

Emanuela Tran
Oscar Øgar Meisal
Simen Bergsvik

Design av trefase vekselretter og motorkontroller til synkronmotor

Three phase inverter & motor controller design
for synchronous motor

Bacheloroppgave i elektronikk og automatiseringsteknikk

Veileder: Irja Gravdahl

Mai 2021

Emanuela Tran
Oscar Øgar Meisal
Simen Bergsvik

Design av trefase vekselretter og motorkontroller til synkronmotor

Three phase inverter & motor controller design for synchronous motor

Bacheloroppgave i elektronikk og automatiseringsteknikk
Veileder: Irja Gravdahl
Mai 2021

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk



Kunnskap for en bedre verden

Bacheloroppgave

Oppgavens tittel: Design av trefase vekselretter og motorkontroller til synkronmotor	Gitt dato: Januar 2021
	Innleveringsdato: 20.05.2021
	Gradering: <input checked="" type="checkbox"/> åpent <input type="checkbox"/> lukket <input type="checkbox"/> åpent fra _____
	Antall sider: 113
Gruppedeltakere: Emanuela Tran Oscar Øgar Meisal Simen Bergsvik	Veiledere: Irja Gravdahl (<i>fra 07.05.21</i>) irja.gravdahl@ntnu.no Fredrik Dessen (<i>04.01.21-30.04.21</i>) fredrik.dessen@ntnu.no
Studieretning: Elektronikk Automatiseringsteknikk	Prosjektnummer: E2108
Oppdragsgiver: Revolve NTNU	Kontaktperson hos oppdragsgiver: Marius von Hafenbrädl marius.vonhafenbradl@revolve.no

Forord

Denne rapporten er sluttproduktet til en bacheloroppgave for elektroningeniørstudiet, utført av tre studenter ved NTNU med automatiseringsteknikk og elektronikk som studieretninger. Det har vært et lærerikt prosjekt, hvor gruppen har utviklet sin kompetanse og hatt en god faglig vekst. Oppgaven har vært ekstremt krevende og ikke minst spennende. Ved å utføre et slikt prosjekt i en tverrfaglig gruppe, har man fått et større overblikk over hvordan de ulike elementene henger sammen, og hvor avhengig man er av forskjellig kompetanse for å utvikle et velfungerende system.

Målet med denne rapporten er å gi leser et innblikk i utførelse og resultat av oppgaven. Samt gjøre rede for avgjørelser tatt underveis.

Gruppen ønsker å takke organisasjonen Revolve NTNU for å la oss få muligheten til å utføre en slik oppgave hos dem. Vi ønsker å takke for samarbeidsvilligheten gjennom hele prosjektet, og håper vårt arbeid er noe Revolve NTNU får et godt utbytte av.

Gruppen ønsker også å rette en stor takk til veilederne ved NTNU, Irja Gravidahl og Fredrik Dessen.

Trondheim, 2021-05-20

Emanuela Tran

Oscar Øgar Meisal

Simen Bergsvik

..... ..... 

Sammendrag

Opggaven dokumenterer utviklingen av en vekselretter-prototype og motorkontroller for en børsteløs DC-motor. Prototypen er laget for Revolve NTNU, og skal i fremtiden kunne brukes som styresystem for organisasjonens førerløse racerbil. Målet med oppgaven har vært å utforske muligheten ved å lage et styresystem og tilrettelegge for videreutvikling.

Motorkontrolleren som har blitt utviklet baserer seg på vektorkontroll, som ved hjelp av transformasjoner gjør om et sett med ulineære vekselstrømmer, og representerer disse som lineære likestrømmer. To strømregulatorer har blitt implementert for å regulere statorstrømmene, i tillegg til en PID-regulator for posisjonsregulering.

Det har blitt laget en vekselretter, som er delt opp i to kretskort. Kontrollkortet er designet for å lese måledata og kjøre kontrollalgoritmen, og powerstage-kortet gjør om likestrøm til vekselstrøm ved svitsjing av transistorer.

Rapporten dokumenterer den underliggende teorien og utviklingsprosessen som har ført til de ulike løsningene i systemet. Gruppen har designet en prototype for vekselretting, og har laget en fungerende modell av kontrollsystemet. Det reflekteres også over valg som har blitt tatt, i tillegg til hvordan systemet kan videreutvikles.

Abstract

This thesis documents the development of an inverter and motor controller prototype for a brushless DC-motor. The prototype is made for Revolve NTNU, and can be used as a steering system for the organization in the future. The scope of the thesis is to explore the possibility of developing a steering system that can be further developed.

The motor controller that has been developed is based on field oriented control which by the use of transforms, converts non-linear ac-currents and represent these as linear dc-currents. Two current controllers have been implemented to control the stator currents, as well as a PID controller for position control.

An inverter has been developed where the system has been divided into two printed circuit boards. The control card will read measurements and run the control algorithm, while the powerstage converts direct current to alternating current by switching transistors.

This thesis documents relevant theory and the development process that has led to the various solutions present in the system. The group has designed a prototype for an inverter, as well as a working model of the control system. Furthermore, it reflects on choices that have been made, in addition to further development of the system.

Innhold

Forord	I
Sammendrag	II
Abstract	II
Forkortelser	IX
1 Innledning	1
1.1 Revolve NTNU	1
1.1.1 Formula Student	2
1.2 Det autonome styresystemet	3
1.3 Problemstilling	4
1.4 Mål for prosjektet	4
1.5 Prosjektdeltakere	6
1.6 Rapportens oppbygning	7
2 Teori og underliggende materiale	8
2.1 DC-motorer	8
2.1.1 Konstruksjonen til børsteløse DC-motorer	8
2.1.2 Virkemåte	9
2.2 Kraftelektronikk	12
2.2.1 Transistorteknologi	12
2.2.2 Snubber-krets	14
2.2.3 Gatedrivere	15
2.2.4 Bootstrapkrets	16
2.2.5 Pulsbreddemodulert vekselretter	18
2.2.6 Trefase vekselretter	21
2.3 Motorkontroll	22
2.3.1 Vektorkontroll	22
2.3.2 Park Clarke transformasjoner	23
2.3.3 Posisjonsmåling	24
3 Utstyr og metode	26
3.1 Metode	26
3.1.1 Planlegging og forarbeid	26

3.1.2	Informasjonsinnhenting og dokumentasjon	27
3.1.3	Kvalitetssikring	28
3.1.4	Kommunikasjon og samarbeid	28
3.2	Økonomi	29
3.3	Tidsforbruk	30
3.4	Utstyr og ressurser	31
3.4.1	Prosjekteringsressurser	31
3.4.2	Motor og gir	32
3.4.3	Programvare	32
3.4.4	Testutstyr	33
4	Arbeid og resultater	34
4.1	Systemspesifikasjon	34
4.2	Hardware design	35
4.2.1	Spesifikasjoner for mønsterkort	36
4.2.2	Elektromagnetisk forstyrrelse og elektrostatisk utladning	37
4.2.3	Effektiv plassering av komponenter	39
4.2.4	Testpunkter	40
4.2.5	Spenningsforsyning	41
4.3	Hardware design powerstage	43
4.3.1	Trefase vekselretter	44
4.3.2	Valg av transistor	45
4.3.3	Valg av DC-link-kondensatorer	45
4.3.4	Valg av gatedriver	47
4.3.5	Design av bootstrapkrets	48
4.3.6	Strømmåling	49
4.4	Kontrollkort	51
4.4.1	Mikrokontroller	51
4.4.2	Enkoder	52
4.4.3	CAN	54
4.4.4	USB-strømforsyning	54
4.5	Programmering av mikrokontroller	55
4.5.1	Valg av type programmering	55
4.5.2	PWM-generering	55
4.6	HIL-test	58
4.6.1	Testoppsett	59
4.6.2	Testresultater for HIL-test	59
4.7	Motorkontroll og simulering	62
4.7.1	Motormodell	63

4.7.2	Overordnet reguleringsystem	67
4.7.3	Strømregulatorer	69
4.7.4	Posisjonsregulator	73
4.7.5	Modelleringsformat og kodegenerering	75
4.7.6	Arbeidsflyt	78
5	Konklusjon	79
5.1	Oppsummering og konklusjon	79
5.2	Diskusjon	80
5.2.1	Prosjektets omfang	80
5.2.2	Prosjektering	81
5.3	Fremtidig arbeid	81
5.3.1	Kretskortdesign	81
5.3.2	Programvare	82
5.3.3	Modellering	82
5.4	Gruppens refleksjon	83
	Kilder	85
A	Vedlegg	88
A.1	Forventningsavklaring	89
A.2	Maxon EC-i 52 180W datablad	91
A.3	Planetgir GP 52 C datablad	92
A.4	Skjematikk Powerstage main	93
A.5	Skjematikk Powerstage power	94
A.6	Skjematikk Powerstage gatedriver	95
A.7	Skjematikk Powerstage current measurement	96
A.8	Skjematikk Powerstage powerstage	97
A.9	Skjematikk Control main	98
A.10	Skjematikk Control power	99
A.11	Skjematikk Control MCU	100
A.12	Skjematikk Control CAN	101
A.13	Skjematikk Control RS422	102
A.14	Poster	103

Figurer

1.1	Revolve NTNUs elektriske racerbil Nova, produsert i 2019	1
1.2	Mekanisk oppsett av styreaktuator	3
1.3	Flytdiagram for det autonome styresystemet	3
2.1	Konstruksjon til BLDC-motor	9
2.2	Direkte- og kvadraturakse til rotor	9
2.3	Seks-steg kommutering av BLDC-motor	10
2.4	Sinusoidal kommutering	11
2.5	Symbol til MOSFET med 3-terminaler: gate(G), source(S) og drain(D)	12
2.6	Oppbygning av en n-kanal MOSFET	13
2.7	i-v karakteristikk for en N-kanal MOSFET	13
2.8	DC-snobberkrets	15
2.9	Bootstrap ladningsvei	16
2.10	Bootstrap utladningsvei	16
2.11	Halv-bro vekselretter	18
2.12	Pulsbreddemodulasjon for en fase	19
2.13	Effekten av dødtid t_{Δ} på en vekselretter	20
2.14	Trefase vekselretter	21
2.15	Pulsbreddemodulasjon for tre faser	21
2.16	Vektorkontroll for posisjonsregulering	22
2.17	Clarke Park transformasjon på et balansert trefase system	23
2.18	Representasjon av posisjonsmåling ved bruk av inkrementell enkoder	25
3.1	Timeforbruk per prosjektdeltaker	30
4.1	Styremoment Formula Student Spain 2019	34
4.2	Illustrasjon av det elektriske systemet	36
4.3	Utklipp av powerstage-kortet som illustrerer bruken av kobberplan og ledning- banene til de seks PWM-signalene som går inn på gatedriverene	37
4.4	Spenningsplan for kontrollkort og powerstage	38
4.5	Utklipp av powerstage som illustrerer TVS diodene, D1, D2 og D3, som er plas- sert nært konnektoren til motoren	39
4.6	Plassering av viaer på powerstage for å lede unna varme	39

4.7	Probing av rippel på 3.3V-regulator	40
4.8	Referanseutlegg fra databladet til TPS5456	41
4.9	Utlegg for spenningsregulatorerne for 5V og 3.3V fra kontrollkortet	41
4.10	Testresultat av 3.3V	42
4.11	Testresultat av 5V	42
4.12	Gjengivelse av powerstage	43
4.13	Utklipp av overordnet skjematikk for vekselretteren	44
4.14	Utklipp fra skjematikk av kondensatorbank	46
4.15	Utklipp av plassering av kondensatorbank i forhold til transistorene	46
4.16	Diagram for elektrisk karakteristik av drain-source i forhold til $r_{DS(on)}$	47
4.17	Forenklet applikasjon av gatedriver UCC27288	47
4.18	Typisk applikasjon av bootstrapkrets for gatedriver UCC27288	48
4.19	Applikasjon av strømmåling	49
4.20	Utklipp fra skjematikken til strømmålingskretsen	50
4.21	Utlegg for strømmålingkrets for fase U og V	50
4.22	Gjengivelse av kontrollkort	51
4.23	Skjematikk av RS-422 mottaker	54
4.24	Oppsett for PWM-generering	56
4.25	Sinusbølger generert som pulsbredden	56
4.26	To komplementære PWM-signaler	57
4.27	500ns dødtid mellom to komplementære PWM-signaler, 100ns/div	57
4.28	HIL-testplan	58
4.29	Testoppsett for HIL-test	59
4.30	Toppide PWM-signaler	60
4.31	Fase V	60
4.32	Fase U	61
4.33	Reguleringsystem og motormodell	62
4.34	PMSM-blokk i Simulink	63
4.35	Parametre for PMSM-blokk i Simulink	65
4.36	Komplett modell av motor, gir og last	66
4.37	Subsystemet <i>Measurements</i> med tidsforsinkelse på fasestrømsignalene	66
4.38	Reguleringsystem i Simulink	67
4.39	Parametre for reguleringsystemet i Simulink	67
4.40	Park- og Clarke-transformasjoner i Simulink	68
4.41	Clarke-transformasjon i Matlab-script	69
4.42	Park-transformasjon i Matlab-script	69
4.43	Strømregulatorer i Simulink	69
4.44	Pådragsbegrensning i <i>dq</i> -regulatorer i Simulink	70
4.45	Forenklet elektrisk modell av BLDC	71

4.46 q -strømregulator ved sprangrespons på komplett system	72
4.47 d -strømregulator ved sprangrespons på komplett system	72
4.48 Posisjonsregulator i Simulink	73
4.49 Posisjonsregulator ved sprang på 30°	73
4.50 Simuleringsresultater ved sprangrespons på hjulvinkel	74
4.51 Innstilling for solver i Simulink	76
4.52 Generert kode for definering av regulatorparametre	77
4.53 Generert kode for begrenning av pådrag fra posisjonsregulator	77
4.54 Vekselretter i Simulink ved bruk av Simscape-pakken	78

Tabeller

1.1 Maksimal poengsum for de ulike delkonkurransene	2
3.1 Sluttkostnad	29
3.2 Systemspesifikasjoner til Maxon EC-i52 og gir	32
3.3 Testutstyr	33
4.1 Sammenligning av forskjellige felteffekttransistorer	45
4.2 Pinnekonfigurasjon og funksjon for UCC27288	48
4.3 Oppløsning på posisjonsmåler	53
4.4 Symboler i likninger for PMSM-blokk i Simulink	64

Forkortelser

AC - Alternating Current

ADC - Analog to Digital Converter

BLDC - Brushless DC

CAN - Controller Area Network

CANOpen - Controller Area Network - open

CAN-FD - Controller Area Network - Flexible Data-rate

CMMR - Common-Mode Rejection Ratio

CV - Combustion Vehicle

DC - Direct Current

ESD - Electrostatic Discharge

EV - Electric Vehicle

FPU - Floating Point Unit

IC - Integrated Circuit

LCM - Least Common Multiple

LV - Low Voltage

MOSFET - Metal Oxide Semiconductor Field Effect Transistor

NTNU - Norges Teknisk-Naturvitenskapelige Universitet

PC - Personal Computer

PMSM - Permanent Magnet Synchronous Motor

PWM - Pulse-Width Modulation

SWD - Serial Wire Debug

TVS - Transient Voltage Suppression

USB - Universal Serial Bus

UVLO - Undervoltage Lockout

HIL - Hardware In the Loop

Innledning

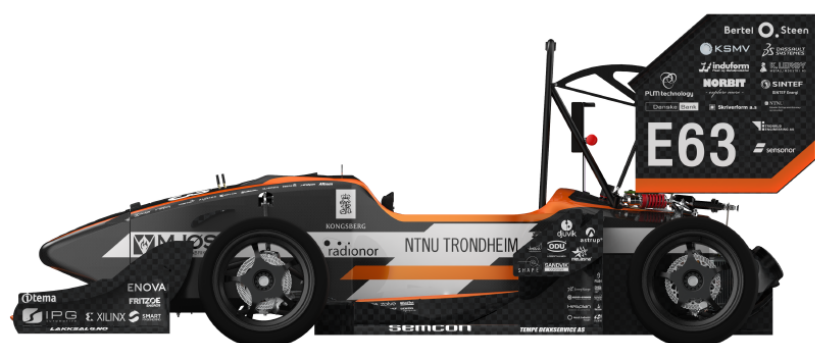
1.1 Revolve NTNU

Revolve NTNU er en teknisk studentorganisasjon som produserer små elektriske racerbiler i formel-1 stil for å delta i verdens største ingeniørkonkurranse for studenter - Formula Student. Organisasjonens visjon er å utvikle medlemmene til å bli verdensklasseingeniører ved å omsette teori til praksis i et utfordrende prosjekt.

I løpet av ett år produserer vanligvis organisasjonen en elektrisk racerbil, i tillegg til å videreutvikle en førerløs racerbil fra tidligere år, som begge deltar i konkurranser på sommeren. I 2022 skal organisasjonen imidlertid kun konkurrere med en racerbil, som skal fungere både førerløst og med fører. Under konkurransene blir racer bilen og teamet vurdert av toppingeniører fra bilindustrien der hvert lag får poeng ut i fra racerbilens ytelse, så vel som studentenes evne til å ta gode designvalg.

Studentorganisasjonen består av omtrent 60-70 studenter fordelt i ulike grupper som er med på å designe ulike systemer på bilen. Gjennom ett år arbeides det med alt fra planlegging og design, til sponsorkontakt og designrapporter. Siden opprettelsen av organisasjonen i 2012 har Revolve NTNU klatret til å være blant de topp 10 best rangerte lagene i verden.

Det overordnede målet til Revolve NTNU er å være topp 3 i alle konkurranser de deltar i.



Figur 1.1: Revolve NTNUs elektriske racerbil Nova, produsert i 2019

1.1.1 Formula Student

Formula Student er den mest etablerte ingeniørkonkurransen for studenter i Europa, med mye støtte fra industrien og høyprofilerte ingeniører. Målet med en slik konkurranse er å utvikle engasjerte og innovative ingeniører, og oppmuntre flere til å velge en karriere innenfor ingeniørfag. Av den grunn består Formula Student av en rekke delkonkurranser. Det skilles mellom dynamiske og statiske delkonkurranser. I de dynamiske delkonkurransene er det selve bilen og dens ytelse som vurderes. I de statiske delkonkurransene er det studentenes ingeniørkunnskaper og valg tatt underveis i prosjektet som blir bedømt.

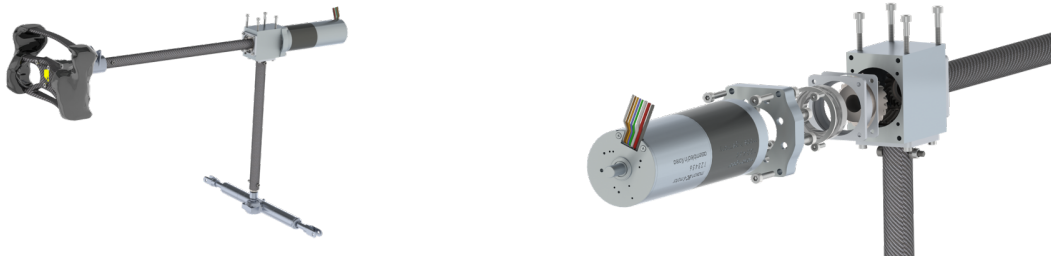
I løpet av konkurransen konkurrerer lagene om å få så høy poengsum som mulig. I tabell 1.1 er de ulike delkonkurransene med respektiv maksimal poengsum fremvist. De dynamiske delkonkurransene står for cirka 2/3 av totalscoren, mens de statiske står for resterende. Det er derfor viktig å ikke kun lage en bil som gjør det bra i de dynamiske delkonkurransene, men også å fokusere på de statiske.

	CV & EV	DC
Static Events:		
Business Plan Presentation	75 points	-
Cost and Manufacturing	100 points	-
Engineering Design	150 points	150 points
Dynamic Events:		
Skid Pad	50 points	-
DV Skid Pad	75 points	75 points
Acceleration	50 points	-
DV Acceleration	75 points	75 points
Autocross	100 points	-
DV Autocross	-	100 points
Endurance	250 points	-
Efficiency	75 points	-
Trackdrive	-	200 points
Overall	1000 points	600 points

Tabell 1.1: Maksimal poengsum for de ulike delkonkurransene

1.2 Det autonome styresystemet

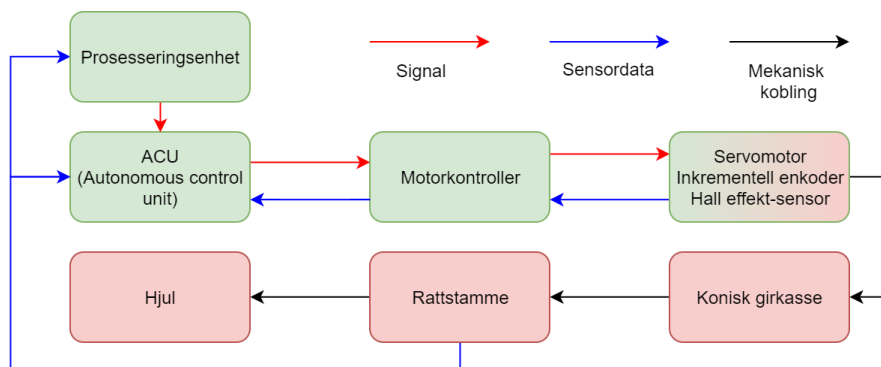
Dagens styresystem består av en børsteløs DC-motor og motorkontroller fra Maxon Group. Motoren er koblet parallelt til styrestaget via en konisk girkasse som vist i figur 1.2.



(a) Fullstendig styresystem med ratt, styrestag, (b) Mekanisk kobling mellom motor og styrestag motor, konisk girkasse og rattstamme via koniske gir

Figur 1.2: Mekanisk oppsett av styreaktuator

Motorkontrolleren har i hovedoppgave å regulere hjulvinkelen til bilen, og dermed også fasestrømmene til motoren. Posisjonsregulatoren er en PID-regulator som bruker rotorvinkelen, målt av en inkrementell enkoder eller en hall effekt-sensor, som tilbakekobling. Settpunktet den mottar blir generert fra en prosesseringsenhet basert på bilens hastighet og plassering i kjørebanelen. Hvordan settpunkt genereres er imidlertid utenfor oppgavens skop og kommer derfor ikke til å bli dekket ytterligere. For å kommunisere med enheten brukes CANopen-protokollen, en protokoll som er mye brukt i industrien, men som ikke brukes ellers i Revolve NTNU, hvor CAN-FD er standarden. Figur 1.3 viser samhandlingen mellom de ulike enhetene i styresystemet.



Figur 1.3: Flytdiagram for det autonome styresystemet

I 2022 skal Revolve NTNU kun produsere én bil som skal konkurrere både med og uten fører. Dette vil by på mange nye utfordringer, blant annet hvordan det førerløse styresystemet skal implementeres på en måte som gir høy ytelse uavhengig om bilen kjører førerløst eller ikke. Organisasjonen ønsker derfor å utforske dette området.

1.3 Problemstilling

I forbindelse med at Revolve NTNU i 2022 skal konkurrere med én racerbil, både førerløst og med fører, ønsker organisasjonen å utforske mulighetene ved å designe en selvlaget motor-kontroller for styresystemet. Det nåværende styresystemet til den førerløse racer bilen baserer seg på kjøpt utstyr. Egendesignede løsninger har derfor ikke blitt undersøkt tidligere. Med dette som utgangspunkt ble problemstillingen formulert slik:

Det skal designes en veksleretter og motorkontroller for autonom styring av hjulvinkelen på Revolve NTNUs biler. Systemet skal dimensjoneres ut i fra nåværende behov, og ha fokus på å tilrettelegge for testing og videreutvikling.

1.4 Mål for prosjektet

Å ha gode mål er en viktig del av ethvert prosjekt. Uten tydelige mål for hva prosjektet skal resultere i, kan det raskt oppstå situasjoner hvor det vil være vanskelig å ta gode valg. Ved å sette konkrete mål for prosjektet, vil disse fungere som retningslinjer underveis.

I forprosjektet ble det derfor diskutert hva gruppen og Revolve NTNU som organisasjon ønsket å oppnå ved prosjektslutt. Gruppen innså imidlertid at målene som ble satt i forprosjektet var for ambisiøse for et prosjekt av denne størrelsen. Det ble derfor utformet nye mål som bedre reflekterte gruppen og organisasjonens ønsker.

Målene baserer seg på problemstillingen, men konkretiserer i større grad hva gruppen ønsker at oppgaven skal resultere i. Diskusjon rundt målsettingen vil bli gått inn på i kapittel 5.

Effektmål

Effektmål beskriver hvilken effekt oppdragsgiver ønsker å oppnå ved gjennomføring av prosjektet og representerer gjerne oppdragsgivers motivasjon.

Revolve NTNU har over tid blitt en stor organisasjon med mange ulike teknologier. Når et nytt lag hvert år inntar rekkene krever det mye for organisasjonen å opprettholde kompetanse innen ulike fagområder. En gjentakende utfordring er at kunnskap om teknologier som kun brukes på enkelte systemer går tapt. Prosjektgruppen har derfor som effektmål å samle kompetansen rundt et mindre antall ulike teknologier. Dette vil legge til rette for at flere medlemmer har kompetanse innen samme område, uten å skape avhengigheter av enkelt-medlemmers kunnskap.

Resultatmål

Resultatmålet prosjektgruppen har satt kan anses som et delmål for å oppnå effektmålet beskrevet i forrige avsnitt. For å samle kompetansen i organisasjonen har prosjektgruppen satt mål om å utvikle et minimumsprodukt med overvekt av teknologier og komponenter som er standardisert ellers i organisasjonen. For at prosjektgruppen skal ha produsert et minimumsprodukt, må gruppen ha designet:

- Et reguleringsystem for posisjonsregulering av en synkronmotor
- En modell for analyse og simulering av det regulerede styresystemet
- En prototype av en vekselretter og et digitalt reguleringsystem som er tilrettelagt for testing og videreutvikling. Dette innebærer at systemet har følgende funksjonaliteter:
 - Trefase vekselretting
 - Posisjonsmåling
 - Strømmåling
 - CAN-kommunikasjon

Prosessmål

Prosessmålene forklarer hva gruppen ønsker å oppnå ved gjennomførelse av prosjektet. Gruppen har som mål:

- Å jobbe for å oppnå toppkarakter
- Å utfordre seg selv på områder de ikke hadde inngående kunnskap om ved prosjektstart. Temaer innen dette området er:
 - Motorkontroll
 - Kraftelektronikk
 - Utvikling og produksjon av kretskort
 - Modellering for implementering av digitale reguleringsystem
- Å utvikle et godt samarbeid og dra nytte av hverandres kompetanse

1.5 Prosjektdeltakere

Emanuela Tran

Studieretning: Elektronikk

Alder: 23

Hjemsted: Lier

Tlf.: +47 410 67 656

E-post: ettran@stud.ntnu.no



Oscar Øgar Meisal

Studieretning: Automatiseringsteknikk

Alder: 24

Hjemsted: Bærum

Tlf.: +47 477 56 062

E-post: oscarom@stud.ntnu.no



Simen Bergsvik

Studieretning: Automatiseringsteknikk

Alder: 22

Hjemsted: Bergen

Tlf.: +47 941 66 866

E-post: simber@stud.ntnu.no



1.6 Rapportens oppbygning

Kapitlene videre i rapporten er delt inn i fire deler: Teori og underliggende materiale, Utstyr og metode, Arbeid og resultater og Konklusjon. I kapitlet - Teori og underliggende materiale - blir nødvendig teori for å designe en vekselretter og motorkontroller presentert. Her vil underliggende teori om motor, kraftelektronikk og motorkontroll presenteres.

I delen - Utstyr og Metode - vil rapporten presentere gruppens metoder for planlegging, informasjonsinnhenting og kvalitetssikring. I tillegg vil rapporten gå inn på hvordan gruppen har kommunisert og samarbeidet under prosjektperioden. I dette kapitlet vil også alle ressurser som har blitt brukt i prosjektet bli presentert.

Videre i rapporten vil gruppens arbeid og resultater bli presentert. Her vil de ulike designvalgene argumenteres for.

Avslutningsvis vil gruppen reflektere rundt det utførte arbeidet og de oppnådde resultatene. Oppgaven vil her konkluderes og det vil bli gitt anbefalinger for fremtidig arbeid.

I slutten av rapporten er det vedlagt en plakat. Denne gjenspeiler prosjektet i et annet medium enn denne rapporten, hvor målgruppen er tenkt til å være for fremtidige søkerkandidater til studentorganisasjonen Revolve NTNU.

Teori og underliggende materiale

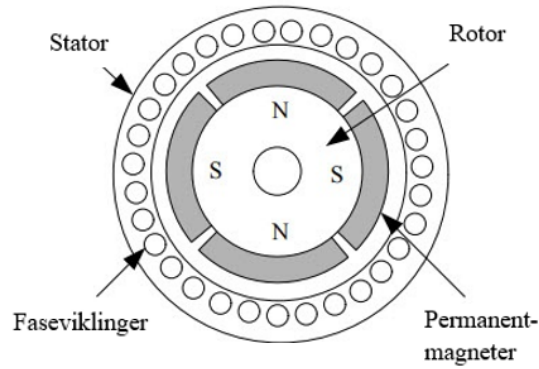
2.1 DC-motorer

DC-motorer er en type elektrisk maskin som omgjør elektrisk energi til roterende mekanisk energi [1]. Det finnes i hovedsak to typer DC-motorer: børstekommuterte og børsteløse motorer. Hovedforskjellen mellom disse er hvordan spolene i motoren kommuteres og hvorvidt spolene er montert i stator eller rotor. I en børstekommutert motor er spolene montert i rotor og disse kommuteres som en konsekvens av at motoren roterer. For en børsteløs motor er spolene montert i stator og krever derfor ekstern elektronikk for å bestemme strømmen gjennom spolene.

2.1.1 Konstruksjonen til børsteløse DC-motorer

En børsteløs DC-motor, heretter kalt BLDC, er en synkron elektrisk motor. Det vil si at hastigheten til rotor er lik hastigheten til det roterende magnetfeltet i stator. I rotor er det montert permanentmagneter i par som sørger for et konstant magnetfelt. Antall polpar har stor påvirkning på motorens karakteristikk, og hvordan magnetene er montert varierer. I noen motorer er magnetene integrert i kjernen til rotor, mens i andre typer er magnetene montert på utsiden.

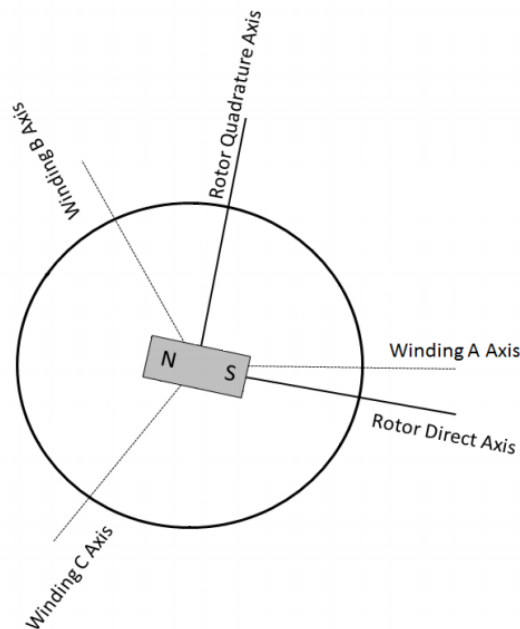
Konstruksjonen til stator er relativt lik en induksjonsmotor og er laget av stål med spor eller tenner til faseviklingene, oftest koblet i en stjernekonfigurasjon. Antall spor bestemmes av om motoren er en, to eller trefaset. I trefase BLDC-motorer er antallet spor delelig på tre, slik at det er et likt antall spoler i hver fase [2]. Hvordan spolene er viklet i stator varierer noe mellom ulike motorkonstruksjoner, men det som kjennetegner BLDC-motorer er at de har konsentrerte statorviklinger. Dette betyr at hver spole er viklet hver for seg, der det er et fysisk skille mellom spolene. Figur 2.1 viser en forenklet modell av en trefase BLDC-motor med to polpar og konsentrerte statorviklinger. De konsentrerte viklingene i stator gjør at flukskoblingen mellom magnetfeltet i stator og magnetfeltet i rotor blir trapesformet. Den motinduserte spenningen som oppstår når motoren roterer er dermed også trapesformet.



Figur 2.1: Konstruksjon til BLDC-motor

2.1.2 Virkemåte

Virkemåten til en BLDC-motor er tilnærmet lik en DC-motor med børster. Ved å påtrykke en spenning på en av faseviklingene vil et elektromagnetisk felt oppstå i stator. Feltet tiltrekker eller frastøter permanentmagnetene i rotor, avhengig av polariteten til den påtrykte spenningen. Dersom det elektromagnetiske feltet i stator ikke er orientert parallelt med direkteaksen til rotor, vil dette føre til at motoren roterer. Størst moment oppnås ved at feltet i stator er orientert langs kvadraturaksen til rotor, med andre ord vinkelrett på direkteaksen, som vist i figur 2.2.

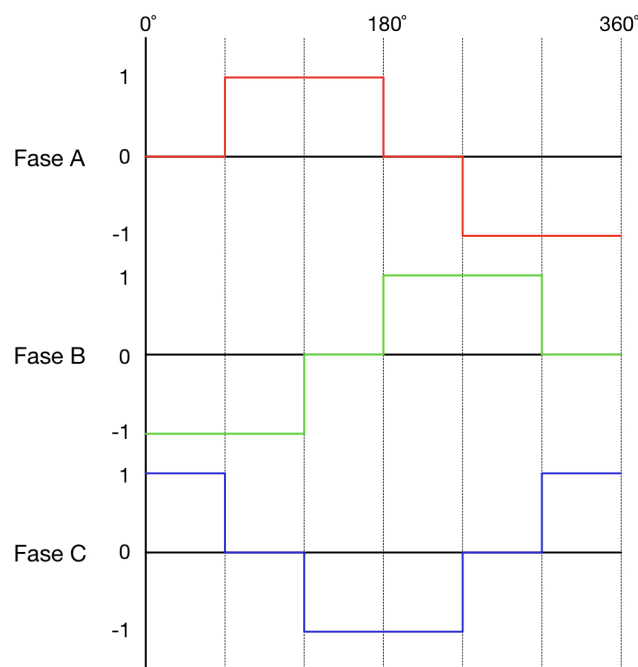


Figur 2.2: Direkte- og kvadraturakse til rotor

For at motoren skal kunne rotere med frekvens f_s , må det elektromagnetiske feltet i stator også rotere med en frekvens lik $n \cdot f_s$, der n er antall polpar i motoren. For å oppnå dette må spolestrømmene reguleres av ekstern elektronikk. Hvordan dette gjøres kalles kommutering og er det som skiller børsteløse DC-motorer fra tradisjonelle DC-motorer med børster. Der børstemotorer kommuteres som en konsekvens av rotasjonen til rotor, må ekstern elektronikk styre hvilke faser som strømsettes til hvilken tid for en BLDC-motor.

Blokkommutering

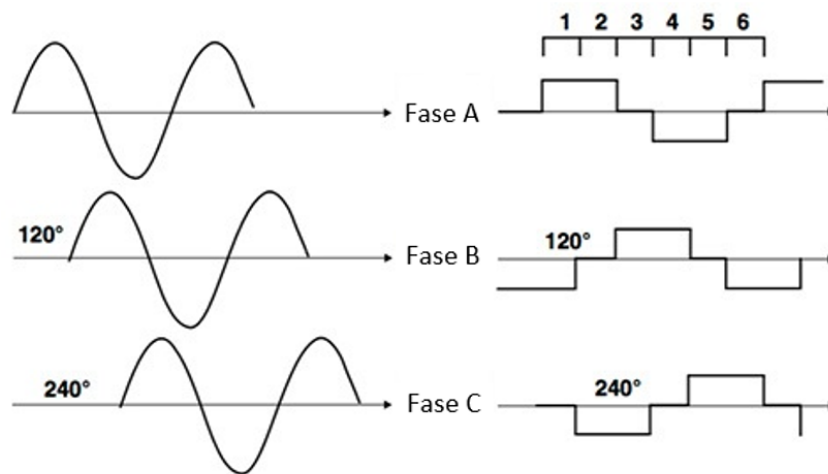
En mye brukt metode for å bestemme hvilke faser som strømsettes er blokkommutering. Denne metoden går ut på at spolene i stator svitsjer mellom å være påtrykt ingen, positiv eller negativ spenning. Når de tre spolestrømmene er faseforskjøvet 120° mellom hverandre, vil spolene oppnå seks ulike tilstander i løpet av en periode [3], som vist i figur 2.3. Denne metoden blir derfor ofte kalt seks-steg kommutering. Fordelene ved denne metoden er at den er relativt enkelt å implementere, og det kreves ikke veldig nøyaktig posisjonsmåling, da spolene kommuteres seks ganger per omdreining av magnetfeltet. Ulempen er derimot at vinkelen mellom elektromagnetfeltet i stator og kvadraturaksen i rotor vil variere periodisk. Dette medfører at produsert moment varierer tilsvarende, og man vil få en momentripping som kan være uønsket.



Figur 2.3: Seks-steg kommutering av BLDC-motor

Sinusoidal kommutering

Et alternativ til blokkommutering, er sinusoidal kommutering. Ved sinusoidal kommutering påføres spolene en tidsvarierende spenning basert på rotors posisjon, som illustrert i figur 2.4. Målet ved denne metoden er å opprettholde 90° mellom magnetfeltet i stator og direkteaksen i rotor. Ved å gjøre dette vil motoren produsere maksimalt moment uavhengig av rotorposisjon [4]. En forutsetning for at dette skal være mulig, er at orienteringen til magnetfeltet i stator kan styres eksakt. Det er derfor nødvendig å kunne styre både amplitude og frekvens til de påtrykte spenningene over statorspolene, i tillegg til at det stilles høyere krav til oppløsningen til posisjonsmåleren.



Figur 2.4: Sinusoidal kommutering

En av de største fordelene ved sinusoidal kommutering er at momentrippelen reduseres betraktelig, da vinkelen mellom elektromagnetfeltet i stator og direkteaksen til rotor reguleres til å være 90° . Dersom man ser på energitapet i faseviklingene, som begrenser hvor mye strøm hver fase tåler, kan man også se at sinusoidal kommutering vil tillate cirka 15 % høyere maksimal strøm, sammenlignet med blokkommutering [5].

Uavhengig av hvilken kommuteringsmetode som benyttes, er det nødvendig med ekstern elektronikk for å styre fasestrømmene i en BLDC-motor. Hvordan elektronikken brukes for å styre fasestrømmene, vil bli omtalt i de neste delkapitlene.

2.2 Kraftelektronikk

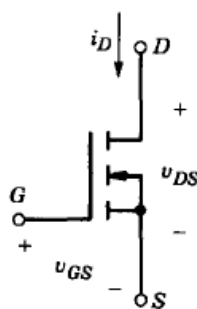
I dette delkapittelet vil det bli presentert generell teori om kraftelektronikk for å få den grunnledende forståelsen for hvordan en trefase vekselretter kan designes. I første del av kapitlet vil den underbyggende teorien bak transistorer bli presentert, hvor fokuset vil være på felt-effekttransistoren kalt MOSFET og dens virkemåte. Videre vil det bli presentert hvorfor gate-drivere er nødvendig i en trefase vekselretter og til slutt vil delkapittelet ta for seg hvordan man modulerer en trefase vekselretter.

2.2.1 Transistorteknologi

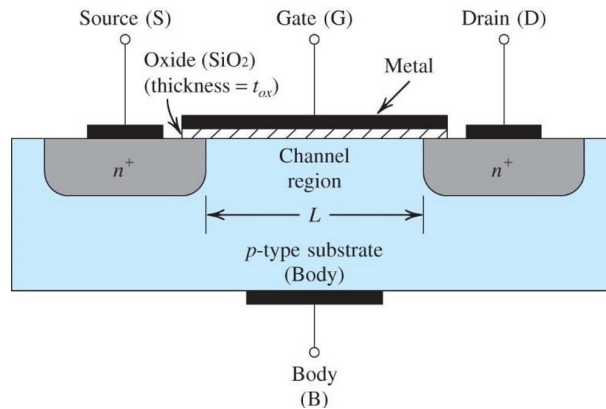
Transistorer er halvlederforsterkere som brukes innenfor elektronikk til forsterkning, kontroll og generering av elektriske signaler. Ved styring av kretser, der det trekkes store strømmer, benyttes krafttransistorer. Slike transistorer brukes ofte som en svitsj og fungerer som en elektrisk bryter som enten leder eller ikke leder.

MOSFET

MOSFET er en 4-terminals komponent som består av gate(G), drain(D), source(S) og body(B). Body-substratet er enten høyt p-dopet eller n-dopet og er kortsluttet til sourceterminalen. MOSFET fungerer derfor i praksis som en 3-terminals komponent, som vist i figur 2.5. I en n-kanal MOSFET som vist i figur 2.6 er drain-source delen n-dopet mens body-substratet er p-dopet. Motsatt ved en p-kanal MOSFET. Gate-terminalen på en MOSFET er elektrisk isolert fra drain-source-terminalene ved at det er et veldig tynt lag med silisiumdioksid som skiller terminalene.



Figur 2.5: Symbol til MOSFET med 3-terminaler: gate(G), source(S) og drain(D)

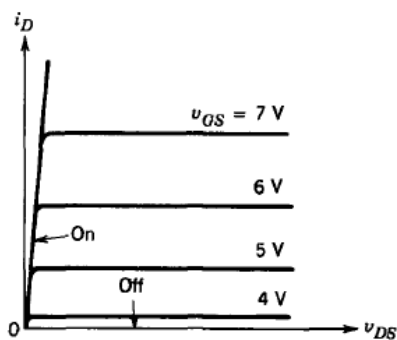


Figur 2.6: Oppbygning av en n-kanal MOSFET

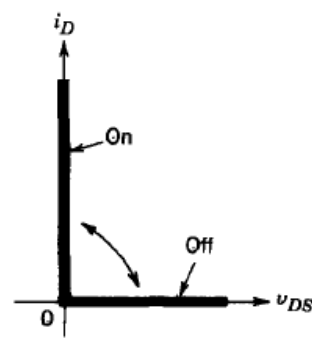
På grunn av mobiliteten til elektroner og oppbygning av de to typene MOSFETer favoriseres ofte n-kanal MOSFET. Gruppen har tatt i bruk n-kanal MOSFET til design av trefase vekselretter og videre i rapporten vil det derfor kun diskuteres om n-kanal MOSFET.

For at MOSFETen skal operere i lukket tilstand krever komponenten en konstant påtrykt spenning over gate-source som er over terskelspenningen $V_{GS(th)}$. Drain-source-strømmen, I_{DS} , er avhengig av drain-source-spenningen V_{DS} over MOSFETen og gate-source spenningen, V_{GS} , som blir påtrykt. Den generelle karakteristikken til I_{DS} som en funksjon av V_{DS} og V_{GS} for en n-kanal MOSFET vises i figur 2.7a.

En ideell MOSFET opererer som en lukket bryter når det påtrykkes en gate-sourcespenning, som vist i figur 2.7b. MOSFETer blir ofte brukt som brytere fordi svitsjingen mellom åpen og lukket tilstand skjer svært raskt. Svitsjetiden kan variere mellom ti til hundre nanosekunder [6].



(a) i-v karakteristik[6]



(b) Ideell i-v karakteristik[6]

Figur 2.7: i-v karakteristik for en N-kanal MOSFET

Motstand mellom drain-source, $r_{DS(on)}$ øker raskt jo nærmere MOSFETen når den maksimale spenningen den tåler BV_{DSS} . $r_{DS(on)}$ kan uttrykkes som en funksjon av den maksimale spenningen MOSFETen tåler, vist i formel 2.1 [6], der k er en konstant som er avhengig av geome-

trien til komponenten.

$$r_{DS(on)} = k \cdot BV_{DSS} \quad (2.1)$$

BV_{DSS} påvirker indremotstanden $r_{DS(on)}$ lineært og på grunn av dette er det kun komponenter med lav spenning som resulterer i lav totalmotstand mellom drain og source. Dette reduserer også ledetap. Ved valg av MOSFET er det derfor ønskelig å velge en transistor med lav indremotstand, men som tåler en høy nok inngangsspenning.

2.2.2 Snubber-krets

Når transistorer svitsjer kan det oppstå høye spenninger og strømmer i korte perioder. Slike transienter kan potensielt være farlige og ødelegge halvlederkomponenten. For å redusere belastningen legges det derfor til en snubber-krets. Snubber-kretser beskytter transistorer under svitsjing ved å begrense spenningene og strømmene. Slike kretser blir plassert rundt halvlederkomponenter for beskyttelse og økt ytelse.

En snubber-krets reduserer belastningen på en transistor under svitsjing ved å:

- Redusere eller eliminere overspenninger og strømmer
- Begrense $\frac{dI}{dt}$ eller $\frac{dV}{dt}$
- Redusere effekttap under svitsjing
- Redusere elektromagnetiske forstyrrelser

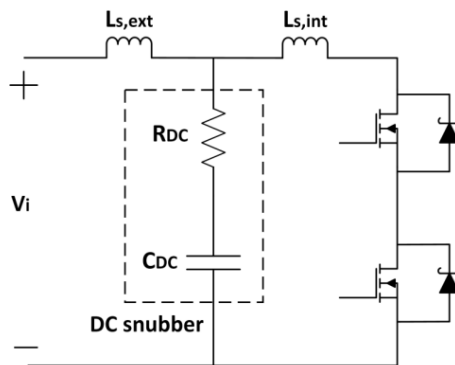
Det finnes i hovedsak tre forskjellige typer snubber-kretser[6]:

- Turn-off snubber
- Turn-on snubber
- Overspenningssnubber

I denne rapporten vil fokuset være på DC-snubber, som er en variant av en overspenningsnubber.

DC-snubberkrets

DC-snubberkretsen demper ringing under svitsjing ved at kretsen plasseres i parallell og nærmest mulig halvlederkomponenten den skal beskytte [7]. Figur 2.8 viser en forenklet applikasjon av en DC-snubber.



Figur 2.8: DC-snubberkrets

Hvor stor kapasitans som må til for å dempe uønskede transienter og beskytte transistorene, beregnes ved hjelp av formel 2.2 [8], der I_m er amplituden til strømmen, T_s er svitsjeperioden, ΔU_{dc_max} er maks spenningsrippel og 0.2156 er spenningsrippel-koeffisienten.

$$C_{link} \geq \frac{0.2165 I_m T_s}{\Delta U_{dc_max}} \quad (2.2)$$

2.2.3 Gatedrivere

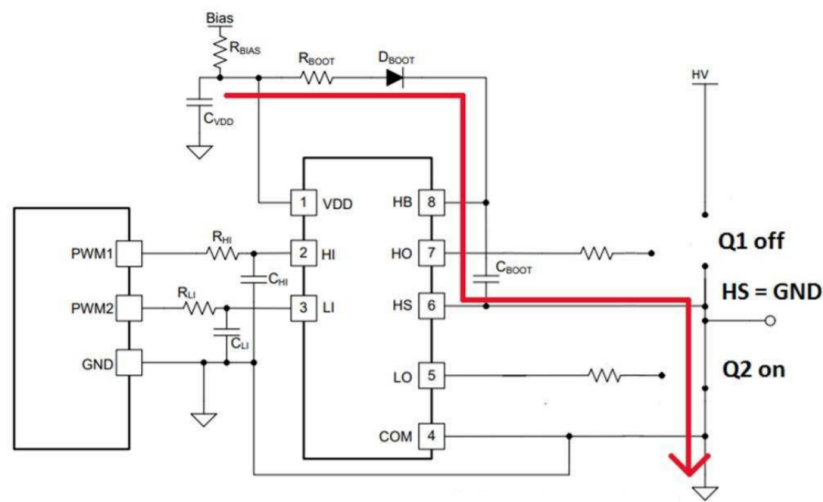
Gatedrivere virker som bindeleddet mellom kontrollsignal og krafttransistorer. Hovedfunksjonen til en gatedriver er å svitsje halvlederkomponenten fra lukket til åpen tilstand og omvendt. Det finnes mange forskjellige kombinasjoner av en gatedriver men, topologien til en gatedriver blir bestemt ut i fra disse tre faktorene: [6]

- Om utgangssignalet fra gatedriveren er unipolar eller bipolar
- Om gatedriveren kan kobles direkte til transistoren eller må være elektrisk isolert
- Om utgangssignalet er koblet parallelt eller i serie med transistoren

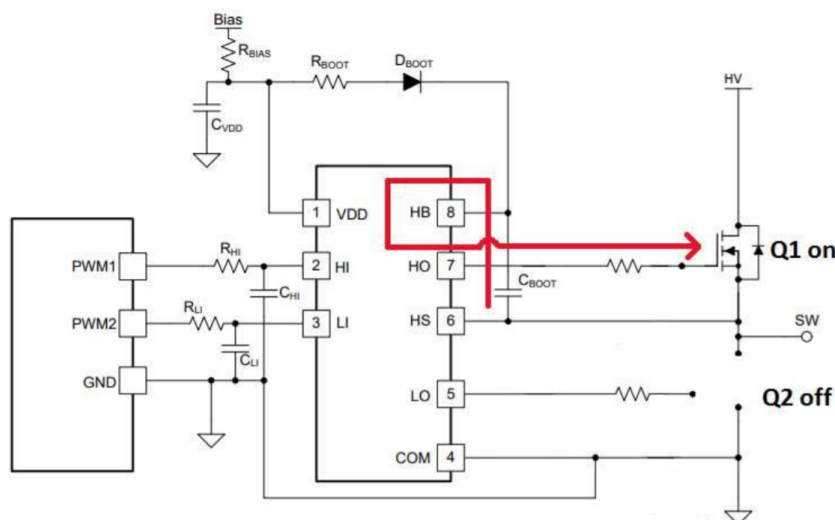
Som tidligere nevnt i delkapittel 2.2.1, trenger transistoren å få påtrykt en høy nok spenning for at den skal gå i lukket tilstand. Den påtrykte spenningen kommer fra gatedriveren. I denne rapporten vil det gås inn på gatedrivere med ekstern bootstrapkrets.

2.2.4 Bootstrapkrets

En bootstrapkrets består i hovedsak av en kondensator, diode, motstand og en avkoblingskondensator. Bootstrapkretser er til for å få gatedriveren til å veksle mellom å påtrykke spenninger på to transistorer. Forenkelt gjøres dette ved å lade og utlade gjennom en kondensator. I dette delkapittelet vil funksjonalitetene til de forskjellige komponentene beskrives, samt hvordan dette gjøres. Figur 2.9 viser ladningsveien til en bootstrapkrets i en forenklet halv-bro konfigurasjon der transistor $Q2$ blir aktivert og kortsluttet til jord. Den lagrede energien i kondensatoren lades ut, og spenningen som er lagret går til å lade $Q1$, som vist i figur 2.10. Svitsjenoden sw blir da dratt opp til HV . [9]



Figur 2.9: Bootstrap ladningsvei



Figur 2.10: Bootstrap utladningsvei

Bootstrapdiode

Bootstrapdioden er plassert etter forsyningen inn på V_{DD} , som vist på figur 2.9, for å beskytte denne inngangen. Når pinne HS kobles til HV er det en risiko for at ladningen fra bootstrapkondensatoren går til pinne V_{DD} . Dioden beskytter inngangen ved at den blokkerer ladningen som kommer ut av bootstrapkondensatoren, da den potensielt kan utløse UVLO og slå av gatedriveren[9].

Schottky-dioder blir ofte brukt ettersom denne typen dioder reagerer raskt. Det at dioden reagerer raskt vil redusere risikoen og minimere strøml lekkasje. Bootstrapdioden bør tåle maks amplitude av transientene som kan oppstå ved svitsjing. Verdien til dioden D_{boot} bestemmes ut i fra dette.

Bootstrapkondensator

Bootstrapkondensatoren blir brukt til lade opp toppsidetransistoren. Verdien på kondensatoren må være stor nok, da den lagrede energien skal drive gateterminalen til toppsidetransistoren. Bootstrapkondensatoren bør være minst ti ganger større enn gatekapasitansen[9].

Gatekapasitansen C_g blir bestemt ut i fra formlene 2.3 og 2.4, der V_{DD} er spenningsforsyningen på gatedriver, V_{Q_g} er spenningen over bootstrapkondensatoren, $V_{BootDiode}$ er spenningsfallet over bootstrapdioden og Q_g er den totale ladningen til gaten på transistoren [9].

$$V_{Q_g} = V_{DD} - V_{BootDiode} \quad (2.3)$$

$$C_g = \frac{Q_g}{V_{Q_g}} \quad (2.4)$$

Når gatekapasitansen er bestemt kan formel 2.5[9] brukes til å bestemme den minste verdien bootstrapkondensatoren kan ha.

$$C_{boot} \geq 10 \cdot C_g \quad (2.5)$$

Avkoblingskondensator

Som vist i figur 2.9, går ladningsveien til bootstrapkondensatoren fra gatedriverforsyningen V_{DD} . $C_{V_{DD}}$ fungerer som en avkoblingskondensator for bootstrapkondensatoren og har som hensikt å støtte opp ved ladning. Kapasitansen til kondensatoren $C_{V_{DD}}$, som vist i formel 2.6, bør være ti ganger større enn bootstrapkondensatoren C_{boot} slik at den ikke blir tappet i løpet av ladningstiden av bootstrapkondensatoren. Forholdet der avkoblingskon-

densatoren er ti ganger større enn bootstrapkondensatoren gir maks 10% rippel på C_{VDD} .

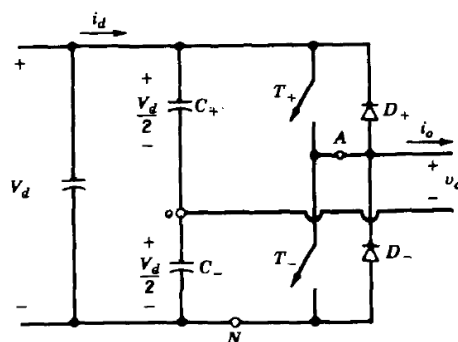
$$C_{VDD} \geq 10 \cdot C_{boot} \quad (2.6)$$

2.2.5 Pulsbreddemodulert vekselretter

En vekselretter er en omformer som gjør om likespenning til vekselspanning. Det finnes ulike typer vekselrettere, der hovedforskjellen er hvordan utgangssignalet blir modulert. For pulsbreddemodulerte vekselrettere, gjøres dette ved å modulere et kontrollsignal som bestemmer amplituden og frekvensen til utgangssignalet. Slike omformere er ofte brukt i applikasjoner der det er behov for vekselspanning, for eksempel ved motordrift. Det finnes ulike topologier for vekselrettere, men i denne rapporten vil det kun fokuseres på pulsbreddemodulerte halv-bro vekselrettere. Delkapittelet vil først gå inn på hvordan vekselretting for ett signal oppnås, etterfulgt av virkemåten til en trefase vekselretter.[6]

Halv-bro vekselretter

En halv-bro vekselretter, som illustrert i figur 2.11, består av et bein med to transistorer i serie. Ved å raskt svitsje mellom åpen og lukket tilstand på de to transistorene, kan gjennomsnittsspenningen på utgangen moduleres til ønsket amplitude og frekvens. De to transistorene kan ikke være i åpen tilstand samtidig, ettersom dette vil kortslutte spenningsforsyningen. Transistorene er derfor komplementære og kan kun være i ulike tilstander. Når T_+ er lukket, vil spenningen over lasten være lik spenningsforsyningen, og når T_- er lukket vil spenningen over lasten være null. Hvilke transistorer som svitsjer og hvor lenge de skal være i hver tilstand, bestemmes av et eksternt kontrollsignal. For å oppnå sinusformet utgangsspenning fra vekselretteren, må kontrollsignalet bestemme hvor ofte transistorene skal svitsje mellom åpen og lukket tilstand[6], slik at gjennomsnittsspenningen øker og avtar i en sinusform.



Figur 2.11: Halv-bro vekselretter

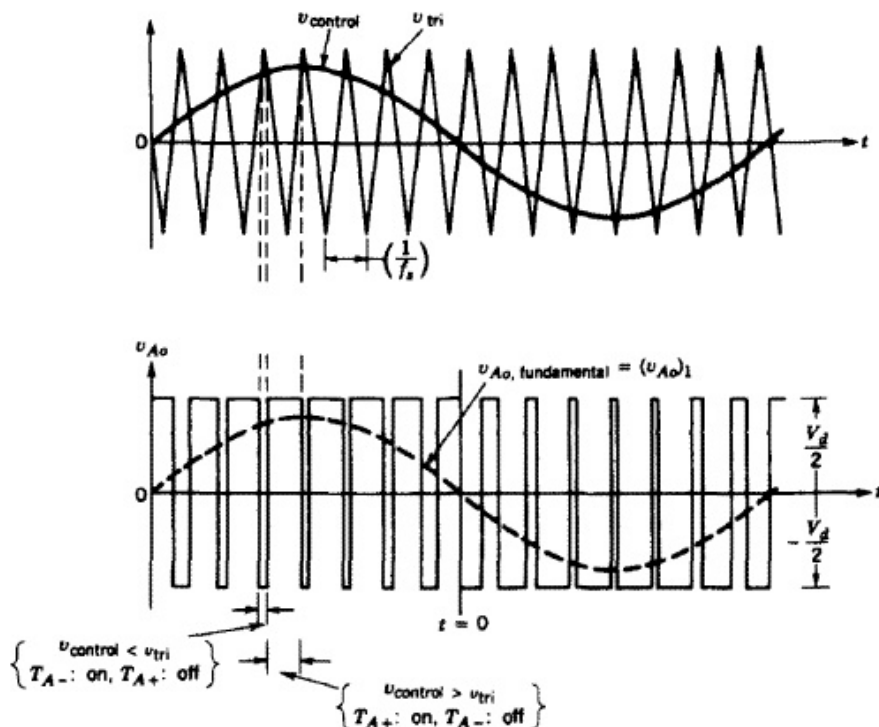
Sinusoidal pulsbreddemodulasjon for halv-bro

Pulsbreddemodulasjon fungerer ved at et kontrollsignal blir sammenlignet med en trekantpuls. Når amplituden til kontrollsignalet er større enn amplituden til trekantpuls vil et høyt signal gis ut, og motsatt når amplituden er lavere. Ved å sammenligne de to signalene vil det dermed genereres pulsbreddemodulerte signaler basert på amplituden og frekvensen til sinussignalet. Figur 2.12 illustrerer hvordan de to signalene sammenlignes og hvordan det resulterende pulsbreddemodulerte signalet ser ut.

Ettersom frekvensen til trekantpuls bestemmer perioden til det pulsbreddemodulerte signalet, bestemmer denne også svitsjefrekvensen til vekselretteren. Forholdet mellom amplituden til kontrollsignalet og trekantpuls kalles amplitudemodulasjonsindeks og er vist i likning 2.7. [6]

$$m_a = \frac{\hat{V}_{control}}{\hat{V}_{tri}} \quad (2.7)$$

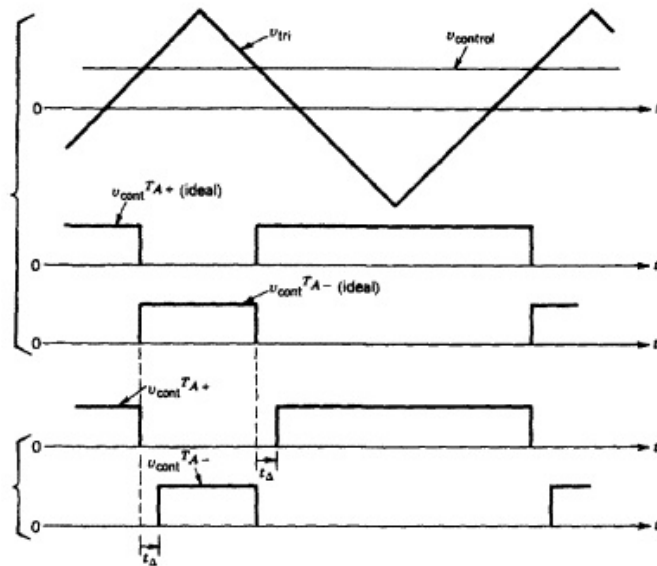
$\hat{V}_{control}$ er amplituden til kontrollsignalet og \hat{V}_{tri} er amplituden til trekantsignalet.



Figur 2.12: Pulsbreddemodulasjon for en fase

Dødtid på pulsbreddemodulerte vekselrettere

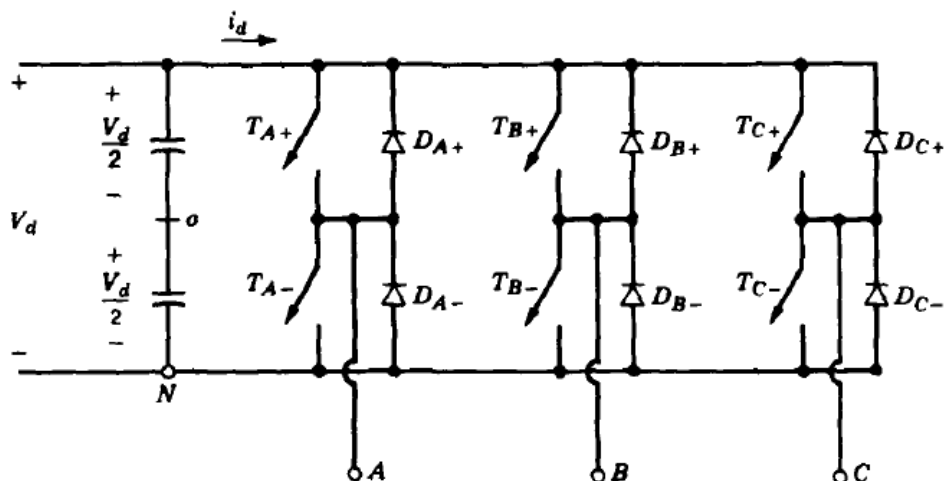
Effekten av dødtid for pulsbreddemodulerte vekselrettere er beskrevet i figur 2.13 med utgangspunkt i en vekselretter med to transistorer. Hittil har transistorene i vekselretteren blitt antatt å være ideell, der svitsjingen mellom åpen og lukket tilstand skjer momentant. I praksis endres imidlertid ikke tilstanden til transistoren momentant. Idet toppsidetransistoren og bunnsidetransistoren svitsjer, vil det derfor være en kort tidsperiode der begge transistorene er i lukket tilstand, slik at spenningsforsyningen kortsluttes. For å unngå dette, legges det til en dødtid mellom kontrollsignalet til de to transistorene. Denne dødtiden settes til noen mikrosekunder for raske svitsjer som MOSFETer, men lengre for andre transistortyper som svitsjer saktere. Som vist i figur 2.13, kan man se et tidsopphold t_{Δ} , der ingen av transistorene er i lukket tilstand.



Figur 2.13: Effekten av dødtid t_{Δ} på en vekselretter

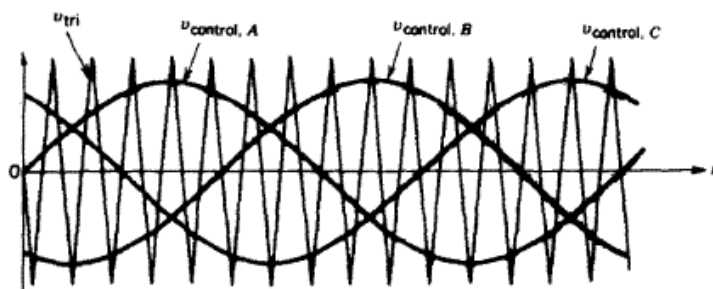
2.2.6 Trefase vekselretter

En trefase vekselretter er en omformer som består av tre halv-broer med felles spenningsforsyning. Vekselretteren omgjør likespenning til tre vekselspenninger som er faseforskjøvet 120° mellom hverandre. Halv-broene kontrollerer sin egen polspenning, altså spenningen fra midtpunktet på benet til den negative polen av spenningsforsyningen. De tre polspenningene definerer spenningene A , B og C , som illustrert i figur 2.14.



Figur 2.14: Trefase vekselretter

I likhet med en enfase vekselretter vil en pulsbreddemodulert trefase vekselretter også kontrollere utgangsspenningene. Kontrollsignalet som styrer svitsjingen til hver halv-bro genereres på samme vis som for en enfase vekselretter. For å oppnå 120° faseforskyvning mellom de tre utgangssignalene, må kontrollsignalene $V_{control}$ til de tre halv-broene sammenlignes med samme trekantpuls, som vist i figur 2.15. Alle transistorene i vekselretteren vil da ha lik svitsjefrekvens og være faseforskjøvet korrekt i forhold til hverandre.



Figur 2.15: Pulsbreddemodulasjon for tre faser

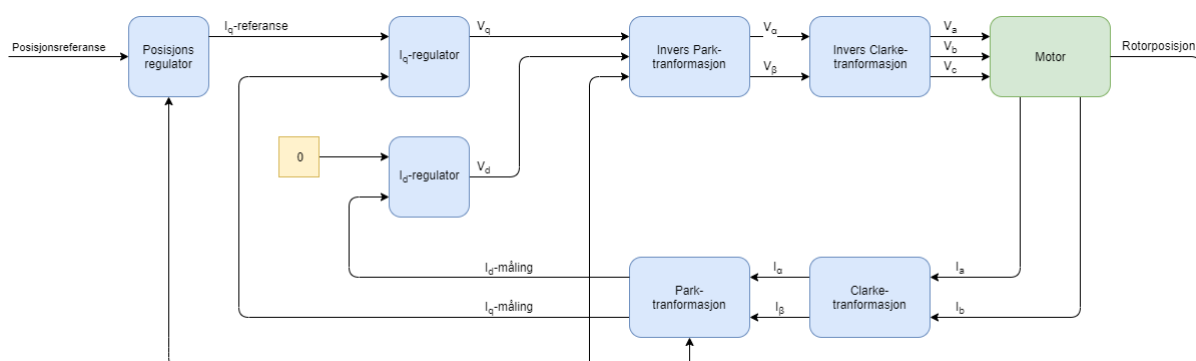
2.3 Motorkontroll

Som nevnt i kapittel 2.1, kreves det ekstern elektronikk for å kommutere en BLDC-motor. I den forbindelse er det også nødvendig med et kontrollsystem for å bestemme hvilke faser som kommuteres til hvilken tid. Det finnes ulike metoder for å regulere fasestrømmene i en BLDC-motor, og hvilken kommuteringsmetode og sensorikk som skal benyttes spiller også en rolle for hva slags kontrollsystem som kan benyttes. I denne rapporten vil det bli fokusert på vektorkontroll med sinusoidal kommutering.

2.3.1 Vektorkontroll

Vektorkontroll er en mye brukt metode for å kontrollere trefase AC-motorer og BLDC-motorer. Metoden går ut på å gjøre et balansert trefase-system med sinusoidale spenninger om til et lineært tidsinvariant system, ved å dekomponere de sinusoidale statorstrømmene til to ortogonale komponenter. Det vil da være mulig å ta i bruk to PI-regulatorer for å regulere statorstrømmene, i tillegg til en PID-regulator for enten posisjon eller hastighetsregulering. De dekomponerte statorstrømmene kalles d - og q -strømmer, ettersom de representerer kraft produsert i direkteaksen og kvadraturaksen til. Ved å kontrollere disse strømmene kan man direkte styre orienteringen til magnetfeltet i stator, og man oppnår dermed kontroll over produsert moment og fluks i motoren. q -strøm vil resultere i at det oppstår en kraft som resulterer i rotasjon, mens d -strøm vil resultere i en kraft som *ikke* fører til rotasjon. For å maksimere det produserte momentet i motoren er det derfor ønskelig å regulere q -strømmen etter ønsket moment, mens d -strømmen minimeres.

Flytdiagrammet i figur 2.16 viser hvordan arkitekturen til vektorkontroll-algoritmen ser ut når den brukes til posisjonsregulering.



Figur 2.16: Vektorkontroll for posisjonsregulering

2.3.2 Park Clarke transformasjoner

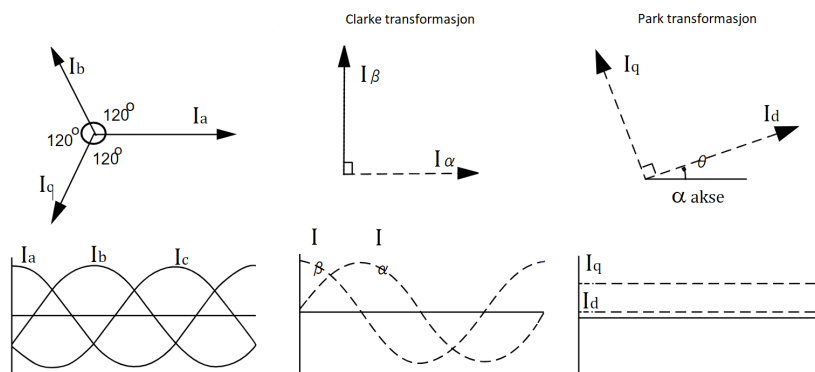
Å regulere en BLDC-motor er i utgangspunktet komplekst ettersom systemet er tidsavhengig hvor både fluks, motindusert spenning og strøm er avhengig av rotasjon. For å forenkle reguleringen, kan Park og Clarke transformasjoner benyttes for å dekomponere trefasestrømmene til to ortogonale dq -strømmer. Clarke transformasjonen projiserer de tre strømvektorene inn i en stasjonær referanseramme, kalt $\alpha\beta$ -rammen. α -aksen ligger på linje med a -aksen, og β -aksen ortogonalt med a -aksen. Transformasjonen brukes på et sett med balanserte strømmer, som vist i likning 2.8. [10]

$$\begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \quad (2.8)$$

Park transformasjonen konverterer vektorene i $\alpha\beta$ -rammen inn i en ortogonal roterende referanseramme, kalt dq -referanseramme. De to strømvektorene vil dermed bli stasjonære i forhold til den roterende referanserammen, hvor d -aksen er på linje med rotorfluksvektoren, og kvadraturaksen (q -aksen) ligger ortogonalt på d -aksen. I likning 2.9 er Park transformasjonen brukt på $\alpha\beta$ strømmene fra likning 2.8, hvor θ er vinkelen mellom d -aksen og α -aksen. [11]

$$\begin{bmatrix} I_d \\ I_q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} \quad (2.9)$$

Ved å bruke Clarke Park transformasjonene kan de tre vekselstrømmene i stator, representeres som to likestrømmer. Transformasjonene er illustrert i figur 2.17.



Figur 2.17: Clarke Park transformasjon på et balansert trefase system

2.3.3 Posisjonsmåling

En forutsetning for å benytte Clarke-Park transformasjonene, er at vinkelen til rotor i forhold til stators magnetfelt er kjent. Denne vinkelen kan estimeres basert på motorparametre og motindusert spenning, men dette krever en grundigere analyse enn det denne oppgaven dekker. Vinkelen til stators magnetfelt θ_e er gitt i likning 2.10, der n er antall polpar og θ_r er rotors vinkel referert til stator.

$$\theta_e = n \cdot \theta_r \quad (2.10)$$

Ved ideell vektorkontroll vil referansestrømmene oppdateres kontinuerlig basert på den eksakte vinkelen til rotors magnetfelt. Det vil da være mulig å oppnå konstant moment i motoren uten rippel. I virkeligheten er det imidlertid to faktorer som begrenser hvor presis de påførte strømmene er:

- Regulatorens hurtighet
- Posisjonsmålerens oppløsning

Strømregulatorens hurtighet avgjør hvor raskt spolestrømmene kan oppdateres basert på nyeste posisjonsavlesning. Dersom regulatorens hastighet er for treg, vil ikke det induserede magnetfeltet være orientert optimalt, og det vil oppstå en momentripping. Hastigheten til strømregulatorene er hovedsaklig begrenset av klokkefrekvensen til systemet.

Posisjonsmålerens oppløsning bestemmer nøyaktigheten til de påførte spolestrømmene i forhold til det de ideellt sett burde være. I tilfellet der oppløsningen er lav, vil avviket mellom ønsket og faktisk magnetfeltorientering variere periodisk i større grad enn dersom en måler med høy oppløsning benyttes. Dette kan resultere i at rotoren får oversving og at systemet potensielt blir ustabil.

Inkrementell enkoder

En mye brukt sensor for fart- og posisjonsmåling innenfor motordrifter er en inkrementell enkoder. Prinsippet ved en slik sensor er å dele en motor-omdreining i flere deler, og sende en puls til kontrolleren for hvert inkrementelle steg. Typisk sender enkoderen firkantpuls-signaler på to kanaler, A og B, med et faseskifte på 90° . Faseskiftet blir utnyttet for å bestemme hvilken vei motoren spinner, hvor kanal A ligger foran i den ene retningen, vil kanal B ligge foran i motsatt retning. Ofte vil en enkoder i tillegg ha et indekssignal som sender ut en firkantpuls for hver hele omdreining. En representasjon for dette er vist i figur 2.18a. Ved å utnytte de ledende og følgende flankene på både kanal A og B, istedenfor bare logisk høy og logisk lav, vil man kunne oppnå fire ganger så høy oppløsning. Dette er kalt kvadraturdekoding og er vist i figur 2.18b. [12]



(a) Representasjon av signaler fra en enkoder

(b) 4x oppløsning ved å utnytte flankene

Figur 2.18: Representasjon av posisjonsmåling ved bruk av inkrementell enkoder

Hall-effect sensor

En annen sensor som er mye brukt for fart- og posisjonsmåling i motordrifter er Hall-effect sensoren. Denne baserer seg på prinsippet om Hall-effect, som går utpå at når det går strøm gjennom en leder i et magnetfelt, vil elektronene som flyter gjennom lederen bli påtrykt en kraft, som fører de mot en av sidene. Det oppstår da en spenningsforskjell mellom de to sidene av lederen, denne kraften blir kalt Lorentz-kraften. Hall-effect sensoren drar nytte av dette prinsippet ved å måle spenningsforskjellen mellom de to sidene av lederen. Dersom det er en spenningsforskjell, betyr det at lederen befinner seg i et magnetfelt. På denne måten kan man måle hver gang rotor passerer hall-effect sensoren og man kan fastslå posisjonen til rotor.

Det er vanlig å bruke en kombinasjon av tre hall-effect sensorer for å måle posisjonen til rotor. Disse er typisk plassert 120° fra hverandre. Når sensorene befinner seg i et magnetfelt vil de generere logisk 1 eller logisk 0, alt ettersom det er nord- eller sørpol som passerer. Rotorposisjonen vil da bli gitt for hver 60° -ende grad.

Fordeler med å bruke hall-effect sensorer er at de er veldig enkle komponenter, som gjør at det er billig. De er samtidig enkle å implementere. I tillegg blir de ikke påvirket av ytre faktorer som temperatur, støv eller vibrasjoner. Ulempen med disse sensorene er at posisjonsmåling ikke blir like nøyaktig som f.eks. ved bruk av en enkoder. Dette kan føre til større moment rippel i motoren. [13]

Utstyr og metode

3.1 Metode

Arbeidsmetodikk og samarbeid i gruppen er avgjørende for prosjektets gjennomføring og resultat. Metoder brukt for å gjennomføre et prosjekt av denne størrelsen legger grunnlaget for hele prosjektet. Gjennom god problemanalyse, planlegging og godt forarbeid sikret gruppen seg en god start på dette prosjektet. Jevnlige møter innad i gruppen, samt møter med veileder har ført til god kommunikasjon gjennom hele perioden. Ved å tidlig sette rammer for prosjektperioden i form av en forventningsavklaring, har gruppen løst utfordringer sammen på en god måte.

3.1.1 Planlegging og forarbeid

Etter å ha vært medlem i organisasjonen Revolve NTNU i omlag et halvt år, så gruppen et stort forbedringspotensiale i det nåværende styresystemet til den førerløse racerbilen. Det ble i samarbeid med teknisk leder derfor formulert en oppgave rundt dette systemet. Problemanalysen ble dermed opp til gruppen å utføre, og det var viktig å gjøre nødvendig forarbeid og planlegging.

For å lykkes i et prosjekt, er det viktig å først gjøre en grundig problemanalyse. Uten et klart overblikk over hvordan oppgaven skal utføres, kan uforutsette problemer raskt oppstå. Ettersom dette prosjektet strekker seg over flere fagområder og utfordret deltakerne faglig, var det spesielt viktig for gruppen å tidlig få en oversikt over prosjektets omfang. Flere av temaene hadde prosjektdeltakerne ingen forkunnskaper om, og det var derfor viktig å avdekke eventuelle utfordringer og hvilken retning oppgaven skulle ha.

På bakgrunn av dette ble det startet med undersøkelser i desember før prosjektstart. Det ble allerede da klart at oppgaven ville kreve mye informasjonsinnhenting og planlegging, hovedsaklig grunnet den korte tidshorizonten til prosjektet. Et forprosjekt ble derfor gjennomført ved prosjektstart for å innhente mer informasjon og planlegge gjennomførelsen.

Da oppgaven var definert, og gruppen hadde skaffet seg en god oversikt over problemet, ble kompetansen innad i gruppen kartlagt. Dette ble gjort for å effektivisere arbeidet, der- som enkelte i gruppen allerede hadde kjennskaper til spesifikke temaer. Gruppen fokuserte deretter på å delegere ansvarsområder, og sørget for at det ble allokert nok tid til de ulike arbeidsoppgavene. Dette ble gjort ved å utforme et *Gantt*-diagram, med en rekke tilhørende arbeidspakker. Her ble det estimert hvor mange timer hver prosjektdeltaker skulle bruke på de ulike arbeidspakkene, og det ble satt tidsfrister for de ulike delene av prosjektet.

Timefordelingen på hver av arbeidspakkene ble satt utifra erfaring fra vervet i Revolve NTNU, i tillegg til tidligere prosjekter utført ved NTNU. Det ble også forsøkt å ta hensyn til at prosjektdeltakerne måtte tilegne seg ny kunnskap innenfor de forskjellige temaene. Gruppen anerkjente imidlertid at enkelte arbeidspakker kunne ta lengre tid enn forventet, og at prosjektplanen kunne komme til å endre seg. Gruppen laget deretter et budsjett, ettersom Revolve NTNU skulle dekke kostnadene for dette prosjektet. Det ble sammenliknet kostnader med annen elektronikk i organisasjonen for å estimere disse.

3.1.2 Informasjonsinnhenting og dokumentasjon

Prosjektet gruppen har arbeidet med har krevd mye informasjonsinnhenting fra flere ulike kilder. Temaer som motorkontroll, elektronikk, kraftelektronikk, kretskonstruksjon og data-kommunikasjon krever mye teoretisk forståelse, og ved prosjektets start brukte gruppen derfor mye tid på å sette seg inn i relevant teori og verktøy som skulle benyttes. For å først forstå systemet som skulle designes, ble det brukt dokumentasjon skrevet av tidligere medlemmer i Revolve NTNU. Utover dette ble det hovedsaklig benyttet ulike vitenskapelige artikler, lærebøker og tidligere masteroppgaver for å sette seg inn i de ulike temaene.

En utfordring ved informasjoninnhenting har vært å plukke ut relevant informasjon. Med tanke på den korte tidsperioden, har ikke gruppen hatt mulighet til å sette seg dypt inn i alle temaer som oppgaven berører. Gruppen har derfor måtte luke ut nødvendig informasjon, og har vært kritisk til unødvendig tidsbruk av informasjonsinnhenting som ikke er direkte knyttet til prosjektet.

Ettersom mesteparten av informasjoninnhenting har blitt gjort på områder gruppen ikke hadde forkunnskaper om, har kildekritikk vært utfordrende. Gruppen har for det meste brukt vitenskapelige artikler og lærebøker for å skaffe seg teoretisk forståelse. Ved bruk av kilder har helthetsinntrykket avgjort om gruppen anser kilden som troverdig. Forfatters troverdighet, språkbruk, utgivelsesdato og med hvilken hensikt kilden har blitt skrevet, har vært de viktigste fokusområdene når gruppen har vurdert troverdighet til kilder. Gruppen har også vært i kontakt med ulike bedrifter, blant annet Maxon Group for å innhente informasjon og veiledning for deres utstyr.

3.1.3 Kvalitetssikring

Gruppen har gjennom hele prosjektet vektlagt kvalitetssikring. Dette har vært svært viktig for gruppen, ettersom det har blitt utviklet kretskort hvor det er mye tid som kan gå til spille dersom små feil ikke avdekkes. Gruppen har derfor brukt mye tid på å gi tilbakemelding på kretskortene som har blitt designet, før de har blitt sendt til produksjon. I tillegg har gruppen to ganger iløpet av prosjektet invitert både nåværende og tidligere medlemmer av Revolve NTNU for å gi tilbakemeldinger til gruppen. Hovedfokus i disse øktene har vært å avdekke skjematikk- og utleggsfeil. Ved å gjøre dette har gruppen fått nødvendige tilbakemeldinger på det utførte arbeidet, og hatt mulighet til å rette opp i dette. Før kretskortene har blitt sendt til produksjon, har gruppen kvalitetssikret produksjonsfilene ved å benytte programmet Gerbv. Ved å kvalitetssikre arbeidet på en slik måte har produksjonen blitt forsinket, men gruppen har vurdert at det har vært nødvendig for å lage en velfungerende prototype.

3.1.4 Kommunikasjon og samarbeid

Gjennom prosjektperioden har gruppen vært klar på å ha tydelig kommunikasjon. For å ha god kontroll over prosjektet, ble det tidlig fastslått at terskelen for å si ifra dersom deler av prosjektet ikke gikk som planlagt, skulle være lav. Det å stille spørsmål og diskutere i plenum har vært en avgjørende faktor som har økt effektiviteten innad i gruppen. I utgangspunktet hadde gruppen planer om å bruke Revolve NTNU sitt kontor som arbeidsplass. I forbindelse med nedstengning i januar, som følge av COVID-19, var ikke dette en mulighet. I en lengre periode var kontorene hos Revolve NTNU stengt, og gruppen jobbet derfor på hjemmekontor. Da gruppen fikk tilgang til Revolve NTNU sitt kontor etter nedsteningen, ble det etablert en skiftplan i organisasjonen med redusert kapasitet. Dette førte til vanskeligheter for gruppen å få til samarbeid, og gruppen endte derfor med å fortsette med delvis hjemmekontor, i tillegg til å samles på skolen tre dager i uken. Halvveis ut i prosjektet fikk gruppen tilgang til kontor i regi av NTNU, hvor gruppen har vært samlet hver dag, og fått til et godt samarbeid.

Møtevirksomhet

Jevnlige møter og felles arbeidstid er blant de viktigste faktorene for god kommunikasjon og et godt samarbeid i gruppen. Ukentlig har gruppen hatt et statusmøte der alle prosjektdeltakerne har presentert hva som har blitt gjort siden forrige møte, og hva som skal arbeides med videre. Møtene har også blitt benyttet til å ta opp diverse saker. Ved å ha møter jevnlig har terskelen vært lav for å ta opp usikkerheter og ytre bekymringer og utfordringer i prosjektet. Gruppen har sammen diskutert hvordan gå frem for å håndtere dette.

Gjennom prosjektperioden har det blitt avholdt jevnlig møter med veileder med to til tre ukers mellomrom. På møtene har gruppen presentert en statusoppdatering, hvor utført arbeid siden forrige veiledningsmøte har blitt fremvist. Gruppen har også brukt møtene for å

stille faglige og administrative spørsmål til veileder, i tillegg til å diskutere hvordan gruppen bør prioritere arbeidet. 1. mai gikk imidlertid gruppens veileder av med pensjon, og gruppen fikk en ny veileder de resterende ukene av prosjektperioden.

Konflikthåndtering

Konflikter i gruppen kan raskt ødelegge et godt samarbeid og hindre progresjon i prosjektet. I forkant av prosjektet hadde derfor gruppen en forventningsavklaring for å bevisstgjøre hverandre på hvilke forventninger som stilles til en selv, og gruppen. Forventningsavklaringen dekket viktige punkter som tilgjengelighet, møtevirksomhet, arbeidsmiljø, tidsfrister, arbeidstid og tilbakemeldinger. Denne er vist i vedlegg A.1. Gruppen ønsket med dette å forebygge eventuelle konflikter, ettersom det ble inngått en felles avtale for prosjektgjennomføringen.

Forventningsavklaringen har i retrospekt hjulpet gruppen med å unngå konflikter. Ved å blant annet ha blitt enige om at kritikk skal være konstruktiv og tilbakemeldinger skal gis direkte og fortløpende, har konflikter vært unngått. Gruppen har istedenfor hatt diskusjoner fortløpende, vært ærlige med hverandre og gitt hverandre konstruktive tilbakemeldinger.

3.2 Økonomi

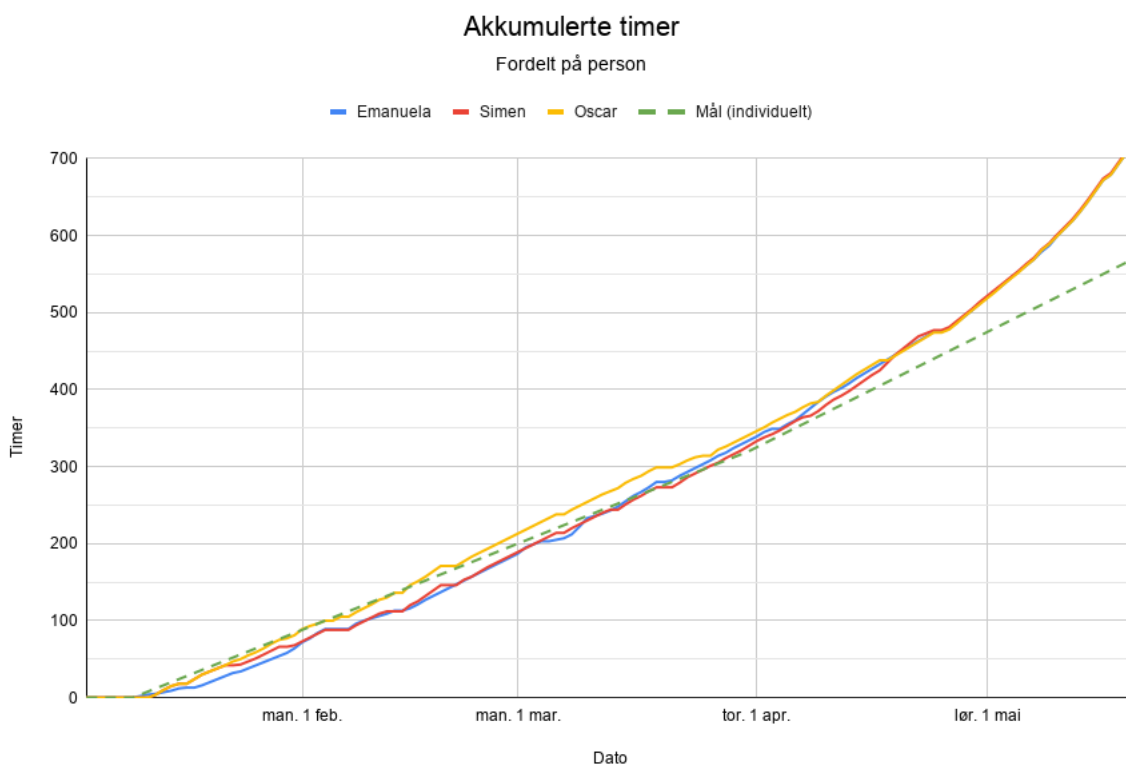
Ved prosjektstart satt gruppen opp et budsjett, og estimerte kostnadene for prosjektet. Alle kostnader var knyttet til produksjon av kretskortene. Av erfaring vet gruppen at komponenter lett kan bli ødelagt av blant annet kortslutninger eller ESD. Det ble derfor kjøpt reservekomponenter, i tilfelle noe ble ødelagt. Produsenten hvor kretskortene ble bestilt fra hadde et minste antall på fem kretskort per bestilling. Tabell 3.1 viser sluttkostnaden for prosjektet.

Post	Kroner [NOK]
Komponenter	2236.83 kr
Produksjon	1 051.47 kr
Totalt	3288.3 kr

Tabell 3.1: Sluttkostnad

3.3 Tidsforbruk

I prosjektmanualen tildelt før prosjektstart er det beskrevet at omfanget på bacheloroppgaven skal tilsvare 550-600 timer per student. Totalt tilsvare dette ca 1700 timer for gruppen, og dette var dermed et naturlig mål å sette. Underveis i prosjektet har prosjektdeltakerne ført timer, hvor totalt akkumulerte timer endte på 2135 timer ved prosjektslutt. Figur 3.1 viser akkumulerte timer per prosjektdeltaker.



Figur 3.1: Timeforbruk per prosjektdeltaker

Gruppen har siden prosjektstart holdt en jevn arbeidsflyt frem til slutfasen. Gjennom prosjektperioden har gruppemedlemmene også hatt et verv i Revolve NTNU som har krevd mange arbeidstimer. Mot slutten av prosjektet var det imidlertid fortsatt mye arbeid som gjenstod, og det ble dermed brukt mer tid enn planlagt for å ferdigstille prosjektet.

3.4 Utstyr og ressurser

Gjennom prosjektperioden har gruppen benyttet ulike ressurser for utvikling og prosjektstyring. Det har også blitt brukt mye ulikt utstyr i forbindelse med produksjon og testing av kretskort.

3.4.1 Prosjekteringsressurser

Git

Git er et versjonskontrollsystem for å holde kontroll på filendringer ved utvikling. I prosjektet har Git blitt brukt mye, da det legger til rette for flere filversjoner gjennom utviklingsprosessen. Ved modellering i Simulink har det blant annet blitt brukt Git for å holde oversikt over ulike modelliterasjoner, i tillegg til at delsystemer kan arbeides på av flere gruppemedlemmer samtidig, uten kollisjoner.

Altium 365

Altium 365 er en skytjeneste av Altium Designer for skylagring av skjematikk, utlegg og andre filer i forbindelse med design og produksjon av kretskort. Tjenesten har blitt brukt hyppig for sikkerhetskopiering og for å legge til rette for å arbeide på samme fil fra ulike datamaskiner.

Microsoft Teams

Microsoft Teams har blitt mye brukt for å samle informasjon, arbeid og resultater underveis i prosjektet. Gruppen har laget en arbeidsmappe, hvor prosjektdeltakerne har lagret arbeid og ressurser. Microsoft Teams har også blitt brukt som møteplattform ved digitale veiledningsmøter.

Overleaf

Overleaf er en web-basert LaTeX-editor som har blitt brukt for rapportskriving i prosjektet.

Confluence

Confluence fungerer i hovedsak som en organisasjons-wiki. Revolve NTNU benytter denne plattformen til å dokumentere arbeid og annen relevant informasjon for organisasjonen. Gruppen har gjennom prosjektperioden derfor også brukt Confluence for å dokumentere arbeid, i tillegg til referater fra interne møter.

Slack

Slack er et kommunikasjonsprogram for meldinger og deling av mindre filer. Gjennom prosjektet har Slack blitt brukt for daglig kommunikasjon.

3.4.2 Motor og gir

I prosjektet har flere systemer blitt dimensjonert ut ifra merkeverdiene til det nåværende styresystemet i Revolve NTNUs førerløse racerbil. Styresystemet består av en Maxon EC-i52 180W BLDC-motor og et planetgir GP 52 C fra Maxon Group. I tabell 3.2 finnes merkeverdiene fra motoren og giret. De fleste verdiene er hentet fra databladet til motoren i vedlegg A.2 og girets datablad i vedlegg A.3. Enkelte verdier var det imidlertid nødvendig å kontakte Maxon Group for å anskaffe.

Motor data	Symbol	Verdi	Enhet
Nominell spenning	U_{nom}	24	V
Nominell strøm	I_{nom}	8.81	A
Nominelt moment	T_{nom}	428	mNm
Momentkonstant	T_c	48.6	$\frac{mNm}{A}$
Motstand (fase-til-fase)	R_{pp}	0.0894	Ω
Induktans (fase-til-fase)	L_{pp}	0.122	mH
Maks effektivitet	η	90	%
Antall polpar	n	8	-
Statisk friksjonskoeffisient (motor)	T_f	17	mNm
Viskøs friksjonskoeffisient (motor)	d_m	0.387	$\frac{\mu Nm}{\frac{rad}{s}}$
Termisk tidskonstant for spolevikling	t_c	10.2	s
Rotors treghetsmoment	J_m	170	gcm^2
Girreduksjon	R_G	66:1	-
Nominelt moment på gir	T_{Gnom}	30	Nm
Maksimalt moment på gir	T_{Gmax}	45	Nm
Girets treghetsmoment	J_G	16.7	gcm^2

Tabell 3.2: Systemspesifikasjoner til Maxon EC-i52 og gir

3.4.3 Programvare

Atmel Studio

Atmel Studio er en integrert utviklingsplattform fra Microchip. Programvaren støtter utvikling og feilsøking av alle AVR- og SAM-mikrokontrollerprogrammer. Atmel Studio gir muligheten til å skrive, bygge og feilsøke programmer som er skrevet i C/C++ eller assembly. Gruppen har brukt denne programvaren til å skrive kode for mikrokontrolleren brukt i prosjektet.

Altium Designer

Altium Designer er et verktøy til tegning av skjematikk for og utlegg for kretskort. I Altium Designer er det lagt til rette for å importere komponenter slik at man kan se kretskortet i 3D.

Gerbv

Gerbv er en programvare laget for å vise filer av kretskort i gerber-format. Gruppen har brukt programvaren som et verktøy for å se over produksjonsfilene, før de har blitt sendt til produksjon. I Gerbv er det enkelt å se de forskjellige lagene på et kretskort og oppdage eventuelle feil.

Matlab

Matlab er et matematikkverktøy for numerisk utregning av ulike problemer. I prosjektet er Matlab brukt sammen med Simulink for å modellere og simulere systemet.

Simulink

Simulink er et simuleringsverktøy for ikke-lineære systemer. I prosjektet har gruppen brukt programmet til å simulere reguleringsystemet og analysere dette. Programmet har også blitt brukt til å generere C-kode.

3.4.4 Testutstyr

Til produksjon og testing av kretskort har det blitt brukt ulikt utstyr som vist i tabell 3.3.

Beskrivelse	Produsent	Til bruk for
Multimeter	Fluke	Testing av kretskort
Oscilloskop	Keysight	Testing av kretskort
Strømforsyning	PeakTech	Testing av kretskort
Loddeutstyr	Weller	Produksjon og testing
Mikroskop	Photonic Optics	Produksjon og testing
ESD-lenke	RS Components	Produksjon og testing
ESD-matte	RS Components	Produksjon og testing
Atmel ICE	Microchip	Programmering

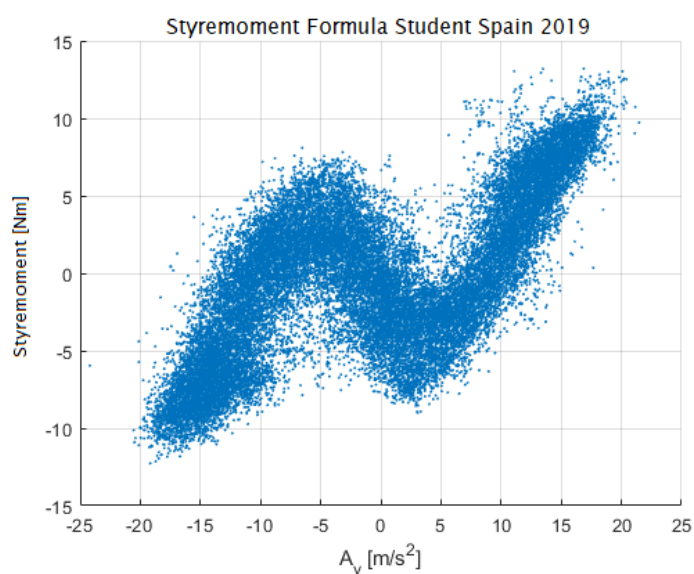
Tabell 3.3: Testutstyr

Arbeid og resultater

4.1 Systemspesifikasjon

For å utvikle elektronikk til et system må det først spesifiseres hva systemet skal oppnå og hvilke faktorer som er avgjørende for design. Av den grunn har det blitt fastsatt en rekke systemspesifikasjoner som må tas hensyn til.

Et av de viktigste kravene for design av et kretskort med kraftelektronikk er at det tåler tilstrekkelig strømgjennomgang. Basert på tidligere analyse av styremoment fra konkurransen Formula Student Spain fra 2019, vist i figur 4.1, kan man se at føreren maksimalt trengte å påføre like under 15 Nm. Med utgangspunkt i dette består derfor det nåværende autonome styresystemet til Revolve NTNU av en Maxon EC-i52 motor, med en girreduksjon på 66:1. Valget av motor og gir ble tatt basert på grundig analyse av behov, og det har blitt testet og validert at systemet leverer tilstrekkelig moment. Det ble derfor tatt et valg om å dimensjonere elektronikken i dette prosjektet med utgangspunkt i det nåværende styresystemet.



Figur 4.1: Styremoment Formula Student Spain 2019

Beregning av maksimal strømgjennomgang

Ifølge Maxon Group kan deres motorer overbelastes over det nominelle momentet i en tidsperiode som tilsvarer den termiske tidskonstanten for spoleviklingene [14]. Dette er aktuelt i applikasjoner der det er stor variasjon i ønsket moment, slik som dette bruksområdet. Ved å ta utgangspunkt i det maksimale momentet giret tåler, kan det regnes ut hvor mye moment motoren maksimalt kan produsere uten å overbelaste giret. Med denne informasjonen kan man benytte momentkonstanten oppgitt i tabell 3.2 for å beregne den maksimale strømmen motoren kan påtrykkes. Utrekning av maksimalt moment og maksimal strøm er vist i likning 4.1 og 4.2.

$$T_{max} = \frac{T_{Gmax}}{R_G} = \frac{45Nm}{66} = 681.8mNm \quad (4.1)$$

$$I_{max} = \frac{T_{max}}{T_c} = \frac{681.8mNm}{48.6\frac{mNm}{A}} = 14.0A \quad (4.2)$$

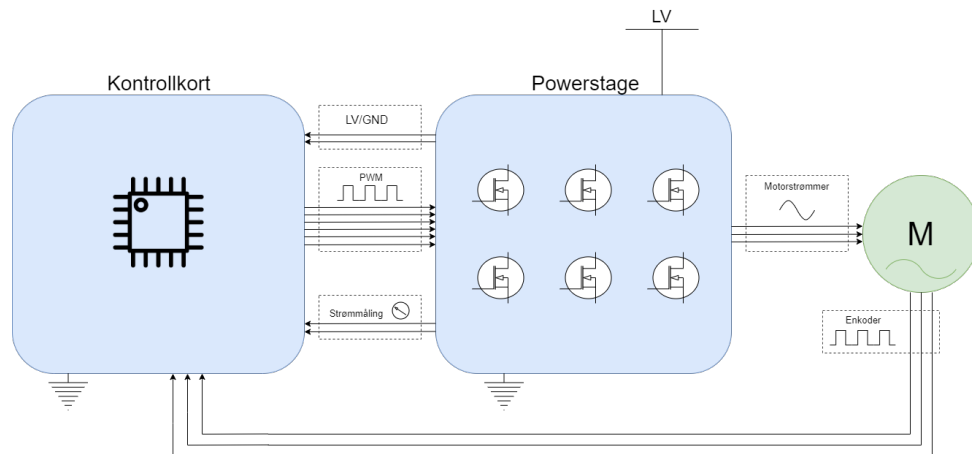
Forsyningsspenning

Forsyningsspenningen i racerbilene til Revolve NTNU har variert fra år til år mellom 24V og 37V. I år er alle kretskortene i organisasjonen dimensjonert for å tåle en forsyningsspenning på 37V, og det vil derfor være naturlig å dimensjonere systemet i dette prosjektet for den samme forsyningsspenningen.

4.2 Hardware design

Da det elektriske systemet ble designet, la gruppen stor vekt på det å kunne teste systemet så godt som mulig. Valget ble derfor å splitte det elektriske systemet til to kretskort. Årsaken til denne oppdeling kommer av at systemet designes fra bunn av og ikke itereres som andre elektriske systemer i Revolve NTNU. Ved å dele opp systemet er det lettere for gruppen å teste og verifisere de ulike funksjonalitetene på gruppens designvalg. Figur 4.2 illustrerer oppdelingen av systemet, der de blå boksene illustrerer kretskortene og den grønne sirkelen representerer motoren. Heretter blir kretskortene kalt kontrollkort og powerstage.

I dette delkapittelet vil rapporten ta for seg hvordan gruppen har designet hardware, og testresultater vil bli presentert.



Figur 4.2: Illustrasjon av det elektriske systemet

Ved design av hardware har gruppen brukt designverktøyene Altium Designer og Gerbv. Altium har blitt brukt til å designe skjematikk og utlegg. Gerbv har blitt brukt til å validere produksjonsfiler før de ble sendt til produksjon. Mønsterkortene er produsert hos JLCPCB og komponentene på kretskortene er kjøpt fra leverandører som Revolve NTNU bruker.

4.2.1 Spesifikasjoner for mønsterkort

Før gruppen kan starte å designe kretskort må enkelte spesifikasjoner bli satt. Kortene som komponentene plasseres på kalles mønsterkort. Disse kortene er ofte laget av plastlaminat der kobberfolie ligger lagvis i kortet. På mønsterkort ligger ledningerbaner som forbinder komponenter. I dette delkapittelet vil det bli gått inn på valg av kobbertykkelse, lag og ledningsbaner til powerstage og kontrollkortet.

Kobbertykkelse

Kobbertykkelse er en av de viktigste faktorene ved design av kretskort og er definert i oz/ft^2 , der $1 oz/ft^2$ er det vanligste. Hvor mye strøm som kan flyte gjennom kretskortet avhenger av tykkelsen på kobberbanene. Kontrollkortet er designet med $1 oz/ft^2$ og powerstage med $2 oz/ft^2$. Årsaken til dette er at powerstage skal tåle høyere strømgjennomgang.

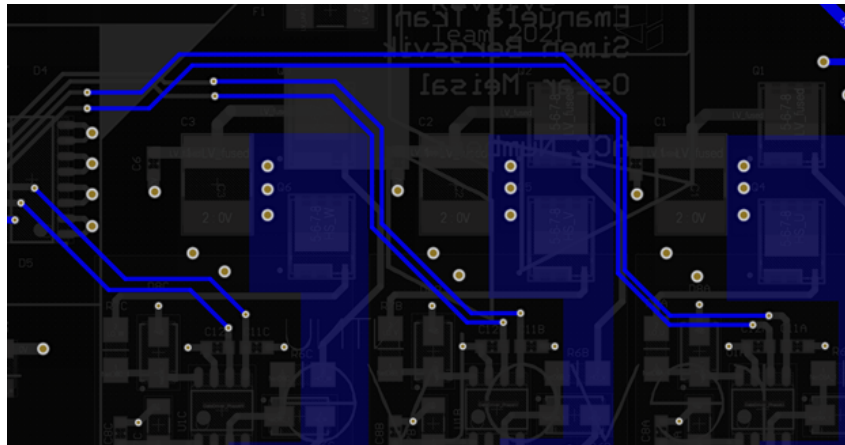
Lag

Det finnes to typer hovedlag på alle kretskort, interne og eksterne lag. Begge lagene består av kobber, men komponenter kan kun plasseres på de eksterne lagene. Både kontrollkortet og powerstagekortet består av fire lag: Jordplan, strømforsyningsplan, toppside og bunnside.

Ledningsbaner

Ledningsbaner blir brukt for å koble sammen to eller flere punkter på et kretskort. Ved design av kretskort bestemmes bredden på banene og klareringen mellom de. Tynne baner kan skape varmeutvikling og kan potensielt ødelegge kretskortet eller komponentene på det. På grunn av dette har gruppen brukt kobberplan på enkelte steder på kretskortet der varmeutvikling kan oppstå. I figur 4.3 kan man se at kobberplan har blitt brukt mellom MOSFETene for å spre varmeutviklingen.

Klareringen mellom ledningsbanene bestemmes utifra hvilke signaler det gjelder. Da ruting av ledningsbanene ble gjort, var fokuset på å holde ledningsbanene korte og unngå at topp- og bunnside ledningsbaner krysser hverandre. Det har ikke alltid vært mulig å oppfylle dette, så prioriteringen har derfor vært på ledningsbaner med sensitive og kritiske signaler, som for eksempel målinger og PWM-signaler.



Figur 4.3: Utklipp av powerstage-kortet som illustrerer bruken av kobberplan og ledningsbanene til de seks PWM-signalene som går inn på gatedriverene

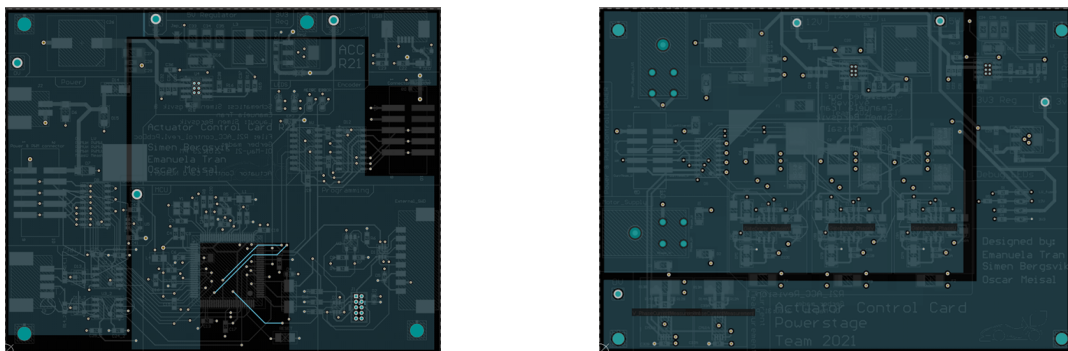
4.2.2 Elektromagnetisk forstyrrelse og elektrostatisk utladning

Ved design av kretskortene har det blitt tatt hensyn til både elektromagnetisk forstyrrelse og elektrostatisk utladning. EMI og ESD er faktorer som kan påvirke det elektriske systemet ved at det kan forstyrre signaler og potensielt ødelegge kortet. Videre vil det bli gått inn på hvordan gruppen har lagt til rette for å unngå dette.

EMI

For å unngå elektromagnetisk forstyrrelse har plassering av ledningsbaner blitt nøye gjennomtenkt. Sensitive signaler, som for eksempel PWM-signaler, er plassert så langt unna transistorkretsen som mulig, ettersom transistorene genererer støy ved svitsjing. Figur 4.3 på forrige side, viser at de seks PWM-signalene er rutet rundt transistorene og den tilhørende kretsen.

Hvordan spenningsplanene er formet har mye å si for den elektromagnetiske forstyrrelsen. Gruppen har valgt å skille spenningsplan og tatt hensyn til eventuelle strømsløyfer som kan oppstå. Figur 4.4 viser hvordan spenningsplanene er formet på kontrollkortet og powerstage.



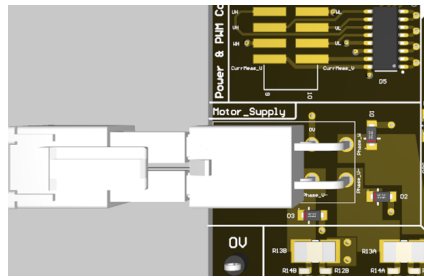
(a) Spenningsplan for kontrollkort som separerer 5V og 3.3V (b) Spenningsplan for powerstage som separerer 12V og 3.3V

Figur 4.4: Spenningsplan for kontrollkort og powerstage

ESD

Elektrostatisk utladning forekommer ved at uønskede strømmer blir påført et system, ofte gjennom konnektorer eller ved direkte berøring av mennesker. Mennesker bygger lett opp elektrostatisk spenning og under hele produksjonsfasen har gruppen tatt i bruk ESD-utstyr.

En ekstra beskyttelse mot ESD er TVS-dioder. Disse har blitt plassert mellom ESD-utsatte punkter og sensitive komponenter. Diodene har en spenningsrate, hvor høyere spenning enn denne vil føre til at linjen blir dratt mot 0V. Dette fører til at dioden beskytter komponenter mot høyere spenning enn de er dimensjonert for. TVS-diodene plasseres med fordel nært komponenter og konnektorer, for å raskt kunne reagere. Figur 4.5 illustrer plassering av TVS-diodene D1, D2 og D3 rundt konnektoren.



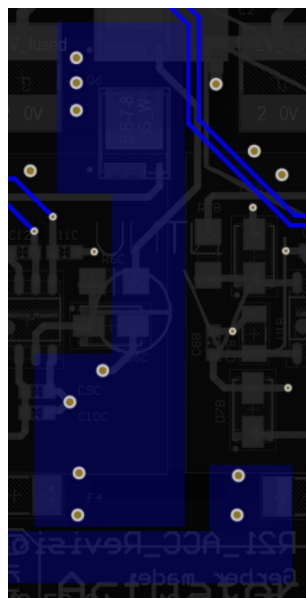
Figur 4.5: Utklipp av powerstage som illustrerer TVS diodene, D1, D2 og D3, som er plassert nært konnektoren til motoren

4.2.3 Effektiv plassering av komponenter

For at kretskort skal fungere så optimalt som mulig, har plassering av komponenter mye å si. Effektiv plassering av komponenter kan spare mye tid ved lodding og feilsøking. Ved utlegg av mønsterkort er det viktig å tenke på strømveien for de ulike komponentene og hvordan komponenter blir lagt ut i forhold til hverandre.

Powerstage har komponenter som kan skape mye støy ved svitsjing. Komponenter har utelukkende blitt plassert på toppsiden av kretskortet, ettersom det da vil være lettere å probe og feilsøke. Komponentene som tilhører vekselretter-kretsen er plassert samlet for å ikke forstyrre andre komponenter og spenningsplan, som for eksempel strømmålingskretsen.

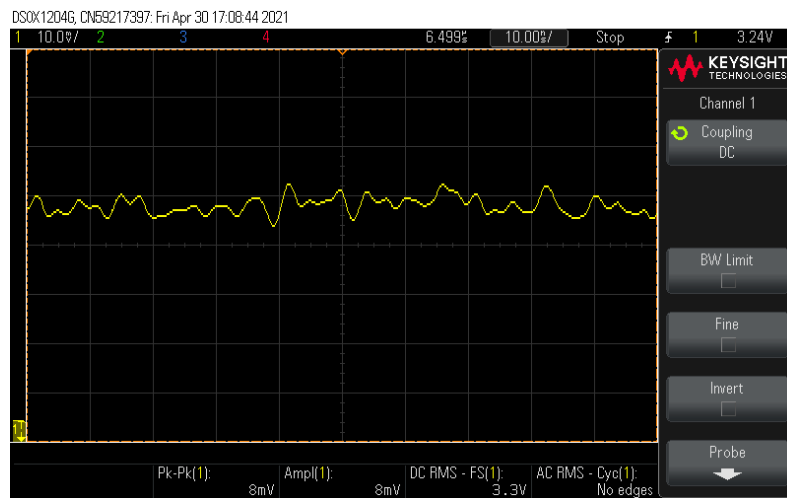
Ved applikasjoner med høyt strømtrekk kan det være lurt å plassere viaer for økt varmespredning. Varmeutvikling er noe som også bør tas i betraktning når komponenter plasseres. Figur 4.6 viser hvordan viaer har blitt brukt til å lede unna varme fra vekselretter-kretsen på powerstage.



Figur 4.6: Plassering av viaer på powerstage for å lede unna varme

4.2.4 Testpunkter

Testpunkter er svært nyttig ved feilsøking og testing av kretskort. For hvert spenningsplan er det derfor lagt til testpunkter. I tillegg til dette er det lagt til 0Ω -motstander etter hver spenningsregulator. Ved lodding og testing plasseres 0Ω -motstandene først etter at spenningsregulatoren gir ut en stabil spenning. Stabil spenning defineres i Revolve NTNU som $\pm 5\%$ av nominell spenning. Dette er for å forsikre at riktig spenning blir forsynt videre til resten av kretskortet, og det minker sannsynligheten for å ødelegge sensitive og kritiske komponenter. Figur 4.7 viser probing av spenningsregulatoren for $3.3V$ med en rippel på $8mV$ fra topp til bunn. Dette tilsvarer 0.2% av den nominelle spenningen.



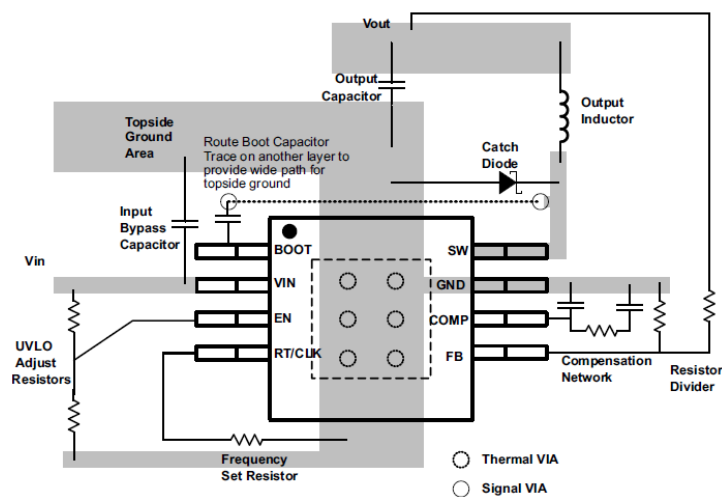
Figur 4.7: Probing av rippel på $3.3V$ -regulator

I tillegg til 0Ω -motstander, er LEDs lagt til for å indikere om de ulike spenningsregulatorene fungerer. For eksempel er det LEDs for $5V$ og $3.3V$ for å indikere at spenningsregulatorene fungerer.

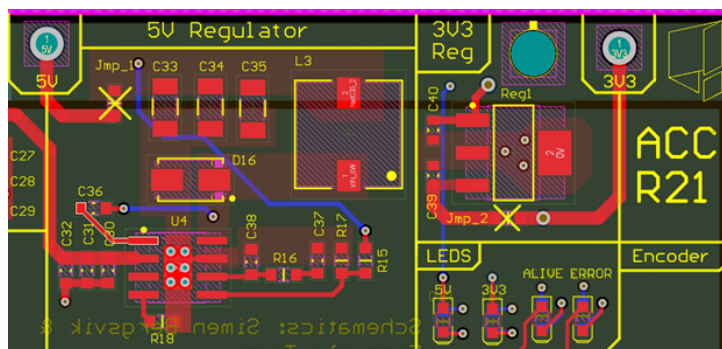
4.2.5 Spenningsforsyning

Spenningsregulatorene som brukes er TPS5456-buckregulatorer [15] og LT1129-3.3-lineærregulatorer [16] fra Texas Instruments. Spenningsregulatorene brukes i Revolve NTNU og er designet etter spesifikasjoner fra datablad. Buckregulatoren brukes til å regulere spenningsforsyningen ned til 12V eller 5V, mens lineærregulatoren brukes til regulere 5V ned til 3.3V. Vedlegg A.5 og A.10 viser skjematikk for strømforsyningen til powerstage og kontrollkortet.

Utleppet for strømforsyningen til de forskjellige spenningsregulatorene er basert på referansedesignet gitt i databladet. Figur 4.8 viser referansedesign til den brukte buckregulatoren. I tillegg har gruppen tatt hensyn til ledningsbaner og plassert kretsene med hensyn på punktene nevnt i delkapittelene 4.2.1 og 4.2.3. Figur 4.9 viser utlegget på kontrollkortet der man kan se hvordan spenningsregulatorene for 5V og 3.3V har blitt rutet.

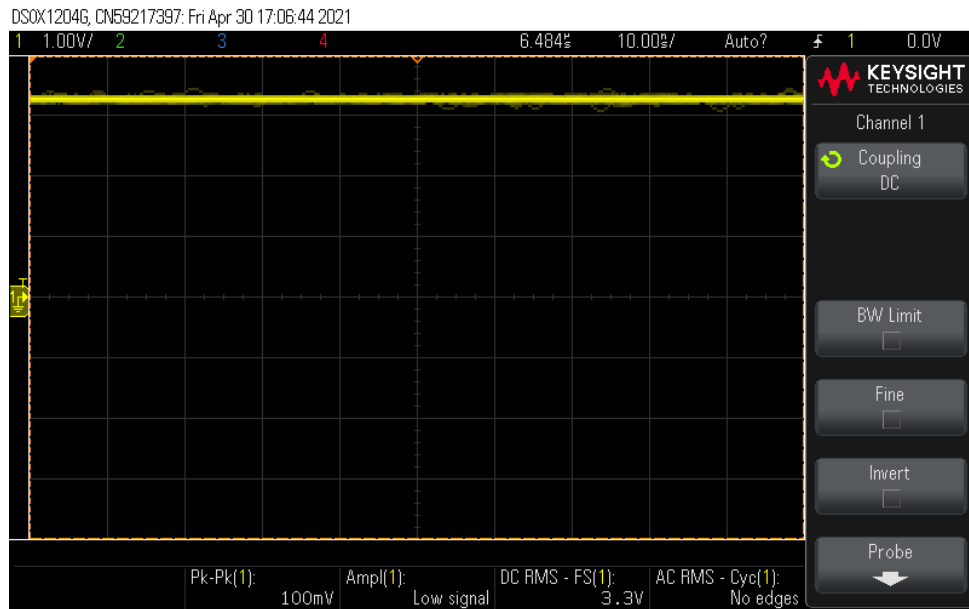


Figur 4.8: Referanseutlegg fra databladet til TPS5456

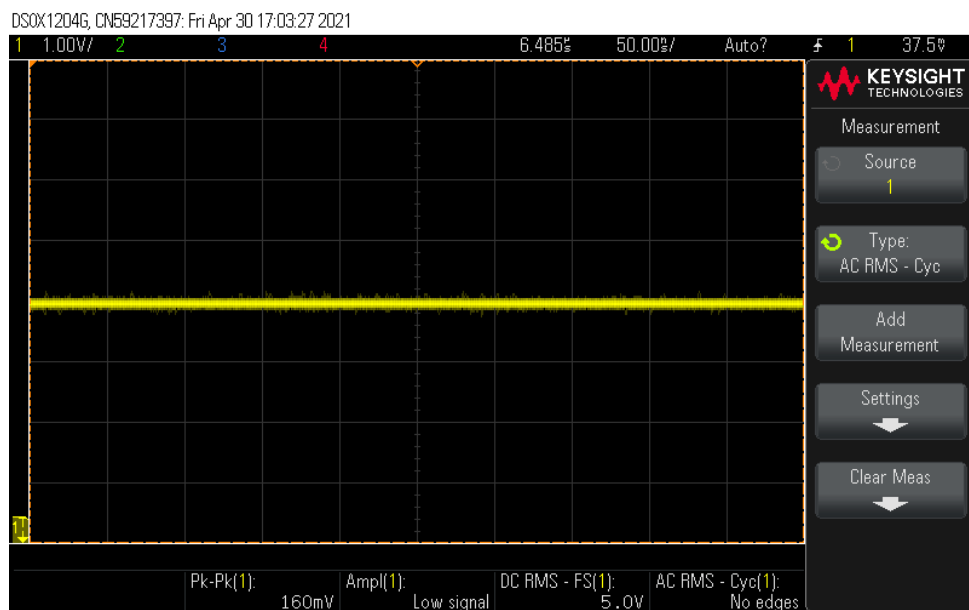


Figur 4.9: Utlepp for spenningsregulatorene for 5V og 3.3V fra kontrollkortet

Som nevnt i delkapittel 4.2.4 om testpunkter, blir spenningen ut av spenningsregulatorerne målt før 0Ω-motstandene loddes på. Spenningen blir målt ved hjelp av et oscilloskop og probes på loddelandet til 0Ω-motstanden. Figurene 4.10 og 4.11 viser utgangen fra spenningsregulatorerne, der man kan observere en stabil 3.3V og 5V spenning. Hvis spenningsregulatorerne gir ut en stabil spenning kan kortet loddet videre.



Figur 4.10: Testresultat av 3.3V



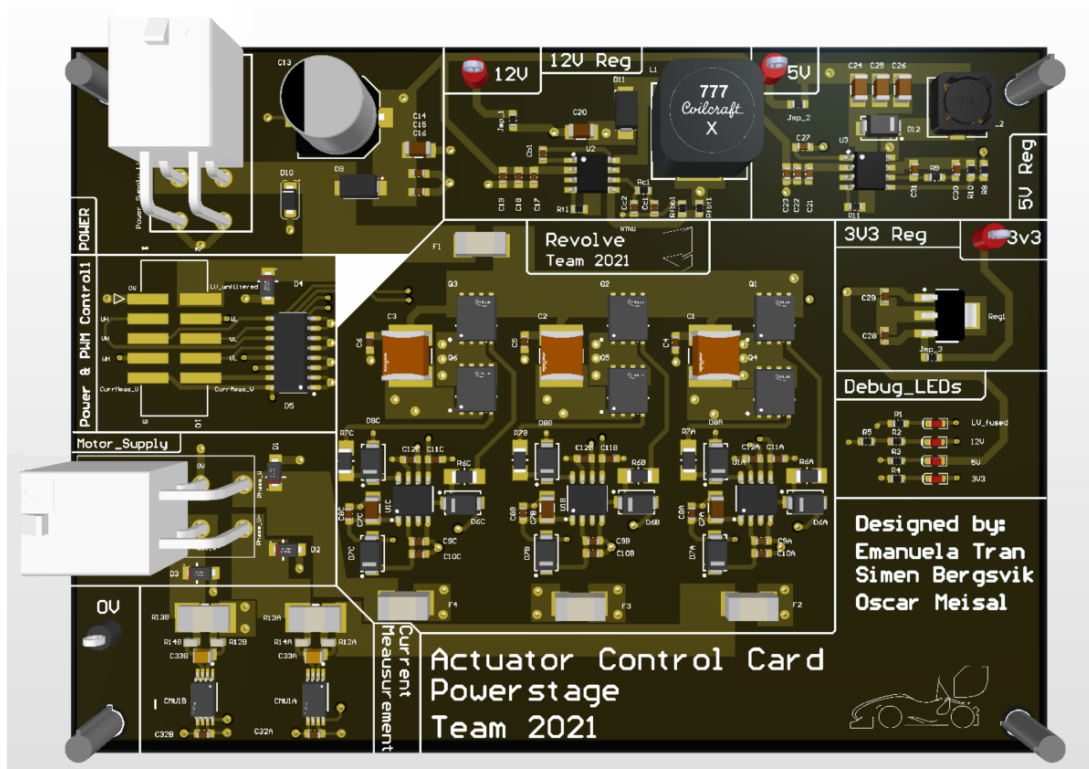
Figur 4.11: Testresultat av 5V

4.3 Hardware design powerstage

Hovedfunksjonen til powerstage er å omgjøre likespenning til trefase vekselspanning, basert på seks PWM-signaler fra kontrollkortet. Kretskortet har i tillegg en strømmålingskrets, som måler de ulike fasestrømmene. Kortet kan deles inn i tre ulike deler: strømforsyning, trefase vekselretter og strømmåling.

Kretskortet er designet til å tåle en inngangsspenning på 37V og tar i bruk spenningsregulatorer til å lage 12V og 3.3V spenninger. I vekselretterkretsen påfører gatedriveren et PWM-signal på 12V på gate-terminalen til transistorene, slik at disse veksler mellom å være på eller av. Strømmålingskretsen måler fasestrømmene for fase *U* og *V*, ved å måle spenningsfallet over en shunt-motstand. Figur 4.12 viser en gjengivelse av kortet.

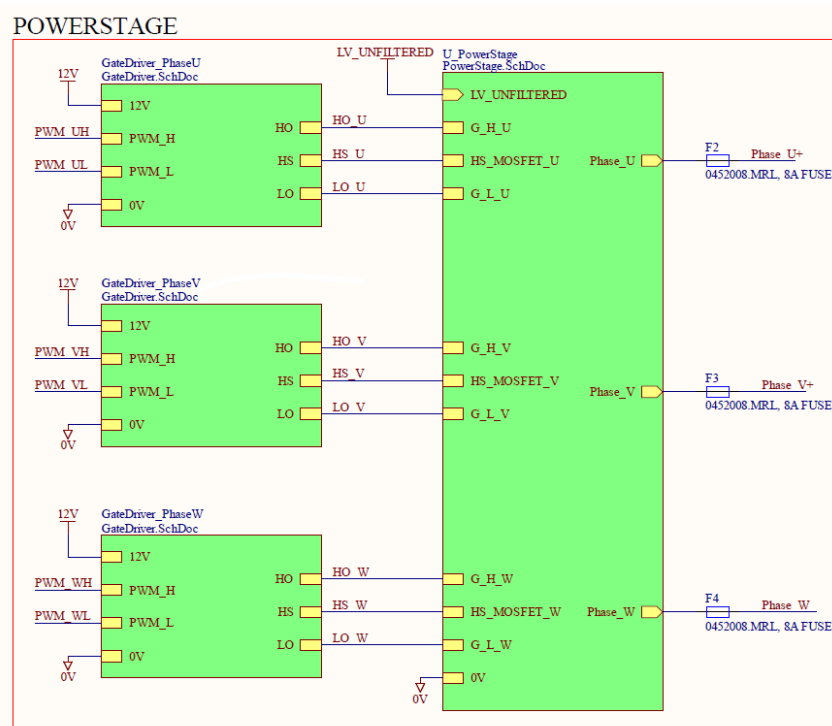
Videre i dette delkapittelet vil rapporten ta for seg hvordan gruppen har gått frem for å designe de forskjellige delene av kretskortet. Først vil delkapittelet ta for seg trefase vekselretteren og hvordan den er bygd opp, med blant annet valg av transistor, gatedriver og design av bootstrapkrets. I tillegg vil det bli gått inn på hvordan gruppen har gått frem ved utlegg og plassering av de forskjellige kretsene. Til slutt vil det bli gått inn på hvordan strømmålingskretsen er designet.



Figur 4.12: Gjengivelse av powerstage

4.3.1 Trefase vekselretter

Vekselretterkretsen består av tre halv-broer, der hver halv-bro består av to transistorer. Den øverste og nederste transistoren kalles henholdsvis toppside- og bunnside transistor. I hver halv-bro er det en gatedriver som tar inn to PWM-signaler og veksler mellom å påtrykke 12V og 0V på toppside- og bunnside transistoren. I tillegg til gatedriverene er det lagt til bootstrapkretser ved hver halv-bro, for å lade opp kondensatoren som påtrykker spenning på toppside transistorene. Figur 4.13 viser et utklipp av den overordnede skjematikken for vekselretteren. Her er det vist tre identiske gatedriverkretser, som styrer de seks transistorene på kortet. Inni blokken *U_Powerstage* finnes de seks transistorene som genererer fasene *U*, *V* og *W*.



Figur 4.13: Utklipp av overordnet skjematikk for vekselretteren

For hver av fasene kan man se fra skjematikken at det er lagt til sikringer. Disse er lagt til for å beskytte motoren, dersom feil skulle oppstå. For eksempel vil sikringen beskytte mot kortslutning mellom to av fasene.

Vedlegg A.8 viser skjematikk for hver halvbro, og viser hvordan toppside- og bunnside transistorene er koblet opp.

Som nevnt i delkapittel 2.2.2 kan svitsjing av transistorene skape uønskede transienter. På grunn av dette er det lagt til en snubberkrets som også fungerer som en DC-link-kondensator for å dempe transientene og for å unngå minst mulig spenningsrippel.

4.3.2 Valg av transistor

Etter grundig undersøkelser ble det valgt å ta i bruk transistorer av typen MOSFET, da denne typen felteffekttransistor fungerer godt som en svitsj som nevnt i delkapittel 2.2.1. Gruppen undersøkte flere ulike felteffekttransistorer og sammenlignet de ut i fra følgende parametre:

- Maks spenning inn på drain-source V_{dss}
- Maks kontinuerlig strøm inn på drain-inngangen I_d ved 25°C
- Internmotstand i transistoren $r_{DS(on)}$
- Tidsforsinkelsen det tar å skru av og på transistoren t_f , t_r , $t_{d(on)}$, $t_{d(off)}$
- Kostnad per transistor

Tabell 4.1 viser en sammenligning av de ulike felteffekttransistorene som gruppen vurderte. Valget endte opp med MOSFETen BSC016N06NS [17] ettersom denne hadde lavest indremotstand og en total tidsforsinkelse som var relativt lav sammenlignet med de andre transistorene.

MOSFET	V_{DSS}	I_d (25°C)	$r_{DS(on)}$	t_f	t_r	$t_{d(on)}$	$t_{d(off)}$	Pris
IRLR2908PBF[18]	80 V	30 A	28 mΩ	55 ns	95 ns	12 ns	36 ns	10.95 kr
FDPF045N10A[19]	100 V	67 A	4.5 mΩ	15 ns	26 ns	23 ns	50 ns	31.19 kr
IRLZ44PBF[20]	60 V	50 A	28 mΩ	110 ns	230 ns	17 ns	42 ns	19.51 kr
SQJA80EP[21]	80 V	60 A	5.8 mΩ	16 ns	5 ns	16 ns	31 ns	10.75 kr
BSC016N06NS	60 V	100 A	1.4 mΩ	9 ns	9 ns	19 ns	35 ns	19.45 kr

Tabell 4.1: Sammenligning av forskjellige felteffekttransistorer

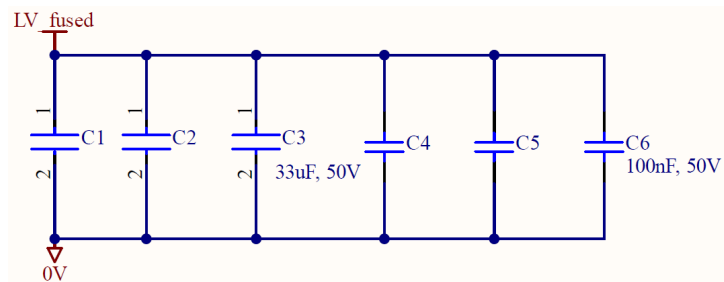
4.3.3 Valg av DC-link-kondensatorer

Ved svitsjing av transistorer kan det oppstå ringing og transienter. Av den grunn er det lagt til en kondensatorbank for å dempe dette. Kondensatorbanken består av tre kondensatorer i parallell på 33μF og tre avkoblings-kondensatorer på 100nF. Den minste verdien kondensatorbanken kan ha, er vist i likning 4.3 ved å bruke formel 2.2 fra delkapittel 2.2.2.

$$C_{link} \geq \frac{0.2165 I_m T_s}{\Delta U_{dc_max}} = \frac{0.2165 \cdot \sqrt{2} \cdot 20A \cdot \frac{1}{50kHz}}{0.1 \cdot 24V} = 51\mu F \quad (4.3)$$

I_m er bestemt ut i fra amplituden til den høyeste strømmen som kan påtrykkes motoren I_{max} fra likning 4.2 i delkapittel 4.1, med en sikkerhetsmargin på cirka 50%. T_s er svitsjefrekvensen til vekselretteren og ΔU_{dc_max} er rippelen i spenningsforsyningen. I dette tilfellet regnes det med 10% rippel ved 24V, som gir størst verdi på C_{link} . For å være på den sikre siden ble kondensatorbanken bevisst overdimensjonert og satt til å være 99μF.

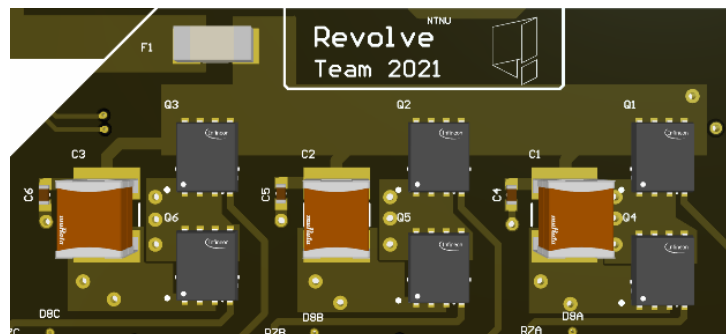
Figur 4.14 viser et utklipp fra skjematikken hvor man kan se kondensatorbanken med tre DC-link kondensatorer og tre tilhørende avkoblingskondensatorer.



Figur 4.14: Utklipp fra skjematikk av kondensatorbank

Kondensatorene som er brukt ble valgt på bakgrunn av nettspenning og kostnad. I tillegg ble disse valgt utifra standard kondensatorstørrelse i Revolve NTNU. Årsaken til at kondensatorbanken ble delt opp var fordi gruppen ønsket å spre kondensatorene rundt de tre transistorbenene for å dempe uønskede transienter.

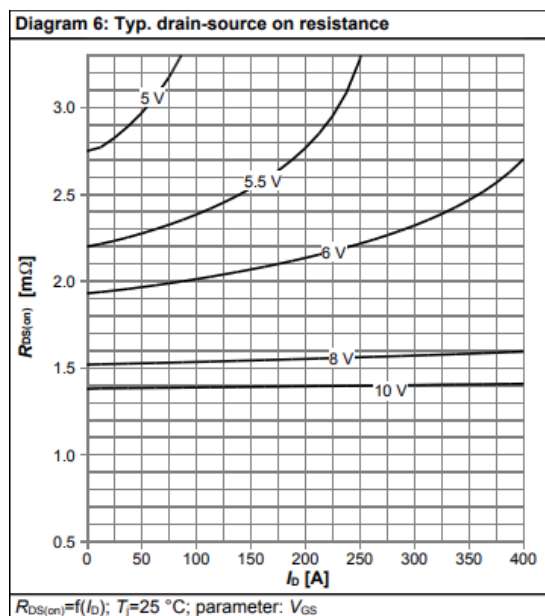
I utlegget har DC-link-kondensatorene blitt plassert mellom toppside og bunnside transistorene, hvor hvert ben har en DC-link kondensator og en avkoblingskondensator. Figur 4.15 viser hvordan gruppen har valgt å plassere kondensatorbanken i forhold til transistorene.



Figur 4.15: Utklipp av plassering av kondensatorbank i forhold til transistorene

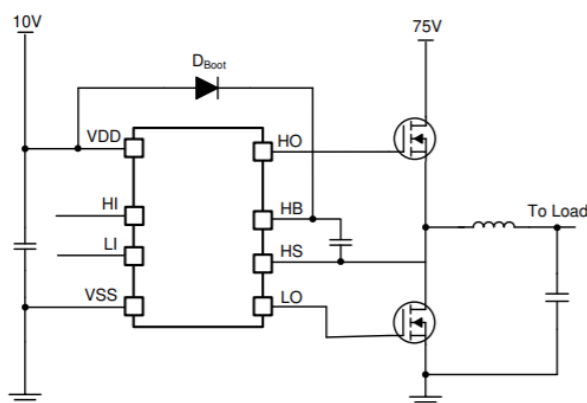
4.3.4 Valg av gatedriver

Ved å se på karakteristikken til transistoren i figur 4.16, ser man at lavere påtrykt gate-spenning vil resultere i større internmotstand $r_{DS(on)}$. Det er ønskelig at internmotstanden er så lav som mulig, da dette reduserer svitsjetap og varmeutvikling. Gruppen har derfor valgt å påtrykke transistoren 12V, da dette vil resultere i en liten internmotstand for den valgte transistoren.



Figur 4.16: Diagram for elektrisk karakteristikk av drain-source i forhold til $r_{DS(on)}$

Den valgte gatedriveren er en to-kanals halv-bro gatedriver, UCC27288 [22]. Figur 4.17 illustrerer en forenklet gatedriver som styrer to transistorer ved hjelp av en ekstern bootstrapkondensator og bootstrapdiode.



Figur 4.17: Forenklet applikasjon av gatedriver UCC27288

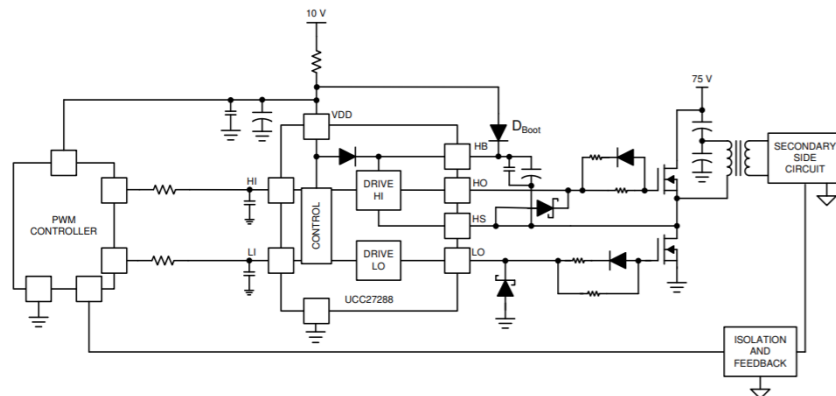
Gatedriveren består av åtte pinner, der pinnene har i oppgave å ta inn to inngangssignaler og svitsje mellom å forsyne topp- og bunnside transistor. Tabell 4.2 viser en oversikt over de ulike pinnene og deres funksjon.

Pinne	Beskrivelse
HB	Toppside bootstrapforsyning
HI	Toppside inngang
HO	Toppside utgang
HS	Toppside nodesvitsj
LI	Bunnside inngang
LO	Bunnside inngang
VDD	Positiv forsyning til bunnside gatedriver
VSS	Negativ forsyning til gatedriver, jord

Tabell 4.2: Pinnekonfigurasjon og funksjon for UCC27288

4.3.5 Design av bootstrapkrets

For at gatedriveren skal fungere optimalt er det nødvendig å legge til en bootstrapkrets. Skjematikk og utlegg for bootstrapkretsen er utført ved å ta utgangspunkt i referansedesignet til den valgte gatedriveren. Figur 4.18 illustrer en typisk bootstrapkrets til den valgte gatedriveren. Utifra dette har gruppen valgt å legge til bootstrapkondensator, avkoblingskondensator, bootstrapdiode og schottky-dioder.



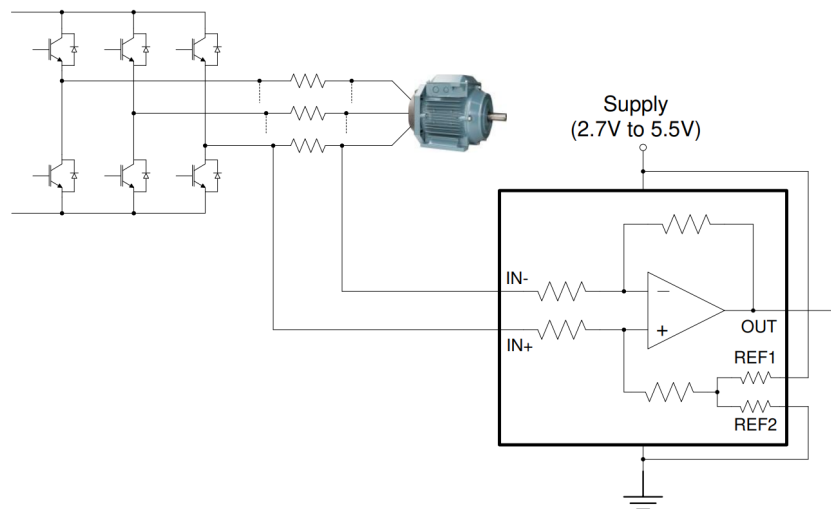
Figur 4.18: Typisk applikasjon av bootstrapkrets for gatedriver UCC27288

Ved hjelp av formlene beskrevet i delkapittel 2.2.4, er det mulig å finne minimumsverdiene på bootstrapkomponentene. Disse ble utregnet til å bli $C_{boot} = 63nF$, $C_{VDD} = 630nF$ og $D_{boot} = 60V$.

Gruppen har ved valg av komponenter til bootstrapkretsen tatt i bruk generiske og standardiserte komponenter som Revolve NTNU bruker. Dette er en av grunnene til at enkelte komponentene er overdimensjonert utover den minste verdien som er regnet ut. I tillegg til de nødvendige komponentene i bootstrapkretsen, har gruppen lagt til flere kondensatorer for å avlaste C_{boot} og C_{VDD} . Skjematikken for gatedriverkretsen er vist i vedlegg A.6.

4.3.6 Strømmåling

For å måle strømmen gjennom de to fasene til motoren, ble det designet en strømmålingskrets. Strømmåleren er i hovedsak bygget opp av en shunt-motstand og en forsterker, hvor shunt-motstanden har blitt implementert i fase U og V , mellom vekselretteren og motoren. Ved å måle spenningsfallet over denne motstanden, kan man ved hjelp av Ohms-lov, regne ut strømmen. Ettersom motstanden er koblet direkte i faseutgangene fra vekselretteren, er det ønskelig at verdien på motstanden er så liten som mulig, for å minimere effekttapet. Ved å forsterke spenningsfallet over shunt-motstanden, kan spenningen leses av i mikrokontrolleren. Figur 4.19 viser hvordan strømmen i en fase måles ved bruk av en shunt-motstand.



Figur 4.19: Applikasjon av strømmåling

Spenningsforsterkeren ble valgt først, for å være sikre kompatibilitet med vekselstrøm og at utgangsspenningen var maksimalt $3.3V$. Det er i tillegg ønskelig med høy CMRR for å dempe felles spenningskomponenter og redusere liketaktstøy [CEVDET]. På grunn av dette ble spenningsforsterkeren INA240A1QPWRQ1 [23]. Gruppen ønsket å sentrere nullpunktet rundt halvparten av forsyningsspenningen, slik at utgangsspenningen ved null strømgjennomgang vil være $\frac{3V_3}{2} = 1.65V$. Dette ble gjort ettersom mikrokontrollerens ADC ikke kan lese negative spenninger. Ved negative strømmer vil dermed utgangsspenningen fra spenningsforsterkeren være mellom null og $1.65V$.

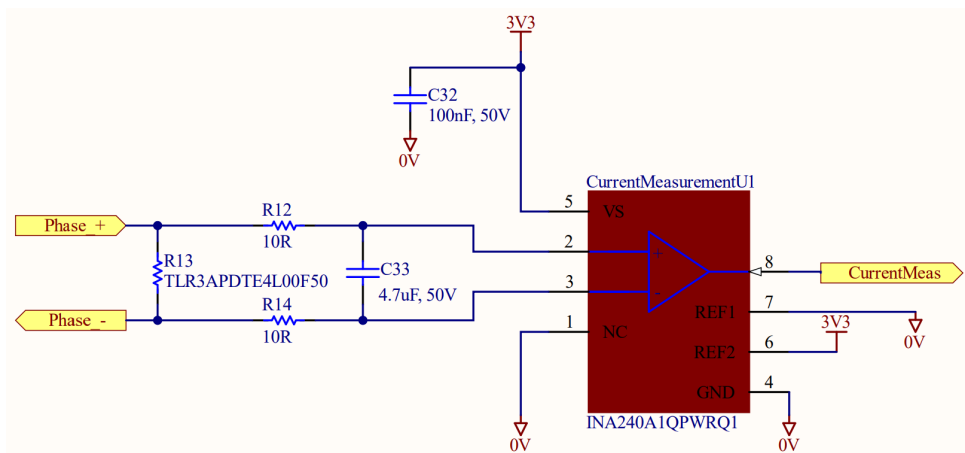
Verdien på shunt-motstanden ble valgt utifra beregninger basert på forsterkning og maksimal strømgjennomgang, og det ble valgt en motstandsverdi på $4m\Omega$. Likning 4.4 viser spenningsfallet over shunt-motstanden ved maksimal fasestrøm. Likning 4.5 og 4.6 viser utgangsspenninger ved maksimal strømgjennomgang i shunt-moststanden.

$$\pm V_{diff_max} = R_{shunt} \cdot (\pm I_{max}) = 4m\Omega \cdot (\pm \sqrt{2} \cdot 14A) = \pm 79mV \quad (4.4)$$

$$V_{max} = +V_{diff_max} \cdot G + V_0 = 79mV \cdot 20 \frac{V}{V} + 1.65V = 3.23V \quad (4.5)$$

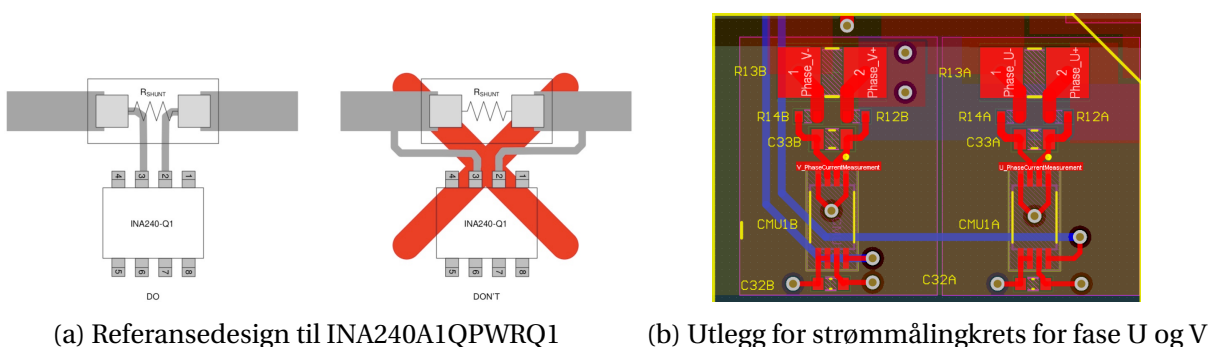
$$V_{min} = -V_{diff_max} \cdot G + V_0 = -79mV \cdot 20 \frac{V}{V} + 1.65V = 0.07V \quad (4.6)$$

Figur 4.20 viser skjematikken til strømmålingskretsen. For å unngå forstyrrelser fra andre deler av kretsen har det blitt implementert en $100nF$ -avkoblingskondensator. I tillegg har gruppen implementert filtrering på inngangsignalene til forsterkeren. Ved å koble referanse-pinne 1 til $0V$ og referanse-pinne 2 til spenningsforsyningen, vil nullpunktet bli sentrert rundt halvparten av forsyningsspenningen.



Figur 4.20: Utklipp fra skjematikken til strømmålingskretsen

Ledningsbanene til strømmålingskretsen har blitt rutet med hensyn til referansedesignet til spenningsforsterkeren. Figur 4.21a viser hvordan rutingen bør være for å få optimal strømmåling med shunt-motstand. Figur 4.21b viser hvordan dette er implementert på kretskortet.



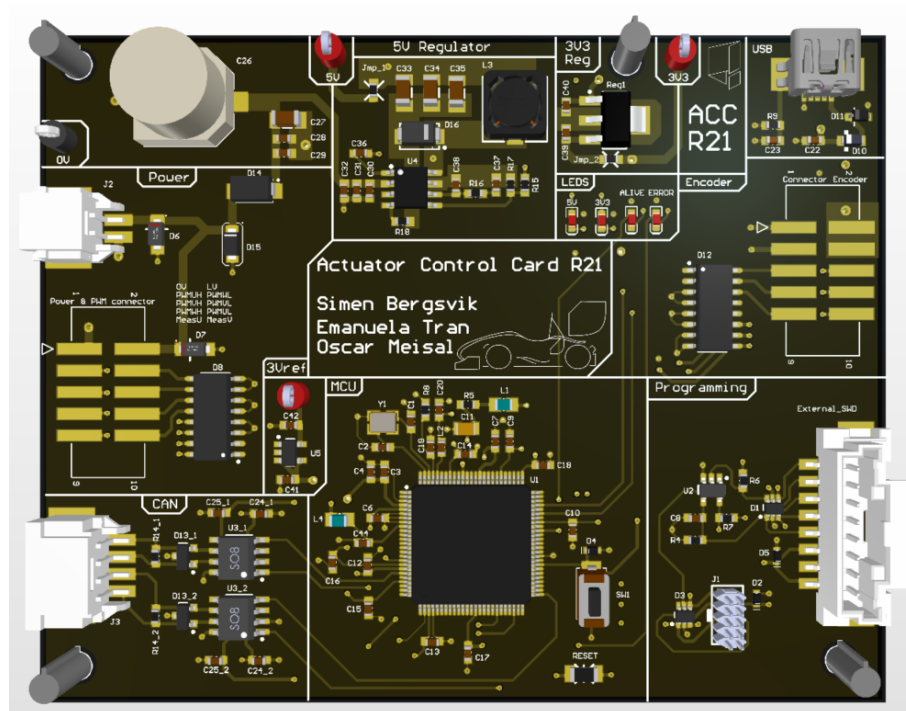
(a) Referansedesign til INA240A1QPWRQ1

(b) Utlegg for strømmålingkrets for fase U og V

Figur 4.21: Utlegg for strømmålingkrets for fase U og V

4.4 Kontrollkort

For å prosessere måledata og kjøre kontrollalgoritmen som styrer svitsjing av transistorene, ble det designet et kontrollkort. Kortet blir forsynt med en spenning mellom 24V og 37V. En 5V buck-regulator og en 3.3V lineær-regulator, som vist i kapittel 4.2.5, blir brukt for å drive de ulike funksjonalitetene på kortet. Kontrollkortet er i hovedsak designet rundt en mikrokontroller. Denne leser måledata fra enkoderen og strømsensoren, og genererer PWM-signaler for å styre svitsjing av transistorene på powerstage-kortet. I tillegg er det lagt til rette for kommunikasjon på to CAN-busser. Figur 4.22 viser en gjengivelse av det ferdig designede kretskortet.



Figur 4.22: Gjengivelse av kontrollkort

4.4.1 Mikrokontroller

Revolve NTNU har flere ulike systemer som bruker en mikrokontroller, der den mest brukte er en ATSAME70N21B [24]. Når nye systemer utvikles er det vanlig at denne brukes, dersom det ikke er andre behov som gjør det nødvendig å ta i bruk andre mikrokontrollere. Ved å bruke samme mikrokontroller på de fleste systemer, vil det gi organisasjonen en mer samlet kompetanse. I tillegg vil det føre til større muligheter til å bidra og hjelpe til på tvers av de ulike systemene. Den totale kostanden er også lavere, ettersom de fleste systemer vil bruke flere av de samme komponentene. Ettersom dette systemet ikke krever ekstra funksjonalitet som ikke støttes av ATSAME70N21B, valgte gruppen å bruke denne.

Den valgte mikrokontrolleren er en ARM Cortex-M7 basert mikrokontroller med FPU og stor båndbredde. Mikrokontrolleren har 100-pinner, og kan operere på en frekvens opp til 300MHz . Den har også integrerte funksjonaliteter som CAN grensesnitt, PWM-moduler, ADC og kvadratur dekode. CAN grensesnittet er viktig for å kommunisere med andre systemer på racerbilen. PWM-modulene er nødvendig for å styre svitsjing av transistorene på powerstage-kortet. ADCen er nødvendig for å lese analog data fra strømmåleren og kvadratur dekode. ADCen brukes for å dekode de ulike signalene fra enkoderen.

I skjematikken i vedlegg A.11 er mikrokontrolleren delt i tre hoveddeler:

- Strømforsyningskrets
- Programmeringskrets
- Inngangs- og utgangspinner

Skjematikken for mikrokontrolleren og komponentene rundt den, er standardisert i Revolve NTNU utifra datablad gitt fra Microchip [24]. Pinnene på mikrokontrolleren har blitt valgt utifra ønsket funksjonalitet og kompatibilitet med mikrokontrolleren. Et eksempel er at enkoder-signalene må være koblet til dekode. ADCen er nødvendig for å lese analog data fra strømmåleren og kvadratur dekode. ADCen brukes for å dekode de ulike signalene fra enkoderen. Alle de brukte pinnene er vist i vedlegg A.11.

Programmeringskrets

For å programmere mikrokontrolleren er det valgt som standard i organisasjonen å bruke en SWD-konnektor. Denne kan kobles til en Atmel Ice, som er et verktøy for programmering og feilsøking av ARM-baserte mikrokontrollere [25]. Plassering av de ulike systemene på racerbilen er forskjellig fra år til år, og det er ikke alltid mulig å programmere en mikrokontroller når kretskortet først er installert i racerbilen. For å legge til rette for fremtidig bruk har det derfor blitt lagt til en ekstra konnektor for programmering.

4.4.2 Enkoder

Kontrollalgoritmen brukt i dette prosjektet er basert på vektorkontroll og er avhengig av å vite posisjonen til rotor i forhold til stator. Som nevnt i kapittel 2.3.3, finnes det ulike måter å måle og estimere rotorposisjon. Det å estimere posisjonen til rotor i forhold til stator basert på motorparametre og motindusert spenning, krever en nøyaktig modell av systemet og er mer komplisert enn å måle posisjon ved hjelp av en sensor. I tillegg egner posisjonsestimering seg dårlig for posisjonsregulering, ettersom det kreves nøyaktig posisjonsavlesning også ved lav hastighet. Gruppen har derfor valgt å måle rotorposisjon ved hjelp av en sensor.

Motoren som gruppen har tatt utgangspunkt i har en inkrementell enkoder og tre hall-effekt sensorer tilgjengelig. Valget har derfor stått mellom disse, ettersom de allerede er integrert i

motoren. Den inkrementelle enkoderen er en *Enkoder 16 EASY XT* [26] fra Maxon Group, og er lagt til som en modulær enhet. De tre hall-effekt sensorene var integrert i motoren som standard fra Maxon, og er ikke en modulær enhet.

Da det skulle velges posisjonsmåler, ble oppløsning og kompatibilitet med mikrokontrolleren vektlagt. Oppløsningen til sensorene er oppgitt i tabell 4.3. Ved å ta i bruk kvadratur dekodning som beskrevet i kapittel 2.3.3, vil oppløsningen til enkoderen være nesten 100 ganger så høy som for de tre hall-effekt sensorene. Den valgte mikrokontrolleren har en innebygd kvadratur dekode, og det er derfor heller ikke nødvendig å skrive en driver for dette.

Sensor	Oppløsning [steg per revolusjon]	Vinkeloppløsning
Enkoder 16 EASY XT	4096	0.09°
Hall-effekt sensor x3	48	7.5°

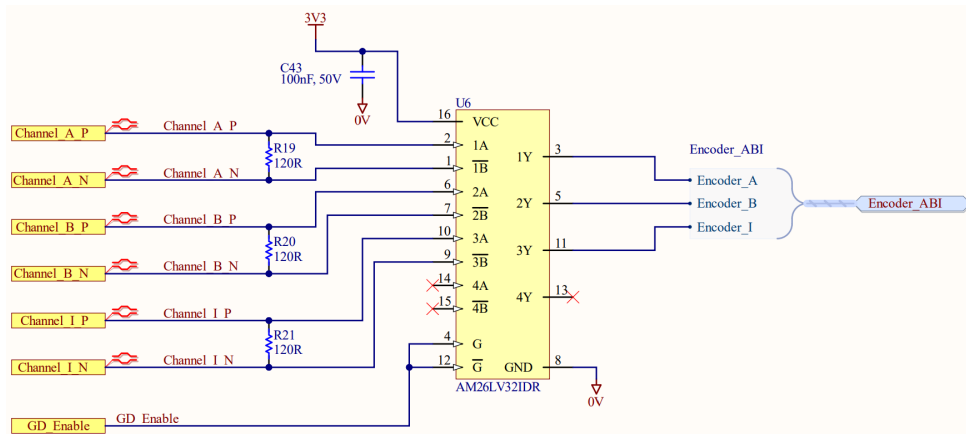
Tabell 4.3: Oppløsning på posisjonsmåler

Det ble derfor valgt å bruke den inkrementelle enkoderen som posisjonsmåler i systemet. En ulempe med interaksjonen mellom enkodere og mikrokontrollere generelt er at de fleste enkodere sender fysiske signaler ved å bruke RS-422 protokollen. De fleste mikrokontrollere har ikke støtte for denne protokollen, og det kreves derfor en ekstern mottaker.

RS-422

RS-422 er en standard for seriell kommunikasjon som vil si at det sendes én og én bit av gangen. En fordel ved RS-422 grensesnittet er at det sendes differensielle signaler, som gjør at den er robust for støy og kan sendes over lengre distanser. Ulempen med å kommunisere over et RS-422 nettverk er som nevnt at mikrokontrolleren valgt i dette prosjektet ikke har støtte for denne typen kommunikasjon. For å kunne lese data fra enkoderen er det dermed nødvendig å implementere en RS-422 mottaker.

Ved å koble de differensielle signalene fra enkoderen til en RS-422 mottakeren, vil de differensielle signalene gjøres om til ensidige signaler, slik at mikrokontrolleren kan lese måledataen. Ved valg av mottaker var det viktig at den hadde støtte for minst tre kanaler, for å fungere med enkoderen. Ettersom mikrokontrolleren ikke har støtte for høyere spenning enn 3.3V, var det derfor et ønske om en mottaker med utgangsspenning tilsvarende dette. På den måten var det ikke nødvendig å lage en spenningsdeler for å kunne lese måledata. Valget falt derfor på AM26LV32IDR fra Texas Instruments [27]. Denne mottakeren har en forsyningsspenning på 3.3V, men kan likevel ta imot inngangsspenninger på 5V, som kommer fra enkoderen. I tillegg har den støtte for fire differensielle signalpar. Det har også blitt brukt termineringsmotstander på 120Ω mellom de differensielle parene for å unngå refleksjoner, som spesifisert i databladet [27]. Figur 4.23 viser hvordan dette er implementert.



Figur 4.23: Skjematikk av RS-422 mottaker

4.4.3 CAN

De elektriske systemene i Revolve NTNU sine racerbiler kommuniserer over kommunikasjonsprotokollen CAN. *Controller Area Network* (CAN) er en kommunikasjonsprotokoll som er spesifisert i ISO 11898-standarden [28]. Protokollen definerer det fysiske laget og datalink laget. Imotsetning til mange andre kommunikasjonsprotokoller som bruker adressering for å sende meldinger til spesifikke noder, er CAN en meldingsbasert protokoll. Det vil si at meldinger blir publisert på et nettverk, og nodene kan selv velge hvilke meldinger de ønsker å lytte etter. Hver melding er definert med en unik ID som skiller meldinger fra hverandre og spesifiserer prioritet.

Systemet i dette prosjektet er avhengig av å kunne kommunisere med andre enheter på racerbilen. Eksempelvis vil settpunkt for hjulvinkel mottas over CAN. Ettersom bilene til Revolve NTNU bruker to forskjellige CAN-busser, ble det implementert to CAN-trancievere for å kunne lese meldinger fra begge bussene. Skjematikk for dette er vist i vedlegg A.12.

4.4.4 USB-strømforsyning

De ulike delene av kontrollkortet bruker en forsyningsspenning på enten 5V eller 3.3V. Det er derfor ikke nødvendig å bruke 24V spenningsforsyning for å teste alle funksjonaliteter. For å muliggjøre strømforsyning av kortet direkte fra en PC, ble det derfor implementert en USB-port. Mye av feilsøking på kontrollkortet er knyttet til mikrokontrolleren, og dette vil da være mulig å gjøre uten tilgang til lab-utstyr. USB-strømforsyningskretsen er vist i ruten *USB* i vedlegg A.9.

4.5 Programmering av mikrokontroller

Programvaren til et system som dette prosjektet omhandler, kan raskt bli omfattende og tidkrevende å utvikle. På grunn av tidsbegrensning har gruppen prioritert design av kretskort framfor programvare, ettersom kretskortene må være på plass for at det i det hele tatt skal kunne implementeres kode som kan verifiseres. Gruppen valgte derfor å kun produsere nødvendig programvare for å teste hovedfunksjonalitetene på kretskortene.

I Revolve NTNU er de fleste systemer programmert enten i sanntidsoperativsystemet FreeRTOS, eller ved å skrive sekvensiell kode, også kalt Bare-Metal kode.

4.5.1 Valg av type programmering

FreeRTOS er et sanntidsoperativsystem som opererer med en planlegger som prioriterer oppgaver og utfører disse parallelt i ulike tidsintervall. Planleggeren velger hvor ofte hver oppgave kan kjøre, i tillegg til hvilke ressurser som tildeles. En fordel ved FreeRTOS er at systemer som har flere oppgaver som skal utføres, enklere kan implementeres. Ulempen er derimot at frekvensen som programmet kjører på er begrenset til cirka 1 kHz .

Bare-Metal-programmering går ut på å skrive kode direkte på mikrokontrolleren uten bruk av et operativsystem. I motsetning til FreeRTOS, som er oppgaveorientert, vil koden kjøres sekvensielt i en loop ved Bare-Metal-programmering. Dette gjør at frekvensen til programmet ikke er begrenset på samme måte som ved FreeRTOS. Det som gjør FreeRTOS til et bedre alternativ enn Bare-Metal er at dersom det er feil i koden, vil andre deler av programmet fortsatt fungere da hele programvaren er oppdelt i oppgaver som ikke nødvendigvis er avhengig av hverandre. Feil i Bare-Metal-skrevet kode vil imidlertid raskere resultere i at koden ikke kjøres.

I gruppens design kjører reguleringsalgoritmen på 50 kHz , og valget har dermed blitt å skrive Bare-Metal-programvare. For å hindre feil og utfordringer ved fremtidig utvikling av kode har Git blitt brukt. Uavhengig av hvordan koden blir skrevet vil det alltid være en fungerende kode i mastergrenen. Ved videreutvikling vil koden skrives i forskjellige grener. Disse grenene vil kun slås sammen ved pull request som i organisasjonen må bli sett over av minst to personer.

4.5.2 PWM-generering

For å teste hovedfunksjonaliteten til kretskortene, var det nødvendig å implementere PWM-signaler. Gruppen allokerte tidlig pinner til PWM-signalene på mikrokontrolleren. For at PWM-signalene skal være synkrone, må de moduleres av samme trekantpuls. Hvordan dette blir gjort er beskrevet i kapittel 2.2.5. Mikrokontrolleren har flere pinner som kan gi ut

PWM-signaler, men for at disse skal bruke samme kontrollsignal og svitsje synkront må alle velges på samme perifer. I tabell 4.24a ser man at alle PWM-signalene er valgt fra perifer *PWMC0*. Det er så valgt tre ulike kanaler på periferen. Hver kanal lager to komplementære signaler, hvor PWM_HX styrer den toppside transistoren og PWM_LX styrer bunnside transistoren på hvert ben av vekselretteren. I figur 4.24b ser man hvordan dette er satt opp i programvaren. Her blir også svitsjefrekvensen bestemt.

Pinne [PIOA]	PWM
11	PWMC0_PWMH0
19	PWMC0_PWML0
12	PWMC0_PWMH1
20	PWMC0_PWML1
13	PWMC0_PWMH2
30	PWMC0_PWML2

(a) Valg av pinner for PWM-generering

```

12 static inverter_opt_t inverter_options =
13 {
14     .pio_bank = PIOA,
15     .pwm_peripheral = PWM0,
16
17     .u_high_pin = 11,
18     .v_high_pin = 12,
19     .w_high_pin = 13,
20     .u_low_pin = 19,
21     .v_low_pin = 20,
22     .w_low_pin = 30,
23
24     .u_high_mux = A,
25     .u_low_mux = A,
26     .v_high_mux = A,
27     .v_low_mux = A,
28     .w_high_mux = A,
29     .w_low_mux = A,
30
31     .u_channel = 0,
32     .v_channel = 1,
33     .w_channel = 2,
34
35     .frequency = 50000,
36 };

```

(b) Oppsett av PWM-signaler i programvare

Figur 4.24: Oppsett for PWM-generering

De ulike pulsbreddene blir bestemt av genererte sinuser i programvaren for testing. I utgangspunktet skal dette bli gjort av kontrollalgoritmen, men for testing implementerte gruppen tre sinuser som er faseforskjøvet 120° . For å bestemme frekvensen til sinusene er det brukt en teller som blir oppdatert for hver gang koden kjører. Frekvensen kan da styres ved å dele telleren på et heltall. Hver gang kvotienten blir et heltall vil pulsbredden være lik 0. Kodeutsnitt er vist i figur 4.25.

```

29     duty_cycles.u_duty = (uint32_t)(100.0 * (0.5 * arm_sin_f32(2 * PI *
(float)(timer)/800.0) + 0.5));
30     duty_cycles.v_duty = (uint32_t)(100.0 * (0.5 * arm_sin_f32(2 * PI *
(float)(timer)/800.0 + (2*PI/3)) + 0.5));
31     duty_cycles.w_duty = (uint32_t)(100.0 * (0.5 * arm_sin_f32(2 * PI *
(float)(timer)/800.0 + (4*PI/3)) + 0.5));
32     timer++;
33     pwm_generator_set_duty(&duty_cycles);

```

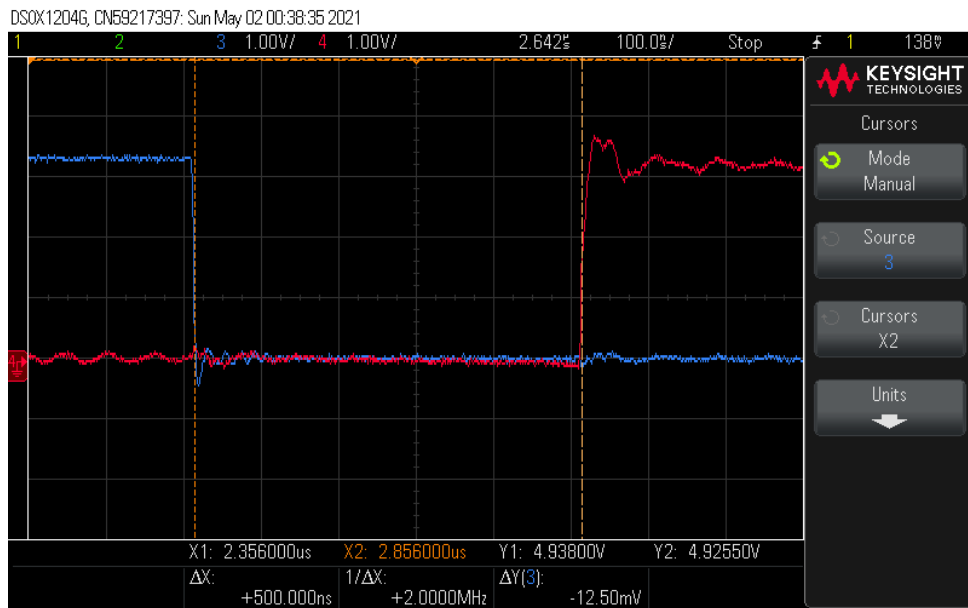
Figur 4.25: Sinusbølger generert som pulsbredder

Resultatet av to komplementære PWM-signaler kan ses i figur 4.26. Det kan sees ringing på det ene signalet som kommer av at det ble brukt en dårlig probe for å måle signalet.



Figur 4.26: To komplementære PWM-signaler

Som nevnt i delkapittel 2.2.5 er dødtid en viktig faktor for svitsjing av transistorer. Dette har blitt implementert i programvaren med en god margin. Dødtiden har blitt satt til 500ns for å forsikre at et ben ikke blir kortsluttet. Dette kan med fordel bli kortet ned, men er en unødvendig risiko for funksjonalitetstesting. Figur 4.27 viser dødtiden mellom de to PWM-signalene i figur 4.26.



Figur 4.27: 500ns dødtid mellom to komplementære PWM-signaler, $100\text{ns}/\text{div}$

4.6 HIL-test

HIL-test, *Hardware-in-the-loop-test*, utføres for å verifisere at ulike systemer kan samhandle. Før en slik test blir gjennomført er det vanlig å teste de ulike systemene individuelt. Dette blir gjort for å enklere kunne isolere og avdekke eventuelle funksjonsfeil før systemene testes sammen.

Gruppen ønsket å utføre en HIL-test for å teste svitsjing av transistorene på powerstage-kortet. Ved å utføre denne testen ville gruppen kunne bekrefte at vekselretteren fungerer og at samhandlingen mellom kontrollkortet og powerstage-kortet fungerer.

For å få mest mulig ut av en HIL-test er det nødvendig med en testplan. Ved å ha en plan for hvordan de ulike funksjonalitetene skal bli testet og hvilke resultater som kreves for en godkjent test, vil man få en bekreftelse på om systemet fungerer som det skal. I tillegg vil man ved å følge en testplan oppdage hvor en eventuell feil oppstår.

HIL-testplanen sørger for god oversikt under testing. Figur 4.28 viser testplanen med kommentarer fra testen. Resultatene fra HIL-testen er ytterligere kommentert i kapittel 4.6.2.

Funksjonalitet	Hvordan	Kriterier for fullført test	Nødvendig utstyr	Teststatus	Kommentarer	Viktighet
PWM	Probe PWM-innganger til gatedriverne på powerstage-kortet.	<ul style="list-style-type: none"> PWM-signalene er firkantpulser med max. 3V3 og min 0V. Dødtid mellom to komplementære PWM-signaler skal være 500ns. 	Standard utstyr	Godkjent	UH, UL, VH, VL, WH, WL	Viktig
Gatedriver	Probe gate-innganger til transistorene	<ul style="list-style-type: none"> Firkantpulser med max. 12V og min 0V. Dødtid på 500ns mellom to komplementære gate-signaler. 	Standard utstyr	Godkjent	G_UH, G_UL, G_VH, G_VL, G_WH, G_WL	Viktig
Fasespenning	Probe en og en fasespenning ut av Powerstage-kortet	<ul style="list-style-type: none"> Firkantpulser med max. 24V og min 0V. 	Standard utstyr	Godkjent		Viktig
Trefase: Sinusidal og faseforskyvning	Probe de tre fasespenningene samtidig.	<ul style="list-style-type: none"> Firkantpulsene representerer en sinuskrive, hvor pulsbredden er stor i noen områder og mindre i andre. Periodisk. Fasene er faseforskjøvet. Dette kan bekreftes ved å måle når de ulike pulsbreddene er av maksimal bredde, og observere at disse er faseforskjøvet mellom hverandre. 	Standard utstyr	Ikke-godkjent	Fase W fungerer ikke. Fase U og V fungerer og er faseforskjøvet.	Viktig

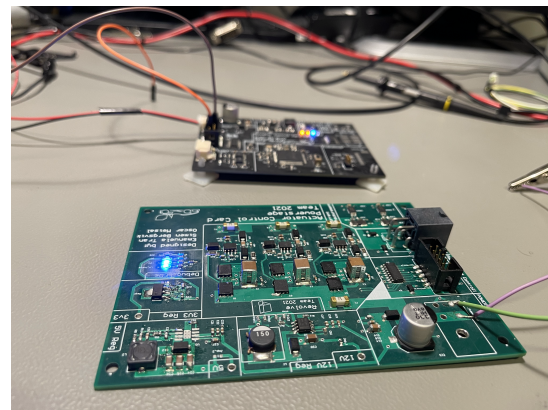
Figur 4.28: HIL-testplan

4.6.1 Testoppsett

Figur 4.29 viser testoppsettet som ble brukt under HIL-testen. Det ble brukt to strømforsyninger med en forsyningspenning på 24V. Minus-terminalene på strømforsyningene ble koblet sammen, for å skape en felles referanse. Oscilloskopet ble brukt for å probe og validere de ulike signalene på kretskortene. Ettersom det ble brukt flere prober samtidig, var det viktig å kun koble én av dem til referansespenningen. Dette ble gjort ettersom probene er koblet til samme referansepunkt inni oscilloskopet, og det kan føre til kortslutning dersom de ulike probene kobles til ulik referansespenning. Testen ble utført på en ESD-matte, og det ble i tillegg brukt ESD-lenke for å unngå skade på elektronikken som følge av elektrostatisk utladning.



(a) Testoppsett



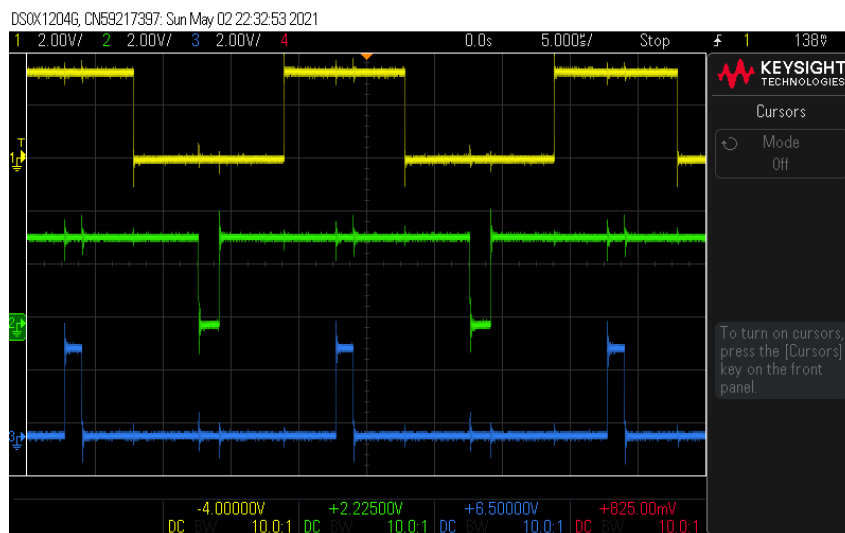
(b) Powerstage- og kontrollkort

Figur 4.29: Testoppsett for HIL-test

4.6.2 Testresultater for HIL-test

For å bekrefte at alle PWM-signalene fra kontrollkortet sendes til powerstage-kortet, ble disse koblet opp individuelt. PWM-signalene ble testet enkeltvis, hvor det ble probet på den tilhørende inngangen på gatedriveren. Årsaken til at dette ble gjort var for å bekrefte at signalene nådde helt frem til gatedriveren, samtidig som eventuelle feilkilder ble isolert.

Figur 4.30 viser alle toppside PWM-signalene etter at de er testet individuelt. Signalene er firkantpulser som veksler mellom en spenning på 3.3V og 0V. I figuren ser man at pulsbredden til de ulike PWM-signalene er ulik. Dette kommer av at de ulike kontrollsignalene som PWM-signalene representerer, er faseforskjøvet med 120° mellom hverandre.



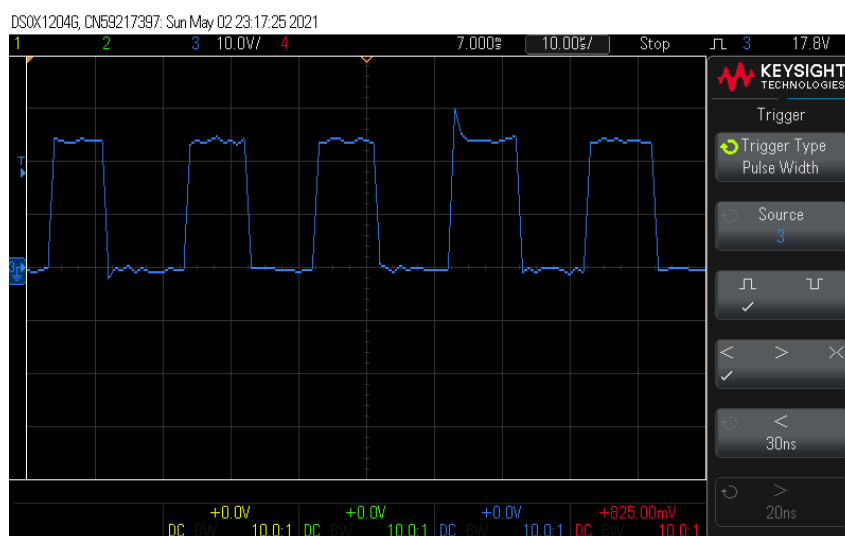
Figur 4.30: Toppside PWM-signaler

Gatedriver

For å bekrefte at gatedriveren gir ut forventede resultater, ble det påtrykt to og to komplementære PWM-signaler. Resultatene av denne testen viste at gatedriveren øker spenningen til PWM-signalene fra 3.3V til 12V, og påtrykker dette på gate-terminalen på transistoren.

Fasespenning

For å bekrefte at svitsjingen av transistorene fungerer, var det nødvendig å analysere utgangene fra transistorene. Dette ble gjort individuelt for hver fase, for å isolere eventuelle feilkilder. Figur 4.31 viser fasepenningen for fase V. Her ser man at fasespenningen er firkantpulset og veksler mellom 24V og 0V. Fase U og fase W viste samme resultat.



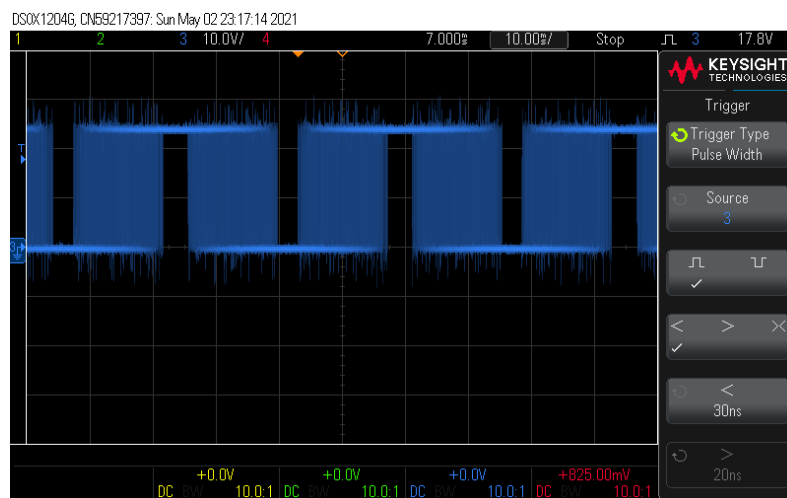
Figur 4.31: Fase V

Trefase

Gruppen ønsket å bekrefte at vekselretteren trefase vekselspenninger, ved å utføre en test hvor alle faseutgangene ble probet samtidig. Faseforskyvning kunne dermed bekreftes ved å sammenligne når de ulike fasespenningene hadde maksimal pulsbredde, og observere at dette skjedde periodisk til ulik tid. Gruppen koblet derfor opp alle PWM-signalene, og koblet prober på hver av faseutgangene. Da testen ble gjennomført skrudde gruppen først på powerstage-kortet og deretter kontrollkortet. Da kontrollkortet ble skrudd på, hørte gruppen at sikring F1 sett i vedlegg A.8, brøt strømmen. Ved nærmere undersøkelser og ettertanke, har gruppen konkludert med at ved oppstart setter mikrokontrolleren alle pinner logisk høy i en kort periode. Dette resulterer i at begge transistorene på hvert ben blir skrudd på samtidig, noe som fører til kortslutning mellom spenningsforsyningen og 0V. Av den grunn gikk sikringen som følge av for høy strøm.

Gruppen ønsket å utføre testen på nytt, og loddet derfor på en ny sikring. Det var imidlertid mistanke om at kortslutningen hadde ført til at enkelte komponenter var blitt ødelagt. Før en ny test ble igangsatt, ble det derfor utført diodetester og det ble målt motstand over alle transistorene. Det ble her observert at alle transistorene hadde ulik motstand mellom drain-source, og det var mistanke om at disse var ødelagt. For å bekrefte dette ønsket gruppen å gjennomføre testen på nytt, hvor kretskortene ble skrudd på i motsatt rekkefølge. Testen viste at fase *V* og fase *W* var defekt, og det var kun fase *U* som fungerte.

Figur 4.32 viser fase *U* over en lengre tidsperiode enn det som ble presentert i figur 4.31. Her ser man at firkantpulsene representerer en sinuskurve, i form av at pulsbredden varierer periodisk. Dersom faseutgangen hadde blitt koblet til en motor ville strømmen sett ut som en glatt sinuskurve. Testen ble dermed ikke godkjent, ettersom alle fasene ikke fungerte. Å lodde opp et nytt kort eller feilsøke og reparere ville det krevd mye tid gruppen ikke hadde. Det ble derfor valgt å avslutte testen.



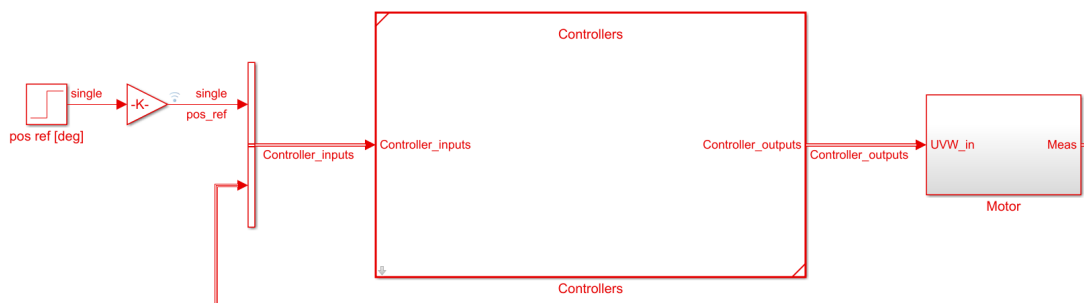
Figur 4.32: Fase U

4.7 Motorkontroll og simulering

Når et digitalt kontrollsystem skal designes og tunes kan dette gjøres på ulike måter. Ett alternativ er å utføre håndregninger og skrive kode ut ifra dette. Et annet er å bruke simuleringsverktøy som Matlab og Simulink for å analysere systemet og tune regulatorene før det implementeres. I dette prosjektet ble det valgt å bruke Simulink til å modellere motoren og kontrollalgoritmen. Det er flere fordeler som ledet opp til denne beslutningen:

- **Forståelse for systemets dynamikk:** Ved å modellere og simulere systemet vil man få en bedre forståelse for systemets dynamikk, og det blir raskere å gjøre analyse.
- **Lavere risiko for skader på elektronikk og motor:** Når kontrollsystemet skal implementeres er det risikabelt å gjøre dette på det fysiske systemet, uten å verifisere at algoritmen først fungerer. Dersom feil oppstår, er risikoen stor for at utstyr skades. Ved å verifisere algoritmen på forhånd reduseres denne risikoen.
- **Enklere implementasjon:** Ved å modellere kontrollalgoritmen i Simulink kan denne benyttes for å generere C-kode som kjører på en mikrokontroller. Dette forenkler software-utviklingen betraktelig sammenlignet med å skulle skrive egen kode, i tillegg til at kodefeil i større grad unngås.

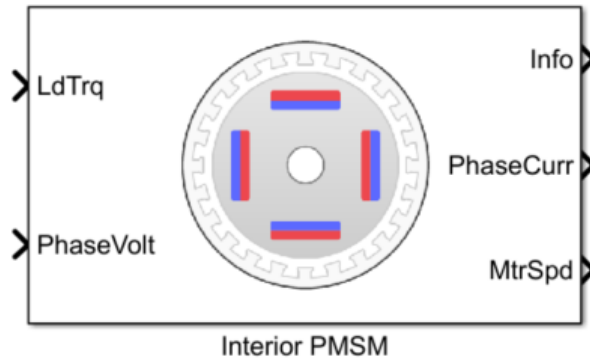
For å legge til rette for videre utvikling av både reguleringsystem og motoren det skulle modelleres med, ble det valgt å lage et tydelig skille mellom motorkontrolleren og resten av systemet. Motorkontrolleren kjører algoritmen som skal kunne implementeres på en mikrokontroller, mens motoren og det mekaniske systemet eksisterer kun som en modell for bruk til dimensjonering, testing og innstilling av motorkontrolleren. Figur 4.33 viser hvordan dette er implementert i Simulink. Her er motorkontrolleren laget i et maskert subsystem, mens motoren er laget i et annet. Pådraget fra motorkontrolleren består av de tre fasespenningene U , V og W i bussen *Controller_outputs*, og bussinngangen *Controller_inputs* inneholder i_U , i_V og rotorposisjonen θ fra motoren, i tillegg til settpunktet til hjulvinkelen.



Figur 4.33: Reguleringsystem og motormodell

4.7.1 Motormodell

For å lage kontrollalgoritmen til motorkontrolleren er det nødvendig med en modell av systemet som skal reguleres. Det ble vurdert om en matematisk modell av Maxon's EC-i52 BLDC-motor skulle utvikles, men det ble raskt tydelig at dette ville være for tidkrevende for prosjektet. Gruppen vurderte da at det var viktigere å lage en solid grunnmodell som fungerer, fremfor å overkomplisere der det ikke var nødvendig. Det ble derfor valgt å bruke den innebygde PMSM-blokken [29] i Simulink for å representere motoren, som vist i figur 4.34.



Figur 4.34: PMSM-blokk i Simulink

Blokken implementerer et sett med likninger som representerer dynamikken til en synkronmotor med permanentmagneter. Forskjellen mellom en PMSM og en BLDC-motor er hovedsaklig hvilken form den motinduserte spenningen har, men kontrollsystemets design er ellers likt. Blokken representerer dermed ikke Maxon's EC-i52 motor helt presist, og det er forventet at noe etterjustering av kontrolleren må gjøres. Ettersom det heller ikke finnes en tilsvarende blokk for en BLDC i simulinks bibliotek, ble det valgt å gå videre med en PMSM. Likning 4.7, 4.8, 4.9 og 4.10 beskriver de elektriske sammenhengene i motorblokken

$$\omega_e = n\omega_m \quad (4.7)$$

$$\frac{d}{dt}i_d = \frac{1}{L_d}v_d - \frac{R}{L_d}i_d + \frac{L_q}{L_d}n\omega_m i_q \quad (4.8)$$

$$\frac{d}{dt}i_q = \frac{1}{L_q}v_q - \frac{R}{L_q}i_q - \frac{L_d}{L_q}n\omega_m i_d - \frac{\lambda_{pm}\omega_m}{L_q} \quad (4.9)$$

$$T_e = 1.5n(\lambda_{pm}i_q + (L_d - L_q)i_d i_q) \quad (4.10)$$

Likning 4.11 viser momentbalansen i motoren

$$\frac{d}{dt}\omega_m = \frac{1}{J}(T_e - T_f - F\omega_m - T_m) \quad (4.11)$$

Tabell 4.4 viser de ulike symbolene brukt og deres betydning.

Symbol	Enhet	Betydning
ω_e	rad/s	Vinkelhastighet til magnetfelt i stator
ω_m	rad/s	Vinkelhastighet til rotor
n		Polpar
i_d	A	d-akse strøm
i_q	A	q-akse strøm
L_d	H	Induktans i d-aksen
L_q	H	Induktans i q-aksen
R	Ω	Fasemotstand
v_q	V	Spenning i q-aksen
λ_{pm}	Wb	Flukskobling
J	kgm^2	Treghetsmoment
T_e	Nm	Elektromagnetisk moment
T_f	Nm	Statisk friksjonsmoment i rotoraksel
F	N	Viskøs friksjon i motor og last
T_m	Nm	Moment i rotoraksel

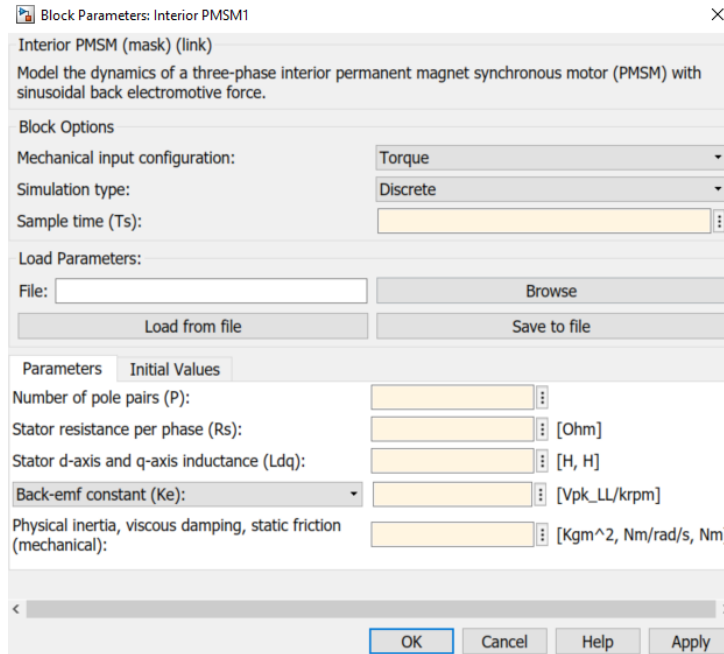
Tabell 4.4: Symboler i likninger for PMSM-blokk i Simulink

Innganger og utganger

Motorblokken tar inn fasespenninger og moment på rotorakselen som argumenter, og ut ifra dette beregnes ulike verdier. *MtrSpd*-utgangen gir ut rotors vinkelhastighet og *Phasecurr* gir ut fasestrømmene i motoren. Utgangen *Info* er konfigurert som en buss der man kan velge hvilken informasjon som hentes ut. Her kan det velges mellom blant annet elektriske og mekaniske tap, spenninger og strømmer. For å styre motoren er det imidlertid kun nødvendig å vite to av fasestrømmene og rotorhastigheten. *Info*-utgangen er derfor terminert.

Blokkparametre

Parametrene i motormodellen ble satt ut ifra merkeverdiene til Maxons EC-i52 BLDC-motor som vist i tabell 3.2. De oppgitte verdiene er imidlertid ikke ekvivalente med de motorblokken krever, og det var derfor nødvendig å konvertere enkelte verdier. Figur 4.35 viser hvilke parametre blokken krever.



Figur 4.35: Parametre for PMSM-blokk i Simulink

d- og q-induktansen (L_d og L_q) motorblokken krever, er blant annet ikke oppgitt i motorens datablad. For Maxons EC-i-serie er $L_d \approx L_q \approx 0.5L_{pp}$, der L_{pp} er induktansen fra fase til fase [30]. Likning 4.10 kan dermed komprimeres og det blir tydelig at q -strømmen bestemmer det produserte momentet i motoren.

$$T_e = 1.5n\lambda_{pm}i_q \quad (4.12)$$

Da motoren er koblet i en stjernekonfigurasjon blir fasemotstanden

$$R = \frac{1}{2}R_{pp} \quad (4.13)$$

der R_{pp} er motstanden fra fase til fase.

Det komplette mekaniske systemet består av motor, gir og last. Ettersom denne rapporten i hovedsak omfatter design av vekselretter og motorkontroller, ble det besluttet å ikke gå inn på lastens dynamikk og parametre. Dynamikken er gitt av de mekaniske egenskapene til styresystemet, og vil kreve videre analyse for å identifisere. Lastmomentet er likevel inkludert i modellen, for å gi mulighet til å inkludere denne påvirkningen i senere tid.

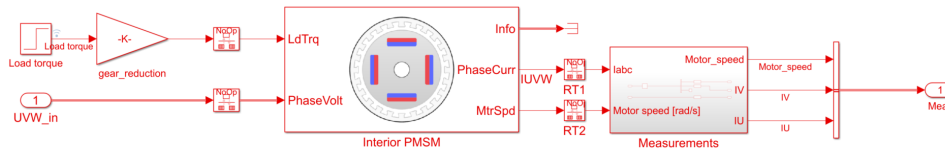
Trehetsmomentet på motorsiden er gitt ved

$$J = J_m + J_G + \frac{1}{n^2} J_L \quad (4.14)$$

der J_L er trehetsmomentet til lasten på utgangen av giret. Den totale viskøse dempingen på motorsiden blir

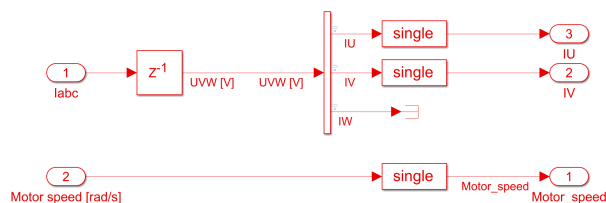
$$d = d_m + \frac{d_L}{n^2} \quad (4.15)$$

Figur 4.36 viser hvordan PMSM-blokken er integrert i motor-subsystemet fra figur 4.33. Det totale subsystemet består dermed av motorblokken, i tillegg til et subsystem for å behandle målingene fra motoren.



Figur 4.36: Komplette modell av motor, gir og last

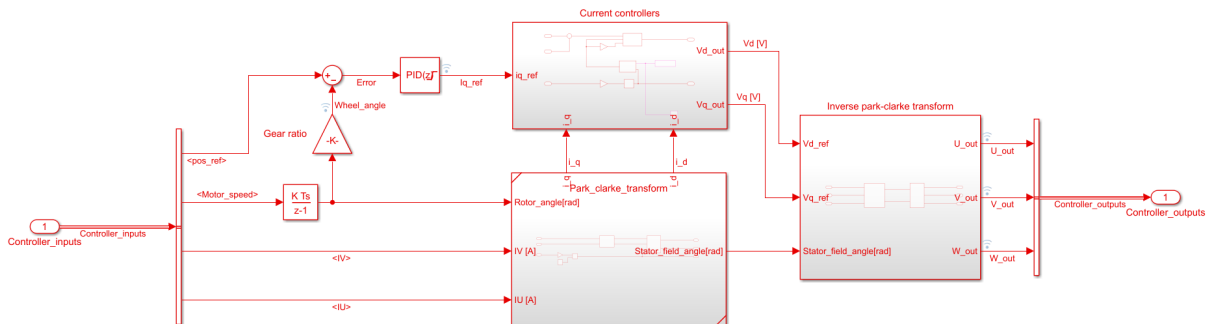
Årsaken til at målingene ikke kan brukes direkte er fordi systemet simuleres diskret og er tilbakekoblet. Når Simulink skal beregne resultatene gjøres dette ved å regne ut resultatene fra én og én blokk sekvensielt. Dersom to blokker koblet i serie regnes ut, vil resultatet til den første blokken være argumentet til den andre. Hvis den andre blokken så er tilbakekoblet til den første, vil ikke Simulink ha en startverdi å regne ut ifra. Ved simulering vil man dermed få en feilmelding om algebraisk loop, og simuleringen stopper. For å løse dette er det implementert en tidsforsinkelse på ett tidssteg på fasestrømmålingene. For motorhastigheten er denne tidsforsinkelsen ikke nødvendig, ettersom den integreres diskret for å beregne rotorposisjon i et senere ledd. Figur 4.37 viser subsystemet *Measurements* og hvordan denne utfordringen løses.



Figur 4.37: Subsystemet *Measurements* med tidsforsinkelse på fasestrømsignalene

4.7.2 Overordnet reguleringsystem

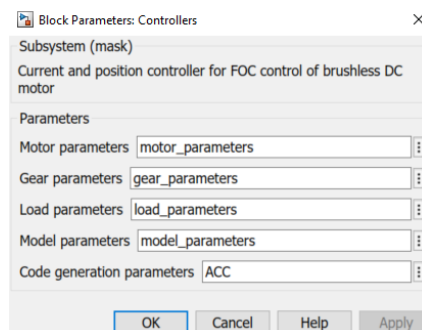
Den overordnede oppgaven til reguleringsystemet er å regulere hjulvinkelen til systemet etter en ønsket vinkel, gitt fra et eksternt signal. For å oppnå dette må fasestrømmene reguleres og magnetfeltet i stator orienteres slik at motoren kan rotere. Ettersom motoren er et balansert trefase-system er det vanskelig å regulere de tre fasestrømmene direkte, og det er derfor brukt vektorkontroll for å forenkle systemet og tillate direkte regulering av fluks og moment i motoren. Figur 4.38 viser innholdet i subsystemet *controllers* fra figur 4.33.



Figur 4.38: Reguleringsystem i Simulink

Kontrollalgoritmen for vektorkontroll kan i hovedsak deles opp i tre ulike komponenter: Posisjonsregulator, strømregulatorer og støttransformasjoner. Algoritmen tar inn ønsket hjulvinkel, fasestrømmålingene for fase U og V , og rotorhastigheten. Basert på denne informasjonen beregnes hvilke spenninger som må påtrykkes motoren for at den skal rotere så utgangen av giret oppnår den ønskede vinkelen.

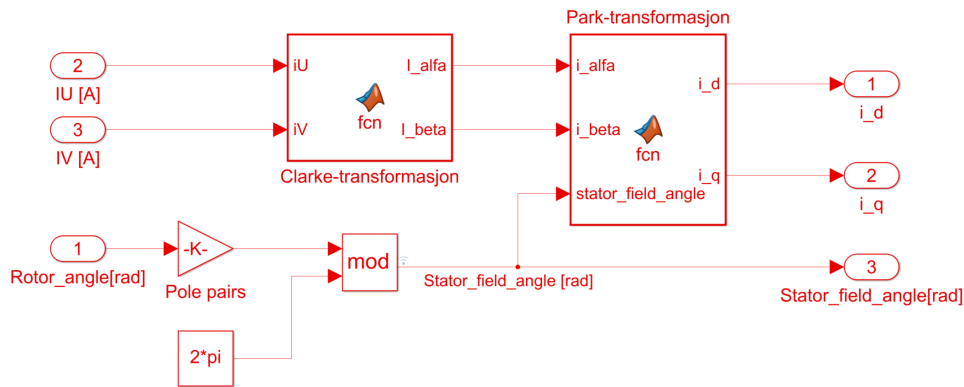
Reguleringsystemet er laget som et maskert subsystem i Simulink. Det at systemet er maskert betyr her at det er laget et eget brukergrensesnitt for subsystemet der man kan spesifisere parametre. Subsystemet ser da omtrent ut som en annen blokk, som vist i figur 4.39. Parametrene som brukes her er definert som structs i et eget script for initialisering av modellen.



Figur 4.39: Parametre for reguleringsystemet i Simulink

Clarke og Park-transformasjoner

For å dekomponere moment og fluks i motoren slik at disse kan reguleres individuelt, er det implementert Clarke og Park-transformasjoner. Statorstrømmene representeres da i dq -planet. Figur 4.40 viser hvordan subsystemet *park_clark_transform* fra figur 4.38 er implementert i Simulink.



Figur 4.40: Park- og Clarke-transformasjoner i Simulink

Transformasjonene tar utgangspunkt i likning 2.8 og 2.9 fra kapittel 2.3.2.

Likning 2.8 tar utgangspunkt i tre fasestrømmer I_a , I_b og I_c , men siden systemet er balansert er det kun nødvendig med to fasestrømmer for å utføre transformasjonen. At systemet er balansert betyr at

$$I_a + I_b + I_c = 0 \quad (4.16)$$

Fasestrømmen I_c kan dermed uttrykkes som

$$I_c = -I_a - I_b \quad (4.17)$$

Clarke-transformasjonen fra likning 2.8 kan dermed uttrykkes som vist i likning 4.18.

$$\begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} = \begin{bmatrix} I_a \\ \frac{1}{\sqrt{3}}(I_a + 2I_b) \end{bmatrix} \quad (4.18)$$

I Simulink er Clarke-transformasjonen implementert ved å skrive et matlab-script som brukes i modellen. Dette er gjort siden det er mer oversiktlig å skrive et script enn å implementere unødvendig mange matematiske operatører.

Figur 4.41 viser scriptet for Clarke-transformasjonen i matlab

```

1 function [I_alfa, I_beta] = fcn(iU,iV)
2 I_alfa = iU;
3 I_beta = 1/(sqrt(3))*(iU+2*iV);
4 end

```

Figur 4.41: Clarke-transformasjon i Matlab-script

For å transformere fasestrømmene fra $\alpha\beta$ -planet til dq -planet benyttes park-transformasjonen. Til dette kreves vinkelen til det roterende magnetfeltet. Fra kapittel 2.1.2 er det gitt at forholdet mellom rotors vinkel θ og vinkelen til det roterende magnetfeltet θ_e er gitt ved

$$\theta_e = n\theta \quad (4.19)$$

der n er antall polpar i motoren. Da vinkelen til rotor fra motorblokken akkumuleres over tid, må vinkelen begrenses til det periodiske området $[0, 2\pi)$. Dette er gjort ved å ta i bruk modulus-funksjonen i Simulink, som vist i figur 4.40. Figur 4.42 viser implementasjonen av Park-transformasjonen i Simulink. På lik linje med Clarke-transformasjonen, er også Park-transformasjonen implementert som et script.

```

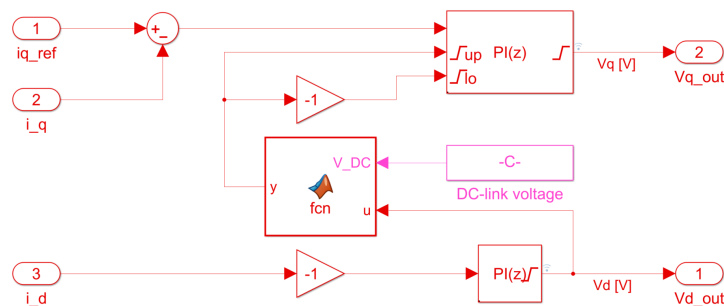
1 function [i_d, i_q] = fcn(i_alfa,i_beta,stator_field_angle)
2 k = [cos(stator_field_angle) sin(stator_field_angle); ...
3      -sin(stator_field_angle) cos(stator_field_angle)];
4 in = [i_alfa; i_beta];
5 out = k*in;
6 i_d = out(1);
7 i_q = out(2);
8 end

```

Figur 4.42: Park-transformasjon i Matlab-script

4.7.3 Strømregulatorer

Strømregulatorene i reguleringssystemet er implementert i et eget subsystem som tar inn en q -referansestrøm og de to målte dq -strømmene (figur 4.43).



Figur 4.43: Strømregulatorer i Simulink

Det er implementert totalt to PI-regulatorer, en for å regulere d -strøm og en for q -strøm. Utgangen fra regulatorene er to dq -spenninger som invers-transformeres og påtrykkes motoren. dq -strømmene reguleres ved å påtrykke ulike spenninger på motorfasene. Settpunktet til d -strømregulatoren settes til 0, ettersom denne ikke vil bidra til å produsere moment i motoren, som forklart i delkapittel 2.3.1. q -strømmen er dermed den eneste strømmen det er ønskelig å regulere til en ikke-null verdi. Settpunktet til denne kommer fra utgangen til posisjonsregulatoren og styrer direkte hvor stort moment motoren produserer.

Begrensninger

Vekselretterens maksimale utgangsspenning avhenger av størrelsen til DC-linkspenningen. For en trefase vekselretter med DC-linkspenning U_{dc} er den teoretisk høyeste utgangsspenningen U_{max} , vist i likning 4.20.

$$U_{max} = \frac{U_{dc}}{\sqrt{3}} \quad (4.20)$$

Summen av dq -spenningene må dermed ikke overstige U_{max} . Dersom regulatorene forsøker å overstige denne vil sinussignalene som vekselretteren lager klippes. For å sørge for at strømregulatorene ikke genererer et pådrag som fører til høyere spenning enn U_{max} er det derfor satt begrensninger i pådragene. Totalspenningens amplitude kan uttrykkes som

$$U_{max}^2 = u_q^2 + u_d^2 \quad (4.21)$$

Ved å sette likning 4.20 inn i likning 4.21 kan det maksimale pådraget til q -regulatoren kan gi, dermed uttrykkes som en funksjon av pådraget til d -strømregulatoren:

$$u_q = \sqrt{\frac{U_{dc}^2}{3} - u_d^2} \quad (4.22)$$

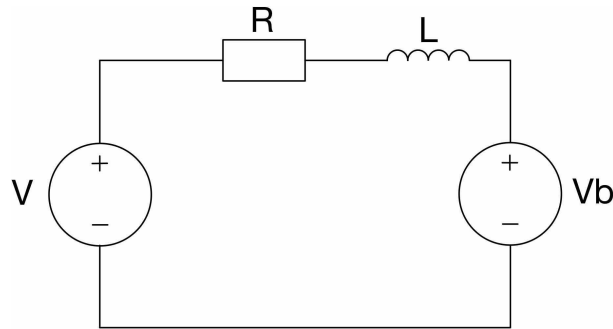
Dette er implementert i simulink ved å skrive en matlab-funksjon, som vist i figur 4.44.

```
1 function y = fcn(V_DC,u)
2 y = sqrt((V_DC/sqrt(3))^2-u^2);
```

Figur 4.44: Pådragsbegrensning i dq -regulatorer i Simulink

Elektrisk modell og tuning

For motoren ble det laget en forenklet elektrisk modell for å analysere systemet og tune strømregulatorene. Da motorstrømmene er dekomponert til to tidsinvariante strømmer, kan motoren modelleres omtrent som en DC-motor med børster. Figur 4.45 viser hvordan en av motorens faser kan forenkles ved å se på den påtrykte spenningen V , fasemotstanden R , faseinduktansen L og den motinduserte spenningen V_b



Figur 4.45: Forenklet elektrisk modell av BLDC

Den motinduserte spenningen kan for analysens del neglisjeres, da $\frac{dV}{dt}$ er mye mindre enn $\frac{di}{dt}$ [31]. Overføringsfunksjonen fra fasespenning til fasestrøm kan dermed uttrykkes som i likning 4.23

$$h_{m,el}(s) = \frac{i(s)}{u(s)} = \frac{1}{R + Ls} = \frac{\frac{1}{R}}{1 + \frac{L}{R}s} \quad (4.23)$$

Ved å regulere systemet ved bruk av en PI-regulator vil den åpne sløyfefunksjonen $h_0(s)$ bli som vist i likning 4.24

$$h_0(s) = h_{PI}(s) \cdot h_{m,el}(s) = K_p \frac{1 + T_i s}{T_i s} \cdot \frac{\frac{1}{R}}{1 + \frac{L}{R}s} \quad (4.24)$$

Dersom T_i settes lik $\frac{L}{R}$ vil polen til $h_{m,el}$ strykes, og reduseres til et førsteordens system. likning 4.24 reduseres dermed til

$$h_0(s) = \frac{K_p}{Ls} \quad (4.25)$$

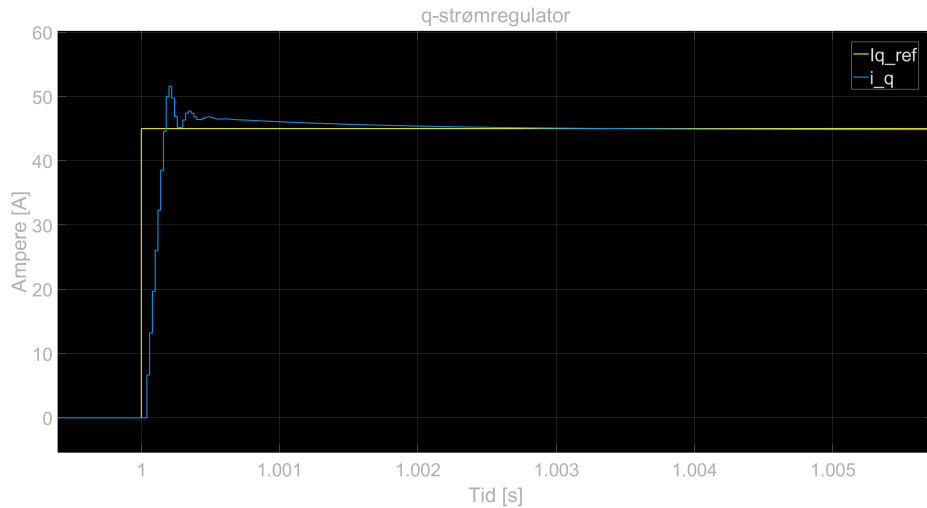
Overføringsfunksjonen til det tilbakekoblede systemet blir dermed

$$h_{sløyfe} = \frac{h_0}{1 + h_0} = \frac{\frac{K_p}{Ls}}{1 + \frac{K_p}{Ls}} = \frac{1}{1 + \frac{L}{K_p}s} \quad (4.26)$$

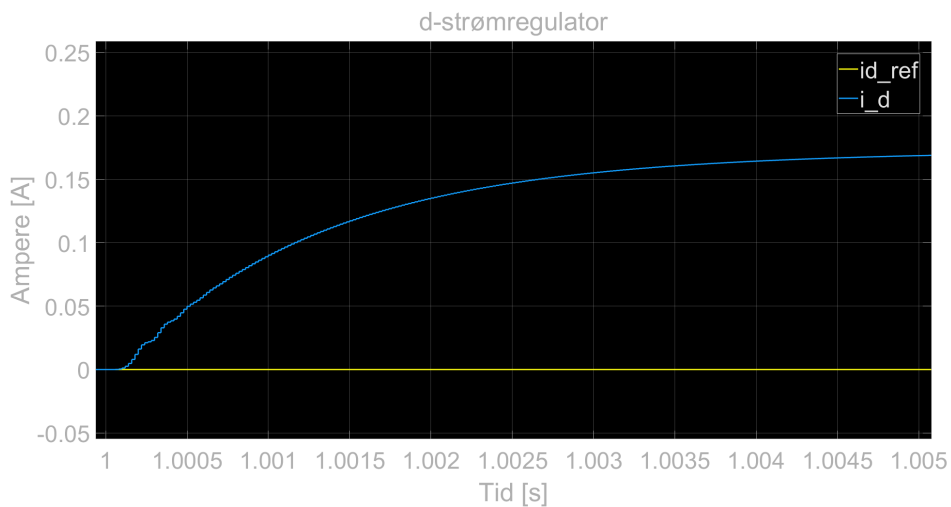
K_p kan dermed settes lik $L\omega_0$, der ω_0 er båndbredden til det regulerede systemet. Båndbredden ble her valgt til $\frac{1}{10}$ av svitsjefrekvensen til vekselretteren (50 kHz). Det regulerede systemet er

dermed stabilt ved frekvenser opp til $5kHz$.

dq -strømregulatorne innstilles dermed til $K_p = L \cdot 5000$ og $T_i = \frac{L}{R}$, der L er faseinduktans og R er fasemotstand. Figur 4.46 og 4.47 viser resultatet til q -strømregulatoren ved en sprangrespons på 30° på hjulvinkelen til det komplette systemet.



Figur 4.46: q -strømregulator ved sprangrespons på komplett system

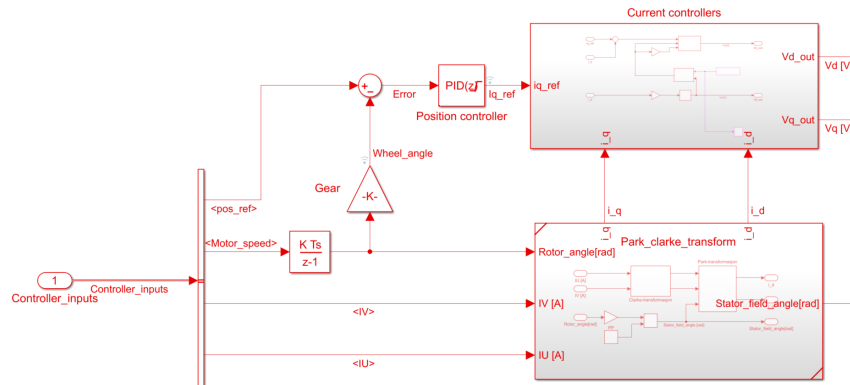


Figur 4.47: d -strømregulator ved sprangrespons på komplett system

Det kan ses av plottene at q -strømregulatoren holder referanseverdien etter kort tid. d -strømmen holdes også minimal, med en toppverdi på $0.172A$ ved oppstart av motoren. Ved videre simulering holdes denne tilnærmet null.

4.7.4 Posisjonsregulator

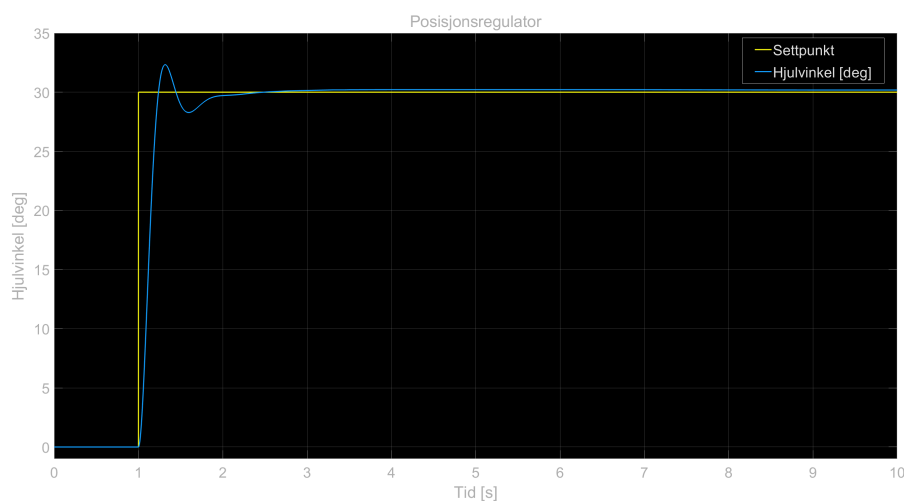
Posisjonsregulatoren i reguleringsystemet er plassert i kaskade før q -strømregulatoren for å regulere q -strømmen i stator, slik at motoren roterer til ønsket posisjon. Det ble her valgt å implementere en diskret PID-regulatorblokk fra Simulinks bibliotek som vist i figur 4.48.



Figur 4.48: Posisjonsregulator i Simulink

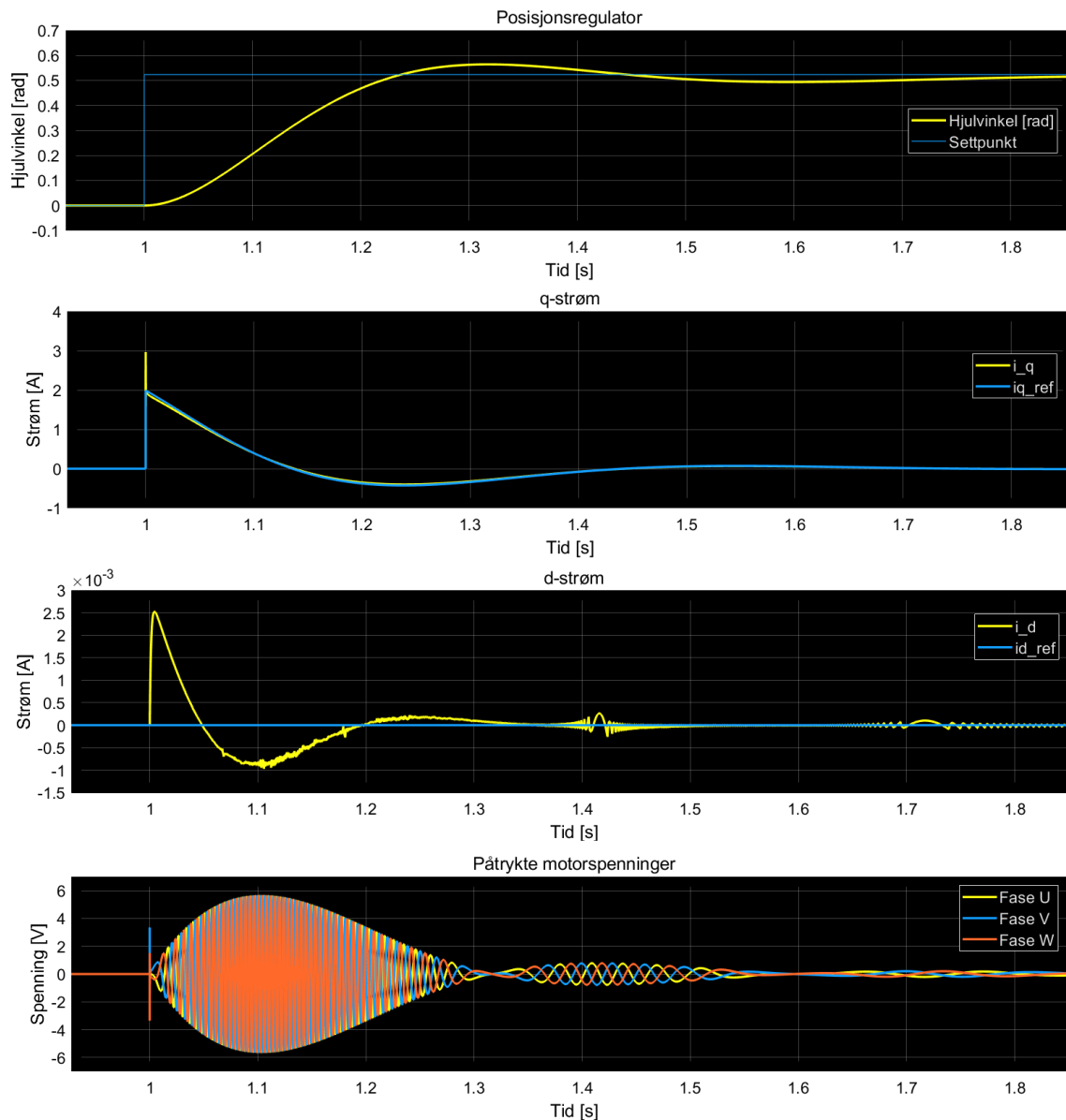
Regulatoren tar inn settpunkt fra et eksternt signal og bruker rotorposisjonen fra motorblokken for å beregne avviket. Ettersom utgangen til motorblokken gir rotorhastighet, blir hastigheten integrert og multiplisert med girforholdet R_G fra tabell 3.2, for å beregne rotorvinkelen.

Det ble vurdert å tune regulatoren analytisk på samme måte som for strømregulatorene. Ettersom reguleringsystemet ikke skulle implementeres på mikrokontrolleren og tiden var knapp, ble dette imidlertid valgt å ikke gjøre. Posisjonsregulatoren ble i stedet innstilt manuelt ved å påtrykke sprang i referanseposisjonen og justere parametrene P , I og D for å få ønsket innsvingning. Resultatet av manuell innstilling av posisjonsregulatoren kan ses i figur 4.49. I denne simuleringen ble $P = 0.8$, $I = 0.05$ og $D = 0.3$.



Figur 4.49: Posisjonsregulator ved sprang på 30°

Figur 4.50 viser sammenlagte simuleringresultater for posisjonsregulator, strømregulator og påtrykte motorspenninger fra motorkontrolleren, etter et sprang på 30° på hjulvinkelen. Umiddelbart etter spranget øker q -strømmen, før den gradvis synker og stabiliserer seg rundt null. Samtidig ser man at d -strømmen holdes minimal, og maksimalt er 3 mA . Fase-spenningene er først sinusoidale, ettersom dette fører til at motoren roterer. Etter cirka 1.3 sekunder endrer motoren retning, som kan ses ved at fase U og V bytter rekkefølge.



Figur 4.50: Simuleringsresultater ved sprangrespons på hjulvinkel

4.7.5 Modelleringsformat og kodegenerering

Et av de mer krevende aspektene ved modelleringen av det regulerte systemet har vært å lage modellen på et format som tillater å generere C-kode for reguleringsystemet. Ved å kodegenerere lager Matlab reguleringsystemet i C-kode som kan kjøres på en ekstern mikrokontroller. Resultatet er flere *.c* og *.h*-filer som kan inkluderes i kodeprosjektet for mikrokontrolleren. Koden kan da bruke den genererte koden ved å først kjøre en initialiseringsfunksjon én gang, og så kalle på modellfunksjonen ved hver sykel. Initialiseringsfunksjonen initialiserer modellen ved å definere parametre, funksjoner og variabler. Modellfunksjonen tar inn argumenter som er gitt i Simulink-modellen og regner ut resultatet fra modellen for ett tidssteg. Returverdien vil være de som defineres i modellen. I dette systemet har funksjonen ett argument - structen *Controller_inputs*. Returverdien fra funksjonen er en struct *Controller_outputs* med de tre fasespenningene som påtrykkes motoren. Det kan raskt ses at dette har flere fordeler:

Lettere implementering

Å implementere det digitale reguleringsystemet ved å skrive kode selv krever mye utviklingstid. Ettersom det i tillegg er mange individuelle systemer som samhandler, kan koden raskt bli stor og uoversiktlig. Ved å generere kode for systemet kan det derfor spares mye tid.

Mer robust kode

Når kode skrives manuelt kan man fort introdusere feil som fører til at systemet ikke fungerer. Jo større koden er, desto vanskeligere er det også å feilsøke for å rette opp feil. Når modellen kodegenereres unngås dette helt, siden den genererte koden er laget systematisk av en datamaskin. Ulempen er derimot at koden struktureres på en måte som ikke er like intuitiv å forstå.

Bedre versjonshåndtering

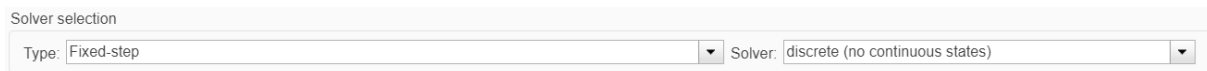
En utfordring når et digitalt reguleringsystem brukes på et fysisk system, samtidig som det utvikles og modelleres, er at man er nødt til å sørge for likt oppsett på mikrokontrolleren og modellen. Dersom noe endres i modellen det simuleres med, må dette også endres koden. Når systemet utvikles på denne måten kan det dermed fort oppstå ulikheter mellom det fysiske systemet og systemet det modelleres med. Ved å kodegenerere kan man sikre at det digitale reguleringsystemet alltid kjører med samme versjon som det modelleres med. Dersom det gjøres endringer, kan modellen raskt og enkelt kodegenereres på nytt og implementeres på mikrokontrolleren. Ved å i tillegg bruke verktøy som Git kan man dermed ha full kontroll over hvilken versjon som kjører på det fysiske systemet og være sikker på at det er likt som det modelleres med.

Den største utfordringen ved kodegenerering er imidlertid at det krever at modellen settes opp på en spesifikk måte i Simulink. Det må blant annet tas spesielt hensyn til

- Samplingstid
- Datatyper
- Innganger til subsystemet som kodegenereres
- Parameterinitialisering

Samplingstid

I Simulink vil måten utregningene gjøres på bestemmes av hvilken *solver* som brukes. Standardinnstillingen for solveren i Simulink er satt til *variable step*, hvilket betyr at programmet dynamisk velger hvor ofte utregningene gjøres basert på blokkene som brukes og hvor raskt verdiene endrer seg. Dersom blokker har ulik utregningstid løser Simulink dette automatisk uten at det gir feilmeldinger til brukeren. Dette kan imidlertid skape problemer når systemet implementeres, ettersom reguleringsystemet skal kjøres på en digital mikrokontroller. Der vil utregninger gjøres sekvensielt med en fast frekvens. Det er derfor viktig å simulere hele systemet diskret, der alle blokkene i subsystemet som skal kodegenereres har lik samplingstid. I modellen er derfor solveren satt til å være diskret med fast utregningsfrekvens, som vist i figur 4.51.



Figur 4.51: Innstilling for solver i Simulink

Datatyper

Simulink er laget for å enkelt kunne simulere dynamiske systemer, og datatypene som brukes i modellen kan derfor ofte ses bort fra. Simulink løser automatisk konvertering mellom ulike datatyper og det sjeldent behov for å ta hensyn til dette. I C-programmering stilles det imidlertid strengere krav til databehandling, og datatyper må derfor spesifiseres i modellen for at det skal kunne kodegenereres. I modellen er det derfor utelukkende brukt datatypen *single*, som opptar 4 bytes i minnet. Når koden genereres blir disse deklarerert som datatypen *float*.

Innganger til subsystem

Inngangene til subsystemet som kodegenereres bestemmer hvordan den genererte koden kan brukes av koden i programmet. Inngangsverdiene til modellen kan blant annet defineres som globale variabler som oppdateres andre steder i programmet. Et annet alternativ er

å definere inngangene som argumenter i en funksjon. Modellen som ble laget for dette systemet er satt opp til å definere inngangene i modellen som argumenter i en funksjon. Dette gir en bedre oversikt over hvor modellfunksjonen får argumentene fra.

Parameterinitialisering

Når det skal genereres kode for et subsystem er det viktig å spesifisere hvordan ulike parametre og signaler i modellen skal behandles. Standardinnstillingen i Simulink er å sette parametre direkte inn i koden som tall og ikke variabler. Når koden genereres vil det dermed være vanskelig å finne igjen verdier som er brukt i modellen. Regulatorparametre som forsterkning og integraltid er for eksempel nyttig å kunne endre raskt, uten å måtte generere ny kode. Det er derfor spesifisert at visse parametre skal være innstillbare, hvilket resulterer i at de lagres som variabler i den genererte koden, fremfor å bli flettet inn i modellkoden som tall. Figur 4.52 viser eksempelvis hvordan den genererte koden har definert regulatorparametrene for posisjonskontrolleren. Figur 4.53 viser hvordan parametrene brukes i koden for å begrense pådraget fra posisjonsregulatoren.

```
67 #ifndef DEFINED_TYPEDEF_FOR_pos_
68 #define DEFINED_TYPEDEF_FOR_pos_
69
70 typedef struct {
71     real32_T P;
72     real32_T I;
73     real32_T D;
74     real32_T max;
75     real32_T min;
76 } pos;
```

Figur 4.52: Generert kode for definering av regulatorparametre

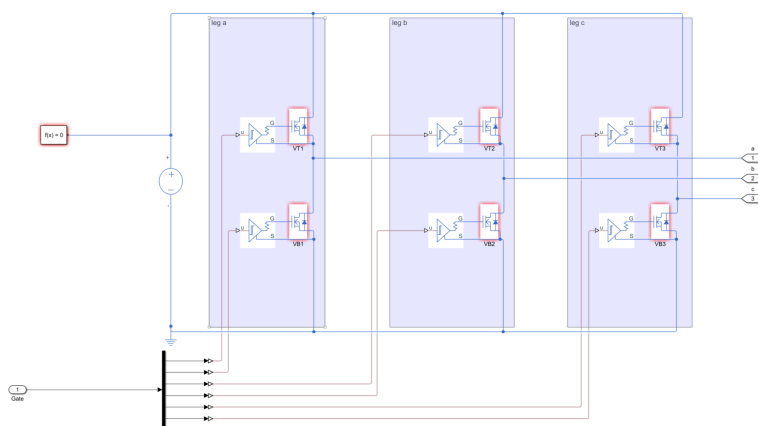
```
154 /* Saturate: '<S146>/Saturation' */
155 if (rtb_Sum > control_parameters.controller_parameters.pos.max) {
156     rtb_Sum = control_parameters.controller_parameters.pos.max;
157 } else {
158     if (rtb_Sum < control_parameters.controller_parameters.pos.min) {
159         rtb_Sum = control_parameters.controller_parameters.pos.min;
160     }
161 }
```

Figur 4.53: Generert kode for begrensnig av pådrag fra posisjonsregulator

4.7.6 Arbeidsflyt

Modellen beskrevet i delkapitlene ovenfor er resultatet av flere iterasjoner. Gjennom prosjektperioden har gruppen brukt mye tid på ulike metoder for å sette opp modellen, da det ikke umiddelbart var tydelig hvilken fremgangsmåte som ville fungere best. Innledningsvis ble for eksempel Simscape-pakken i Simulink brukt for å modellere vekselretteren og motoren. Simscape skiller seg fra vanlig Simulink-modellering ved at man lager fysiske kretser der signaler ikke har en definert retning. Oppsettet kan sammenlignes med å lage kretstegninger og simulere med dette.

Fordelen ved å bruke Simscape er at vekselretteren kan modelleres ved å sette opp transistorer i tre halvbro-konfigurasjoner slik at dynamikken til denne også representeres i modellen. Ulempen var derimot at modellen raskt ble kompleks med mange ulike signaler og innstillinger å ta hensyn til. En viktig faktor som resulterte i at Simscape ble lagt vekk er at simuleringene tok svært lang tid å utføre. Ved Simulering må det finnes en balanse mellom hvor raskt man ønsker å simulere opp mot hvor nøyaktige resultater som kreves. Med Simscape var denne balansen vanskelig å finne, da oppløsningen til resultatene enten ble for lav, eller simuleringene tok for lang tid å gjennomføre. For å oppnå akseptabel oppløsning tok det for eksempel ett minutt å simulere 0.1 sekunder i modellen. Figur 4.54 viser hvordan vekselretteren ble implementert ved å bruke Simscape i en tidligere versjon av modellen.



Figur 4.54: Vekselretter i Simulink ved bruk av Simscape-pakken

I forbindelse med implementeringen av vekselretteren i Simulink var det også nødvendig å generere PWM-signaler for å styre denne. Tidlig i prosjektet begynte derfor å implementere romvektor-pulsbreddemodulasjon. Denne metoden er mer effektiv enn sinusoidal pulsbreddemodulasjon, ettersom den reduserer momentripping og varmeutvikling i vekselretteren [32]. Metoden skiller seg fra sinusoidal pulsbreddemodulasjon ved at tiden hver transistor er på beregnes individuelt basert på de tre fasespenningene fra regulatoren.

Konklusjon

5.1 Oppsummering og konklusjon

I løpet av prosjektet har det blitt designet og produsert en trefase vekselretter og det har blitt laget et reguleringsystem for posisjonsregulering av hjulvinkel til en førerløs racerbil. Prosjektet har resultert i denne rapporten, der arbeid og resultater er dokumentert. Underveis i prosjektet har gruppen tatt valg på bakgrunn av systemkrav og hatt et overordnet fokus på videreutvikling. Arbeidet som har blitt gjort har resultert i en prototype som er kompatibel med Revolve NTNUs racerbiler, og som er laget i tråd med gjeldende standarder i organisasjonen.

Oppdragsgivers ønske for prosjektet var å utforske mulighetene rundt design av en egen vekselretter og posisjonsregulator for en synkronmotor, for regulering av hjulvinkelen til en førerløs racerbil. I forprosjektet satt derfor gruppen mål om å produsere et minimumsprodukt. Etter kort tid innså gruppen imidlertid at deres ambisjoner var for høye, sett i perspektiv med lengden og omfanget av prosjektet. Problemstillingen og målene ble derfor revurdert og tilpasset oppgavens omfang, samtidig som oppdragsgivers ønsker ble ivaretatt.

I forprosjektet var problemstillingen og målene formulert rundt implementasjon, men underveis i prosjektet har gruppens fokus blitt rettet mot design og tilrettelegging for videreutvikling. Prosjektgruppen anser ikke denne omstillingen som et nederlag, men mener tvert imot at dette har resultert i et mer gjennomført design.

Resultatet av prosjektet er en prototype som er tilrettelagt for at ulike funksjoner og reguleringsystem skal kunne testes videre. Ved å blant annet dele opp vekselretteren i to ulike kretskort, er det lagt opp til at disse skal kunne testes grundig, både individuelt og sammen. Kretskortene er bevisst laget i et stort format med flere testpunkter for å legge til rette for feilsøking. Komponenter og metoder som er brukt, er valgt ut ifra gjeldende standarder som brukes ellers i organisasjonen. Ved å gjøre dette, mener gruppen å ha senket terskelen for at medlemmer med andre ansvarsområder i organisasjonen, også kan bidra på dette systemet.

Reguleringssystemet som har blitt laget, tar utgangspunkt i det nåværende styresystemet til den førerløse racer bilen. Basert på dette har det blitt laget en modell av motoren og et reguleringssystem i Simulink, som er konfigurert til å generere C-kode direkte fra modellen. Ved å ha lagt grunnlaget for kodegenerering, har gruppen i stor grad tilrettelagt for at reguleringssystemet skal kunne videreutvikles og implementeres. Dette resulterer i et kontrollsystem som er lettere å implementere og teste, ettersom det ikke krever utvikling av programvare for det digitale reguleringssystemet.

Oppsummert mener gruppen at problemstillingen har blitt besvart på en god måte, prosjektets mål er oppnådd, og at arbeidet har resultert i et godt grunnlag for testing og videre utvikling.

5.2 Diskusjon

Gjennom prosjektperioden har flere valg blitt tatt som enten har endret prosjektets retning, eller som har hatt en stor påvirkning på arbeidet som har blitt utført. Disse valgene, i tillegg til prosjektet som helhet, vil bli reflektert over og diskutert i dette delkapittelet.

5.2.1 Prosjektets omfang

I retrospekt, syntes gruppen at problemstillingen de satt i forprosjektet var lite avgrensende. Problemstillingen dekket et bredt spekter av ulike fagområder, i tillegg til anvendelse og implementasjon. Etterhvert som gruppen innhentet informasjon fra ulike masteroppgaver og vitenskaplige artikler, innså gruppen at det ville være for tidskrevende å få til både design og implementasjon iløpet av prosjektperioden. Dersom den originale problemstillingen og de tilhørende målene skulle beholdes, var gruppen bekymret for at resultatet ikke ville tilfredsstille oppdragsgivers ønsker.

En viktig faktor som gruppen vurderte, var at dette systemet skulle videreutvikles av neste års medlemmer i Revolve NTNU. Gruppen konkluderte med at i dette tilfellet ville det være viktigere å prioritere kvalitet på arbeidet og sikre en god grunnbase for å legge til rette for videre utvikling. Fokuset i prosjektet ble derfor rettet mot å lage et godt design, fremfor å fremskynde implementasjon. Et av tiltakene som ble gjort, var å endre problemstillingen for at denne skulle reflektere dette.

Gruppen hadde opprinnelig planlagt å produsere to revisjoner av kretskortene, men dette ble endret til kun én revisjon. Ved å kun skulle produsere én revisjon, ble det frigjort tid til å kvalitetssikre og i større grad gjøre et grundig arbeid. Gruppen mener at avgjørelsen som ble tatt ga stor nytte for prosjektet, da det underveis ble avdekket flere kritiske feil i designet av kretskortene. Ved å endre problemstillingen og bruke mer tid på design, mener gruppen dette har bidratt til at kvaliteten på arbeidet totalt sett har blitt bedre.

5.2.2 Prosjektering

I og med at problemstillingen og målene for prosjektet ble endret, måtte også prosjektplanen endres. Gruppen kunne derfor ikke bruke Gantt-diagrammet som ble laget i forbindelse med forprosjektet. Det ble forsøkt å lage en ny og oppdatert prosjektplan, men grunnet vanskeligheter rundt design av skjematikk og utlegg for kretskort, ble tidsfrister stadig forskjøvet. Ofte ble nye feil funnet ved gjennomgang av produksjonsfiler, og gruppen syntes etterhvert at tidsfrister for når kretskort skulle være klar til produksjon, hadde liten verdi. Gruppen følte imidlertid at kvalitetsikring var strengt nødvendig, og at det var bedre å overskride tidsfristene, enn å produsere et kretskort med feil eller mangler.

Gruppen tar selvkritikk på at prosjekteringsdelen av prosjektet kunne vært mer gjennomført, men mener samtidig at det resulterende arbeidet er av høy kvalitet. Gruppen tror at dette kommer av et godt samarbeid gjennom hele prosjektperioden. Ved å blant annet ha gjennomført ukentlige møter og arbeidet sammen, har alle i gruppen til enhver tid vært oppdatert i prosjektforløpet. Når forsinkelser har oppstått har gruppen diskutert hvordan dette påvirker prosjektet, og tatt løpende vurderinger for videre fremgang.

5.3 Fremtidig arbeid

I løpet av prosjektperioden har gruppen oppdaget flere forbedringspotensialer ved prototypen som er laget og rundt arbeidet som er utført. I dette delkapittelet vil det diskuteres hva gruppen mener er de største forbedringspotensialene for videre utvikling, der temaene vil være rundt design av kretskort, modellering og implementasjon.

5.3.1 Kretskortdesign

Systemet som gruppen har produsert er en prototype hvor det er tatt bevisste designvalg for å forenkle feilsøking. I designfasen ble systemet delt opp i to kretskort, der mikrokontrollerdelen og kraftelektronikken ble adskilt. Gruppen har hatt et fokus på å designe en fungerende prototype og teste all funksjonalitet isolert. Størrelsen på kretskortene har derfor ikke vært noe gruppen har tatt hensyn til.

Dersom systemet skulle vært implementert på en racerbil ville strenge krav vært stilt til kretskortenes størrelse. Dette skyldes at det fysisk er lite plass til elektriske systemer og det er ønskelig at disse er så små som mulig. Vanlig praksis er å tidlig fastsette systemers volum og geometri, slik at det er sikkert at systemene passer på racerbilen etter produksjon. Gruppen mener at ved fremtidige revisjoner, er kretskortets designområde noe som bør fokuseres på, ettersom dette er kritisk for at systemet skal kunne implementeres.

Vurderingen til gruppen er at systemet bør være samlet på ett kretskort, der designområdet blir brukt godt ved å ta i bruk både topp- og bunnside av kretskortet. Et annet alternativ kan være å implementere kontrolldelen av systemet som en modul som plasseres på powerstage. Arealet til systemet vil i dette tilfellet reduseres, mot at høyden øker.

Som følge av at mikrokontrolleren setter alle pinnene logisk høy ved oppstart, kortsluttet transistor-kretsen under HIL-testen. I prosjektet kunne dette blitt unngått ved spennings-sette kontrollkortet først, og deretter powerstage. På racerbilen hadde dette imidlertid vært umulig, ettersom begge kortene ville blitt spenningsatt samtidig. For å forhindre dette må det designes et sikkerhetsnett som gjør det fysisk umulig å kortslutte transistorene på denne måten. For videreutvikling vil gruppen anbefale å undersøke en løsning ved å ta i bruk en vippe eller en logisk krets som sørger for at gatedriverne ikke skrur seg på ved oppstart.

Oppsummert anbefaler gruppen at Revolve NTNU i fremtiden bør se på å slå sammen systemet til ett kretskort, bruke designområdet godt ved å plassere komponentene på både topp- og bunnside av kretskortet, og gjøre det fysisk umulig å kortslutte transistorene ved oppstart.

5.3.2 Programvare

Programvaren som har blitt utviklet i dette prosjektet har hatt begrenset funksjonalitet, ettersom det i oppgaven har vært fokusert på design av kretskort og utvikling av en systemmodell. Det har blitt generert kode for motorkontrollalgoritmen, men denne koden har ikke blitt testet på systemet. Gruppen mener derfor at ved videreutvikling av systemet, bør det settes av tilstrekkelig med tid for å utvikle en solid kodebase og teste denne.

Gruppen har diskutert at en tilstandsmaskin kan være lurt å implementere. Det kan i tilstandsmaskinen lages ulike tilstander hvor aktuatoren eksempelvis følger et ønsket settpunkt, eller deaktiveres helt. Dette vil gi full kontroll over hvorvidt det autonome styresystemet er aktivt eller ikke.

5.3.3 Modellering

Gruppen mener at modellen som er laget inneholder alle nødvendige grunnfunksjoner, og selv om koden ikke har blitt testet, har de stor tro på at systemet vil fungere dersom det implementeres. Det er brukt mye tid på å validere modellen ved å simulere systemet, fremfor å implementere mer kompleks dynamikk. I tillegg har gruppen brukt mye tid på å tilrettelegge for kodegenerering og implementasjon.

Gruppen tror dette har vært mer verdifullt for organisasjonen, enn at det er laget et komplekst system som ikke er tilrettelagt for implementasjon. Ettersom det nå er lagt til rette for implementasjon, mener gruppen det fremover bør fokuseres på å:

- Analysere det mekaniske systemet og lage en modell for den dynamiske lasten på styre-aktuatoren
- Lage en mer virkelighetsnær motormodell
- Utbedre reguleringsystemet og legge til foroverkobling, sensordynamikk og vekselretterdynamikk

5.4 Gruppens refleksjon

Bacheloroppgaven er det emnet gruppemedlemmene mener har vært det mest lærerike og utfordrende på hele studiet. Oppgaven har gitt gruppen et innblikk i hvordan man går frem for å utvikle et produkt. Fra innhenting av relevant teori, til design, produksjon og testing. Gruppen har også fått en forståelse for hva som må til for et godt samarbeid i et prosjekt.

Før prosjektets start ble arbeidsoppgaver fordelt mellom gruppemedlemmene, men det har i løpet av hele prosjektet vært fokus på at bacheloroppgaven har vært en samlet oppgave der alle skal bidra. For eksempel når noen har stått fast, har dette blitt kommunisert til resten av gruppen slik at de kunne hjelpe til. Gruppen mener at god kommunikasjon og tilbakemeldinger har vært en viktig faktor for gruppens fremgang.

Fra starten visste gruppen at den valgte oppgaven var stor og omfattende, men gruppen var bestemt på at dette var mulig dersom alle la inn en god innsats og samarbeidet godt. På grunn av dette ble en forventningsavklaring holdt ved prosjektets start. Ved å ha felles mål og avklare forventninger til hverandre og seg selv, har dette drevet gruppen gjennom hele prosjektperioden.

Oppgaven har bydd på mange utfordringer der gruppemedlemmene har utfordret seg selv på ukjente fagområder. Simen, som studerer automatiseringsteknikk og aldri har designet et kretskort før, har stått ansvarlig for å designe elektronikken på kontrollkortet. Emanuela har hatt ansvar for design av kraftelektronikken, der hun måtte lære seg det grunnleggende innenfor fagområdet. Både Simen og Oscar hadde først ansvar for motorkontrolldelen, men etter at arbeidskraften måtte fordeles over til utvikling av elektronikk, tok Oscar fullt over. Oscar ferdigstilte reguleringsystemet for motorkontrolleren og satt seg også grundig inn i hvordan systemet kunne kodegenereres. Han brukte mye tid på å tilrettelegge for implementasjon, slik at det ved videreutvikling ville være lettere å implementere systemet. Da gruppen hadde sendt produksjonsfilene, ble det skrevet nødvendig programvare for å teste hovedfunksjonalitetene på kretskortene de hadde designet.

Ettersom oppgaven har omhandlet flere ulike fagområder som er tett knyttet opp mot hverandre, var det kritisk å få alle gruppemedlemmene opp på et nivå, for å få et fullstendig bilde av systemet de skulle designe sammen.

Tidlig i prosjektet merket gruppen viktigheten ved å jobbe sammen. På grunn av Covid-19 mistet gruppemedlemmene tilgang på Revolve NTNU sine lokaler. Gruppen presterte å møtes på skolen når det lot seg gjøre, men også være aktiv på digitale plattformer for å jobbe sammen. Rundt påsketider fikk heldigvis gruppen tilgang på et kontor som har vært til stor nytte. Da gruppen mistet tilgang til Revolve NTNUs lokaler, oppstod det utfordringer ved lodding og testing av kretskort, ettersom labutstyr var nødvendig. Gruppen løste dette ved å låne utstyr fra organisasjonen og loddet kretskortene hjemme.

Ved design og produksjon av kretskort kan det oppstå mange utfordringer. Feil i skjematikk, dårlig ruting av utlegg, defekte komponenter, produksjonsfeil, dårlig lodding og ESD-skader, er noen eksempler på dette. Dersom noen av disse problemene oppstår, risikerer man at produktet er defekt. Covid-19 har vært en faktor som har ført til forsinkelser av både mønsterkort og komponenter. Mangelen på halvlederkomponenter i verden i dag, er et kjent problem, og er noe som også har påvirket prosjektet. Utfordringene nevnt ovenfor, er noe gruppen gjennom hele prosjektperioden har måttet ta i betraktning.

Som nevnt er det mange faktorer som spiller inn i produksjonsløpet. Gjennom prosjektet har gruppen fått en bedre forståelse for hvor viktig planlegging, løpende vurderinger, kvalitets-sikring og kommunikasjon har å si for gjennomføringen av et prosjekt. Gruppen er stolt over prototypen de har designet, og mener denne gjenspeiler det harde arbeidet de har lagt inn. Ved videreutvikling av systemet, håper gruppen at Revolve NTNU vil ha stor nytte av arbeidet som er utført.

Kilder

- [1] Frode Høyte. *Elektrisk motor*.
https://snl.no/elektrisk_motor.
- [2] Pushek Madaan. *Brushless DC Motors – Part I: Construction and Operating Principles*.
<https://www.edn.com/brushless-dc-motors-part-i-construction-and-operating-principles/>.
- [3] Jian Zhao/Yangwei Yu. *Brushless DC Motor Fundamentals*.
https://www.monolithicpower.com/pub/media/document/Brushless_DC_Motor_Fundamentals.pdf.
- [4] Danielle Collins. *FAQ: What is sinusoidal commutation for dc motors?*
<https://www.motioncontroltips.com/what-is-sinusoidal-commutation-for-dc-motors/>.
- [5] Maxon group. *Parameters of maxon-EC-Motors Driven By Servo Amplifiers With Sinus-Commutation*.
https://support.maxongroup.com/hc/en-us/article_attachments/360008110913/SineCommutationMMC.pdf.
- [6] N.Undeland T.Mohan og W.Robbins. *Power Electronics(1995)*. John Wiley & Sons, Inc.
- [7] B.N.Trosæter. *Evaluation of Switching Characteristics, Switching Losses and Snubber Design for a full Half-Bridge Power Module(2016)*. Trondheim, Norway:Master thesis.
- [8] S.Li T.Chen og B.Fahimi. *Analysis of DC-Link Voltage Ripple in Voltage Source Inverters without Electrolytic Capacitor*.
https://studntnu-my.sharepoint.com/:b:/g/personal/oscarom_ntnu_no/EeX2DAfYIb9Co1TKTGj pzW4BonhhrTr8kqCDSY5A0fVm0g.
- [9] Texas Instruments. *Bootstrap Circuitry Selection for Half-Bridge Configurations*.
<https://www.ti.com/lit/an/slva887/slva887.pdf>.
- [10] Texas Instruments. *Clarke Transform*.
<https://www.ti.com/lit/an/bpra048/bpra048.pdf>.
- [11] Microsemi. *Park Transform*.
https://studntnu-my.sharepoint.com/:b:/g/personal/oscarom_ntnu_no/Eb99zucfvORMmsi1Pp7T7CABCljLt7qyoH_eSfr9uaHFKg.

- [12] **Jean Pollefliet.** *Current -, Angular Position -, Speed Transducers.*
<https://www.sciencedirect.com/topics/engineering/incremental-encoder>.
- [13] **Chris Woodford.** *Hall-effect Sensors.*
<https://www.explainthatstuff.com/hall-effect-sensors.html>.
- [14] **Maxon Group.** *Maximum Output Current.*
https://www.maxongroup.com/medias/sys_master/root/8815460712478/DC-EC-Key-Information-14-EN-42-50.pdf?attachment=true.
- [15] **Texas Instruments.** *TPS54560B.*
https://www.ti.com/lit/ds/symlink/tps54560b.pdf?ts=1621067025629&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FTPS54560B.
- [16] **Analog Devices.** *LTS1129-3.3.*
<https://no.mouser.com/datasheet/2/609/112935ff-1269715.pdf>.
- [17] **Infineon.** *BSC016N06NS.*
https://no.mouser.com/datasheet/2/196/Infineon-BSC016N06NS-DataSheet-v02_05-EN-1226200.pdf.
- [18] **Infineon.** *IRLR2908PBF.*
<https://no.mouser.com/datasheet/2/196/irlr2908pbf-1732803.pdf>.
- [19] **ON Semiconductor.** *FDPF045N10A.*
<https://www.onsemi.com/pdf/datasheet/fdpf045n10a-d.pdf>.
- [20] **VISHAY.** *IRLZ44PBF.*
<https://no.mouser.com/datasheet/2/427/sihlz44-1768805.pdf>.
- [21] **VISHAY.** *SQJA80EP.*
<https://www.vishay.com/docs/75519/sqja80ep.pdf>.
- [22] **Texas Instrument.** *UCC27288.*
https://www.ti.com/lit/ds/symlink/ucc27288.pdf?ts=1615281883785&ref_url=https%253A%252F%252Fno.mouser.com%252F.
- [23] **Texas Instruments.** *INA240A1QPWRQ datablad.*
https://www.ti.com/lit/ds/symlink/ina240.pdf?ts=1621153721121&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FINA240.
- [24] **Microchip.** *Datablad ATSAME70N21B.*
<https://ww1.microchip.com/downloads/en/DeviceDoc/SAM-E70-S70-V70-V71-Family-Data-Sheet-DS60001527E.pdf>.
- [25] **Microchip.** *Atmel Ice.*
<https://www.microchip.com/DevelopmentTools/ProductDetails/ATATMEL-ICE>.

- [26] Maxon Group. *Encoder 16 EASY XT*.
https://www.maxongroup.com/medias/sys_master/root/8846492205086/EN-451-452.pdf.
- [27] Texas Instruments. *RS-422 transceiver*.
<https://www.ti.com/lit/ds/symlink/am26lv32.pdf?HQS=dis-dk-null-digikeymode-dsf-pf-null-ww&ts=1620201049944>.
- [28] Den internasjonale standardiseringsorganisasjonen. *ISO 11898-1:2015*.
<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=796309>.
- [29] Mathworks. *Interior PMSM*.
<https://se.mathworks.com/help/autoblks/ref/interiorpmsm.html>.
- [30] Walter Schmid. *Motor data and simulation*.
<https://support.maxongroup.com/hc/en-us/articles/360013761160-Motor-data-and-simulation>.
- [31] Texas instruments Dave Wilson. *Teaching your PI controller to behave - part II*.
https://e2e.ti.com/blogs_/b/industrial_strength/posts/teaching-your-pi-controller-to-behave-part-ii.
- [32] Yngve Solbakken. *Space vector PWM intro*.
<https://www.switchcraft.org/learning/2017/3/15/space-vector-pwm-intro>.

Vedlegg

Forventningsavklaring - bacheloroppgave

Denne forventningsavklaringen ble holdt 7. januar i bachelorgruppen for å avklare hvilke forventninger vi stiller til oss selv og gruppen.

Tilgjengelighet:

- Alle skal være tilgjengelig på Slack i døgnets våkne timer.
- Gruppen skal møtes minst 3 ganger i uken for å jobbe sammen. Utover det skal alle arbeide med oppgaven selvstendig.
- Gruppen er samstemte om at det er greit å ringe hverandre når som helst for å diskutere oppgaven.
- Dersom veileder kontaktes skal gruppen holdes oppdatert på korrespondansen.
- Dersom veileder kontakter gruppen skal det sendes svar senest 24 timer etter.

Møter:

- Alle skal prioritere møter med gruppen og veileder foran andre møter som måtte oppstå.
- Dersom et møte må utsettes skal det ikke utsettes lengre enn 48 timer etter original møtetid.
- Etter hver periode skal gruppen ha et møte der alle oppdateres på status i prosjektet og formidler relevant informasjon. I tillegg skal det presenteres en plan for arbeidet fremover.

Arbeidsmiljø:

- Alle skal sørge for at det blir et godt arbeidsmiljø i gruppen.
- Gruppemedlemmene skal være ærlige overfor hverandre og si ifra dersom noen oppfører seg ufint, jobber for lite eller ikke leverer arbeid med høy nok kvalitet.

Tidsfrister:

- Alle har ansvar for å opplyse gruppen dersom man ikke rekker en tidsfrist eller ligger an til å falle bak tidsskjema.
- Alle har ansvar for å oppdatere gruppen når en tidsfrist er nådd og man begynner å arbeide videre.

Arbeidstid:

- Gruppen skal arbeide sammen på Gløshaugen fra 08:15-14:00 minst tre dager i uken og på Revolve-kontoret 2 dager i uken. Utover dette skal samtlige jobbe individuelt, men kan da velge hvor man jobber.

Karaktermål:

- Alle i gruppen skal gjøre sitt aller beste og har som mål å oppnå karakteren A.

Tilbakemeldinger:

- Alle skal gi hverandre ærlige tilbakemeldinger på arbeidet som utføres. Dette gjelder spesielt oppgaveskriving, da formidlingen av oppgaven er svært viktig.
- Tilbakemeldinger skal være konstruktive og bidra til at prosjektet blir bedre.
- Dersom noen er misfornøyd med andres arbeid må det meldes ifra om.

Signaturer:

Emanuela Tran

Emanuela Tran \ Trondheim, 15.01.2021

Simen Bergsvik

Simen Bergsvik \ Trondheim, 15.01.2021

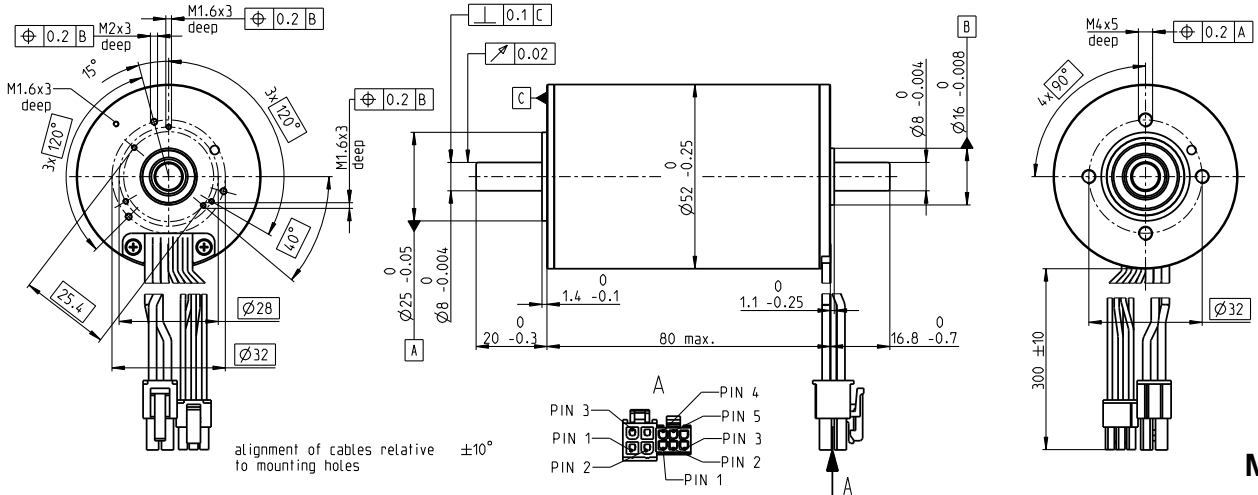
Oscar Øgar Meisal

Oscar Ø. Meisal \ Trondheim, 15.01.2021

EC-i 52 Ø52 mm, brushless, 180 Watt

High Torque

EC-i



M 1:2

- Stock program
- Standard program
- Special program (on request)

Part Numbers

with Hall sensors

574740 574741 579164 579165

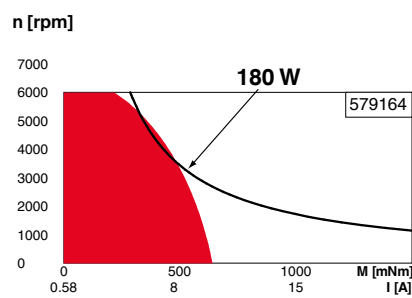
Motor Data (provisional)

Values at nominal voltage		18	24	36	48
1 Nominal voltage	V	18	24	36	48
2 No load speed	rpm	4820	4680	4820	4900
3 No load current	mA	1010	726	507	390
4 Nominal speed	rpm	4360	4200	4360	4450
5 Nominal torque (max. continuous torque)	mNm	388	428	438	412
6 Nominal current (max. continuous current)	A	11.1	8.81	6.18	4.47
7 Stall torque ¹	mNm	11500	13000	15900	15700
8 Stall current	A	325	268	225	169
9 Max. efficiency	%	89.3	90	90.8	90.7
Characteristics					
10 Terminal resistance phase to phase	Ω	0.0555	0.0894	0.16	0.284
11 Terminal inductance phase to phase	mH	0.0643	0.122	0.257	0.443
12 Torque constant	mNm/A	35.3	48.6	70.6	92.7
13 Speed constant	rpm/V	270	197	135	103
14 Speed/torque gradient	rpm/mNm	0.425	0.362	0.306	0.316
15 Mechanical time constant	ms	0.756	0.645	0.544	0.562
16 Rotor inertia	gcm ²	170	170	170	170

Specifications

- Thermal data**
- 17 Thermal resistance housing-ambient 4.32 K/W
 - 18 Thermal resistance winding-housing 0.63 K/W
 - 19 Thermal time constant winding 10.2 s
 - 20 Thermal time constant motor 1780 s
 - 21 Ambient temperature -40...+100°C
 - 22 Max. winding temperature +155°C
- Mechanical data (preloaded ball bearings)**
- 23 Max. speed 6000 rpm
 - 24 Axial play at axial load < 15 N 0 mm
 - > 15 N 0.14 mm
 - 25 Radial play preloaded
 - 26 Max. axial load (dynamic) 12 N
 - 27 Max. force for press fits (static) (static, shaft supported) 150 N
 - 28 Max. radial load, 5 mm from flange 6000 N
 - 110 N

Operating Range



Comments

- **Continuous operation**
In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient.
= Thermal limit.
- Short term operation**
The motor may be briefly overloaded (recurring).
- **Assigned power rating**

Other specifications

- 29 Number of pole pairs
- 30 Number of phases
- 31 Weight of motor

Values listed in the table are nominal.

Connection motor (Cable AWG 16)

- red Motor winding 1 Pin 1
- black Motor winding 2 Pin 2
- white Motor winding 3 Pin 3
- N.C. Pin 4

Connector Article number

Molex 39-01-2040

Connection sensor (Cable AWG 26)

- yellow Hall sensor 1 Pin 1
- brown Hall sensor 2 Pin 2
- grey Hall sensor 3 Pin 3
- blue GND Pin 4
- green V_{Hall} 4.5...24 VDC Pin 5
- N.C. Pin 6

Connector Article number

Molex 430-25-0600

Wiring diagram for Hall sensors see p. 49

¹Calculation does not include saturation effect (p. 61/168)

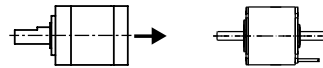
maxon Modular System

Details on catalog page 36

83 g

Planetary Gearhead

Ø52 mm
4 - 30 Nm
Page 402



Recommended Electronics:

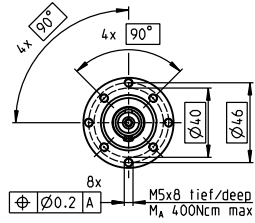
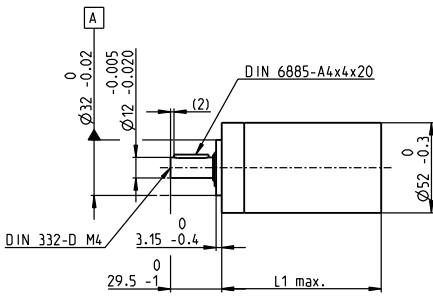
- Notes Page 36
- ESCON Mod. 50/8 (HE) 488
- ESCON 70/10 489
- EPOS4 Mod./Comp. 50/8 497
- EPOS4 Mod./Comp. 50/15 497
- EPOS4 70/15 501

- Encoder 16 EASY/XT**
128 - 1024 CPT, 3 channels
Page 450/452
- Encoder 16 EASY Absolute/XT**
4096 steps
Page 454/456
- Encoder 16 RIO**
1024 - 32768 CPT, 3 channels
Page 467
- Encoder AEDL 5810**
1024 - 5000 CPT, 3 channels
Page 470
- Encoder HEDL 5540**
500 CPT, 3 channels
Page 477

Planetary Gearhead GP 52 C $\varnothing 52$ mm, 4.0–30.0 Nm

Ceramic Version

gear



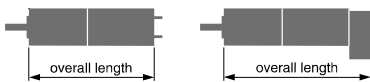
M 1:4

Technical Data

Planetary Gearhead	straight teeth
Output shaft	stainless steel
Bearing at output	preloaded ball bearings
Radial play, 12 mm from flange	max. 0.06 mm
Axial play at axial load	< 5 N 0 mm > 5 N max. 0,3 mm
Max. axial load (dynamic)	200 N
Max. force for press fits	500 N
Direction of rotation, drive to output	=
Max. continuous input speed	6000 rpm
Recommended temperature range	-15...+80°C
Extended range as option	-40...+100°C
Number of stages	1 2 3 4
Max. radial load, 12 mm from flange	420 N 630 N 900 N 900 N

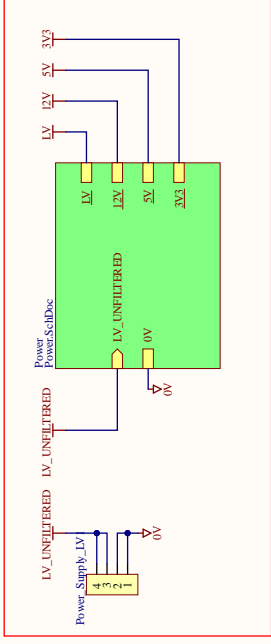
- Stock program
- Standard program
- Special program (on request)

Gearhead Data	Part Numbers						
	223080	223083	223089	223094	223097	223104	223109
1 Reduction	3.5:1	12:1	43:1	91:1	150:1	319:1	546:1
2 Absolute reduction	2/3	49/4	343/8	91	240/16	637/2	546
10 Mass inertia	gcm ² 20,7	17,6	17,3	16,7	17,3	16,8	16,4
3 Max. motor shaft diameter	mm 10	10	10	10	10	10	10
Part Numbers	223081	223084	223090	223095	223099	223105	223110
1 Reduction	4.3:1	15:1	53:1	113:1	186:1	353:1	676:1
2 Absolute reduction	13/5	91/6	637/12	338/3	4459/24	2857/61	676
10 Mass inertia	gcm ² 12	16,8	17,2	9,3	17,3	9,4	9,1
3 Max. motor shaft diameter	mm 8	10	10	8	10	8	8
Part Numbers		223085	223091	223096	223101	223106	223111
1 Reduction		19:1	66:1	126:1	230:1	394:1	756:1
2 Absolute reduction		189/9	1183/18	126	826/36	1183/3	756
10 Mass inertia	gcm ²	9,5	16,7	16,4	16,8	16,7	16,4
3 Max. motor shaft diameter	mm	8	10	10	10	10	10
Part Numbers		223086	223092	223098	223102	223107	223112
1 Reduction		21:1	74:1	156:1	257:1	441:1	936:1
2 Absolute reduction		21	147/2	156	1029/4	441	936
10 Mass inertia	gcm ²	16,5	17,2	9,1	17,3	16,5	9,1
3 Max. motor shaft diameter	mm	10	10	8	10	10	8
Part Numbers		223087	223093		223103	223108	
1 Reduction		26:1	81:1		285:1	488:1	
2 Absolute reduction		26	219/27		15379/54	4384/9	
10 Mass inertia	gcm ²	9,1	9,4		16,7	9,4	
3 Max. motor shaft diameter	mm	8	8		10	8	
4 Number of stages		1	2	3	3	4	4
5 Max. continuous torque	Nm	4	15	30	30	30	30
6 Max. intermittent torque at gear output	Nm	6	22,5	45	45	45	45
7 Max. efficiency	%	91	83	75	75	68	68
8 Weight	g	460	620	770	770	920	920
9 Average backlash no load	°	0,6	0,8	1,0	1,0	1,0	1,0
11 Gearhead length L1	mm	49,0	65,0	78,5	78,5	92,0	92,0

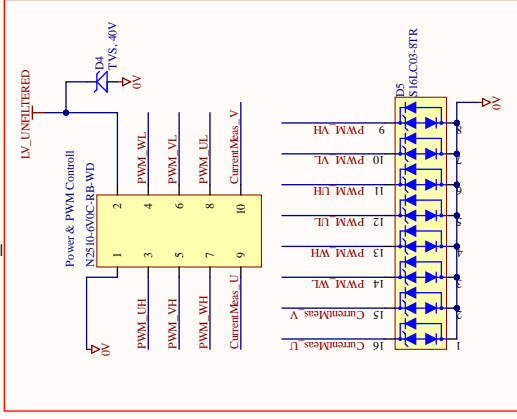


maxon Modular System											
+ Motor	Page	+ Sensor	Page	Brake	Page	Overall length [mm] = Motor length + gearhead length + (sensor/brake) + assembly parts					
RE 40, 150 W	141					120,1	136,1	149,6	149,6	163,1	163,1
RE 40, 150 W	141	MR	464			131,5	147,5	161,0	161,0	174,5	174,5
RE 40, 150 W	141	HED_5540	471/474			140,8	156,8	170,3	170,3	183,8	183,8
RE 40, 150 W	141	HEDL 9140	478			174,1	190,1	203,6	203,6	217,1	217,1
RE 40, 150 W	141			AB 28	519	156,2	172,2	185,7	185,7	199,2	199,2
RE 40, 150 W	141			AB 28	520	164,2	180,2	193,7	193,7	207,2	207,2
RE 40, 150 W	141	HED_5540	471/474	AB 28	519	173,4	189,4	202,9	202,9	216,4	216,4
RE 40, 150 W	141	HEDL 9140	478	AB 28	520	184,6	200,6	214,1	214,1	227,6	227,6
RE 50, 200 W	142					157,1	173,1	186,6	186,6	200,1	200,1
RE 50, 200 W	142	HED_5540	472/474			177,8	193,8	207,3	207,3	220,8	220,8
RE 50, 200 W	142	HEDL 9140	479			219,5	235,5	249,0	249,0	262,5	262,5
RE 50, 200 W	142			AB 44	524	219,5	235,5	249,0	249,0	262,5	262,5
RE 50, 200 W	142	HEDL 9140	479	AB 44	524	232,5	248,5	262,0	262,0	275,5	275,5
EC 40, 170 W	229					129,1	145,1	158,6	158,6	172,1	172,1
EC 40, 170 W	229	HED_5540	472/474			152,5	168,5	182,0	182,0	195,5	195,5
EC 40, 170 W	229	Res 26	481			156,3	172,3	185,8	185,8	199,3	199,3
EC 40, 170 W	229			AB 32	521	171,8	187,8	201,3	201,3	214,8	214,8
EC 40, 170 W	229	HED_5540	472/474	AB 32	521	190,2	206,2	219,7	219,7	233,2	233,2
EC 45, 150 W	230					160,3	176,3	189,8	189,8	203,3	203,3
EC 45, 150 W	230	HEDL 9140	478			175,9	191,9	205,4	205,4	218,9	218,9
EC 45, 150 W	230	Res 26	481			160,3	176,3	189,8	189,8	203,3	203,3
EC 45, 150 W	230			AB 28	520	167,7	183,7	197,2	197,2	210,7	210,7
EC 45, 150 W	230	HEDL 9140	478	AB 28	520	184,7	200,7	214,2	214,2	227,7	227,7
EC 45, 250 W	231					193,1	209,1	222,6	222,6	236,1	236,1
EC 45, 250 W	231	HEDL 9140	478			208,7	224,7	238,2	238,2	251,7	251,7
EC 45, 250 W	231	Res 26	481			193,1	209,1	222,6	222,6	236,1	236,1

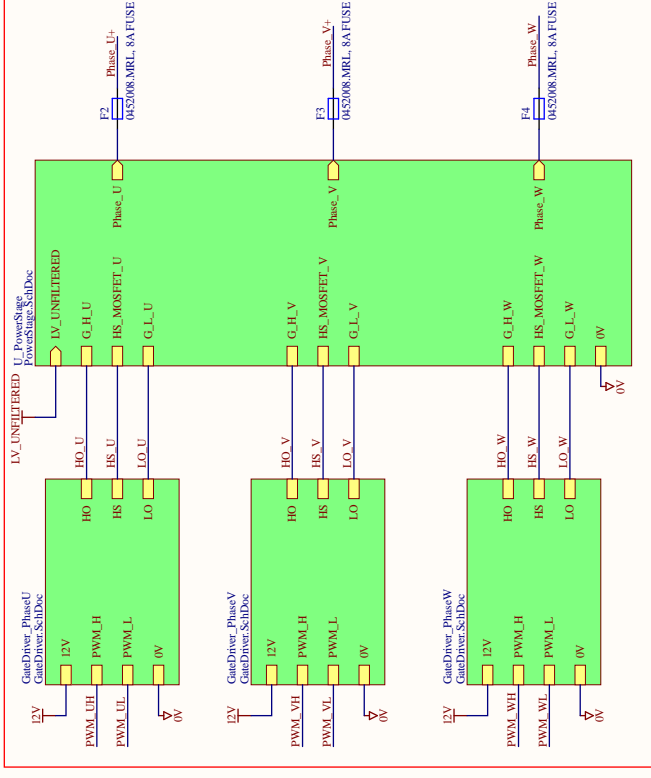
POWER



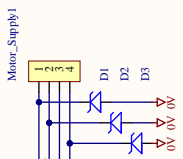
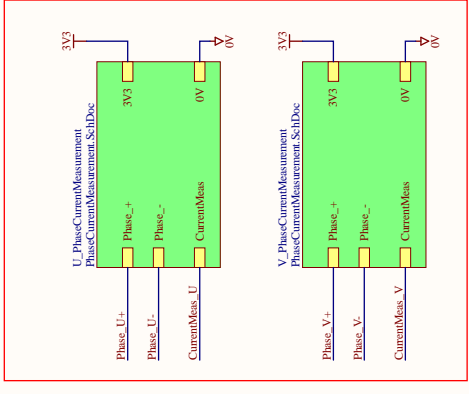
POWER&PWM_CONTROL



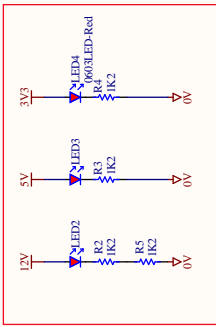
POWERSTAGE



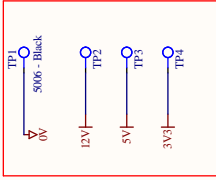
PhaseCurrentMeasurement



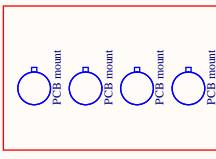
POWER LEDs



TEST POINTS



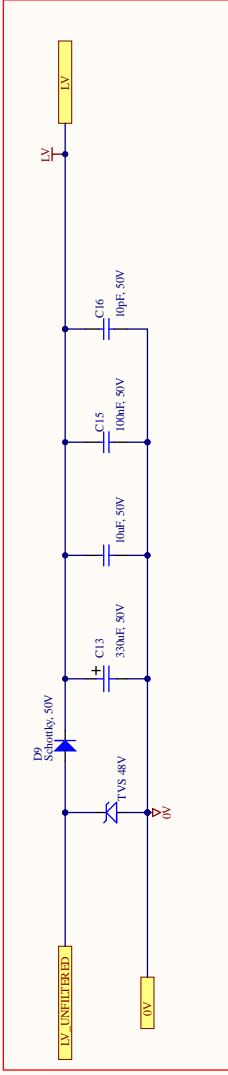
MOUNTING



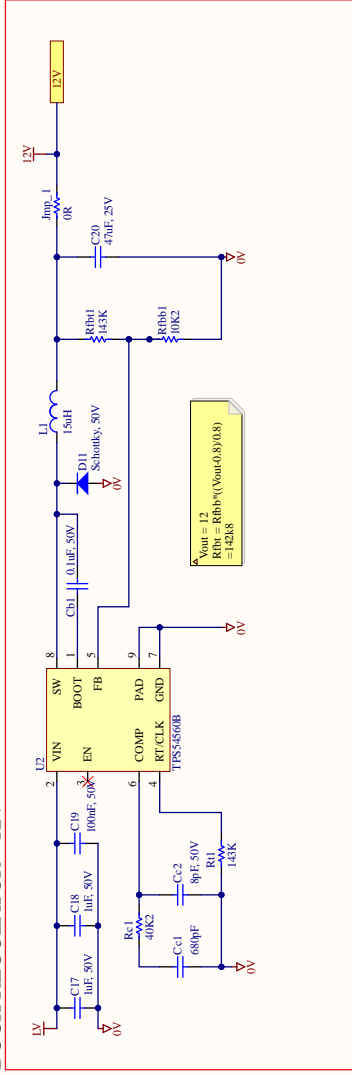
Project: R21_ACC_powerstage.PjtPcb		Sheet number: 1 out of 5	Date: 06-May-21
Title: Main.SchDoc		Designer: c75a058ae76f26783d78c819d459fad1ba6c7eabd	
Revision: e75a058ae76f26783d78c819d459fad1ba6c7eabd		Date: 06-May-21	



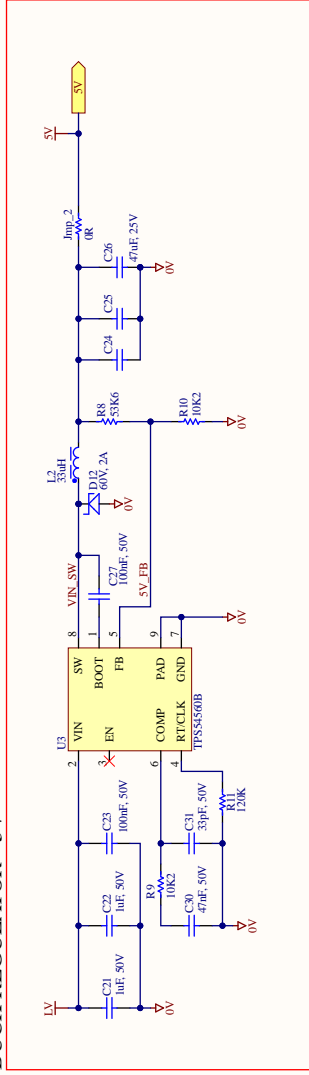
FILTER - LV



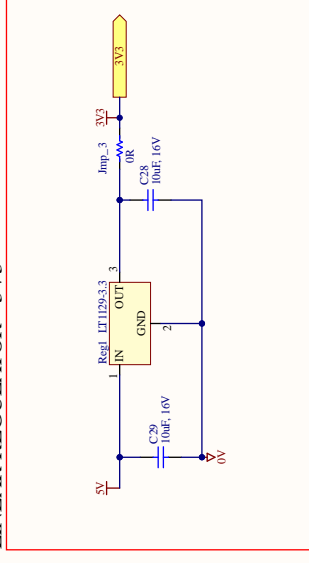
BUCK REGULATOR - 12V



BUCK REGULATOR - 5V



LINEAR REGULATOR - 3V3



Project:
R21_ACC_powerstage.PjtPcb

Title:
PowerSchDoc

Revision:
e75a058ae76f26783d78c819d4959ad1ba6c7a4fd

Sheet number:
2 out of 5

Date:
06-May-21





Project:
R21_ACC_powerstage_PjFeb

Title:
GateDriver_SchDoc

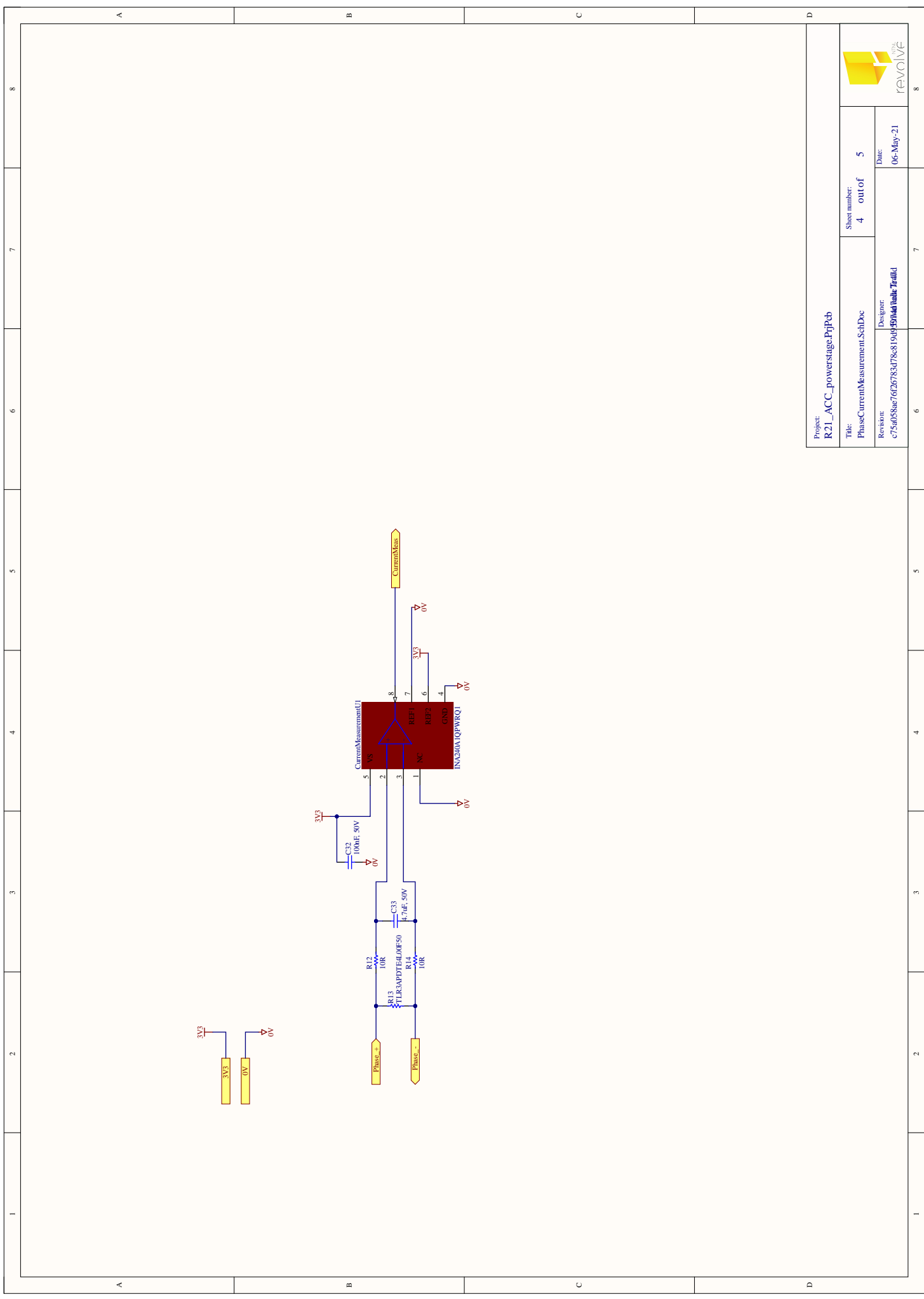
Sheet number:
5 out of 5

Date:
18-May-21

Revision:
c75d058ae76126783d78e819d4f30a4e4c7aaid

Designer:
FouadKhalilJasaid





Project:
R21_ACC_powerstage.PjtPcb

Title:
PhaseCurrentMeasurement.SchDoc

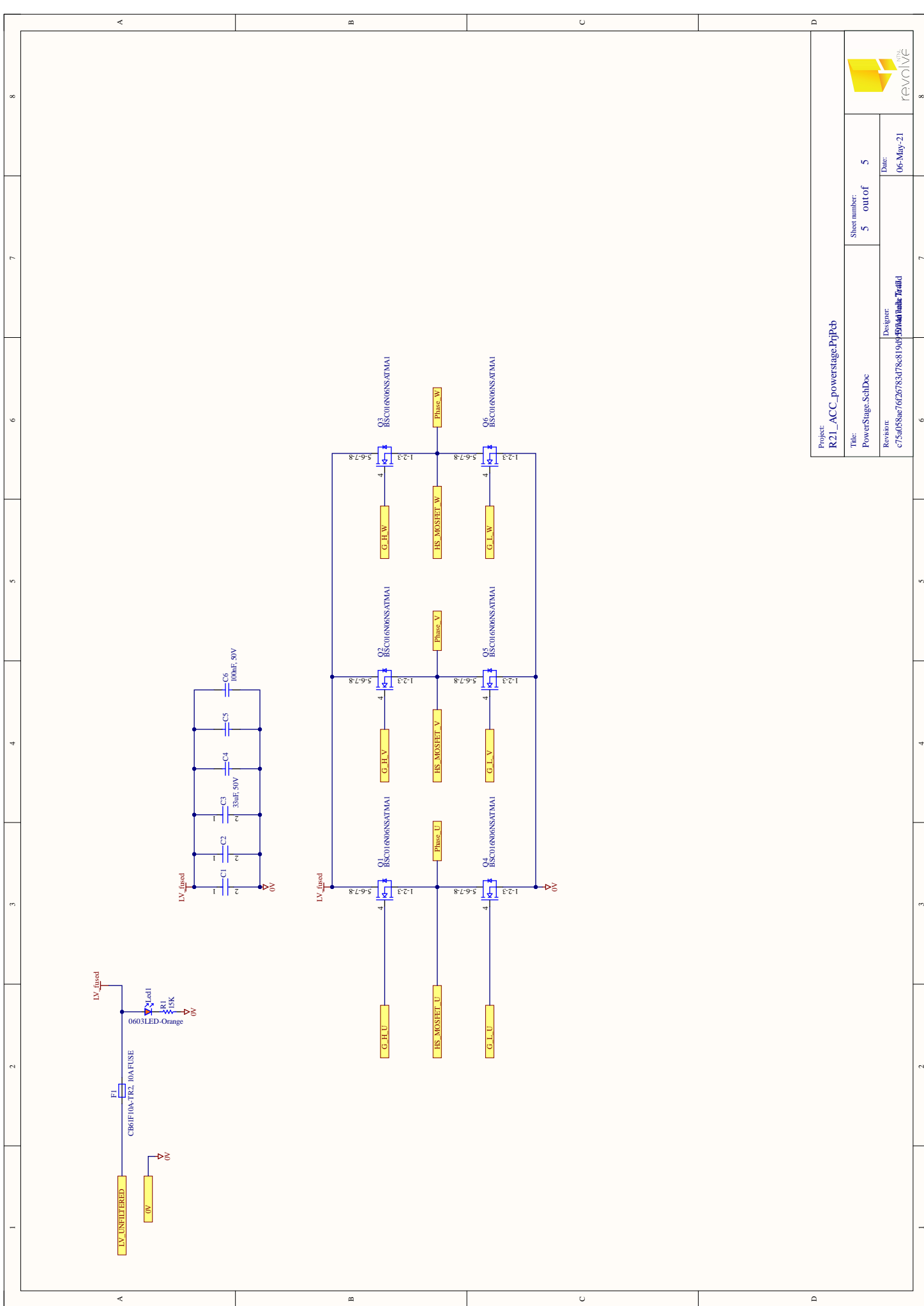
Revision:
e75a058ae76f26783c78c819c9199ad1ba6c7c4fd

Sheet number:
4 out of 5

Date:
06-May-21

Designer:
Dimitar Ivanchev





Project:
R21_ACC_powerstage.PjtPcb

Title:
PowerStage.SchDoc

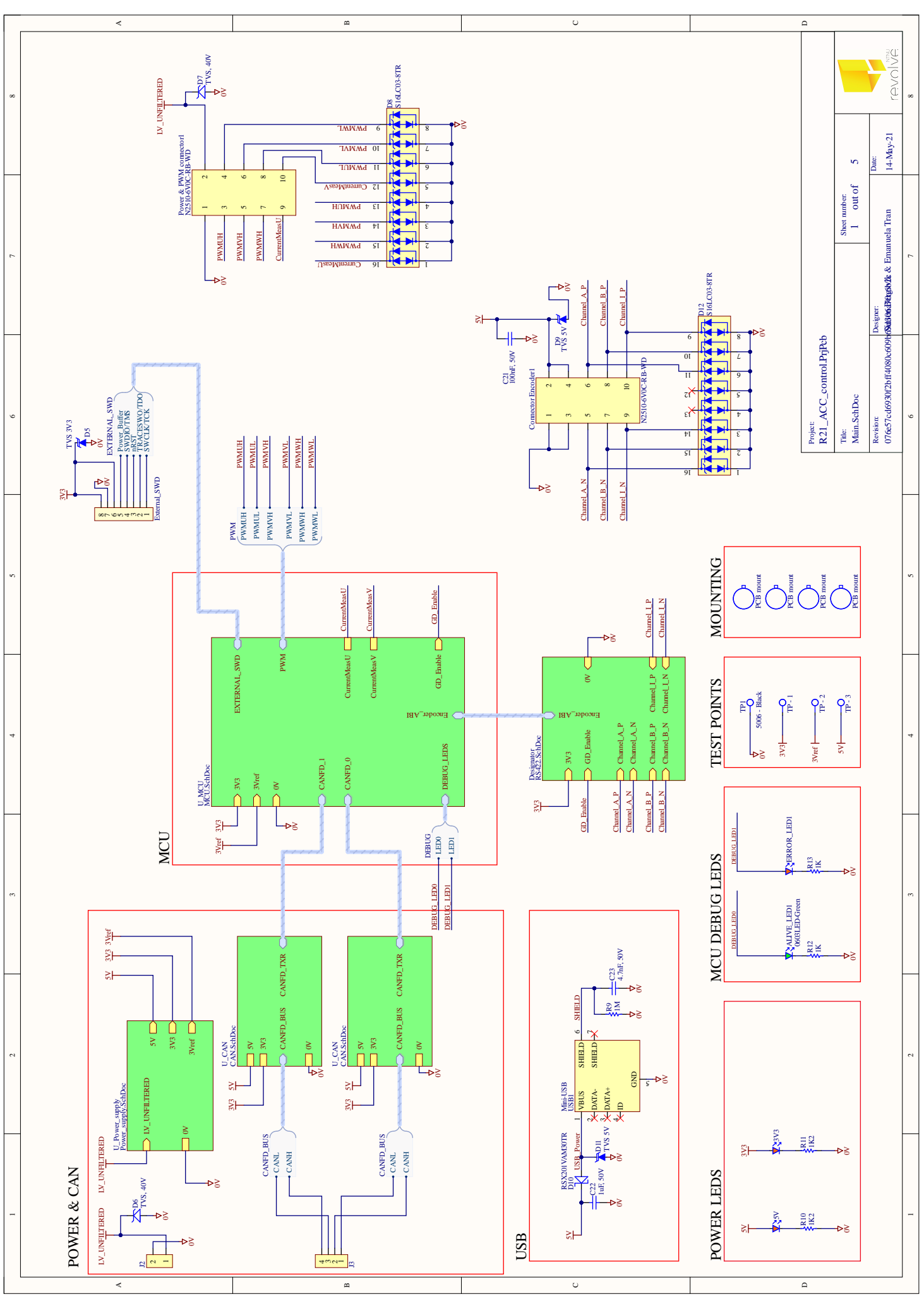
Sheet number:
5 out of 5

Date:
06-May-21

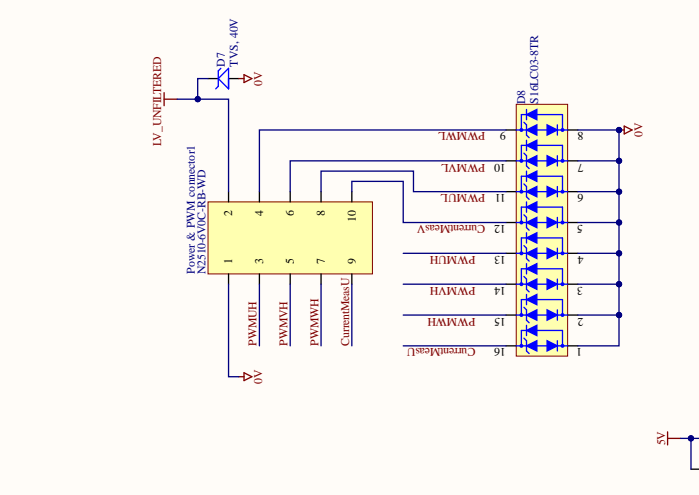
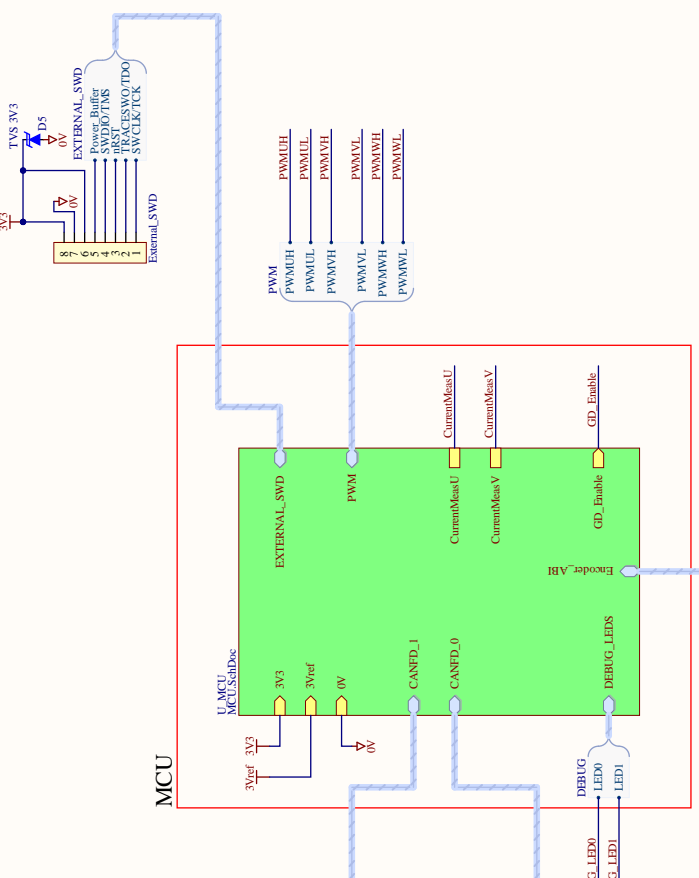
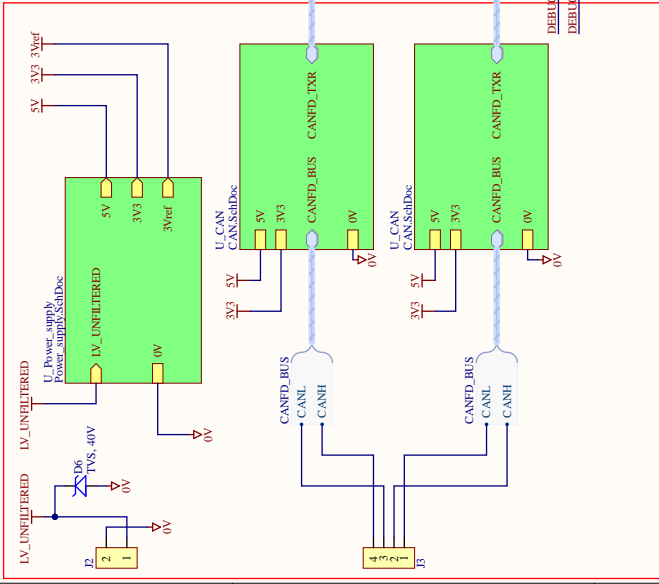
Revision:
e75a058ae76f26783d78c819d459fdd1ba6c7c4fd

Designer:
Svenja Heide

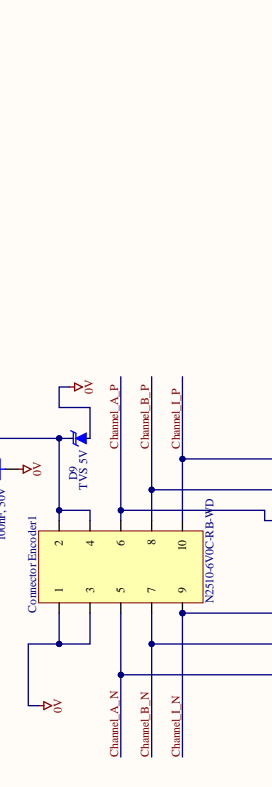
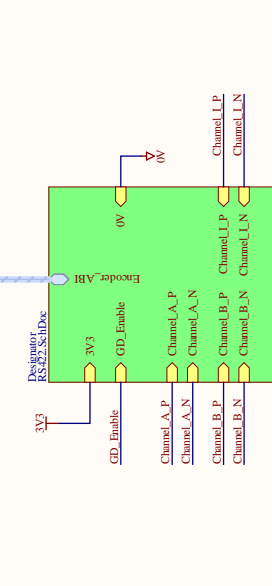
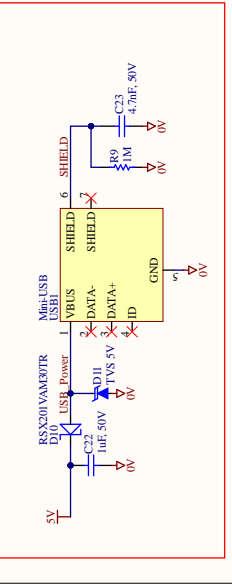




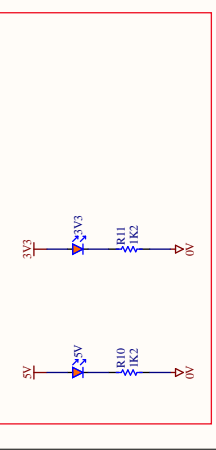
POWER & CAN



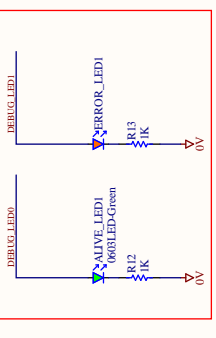
USB



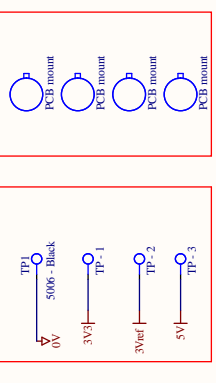
POWER LEADS



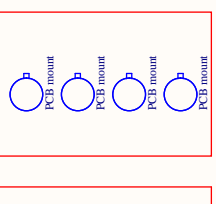
MCU DEBUG LEADS



TEST POINTS



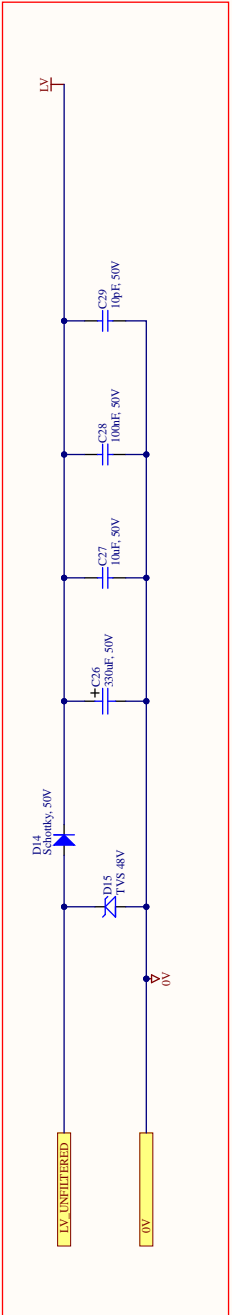
MOUNTING



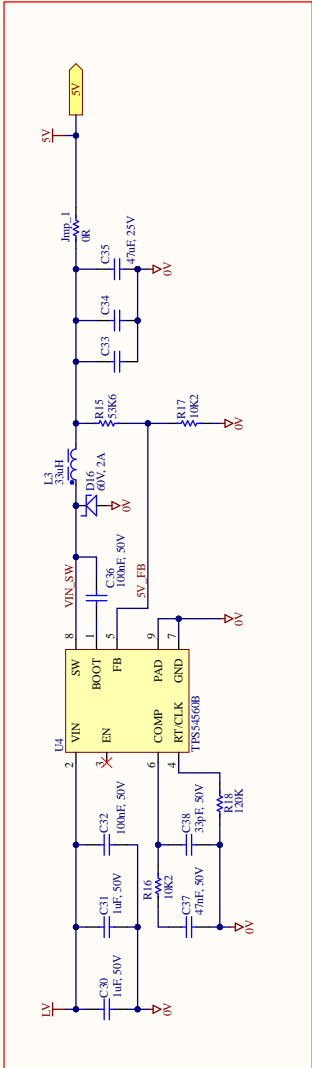
Project: R21_ACC_control.PjtPcb
 Title: Main.SchDoc
 Revision: 07f657c69302bf4f408bc09408016406
 Designer: Sebastian Paganik & Emanuela Tran
 Sheet number: 1 out of 5
 Date: 14-May-21



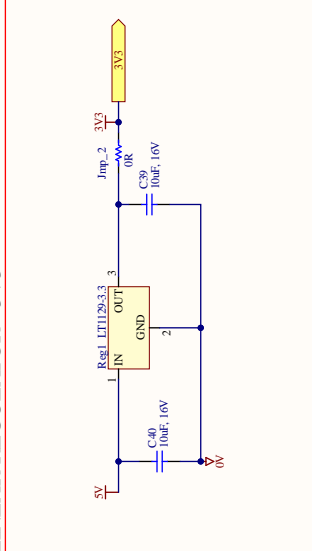
FILTER - LV



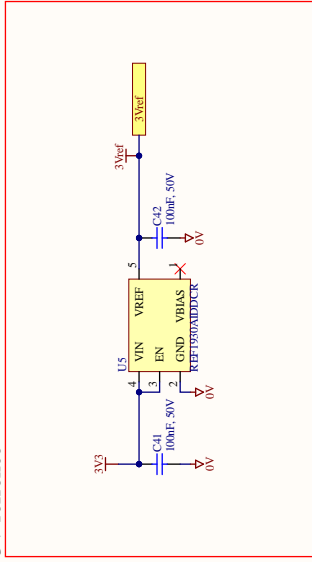
BUCK REGULATOR - 5V



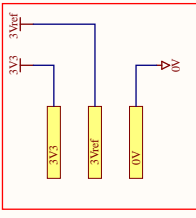
LINEAR REGULATOR - 3V3



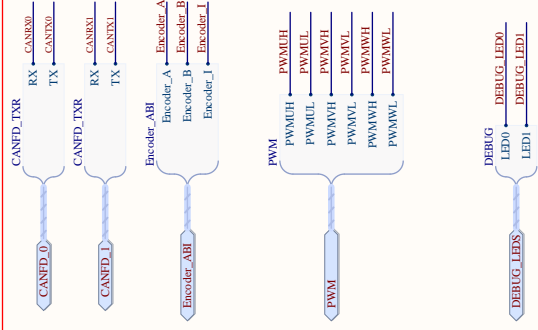
3V reference



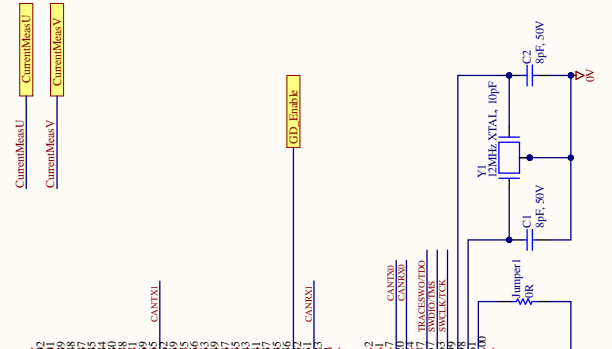
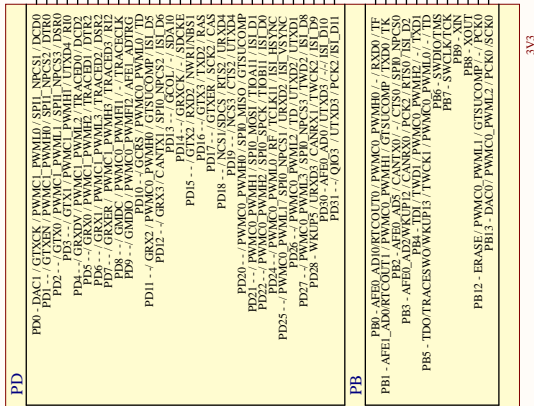
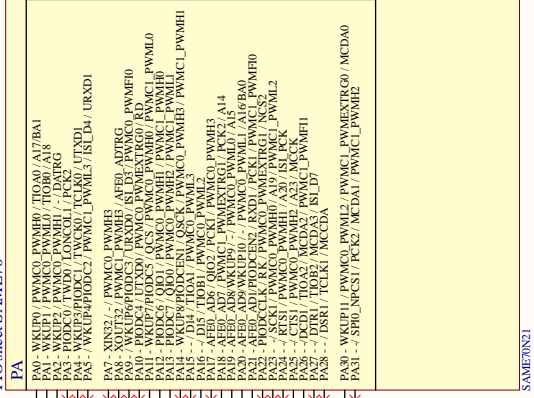
POWER IN



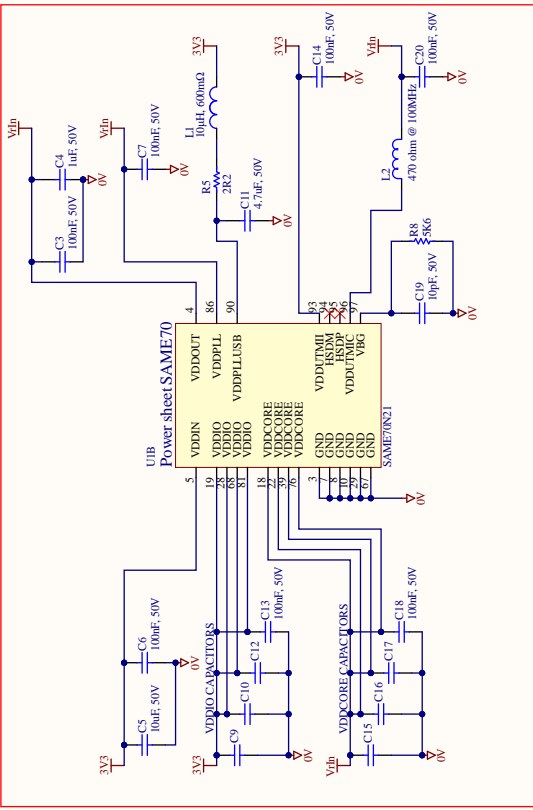
ATSAME70N21



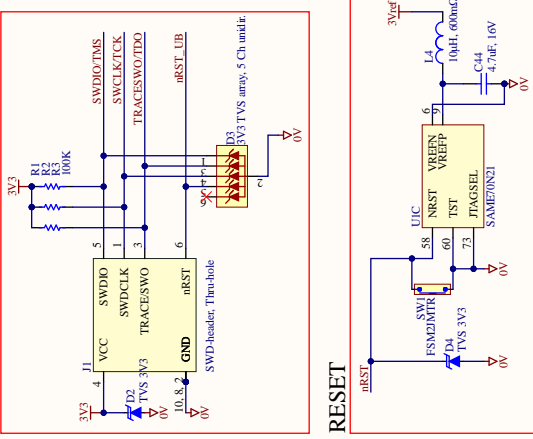
**UIA
PIO sheet SAME70**



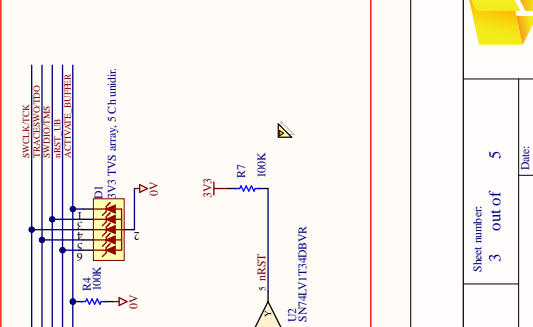
MCU POWER SUPPLY



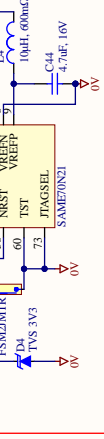
PROGRAMMING



PROGRAMMING_BUFFER



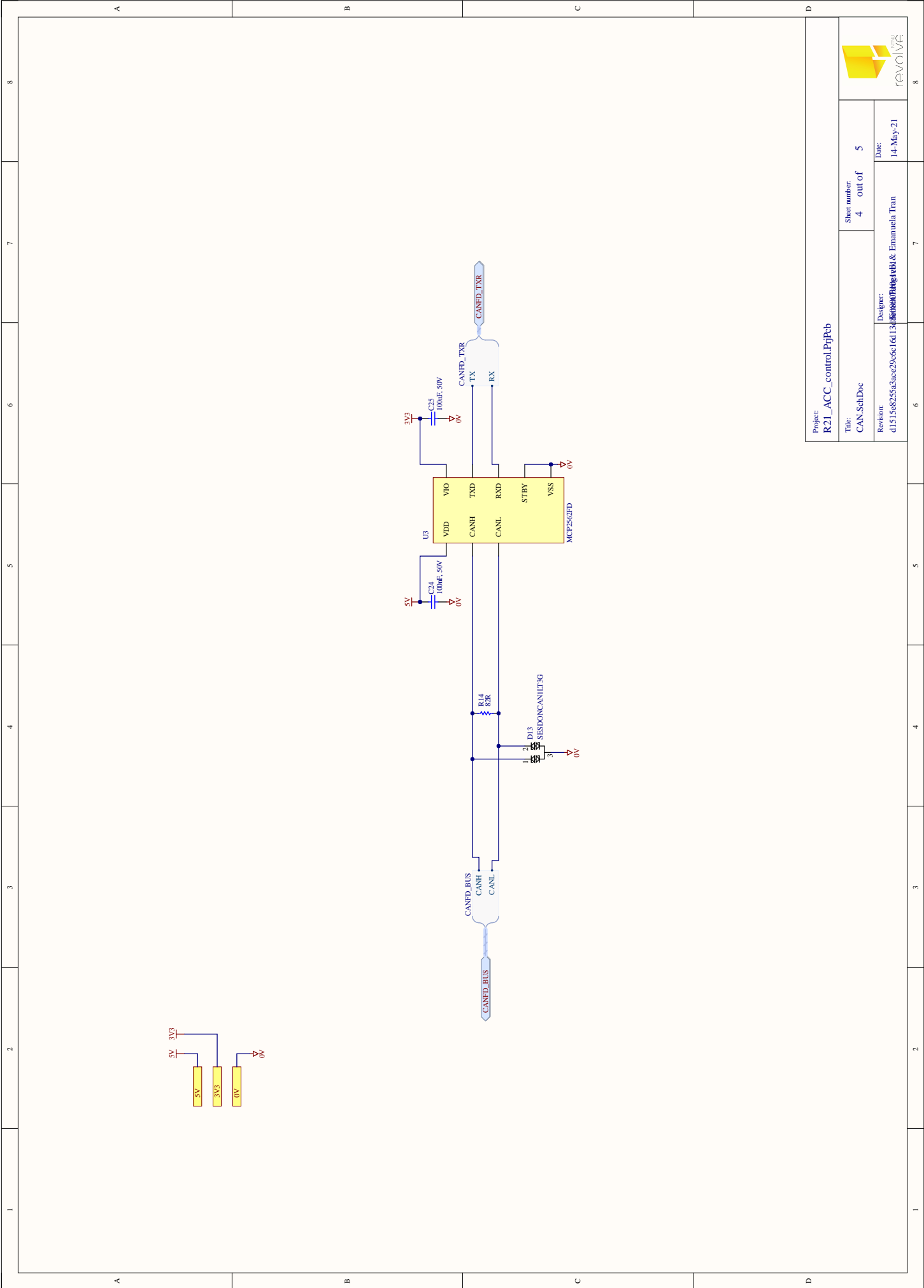
RESET



R21_ACC_control.PjPcb

Project:	R21_ACC_control.PjPcb
Title:	MCU_SchDoc
Revision:	9f49629677f003222b241dd16a6a184f698c8_Emanuela Tran
Sheet number:	5 out of 5
Date:	14-May-21





Project:
R21_ACC_control.PjtPeb

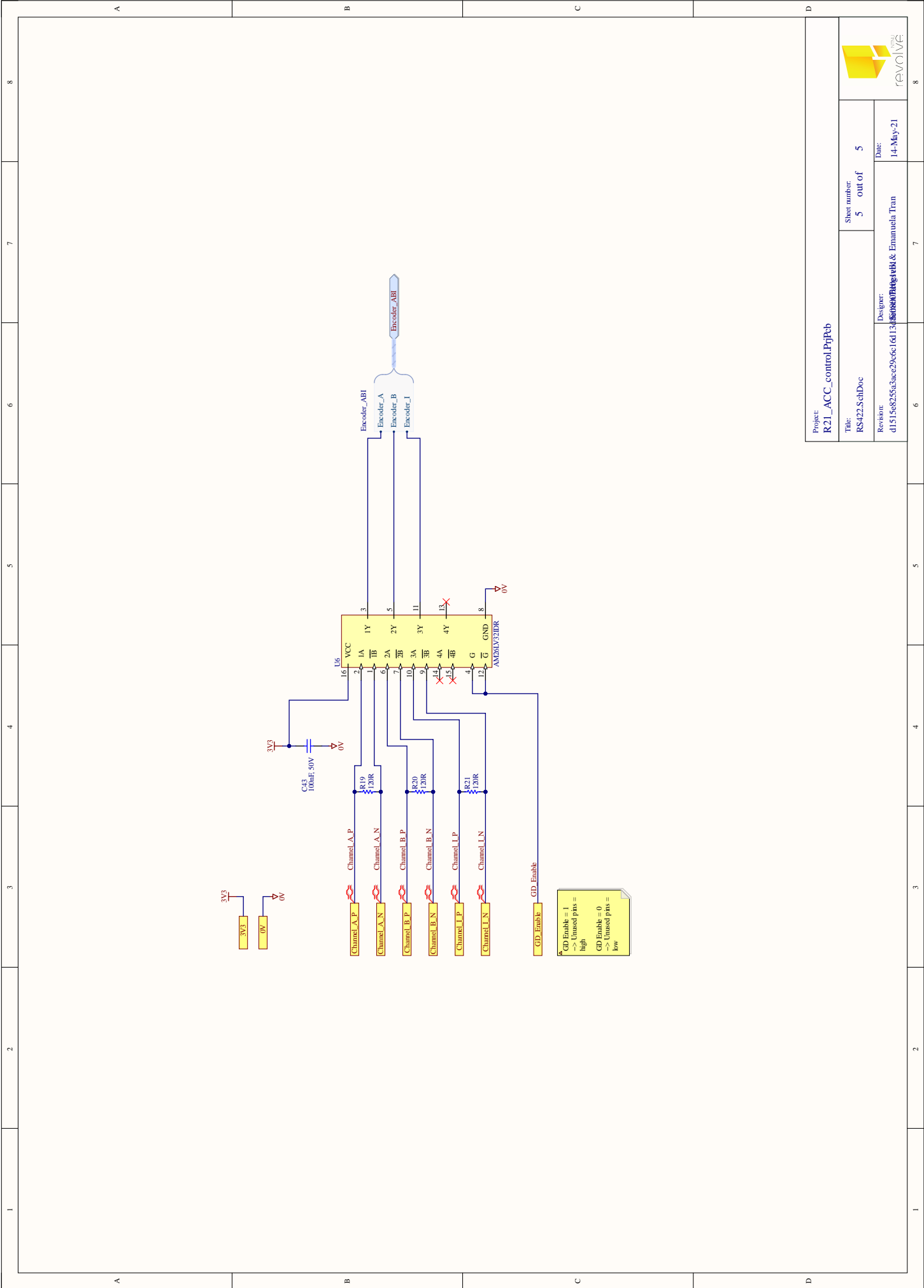
Title:
CAN_SchDoc

Revision:
d151568255a3ace2966c16d1338580784e2b4& Emanuel Tran

Sheet number:
4 out of 5

Date:
14-May-21





Project:
R21_ACC_control.PjtPeb

Title:
RS422_SchDoc

Sheet number:
5 out of 5

Revision:
d151568255a3ace2966c16d133880078e4e4& Emanuel Tran

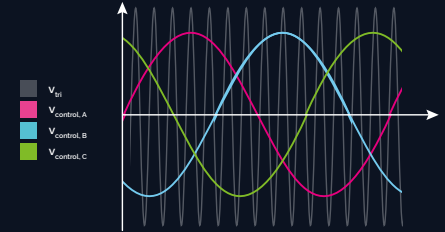
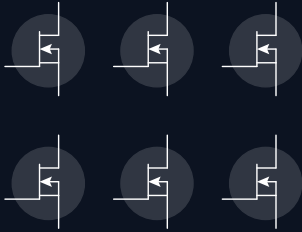
Designer:
Emanuel Tran

Date:
14-May-21



How can a car steer itself?

In order to design an autonomous steering system, a three-phase inverter and motor controller must be developed.



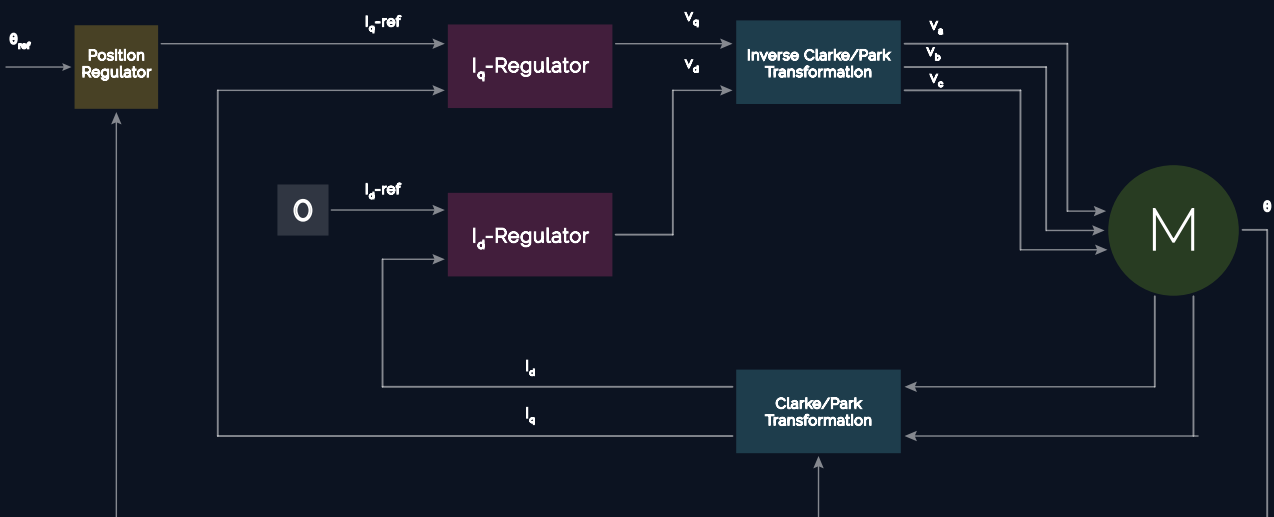
Three-Phase Inverter

A Three-Phase Inverter consists of transistors that work as switches. By the use of six MOSFETs switching every 20th microsecond, a sinusoidal ac-waveform can be produced from a dc-input voltage.

In order to protect vital components from harmful transients, the inverter also features bootstrap and snubber circuits to ensure reliable switching.

Sinusoidal Pulse-Width-Modulation

Pulse-Width-Modulation is a technique used to shape and control the frequency and pulse-width of a control signal. To obtain balanced three-phase output voltages in a three-phase PWM-inverter, triangular voltage waveforms are compared to three sinusoidal control voltages that are 120° out of phase. This results in three pairs of complimentary PWM-signals that controls the inverter.



Field Oriented Control

Field Oriented Control is a vector control algorithm to accurately control both torque and flux generated in brushless DC motors. By using Park and Clarke transforms, the three stator currents are decomposed into two orthogonal dq-currents which can be controlled by PI controllers. The output of the control algorithm produce three sinusoidal voltage references, which are then modulated as PWM-signals before being applied to the inverter.

