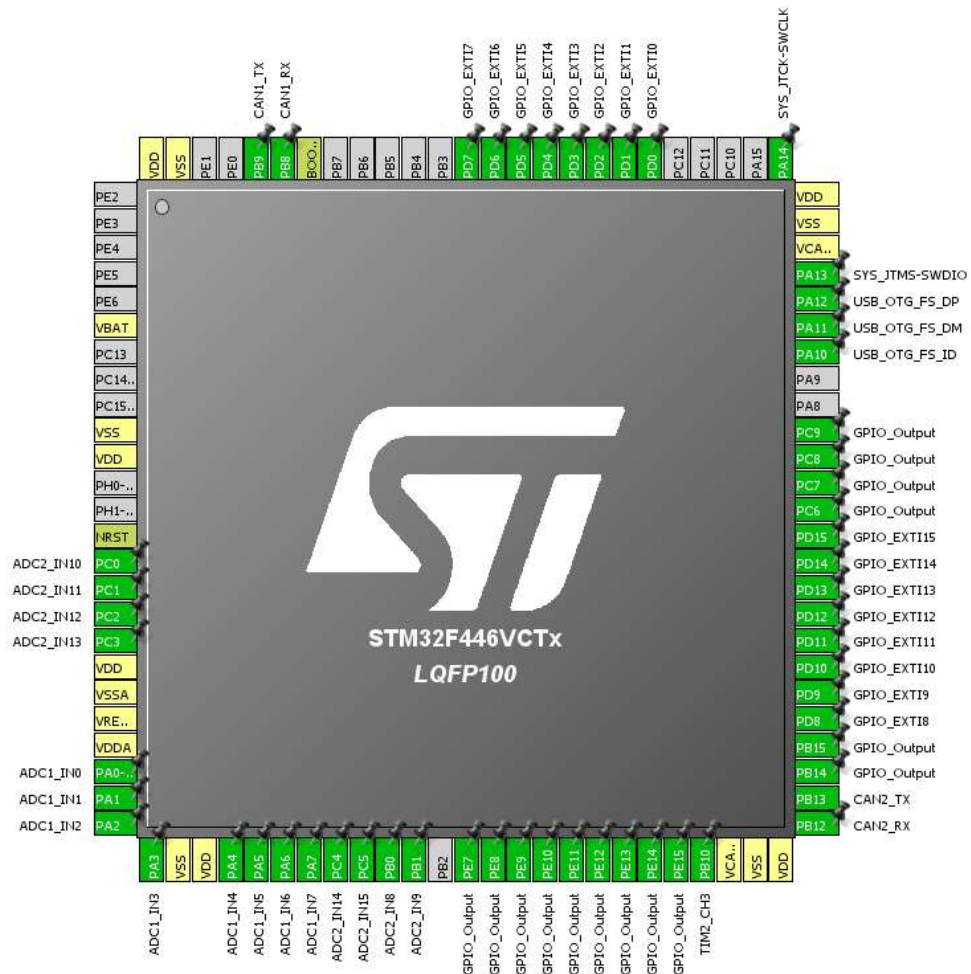
STM32CubeMX projekti alkaa prosessorin valinnasta. CAN-moduulin ja servonoh- jaimeen molempiin käytettiin samaa prosessoria, STM32F446VET6.

Kuvassa 29 prosessorin kytkentä CAN-moduulin piirilevyllä. Kytkentää voi lähteä raken- tamaan kahdella eri tavalla, joko valitsemalla ominaisuuksia vasemman reunan Configu- rations -> Peripherals valikosta (kuva 27) tai valitsemalla prosessorin jalkoihin haluttuja ominaisuuksia.



Kuva 27. Valittavissa olevia ominaisuuksia.

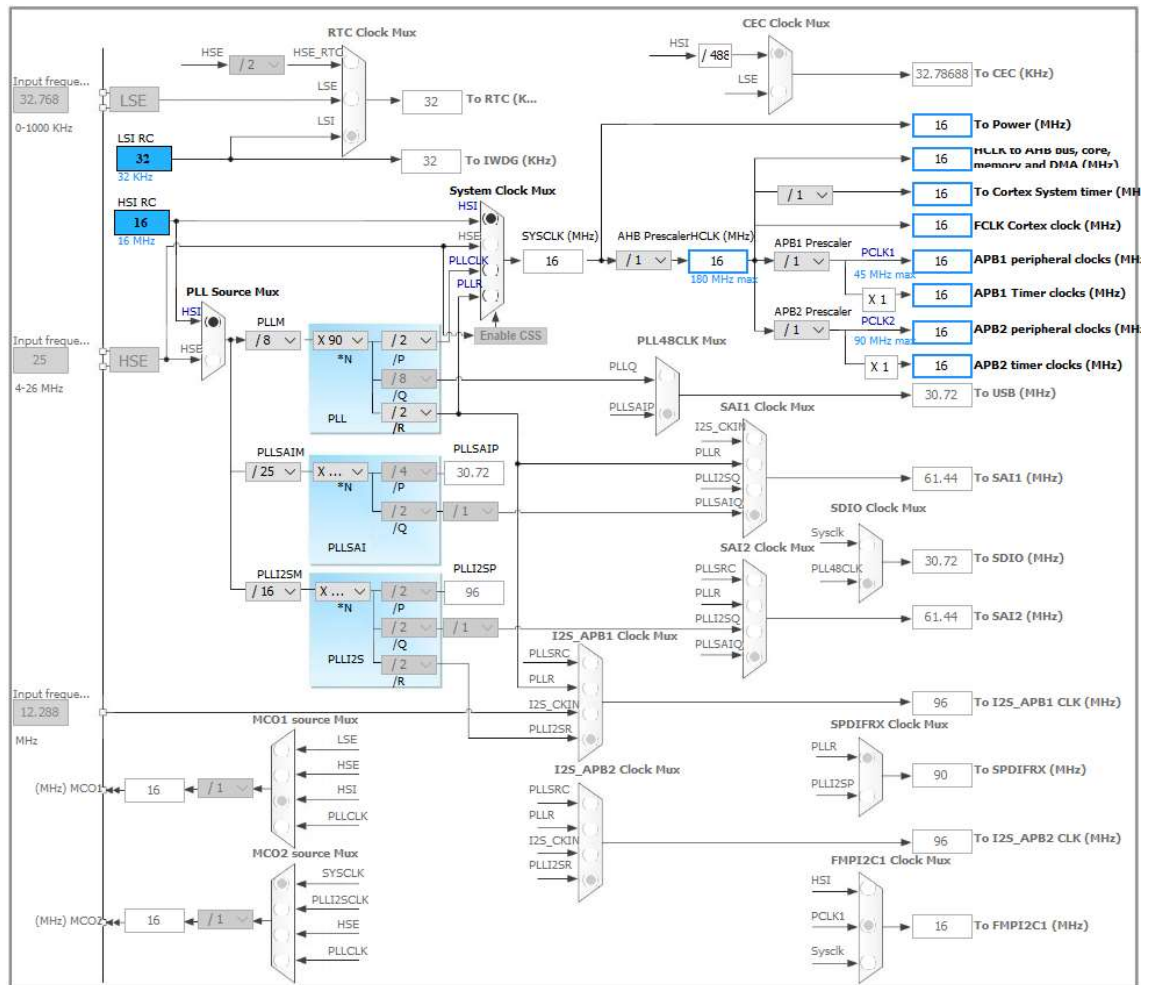
Valitut ominaisuudet poissulkevat aina muita funktioita samasta pinnistä. Esimerkiksi B-portin pinnissä 9 (kuva 27) on valittuna CAN1-väylän transmit-funktio. Samalla tämä pois sulkee mahdollisuudet käyttää esimerkiksi Digital Analog Converterin (ADC) kanavaa 9, I2C1-väylää, joidenkin ajastimien kanavia, analogisia kanavia ja ulkoista keskeytystä yms. Pinneistä poissuljetut ominaisuudet löytyvät usein suoraan tai samankaltaisina muista pinneistä. Jos muut funktiot ovat poissulkeneet kokonaan mahdollisuuden funktion käyttöön, näkyy kyseinen funktio Peripherals-listassa punaisena (kuva 27). Kustomoitavuus onkin ARM-prosessorin suuria etuja 8-bittisiin prosessoreihin verrattuna. Kustomoitavuus antaa vapauksia piirilevy suunnitteluun, kun ominaisuuksia ei ole ennalta määrätty.



Kuva 28. CAN-moduulin prosessorin pinnijärjestys.

### 8.1.2 Kellotaajuuspuu

Kellotaajuuksilla on suuri merkitys kaikkiin aikakriittisiin operaatioihin. Perus kellotaajuus syntyy HSI (High Speed Internal), HSE (High Speed External) kellokriteistä tai PLL-kellosta. Molemmissa piireissä ARM-prosessorin sisäisen HSI-kellon nopeus ja tarkkuus olivat riittäviä. Peruskellotaajuutta voidaan nostaa kellon PLL-kertoimella. Kellotaajuuden nostaminen lisää virrankulutusta ja pienemmällä kellotaajuudella käyvä prosessori toimii usein vakaammin, joten tässä päädyttiin käyttämään HSI-kellon perustaajuutta 16 MHz.



Kuva 29. STM32Cube kellotaajuuden määrittäminen.

Kun prosessorin pääkellotaajuus SYSCLK on valittu, voidaan muiden toimintojen kellotaajuuksia jakaa prescalereiden avulla. Tämä tulee kyseeseen, kun pääkellotaajuus on suurempi kuin tiettyjen ominaisuuksien maksimikellotaajuus, kuten APB1 max 45 MHz tai APB2 max 90 MHz.

STM32 Datasheet ja Reference manual kertovat mitä kelloa mikäkin ominaisuus käyttää.

Työn kannalta tärkeät ominaisuudet eri kellojen takana

AHB1 kello                      GPIOA, GPIOB, GPIOC, GPIOD, GPIOE, DMA1, DMA2

APB1 kello                      CAN1, CAN2, TIM2

APB2 kello                      SYSCFG

(STmicroelectronics 2018a, 180-188)

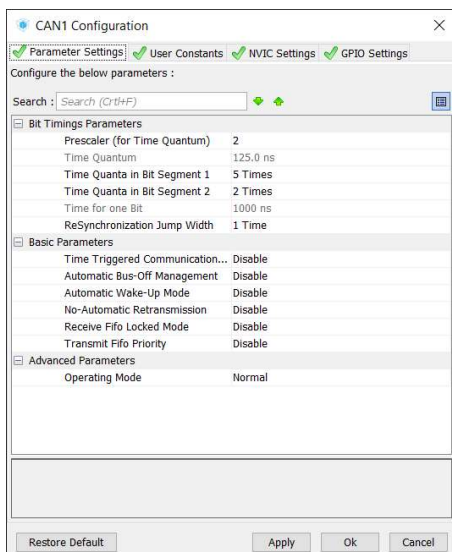
### 8.1.3 Parametrit

Parametrit ja keskeytykset väylille määritellään Configuration -välilehdeltä (kuva 30).

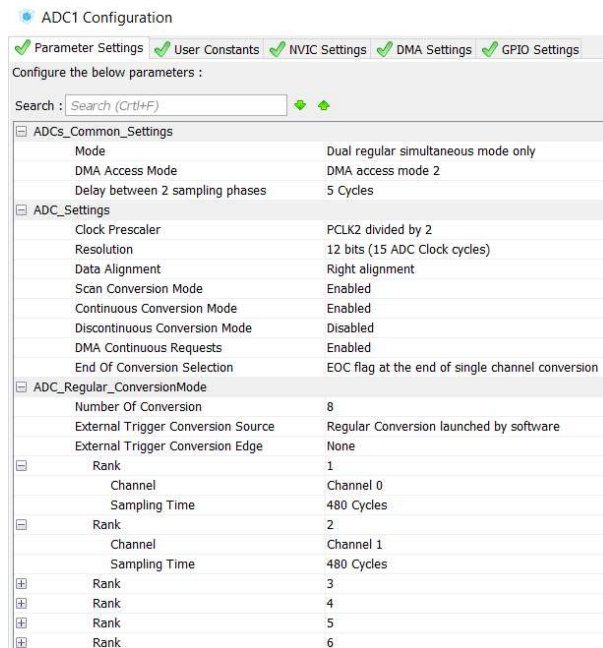


Kuva 30. STM32Cube Configurations -välilehti.

CAN-väylän asetuksista (kuva 31) määritellään aiemmin käsitellyt prescaler ja bit segmentit, jotka määräävät väylän nopeuden.



Kuva 31. CAN-väylän valittavissa olevat parametrit.



Kuva 32. ADC-muuntimen asetukset.

Kuvan 32 ADC-asetuksista määritellään mittausmoodi Dual/Independent ja niiden monet eri variaatiot. Tässä käytetään ADC1 ja ADC2 dual modessa, jolloin yksi DMA hoitaa molempien muuntimien datansiirron. ADC-asetuksissa myös määritellään mitattavat kanavat, niiden järjestys ja sample rate. Sample ratena käytetään suurinta 480 kello syklin muunnosta. Tällä saavutetaan paras mahdollinen muunnostarkkuus. Muuntimet asetetaan continuous tilaan, jolloin mittauksien päätyttyä ne aloittavat uuden kierroksen. ”DMA request” otetaan myös käyttöön, jolloin DMA siirtää datat haluttuun muuttujaan jokaisen mittauksen jälkeen. DMA funktion asetuksista käydään vaihtamassa circular moodi päälle, jolloin DMA ei lopeta datan siirtoa, vaan tekee sen aina uudelleen ja uudelleen. ”Increment memory address” otetaan käyttöön, jolloin data siirtyy aina arrayn seuraavaan muistipaikkaan, eikä kirjoita edellisen arvon päälle.

GPIO-asetuksista määritellään pinnin tila joko input, output tai alternative (CAN, PWM, ADC jne.) ja mahdollisten sisäisten ylösvetovastusten tilat. Täältä määritellään myös digitaalikanavan tilan muutoksesta syntyvät EXTI-keskeytykset.

### 8.1.4 Generoitu koodi

STM32CubeMX luo pelkän C-koodin sijasta valmiin projektin haluamalleen kehitysympäristölle. Project Settings -valikoista voi valita työkansion ja kehitysympäristön (Toolchain / IDE), tässä tapauksessa Keil MDK-ARM V5.

```

136  /* USER CODE BEGIN Init */
137
138  /* USER CODE END Init */
139
140  /* Configure the system clock */
141  SystemClock_Config();
142
143  /* USER CODE BEGIN SysInit */
144  UserConfig();
145
146  /* USER CODE END SysInit */
147
148  /* Initialize all configured peripherals */
149  MX_GPIO_Init();
150  MX_DMA_Init();
151  MX_ADC1_Init();
152  MX_ADC2_Init();
153  MX_CAN1_Init();
154  MX_CAN2_Init();
155  MX_TIM2_Init();
156
157  /* USER CODE BEGIN 2 */
158
159  HAL_ADC_Start_DMA(&hadc1,ADC1_Buffer,8);
160  HAL_ADC_Start_DMA(&hadc2,ADC2_Buffer,8);

```

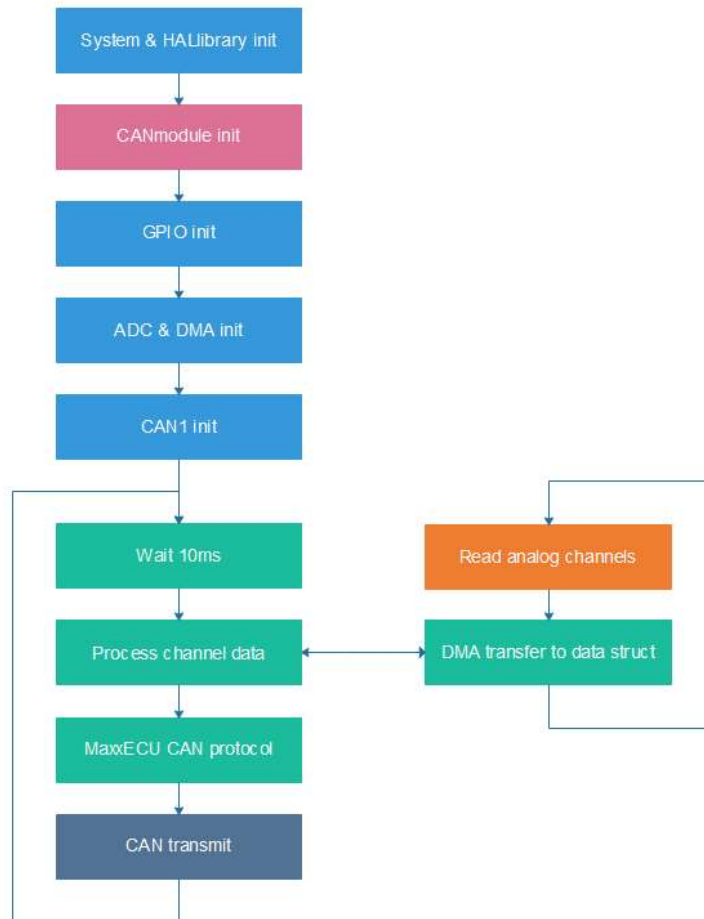
Kuva 33. Pätkä STM32Cubesta generoitua koodia.

Generoidussa koodissa (kuva 33) on merkitty paikat USE CODE BEGIN ja USE CODE END välissä, joihin käyttäjä voi kirjoittaa omaa koodiaan. Tämän ansiosta on aina mahdollista palata takaisin STM32CubeMX ja muuttaa koodin alustusta. Kaikki virheellisiin väleihin kirjoitetut omat koodit katoavat, jos STM32CubeMX -projekti käännetään uudelleen.



## 8.2 CAN-moduuli C-Koodi

STM32CubeMX luoman Keil-projektin pohjalle alettiin kirjoittamaan C-koodia. Kuvan 34 vuokaavion siniset osat kääntyivät automaattisesti aiemmin CubeMX asetetuun parametrein.



Kuva 34. CAN-moduulin koodin vuokaavio.

## 8.2.1 CAN-moduulin muuttujien alustus

CAN-moduulin koodin muuttujat keskittyvät kanavien ympärille ja toimivat kaikissa kanavissa samalla tavalla. Muuttujille luotiin ”struct” rakenne, joka alustettiin jokaisen kanavan käyttöön (kuva 35). Data struct sisältää kanavan tyyppin, analogikanavan väliaikais-tiedot ja lasketut arvot ja digitaalikanavan väliaikaistiedot ja lasketut arvot.

```

struct Ch {
    uint16_t    SensorType;
    struct AnalogData {
        uint32_t    Total;
        uint16_t    Count;
        uint16_t    Value;
        uint16_t    LastValue;
        uint16_t    PeakValue;
        int         DigitalStatusADC;
        uint32_t    SampleRate;
        uint32_t    Millivolts;
        double      Calfactor;
        uint32_t    MaxxECU;
    }Analog;

    struct DigitalData {
        uint32_t    FallingEdgeTotal;
        uint32_t    RisingEdgeTotal;
        uint16_t    FallingEdgeCount;
        uint16_t    RisingEdgeCount;
        double      PulseDutyTemp;
        uint32_t    PulseLowTime;
        uint32_t    PulseHighTime;
        uint32_t    PeriodTime;
        uint32_t    Frequency;
        uint16_t    PulseDuty;
        uint8_t     DigitalStatus;
    }Digital;
};

```

Kuva 35. Kanavan muuttujat.

Jos halutaan viitata esimerkiksi kolmannen kanavan analogiseen millivoltti arvoon, muuttuja on 32 bittinen unsigned int, jonka nimi on Ch3.Analog.Millivolts.

Kun analogikanavissa on jännitteenjakovastuksia, ylös- ja alaspäin suuntaisia vastuksia ja muita passiivisia komponentteja, niiden toleranssit vaikuttavat mittaustulokseen. Calfactor -muuttujaan alustetaan korjauskerroin, jonka tarkoituksena on mittauskanavasta johtuvat epätarkkuu-

det. Calfactor-arvo on oletuksena kaikilla kanavilla 1.000. Jos kanavaan syötetään tarkalla jännitelähteellä 5.00 voltia ja moduuli lukee kanavan arvoksi esimerkiksi 5.08 voltia, voidaan uusi korjauskerroin laskea kaavan 26 avulla.

$$\begin{aligned} \text{Calfactor}_{\text{UUSI}} &= \text{Calfactor}_{\text{VANHA}} \cdot \frac{\text{Todellinen jännite}}{\text{Mitattu jännite}} & (26) \\ \text{Calfactor}_{\text{UUSI}} &= 1.000 \cdot \frac{5.00}{5.08} = 0.984 \end{aligned}$$

Muuttamalla korjauskerroin 0.984, koodissa tehdään -1.6 % korjaus kaikille mitatuille arvoille.

Samantyylinen ”data struct” luotiin lähteviä CAN-viestejä varten.

Alustuksen yhteydessä määriteltiin myös kanavien jännitteenjakovastuksien koot, CAN-lähetystaajuus ja nopeus, mitkä valmiiksi määritetyt CAN-viestit ovat käytössä ja mahdolliset moduulin sisällä suoritettavat laskennat kanavan mittaustiedosta.

### 8.2.2 Analogikanavien luku

Analogikanavien lukuun käytetään prosessorin DMA (Direct Memory Access) toimintoa. Piirilevyn kanavat on jaoteltu kahdelle eri analogimuuntimelle, kanavat 1-8 ADC1 ja kanavat 9-16 ADC2. Analogimuuntimet lukevat kanavan arvoja saman aikaisesti, ADC1 ollessa master ja ADC2 slave modessa. Jokaisen suoritettun ADC-muunnoksen jälkeen ADC1 lähettää DMA:lle pyynnön ja DMA siirtää mitatun arvon talteen väliaikaiseen array muotoiseen taulukkoon. Ensimmäisessä DMA-siirrossa menevät kanavien 1 ja 9 arvot arrayn ensimmäiseen datakenttään. Siirron jälkeen arrayn datakentän arvoa kasvatetaan (increment) ja seuraavien kanavien arvot siirretään seuraavaan datakenttään.

Kun DMA on siirtänyt arvot kahdeksannen kerran talteen, kaikki kanavat on kertaalleen luettu. Viimeisen siirron seurauksena prosessorille tulee keskeytys ja väliaikaisen arrayn tiedot käsitellään. Keskeytyksen aikana muu koodin suorittaminen pysähtyy, joten keskeytyksessä kannattaa suorittaa vain pakolliset laskennat. Jokaisen kanavan mitattu arvo siirretään LastValue-muuttujaan, sen arvo lisätään Total-muuttujaan ja Count-muuttujaa

kasvatetaan yhdellä. Näiden toimintojen jälkeen analogimuunnin jatkaa uutta kierrosta taustalla.

### 8.2.3 Datan käsittely ja lähetys väylään

Data lähetetään väylään 100 Hz taajuudella eli 10 ms välein. Datan käsittely tapahtuu myös samalla taajuudella, koska tiheämmällä käsittelytaajuudella ei saavuteta mitään.

Kanavien Total- ja Count-muuttujista lasketaan ensin mittaustaajuus ja keskiarvoistettu analogimuunnoksen arvo. Noin 50 mittauksen keskiarvolla saadaan analogimuunnoksen arvosta suodatettua häiriöitä pois ja lähetettävästä signaalista mahdollisimman tarkka.

Tämän jälkeen analogimuunnoksen arvo muutetaan millivoltti tiedoksi ja ylösvetovastuksella varustetussa lämpötilakanavassa myös resistanssitiedoksi. Muunnoksessa käytetään alustuksessa määriteltyjä vastusarvoja ja calfactory korjauskerrointa.

Tämän jälkeen arvosta voidaan laskea esimerkiksi paineanturin painetietoa, ratinasento kulmana tai muuta vastaavaa, jolle on koodiin kirjoitettu valmis kaava. MaxxECUn analogiakanavan sisään luku väylästä tapahtuu ADC muunnoksen 12 bittisenä arvona, jossa 5000 mV vastaa ADC arvoa 4095. Arvo lasketaan muuttujaan kaava 27 mukaan.

$$Chx.Analog.MaxxECU = \frac{Chx.Analog.Millivolts}{5000/4095} \quad (27)$$

Laskennan jälkeen kanavien arvot siirretään CAN-lähetys structiin. Tässä vaiheessa määräytyy, mikä kanava näkyy moottorin ohjaimella missäkin analogia kanavassa. Esimerkiksi Ch3.Analog.MaxxECU arvo siirretään CAN-lähetys structin analogiakanavaan 1, jolloin se näkyy MaxxECUn input moduulin analogiakanavassa 1.

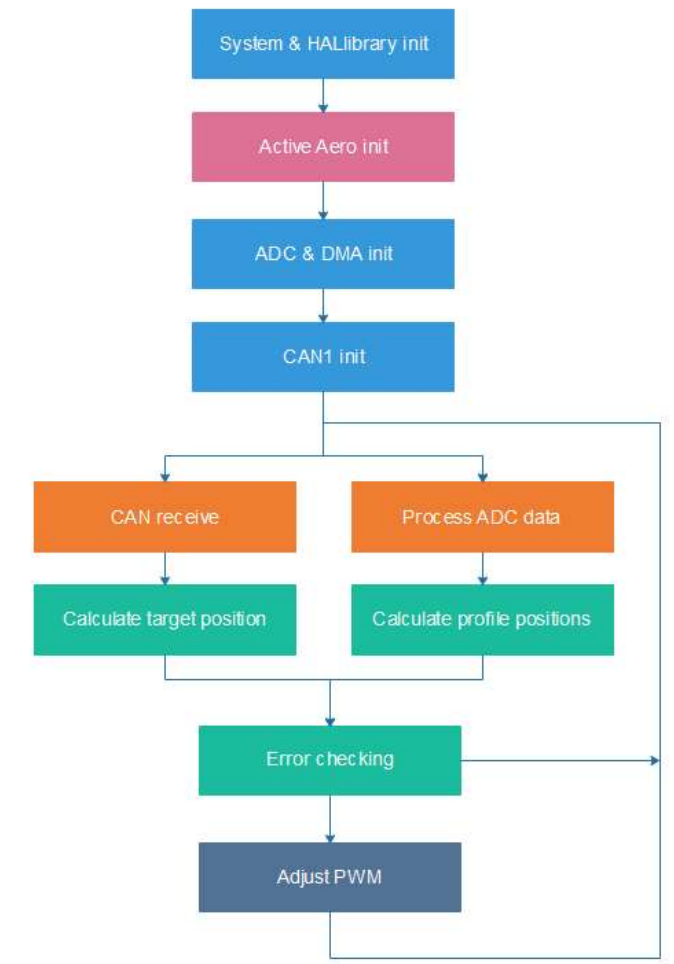
CAN-viestin lähetykseen käytetään HALlibraryn valmista HAL\_CAN\_Transmit\_IT() funktiota. Funktiossa on valmiiksi toteutettu CAN-liikenteen ”mailbox” tyyppinen lähetys. Funktio siirtää lähetettävän viestin ”mailboxiin”. Kun väylä on vapaa ja viestiä lähetetään, prosessori katsoo ”mailboxeista” suurimmalla prioriteetillä olevan viestin (pienin

ID) lähetettäväksi. Näin ohjelman ei tarvitse odottaa väylän vapautumista ja ”mailboxit” lähettävät viestit väylälle niiden sinne mahtuessa.

Koodissa määritellään viesteille vain ID, DLC eli datan pituus ja itse datat.

### 8.3 Servo-ohjain C-Koodi

Servo-ohjaimen prosessorin ollessa sama kuin CAN-moduulissa, koodi on hyvin pitkälti yhteensopivaa. Analogi ja digitaalitietojen luku on toteutettu täysin samalla tavalla kuin CAN-moduulissakin. Servo-ohjaimen koodista on vain poistettu ylimääräiset kanavat ja CAN-väylään lähetys (kuva 36).



Kuva 36. Servo-ohjaimen koodin vuokaavio

### 8.3.1 CAN-viestin vastaanotto

CAN-viesteistä on helppo filtteröidä ja maskata vain halutut viestit perille. Mask-rekisterillä ilmoitetaan mitkä ID:n bitit pitää täsmätä filttiin. Extended CAN ID on 29 bittiä pitkä ja rekisterit ovat 32-bittisiä, joten kolme eniten merkitsevää bittiä rekisterissä ovat merkityksettömiä.

Esimerkiksi, jos halutaan sallia ID:t 100-103

- Kaikki muut paitsi kaksi vähiten merkitsevää bittiä tulee täsmätä filteristä
  - 0001 1111 1111 1111 1111 1111 1111 1100
  - Hex 1FFF FFFC
- Filteririn
  - 0000 0000 0000 0000 0000 0000 0110 0100
  - Hex 0000 0064
- Filterin ja Maskin läpi pääsevät ID:t
  - Hex 64-67

DRS-asentotieto lähetetään ID:llä desimaali 50 (HEX 0x32) ja halutaan sallia vain kyseisen viestin pääsy. Maskiksi asetetaan 0x1FFFFFFF, jolloin kaikki bitit tarkistetaan ja filterin ollessa vastaanotettava ID 0x00000032, vain nämä viestit pääsevät läpi.

Viestit luetaan HALlibraryn funktiolla HAL\_CAN\_Receive().

### 8.3.2 Laskenta

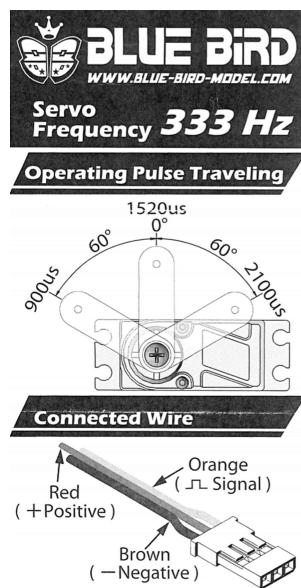
Analogikanavilla mitatut piirilevyn kaksi potentiometriä säättävät siiven suljetun asennon paikkaa eli profiilin nollakohtaa ja toiset kaksi potentiometrit siiven servojen keskinäistä asentoa. Nollakohtaa säättävät potentiometrit vaikuttavat suotaan PWM-signaalin offsettiin. Saman profiilin servot pyörivät fyysisesti erisuuntiin, joten toisen servon PWM-signaali on invertoitu. Invertoidun servon ”offsetillä” muutetaan servojen keskinäistä asentoa.

Jokaisen servon takaisinkytkentä on johdotettu ohjainpiirille. Takaisinkytkennästä seurataan servojen keskinäistä asentoa. Kun kiinni ja auki asennot ovat kalibroitu koodiin oikein, voidaan servojen asentoa seurata prosenteissa 0-100 %. Jos saman profiilin servojen arvot eroavat yli 5 %, profiili ei ole enää suorassa ja turvatoimena molemmilta servoilta katkaistaan ohjaus.

Kun kaikki arvot ovat kunnossa, ajastimille lähetetään uudet PWM-ohjausarvot.

### 8.3.3 Servon ohjaus

Servomoottori toimii 333 Hz taajuudella ja asento määräytyy pulssinleveydestä. 0° 1500 µs, -60° 900 µs, +60° 2100 µs (kuva 37). Jos pulssinleveys on jotain muuta, servo on pois käytöstä ja pyörii vapaana.



Kuva 37. Servon ohjaussignaali.

Pulssin ”duty” kertoo, kuinka monta prosenttia ajasta pulssi on ylhäällä. Duty saadaan laskettua kaavasta 28.

$$D = \frac{PW}{T} \cdot 100\% \quad (28)$$

Missä

$D = \text{Duty cycle}$

$PW = \text{Pulse Width (Pulssinleveys)}$

$T = \text{Total period (Kokonais jaksonaika)}$

$$D_{-60^\circ} = \frac{900\mu\text{s}}{\frac{1}{333\text{Hz}}} \cdot 100\%$$

$$D_{-60^\circ} = 29,97\% \approx 30\%$$

$$D_{0^\circ} = 49,95\% \approx 50\%$$

$$D_{60^\circ} = 69,93\% \approx 70\%$$

### 8.3.4 Prosessorin PWM-lähtö

Prosessorissa ajastimet toimivat APB1 kellon mukaan eli tässä tapauksessa 16 MHz taajuudella. TIM4 parametreihin määriteltiin kellotaajuuden jakajaksi 48, jolloin TIM4 taajuudeksi saadaan 333 kHz kaavalla 29.

$$TIM4_{CLK} = \frac{16\,000\,000\text{Hz}}{48} = 333\,333\text{Hz} \quad (29)$$

Ulostulevan signaalin taajuuteen vaikuttaa TIM4 periodin pituus. Kun kellotaajuus jaetaan 1000 periodiin, saadaan ulostulevan signaalin taajuudeksi 333 Hz.

$$PWM_{CLK} = \frac{333\,333\text{Hz}}{1000} \approx 333\text{Hz} \quad (30)$$

TIM4 on neljä kanavaa. Kanavien taajuus pysyy aina samana, mutta ”dutyä” voidaan muuttaa yksilöllisesti TIM4->CCRx rekistereiden avulla. CCR-rekisteri määrittelee kuinka monta TIM4 kellosykliä lähtö on ylhäällä.

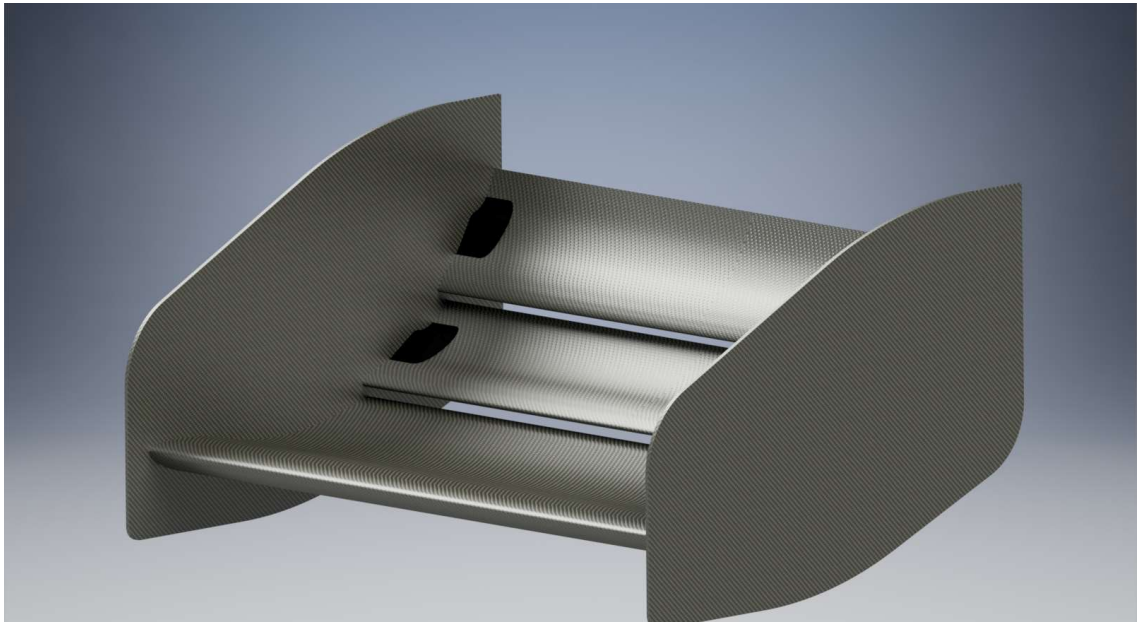
Esimerkiksi

- Kun CCR on 500 ja period 1000
  - PWM duty = 50%
- Kun CCR on 250 ja period 1000
  - PWM duty = 25%



## 9 KOKOONPANO

Kuvassa 38 takasiiven 3D-malli, josta näkyvät myös profiilin sisällä olevat moottorit.



Kuva 38. FS017 Takasiiven 3D-malli.

### 9.1 Ohjaussignaali

Siipi aukeaa, kun

- Kaasuläpän asento on yli 95 %
- Ratin asento on suorassa tai maksimissaan  $\pm 20^\circ$
- Jarrua ei paineta, jarrupaine alle 5 bar
- 250 ms viive

Ohjauksesta haluttiin helposti muokattava, joten toteutus tehtiin MaxxECUn moottorinohjaimen ”Internal output” funktioon. Moottorinohjain on hyvä graafinen käyttöliittymä eikä tarvittavia muutoksia varten tarvitse koskea piirilevyjen koodeihin. Ohjaus toteutettiin kuvan 39 mukaisella neljän muuttujan AND-piirillä ja turn on delayllä.

**Internal output 1**

Function name: DRS

Description: ?

**Condition A**

Variable: ? Throttle position ... > (greater than) 95 % [Inactive]

**Condition B**

Variable: ? User AIN5 (Steering p ... < (less than) 20 Deg [Inactive]

**Condition C**

Variable: ? User AIN5 (Steering p ... > (greater than) -20 Deg [Inactive]

**Condition D**

Variable: ? User AIN6 (BrakePr) ... < (less than) 5 Bar [Inactive]

**Output configuration**

Output function: ? A and B and C and D

Turn on delay: ? 0.250 s

Turn off delay: ? 0.0 s

Output: ? Normal

Output flash: ? No

Kuva 39. Ohjauksen ehdot MaxxECUn internal output funktiossa.

Ehtojen täytyessä Internal output 1 arvo on yhden bitin mittainen boolean true/false. Arvo lähetetään kustomoitavana CAN-viestinä CAN-väylään desimaalilukuisella ID:llä 50.

**CAN output 1**

**CAN output Value 1**

Enable: ? Enable

CAN Bus: ? CAN 1

CAN Message ID Type: ? Standard

CAN Message ID: ? 50 (decimal)

Message rate: ? 100Hz

Endian: ? Little endian

Values in this package: ? 1

**Data value 1**

Type: ? Variable - signed 8 bit

Variable: ? Internal output 1 Activ ...

Offset: ? 0 Output = (Indata\*Multiplier/Divider) + Offset

Multiplier: ? 1

Divider: ? 1

Kuva 40. MaxxECUn CAN-lähetys funktio.

## 9.2 Servomoottoreiden kiinnitys

Servomoottorit eivät sopineet kokonaan siipiprofiilin sisään, joten moottorille rakennettiin erillinen kotelo. Kotelo 3D-tulostettiin PLA-muovista. Muovikoteloon upotettiin neljä M4 kierre inserttiä, jotka asennettiin noin 200 asteisina tiukkaan sovitteeseen. Lämmin insertti sulattaa vähän muovia ympäriltä ja jäähtyessään jämähtää kiinni muoviin. Moottori kiinnittyy muovikoteloon neljällä M4x30 pultilla.

3D-tulostus on haurasta tulostuskerrosten välistä, joten kokonaisuutta vahvistettiin lami-noimalla hiilikuitukerros 3D-tulostuksen päälle. Siipiprofiiliin leikattiin sopivan kokoi-nen reikä ja kokonaisuus liimattiin kasaan.

## 9.3 Ohjainlaite

Ohjainlaite sijaitsee siiven päätylevyssä (kuva 41), jolloin johdotus auton muuhun järjes-telmään on helpompaa pelkän käyttöjännitteen ja väylän myötä. Servomoottorien johdot tuotiin hiilikuidusta läpi, jättäen pieni lenkki siiven kääntymistä varten. Toisen puolen moottorin johdot tuotiin siipiprofiilin sisällä, jolloin ei tarvinnut kuin yhden läpiviennin.



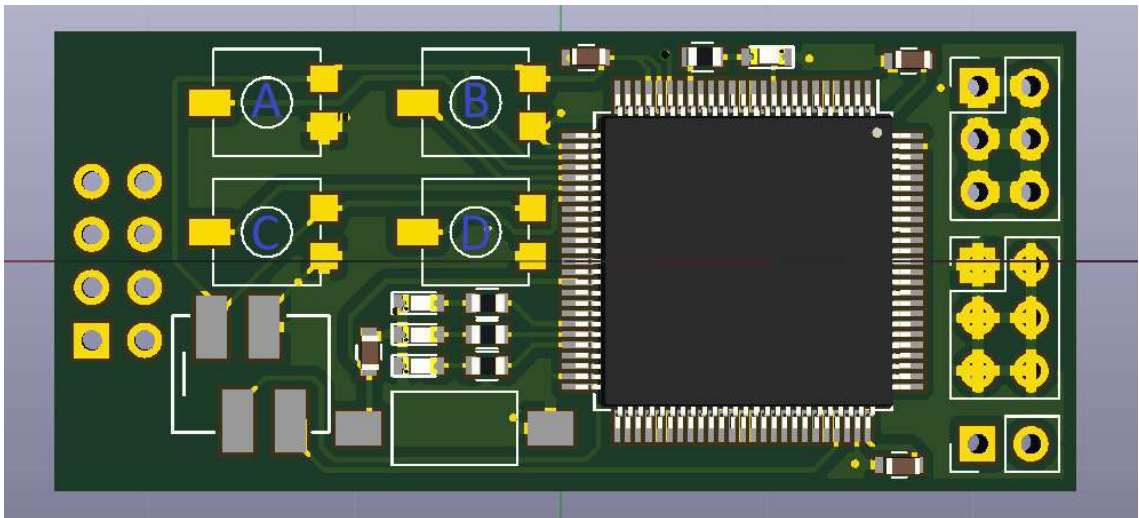
Kuva 41. Servo-ohjainlaite.

Ohjainlaitteen 3D-tulostetussa kotelossa on neljä Deutsch IMC-liitintä servoille. Liittimet ovat värikoodattuja, mutta ne on mahdollista kytkeä väärin.

## 9.4 Profiilien kiinnittäminen

Seuraavat asiat ovat tärkeitä siipiprofiileita kiinnittäessä päätylevyihin. Huomioi kuvan 42 potentiometrien paikat.

1. Kytke sähköt moottoreihin.
2. Säädä kaikki ohjaimen potentiometrit säätöalueen keskelle.
3. Aseta molempien profiilien vasemmat reunat kiinni päätylevyihin ja ruuvit paikalleen. Älä vielä kiristä tai inserttiosa voi vääntyä irti päätylevystä.
4. Aseta oikeat reunat kiinni, jos ne menevät booreille. Jos ne eivät mene paikalleen, katso a ja b kohdat.
  - a. Jos alempi profiili ei mene paikalleen, säädä potentiometrillä B.
  - b. Jos ylempi profiili ei mene paikalleen, säädä potentiometrillä D.
5. Kiristä ruuvit kireälle.
6. Jos siiven servoista kuuluu normaalia kovempi ääni, säädä 4. kohdan potentiometrejä niin, että ääni hiljenee ja servot eivät väännä toisia vasten.
7. Säädä potentiometreillä A ja C profiilien asennot kohdalleen.



Kuva 42. Servo-ohjain, potentiometrit merkittynä kirjaimin.

## 9.5 Muita kokoonpanoon liittyviä asioita

Mekaanista stopparia ei asennettu paikalleen. Alempi profiili kiinni ollessaan tukeutuu isoon pohjaprofiiliin ja ylempi profiili tukeutuu alempaan kääntyvään profiiliin. Näin siiven ollessa kiinni, moottorien ei tarvitse pitää siipiprofiilia paikallaan. Mekaaniset stopparit on mahdollista lisätä päätylevyihin, jos sen näkee aiheelliseksi.

Johtoreittejä varten asennettiin liimattavia nippusidekiinnikkeitä siiven päätylevyn sisäpuolelle ja profiilin pintaan. Ne näkyvät aiemmassa kuvassa 41.



Kuva 43. Takasiipi kiinni ja auki.

## 10 POHDINTA

Servomoottorin valinnan jälkeen BlueBirdiltä on tullut uusi BLS-mallisarja. Mallisarjan tehokkain BlueBird BLS-H50B tarjoaa 60 % suuremman vääntömomentin samankokoisella servolla, kun käytössä ollut BMS-35A. Uusia siipiprofiileita rakentaessa moottorin vaihtoa kannattaa pohtia.

Siiven servojen toiminnan testaaminen myöhästyi kesälle, yhtenä syynä myöhään valmistunut siipipaketti. Kuormalla siiven ottama maksimivirta nousi oletettua korkeammalle ja jännitteen syötön kanssa oli pieni epävarmuus. Testikäytössä käytettiin 7.4 voltin LiPo akkua, jossa virta on reilusti ylimitoitettu. Jännitteensyöttö vaatii joko uuden isomman regulaattoripiirin tai vaihtoehtoisesti ison kondensaattorin, jonka varaus riittää siiven liikkeelle lähdön virtapiikkiin.

Servojen lepovirta jäi alle 100 mA, mutta moottoreiden jatkuva paikan hakeminen pitää odotettua suurempaa ääntä. Profiilin ollessa kiinni ja paikallaan, servot kytkettiin pois käytöstä, jolloin myös ääni hiljeni.

Kuten laskennallisesta hyödystä kävi ilmi, noin puolet siiven aukeamisista ovat alle puolen sekunnin luokkaa. Lyhyellä ajalla ei saavuteta mitään hyötyä. Ohjausta taas on vaikea muuttaa ”ennustamaan” kohdat, joissa siipeä ei avata. Ohjaus voisi toimia paremmin kuskin ohjatessa siiven toimintaa esimerkiksi ratissa olevan napin avulla.

## LÄHTEET

Blue Bird Technology. n.d. BMS-35A. Viitattu 15.7.2018  
[https://www.blue-bird-model.com/products\\_detail/48.htm](https://www.blue-bird-model.com/products_detail/48.htm)

Bourns. 2017. SF-0603F SinglFuse. Viitattu 15.7.2018.  
<https://www.bourns.com/pdfs/SF0603F.pdf>

Bosch. 1991. CAN Specification. Viitattu 15.7.2018.  
<http://esd.cs.ucr.edu/webres/can20.pdf>

Bosch. 2017. Temperature Sensor NTC M12. Viitattu 15.7.2018. [http://www.bosch-motorsport.com/media/catalog\\_resources/Temperature\\_Sensor\\_NTC\\_M12\\_Datasheet\\_51\\_en\\_2782569739pdf.pdf](http://www.bosch-motorsport.com/media/catalog_resources/Temperature_Sensor_NTC_M12_Datasheet_51_en_2782569739pdf.pdf)

Case, D. 1996. Formula SAE - Competition History 1981 - 1996. Viitattu 28.10.2017.  
<http://papers.sae.org/962509/>

CompuPhase. n.d. Reflow soldering profiles. Viitattu 15.7.2018  
<https://www.compuphase.com/electronics/reflowsolderprofiles.htm>

Diodes Incorporated. 2016. SDM2100S1F. Viitattu 15.7.2018.  
[http://www.farnell.com/datasheets/2263993.pdf?\\_ga=2.97742555.530782886.1519934563-1844524624.1513943921](http://www.farnell.com/datasheets/2263993.pdf?_ga=2.97742555.530782886.1519934563-1844524624.1513943921)

Ed Ramsden. 2000. Hall vs. VR: Which speed sensor is wrong for you? Viitattu 15.7.2018 [https://www.electronicproducts.com/Electromechanical\\_Components/Hall\\_vs\\_VR\\_Which\\_speed\\_sensor\\_is\\_wrong\\_for\\_you.aspx](https://www.electronicproducts.com/Electromechanical_Components/Hall_vs_VR_Which_speed_sensor_is_wrong_for_you.aspx)

Electronics Tutorials. n.d. Operational Amplifiers Summary. Viitattu 20.7.2018  
[https://www.electronics-tutorials.ws/opamp/opamp\\_8.html](https://www.electronics-tutorials.ws/opamp/opamp_8.html)

Eurocircuits. n.d. PCB Design Guidelines. Viitattu 29.10.2017.  
<https://www.eurocircuits.com/pcb-design-guidelines/>

Formula One. 2018. Drag Redution System. Viitattu 26.2.2018  
[https://www.formula1.com/en/championship/inside-f1/rules-regs/Drag\\_Reduction\\_System.html](https://www.formula1.com/en/championship/inside-f1/rules-regs/Drag_Reduction_System.html)

Institution of Mechanical Engineers. 2016. History of Formula Student. Viitattu 26.2.2018  
<https://www.imeche.org/events/formula-student/about-formula-student/history-of-formula-student>

ISO 11898-2. 1999. Road vehicles – controller area network (CAN). Viitattu 15.7.2018.  
[http://d1.amobbs.com/bbs\\_upload782111/files\\_23/ourdev\\_515258.pdf](http://d1.amobbs.com/bbs_upload782111/files_23/ourdev_515258.pdf)

King B. & Nowakwski R. 2006. Choosing the optimum switching frequency of your DC/DC converter. Viitattu 20.7.2018 [https://www.eetimes.com/document.asp?doc\\_id=1272335](https://www.eetimes.com/document.asp?doc_id=1272335)

Kingbright. 2012. 1.6mm x 0.8mm SMD Chip LED lamp. Viitattu 20.7.2018. [http://www.farnell.com/datasheets/2045827.pdf?\\_ga=2.212570397.885532097.1534974270-565012104.1530440524](http://www.farnell.com/datasheets/2045827.pdf?_ga=2.212570397.885532097.1534974270-565012104.1530440524)

Renesas. n.d. Linear vs. Switching Regulators. Viitattu 20.7.2018. <https://www.intersil.com/en/products/power-management/linear-vs-switching-regulators.html>

ROHM Semiconductor. 2016. RB706F-40FH. Viitattu 20.7.2018. <https://www.rohm.co.kr/datasheet/RB706F-40FH/rb706f-40fht106-e>

SAE International. n.d. History of Formula SAE. Viitattu 28.10.2017 <https://www.fsaeonline.com/page.aspx?pageid=c4c5195a-60c0-46aa-acbf-2958ef545b72>

SAE International. 2017. About Formula SAE Series. Viitattu 28.10.2017 <http://students.sae.org/cds/formulaseries/about/>

STmicroelectronics. 2015. SM4TY. Viitattu 15.7.2018. <https://www.st.com/resource/en/datasheet/sm4ty.pdf>

STmicroelectronics. 2018a. Reference manual. Viitattu 15.7.2018. [https://www.st.com/resource/en/reference\\_manual/DM00135183.pdf](https://www.st.com/resource/en/reference_manual/DM00135183.pdf)

STmicroelectronics. 2018b. STM32Cube MCU Packages. Viitattu 15.7.2018. <http://www.st.com/en/embedded-software/stm32cube-mcu-packages.html?querycriteria=productId=LN1897>

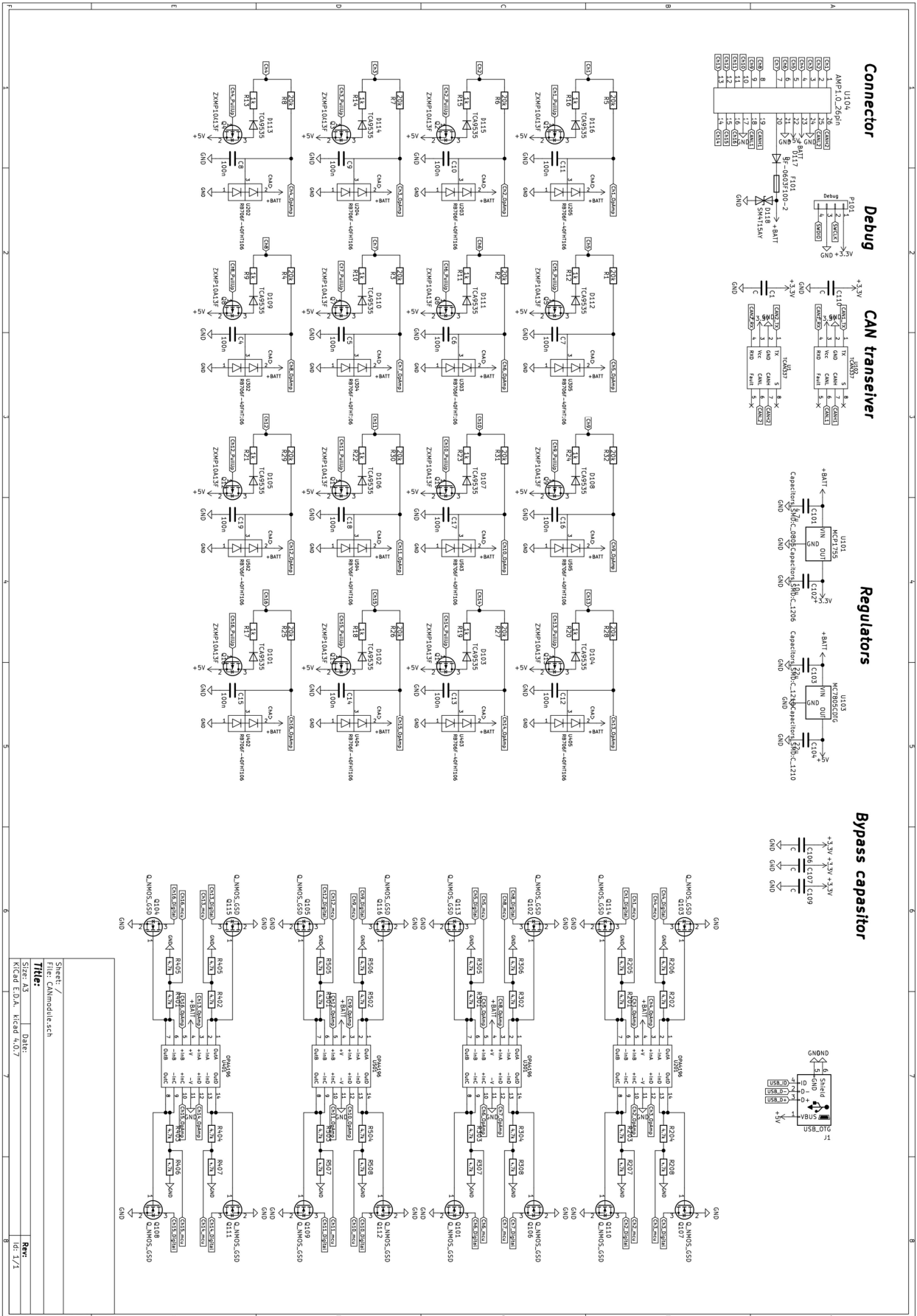
TEconnectivity. 2010. AMP SUPERSEAL Connectors 1.0 mm System. Viitattu 15.7.2018 [http://www.powerandsignal.com/images/pdfs/TEAMPSUPERSEAL1\\_0.pdf](http://www.powerandsignal.com/images/pdfs/TEAMPSUPERSEAL1_0.pdf)

Texas Instruments. 2008. Controller Area Network Physical Layer Requirements. Viitattu 15.7.2018 <http://www.ti.com/lit/an/slla270/slla270.pdf>

Texas Instruments. 2013. TPS54540 Step Down DC-DC Converter. Viitattu 20.7.2018. <http://www.ti.com/lit/ds/symlink/tps54540.pdf>

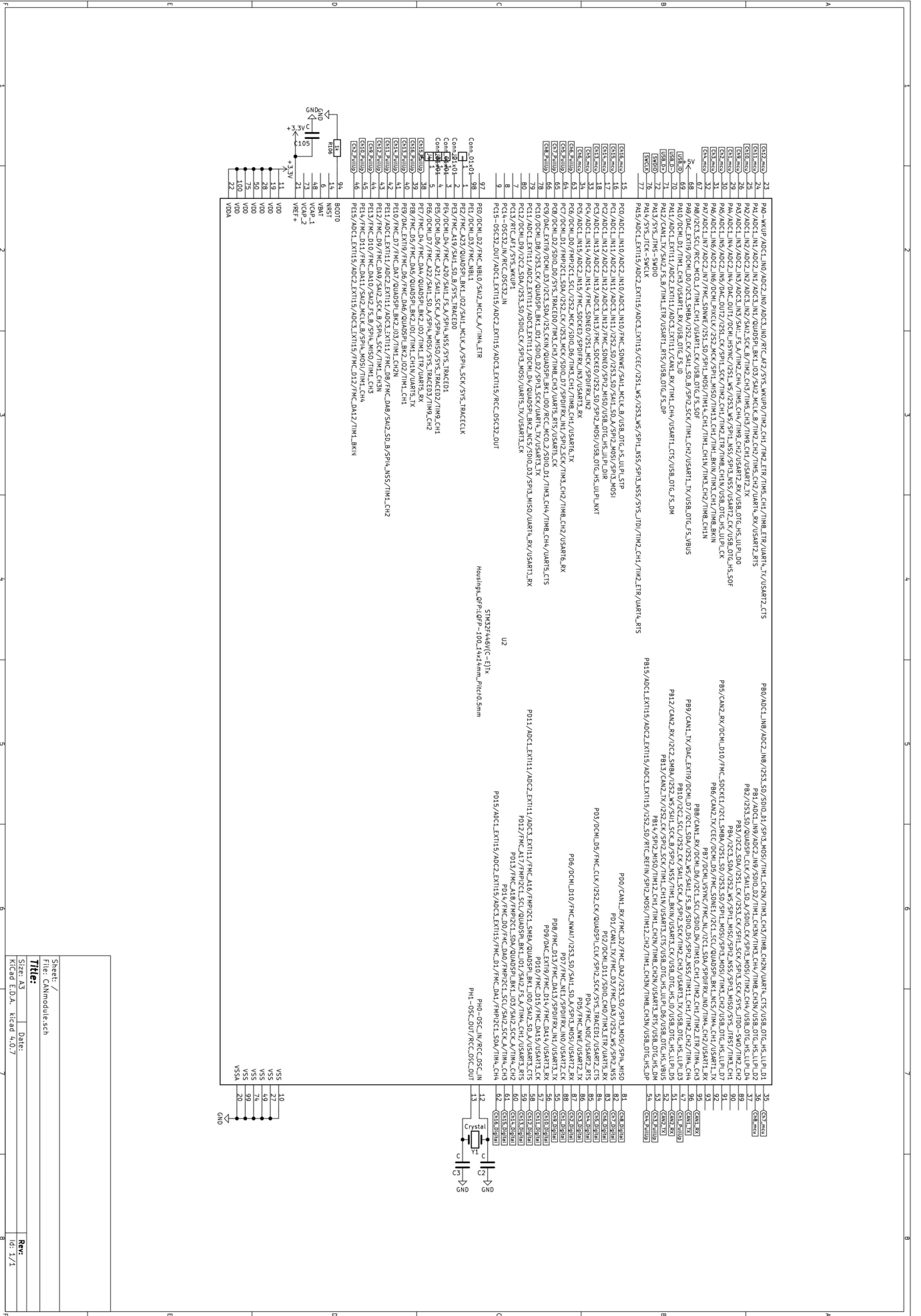


Liite 1. CAN-moduuli piirilevy kytkentä.



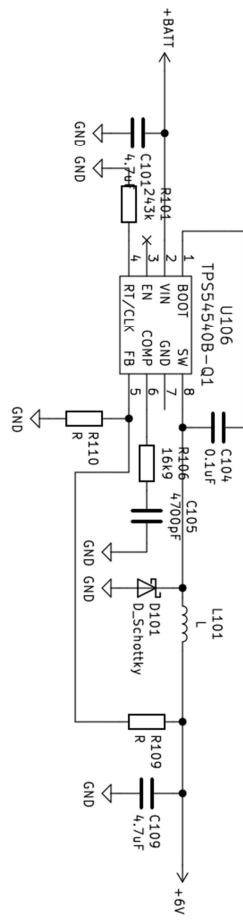
Sheet /  
 File: Ammoduc5th  
 Title: **TH1e**  
 Date: 2013  
 Rev: 1/1  
 K:\Cad E.D.A. KiCad 4.0.7

Liite 2. CAN-moduuli piirilevy kytkentä 2.



Sheet: /  
 File: CANmodule.sch  
**Title:**  
 Size: A3  
 Kiead E.D.A. Kiead 4.0.7  
 Date:  
 Row:  
 of: 1/1

## Liite 3. Servon jännitteensyötön piirilevy.



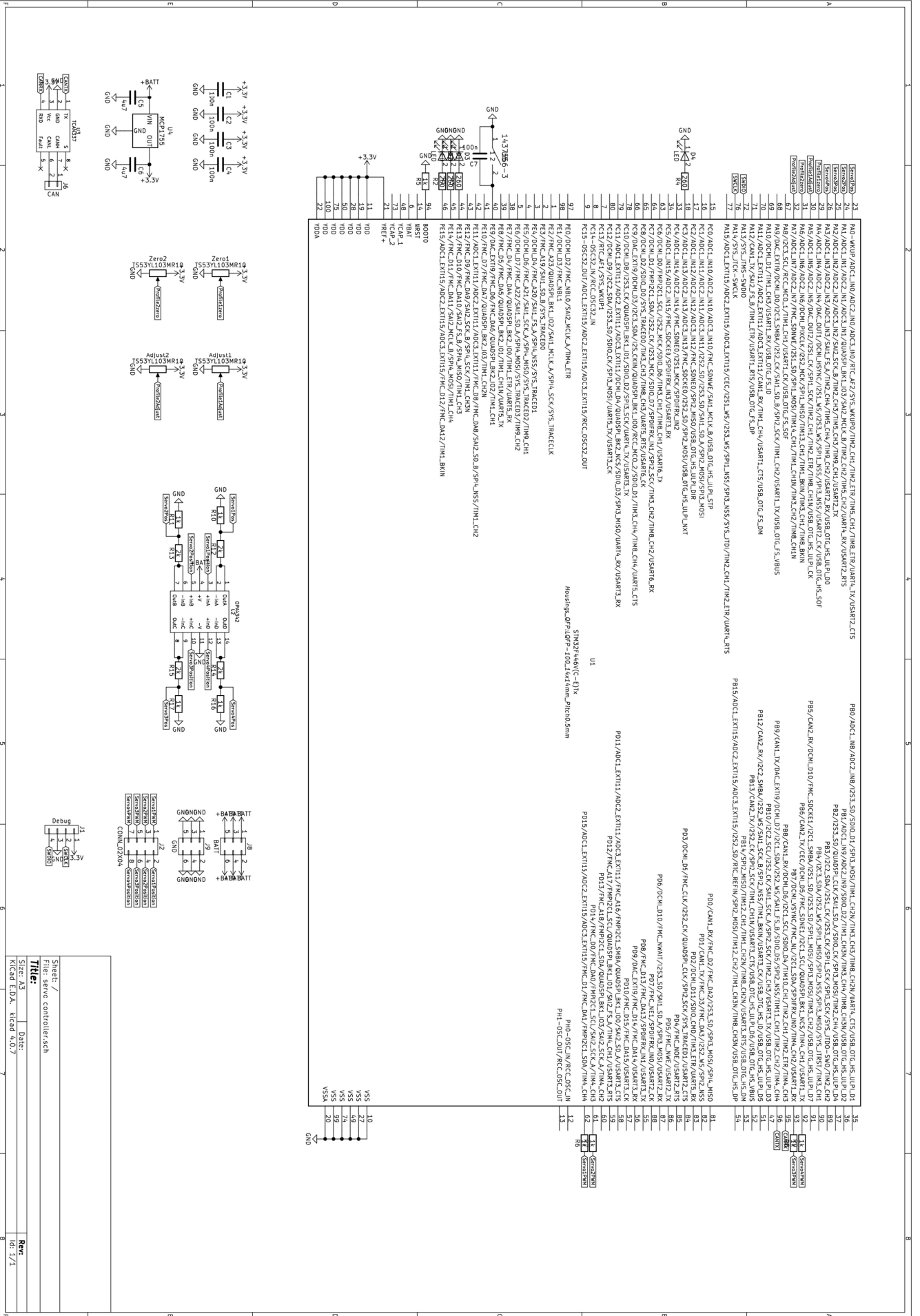
Sheet: /  
 File: ActiveAero\_servocontroller.sch

**Title:**

Size: A4 Date:  
 Kicad E.D.A. kicad 4.0.7

Rev:  
 Id: 1/1

Liite 4. Servo-ohjaimen piirilevy.



Sheet: /	File: servo controller.sch
Size: A3	Date:
Kicad EDA, Hicad 4.0.7	Rev: 1/1

1	PCB1/PCB1	PCB1/PCB1	PCB1/PCB1
2	PCB2/PCB2	PCB2/PCB2	PCB2/PCB2
3	PCB3/PCB3	PCB3/PCB3	PCB3/PCB3
4	PCB4/PCB4	PCB4/PCB4	PCB4/PCB4
5	PCB5/PCB5	PCB5/PCB5	PCB5/PCB5
6	PCB6/PCB6	PCB6/PCB6	PCB6/PCB6
7	PCB7/PCB7	PCB7/PCB7	PCB7/PCB7
8	PCB8/PCB8	PCB8/PCB8	PCB8/PCB8
9	PCB9/PCB9	PCB9/PCB9	PCB9/PCB9
10	PCB10/PCB10	PCB10/PCB10	PCB10/PCB10
11	PCB11/PCB11	PCB11/PCB11	PCB11/PCB11
12	PCB12/PCB12	PCB12/PCB12	PCB12/PCB12
13	PCB13/PCB13	PCB13/PCB13	PCB13/PCB13
14	PCB14/PCB14	PCB14/PCB14	PCB14/PCB14
15	PCB15/PCB15	PCB15/PCB15	PCB15/PCB15
16	PCB16/PCB16	PCB16/PCB16	PCB16/PCB16
17	PCB17/PCB17	PCB17/PCB17	PCB17/PCB17
18	PCB18/PCB18	PCB18/PCB18	PCB18/PCB18
19	PCB19/PCB19	PCB19/PCB19	PCB19/PCB19
20	PCB20/PCB20	PCB20/PCB20	PCB20/PCB20
21	PCB21/PCB21	PCB21/PCB21	PCB21/PCB21
22	PCB22/PCB22	PCB22/PCB22	PCB22/PCB22