

Sensur av hovedoppgaver

Universitetet i Sørøst-Norge

Fakultet for teknologi og maritime fag



Prosjektnummer: **2019-04**

For studieåret: **2018/2019**

Emnekode: **SFHO3201-1 18H Bacheloroppgave**

Prosjektnavn

Guided Wolf

Utført i samarbeid med: Kongsberg Defence & Aerospace

Ekstern veileder: Øivind Grønli

Sammendrag: KDA har startet et langvarig prosjekt som skal vare over flere bacheloroppgaver og sommerprosjekter fremover. En vanlig ATV skal utvikles til å bli autonom. Denne bacheloroppgaven starter opp dette prosjektet og det skal implementeres løsninger for å kontrollere ATV-en uten fører. Det skal designes festeanordninger til alle nødvendige komponenter samtidig som det skal tas høyde for at det er nok plass igjen på ATV-en til å montere en våpenstasjon samt en plattform for drone.

Stikkord:

- Instrumentering av ATV
- Førerløs
- Bygging og integrasjon av løsning

Tilgjengelig: JA

Prosjekt deltagere og karakter:

Navn	Karakter
Martin Mjøhus Helle	
Torger Tobias Tokvam	
Kristian Vollmo Leirvik	
Sigurd Aftret Mørtvedt	
Linn Tandberg Thorud	
Ola Loe	

Dato: 14. juni 2019

Filippo Sanfilippo
Intern Veileder

Karoline Moholth
Intern Sensor

Øivind Grønli
Ekstern Sensor



Guided Wolf

KRISTIAN VOLLMO LEIRVIK
SIGURD AFTRET MØRTVEDT
MARTIN MJØLHUS HELLE
TORGER TOBIAS TOKVAM
LINN TANDBERG THORUD
OLA LOE

23. mai 2019



KONGSBERG

USN Universitetet
i Sørøst-Norge

Abstrakt

KDA har startet et langvarig prosjekt som skal vare over flere bacheloroppgaver og sommerprosjekter fremover. En vanlig ATV skal utvikles til å bli autonom med tilhørende våpenstasjon.

Denne bacheloroppgaven starter opp dette prosjektet og skal implementere løsninger for å kontrollere ATV-en uten fører. Videre skal dette fjernstyres gjennom en ekstern enhet. Det skal designes festeanordninger til aktuatorer samtidig som det skal tas høyde for at det er nok plass igjen på ATV-en til å montere en våpenstasjon samt en plattform for drone.



CAN-AM OUTLANDER XT 650 T

Gruppemedlemmer



Martin Mjølhus Helle
Prosjektleder
Maskin



Ola Loe
Dokumentansvarlig
Elektro



Linn Tandberg Thorud
Kravansvarlig
Elektro



Sigurd Aftret Mørtvedt
Testansvarlig
Elektro



Torger Tobias Tokvam
CAD ansvarlig
Maskin



Kristian Vollmo Leirvik
Risikoansvarlig
Elektro

Innhold

1	Abstrakt	I
2	Gruppemedlemmer	II
3	Forkortelser	1
4	Oppgaven	2
4.1	Oppdragsgiver	3
4.2	Oppgavebeskrivelse	3
4.3	Målsetting for prosjektet	5
4.4	Begrensninger og rammeverk	5
5	Prosjektplan	7
5.1	Gruppekontrakt	8
5.1.1	Arbeidsmiljø og ordninger	8
5.1.2	Arbeidstider	8
5.2	Arbeidsmodell	9
5.2.1	SCRUM	9
5.2.2	Proessen	10
5.3	Oppdatert arbeidsmodell	12
5.3.1	Kanban	12
5.3.2	Gantt diagram	15
5.4	Kommunikasjon og arbeidsverktøy	17
5.5	Dokumentasjon	18
5.6	Profilering	18
5.6.1	Logo	18
5.6.2	Nettside	18
5.6.3	Expo	19
6	Krav	20
6.1	Generelt	21
6.2	Use cases og stories	21
6.3	1. iterasjon til krav	25
6.4	Kvalitetssikring	27

6.5	Krav til delsystem	28
6.5.1	Overordnet	28
6.5.2	Gass	29
6.5.3	Brems	29
6.5.4	Gir	29
6.5.5	Styring	30
6.5.6	Nødstop	30
6.5.7	Våpenstasjon	31
6.5.8	Batteri	31
6.5.9	Velt	31
6.5.10	Fjernstyring	32
6.6	UML: Overordnet	32
6.7	2. Iterasjon til krav	34
6.7.1	Nye krav	37
6.7.2	Oppdatert liste av krav	38
7	Risiko	42
7.1	Generelt om risiko	43
7.2	Systemrisiko	44
7.3	Prosjektrisiko	48
8	Målinger	50
8.1	Maskinmålinger	51
8.1.1	Kraftmålinger på delsystemene	51
8.1.2	Ramme	53
8.1.3	Lokalisering av festepunkter	54
8.2	Elektromålinger	55
8.2.1	Målinger av gassensorene	56
8.2.2	Målinger av instrumenter på styret	59
9	Konsept	62
9.1	Delsystemkonsepter	63
9.1.1	Styringsenhet	63
9.1.2	Håndbrems	66
9.1.3	Fotbrems	67
9.1.4	Gir	68

9.1.5	Gass	69
9.1.6	Styring	69
9.1.7	Nødstop	70
9.1.8	Batteri	71
9.1.9	Våpenstasjonsfeste	72
9.2	Pugh-matriser for delsystemene	73
9.2.1	Håndbrems	76
9.2.2	Fotbrems	77
9.2.3	Gir	78
9.2.4	Gass	79
9.2.5	Styring	80
9.2.6	Nødstop	81
9.2.7	Batteri	81
9.2.8	Våpenstasjonsfeste	82
9.3	Løsninger til delsystemene	83
9.3.1	Gass	83
9.3.2	Håndbrems	83
9.3.3	Fotbrems	83
9.3.4	Gir	83
9.3.5	Styre	84
9.3.6	Nødstop	84
9.3.7	Batteri	84
9.3.8	Ramme	84
9.3.9	Fjernstyring	84
9.4	Overordnede systemet	84
9.5	Kontrollsystem	87
10	Research	89
10.1	PID kontroller	90
10.1.1	Iterasjon 1	90
10.1.2	Iterasjon 2	92
10.1.3	P-kontroller	93
10.1.4	I-kontroller	93
10.1.5	D-kontroller	93
10.1.6	Anti-Windup	94

10.1.7	Tuning metoder - PID-kontroller	94
10.2	Fjernstyrt Corvette	96
10.3	Forsterke ATV-ens servo	96
10.4	Robot Operating System (ROS)	96
10.5	Fartsmåling med photodiode	99
10.6	MODBUS	100
10.6.1	Modbus som kommunikasjonsverktøy	100
10.6.2	Lagring av data i modbus	101
10.7	Transistor	103
10.8	Bremse-arm	105
11	Utvikling-EL	108
11.1	Valg av aktuatorer	109
11.1.1	Brems	109
11.1.2	Gir	111
11.2	Bestilling av aktuatorer	112
11.2.1	Bremsesystem	113
11.2.2	Girsystem	113
11.2.3	Kontrollenhet for aktuator	114
11.2.4	Bestillingsoversikt - aktuatorer	116
11.3	Driver - TR-EM-288-H	116
11.4	Styring av ATV	119
11.4.1	Motor-kontroll	119
11.4.2	Stepper-motor	120
11.5	Valve Controll	121
11.5.1	Kontroll via I/O styreenhetsprofil (ventil)	121
11.5.2	Arduino som PLC	122
11.5.3	Konfigurering av motordriver	124
11.5.4	Konfigurering av motordriver - iterasjon 2	128
11.5.5	Kontroll via I/O styreenhetsprofil - iterasjon 2	130
11.5.6	Arduino som PLC - iterasjon 2	130
11.6	PWM til DC verdier	131
11.7	Software	132
11.7.1	Processing og Arduino	132
11.7.2	Oppsett	133

11.7.3	Kontrollerlayout (PS4)	133
11.7.4	UML: Delsystem	135
11.7.5	Interface	137
11.7.6	Analog/Digital - Pin i Processing	140
11.7.7	Oppstartsfunksjon	141
11.7.8	PID Kontroller	142
11.7.9	Hastighetsmåling	145
11.7.10	Gass	147
12	Mekanisk design	149
12.1	Design av ramme	150
12.1.1	Krav som må møtes	150
12.1.2	Hva finnes fra før	153
12.1.3	Design	154
12.1.4	Oppsett av analyser	160
12.1.5	Analyser	166
12.2	Design av håndbrems	171
12.2.1	Designparametere	173
12.2.2	Første utkast	176
12.2.3	Design - begrunnelser	179
12.2.4	Evaluering	181
12.3	Design av fotbrems	182
12.3.1	Designparametere - fotbrems	182
12.3.2	Første utkast	184
12.3.3	Evaluering	187
12.3.4	Andre utkast	188
12.4	Samhandling mellom bremsene	190
12.5	Design av girsystemet	191
12.5.1	Designparametere	192
12.5.2	Første utkast	194
12.5.3	Evaluering	195
12.5.4	Andre Utkast	196
12.5.5	Evaluering	197
12.5.6	Tredje Utkast	200
12.5.7	Evaluering	201

12.5.8	Tiltak for posisjonsregistrering	201
12.6	Design av styre	203
12.6.1	Kraftoverføring til styre	205
13	Test	219
13.1	Gass	222
13.1.1	PWM til DC spenning	222
13.2	Brems	228
13.2.1	Lineæraktuator	228
13.2.2	Hastighetsregulering med PWM	231
13.2.3	Håndbrems PID med potmeter	233
13.2.4	Bremsing med PID-kontroll	234
13.2.5	Strømsensor stopfunksjon	235
13.3	Gir	237
13.3.1	PID kode med potmeter	237
13.3.2	PID kontroller med LA14 aktuator	239
13.4	Styring	241
13.4.1	Stepper-motor(28BYJ-48)	241
13.4.2	Mapping av steppermotor	246
13.4.3	Test av stepper-motor ved strømbrudd	247
13.4.4	Gradvis styring	248
13.4.5	Styring av stepper med arduino	249
13.4.6	Test arduino med transistor	253
13.4.7	Test transistor som switch	255
13.4.8	Test av ulike transistorer	256
13.5	Hardware	258
13.5.1	Bluetooth	258
13.5.2	RC - sender/mottaker	260
13.6	Software	262
13.6.1	Grensesnitt - PS4 kontroller med processing	262
13.6.2	Trådløst med Raspberry PI	263
13.6.3	Skjermdeling	263
13.6.4	IR-sensor for hastighet	265
13.6.5	Video og objektgjenkjenning	265

14 Implementering	267
14.1 ATV-en den 21.05.2019	268
14.2 Kretskjema	269
14.3 Ramme, festepunkter og aktuatorplassering	271
15 Modell	272
15.1 Software	273
15.1.1 Første Presentasjon	273
15.1.2 Andre Presentasjon	276
15.2 Mekanikk	278
15.2.1 3D printing	278
15.2.2 Laserkutting	279
16 Bestillingsoversikt	282
17 Oppsummering	284
18 Videre anbefalinger	284
18.1 Mekanisk design	284
18.1.1 Håndbrems	284
18.1.2 Gir	284
18.1.3 Styre	285
18.1.4 Eksponerte områder på ATV-en etter demontering	285
18.2 Elektrisk design	285
18.2.1 Strømsensor	285
18.2.2 IR-sensorer - hastighetsmåling	286
18.2.3 Video	286
18.2.4 Modbus/styring	286
18.2.5 Nødstop - funksjon	286
18.2.6 Nødstop - direkte knapp	286
18.2.7 Batteri	287
18.2.8 Trådløs overføring	287
19 Referanser	288
A Vedlegg	294
A.1 Arduino: Standard Firmata	295

A.2 Processing: Game Control Plus, Configurator	295
A.3 Processing kode	295
A.4 Google Drive: Vedlagt på minnepenn	295
A.5 Demonteringsoversikt	298
A.6 Teknisk Håndbok	344
A.7 Kraftmålinger på ATV	375
A.8 Kretsskjema for ATV	396
A.9 FEM analyse, ramme og innfestning	414
A.10 FEM analyse, overgang steppermotor-tannhjul	451
A.11 Styret stagløsning	462

Tabeller

1	Oversikt over forkortelser som går igjen i dokumentet	1
2	Arbeidstid før påske	8
3	Arbeidstid etter påske	8
4	Overordnede krav	28
5	Krav gass	29
6	Krav brems	29
7	Krav gir	29
8	Krav styring	30
9	Krav nødstop	30
10	Krav våpenstasjon	31
11	Krav batteri	31
12	Krav velt	31
13	Krav fjernstyring	32
14	Krav som ikke er blitt møtt eller tilføyd	35
15	Risikorangering (1-5)	44
16	Eksempel på risiko som kan inntreffe	44
17	Forklaring på tilkoblingspunktene	57
18	Oversikt over tilkoblingspunkt på styret.	61
19	Pugh-matrise for håndbrems	76
20	Pugh-matrise fotbrems	77
21	Pugh-matrise gir	78
22	Pugh-matrise gir	78
23	Pugh-matrise gass	79
24	Pugh-matrise styring	80
25	Pugh-matrise nødstop	81
26	Pugh-matrise våpenstasjonsfeste	82
27	Zeigler-Nichols tabell	95
28	Data lagring i standard modbus	101
29	Bestillingsoversikt fra Linak - aktuatorer	116
30	Beskrivelse av valgt motor-kontroll	120
31	Beskrivelse av valgt motortype	121
32	Tabell for test	220
33	Test gass - PWM til DC spenning	222

34	Test av DAC Breakout Development Board Module	224
35	Test gass - PWM til DC spenning	225
36	Gassfunksjon på ATV	226
37	Gassfunksjon på ATV	227
38	Test lineæraktuator	228
39	Test hastighetsregulering med PWM	231
40	Test PID kontroller med LA12 aktuator, brems	233
41	Test PID kontroller med LA33 aktuator, brems	234
42	Strømsensor begrensning	235
43	Test PID kontroller med LA12 aktuator, gir	237
44	PID kontroller med LA14 aktuator	239
45	Test stepper-motor: styring	241
46	Tabell for sekvenslogikk for en full step motor	242
47	Steppermotor Festo mapping	246
48	Test styring - EMMS-ST	247
49	Test homing - EMMS-ST	248
50	Test gradvis styring - EMMS-ST	248
51	Styring av stepper-motor med Arduino	249
52	Test av NPN transistor - 2N3904 med arduino	253
53	Test av NPN transistor - 2N3904	255
54	Test av ulike transistor	256
55	Test av trådløs styring med bluetooth	258
56	Test av trådløs styring med bluetooth, iterasjon 2	259
57	Test av trådløs styring med RC-kommunikasjon, HC-12	260
58	Test av HC-12, kommunikasjon begge veier	261
59	Test grensesnitt med PS4 kontroller og processing	262
60	Test av trådløs styring med Raspberry PI	263
61	Test skjermdeling - Discord	263
62	Test av IR-Sensor - hastighet	265
63	Test direkteoverført video i Processing	265
64	Oversikt over koblingspunkt til prototype	274
65	Bestillingsoversikt	283

Figurer

1	SCRUM-modellen	9
2	Trello-board	10
3	Arbeidstavle	13
4	Arbeidstavle for Kanban utarbeidet i Trello.	14
5	Gantt-diagram	16
6	Use case U1	22
7	Use case U2	23
8	Use case U3	23
9	Use case U4	24
10	Use case U5	24
11	Krav til systemet	26
12	UML: Overordnet av systemets funksjoner	33
13	Oppdatert kravliste inkludert test-ID	41
14	Illustrasjon av risikovurdering	43
15	Risikoanalyse av funksjonskravene til systemet	47
16	Risikoanalyse over prosjektet	49
17	Statisk beregning av håndbrems	52
18	Festepunkter for fothvilere	53
19	Modell av fører-fothviler	54
20	Festepunkt for fører-fothviler	54
21	Sekundærfestepunkter	55
22	Oversikt over gassensorer	56
23	Oversikt over oppsettet for målingene	57
24	Måleresultatet fra måling med Ee board	58
25	Måleresultat fra måling med Ee board 2	58
26	Kretsskjema - koblingspunkt 1	59
27	Kretsskjema - koblingspunkt 2	59
28	Kretsskjema - koblingspunkt 3	60
29	Kjøretøyet styres av en PC	63
30	Kjøretøyet styres av en håndkontroll for spillkonsoll	64
31	Kombinasjon av konsept 1 og 2	64
32	Konsept 4 RC-kontroller	65
33	Konsept 1 for håndbremsen	66

34	Konsept 2 for håndbremsen	66
35	Konsept 3 for håndbremsen	67
36	Konsept 2 for fotbrems	67
37	Konsept 1 for gir	68
38	Konsept 2 for gir	68
39	Konsept 3 for gir	68
40	Konsept 1 for gass	69
41	Konsept 2 for gass	69
42	Konsept 1 for styret	69
43	Konsept 2 for styret	70
44	Konsept 1 for nødstop	70
45	Konsept 2 for nødstop	71
46	Konsept 1 for batteri	71
47	Konsept 2 for batteri	71
48	Konsept 1 for våpenstasjonsfeste	72
49	Skisse av konsept 2 for våpenstasjonsfeste	72
50	Tilkoblinger til gir	83
51	ATV foran fra siden	85
52	ATV foran fra siden	85
53	ATV bak fra siden.	86
54	ATV bak fra siden.	86
55	LA12 aktuator CAD-modell	87
56	UML flowchart av systemets funksjoner	88
57	Elektrisk krets for motorarmaturen	90
58	Mekanisk modell	91
59	Krets med PID-kontroller og aktuator	93
60	ROS Filsystem	96
61	Illustrasjonsbilde av fartsmåling med photodiode	99
62	Kommunikasjon mellom motor og arduino via modbusprotokoll	101
63	Kretsskjema for NPN felles base-emitter	103
64	Cut-off for transistor	104
65	Saturation for transistor	105
66	Den eksisterende bremsehantelen som er på ATV-en	106
67	Bilde av ATV-ens eksisterende fotbremssystem	107
68	Tekniske spesifikasjoner for LA28	110

69	Tekniske spesifikasjoner for LA30	110
70	Tekniske spesifikasjoner for LA33	111
71	Tekniske spesifikasjoner for LA36	111
72	Tekniske spesifikasjoner for LA12	112
73	Tekniske spesifikasjoner for LA14	112
74	Aktuator LA33	113
75	Aktuator LA14	114
76	TR-EM-236	114
77	TR-EM-288	115
78	Pinne nr og hva de gjør	117
79	Tilkoblinger brukt i systemet.	118
80	Komponentene til CMMO-motor kontroller	119
81	Stepper-motor EMMS-ST	121
82	Arkitektur på releoppsett	122
83	Koblingskjema for releer	123
84	Konfigurering av kontroller og stepper motor	124
85	Ventilkonfigurasjon for steppermotor	125
86	Homingkonfigurasjon for steppermotor	126
87	Joggingkonfigurasjon for steppermotor	126
88	Joggingkonfigurasjon av grenser	127
89	Manuell styring av steppermotor	127
90	Konfigurering av motor - gir	128
91	Konfigurering av homing - Axis zero point	129
92	Konfigurering av styrepinner - vinkeldefinisjon	129
93	PWM - DC prototype kretskort	131
94	Konfigurasjoner på skulderknappene.	134
95	Knappekonfigurasjoner på front	134
96	UML - Gass	135
97	UML - Brems	135
98	UML - Gir	136
99	UML - Styring	136
100	1.Utkast til brukergrensesnitt	137
101	Gass grensesnitt i Processing	138
102	2.Utkast til brukergrensesnitt	139
103	Kaller oppstartsfunksjonen i draw	141

104	Oppstartsfunksjonen	142
105	Målepunkter på framhjul	146
106	Krav i forbindelse med våpenstasjonen	150
107	Horisontal bjelke med en last holdes oppe av en vertikal bjelke [53]	153
108	Veltebur, BMW E36	154
109	Oversikt over monteringsplan på ATV-en	154
110	fothvilerovergang for ramme	155
111	Festebrakett for de øvre festepunktene på ATV-en	156
112	Monteringsplate for våpenstasjon	156
113	Enkelt førsteutkast av rammen	157
114	Videreutvikling av den første rørrammen	157
115	Fothvilerovergang	158
116	Monteringsplate for våpenstasjonen	159
117	Rammen designet i firkantrør	160
118	Ramme, områder låst for bevegelse	161
119	Kontakt eksempel, overflate på overgang	163
120	Kontakt eksempel, rør	163
121	Kontakt eksempel, fothvilerovergang	163
122	Kontakt eksempel, base	163
123	Kraft, ytre rør	164
124	Kraft nedover	164
125	Kraft framover	164
126	Mesh	165
127	Eksempel på bjelker med forfinet mesh	166
128	Fothvileroverganger	167
129	Forbedrede spenningskonsentrasjoner	167
130	spenningskonsentrasjoner i platen, i grønne områder.	168
131	Spenningskonsentrasjoner i monteringsplaten	168
132	Spenninger i rammen	169
133	Spenninger i monteringsplaten	170
134	Ferdigstilt ramme med designede komponenter, aktuatorer og steppermotor, sett forfra	170
135	Ferdigstilt ramme med designede komponenter, aktuatorer og steppermotor, sett bakfra	170
136	CAD-modell av brems	171

137	CAD-modell av brems ovenfra	171
138	Bilde av frakoblet bremsehantel	172
139	Hengselet mellom brems og ATV	173
140	CAD-Modell av Håndbrems	174
141	Graf over pitch	174
142	Matematisk modell - Håndbrems	175
143	Modell - håndbrems komponent	176
144	Systemet sett ovenfra	177
145	Interaksjonene mellom kulehullet og bremsestemplet	177
146	Håndbrems: Isometrisk synsvinkel	178
147	Håndbrems: sett ovenfra	178
148	Håndbrems: sett fra siden	178
149	Sectionview av stempelhullet	178
150	2D tegninger med dimensjoner av komponenten	179
151	FEM analyse av håndbrems	181
152	Interface mellom aktuatoren og delen	182
153	Bremsestempel	183
154	Matematisk modell over buelengde	183
155	Fotbrems: Isometrisk synsvinkel	184
156	Fotbrems: Isometrisk synsvinkel	184
157	Fotbrems: Sett fra siden	184
158	Bilde - fotbrems original	185
159	Samhandling mellom aktuatoren og delen	185
160	Modell - Fotbrems hengsel	186
161	Slaglengde fotbrems	186
162	Modell - fotbrems geometri	187
163	CAD modell - Fotbrems	188
164	Fotbrems dimensjoner	188
165	CAD designteknikk	189
166	Matematisk modell - brems	190
167	Hengselled til LA14	191
168	Hengselled til LA14	192
169	Beskrivelse av LA14	193
170	Girarm: Isometrisk synsvinkel	194
171	Girarm: Isometrisk synsvinkel før sammensveising	194

172	Girarm: Sett fra siden	194
173	Viser maksspenninger på 11.92 MPa.	195
174	Gir: Isometrisk synsvinkel	196
175	Gir: Sett ovenfra	196
176	Gir: Sett fra siden	196
177	Girdeksel feste	197
178	Cad modell - Gir	198
179	Girspak - stagposisjon	199
180	Gir: Isometrisk synsvinkel	200
181	Gir: Sett fra siden	200
182	Bøyepening på LA14	201
183	Tilsagd deksel	202
184	Tilsagd deksel 2	203
185	Torque til angular velocity graf	204
186	Modell: Styre - skisse	205
187	Modell: Styre - skisse	206
188	Tannhjul 1:1 forhold	207
189	Ulike konsepter for Tannhjulløsning	208
190	Koniske tannhjul	209
191	Kjededrift fra motor	210
192	Design av stagløsning	211
193	Modell: 2 stk rådrev	212
194	Stramming av kjede	213
195	Girtupp	214
196	Koblingsstykke	215
197	Overgang fra utvekslingsgir til tannhjul	216
198	Fem av Overgang fra utvekslingsgir til tannhjul	217
199	Giraksling med tannhjul	217
200	Designet av hele styresystemet for seg selv.	218
201	Designet av hele styresystemet sammen med ATV.	218
202	Krets til T1-GA.01	222
203	Resultatet av simuleringen av T1-GA.01	223
204	Måling med oscilloskop	223
205	PWM til DC revidert	225
206	Bilde fra første implementeringstest	226

207	Skisse til PCB sammen med resultat	227
208	Koblingsskjema - driver og motor	228
209	Koblingsskjema for simulink modellen	229
210	Signalene til styringspinnene på Arduinoen.	230
211	Aktuator med PWM	231
212	Hastighetsregulering med PWM	232
213	Hastighetsregulering med PWM 2	233
214	LA33 Kraft mot Ampere	235
215	Oppkobling til LA12 med TR.EM-288-H og potmeter	237
216	Coil-logikk for stepper-motor	242
217	Koblingsskjema for arduinoen	243
218	Stepper-motor: hel rotasjon	245
219	Tabell med stepfordeling	249
220	Step i grader	249
221	Kretstegning - styring av stepper med arduino	250
222	Transistorkrets som bryter	254
223	Krets med 2N3904 - transistor	255
224	HC-12 sender/mottaker	260
225	Gass til venstre, brems til høyre	262
226	Videotest i Processing	266
227	Test med fargegjenkjenning i Processing	266
228	Oversiktbilde av ATV-en, ovenfra	268
229	Oversiktbilde av ATV-en, høyre side	269
230	Kretsskjema til systemet	270
231	Oversiktbilde av ramme, festepunkter og aktuatorplassering	271
232	Modell for prototype	273
233	Krets til modellbil	274
234	Fremre innfestning og teststykker	278
235	Ferdigkuttete plater	281
236	Kretsskjema - koblingspunkt 2	287

Forkortelser

Tabell 1 viser en oversikt over ulike forkortelser av fulle navn som går igjen i dokumentet.

Tabell 1: Oversikt over forkortelser som går igjen i dokumentet

Forkortelse	Fullt navn
MMH	Martin Mjøhus Helle
OL	Ola Loe
TTT	Torger Tobias Tokvam
SAM	Sigurd Aftret Mørtvedt
LTT	Linn Tandberg Thorud
KVL	Kristian Vollmo Leirvik
ØG	Øivind Grønli (ekstern sensor og veileder)
FS	Filippo Sanfilippo (intern veileder)
KDA	Kongsberg Defence & Aerospace, Protech Systems
PWM	Pulse Width Modulation
CAD	Computer Aided Design
ATV	All-Terrain Vehicle
FCT	Festo Configuration Tool
DC	Direct current
AC	Alternating current
IR	Infrared
PID	Proporsjonal Integrasjon Derivasjon
RC	Radio communication
LED	Light Emitting Diode

Oppgaven

Dette kapitlet omhandler beskrivelse av oppgaven som er blitt tildelt. Det vil bli gjennomgått diverse aspekter rundt oppgaven; Hva som er forventet ift prosess og sluttprodukt.

Innhold

- Oppdragsgiver
- Oppgavebeskrivelse
- Målsetting
- Begrensninger
- Rammeverk



KONGBERG



USN Universitetet
i Sørøst-Norge

Oppgaven omfatter endringer av en vanlig ATV slik at den kan fjernstyres fra en datamaskin. Det skal i tillegg designes et feste slik at en våpenstasjon fra KDA skal kunne monteres på ATV-en.

4.1 Oppdragsgiver

Kongsberg Defence & Aerospace (KDA) er en av de tre avdelingene som til sammen utgjør Kongsberg Gruppen. Kongsberg Gruppen er en videreføring av firmaet Kongsberg Våpenfabrikk som ble startet opp i 1814, som siden da har vært en viktig bedrift for forsvarsindustrien i Norge. [39]

KDA utvikler og produserer missiler, luftvernsmissiler, kommunikasjonssystemer, fjernstyrte våpenstasjoner og mer. [38]

4.2 Oppgavebeskrivelse

KDA har tidligere kjørt et sommerprosjekt hvor studenter har fått i oppgave å lage et fjernstyrt kjøretøy med en paintball våpenstasjon. Denne hadde som mål å identifisere skyteskiver automatisk.

Bachelorprosjektet vil være en videreføring av dette prosjektet, hvor alt skal skaleres opp. Hovedprosjektet vil i første omgang være å fjernstyre kjøretøyet, men også å se på andre løsninger som kan være nyttige for prosjektet fremover.

KDA ser derfor på mulighetene for å bygge et autonomt kjøretøy hvor utgangspunktet er en standard ATV med bensinmotor. Oppgaven blir å instrumentere kjøretøyet med nødvendige aktuatorer og sensorer for å kontrollere styring, gass, gir og brems med en trådløs forbindelse. Samtidig må det tas høyde for at det skal være plass til å montere en plattform for våpenstasjon og drone.

KDA ga følgende punkter som skal løses,

- Design av systemløsning
- Valg av aktuatorer og sensorer
- Valg av datamaskin om bord i kjøretøyet

- Design av sikkerhetsløsning (nødstop)
- Design av batterisystem for kjøretøy med våpenstasjon
- Dokumentasjon og bestilling av deler
- Bygging og integrasjon av løsning
- Demonstrasjon av kjøring med instrumentert ATV

Dette skal oppnås uten at ATV-en modifiseres på en sånn måte at det ikke er mulig å sette den tilbake til original stand. Det skal altså ikke kuttes, bores eller sveises i kjøretøyet.

4.3 Målsetting for prosjektet

Gruppen har som hovedmål å kunne utvikle et komplett system for fjernstyring av ATV-en som møter oppgavebeskrivelsen og krav definert av prosjektgruppen og oppdragsgiver.

Gruppen vil også jobbe mot en ferdigstillelse og implementering av det utviklede systemet.

Gruppen vil hvis tiden strekker til, jobbe for å møte de nedprioriterte kravene som ikke er kritiske for å oppnå hovedmålene.

4.4 Begrensninger og rammeverk

I begynnelsen av prosjektet er det viktig å klargjøre hva som er potensialet og begrensningene i prosjektet. Det er fire essensielle hovedpunkter;

- Tid
 - Modeller, samtaler i gruppen og med veileder har gitt en indikasjon på hva som er realistisk å få til på tidsrommet frem til juni 2019. Det er satt opp innledende modeller for å få en grov oversikt, med forbehold om endringer. Planlagte tidsintervaller har fått en fleksibel tilnærming da det er mye som kan forandre seg i løpet av prosjektet. Det kan være oppgaver som er mer utfordrende enn antatt eller deler som blir forsinket.
- Budsjett/ressurser
 - Det har blitt tatt opp med intern og ekstern sensor på det grunnlag av at skolen og KDA har ressurser tilgjengelig til prosjektet. Dette kan være alt fra kontorer, tilgjengelighet på ATV-en, budsjett til innkjøp av deler, programvare og tid til møter.
- ATV begrensning
 - En av de største begrensningene for rask framgang for prosjektet, omhandler tilgang til ATV.

ATV-en står innenfor et strengt regulert område, hvor kun personell med

sikkerhetsklarering har tilgang. Det er heller ikke tilrettelagt for at området kan besøkes på alle tider av døgnet; Helligdager og besøk etter kontortid (07.30-16.00) er ekskludert fra besøkstider.

- Krav og ønsker fra oppdragsgiver
 - Hva KDA ønsker er helt avgjørende for suksessen til oppgaven. Systemet som skal designes skal bli en autonom ATV med våpenstasjon. Beslutninger må tas i betrakning i forhold til hvordan det skal videreutvikles i senere tid.

Prosjektplan

Dette kapitlet inneholder en generell beskrivelse av prosjektplanen. Det tar for seg ordninger ift arbeidsmiljø, tider, arbeidsmodell, prosess, kommunikasjon og arbeidsverktøy.

Innhold:

- Gruppekontrakt
- Arbeidstider
- Arbeidsmodell
- Oppdatert arbeidsmodell
- Kommunikasjonsverktøy
- Dokumentasjon
- Profilering



5.1 Gruppekontrakt

5.1.1 Arbeidsmiljø og ordninger

For å sørge for at alle medlemmene i gruppen har et reglement å følge har det blitt opprettet en gruppekontrakt. Denne inneholder arbeidstider og hva slags oppførsel som forventes innad i gruppen. Dersom noen bryter noen av reglene i gruppekontrakten er det fastsatt at vedkommende skal kjøpe en pose kaffe til gruppen. Det blir tatt forbehold om annen straff ved gjentatte kontraktbrudd dersom det er behov for det.

5.1.2 Arbeidstider

Gruppens definerte arbeidstider er vist i tabell 2 og 3. Disse tidene er fastsatt inntil annet blir bestemt. Gruppen vil i all hovedsak holde til på rom i115 i dette tidsrommet, med unntak av møter og andre formaliteter. Utover fastsatte tider, står hver enkelt gruppemedlem fritt til å jobbe når de vil. I henhold til gruppekontrakten skal hvert gruppemedlem jobbe minimum 8 timer utenom fastsatte tider etter påske. Det vil bli holdt morgenmøter hver dag på 15 minutter hvor gruppen går gjennom hver enkelts arbeidsoppgaver og planer for dagen.

Arbeidstid før påske, 14.01 - 17.04:

Tabell 2: Arbeidstid før påske

Dag	Arbeidstid	Oppgaver
Mandag	09:00 - 15:00	Avsatt arbeidstid
Tirsdag		Annen undervisning
Onsdag	09:00 - 15:00	Avsatt arbeidstid
Torsdag	09:00 - 15:00	Avsatt arbeidstid
Fredag	09:00 - 15.00	Avsatt arbeidstid

Arbeidstid etter påske 23.04 - 31.05:

Tabell 3: Arbeidstid etter påske

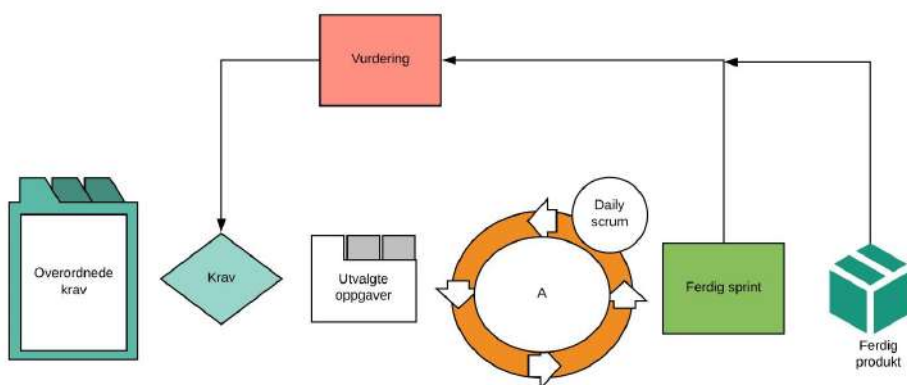
Dag	Arbeidstid	Oppgaver
Mandag	09:00 - 17:00	Avsatt arbeidstid
Tirsdag	09:00 - 15:00	Planlegging og avsatt arbeidstid
Onsdag	09:00 - 15:00	Avsatt arbeidstid
Torsdag	09:00 - 15:00	Avsatt arbeidstid
Fredag	09:00 - 15.00	Avsatt arbeidstid

5.2 Arbeidsmodell

5.2.1 SCRUM

Scrum fra rugby: *"An ordered formation of players, used to restart play, in which the forwards of a team form up with arms interlocked and heads down, and push forward against a similar group from the opposing side. The ball is thrown into the scrum and the players try to gain possession of it by kicking it backwards towards their own side."*
 - wikipedia [75]

På samme måte har prosjektgrupper begynt å jobbe verden over, sammen om å dytte videre mot et felles mål. Dette har blitt en populær metode å jobbe på, en iterativ prosess hvor man får en viktig rolle som et individ som skal bære laget sitt fremover. Videre blir SCRUM-modellen forklart nærmere fra de offisielle sidene til SCRUM [74].

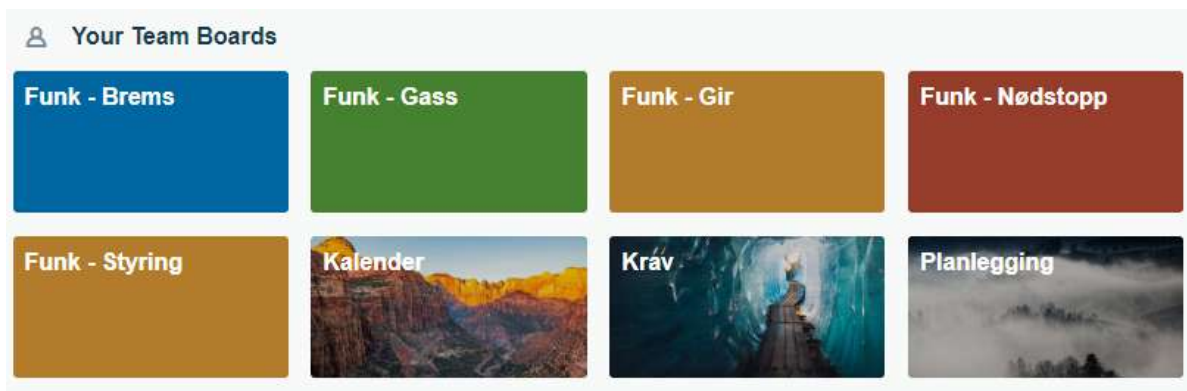


Figur 1: SCRUM-modell som viser forløpet til hver enkelt sprint

Hva er SCRUM?

SCRUM er veldig enkel å bryte opp i mindre oppgaver som kan ferdigstilles uavhengig av hverandre. Dette gjør det svært praktisk å bruke en modell som har dette i bunnen. Egentlig bygger SCRUM-modellen på user-cases hvor man skal finne små krav til enkeltbrukere, men allerede i hovedfunksjonene ble systemet delt opp godt. SCRUM gjør at alt av oppgaver blir lett å spore, og det er også lett å revurdere oppgavene over hva som har høy prioritet de neste ukene. Trello skal brukes til å organisere disse oppgavene. I

figur 2 kan man se hva de ulike tavlene i Trello heter, her deles hvert delområde i mindre deler, hvor man kan flytte oppgavene mellom kategoriene gjøremål->pågår->testing og ferdig. Fordelen med å bruke Trello til dette utover vanlige post-it lapper er at det blir mer gjennomskiktig og man har muligheten til å notere på "lappene" etter hvert som nødvendig.



Figur 2: Trello-board

5.2.2 Prosessen

Overordnede krav

Dette er hovedoppgavene som skal oppnås i prosjektet. Disse overordnede kravene må brytes ned til mindre prosesser. Noen av disse overordnede kravene kommer fra KDA, mens noen måtte utformes selv. Disse overordnede kravene har blitt godkjent av Øvind Grønli (ØG) før prosessen kunne fortsette.

Krav

Kravene må så brytes ned for å finne ut hvordan man kan gjøre disse i mindre intervaller. Disse kravene skal få en tidsramme, hvor det skal anslås hvor lang tid hver enkelt oppgave tar. Disse oppgavene skal få en tidsramme på 1/3/5 timer eller 1/3/5 dager. Dette anslaget gjøres før videre løp. Da kan man se hvor mange av disse kravene som er mulig å gjøre iløpet av en 2 ukers periode, som kalles en sprint.

Sprint og Daily SCRUM

En sprint er en arbeidsperiode hvor det arbeides intensivt med oppgavene. En sprint består av daglige sprinter som varer i 24 timer, altså en daily SCRUM. Hver dag blir startet av et 15 minutters møte hvor man går igjennom oppgavene som skal gjøres. Her skal hver enkelt få oppgaver fordelt fra kravene som skal gjøres i løpet av sprinten. I

løpet av møtet skal det bli klart hva alle skal gjøre den dagen, hvilke utfordringer som har kommet opp og hva man trenger hjelp til for videre arbeid.

Etter endt sprint skal man ta en vurdering på hvordan det har gått. Da skal man se på oppgavene som har blitt gjort og hver enkelt forklarer hva man har gjort og hvilke problemer man har hatt. Dette brukes så videre til å gjøre en ny vurdering på hvilke oppgaver som skal prioriteres til neste sprintperiode. Denne prosessen skal gjennomgås etter hver endt sprint.

5.3 Oppdatert arbeidsmodell

5.3.1 Kanban

Ved en nærmere gjennomgang av prosjektmodellen, har det blitt enighet om å skifte prosjektmodell fra Scrum til en hybrid av Scrum og Kanban (07.03.2019); En modell med utgangspunkt i Kanban, men med noen elementer fra Scrum-modellen.

Dette har blitt gjort på bakgrunn av hvordan systemet er oppbygd. Det er flere delsystemer som må utvikles parallelt med hverandre for å kunne komme i mål med prosjektet.

Det vil fortsatt bli satt tidsfrister for de mest kritiske deloppgavene, men modellen kan ansees som mer dynamisk for prosjektets del, da en oppgave, ikke nødvendigvis må ferdigstilles innenfor en gitt tidsperiode. Man kan av den grunn, hoppe frem og tilbake på de forskjellige oppgavene.

Kanban modellen er mer inkluderende i form av at man må kunne se systemet i helhet og hvordan samholdet mellom alle delsystemene vil fungere. Alle vil få mer innsikt i hvert delsystem og derfor få et større eierskap til prosjektet. Kanban, i motsetning til Scrum, vil effektivisere prosessen ytterligere da alle kan, ved hvilket som helst tidspunkt, "hoppe" fra tildelt oppgave og bistå noen andre som trenger mer arbeidskraft.[31]

Ansvarområdene som er blitt tildelt fra scrum-prosessen (prosjektleder, dokumentansvarlig etc.) vil fortsette som før.

Her kan forandringer forekomme når som helst, i motsetning til scrum, hvor man ikke kan forandre på krav og konsept midt i en sprint. Det vil også bli utført oppgaver fortløpende, slik at man alltid kan vise til resultater under prosjektløpet.

Arbeidsmengden vil distribueres på en slik måte at ting skal kunne fullføres og at arbeidstavlen (Figur 3) ikke fylles opp og blir uoverkommelig. Oppgavene skal gjøres iht prioriteringer, slik at oppgavene med høyest prioritet kommer først i køen.

Backlog	Gjøre mål	Planlegging	Utvikling	Test	Implementering	Utført
Backlog	Oppgave	Oppgave UO UO				
Backlog	Oppgave	Oppgave	Oppgave UO UO			
Backlog						Oppgave

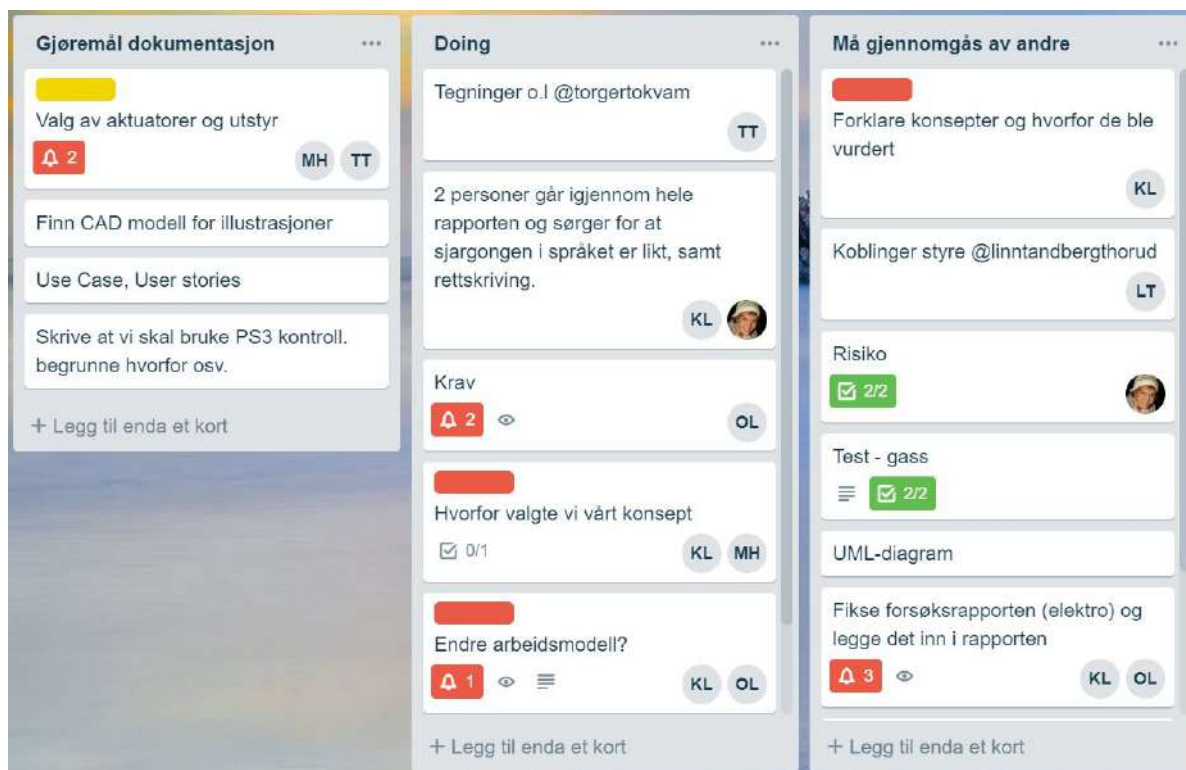
Figur 3: Prosessen for de forskjellige oppgavene og underordnede oppgaver(UO) som beskriver status.

Scrum modellen passer bedre der hvor delsystemene har en lav grad av tilknytning til hverandre og delsystemene fungerer godt isolert. I Kanban vil det være nødvendig at alle delsystemene samspiller godt med hverandre. Det vil derfor være mer effektivt dersom alle er involvert og ser helheten til systemet og hvordan delsystemene påvirker hverandre.

Fordeler med Kanban:

- Kortere iterative prosesser; raskere resultat, større motivasjon.
- Kan når som helst endre aspekter ved en oppgave.
- Kan bytte mellom oppgaver og evt sette oppgaver på "vent".
- Kan endre prioritet på gitte oppgaver.
- Tettere samarbeid mellom gruppe medlemmer.

Arbeidstavlen vil hjelpe prosjektgruppen med å holde orden på hvor langt i prosjektfasen hver enkelt deloppgave har kommet. Figur 2 viser til hvordan en tidsaktuell (08.03.19) tavle vil se ut:



Figur 4: Arbeidstavle for Kanban utarbeidet i Trello.

Her får man god oversikt over forløpet til oppgavene; hvor langt de har kommet i prosessen og eventuelt hvilken tilknytning de har til hvert enkelt delsystem.

Det har , på bakgrunn av oppbyggingen av dette prosjektet, blitt tatt en beslutning om å bytte prosjektmodell. Dette vil tjene bedre i det lange løp ift. omprioriteringer og at det gir rom for en mer dynamisk arbeidsmetode. Forandringer kan og vil forekomme uten at det vil påvirke prosjektet negativt.

5.3.2 Gantt-diagram

Gantt-diagram er et verktøy som kan brukes til planlegging og administrering av prosjekter. Denne type framstilling av oppgaver, ansvarsforhold og tidsbruk kan bidra til å skape et mer oversiktlig og forutsigbart bilde for hvordan forløpet til en prosess skal foregå. Dette skaper en bedre samhandling i gruppen.

Diagrammet illustrerer arbeidsforløpet og mengde til enhver tid. Dette har blitt satt opp i forhold til de planlagte sprintene som skal gjennomføres. Fullt gantt-diagram kan sees i Figur 5.

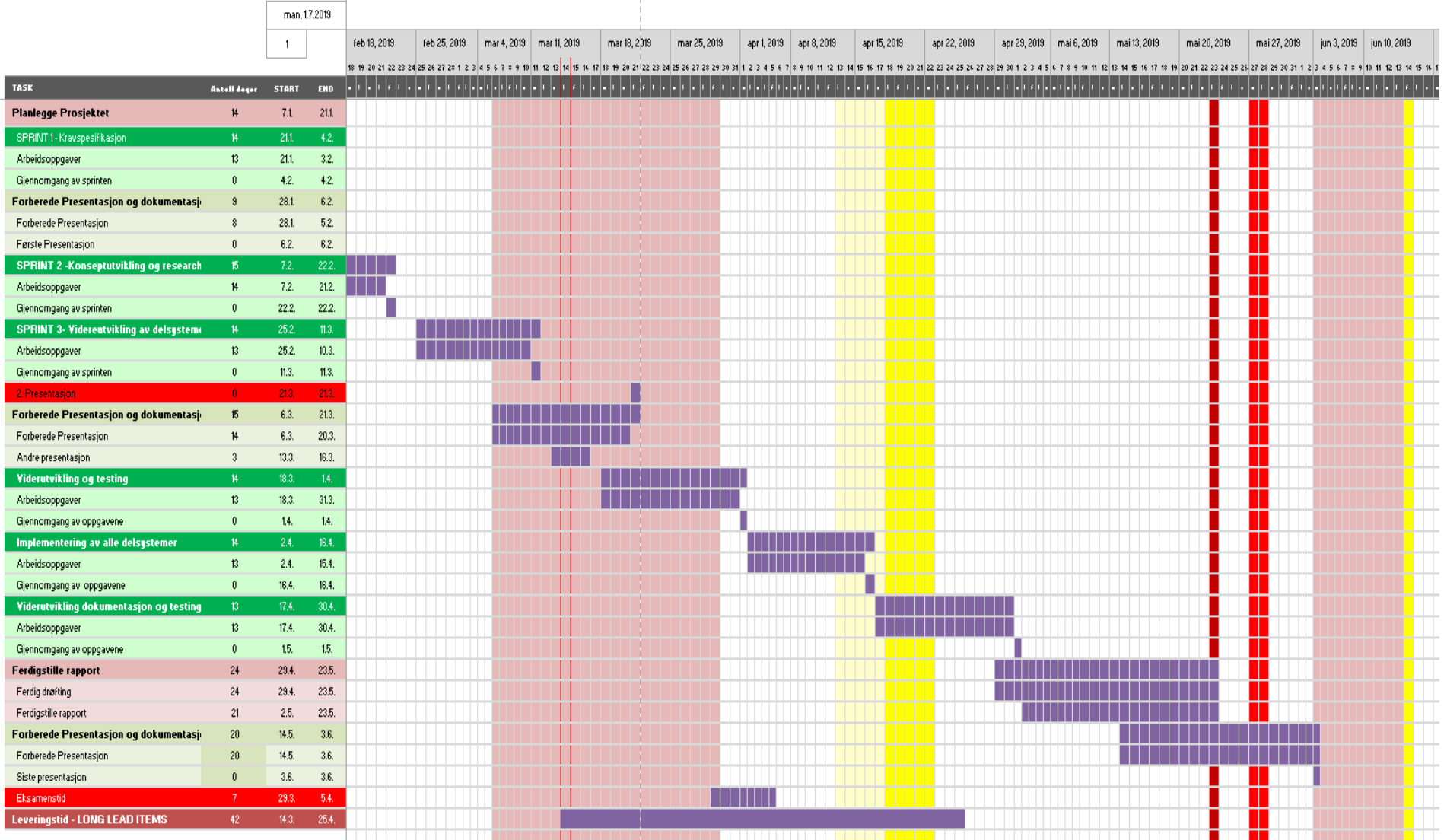
Fargekodene indikerer arbeidsperiodene; planleggingsfase, testfase, implementeringsfase o.l. Dette er en initialmodel, og er dermed ikke et fastsatt hendelsesforløp. Det gir et inntrykk, i grove trekk, om hvordan det er tenkt å dele opp prosjektet fremover.

Alle tidsfrister som er satt, hvor særlig long lead items er særs viktig, gir gruppen et estimat basert på forventet leveringstid på noen av produktene.

Det har også blitt lagt inn fargekoder på viktige datoer, som presentasjonene, helligdager og expo. Det har blitt printet ut en enkel versjon av gantt-diagrammet som er hengt opp på grupperommet, noe som gjør at man raskt kan se hvor langt unna neste deadlines eller andre viktige hendelser vil være.

Instrumentering av ATV

Bachelor - Gruppe 4



Figur 5: Gantt-diagrammet over hele prosjektperioden

5.4 Kommunikasjon og arbeidsverktøy

Slack

Det har blitt opprettet en egen kanal i Slack hvor noe av kommunikasjonen mellom gruppemedlemmene foregår. Her har det blitt delt opp i underkategorier med forskjellige temaer. Slack brukes hovedsakelig til å dele litteratur og prosjektverktøy. Det vil også bli sendt over oppdatert kodeversjoner til andre gruppemedlemmer.

Facebook messenger

De fleste i gruppen er ikke vant til å sjekke Slack regelmessig. Derfor blir messenger brukt til korte beskjeder som oppmøtetid eller påminnelser.

Snapchat

Det blir brukt Snapchat til mange av de samme funksjonene som Facebook messenger men blir hovedsakelig brukt til sosialt samhold.

Trello

Trello blir brukt til å organisere krav og oppgaver. Det har blitt opprettet eget "board" til alle krav som man legger alle delkravene under. Det har også blitt opprettet egne til planlegging, overordnede krav og kalender. Dette gjør at alt av organisering blir samlet på ett sted.

Overleaf

Gruppen har blitt enige om å bruke Latex til skriving og bruker Overleaf til dette. Dette gjør at det er enkelt å gjøre korrektur underveis.

Google Drive

Drive vil være veldig viktig for at all informasjon og dokumentasjon samles på ett sted underveis i prosjektet. Her lagres alt av møtereferater og kladd til dokumenter.

5.5 Dokumentasjon

Underveis i prosjektet skal alt arbeid av gruppen dokumenteres. Hovedsakelig skal dette gjennomføres gjennom følgende punkter:

- Dagsreferat for å oppsummere alt arbeid fra dag til dag
- Timelister som viser hvilket arbeid som er gjort og når det ble gjort
- Møtereferat med interne- og eksterne veiledere
- Dokumenter, diagrammer, modeller osv. som er relevante for prosjektarbeidet lastes opp på google drive.

5.6 Profiling

For å lage en helhetlig profiling av prosjektet, har gruppen blitt enige om design på nettside og logo. Navnet Guided Wolf ble laget etter en idémyndring hvor alle gruppe-medlemmene bidro. Etter dette ble det laget nettside og logo til prosjektet.

5.6.1 Logo

Logoen ble designet ut ifra to forskjellige bilder. Ett kompass hentet fra Wikimedia [83] og ulv med måne fra freepik [77]. Disse to bildene ble satt sammen i Adobe Photoshop sammen med teksten Guided Wolf med fonten coolvetica [82]. For å passe sammen med fargen på logoen ble farge #3C3C3C brukt.

5.6.2 Nettside

Under prosjektperioden vil nettsiden oppdateres med jevne mellomrom. Etter litt leting landet gruppen på en ferdig nettside fra Os-templates [45]. Disse er gratis å bruke og er enkle å skreddersy til eget bruk. Ved å velge en slik løsning blir arbeidet med nettsiden minimalt, samtidig som det blir et fint resultat.

Det har blitt lagt inn ett gantt diagram på nettsiden fra developers.google [24]. Dette ble lagt til som en test, for å se om det ga en rask oversikt over prosjektperioden online. Dette viste seg å være bedre i teorien enn i praksis, da den er vanskelig å oppdatere fortløpende.

5.6.3 Expo

Til Expo 28. mai har det blitt avtalt med ØG at det skal sjekkes opp med roll-ups og giveaways fra KDA. Gruppen lager selv en A0 plakat som skal henge på standen. Det vil forhåpentligvis bli laget en liten video av at ATV-en kjører, dersom det blir tid til dette.

Krav

I dette kapitlet vil det bli utdypt om hvordan "User cases og stories" har blitt brukt for å definere krav og spesifikasjoner til systemet.

Innhold:

- Generelt om krav
- Use case -og story
- 1. iterasjon til krav
- Kvalitetssikring
- UML: Overordnet
- 2. iterasjon til krav



KONGSBERG



USN Universitetet
i Sørøst-Norge

6.1 Generelt

Etter å ha fått presentert oppgaven fra KDA ble det først enighet om, ut ifra oppgavebeskrivelsen, hvilke overordnede funksjoner og krav som var viktige for systemet.

Etter rådføring med ØG, fikk gruppen et større perspektiv over hva som måtte på plass. Dette hjalp gruppen til å sette opp "use cases" og stories.

6.2 Use cases og stories

Dette er et verktøy brukt for å kunne se hvilke krav som skal settes til systemet. Dette tilbyr en ny innfallsvinkel sett fra brukeren sin side.

I begynnelsesfasen av prosjektet er det kritisk å få på plass definisjoner på hva systemet bør inneha i form av ytelser og brukervennlighet

Det er viktig å få definert om hva brukeren ser etter i systemet. Det er derfor behjelpelig med såkalte "use cases" som man kan basere og bygge systemet ut ifra.

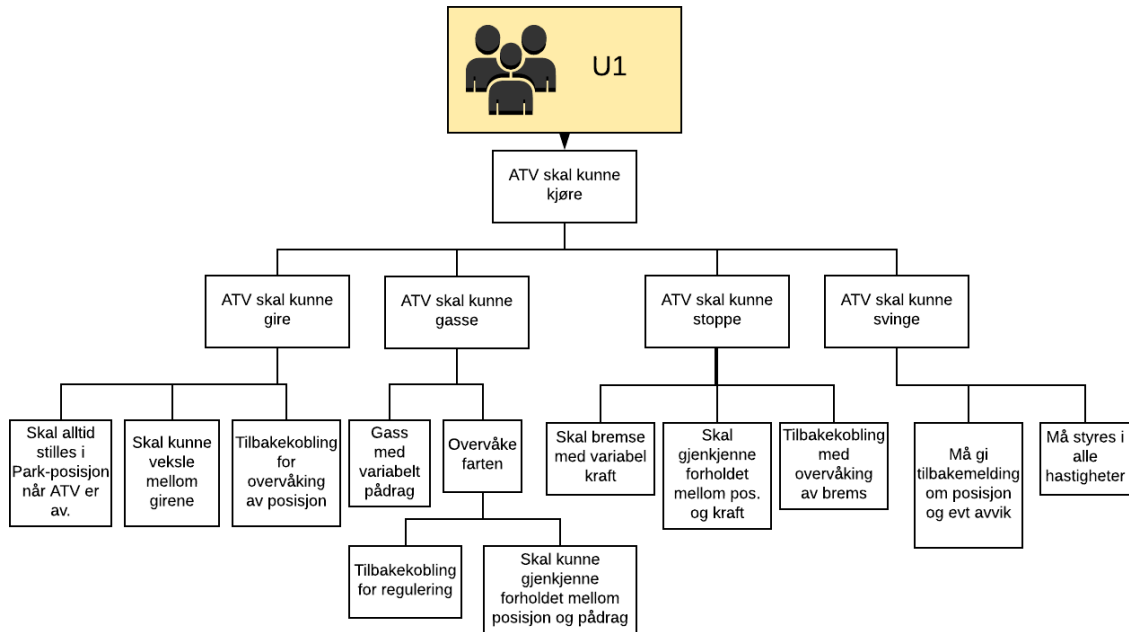
En "use case" brukes for å beskrive systemet fra flere perspektiver, og kan bygges opp med følgende setning: "Som bruker, vil jeg at.."

- (U1) **Som bruker, vil jeg** at ATV skal kjøre
 - **Som bruker, vil jeg** at ATV skal kunne gasse med variabelt pådrag
 - **Som bruker, vil jeg** at ATV skal kunne bremse
 - **Som bruker, vil jeg** at ATV skal kunne skifte mellom nærløys og fjernløys
 - **Som bruker, vil jeg** at ATV skal kunne skifte mellom gir.
 - **Som bruker, vil jeg** at ATV skal kunne signalisere.
- (U2) **Som bruker, vil jeg** at ATV skal være trygg.
- (U3) **Som bruker, vil jeg** at den skal være enkel å bruke (styre).
- (U4) **Som bruker, vil jeg** at ATV ikke skal skades.
- (U5) **Som bruker, vil jeg** at ATV-ens funksjoner skal kunne overvåkes.

- **Som bruker, vil jeg** at ATV skal gi tilbakemelding om gass, gir, brems og styring.

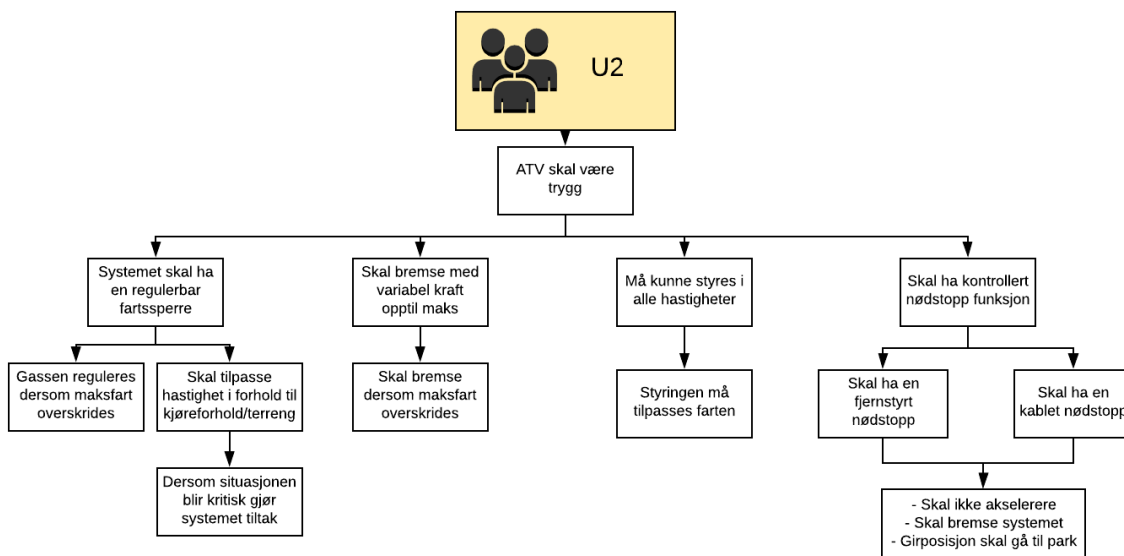
Ved hjelp av disse, kan det settes opp tankekart ift krav som kan leses ut fra user casene. Dette gjør det enklere å se hvor alle krav har sine opphav fra.

Use Case U1:



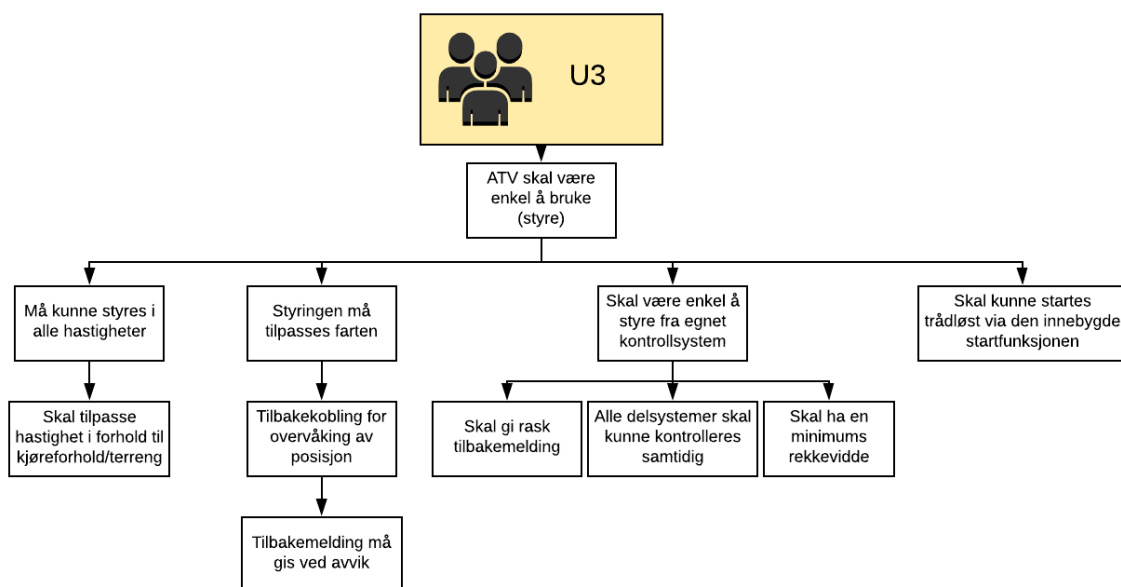
Figur 6: Use case U1

Use Case U2:



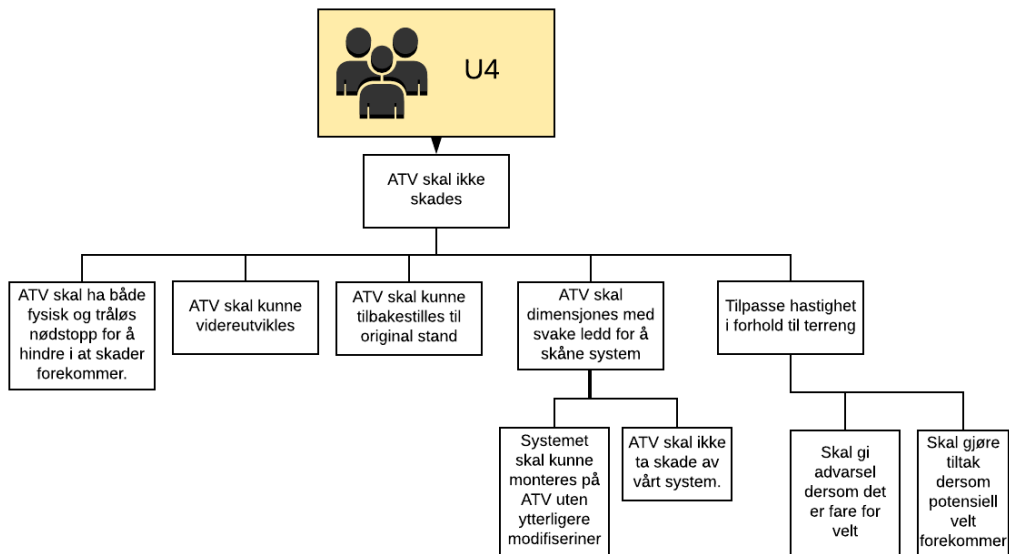
Figur 7: Use case U2

Use Case U3:



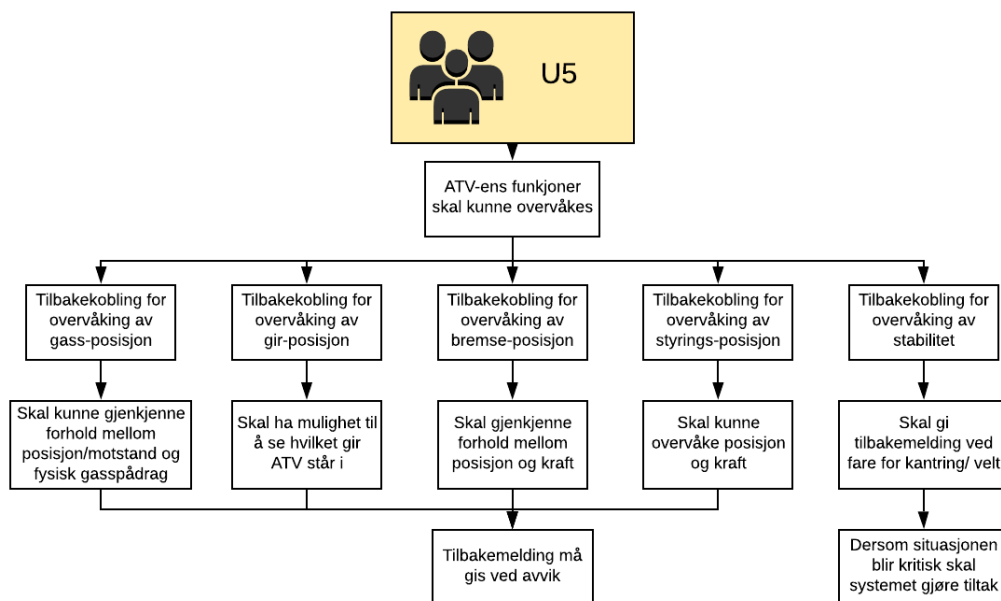
Figur 8: Use case U3

Use Case U4:



Figur 9: Use case U4

Use Case U5:

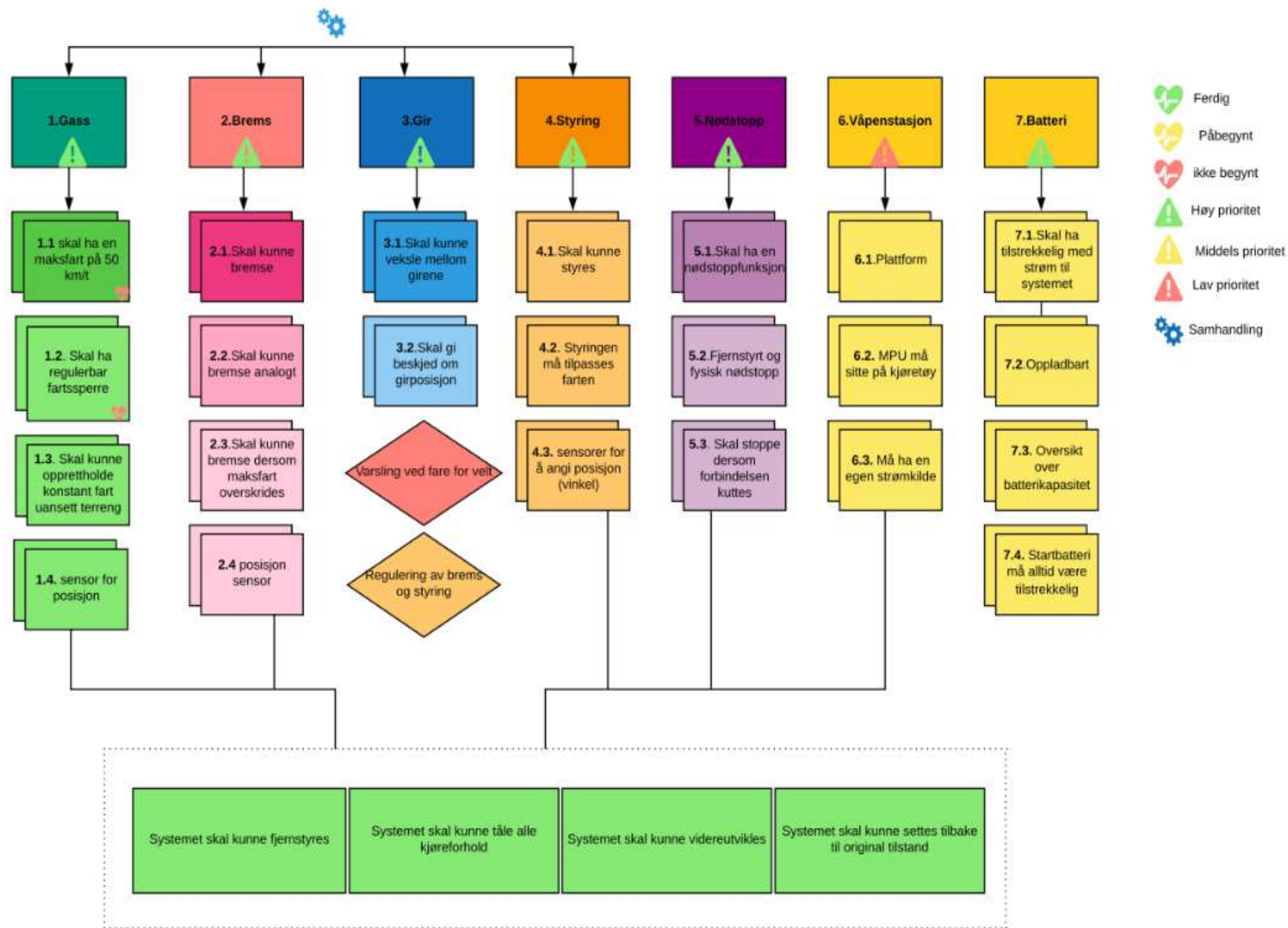


Figur 10: Use case U5

User casene er satt opp i Lucidchart [41].

6.3 1. iterasjon til krav

Basert på use caser og innspill fra ØG ble det tatt et valg om å prioritere gass, brems, gir og styring i første omgang. Disse skal kunne fjernstyres, for så å kunne videreutvikles til å bli autonome i senere prosjekter. Designet skal også ta hensyn til at systemet skal kunne få plass til en våpenstasjon med tilhørende eksternt batteri og en plattform med drone. Figur 11 er satt opp i Lucidchart [41] og viser funksjonskravene som tenkt fram til nå. Dette er en oversikt over de overordnede funksjonene sammen med de viktigste kravene til systemet.



Figur 11: Krav til systemet

6.4 Kvalitetssikring

For å sikre at systemene som designes holder mål, skal det gjennomføres flere tiltak for å kvalitetssikre produktet. Det skal brukes forskjellige modeller som kan forsvare de enkelte valg som blir tatt. Dette gjøres for å ta beslutninger når alternativene ikke nødvendigvis gir et entydig svar. Et eksempel på dette er Pugh-matriser som kan være hjelpelige til å sammenligne de ulike alternativene, for å kunne sikte seg inn på de riktige løsningene.

Det skal være mulig å demonstrere delsystemer i løsningen. Disse skal kunne fungere like godt alene som når de er i samhandling med hverandre. Delsystemene skal tilfredstille iht. kravene som blir satt og kunne demonstreres på en effektiv og trygg måte. I dialog med ØG ble det klart at alt av deler og komponenter skal holde industristandard. Det vil si at komponenter skal være av god kvalitet i forhold til hva systemet skal kunne yte, tåle og i forhold til forventet levetid.

Det blir derfor nødvendig å bruke følgende metoder for å validere om produktet (komponenter, aktuatorer og festeinnretninger) holder standarden:

- Simuleringsverktøy
- Småskala-testing
- Styrkeberegning

Det vil også bli utført testkjøring av ATV-en for å verifisere at systemet fungerer. Testingen vil forgå på et avsperrt område i teknologiparken.

6.5 Krav til delsystem

Hvert enkelt delkrav vil bli beskrevet med bokstavkoder etterfulgt av et tall, slik at det enkelt kan refereres til fra tester og risikoer. Her skal det utdypes mer om kravene til hvert enkelt delsystem. Hvert krav har en prioritet som er rangert fra A - C, der A har høyest prioritet og C har lavest prioritet. Prioriteten settes etter hvert krav. Kravene er utarbeidet og basert på user casene.

6.5.1 Overordnet

Krav-ID: O

Tabell 4: Overordnede krav

Hovedkategori	Krav ID	Krav	Prioritet
Overordnet	O.01	Systemet skal kunne videreutvikles	A
	O.02	Kjøretøyet skal kunne tilbakestilles til orginal stand (Ingen permanente løsninger)	A
	O.02.01	Systemet skal kunne monteres på ATV uten ytterligere modifiseringer	A
	O.03	Systemet skal kunne opereres i alle forhold	A
	O.04	Systemet skal ha industristandard	A
	O.05	System skal dimensjoneres med svake ledd for å skåne ATV	A
	O.06	Sikkerhetiltak som følge av systemsvikt	A
	O.07	Systemet skal være brukervennlig	A
	O.08	Systemet skal ha en regulerbar fartssperre	A
	O.08.01	Skal regulere gassen dersom maksfart overskrides	A
	O.09	Skal dimensjoneres for en hastighet på minimum 50km/t	A
	O.10	Skal bremse dersom maksfart overskrides	A
O.11	Skal kun skifte gir når ATV-en er stillestående og brems er aktivert	A	
O.12	Systemet skal kjøre en fullstendig systemsjekk før start av ATV-en	A	

6.5.2 Gass

Krav-ID: **GA**

Tabell 5: Krav gass

Hovedkategori	Krav ID	Krav	Prioritet
Gass	GA.01	Skal kunne gasse med variabelt pådrag	A
	GA.01.01	Tilbakekobling for regulering	A
	GA.02	Skal kunne overvåke posisjon/motstand	A
	GA.02.01	Skal kunne gjenkjenne forhold mellom posisjon/motstand og fysisk gasspådrag	A

6.5.3 Brems

Krav-ID: **BR**

Tabell 6: Krav brems

Hovedkategori	Krav ID	Krav	Prioritet
Brems	BR.01	Skal ha muligheten til å bremse med variabel kraft opptil maks	A
	BR.02	Skal kunne overvåke posisjon/kraft	A
	BR.02.01	Systemes skal gjenkjenne endringer i forholdet mellom posisjon og kraft	A

6.5.4 Gir

Krav-ID: **GI**

Tabell 7: Krav gir

Hovedkategori	Krav ID	Krav	Prioritet
Gir	GI.01	Systemet skal kunne veksle mellom girene	A
	GI.01.01	Skifte av gir må skje i en bestemt bevegelse	A
	GI.02	Skal kunne overvåke girposisjon (monitor)	A
	GI.03	Skal kunne gjenkjenne kraftbruk	A
	GI.04	Skal alltid stå i park eller nøytral når ATV starter eller stopper	A
	GI.05	Må ha muligheten til å se hvilket gir ATV-en er i.	A

6.5.5 Styring

Krav-ID: ST

Tabell 8: Krav styring

Hovedkategori	Krav ID	Krav	Prioritet
Styring	ST.01	Må kunne styres i alle hastigheter	A
	ST.02	Styringen må tilpasses farten	A
	ST.03	Skal kunne overvåke posisjon/kraft (monitor)	A
	ST.03.01	Må gi tilbakemelding om avvik	A

6.5.6 Nødstop

Krav-ID: NS

Tabell 9: Krav nødstop

Hovedkategori	Krav ID	Krav	Prioritet
Nødstop	NS.01	Kontrollert nødstop	A
	NS.01.01	Skal stoppe gass	A
	NS.01.02	Må bremse systemet	A
	NS.01.03	Systemet skal gjøre en kontrollert nødstop i forhold til hastigheten	C
	NS.01.04	Fjernstyrt nødstop	A
	NS.01.05	Settes i park-giret ved nødstop	A
	NS.02	Systemet skal gjøre en systemsjekk ved oppstart eller kjøring, ved feil stoppes systemet	A
	NS.02.01	Skal gjøre kontrollert nødstop dersom forbindelsen kuttes	A
	NS.02.02	Dersom styring ikke gir tilbakemelding	A
	NS.02.03	Dersom gass ikke gir tilbakemelding	A
	NS.02.04	Dersom brems ikke gir tilbakemelding	A
	NS.02.05	Dersom kantring systemet ikke gir tilbakemelding	B
	NS.02.06	Dersom batteri ikke gir tilbakemelding	B
	NS.03	Kritisk nødstop	A
	NS.03.01	Kablet nødstop	A
	NS.03.02	Fysisk nødstop på ATV	A

6.5.7 Våpenstasjon

Krav-ID: VS

Tabell 10: Krav våpenstasjon

Hovedkategori	Krav ID	Krav	Prioritet
Våpenstasjon	VS.01	En våpenstasjon skal kunne festes enkelt på ATV	B
	VS.01.01	Våpenstasjonen må ikke gjøre ATV ustabil	B
	VS.02	MPU - Main Processing Unit må sitte på ATV	C
	VS.03	Våpenstasjon må ha en egen strømkilde	C
	VS.04	DCP - Display Control Panel må sitte på ATV	C

6.5.8 Batteri

Krav-ID: BA

Tabell 11: Krav batteri

Hovedkategori	Krav ID	Krav	Prioritet
Batteri	BA.01	Batteriet skal gi nok strøm til å dekke hele systemet og våpenstasjon	A
	BA.02	Batteriet skal være oppladbart	A
	BA.03	Dersom batteriet har lite strøm skal det komme et varsel	B
	BA.04	Batteriet skal kunne kobles på en dynamo slik at systemet lader opp batteriet under bruk	B
	BA.05	Aldri tømme startbatteriet	A

6.5.9 Velt

Krav-ID: VE

Tabell 12: Krav velt

Hovedkategori	Krav ID	Krav	Prioritet
Velt	VE.01	Skal tilpasse hastighet i forhold til kjøreforhold/terreng	B
	VE.02	Skal gi advarsel dersom det er fare for velt	B
	VE.02.01	Dersom situasjonen blir kritisk, gjør systemet tiltak	C

6.5.10 Fjernstyring

Krav-ID: **FS**

Tabell 13: Krav fjernstyring

Hovedkategori	Krav ID	Krav	Prioritet
Fjernstyring	FS.01	Systemet skal være enkelt å styre fra egnet kontrollsystem	A
	FS.02	Systemet skal ikke kunne styres fra andre enn de som skal ha tilgang	C
	FS.03	Skal ha rask responstid	A
	FS.04	Nødstop skal ha en ekstra forbindelse med egen kontrollenhet	A
	FS.05	Skal ha en satt minimumsrekkevidde	A
	FS.06	Alle delsystemer skal kunne kontrolleres samtidig	A

6.6 UML: Overordnet

Det er viktig for å forstå hvordan systemet skal operere, i hvilken rekkefølge delsystemene fungerer. Det er derfor oversiktlig å sette opp systemet og dets funksjoner i et flowchart. Her kan man se hvordan systemet skal oppføre seg innenfor en vanlig syklus; fra start til stopp.

UML ATV flowchart



Ola Loe

Figur 12: UML: Overordnet av systemets funksjoner

6.7 2. Iterasjon til krav

Underveis i prosjektet har et utvalg av kravene vist seg å være mer krevende og innfri enn andre. Noen av kravene har også blitt nedprioritert på grunn av tid. På grunn av knapp tid, ble noen av kravene omprioritert. Alle krav var ikke like viktig for å få et fungerende system. I tabell 14 er kravene som ikke har blitt innfridd under prosjektperioden og eventuelle endringer blitt listet opp.

Tabell 14: Krav som ikke er blitt møtt eller tilføyd

Hovekategori	Krav ID	Krav	Prioritet	Kommentar
Overordnet	O.06	Sikkerhetstiltak som følge av systemsvikt - Tiltak iverksettes dersom systemet ikke gir tilbakemelding om at alt fungerer	A	
Gir	GI.03	Skal kunne gjenkjenne kraftbruk	A	Fjernes da det ikke er relevant
	GI.05	Må ha muligheten til å se hvilket gir ATV-en er i	A	Fjernes, kravet overlapper med GI.02
Styring	ST.01	Må kunne styres i alle hastigheter	A	Per dags dato ustabil å styre -> ikke møtt
	ST.04	Skal svinge kontrollert og jevnt	A	Nytt krav
	ST.05	Skal oppnå maks rattutslag på et sekund	A	Nytt krav
Brems	BR.03	Skal oppnå full bremseeffekt i løpet av et sekund	A	Nytt krav
Nødstop	NS.01.05	Settes i park-giret ved nødstop	A	Denne funksjonen er ikke møtt
	NS.02	Systemet skal gjøre en systemsjekk ved oppstart eller kjøring, ved feil stoppes systemet	A	Dette kravet samt underkravene er ikke møtt - ikke prioritert
	NS.02.01	Skal gjøre kontrollert nødstop dersom forbindelsen kuttes	A	Kravet er ikke møtt
	NS.02.02	Dersom styring ikke gir tilbakemelding	A	Kravet er ikke møtt
	NS.02.03	Dersom gass ikke gir tilbakemelding	A	Kravet er ikke møtt
	NS.02.04	Dersom brems ikke gir tilbakemelding	A	Kravet er ikke møtt
	NS.02.05	Dersom kantringsys. ikke gir tilbakemelding	B	Kravet er ikke møtt
	NS.02.06	Dersom batteri ikke gir tilbakemelding	B	Kravet er ikke møtt
Våpenstasjon	VS.02	MPU - Main Processing Unit må sitte på ATV	C	Ikke lenger relevant, annen type kontroll-enhet vil festes til ramme på ATV.
	VS.03	Våpenstasjon må ha egen strømkilde	C	Ikke relevant under prosjektperioden, KDA vil ta hånd om dette senere
	VS.04	DCP - Display Control Panel må sitte på ATV	C	Ikke lenger relevant, annen type kontroll-enhet vil festes til ramme på ATV.
Batteri	BA.03	Dersom batteriet har lite strøm skal det komme et varsel	B	Kravet er ikke møtt
	BA.05	Aldri tømme startbatteriet	A	Kravet er ikke møtt
Velt	Alle		B	Ikke prioritert under prosjektperioden
Fjernstyring	FS.05	Skal ha en satt minimumsrekkevidde	A	Ingen spesifikk minimumsrekkevidde er satt

Nedenfor følger en mer beskrivende forklaring på årsaken til endring og/eller hvorfor kravene ikke er blitt tilfredsstilt.

Overordnet - O.06

Sikkerhetstiltak som følge av systemsvikt: Dette kravet jobbes med å møte, men på grunn av tiden til rådighet er det usikkert om kravet vil bli tilfredsstilt.

Gir - GI.03

Skal kunne gjenkjenne kraftbruk: Dette kravet er ikke relevant i forhold til girsystemet med aktuator og blir derfor fjernet. Aktuatorene vil alltid flytte girspaken med samme kraft, og bli stående i denne posisjonen. Når ATV-en er stillestående kan man endre girposisjon på nytt.

Gir - GI.05

Må ha muligheten til å se hvilket gir ATV-en er i: Dette kravet fjernes da det overlapper med GI.02 - "Skal kunne overvåke girposisjon (monitor)".

Styring - ST.01

Må kunne styres i alle hastigheter: Dette kravet viser seg å være mer utfordrende å møte enn antatt. Siden ATV-en skal fjernstyres må stepper-motoren kontrolleres via en Arduino og relèer. Dette har vist seg å opptre mer ustabil enn det bør og kravet er derfor ikke innfridd.

Nødstop - NS.02

Systemet skal gjøre en systemsjekk ved oppstart eller kjøring, ved feil stoppes systemet: Dette kravet, samt underkravene, har ikke blitt tilfredsstilt til det nivået som var ønsket på grunn av mangel på tid. Den kablede- og fjernstyrte nødstoppen gjør likevel at ATV-en vil stoppe dersom feil og mangler oppdages.

Våpenstasjon - VS.02-VS.04

MPU - Main Processing Unit og DCP - Display Control Panel må sitte på ATV: Disse er ikke lenger aktuelle i forhold til plassering på ATV, men erstattes av en annen komponent som må ha plass på ramme.

Våpenstasjon - VS.04

Våpenstasjon må ha egen strømkilde: Dette vil ikke være relevant under prosjektpe-

rioden og kravet vil derfor ikke innfris.

Batteri - BA.03

Dersom batteriet har lite strøm skal det komme et varsel: Det har ikke blitt prioritert eller satt av tid til å møte dette kravet da dette ikke er prekært i forhold til resten av systemet.

Batteri - BA.05

Aldri tømme startbatteriet: Dette kravet er foreløpig ikke tilfredsstilt og faren for at dette vil skje er til stede. Det vurderes og anskaffe et ekstra batteri for å unngå at dette skal skje.

Velt - Alle krav

VE.01 - Skal tilpasse hastighet i forhold til kjøreforhold/terreng

VE.02 - Skal gi advarsel dersom det er fare for velt

VE.02.01 - Dersom situasjonen blir kritisk, gjør systemet tiltak

Disse kravene har ikke blitt prioritert under prosjektarbeidet på grunn av tiden som har vært til rådighet.

Fjernstyring - FS.05

Skal ha en satt minimumsrekkevidde: Dette kravet er ikke tilfredsstilt, men av sikkerhetsmessige årsaker vil ATV-en ha en kablet nødstop som gjør at rekkevidden kun vil være på 10 meter.

6.7.1 Nye krav

Styring - ST.04

Skal svinge kontrollert og jevnt: Under testing av stepper-motor til styring kom det frem at kravene som var satt ikke dekket ønskene for hvordan systemet skal driftes på en god nok måte. Dette ble dermed definert som et nytt krav.

Styring - ST.05

Skal oppnå maks rattutslag på et sekund: Det har blitt definert et krav fra KDA om at ATV-en skal kunne oppnå maks rattutslag (0 - 45 grader) i løpet av ett sekund.

Brems - BR.03

Skal oppnå full bremseeffekt i løpet av et sekund: Man skal kunne oppnå full bremseeffekt i løpet av et sekund.

6.7.2 Oppdatert liste av krav

Figur 13 viser en oppdatert liste av alle krav. Ved kravene som har blitt testet finner man også en eller flere test-IDer.

Prioritets ID:					
A	Krav som skal og må oppfylles				
B	Krav som bør oppfylles				
C	Krav som ikke nødvendigvis må oppfylles under bachelorprosjektet, men de må tas høyde for med tanke på videreutvikling				
Hovedkategori	Krav ID	Krav	Test	Prioritet	Kommentar
Overordnet	O.01	Systemet skal kunne videreutvikles	TP.2-O.01	A	Skriver en teknisk manual, sørger for at det som implementeres lett kan gjøres om på senere
	O.02	Kjøretøyet skal kunne tilbakestilles til original stand (Ingen permanente løsninger)	TP.2-O.01	A	Skal kunne gjøres iløpet av en arbeidsdag
	O.02.01	Systemet skal kunne monteres på ATV uten ytterligere modifiseringer		A	Ikke kuttes, borres eller sveises
	O.03	Systemet skal kunne opereres i alle forhold		A	Alt av sårbar elektronikk må værnes, ta høyde for rust på ramme mm.
	O.04	Systemet skal ha industristandard		A	
	O.05	System skal dimensjoneres med svake ledd for å skåne ATV		A	Det vi lager skal gå i stykker før deler på ATV.
	O.06	Sikkerhetstiltak som følge av systemsvikt		A	Tiltak iverksettes dersom systemet ikke gir tilbakemelding om at alt fungerer
	O.07	Systemet skal være brukervennlig	TP.6-O.07	A	
	O.08	Systemet skal ha en regulerbar fartssperre	TP.2-O.01 TT.4-O.08 TP.18-GA	A	
	O.08.01	Skal regulere gassen dersom maksfart overskrides		A	
	O.09	Skal dimensjoneres for en hastighet på minimum 50km/t	TP.2-O.01 TT.4-O.08	A	
	O.10	Skal bremse dersom maksfart overskrides	TT.4-O.08 TP.32-BR.01	A	
	O.11	Skal kun skifte gir når ATV-en er stillestående og brems er aktivert		A	
O.12	Systemet skal kjøre en fullstendig systemsjekk før start av ATV-en		A		
O.13	ATV-en skal kunne fjernstyres	TP.10-O.13 TP.11-O.13 TP.12-O.13 TP.13-O.13 TP.14-O.13	A		
Gass	GA.01	Skal kunne gasse med variabelt pådrag	TP.3-GA.01 TT.4-O.08 TP.13-O.13 TP.15-GA.01 TT16-GA.01 TP.18-GA TI.27-GA.01 TP.28-GA.01	A	Analog styring
	GA.01.01	Tilbakekobling for regulering	TP.18-GA TT.19-GA.01.01	A	Sensor for posisjon eller motstand
	GA.02	Skal kunne overvåke posisjon/motstand	TT.4-O.08 TP.13-O.13 TP.18-GA	A	
	GA.02.01	Skal kunne gjenkjenne forhold mellom posisjon/motstand og fysisk gasspådrag	TP.18-GA	A	For å kunne gjenkjenne eventuelle feil
Brems	BR.01	Skal ha muligheten til å bremse med variabel kraft opptil maks	TP.7-BR.01 TP.8-BR.01 TP.9-GI.01 TP.32-BR.01	A	Analog styring
	BR.02	Skal kunne overvåke posisjon/kraft	TP.1-GI.02 TP7-BR.01 TP.25-BR.02 TP.26-BR.02	A	Tilbakekobling som forteller pådrag og kraft
	BR.02.01	Systemet skal gjenkjenne endringer i forholdet mellom posisjon og kraft	TP.32-BR.01	A	For å kunne gjenkjenne feil i systemet
	BR.03	Skal oppnå full bremseeffekt i løpet av et sekund		A	Må ha et minimumskrav til ytelse
Gir	GI.01	Systemet skal kunne veksle mellom girene	TP.8-BR.01 TP.9-GI.01 TP.23-GI.01	A	Høy, lav, nøytral, revers, park

	GI.01.01	Skifte av gir må skje i en bestemt bevegelse		TP.23-GI.01	A	
	GI.02	Skal kunne overvåke girposisjon (monitor)		TP.1-GI.02 TP.7-BR.01 TP.23-GI.01	A	
	GI.03	Skal kunne gjenkjenne kraftbruk			A	For å kunne gjenkjenne feil
	GI.04	Skal alltid stå i park eller nøytral når ATV starter eller stopper			A	
Styring	ST.01	Må kunne styres i alle hastigheter		TP.8-BR.01 TP.29-ST.04 TP.30-ST.04 TP.31-ST.04	A	
	ST.02	Styringen må tilpasses farten		TP.8-BR.01 TP.29-ST.04 TP.30-ST.04 TP.31-ST.04	A	
	ST.03	Skal kunne overvåke posisjon/kraft (monitor)		TP.22-ST.04 TP.24-ST.04 TP.29-ST.04 TP.30-ST.04 TP.31-ST.04 TP.33-ST.04	A	Maks svingradius
	ST.03.01	Må gi tilbakemelding om avvik		TP.20-NS.02.02 TP.21-NS.02.02 TP.29-ST.04 TP.30-ST.04 TP.31-ST.04 TP.33-ST.04 TP.34-ST.04	A	
	ST.04	Skal svinge kontrollert og jevnt		TP.22-ST.04 TP.24-ST.04 TP.29-ST.04 TP.30-ST.04 TP.31-ST.04 TP.34-ST.04	A	
	ST.05	Skal oppnå maks rattutslag på et sekund		TP.22-ST.04	A	45 grader
Nødstop	NS.01	Kontrollert nødstop			A	
	NS.01.01	Skal stoppe gass			A	
	NS.01.02	Må bremse systemet			A	
	NS.01.03	Systemet skal gjøre en kontrollert nødstop i forhold til hastigheten			C	Ikke kritisk under bachelorprosjektet, men burde komme på plass til kjøring i høy hastighet
	NS.01.04	Fjernstyrt nødstop			A	
	NS.01.05	Settes i park-giret ved nødstop			A	
	NS.02	Systemet skal gjøre en systemsjekk ved oppstart eller kjøring, ved feil stoppes systemet			A	
	NS.02.01	Dersom forbindelsen kuttes			A	Ved senere tidspunkt kan dette endres til å kjøre tilbake til angitt posisjon
	NS.02.02	Dersom styring ikke gir tilbakemelding			A	
	NS.02.03	Dersom gass ikke gir tilbakemelding		TP.20-NS.02.02 TP.21-NS.02.02	A	
	NS.02.04	Dersom brems ikke gir tilbakemelding			A	
	NS.02.05	Dersom kantring systemet ikke gir tilbakemelding			B	
	NS.02.06	Dersom batteri ikke gir tilbakemelding			B	
	NS.03	Kritisk nødstop			A	Skrur av tenning til ATV og slår av vårt system
	NS.03.01	Kablet nødstop			A	
	NS.03.02	Fysisk nødstop på ATV			A	
	NS.03.03	Fjernstyrt nødstop			A	
Våpenstasjon	VS.01	En våpenstasjon skal kunne festes enkelt på ATV			B	Plate i bunn - bolter - to kraftige kabler en til MPU og en til batteri
	VS.01.01	Våpenstasjonen må ikke gjøre ATV ustabil			B	
	VS.02	Komponenter til våpenstasjon må sitte på ATV			C	
	VS.03	Våpenstasjon må ha en egen strømkilde			C	
Batteri	BA.01	Batteriet skal gi nok strøm til å dekke hele systemet og våpenstasjon			A	

	BA.02	Batteriet skal være oppladbart			A	
	BA.03	Dersom batteriet har lite strøm skal det komme et varsel			B	
	BA.04	Batteriet skal kunne kobles på en dynamo slik at systemet lader opp batteriet under bruk			B	
	BA.05	Aldri tømme startbatteriet			A	
Velt	VE.01	Skal tilpasse hastighet i forhold til kjøreforhold/terreng			B	
	VE.02	Skal gi advarsel dersom det er fare for velt			B	
	VE.02.01	Dersom situasjonen blir kritisk, gjør systemet tiltak			C	F.eks ved tipping bakover i oppoverbakke
Fjernstyring	FS.01	Systemet skal være enkelt å styre fra egnet kontrollsystem		TP.6.O.07 TP.17-FS.01 TP.24-ST.04 TP.30-ST.04 TP.31-ST.04	A	
	FS.02	Systemet skal ikke kunne styres fra andre enn de som skal ha tilgang			C	Viktig punkt, men ikke under vår prosjektperiode
	FS.03	Skal ha rask responstid		TP.17-FS.01	A	
	FS.04	Nødstop skal ha en ekstra forbindelse med egen kontrollenhet			A	
	FS.05	Skal ha en satt minimumsrekkevidde			A	Avhenger litt av løsning - 4G / WLAN
	FS.06	Alle delsystemer skal kunne kontrolleres samtidig			A	
Annet	AN.01	Skal kunne velge firhjulsdrift eller tohjulsdrift			C	
	AN.02	ATV-en skal kunne startes trådløst via den innebygde startfunksjonen			B	
	AN.02.01	Startknappen må kunne brukes som vanlig etter at vårt system er nedmontert			A	

Figur 13: Oppdatert kravliste inkludert test-ID

Risiko

En generell beskrivelse om bruk av modeller i arbeid med å kartlegge risiko og sannsynligheten for at en eller flere av disse skal oppstå, samt alvorlighetsgraden av disse.

Innhold:

- Generelt om risiko
- Systemrisiko
- Prosjektrisiko



KONGSBERG



USN Universitetet
i Sørøst-Norge

7.1 Generelt om risiko

For å identifisere risikoelementer vil alle prosjektdelene bli gjennomgått slik at man kan finne sannsynligheten for at noe galt kan forekomme. Sannsynligheten for at en risiko vil kunne oppstå blir tallfestet med tallene 1 - 5, hvor:

1 = Liten sannsynlighet

5 = Meget stor sannsynlighet

Videre vil konsekvensen av at dette skjer også tallfestes:

1 = Liten konsekvens for prosjektets mål, tidsfrist og rammer

5 = Meget stor konsekvens for prosjektets mål, tidsfrist og rammer

Som figur 14 indikerer, vil risikoen bli beregnet ut ifra sannsynlighet og konsekvens, slik at:

$$\text{Risiko} = \text{Sannsynlighet} \times \text{Konsekvens}$$

Det vil bli sett på sannsynligheten for at et tenkt scenario oppstår og i hvor stor grad dette kan påvirke prosjektet negativt. Ut ifra denne informasjonen tas det vurderinger i hvilken grad det skal forebygges for å unngå at de ulike risikoscenarioene forekommer.

Risikonivå	1 Akseptabelt	2 Tolererbart	3 Uønsket	4 Ikke-tolererbart	5 Kritisk
1 Svært usannsynlig	1	2	3	4	5
2 Usannsynlig	2	4	6	8	10
3 Sannsynlig	3	6	9	12	15
4 Svært sannsynlig	4	8	12	16	20
5 Kommer til å skje	5	10	15	20	25

Figur 14: Illustrasjon av risikovurdering

Tabell 15: Risikorangering (1-5)

Risiko	Beskrivelse	Sannsynlighet	Konsekvens	Klassifisering	Tiltak	Ansvarlig
25		Kommer til å skje	Kritisk	Kritisk		
16		Svært sannsynlig	Ikke tolererbart	Kritisk		
9		Sannsynlig	Uønsket	Alvorlig		
4		Usannsynlig	Tolererbart	Neglisjerbar		
1		Svært usannsynlig	Akseptabelt	Neglisjerbar		

Tabellene skal bidra til at man kan utarbeide en løsning som skal forhindre at et risiko-scenario inntreffer. Tabellene skal også gi et innblikk i hvilke risikoer man skal prioritere først, dersom flere risikoer inntreffer samtidig; Hva som eventuelt kan neglisjeres og hvilke som krever umiddelbar handling og ressursbruk.

Tabell 16 viser et eksempel på hvordan klassifisering av risikorangering vil fremvises. Det blir nødvendig å gjøre en slik risikoanalyse for alle funksjoner og krav.

Tabell 16: Eksempel på risiko som kan inntreffe

Risiko	Beskrivelse	Sannsynlighet	Konsekvens	Klassifisering	Tiltak	Ansvarlig
15	Gass- oppheng	Sannsynlig	Kritisk	Kritisk	Nødstopp som overstyrer systemet. (Ekstra Arduino)	KVL

7.2 Systemrisiko

Her er en analyse av mulige risikohendelser for hvert enkelt krav i delsystemene. Risikoene vil bli rangert i et tallsystem fra **1 - 25** og er delt inn i tre hovdekategorier:

1 - 5	Risiko som ikke krever umiddelbare tiltak
6 - 12	Risiko som krever tiltak
15 - 25	Risiko som krever umiddelbare tiltak

Risiko-ID blir etablert fra alvorlighetsgraden til hendelsen, etterfulgt av : **R1** - Krav-ID

Ett krav kan gi flere risikohendelser, ved disse hendelsene vil det bli etterfulgt av påfølgende tall, f.eks:

4-R1-ST.01: Risiko for krav ST.01, med alvorlighetsgrad 4.

10-R2-ST.01: Ekstra risiko for kravet med alvorlighetsgrad 10

Dersom en risiko får alvorlighetsgrad på 6 eller over, vil det bli satt inn mulige tiltak for å unngå denne hendelsen som vist i figur 15.

		Prioritets ID:							
		1 - 5	Risiko som ikke krever umiddelbare tiltak						
		6 - 12	Risiko som krever tiltak						
		15 - 25	Risiko som krever umiddelbare tiltak						
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar	
Overordnet	O.01	R1-O.01	Systemet blir for dårlig dokumentert til å vise hva vi har gjort	2	4	8	Lage eget dokument til overtagelse av systemet		
	O.02	R1-O.02	ATV blir endret permanent	1	4	4			
	O.02.01	R1-O.02.01	Systemet kan ikke monteres uten ytterligere modifiseringer på ATV	1	4	4			
	O.03	R1-O.03	Systemet blir ikke tilpasset i forhold til alt vær	2	4	8	Designe bokser til sensorer og aktuatorer som er vannavvisende, samt utstyr som tåler aktuelle temperaturer		
	O.04	R1-O.04	Systemet opprettholder ikke god nok standard	1	5	5			
	O.05	R1-O.05	Systemet skader ATV under drift	1	5	5			
	O.06	R1-O.06	Sikkerhetstiltak slår ikke inn ved systemsvikt	2	5	10	Systemet skal ikke kunne operere dersom det oppdages svikt (kablet nødstop)		
	O.07	R1-O.07	Systemet blir ikke brukervennlig	1	3	3			
	O.08	R1-O.08	Fartssperren virker ikke	3	5	15	Kablet og fjernstyrt nødstop		
	O.08.01	R1-O.08.01	Regulering av gassen slår ikke inn automatisk når maksfart overskrides	3	5	15	Kablet og fjernstyrt nødstop		
	O.09	R1-O.09	Oppnår ikke hastigheten på minimum 50 km/t	3	1	3			
	O.10	R1-O.10	Systemet bremses ikke når maksfart overskrides	3	5	15	Kablet og fjernstyrt nødstop		
	O.11	R1-O.11	ATV er ikke stillestående under skifte av gir	3	5	15	Oversiktsbilde over hastigheten, slik at man med sikkerhet kan vite at ATV er stillestående	Ikke aktuelt under vårt prosjekt, da vi kommer til å ha ATV under oppsyn hele tiden	
		R2-O.11	Brems er ikke aktivert under skifte av gir	3	5	15	Sensor som viser at brems er trykt inn før skifte av gir foretas		
	O.12	R1-O.12	Systemet kjører ikke fullstendig systemsjekk før oppstart	3	4	12	Systemet kan ikke starte dersom en systemsjekk ikke foretas - Rapport om utførte oppgaver.		
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar	
Gass	GA.01	R1-GA.01	Gir ikke gass med variabelt pådrag	3	5	15			
	GA.01.01	R1-GA.01.01	Tilbakekobling virker ikke	2	5	10			
	GA.02	R1-GA.02	Har ikke kontroll over posisjon og motstand.	3	4	12			
	GA.02.01	R1-GA.02.01	Gjenkjenner ikke forholdet mellom posisjon/motstand og fysisk gasspådrag	2	4	8			
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar	
Brems	BR.01	R1-BR.01	Får ikke bremses med variabel kraft	3	4	12	Fjernstyrt og kablet nødstop		
	BR.02	R1-BR.02	Har ikke kontroll over posisjon og motstand.	3	4	12	Skal rapportere om feil gjennom systemsjekk		
	BR.02-01	R1-BR.02-01	Systemet gjenkjenner ikke forholdet mellom posisjon og kraft	3	4	12	Skal rapportere om feil gjennom systemsjekk		
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar	
Gir	GI.01	R1-GI.01	Systemet veksler ikke mellom girene	3	3	9	Mulig svikt ved sensor eller aktuator. Må foreta tiltak og fikse evt problemer med aktuator og/eller sensor.		
	GI.01.01	R1-GI.01.01	Skifter ikke gir i bestemt bevegelse slik at kreftene blir for store	2	4	8	Må foreta sjekk i girets bevegelsesakse og se etter uregelmessigheter.		
	GI.02	R1-GI.02	Har ikke kontroll over posisjon og motstand.	3	4	12	Skal rapportere om feil gjennom systemsjekk		
	GI.03	R1-GI.03	Får feil kraftforbruk	1	5	5			
	GI.04	R1-GI.04	Står ikke i park eller nøytral når den starter	1	1	1			
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar	
Styring	ST.01	R1-ST.01	Får ikke styrt i lav hastighet	2	2	4			
		R2-ST.01	Får ikke styrt i høy hastighet	2	5	10	Nødstop må kobles inn dersom styre ikke respondere		
	ST.02	R1-ST.02	Styrer ikke i henhold til farten	2	4	8		Ikke aktuelt i vårt prosjekt	
	ST.03	R1-ST.03	Har ikke kontroll over posisjon og motstand	3	5	15	Skal rapportere om feil gjennom systemsjekk - nødstop		
	ST.03.01	R1-ST.03.01	Gir ikke tilbakemelding om avvik	3	5	15	Skal rapportere om feil gjennom systemsjekk - nødstop		

Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Nødstop	NS.01	R1-NS.01	Nødstopfunksjonen(e) svikter	1	5	5		
	NS.01-01	R2-NS.01	Stopper ikke gass	1	5	5		
	NS.01-02	R1-NS.01-02	Bremser ikke	2	4	8	ATV slår seg av	
	NS.01-03	R1-NS.01-03	Systemet tar ikke hensyn til hastigheten under nødstop	5	1	5		Tas ikke hensyn til under vårt prosjekt, da vi skal kjøre i lave hastigheter
	NS.01-04	R1-NS.01-04	Får ikke kontakt med den fjernstyrte nødstoppen	3	5	15	Braker den fysiske nødstoppen	
	NS.01-05	R1-NS.01-05	Den fysiske nødstoppen på ATV virker ikke	1	5	5	Braker den fjernstyrte nødstoppen	
	NS.02	R1-NS.02	Stopper ikke dersom fobindelsen kuttes	2	5	10	Fjernstyrt og kablet nødstop	
	NS.03	R1-NS.03	Systemet kjører uten systemsjekk eller ved feil	2	5	10	Skal rapportere om feil gjennom systemsjekk - systemet virker ikke når rapporten er ufullstendig.	
	NS.03-01	R1-NS.03-01	Får ikke tilbakemelding fra gir	3	3	9		
	NS.03-02	R1-NS.03-02	Får ikke tilbakemelding fra styring	3	4	12		
	NS.03-03	R1-NS.03-03	Får ikke tilbakemelding fra gass	3	5	15		
	NS.03-04	R1-NS.03-04	Får ikke tilbakemelding fra brems	3	5	15		
	NS.03-05	R1-NS.03-05	Får ikke tilbakemelding fra kanteringssystem	3	3	9		
	NS.03-06	R1-NS.03-06	Får ikke tilbakemelding fra batteri	3	3	9		
NS.04	R1-NS.04	Har ikke nødstopp tilkoblet med kabel	1	5	5			
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Våpenstasjon	VS.01	R1-VS.01	Våpenstasjon festes ikke lett på ATV	2	1	2		
	VS.01-01	R2-VS.01	Våpenstasjon gjør ATV ustabil	2	5	10	Stabiliser feste. regulering av tyngdepunkt	
	VS.02	R1-VS.02	MPU er ikke festet på ATV	1	5	5		
	VS.03	R1-VS.03	Våpenstasjon har ikke ekstern strømkilde	1	5	5		
	VS.04	R1-VS.04	DCP er ikke festet på ATV	1	5	5		
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Batteri	BA.01	R1-BA.01	Batteriet gir ikke nok strøm til å dekke hele systemet	2	5	10	Skal rapportere om feil gjennom systemsjekk	
	BA.02	R2-BA.01	Batteriet lader ikke	1	4	4		
	BA.03	R1-BA.03	Varsler ikke ved lav strøm på batteri	2	4	8	Rapportere om mistet signal til batterisensor	Dokumentere kjøretid - så vet vi anslått kjøretid igjen
	BA.04	R1-BA.04	Systemet lader ikke batteriet under bruk	2	2	4		
	BA.05	R1-BA.05	Startbatteriet tømmes for strøm	2	5	10	Rapportere om mistet signal til batterisensor	
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Velt	VE.01	R1-VE.01	Tilpasser ikke hastighet i forhold til terreng	5	5	25		
	VE.02	R2-VE.01	Gir ikke advarsel ved fare for velt	4	5	20		
	VE.02-01	R1-VE.02-01	Systemet gjør ikke nødvendige tiltak ved kritiske situasjoner	4	5	20		
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Fjernstyring	FS.01	R1-FS.01	Systemet er vanskelig å styre	1	4	4		
	FS.02	R2-FS.01	Uvedkommende får tilgang til å styre systemet	3	5	15	Kryptere signalet	Ikke aktuelt for vårt prosjekt
	FS.03	R1-FS.03	Responderer for sakte	1	5	5		
	FS.04	R1-FS.04	Har ingen ekstra forbindele til trådløs nødstop	1	5	5		
	FS.05	R1-FS.05	Fjernstryingen fungerer ikke innenfor angitt rekkevidde	2	5	10	Fjernstyrt og kablet nødstop, stopper ved mistet kontakt	
Hovedkategori	Krav ID	Risiko ID	Beskrivelse	Sannsynlighet	Risk	Alvorlighetsgrad	Mulige tiltak	Kommentar
Annet	AN.01	R-AN.01	Ikke kunne skifte mellom modusene	4	1	4		
	AN.02	R-AN.02	Får ikke startet trådløst	4	1	4		
	AN.02.01	R-AN.02.01	Startknappen virker ikke etter montering	3	1	3		

Figur 15: Risikoanalyse av funksjonskravene til systemet

7.3 Prosjektrisiko

I tillegg til risiko knyttet til de tekniske aspektene rundt prosjektet har gruppen forsøkt å kartlegge noen risikoscenarioer knyttet til det faglige og sosiale. Dette gjøres for å utbedre tiltak som kan minske konsekvensene for at disse inntreffer.

På samme måte som risiko ble kartlagt tidligere er det nedenfor utarbeidet en tabell over noen tenkte projektrisikoscenarioer som kan oppstå.

I motsetning til tidligere er ikke disse risikoene ferdig før prosjektet er ferdig og er noe gruppemedlemmene kontinuerlig må tenke på.

		Prioritets ID:				
		1 - 5	Risiko som ikke krever umiddelbare tiltak			
		6 - 12	Risiko som krever tiltak			
		15 - 25	Risiko som krever umiddelbare tiltak			
Hovedkategori	Beskrivelse	Sannsynlighet	Alvorlighetsgrad	Prioritet	Konsekvens	Mulige tiltak
PR-1	Svak prosjektstyring eller oppfølging av oppgavene i ansvarsområdene.	3	4	12	Prosjektets delmål blir ikke oppfylt	Sørge for at alle vet hva de skal gjøre. Kommunisere til hverandre hvilke oppgaver som er av høyeste prioritet.
PR-2	Grappa følger ikke prosjektmodellen slik som bestemt.	3	3	9	Det ferdige produktet kan bli dårligere fordi viktige innfallsvinkler og/eller krav blir oversett	Medlemmet med hovedansvaret for prosjektmodellen tar opp problemstillinga med gruppa og informerer om hva som må gjøres for å komme tilbake på rett spor.
PR-3	Noen av gruppemedlemmene gjør ikke det de skal eller har blitt bedt om å gjøre.	3	2 - 5	6 - 15	Oppgaver med varierende viktighetsgrad blir ikke gjort i tide, noe som forsinker arbeidet til andre og prosjektet.	Gruppeleder tar opp problemet med vedkommende. Ekstra ressurser settes inn ved behov. Gjentakelser vil føre til konsekvenser i følge gruppekontrakten.
PR-4	Grappa deler ikke informasjon vedrørende prosjektet på en oversiktlig eller beskrivende nok måte.	2	4	8	Informasjonen vi trenger til å fullføre oppgavene våre blir vanskelig å finne. Noe som vil påvirke produktiviteten vår negativt.	Grappa må bruke verktøy som forenkler deling og sørge for et oversiktlig oppsett av dokumenter. Samt kommunisere om oppgavene sine med de andre medlemene.
PR-5	Lengre sykdom eller fravær under prosjektet	2	3	9	Ved og ikke planlegge for lengre sykdom eller fravær kan vi risikere at viktige biter av prosjektet ikke blir gjennomført.	Sørge for at alle vet hva de skal gjøre og hva andre gjør slik at andre kan overta oppgavene ved sykdom. Eller at det tilrettelegges for at den/de syke kan jobbe hjemmefra.
PR-6	Plagiat	1	5	5	Ved feil/manglende kildeføring eller direkte plagiat vil gruppa kunne stryke i faget.	Alle er innforstått med konsekvensene av plagiering. Alle gruppemedlemmene har fått informasjon om riktig kildeføring.
PR-7	Kommunikasjonen mellom gruppemedlemmene er dårlig	2	5	10	Viktig informasjon kan bli borte og samholdet blir dårligere	Tiltak for å forsikre god kommunikasjon
PR-8	Kommunikasjonen mellom gruppa og intern veilder er dårlig	2	5	10	Tips til prosjektet kan bli oversett og nødvendig informasjon som skolen skal ha for å vurdere gruppa kan bli borte	Tiltak for å forsikre god kommunikasjon
PR-9	Kommunikasjonen mellom oppdragsgiver og gruppa er dårlig	2	5	10	Produktet som skal lages kan bli feil i forhold til arbeidsgivers ønske. I tillegg vil vurderingsgrunnlaget for karakteren bli dårligere	Tiltak for å forsikre god kommunikasjon
PR-10	Arbeidsoppgavene i en gitt tidsintervall blir ikke gjennomført innen sprintperioden er ferdig.	4	3	12	Uferdige oppgaver etter sprinter fører til forsinkelse på hele prosjektet.	Daily scrum hjelper oss med å daglig omprioritere oppgaver og hjelpe hverandre med krevende oppgaver.
PR-11	Noen av oppgavene blir for komplekse for den ansvarlige.	4	2	12	Forsinkelse av prosjektet eller dårligere løsninger må til.	Søke råd fra veiledere eller aktuelle lærere. Hjelpe hverandre å dele på oppgavene.
PR-12	Kunnskap forsøkt lært underveis i prosjektet tar for lang tid å lære.	3	4	12	Det går bort mye tid til noe som ikke fører til resultater for prosjektet.	Hva som må læres må bli kartlagt i startfasen av prosjektet og en realistisk tidperiode for læringen må utarbeides så raskt som mulig.
PR-13	Arbeidsmodellen er feil i forhold til systemet som skal lages	3	4	12		Grappa må se på alternative arbeidmodeller eller tilpasse modellen slik at den passer.

Figur 16: Risikoanalyse over prosjektet

Målinger

Her vil det bli beskrevet hva slags målinger som er blitt gjort, både for maskin og elektro.

Innhold:

- Maskinmålinger
- Elektromålinger



Ut ifra kravene trengs det å vite hva som kreves for å bremse, gire og gasse på ATV-en. Denne informasjonen hjelper til med å bestemme hvilke systemer som realistisk sett kan implementeres. Det skal foretas forskjellige målinger slik at det er mulig å vurdere flere løsninger.

8.1 Maskinmålinger

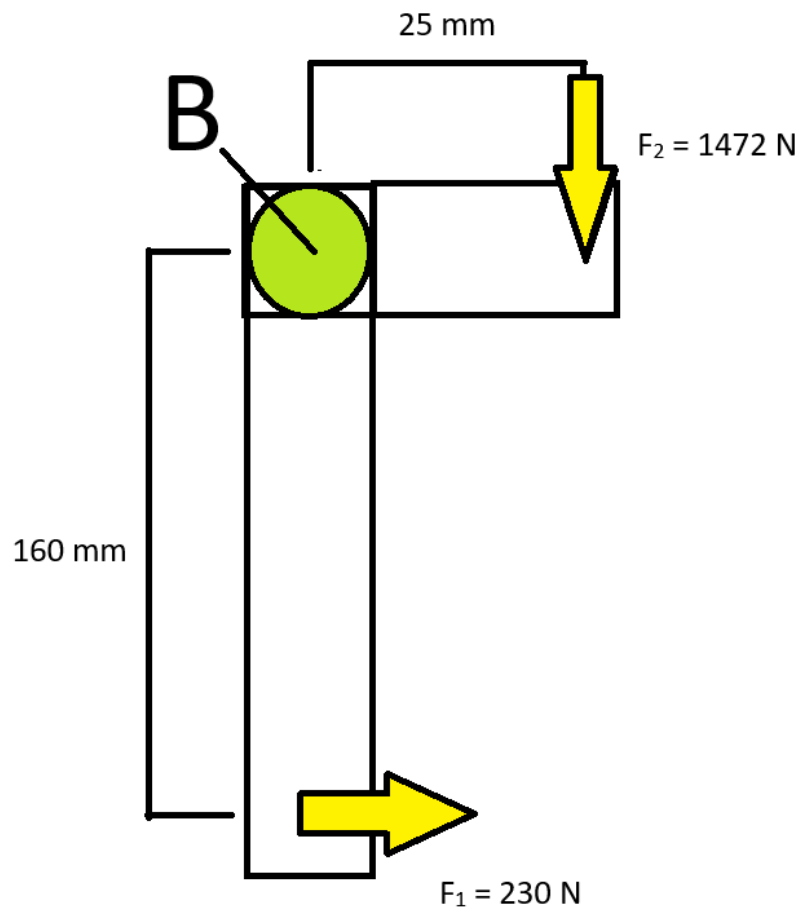
Det ble foretatt styrkeberegninger for å se hvor mye kraft som kreves for å trykke inn hånd- og fotbrems. Det ble brukt en bagasjevekt, målebånd, meterstokk og skyvelære for å etablere hvor stor kraft som trengtes, men også lengden på armen slik at momentet kunne regnes ut.

8.1.1 Kraftmålinger på delsystemene

Hånd- fotbrems og gir

En simplifisert modell av geometrien, kreftene og momentet ble laget for å finne den lineære kraften som kreves for å trykke inn bremsen. (figur 17)

F_1 er den målte kraften med bagasjevekt. Lengden fra hengslet, punkt B, er blitt målt med skyvelære. Utregningene baseres på at det blir brukt en tilnærmet statisk modell. F_2 er stampelet som trykker inn bremsen og genererer reaksjonskrefter når momentet fra F_1 øker.



Figur 17: Statisk beregning av håndbrems

Utgangspunkt av F_2 :

Positiv orientering med klokken.

$$\sum M_b = 0 \quad (1)$$

$$F_2 \times 25\text{mm} - 230\text{N} \times 160\text{mm} = 0 \quad (2)$$

$$F_2 \times 25\text{mm} = 36.8\text{Nm} \quad (3)$$

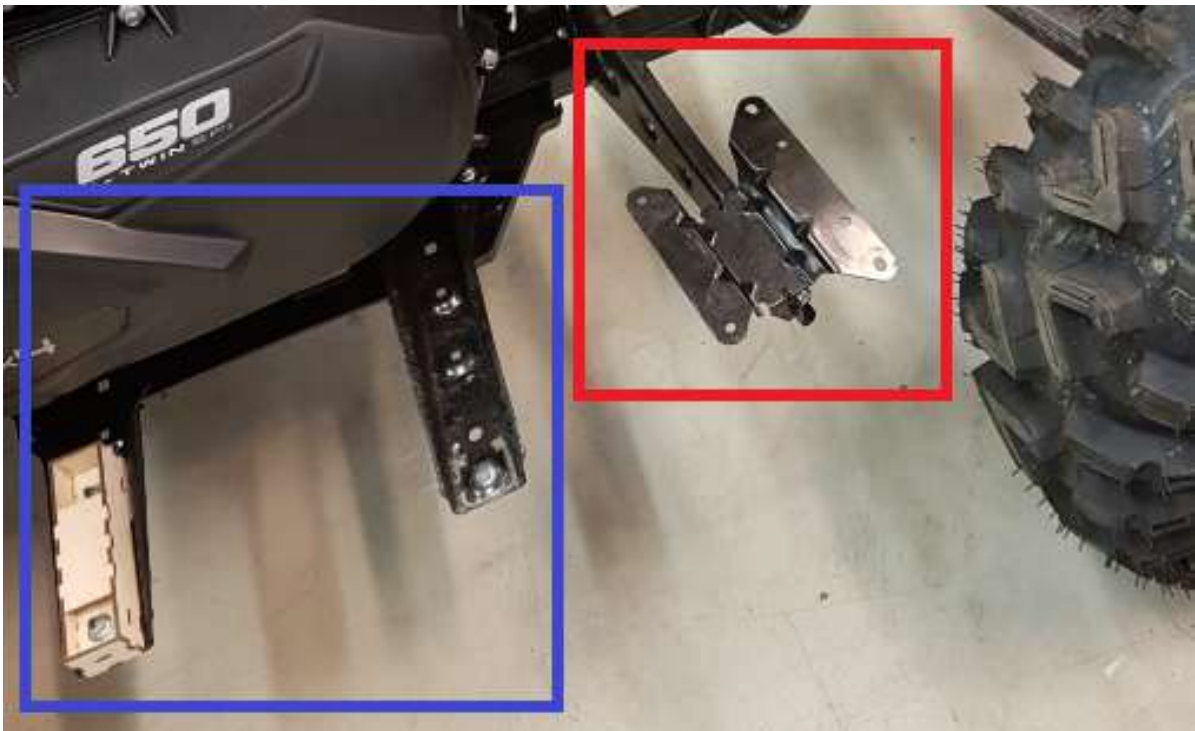
$$F_2 = \frac{36.8Nm}{0.025m} \quad (4)$$

$$F_2 = 1472N \quad (5)$$

8.1.2 Ramme

Et av designkravene er at ATV-en skal kunne tilbakestilles til original stand. Det vil ikke tillates å bruke andre festepunkter enn de som allerede eksisterer på ATV-en. Festepunktene som skal brukes, lokaliseres i forhold til hverandre. Rammen festes i disse.

Hovedfestepunktene til rammen vil være festepunktene til fothvilerne. Disse er festet til ATV-en og har et mønster bestående av fire hull på hver side. Disse er originalt festepunkter for fører-fothvilerne. Det finnes tilsvarende fothvilere for passasjer. (figur: 18)



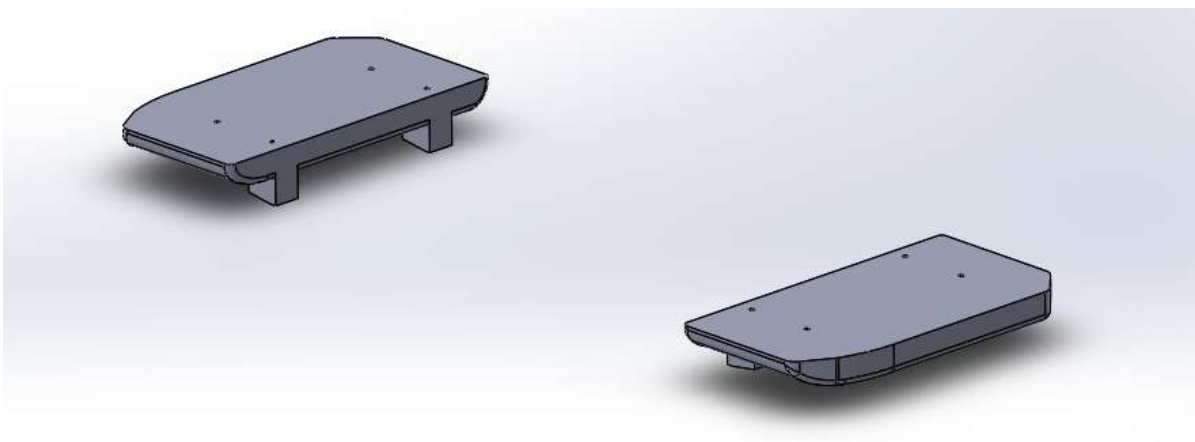
Figur 18: Festepunkter for fører-fothviler (blå) og festepunkter for passasjer-fothviler (rød)

Sekundærfestepunktene, som vil føre til oppstiving av rammen, er lokalisert under passa-

sjerstolen. Dette har samme hullmønster som fører-fothvilerne. Festepunktet vil fungere som en avstiver for rammen, den vil ta opp krefter som ellers ville ført til et stort moment på hovedfestepunktene lenger ned på ATV-en.

8.1.3 Lokalisering av festepunkter

For å lokalisere hovedfestepunktene, ble det designet en forenklet versjon av fører-fothvilerne (figur 19). Disse ga en nøyaktig måleprosess med skyvelæret. Lokasjonene ble testet ved å lage laserkuttete deler i kryssfiner (figur 20). Disse delene har to funksjoner. De skal bekrefte hullmønsteret til fører-fothvilerne og forenkle måling av hullmønsteret til passasjer-fothvilerne.



Figur 19: Forenklet modell av original fører-fothviler



Figur 20: Festepunkt for fører-fothviler med laserkuttete testklosser montert

Passasjer-fothvilerne står ikke i samme vinkel som fører-fothvilerne. Det ble derfor gjort en midlertidig beslutning om å ikke bruke disse festepunktene om det viser seg å være tilstrekkelig uten.



Figur 21: Laserkuttet del for lokalisering av sekundærfestepunktene.

Det samme problemet gjelder også sekundærfestepunktene. Disse står vinklet i forhold til de andre festepunktene, rammen og hverandre. Det ble designet en testdel i kryssfiner for å forenkle målingene frem til rammen. (figur: 21)

Måleutstyret er ikke avansert nok til å kunne gjøre gode nok målinger. Det vil være mer hensiktsmessig å designe en tilnærmet versjon av festepunktets overganger. Dette vil gi tilnærmet like resultater i FEM-analysene. Forskjellen mellom CAD-tegningen og den fysiske rammen vil være at innfestningene til ATV-en vil spesialtilpasses under produksjon.

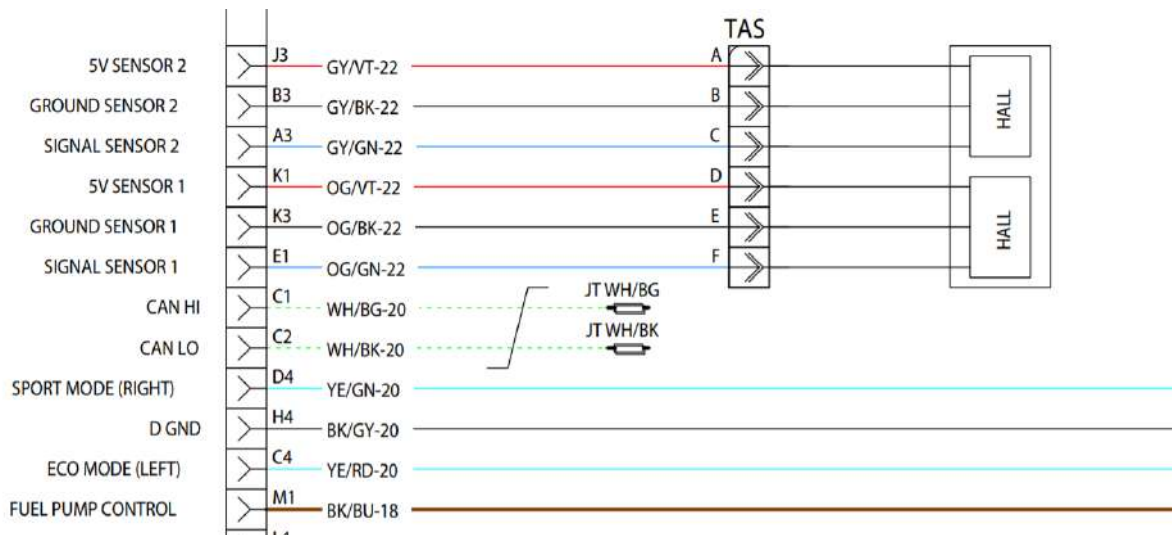
8.2 Elektromålinger

Her vil det bli foretatt diverse målinger på de ulike delsystemene; aktuatorer og komponenter.

8.2.1 Målinger av gassensorene

Det ble gjennomført målinger på sensorerne i ATV-en for gasspådraget. Det ble brukt en Arduino Mega 2560 og et EE-board med tilhørende software (Waveforms) for å få tallverdier.

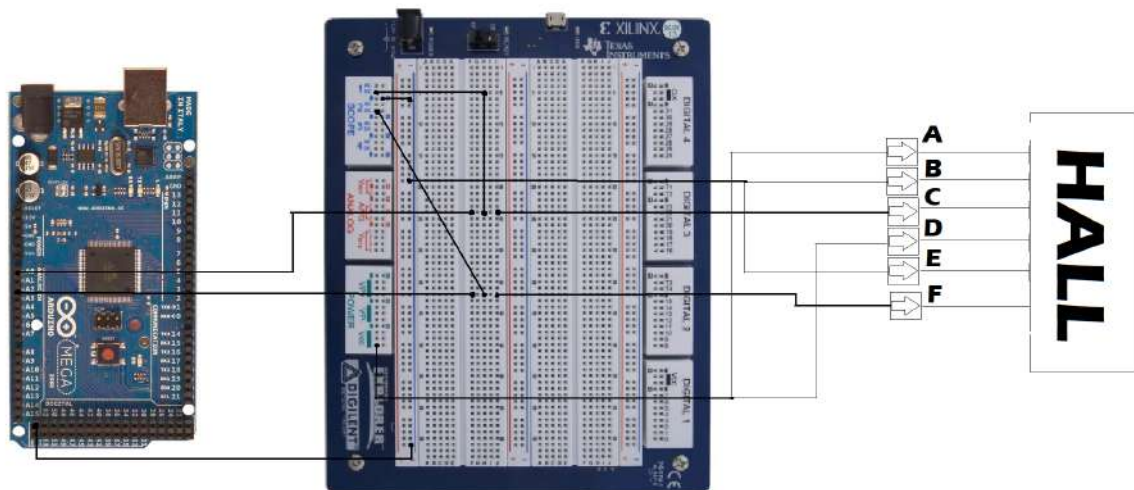
Figur 22 viser kretsskjemaet for sensorerne slik de er satt opp i databladet. (Vedlegg A.8)



Figur 22: Oversikt over gassensorer

Gasspådraget blir regulert av en DC spenningskilde med tilhørende sensorer som gir tilbakemelding til systemet. Gassen styres med en spenning på 5V, hvor 5V antas å gi maks pådrag til systemet.

Figur 23 viser hvordan sensorerne ble målt.

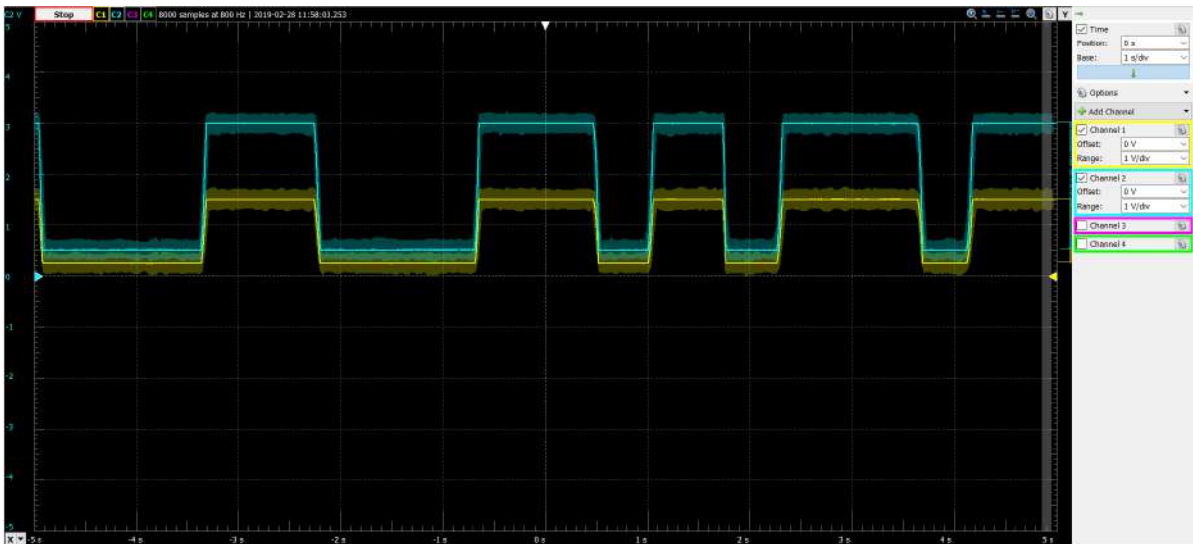


Figur 23: Oversikt over oppsettet for målingene

Tabell 17: Forklaring på tilkoblingspunktene

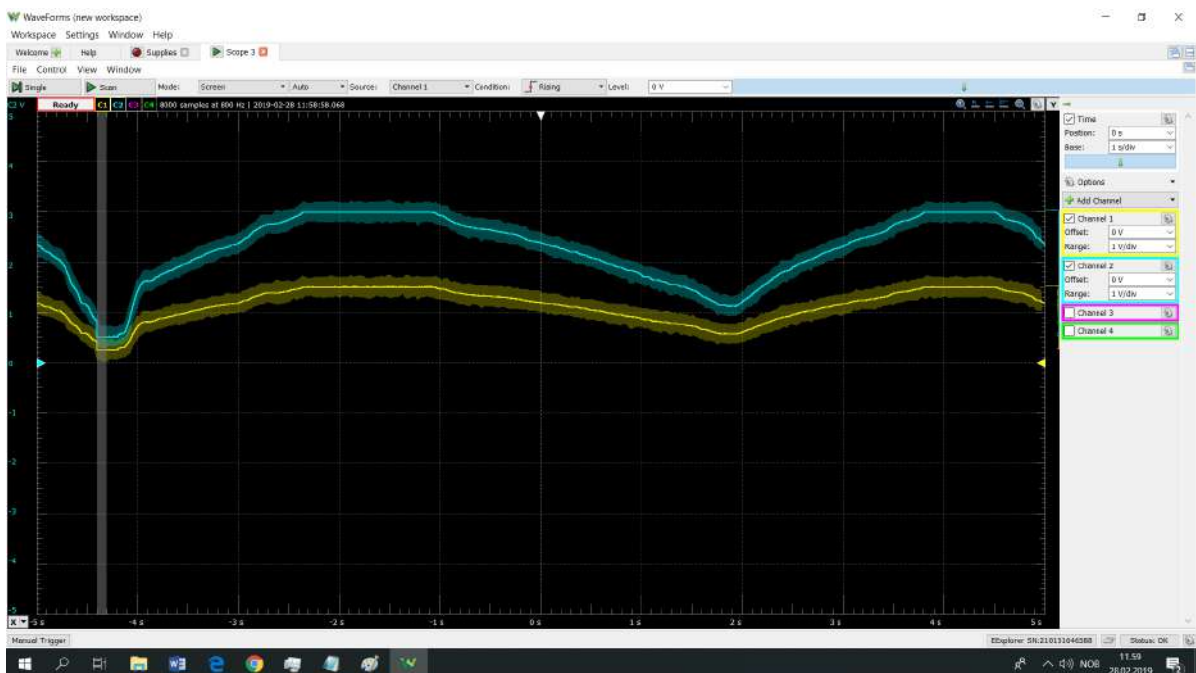
Bokstav for tilkoblingspunkt	Forklaring
A	5 V sensor 2
B	Ground sensor 2
C	Signal sensor 2
D	5 V sensor 1
E	Ground sensor 1
F	Signal sensor 1

Det ble brukt en spenning på 5 V inn til sensorene, som ble forsynt av EE-boardet. Det ble målt med scopene på brettet. Databladet for EE-board ligger i referanser. [10]



Figur 24: Måleresultatet fra måling med Ee board

Figur 25 viser målingen ved gradvis økning av gasspådraget.



Figur 25: Måleresultatet fra måling med Ee board

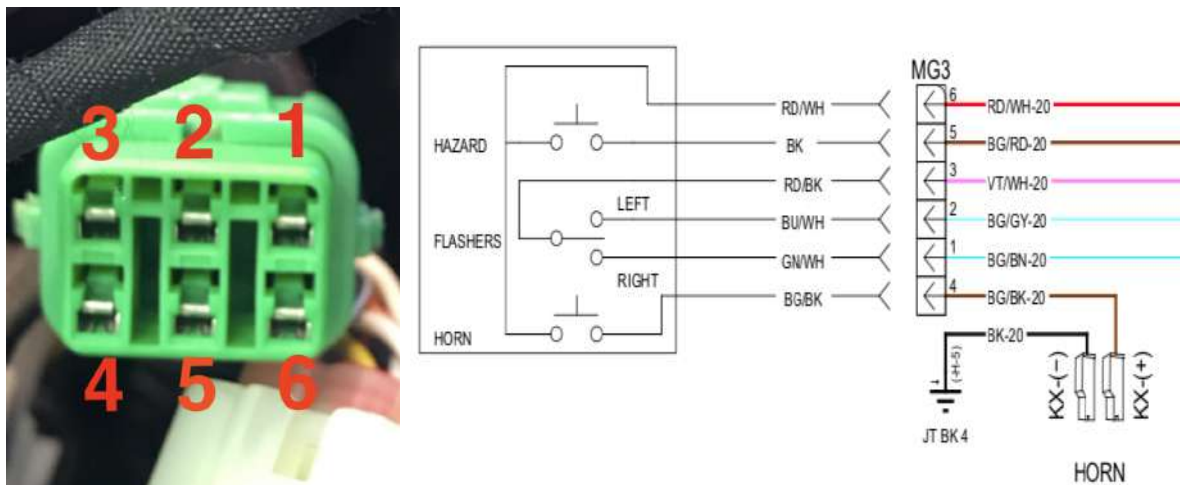
Det kan sees fra målingene at sensor 1 eller tilkoblingspunkt F ligger to ganger høyere enn det sensor 2 gjør. Ved null pådrag ligger sensorene på et fast punkt forskjellig fra null. Det vil si at sensorene (C og F) med en 5 V spenning i punkt A og D, aldri kan ligge på 0 V.

8.2.2 Målinger av instrumenter på styret

Det ble utført målinger på styret av ATV for å finne tilkoblingspunktene til de ulike bryterne.

Koblingspunkt 1 - MG3

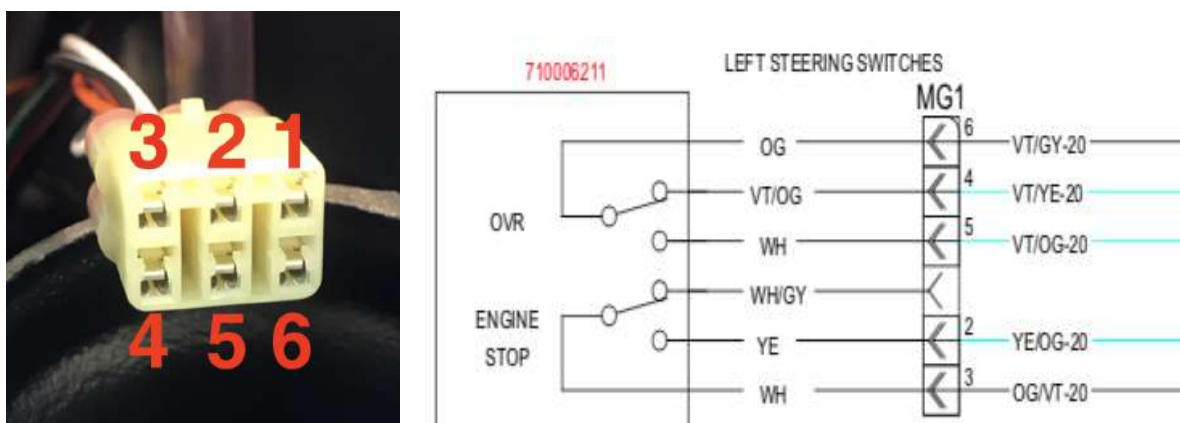
I koblingspunkt 1 er signalene til høyre- og venstre blinklys, signalhorn og nødblink. Kretsskjema MG3 og koblingspunkt 1 i figur 26 viser en oversikt over hvordan dette er koblet sammen.



Figur 26: Kretsskjema - koblingspunkt 1

Koblingspunkt 2 - MG1

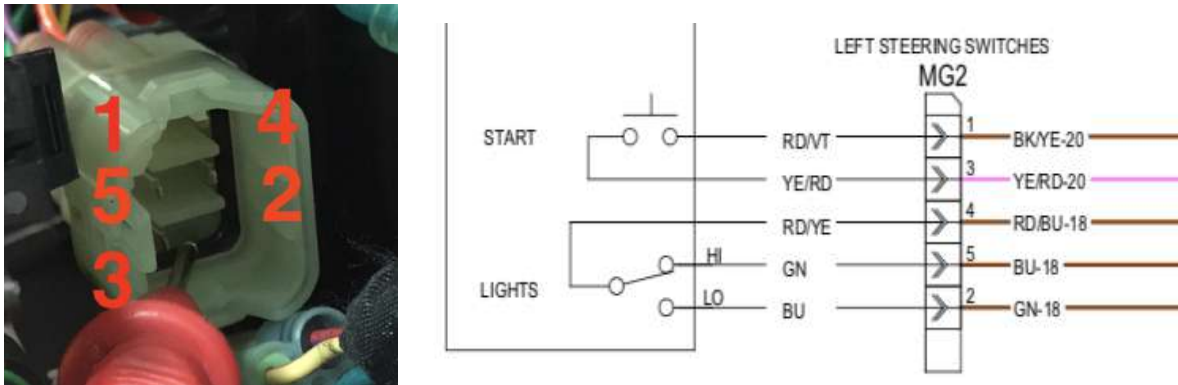
I koblingspunkt 2 er signalene til nødstop og DPS (kjøremodus). Kretsskjema MG1 og koblingspunkt 2 i figur 27 viser en oversikt over hvordan dette er koblet sammen.



Figur 27: Kretsskjema - koblingspunkt 2

Koblingspunkt 3 - MG2

I koblingspunkt 3 er signalene til startknappen, fjernlys og nærlys. Kretsskjema MG2 og koblingspunkt 3 i figur 28 viser en oversikt over hvordan dette er koblet sammen.



Figur 28: Kretsskjema - koblingspunkt 3

Tabell 18 viser en oppsummert oversikt over tilkoblingspunktene og deres funksjoner.

Tabell 18: Oversikt over tilkoblingspunkt på styret.

	Kob.pkt 1 - MG3	Farge ledning	Kob.pkt 2 - MG1	Farge ledning	Kob.pkt 3 - MG2	Farge ledning
1	Høyre blink	Grønn/Hvit	Nødstopp, strøm når ikke trykt inn	Hvit/Grå	Startknapp, koblet med #3	Rød/Lilla
2	Venstre blink	Blå/Hvit	Nødstopp, strøm når trykt inn	Gul	Nærlys	Blå
3	Begge blink	Rød/svart	Nødstopp, ground	Hvit	Starknapp, koblet med #1	Gul/Rød
4	Tut	Beige/Svart	DPS, strøm når ikke trykt inn	Lilla/Orange	Spenning til lys	Rød/Gul
5	Nødblink	Svart	DPS, strøm når trykt inn	Hvit	Fjernlys	Grønn
6	Signalhorn og nød- blink	Rød/Hvit	DPS, ground	Orange		

Konsept

Utarbeidelse av de ulike konseptene, samt Pugh-matriser for å sammenligne og for å sikte seg inn på et konsept for hvert av delsystemene.

Innhold:

- Overordnede system
- Kontrollsystem
- Delsystemkonsepter
- Pugh-matriser
- Løsninger til delsystemene



KONGSBERG



USN Universitetet
i Sørøst-Norge

Ut ifra målingene og krav som har blitt satt, kan det utvikles flere konsepter til hvert av delsystemene. Dette har skjedd gjennom en idemyldring i fellesskap hvor gruppen har kommet fram til flere ideer. Noen av disse har blitt videreutviklet til å bli konsepter.

Disse konseptene har blitt presentert til bedriften for å diskutere hvilke av disse som er mest fornuftig å gå videre med, samtidig som at bedriftsrepresentanter har kommet med egne innspill.

I denne seksjonen vil det greies ut om de ulike konseptene; hva de er, fordeler og ulemper og hvilke som er egnet for videreutvikling og implementering.

9.1 Delsystemkonsepter

Her presenteres de ulike konseptene tilknyttet delsystemene. I denne delen blir konseptene kort introdusert før de settes opp mot hverandre i en PUGH-matrise, hvor de viktigste funksjonene blir vektlagt. Det blir så tatt en helhetsvurdering for å komme frem til det beste konseptet.

9.1.1 Styringsenhet

Kjøretøyet skal kunne styres av en operatør over en trådløs forbindelse. Dette kan gjøres ved at styringsenheten kommuniserer med datamaskinen på kjøretøyet. Dette kan gjøres med 4G, WLAN eller radio. Gruppen har kommet opp med fire ulike konsepter for hvordan dette kan løses.

Konsept 1

Operatøren bruker en PC/laptop som kommuniserer med datamaskinen eller mikroprosessen på kjøretøyet.



Figur 29: Kjøretøyet styres av en PC [30]

Konsept 2

En spillkontroller kan brukes til å styre kjøretøyet. Dette kan gjøres via bluetooth på datamaskinen/mikroprosessen om bord på kjøretøyet.



Figur 30: Kjøretøyet styres av en håndkontroll for spillkonsoll [55]

Konsept 3

Figur 31: Kombinasjon av konsept 1 og 2

Konsept 3 er en kombinasjon av konsept 1 og 2, hvor man bruker en spillkontroll koblet opp mot en PC via bluetooth som videre sender signalet til datamaskinen eller mikroprosessen om bord på kjøretøyet.

Konsept 4

Konsept 4 går ut på å bruke en sender og en mottaker fra et radiostyrt kjøretøy og eventuelt modifisere kontrolleren slik at den egner seg til å styre ATV-en. Signalene sendt til mottakeren går igjennom en signalprosess før de sendes til Arduinoen hvor de blir tolket og videresendt.



Figur 32: Konsept 4 RC-kontroller [76]

9.1.2 Håndbrems

Konsept 1

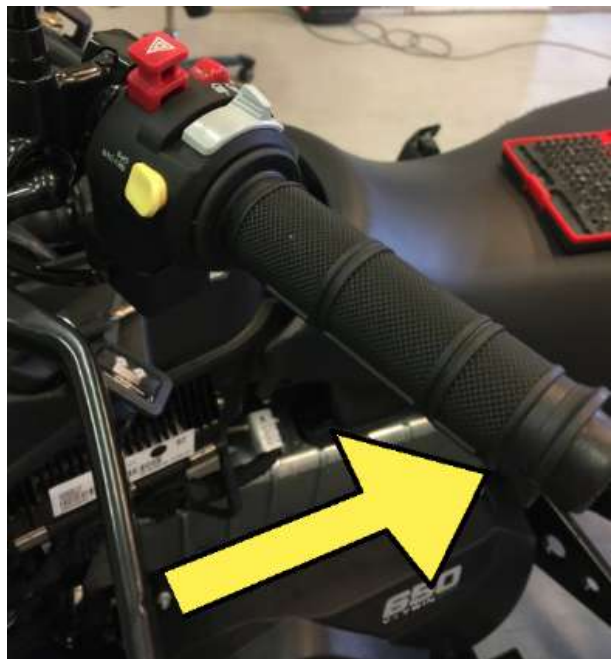
En lineæraktuator blir brukt for å trykke inn stempelet som regulerer trykket for brems. Ved målinger er det anslått at man trenger 1472 N for å trykke inn bremsen. Eventuelt mindre arm til aktuatoren for å ha mindre banelengde, slik at tregere aktuatorer kan brukes.



Figur 33: Konsept 1 for håndbremsen

Konsept 2

En lineæraktuator blir brukt for å trykke inn den eksisterende bremsehendelen. Målingen viste at 230 N var nødvendig for å trykke inn hendelen på det ytterste punktet når armen er 160 mm.



Figur 34: Konsept 2 for håndbremsen

Konsept 3

En hovedsylinder fra et kjøretøy blir brukt for bremsing.



Figur 35: Konsept 3 for håndbremsen [32]

9.1.3 Fotbrems**Konsept 1**

En servomotor som genererer et moment på 78,35 Nm kan erstatte armen som brukes til å bremse.

Konsept 2

En lineæraktuator kan brukes for å trykke inn den eksisterende fothendelen. På ytterpunktet på armen (250 mm), behøves en kraft på 315 N for å trykke hendelen helt inn. Noe som gir et moment på 78,35 Nm.



Figur 36: Konsept 2 for fotbrems

9.1.4 Gir

Konsept 1

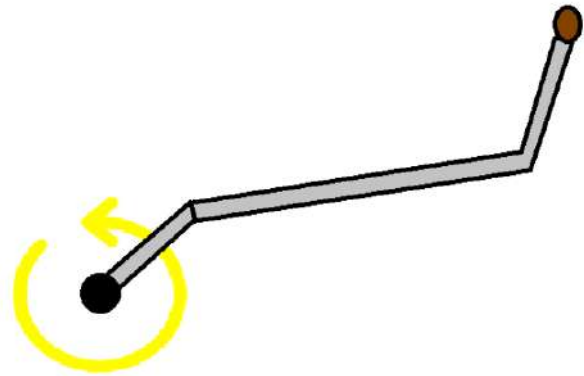
Dersom man fjerner plastikkdekslet på figur 37 kan girspaken bevege seg langs en akse. En lineæraktuator kan brukes til å bytte gir når ATV-en står stille. Det ble målt at 70 - 90 N var nødvendig for å bytte gir.



Figur 37: Konsept 1 for gir

Konsept 2

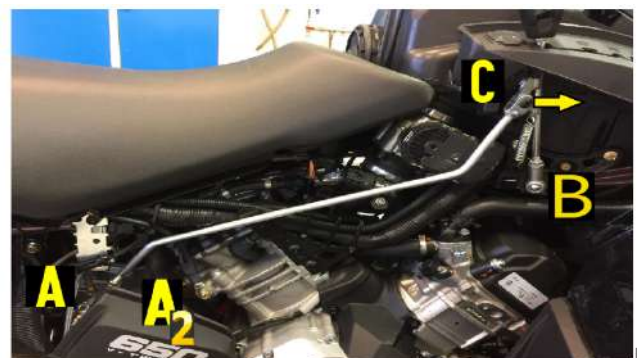
En servomotor kan brukes til å skifte gir ved å rotere om punktet A_2 til riktig posisjon. Antatt moment er 8,978 Nm som tilsvarer cirka 100 N lineær kraft oppe ved girspaken om man manuelt skal dra/dytte.



Figur 38: Konsept 2 for gir

Konsept 3

En lineæraktuator kan brukes til å dytte på stangen mellom punktet A til A_2 på figur 39. Festepunktet mellom stangen og aktuatoren befinner seg i punkt A. For å skifte gir i denne posisjonen er det antatt at det trengs 163 N med en arm på 55 mm.



Figur 39: Konsept 3 for gir

9.1.5 Gass

Konsept 1

Den eksisterende gasshendelen kan brukes sammen med en lineærak-tuator og lage et styringssystem basert på dette.



Figur 40: Konsept 1 for gass [64]

Konsept 2

Koble til ledningsnett for den eksisterende el-gassen.



Figur 41: Konsept 2 for gass [64]

9.1.6 Styring

Konsept 1

En aktuator brukes for å presse på styret med en armlengde på 41 cm. Det kreves en kraft på 175 N på det ytterste punktet. Denne målingen ble tatt når ATV-en sto stille uten ekstra motstand mot dekkene.



Figur 42: Konsept 1 for styret

Konsept 2

En servomotor brukes for å rotere styreakselen ved basen av rattet. Et minimum moment på 71,75 Nm er nødvendig.



Figur 43: Konsept 2 for styret

9.1.7 Nødstopp

I tillegg til konseptforslagene for nødstopp er det fremmet et krav fra oppdragsgiver om en kablet nødstopp. En person skal kunne stoppe ATV-en hvis det trådløse alternativet feiler. Knappen skal være konfigurert slik at hvis man slipper knappen, stopper ATV-en.

Konsept 1

Koble seg på nødstoppskretsen til ATV-en og bruke sender og mottaker for å sende signaler trådløst for å stoppe ATV-en.



Figur 44: Konsept 1 for nødstopp

Konsept 2

En aktuator brukes for å trykke på knappen som allerede er på ATV-en.



Figur 45: Konsept 2 for nødstop

9.1.8 Batteri**Konsept 1**

Et 24 V batteri brukes til å drifte aktuatorene i delsystemene.



Figur 46: Konsept 1 for batteri [6]

Konsept 2

Det eksisterende 12 V batteriet i ATV-en brukes for å drifte systemet.



Figur 47: Konsept 2 for batteri

9.1.9 Våpenstasjonsfeste

Konsept 1

Lage et feste til våpenstasjonen som kan sikre stabil festing og posisjonere våpenstasjonen langs en horisontal akse. Dette gjøres for å flytte felles massepunkt for å kompensere for helninger i terrenget.



Figur 48: Konsept 1 for våpenstasjonsfeste

Konsept 2

Lage et statisk stativ for ATV-en.



Figur 49: Skisse av konsept 2 for våpenstasjonsfeste

9.2 Pugh-matriser for delsystemene

For å velge det konseptet som passer systemet best, kan det være behjelpelig å bruke et sammenligningsverktøy. For å sammenligne de ulike konseptene med grunnlag i komparative kriterier, utarbeidet gruppen en oversikt. Pugh-matriser er et typisk vurderingsverktøy hvor kriterier rangeres fra '-1' til '+1'. Gruppen har blitt enig i at dette vil gi en altfor vag framstilling, da noen vil være langt mer viktig enn andre. Det har blitt valgt å sette en rangering fra 1 til 5 på hvert enkelt vurderingskriterie.

Her sammenlignes konseptene med utgangspunkt i kriterier som gruppen mener er viktige for systemet. Disse vil utgjøre en endelig poengsum som vil være til stor hjelp for å stadfeste hvilket konsept som er best.

Følgende kriterier er nødvendig for delsystemene:

- Funksjonalitet
- Sikkerhet
- Kostnad
- Tid
- Pålitelighet
- Montering
- Kompleksitet
- Testbarhet.

Disse kriteriene blir tatt i betraktning når valget på endelig konsept faller.

Med vurderingskriteriene menes:

Funksjonalitet: Funksjonalitet beskriver hvor godt prosessen jobber mot sitt mål; Om løsningen jobber mer eller mindre effektivt ift andre konsepter. Hvordan møter konseptets funksjonalitet alle kravene som er blitt satt opp til systemet? Vil funksjonaliteten påvirke andre faktorer ved systemet?

Sikkerhet: Med sikkerhet menes hvor sikkert konseptet er for omgivelsene og hvor sikkert det er ift. skade på eksisterende system - hvor godt det kan implementeres i henhold til alle sikkerhetskravene.

Kostnad: Hvor dyrt er innkjøp av deler og hvor dyr er implementering av løsningen? Vil det påløpe ekstra kostander i løpet av utvikling og implementering? Er løsningen levedyktig - vil det påløpe ekstra kostander i vedlikehold?

Tid: Hvor lang tid vil det ta å planlegge, utvikle og implementere konseptet? Vil konseptet være for tidkrevende og påvirke i den grad av at det vil gå på bekostning av tid disponert til andre konsepter?

Pålitelighet: Er konseptet pålitelig nok? Vil løsningen prestere godt nok og virke plikt-oppfyllende i forhold til kravene?

Montering: Hvor lang er den estimerte monteringstiden? Hvordan vil dette påvirke kravet om at ATV-en skal kunne tilbakestilles? Er løsningen plassbesparende?

Kompleksitet: Hvordan vil graden av kompleksitet øke sjansene for feil i bakgrunns-teorien? I hvilken grad vil dette øke med tanke på monteringen og implementeringen? Vil kompleksiteten og sjansene for feil gå utover tidsbruken?

Testbarhet: Hvor enkelt er det å teste konseptet? Lar det seg gjøre på en enkel og velfungerende måte?

Eksempel på poengsum:

Funksjonalitet med vektning $(5) \times \text{Konsept 1 (4)} = 20$

Funksjonalitet med vektning $(5) \times \text{Konsept 2 (3)} = 15$

I noen tilfeller hvor det er lite som skiller konseptene, kan det være aktuelt å gå for det konseptet med lavest samlet poengsum. Dette kan skyldes rådføring fra veiledere og intuisjon.

9.2.1 Håndbrems

Tabell 19: Pugh-matrise for håndbrems

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		Lineær	Lineær
Tilkobling		Kraftoverføring direkte på stempel	Kraft*Arm
Kraftoverføring		1472 N	230N v/160mm arm
Kommentar		En løsning som vil være mindre plasskrevende	Mindre kraftoverføring
Funksjonalitet	5	4	3
Sikkerhet	5	2	4
Kostnad	2	2	4
Tid	5	4	3
Pålitelighet	4	2	3
Montering	3	4	3
Kompleksitet	1	3	3
Testbarhet	4	3	3
Total Sum		89	94

Konsept 2, håndbrems med arm skal videreutvikles. Den scorer veldig bra på vurderingskriteriene samtidig som gruppen og bedriften er enige om at dette er en fornuftig løsning.

Det vil bli nødvendig å designe en arm som er kompatibel med aktuatoren. Denne skal brukes til å trykke inn bremsestempelen. Ettersom det er mindre krefter i dette konseptet ansees det som både tryggere, billigere og mer pålitelig. Det vil heller ikke ta mye lengre tid å designe enn konsept 1.

9.2.2 Fotbrems

Tabell 20: Pugh-matrise fotbrems

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		Lineær	Lineær
Tilkobling		Kraftoverføring direkte på stempel	Kraft * Arm
Kraftoverføring		Foreløpig ukjent kraft	315 N v/250 mm arm
Kommentar		En løsning som vil være mindre plasskrevende, men som vil kreve mer kraft	Lineæraktuator med elektrisk pådrag
Funksjonalitet	5	4	3
Sikkerhet	5	2	4
Kostnad	2	2	4
Tid	5	4	3
Pålitelighet	4	2	3
Montering	3	4	3
Kompleksitet	1	3	3
Testbarhet	4	3	3
Poengsum	—	89	94

Konsept 2, lineær aktuator med arm skal videreutvikles. Det skal være lett å regulere og tilpasse bremsekraften. Å bruke en lineær aktuator på begge bremsene vil være tidsbesparende i forhold til utvikling og implementering.

9.2.3 Gir

Tabell 21: Pugh-matrise gir

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator Tilkobling Kraftoverføring Kommentar		Lineær	Lineær
		Bruke girspaken uten deksel	Servomotor i punkt A2
		100 N	8,978 Nm
		Fjerne plastikkdekslet og la girspaken bevege seg langs en akse	Bruke en servomotor til å rotere girsystemet til riktig posisjon ved basen av stanga inn til girkassa.
Funksjonalitet	5	4	4
Sikkerhet	5	4	3
Kostnad	2	3	2
Tid	5	4	3
Pålitelighet	4	4	2
Montering	3	4	2
Kompleksitet	1	5	3
Testbarhet	4	4	1
Poengsum	—	115	75

Tabell 22: Pugh-matrise gir

Kategori	Vekting	Konsept 3	
Aktuator Tilkobling Kraftoverføring Kommentar		Servo	
		Aktuator på armen om punktet A	
		163 N med 55mm arm	
		Må demontere enda mer av karosseriet	
Funksjonalitet	5	4	
Sikkerhet	5	3	
Kostnad	2	3	
Tid	5	3	
Pålitelighet	4	3	
Montering	3	2	
Kompleksitet	1	4	
Testbarhet	4	2	
Poengsum	—	86	

Konsept 1, en lineær aktuator oppe ved girspaken skal videreutvikles. Dette systemet kan monteres ved rammen til våpenfestet. Konseptet ligner på løsningene som brukes ved bremsene.

Det må lages en ny plastikkramme som indikerer girposisjonen for å se at man er i riktig gir.

9.2.4 Gass

Tabell 23: Pugh-matrise gass

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		Lineær	Elektrisk pådrag
Tilkobling		Gasshendelen	Koble direkte til ledningsnettet
Kraftoverføring			—
Kommentar		Bruke den eksisterende løsningen og påføre trykk for å regulere pådraget	Koble seg til det elektriske nettverket og bruke spenning for å regulere gasspådraget
Funksjonalitet	5	3	5
Sikkerhet	5	3	4
Kostnad	2	2	5
Tid	5	3	4
Pålitelighet	4	3	5
Montering	3	3	5
Kompleksitet	1	3	3
Testbarhet	4	2	4
Poengsum	—	81	129

Konsept 2 har blitt ansett som best fra alle parter. Dersom det blir mulig å koble seg til ledningsnettet og sende samme signal til gassen, vil dette være den beste løsningen.

9.2.5 Styring

Tabell 24: Pugh-matrise styring

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		Lineær	Servo
Tilkobling		Kraft*Arm	Rotasjonskraft
Kraftoverføring		175 N v/41cm arm (71.75 Nm)	71.75 Nm
Kommentar		Kraften som trengs for å vri på hele rattet uten at det er ekstra motstand (stein, gjørme o.l)	Kraften som trengs for å vri på hele rattet uten at det er ekstra motstand (stein, gjørme o.l)
Funksjonalitet	5	4	4
Sikkerhet	5	4	4
Kostnad	2	3	3
Tid	5	3	3
Pålitelighet	4	3	3
Montering	3	4	3
Kompleksitet	1	3	2
Testbarhet	4	3	4
Plassbehov	2	4	2
Poengsum	—	108	104

Konsept 2, servomotor montert på styreakselen skal videreutvikles. Dette innebærer at det finnes en servomotor som er sterk og rask nok.

Det vil være mulig å regulere vinkelutslaget og vinkelhastighet ved behov. Løsningen er lett å teste, da man kan se servomotoren og hele styreakselen.

Det ble også foreslått fra KDA å se på et system med kjede som kan styres med en billigere aktuator.

9.2.6 Nødstop

Tabell 25: Pugh-matrise nødstop

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		Elektrisk pådrag	Lineær
Tilkobling		Bryter på det elektriske nettverket	Trykke inn nødstopknapp
Kraftoverføring		—	—
Kommentar		Koble seg på nødstopkretsen, designe et system som sender signaler for å stoppe ATV-en dersom kravene for sikker drift ikke oppfylles.	Bruke en aktuator for og fysisk trykke inn den opprinnelige nødstoppen på ATV-en.
Funksjonalitet	5	5	2
Sikkerhet	5	5	2
Kostnad	2	5	1
Tid	5	5	2
Pålitelighet	4	5	2
Montering	3	5	2
Kompleksitet	1	5	2
Testbarhet	4	5	2
Poengsum	—	145	56

Konsept 1 skal videreutvikles. Sikkerheten og funksjonaliteten til systemet var det viktigste i vektleggingen av konseptforslagene.

9.2.7 Batteri

Fra veiledermøte og gjennomgang av konseptene (28.02.19) ble det fastsatt at systemet skulle ha ett ekstra 24 V batteri. Ved revidering (07.03.19) ble det fastsatt at 12 V batteriet på ATV skulle brukes til å drifte eksisterende system og aktuatorer. Et eksternt batteri skal kjøpes inn ved behov.

9.2.8 Våpenstasjonsfeste

Tabell 26: Pugh-matrise våpenstasjonsfeste

Kategori	Vekting	Konsept 1	Konsept 2
Aktuator		—	—
Tilkobling		Feste til våpenstasjon	—
Kraftoverføring		—	—
Kommentar		Lage et feste til våpenstasjonen som i tillegg til å sikre stabilt og sikkert feste kan posisjonere våpenstasjonen langs en horisontal akse for å flytte sentermassepunktet til ATV-en. For å kompensere for ujevnt terrenget.	Lage et statisk stativ.
Funksjonalitet	5	2	3
Sikkerhet	5	2	4
Kostnad	2	1	5
Tid	5	1	4
Pålitelighet	4	3	5
Montering	3	3	4
Kompleksitet	1	2	4
Testbarhet	4	2	4
Poengsum	—	58	117

Konsept 2, statisk stativ skal videreutvikles. Kravprioritet og tidsbruk ble vektlagt høyest for avgjørelsen. Ved å få på plass et stativ tidlig i prosjektet kan det også brukes til å feste delsystemer lettere.

9.3 Løsninger til delsystemene

Løsninger til delsystemene er basert ut ifra målinger, beregninger og diskusjoner. Når det gjelder valg av aktuatorer, er det viktig at hurtighet og kraft tas i betraktning.

9.3.1 Gass

Ut ifra målingene som ble gjort av gass på ATV, kom det frem at sensorutgangene lå mellom 0 og 5V. Det vil derfor være mulig å emulere rett fra en Arduino samt en krets for å gjøre om signalene til analoge verdier.

9.3.2 Håndbrems

Konsept 2. Det skal designes en arm som generer moment for å tilføre kraft på bremsestempet.

9.3.3 Fotbrems

Konsept 2. Det skal designes en arm som generer moment for å tilføre kraft på bremsestempet.

9.3.4 Gir

Konsept 1. Det skal designes en komponent som gjør at en lineæraktuator kan bevege girspaken. Et godt festepunkt er hengselleddet punkt C (figur 50)



Figur 50: En potensiell tilkobling med denne løsningen er å koble til aktuatoren til punkt C

9.3.5 Styre

Konsept 2, servomotor montert på styreakselen.

9.3.6 Nødstop

Det skal designes en fjernstyrt og en kablet nødstop som sitter på ATV. Den kablede nødstoppen må kobles slik at en knapp holdes inne så lenge ATV er i drift og stopper dersom knappen slippes. Det vil her bli brukt en Arduino som festes på ATV som signalet blir sendt til.

9.3.7 Batteri

Det er bestemt at alt skal kjøres på 12 V batteri. Til å begynne med kjører ATV kun på ett 12 V batteri, men på sikt vil det være 2×12 V batteri.

9.3.8 Ramme

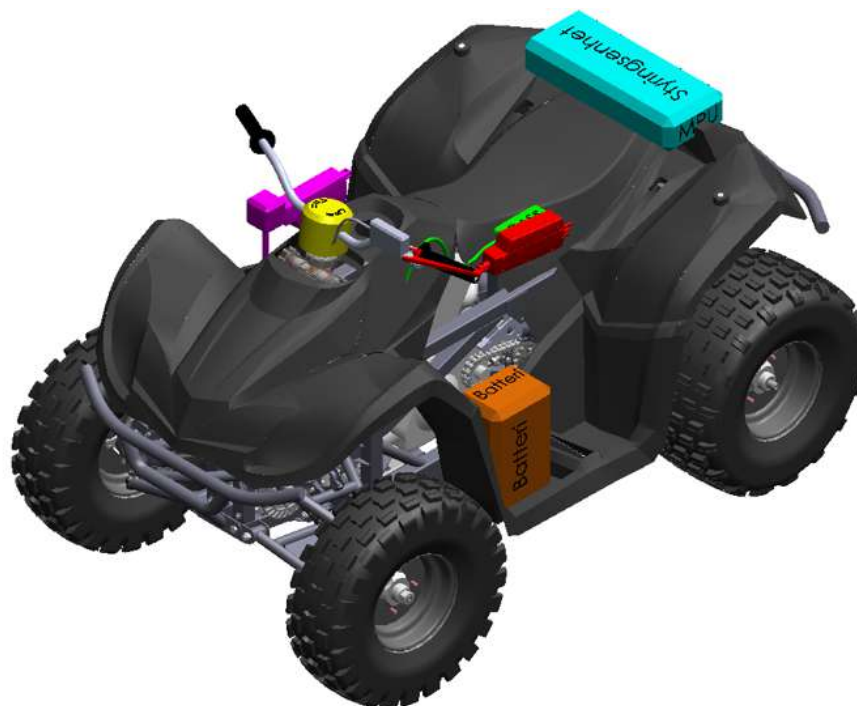
Det skal designes en ramme for montering av våpenstasjonen. Dette vil være en statisk ramme med flere innfestningspunkter til ATV-en. Rammen vil også fungere som innfestning til de andre delsystemene som skal monteres. Da blir montering og demontering av alle delsystemene raskere. Rammen vil designes slik at gis plass og mulighet for videreutvikling og montering av andre komponenter i senere tid, som større batteri, sensorer og styreenheter.

9.3.9 Fjernstyring

Det skal brukes bluetooth eller RF sender og mottaker mellom to arduinoer for å styre ATV-en, samt vise sensordata. Den ene Arduinoen skal sitte på ATV, mens den andre skal være tilkoblet en datamaskin. Dette skal gjøre det kjapt å sende over alt av styring og sensordata trådløst. På datamaskinen tenkes det å koble til en ekstern spillkontroller, samt vise et grensesnitt som inneholder informasjon som kjøreren trenger.

9.4 Overordnede systemet

det har blitt laget en CAD-modell for å illustrere det foreløpige designet (ppd. 21.03.19) uten ramme og våpenstasjon .



Figur 51: Foran fra siden, ATV modellen er hentet fra brukeren BRESSON Roch [5]



Figur 52: Foran fra siden, ATV modellen er hentet fra brukeren BRESSON Roch [5]

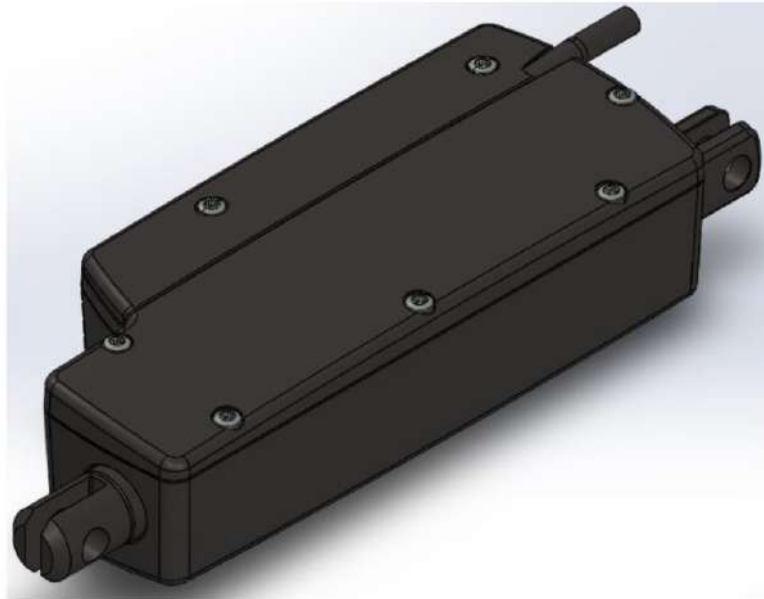


Figur 53: Bak fra siden, ATV modellen er hentet fra brukeren BRESSON Roch [5]



Figur 54: Bak fra siden, ATV modellen er hentet fra brukeren BRESSON Roch [5]

En CAD-modell av LA12 fra Linak har blitt brukt i modellen ovenfor. Dette viser hvordan noen av aktuatorene kan plasseres på ATV-en.



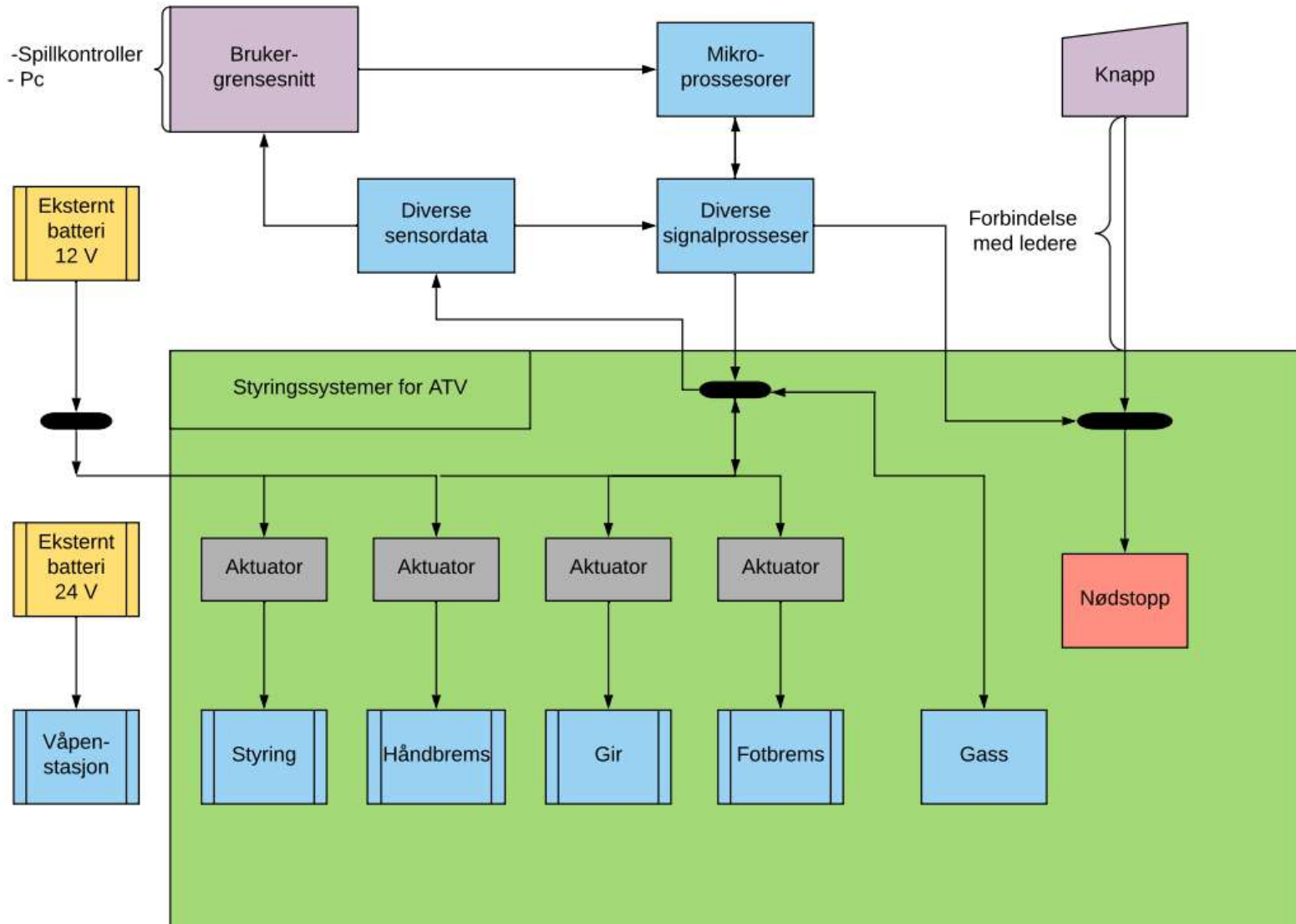
Figur 55: LA12 aktuator CAD-modell - Fra Linak [58]

9.5 Kontrollsystem

For å kontrollere alle delsystemene vil det bli brukt mikroprosessorer. Siden gruppemedlemene har best kjennskap til Arduino og C++ vil dette bli brukt.

For å kontrollere systemet brukes en spillkontroller via en arduino, som gir kommandoer til ATV.

En overordnet oversikt over konseptet kan sees i figur 56. Dette er satt opp i Lucidchart [41].



Figur 56: UML flowchart av systemets funksjoner

Research

Her, i dette kapitlet blir det gjennomgått hva slags research som er blitt gjort i forhold til oppgaven.

Innhold:

- PID-kontroller
- Fjernstyrt Corvette
- PWM til DC
- ROS - Robot Operating System
- Fartsmåling med photodiode
- ModBus
- Transistor
- Maskinkomponenter



KONGSBERG



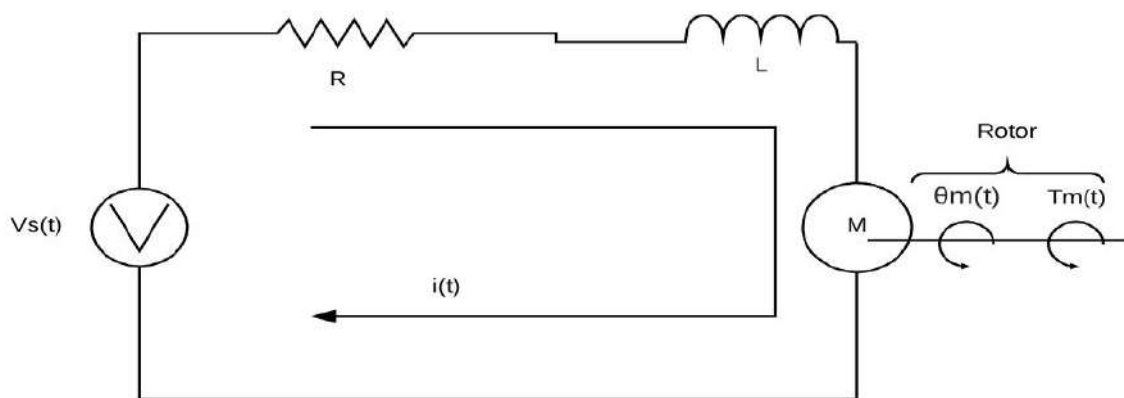
USN Universitetet
i Sørøst-Norge

10.1 PID kontroller

10.1.1 Iterasjon 1

Det er tenkt å implementere en PID kontroller til Arduinoen, slik at responsen skal bli som ønsket. For å kunne modellere og simulere lineæraktuatoren med en PID-kontroller i simulink, må det utarbeides en matematisk modell for det elektriske og mekaniske systemet[78]. Derfra kan det utledes en transferfunksjon som beskriver forholdet mellom referansespenningen og utgangen til systemet.

Først tegner man opp den elektriske kretsen for armaturen til motoren (Figur 57 - satt opp i Lucidchart [41]).



Figur 57: Elektrisk krets for motorarmaturen

Deretter finner man et uttrykk for inngangsspenningen V_t . Kirschoff's spenningslov sier at summen av alle spenningsfall i kretsen skal tilsvare bidraget fra spenningskilden. Siden den består av en krets uten forgreninger, kan man bruke sløyfeanalyse ved hjelp av Ohm's lov for å finne spenningsfallene over hver enkelt komponent.

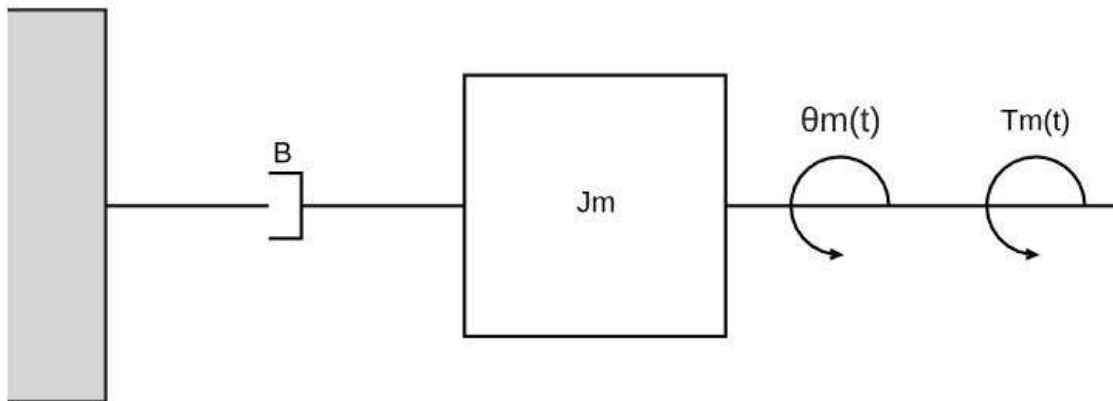
$$V_s(t) = i(t)R + L \frac{\delta i(t)}{\delta t} + V_B(t) \quad (6)$$

$i(t)R$ = Spenningsfall over resistor

$L \frac{\delta i(t)}{\delta t}$ = Spenning over induktans(spole).

$V_B(t)$ = Spenningsfall over motor = Elmotorisk kraft; K_e = fem konstant $V_B(t) = K_e \frac{\delta \theta_m(t)}{\delta t}$

Nå setter man opp den mekaniske modellen (Lucidchart [41]).



Figur 58: Mekanisk modell

Torque til motor kan beskrives ved følgende likninger:

$$T_m(t) = K_t i(t) \quad , \quad K_t = \text{Momentkonstant} \quad (7)$$

$$\text{Vinkelhastighet } w(t) = \frac{\delta\theta_m(t)}{\delta t}$$

$$T_m(t) = J_m \frac{\delta^2\theta_m(t)}{\delta t} + B \frac{\delta\theta_m(t)}{\delta t} \quad (8)$$

Setter sammen likning 7 og 8:

$$J_m \frac{\delta^2\theta_m(t)}{\delta t} + B \frac{\delta\theta_m(t)}{\delta t} = K_t i(t) \quad (9)$$

Forenkler likning 6 og 9 for utseendes skyld:

$$V_s(t) = i(t)R + L\dot{i}(t) + K_e\dot{\theta}(t) \quad (10)$$

$$J_m\ddot{\theta}(t) + B\dot{\theta}(t) = K_t i(t) \quad (11)$$

Tar deretter Laplace-transformen på likning 10 og 11:

$$\mathcal{L}(V_s(t) = i(t)R + L\dot{i}(t) + V_B(t)) = V(s) = RI(s) + sLI(s) + sK_e\Theta_m(s) \quad (12)$$

$$\mathcal{L}(J_m\ddot{\theta}(t) + B\dot{\theta}(t) = K_t i(t)) = s^2J_m\Theta_m(s) + sB\Theta_m(s) = K_t I(s) \quad (13)$$

Finner et uttrykk for armaturstrømmen $I(s)$ fra likning 12 og setter inn for i likning 13.

$$I(s) = \frac{V(s) - sK_m\Theta_m(s)}{R + Ls} \quad (14)$$

$$s^2J_m\Theta_m(s) + sB\Theta_m(s) = K_t\left(\frac{V(s) - sK_e\Theta_m(s)}{R + Ls}\right) \quad (15)$$

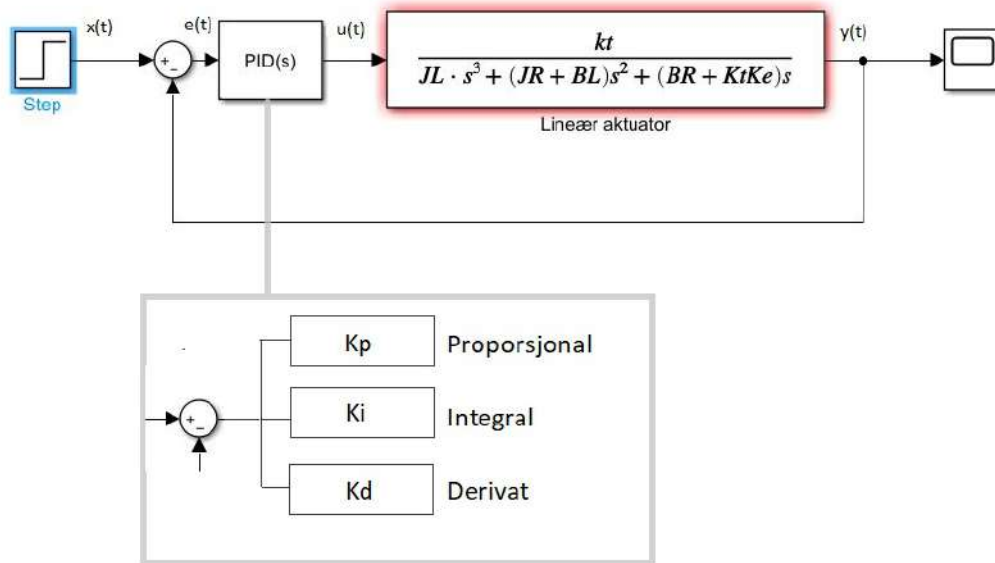
Finner forholdet mellom utgang og inngang; $\frac{\Theta_m(s)}{V(s)}$:

$$\frac{\Theta_m(s)}{V(s)} = \frac{K_t}{JLs^3 + (JR + BL)s^2 + (BR + K_tK_e)s} \quad (16)$$

Etter revidering med maskiningeniør, viser det seg å være vanskelig å finne nøyaktige innsatte verdier for transferfunksjonen da disse vil variere med lasten. Det vil derfor bli sett på andre tilnærminger for å finne en controller.

10.1.2 Iterasjon 2

For å redusere error og oppnå ønsket verdi på utgangssignalet brukes en PID-kontroller med tilbakekobling. En PID-kontroller holder utgangen slik at erroren mellom ønsket verdi og tilbakekoblingen i en closed-loop blir null. På grunn av tilbakekoblingen kan den dermed levere en kontrollutgang på ønsket nivå. PID står for Propotional-Integral-Derivative,- disse kontrollerene blir kombinert slik at det produseres et kontrollsignal. Figur 59 er satt opp i Simulink og viser kretsen med PID-kontroller og lineæraktuator.



Figur 59: Krets med PID-kontroller og aktuator

Nedenfor forklares virkemåten til hver enkelt kontroller [28].

10.1.3 P-kontroller

En P-kontroller gir en utgang som er proporsjonal med strømfeilen. Den sammenligner den ønskede inngangsverdien med faktisk verdi fra tilbakekoblingen. Feilen blir multiplisert med en proporsjonal konstant, K_p , som gir utgangsverdien slik at man får $K_p e(t)$. Ved å øke proporsjonalkonstanten K_p økes også farten på responsen.

10.1.4 I-kontroller

En I-kontroller gjør den nødvendige handlingen for å kunne eliminere steady-state erroren. Den integrerer feilen over en tidsperiode slik at feilverdien til slutt blir null og man har $K_i \int_0^t e(t) dt$. Dersom det oppstår en negativ feil vil utgangen reduseres og stabiliteten til systemet vil bli påvirket, samt at farten på responsen begrenses. For å øke farten på responsen kan man minske integralforsterkningen K_i .

10.1.5 D-kontroller

En D-kontroller forutser fremtidig oppførsel på feil. Med hensyn til tiden, avhenger utgangsverdien av forandringshastigheten multiplisert med derivasjonskonstanten K_d , slik at man får $K_d \frac{de(t)}{dt}$. På denne måten oppnår man en kick-start på utgangen og system-

responsen økes.

Disse verdiene summeres før utgangssignalet videresendes til aktuatoren.

$$u(t) = k_d e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (17)$$

10.1.6 Anti-Windup

Dersom man bruker integral-kontroller, vil det også være nødvendig å bruke anti-windup [79]. Dette er fordi tilbakekoblingen iblant vil nå høye verdier, og kontrollere vil bruke lang tid på å komme ned til verdier hvor det kan kontrolleres. Ved å bruke en antiwindup vil det si at dersom feilmarginen når over en viss maksverdi, vil den settes tilbake til maksverdien.

10.1.7 Tuning metoder - PID-kontroller

For å sette riktig verdier på K_p , K_i og K_d er det flere ulike metoder man kan ta i bruk. Noen av tuning metodene som har vært aktuelle i prosessen er [28]:

- En metode hvor man prøver og feilsøker. Her tar man utgangspunkt i oscilleringen kontrollere gir når den er i gang for å bestemme verdiene på K_p før man så justerer K_i og K_d .
- En teknikk som tar for seg responskurven slik at man kan regne de ulike verdiene som skal inn i P, I og D-kontrollere ved å bruke vekstfart, død-tid og stigningstid.
- Zeigler-Nichols metoden - En oscillerende metode hvor man setter K_p til en bestemt verdi og K_i og K_d settes lik 0. Man øker verdien på K_p helt til det oppnås en konstant amplitude på oscilleringen. Amplituden kalles ultimate gain, K_u , og perioden blir kalt ultimate period, P_u . Videre kan man da bestemme de ulike verdiene til K_p , K_i og K_d ved hjelp av disse (se tabell 27) [69].

Tabell 27: Zeigler-Nichols tabell

	K_p	K_i	K_d
P	$\frac{K_u}{2}$		
PI	$\frac{K_u}{2.2}$	$\frac{P_u}{1.2}$	
PID	$\frac{K_u}{1.7}$	$\frac{P_u}{2}$	$\frac{P_u}{8}$

10.2 Fjernstyrt Corvette

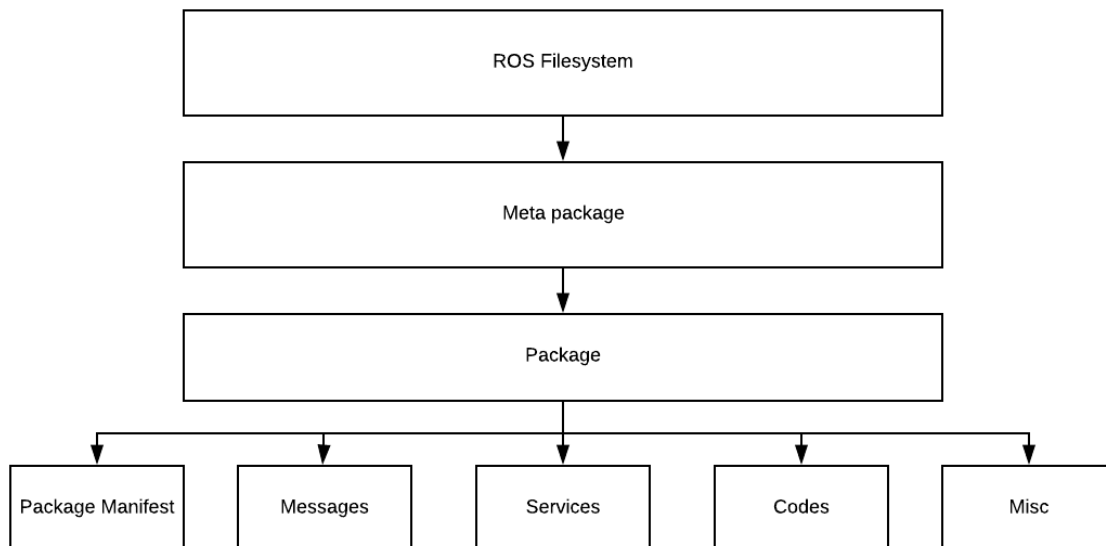
En video på Youtube [33] viser hvordan en Nederlansk ingeniør, Bjorn Harms har laget en fjernstyrt Corvette. Denne er lovlig å kjøre på vei som har likheter med hva KDA ønsker fra prosjektet. Det endte opp med at gruppen kontaktet vedkommende via Facebook. Han var veldig sikkerhetsorientert, så mange av rådene dreide seg om trygge testområder, ekstra nødstopper, mottakere og transmittere, samt sikkerhetsnett.

10.3 Forsterke ATV-ens servo

I ATV-en eksisterer det allerede en servo. Den fungerer når motoren er på og assisterer en bevegelse som en vanlig ATV-operatør allerede gjør. Når føreren svinger på rattet hjelper servoen til. Superatv [34] ble kontaktet på telefon for å høre om det er mulig å bruke servoen som sitter i ATV-en fra før. Servoen på ATV-en aktiveres kun ved bevegelse. Dette gjorde at denne løsningen ikke ble valgt.

10.4 Robot Operating System (ROS)

ROS er et utviklingspråk som blir brukt til simulering av roboter. Dette kan brukes til å simulere kjøringen av ATV-en. Dette vil vise om det har blitt valgt riktige aktuatorer og hva slags sensorer som gir gode målinger til de forskjellige kravene.



Figur 60: ROS Filsystem

Det ble innskaffet informasjon via nettsider og sett på hvordan ROS er bygd opp. Det har sin egen unike oppbygning som er viktig for arbeidet med det. I figur 60 kan man se grovt hvordan dette er. I boken ROS Mastering [80] beskriver de denne oppbygningen slik:

Nodes

Dette er alle tingene som er nederst på modellen, altså "package manifest", "messages" osv. Dette er alle de små delene av ett program. De er en prosess som gjør utregning, hvor hver node er skrevet i ROS client bibliotek som roscpp (c++) og rospy(python). Ved å bruke disse API-ene kan man implementere forskjellige ROS funksjonaliteter, som f.eks. at nodene snakker med hverandre. Poenget med å lage mange små noder er at man skal lage små prosesser istedenfor en stor, slik at de blir lettere å lese samt debugging.

Message

Dette er hvordan nodene kommuniserer. Først lager en node en "message" som blir sendt ved bruk av "busser" med ulike navn som kalles "topics". Når en node sender en melding via en topic, sier man at noden publiserer en topic. En annen node kan man sette til å ta imot denne topicen, som man da sier at noden er en subscriber til en topic. Disse nodene opererer uavhengige av hverandre.

Services

I noen robot applikasjoner passer ikke "topics", da sier man at det er mer en request/respons interaksjon. Hvor en node spør en annen node om å gjøre noe.

Misc

Ros tilbyr ett omfattende loggesystem for å lagre data. Dette innebærer alt av sensor data som ellers kan være vanskelig/umulig å få tak i, men som kan være nødvendige for å utvikle robotalgoritmene som behøves. "Bags" er en del av dette, som kan inneholde meldingsdata fra "topics" og "services", da brukes en .bag filekstensjon for å representere disse filene.

Packages

Ros Package er det mest grunnleggende i ROS-programmet. Denne skal knytte alle nodene sammen for at "topics" og abonnere kommer til riktige plasser. Dette er mer en overordnet fil som holder programmet sammen.

Meta Package

Disse brukes dersom et ROS program skal bestå av flere sammensatte program. Altså at man kan knytte de to programmene sammen.

Oppstart

Etter første møte med veileder som anbefalte sterkt å vurdere ROS gikk SAM i gang med research og utforskning på hva dette var. Det kom frem tidlig at Linux var nødvendig, og det ble da installert versjon 18.04.1 LTS. Dette er den nyeste versjonen av Ubuntu som skal ha long term support.

Etter å ha funnet mye informasjon på wiki.ros ble den nyeste versjonen av ROS, Melodic Morenia, installert. Dette var også en LTS versjon. Denne ble installert da ingen av de andre versjonene av ROS kan installeres på Ubuntu 18.04. Dette ble kilden til ett langt problem, da det meste av informasjon finnes fra tidligere versjoner av ROS.

Mange opplæringsvideoer og de to bøkene brukte ROS versjon Kinetic Kame. Denne versjonen er litt annerledes bygget opp enn Melodic, og de fleste programmene er derfor ikke støttet i Melodic. Etersom ROS er såpass komplekst var det vanskelig å skjønne dette i starten og det ble brukt mye tid på bortkastet jobbing da arbeidet med stoffet tok mye lengre tid ettersom ting ikke fungerte.

Etter andre veileder møte ble dette tatt opp, og Ubuntu 16.04 samt Kinetic Kame ble sterkt anbefalt å installere. Arbeidet fremover vil bli brukt til å følge Ros Programming [84] og Ros Mastering [80] som er to bøker om emnet som ser ut til å gi en god gjennomgang fra begynnelsen.

Emne fra skolen - FE-PRO3910

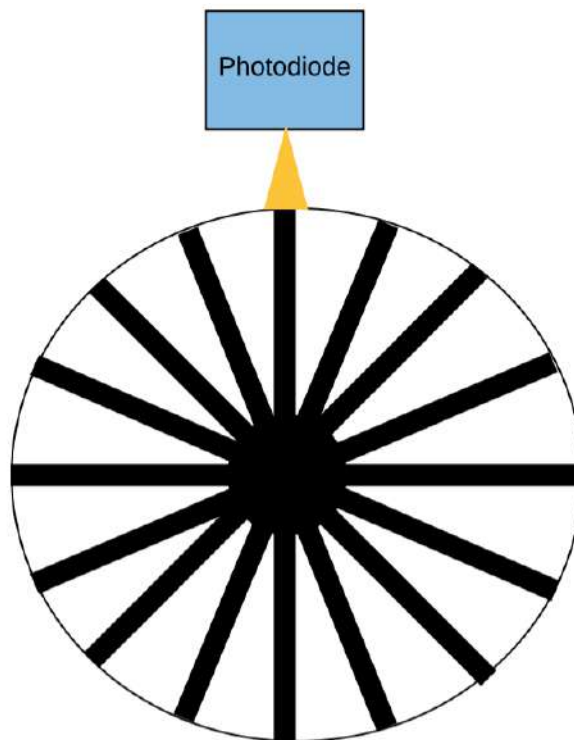
Alle elektro studentene fikk tilbud til å delta i et emne som ble kjørt fra USN-Campus Vestfold. Dette virket bra i starten, men etter å ha kommet et stykke ut i dette ble det klart at ingen hadde kodekunnskapen og grunnlaget kurset bygget på, og det ble meget krevende å få noe ut av ROS under prosjektperioden. Derfor ble det bestemt å gå bort fra bruk av ROS, selvom det hadde vært en stor ressurs for valg av sensorer.

10.5 Fartsmåling med photodiode

Det ble foreslått av gruppens interne veileder å bruke photodioder for å måle farten til ATV-en.

Photodiode-kretser er kretser som konverterer lys om til elektrisk strøm.

Fartsmålingene blir gjort ved at man bruker photodiode-kretser, et for hvert hjul. Og bruker et mønster med svarte biter med et fast mellomrom og et bestemt antall, slik som sett på figur 61 (Satt opp i Lucidchart [41]).



Figur 61: Illustrasjonsbilde av fartsmåling med photodiode

Ved å bruke photodiode-kretsen til å måle tiden på sekvensen som består av en runde rundt med alle de svarte bitene. Kan man med en nøyaktighet avgjort av antall svarte biter og photodiodens nøyaktighet beregne farten til hjulene og dermed ATV-en.

10.6 MODBUS

Modbus er et robust kommunikasjonsverktøy som ofte brukes innenfor industriapplikasjoner. Det brukes i hovedsak for å opprette kontakt mellom flere elektriske enheter samtidig [37].

PLC blir ofte brukt som et bindeledd mellom master og slave for modbus protokollen og står for programmable logic controllers. Det er blitt bestemt å bruke en arduino som PLC, siden resten av systemet skal utvikles gjennom den. Siden arduino egentlig ikke er egnet å bruke sammen med industrielle applikasjoner, som krever mer spenning og robusthet, må man se på løsninger som kan gjøre arduinoen ideell for prosjektets bruk.

Master/slave relasjonen er en kommunikasjonsprotokoll mellom en masterenhet og flere slaves, der master fungerer som en sentral som sender og mottar informasjon fra slavene [37].

RTU; RS-485, RS-232 En kommunikasjonsenhet for å sende over telemetrisk data fra master til slave. Denne enheten er noe utdatert og brukes ikke sammen med nyere system.

TCP - Ethernet kommunikasjon Dette er kommunikasjonsprotokollen som vil bli brukt for dette systemet. Ethernettilkobling blir brukt for å konfigurere motordriveren via software som fulgte med. [18].

Modbus Register typer

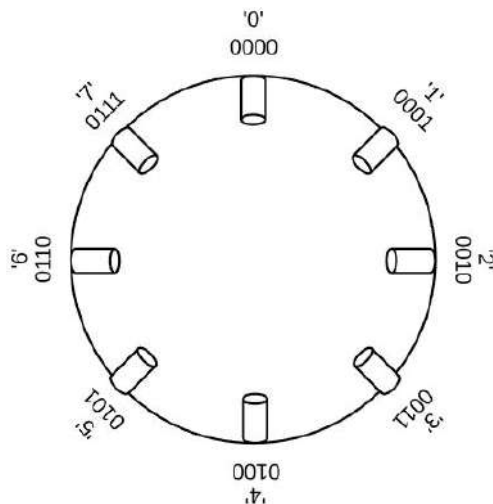
- Coil
- Discrete input
- Input register
- Holding register

10.6.1 Modbus som kommunikasjonsverktøy

Etter samtaler med leverandør som selger steppermotor (Festo), kom det fram at de ikke var kjent med at motoren kunne kontrolleres via arduino.

Modbus kan altså etablere kommunikasjon mellom flere enheter koblet til samme nettverk og blir ofte brukt til ulike målinger som videresender resultatene til en PC.

Ulikt fra en arduino, som sender høye og lave signaler, vil Modbus protokollen dele inn bitene som blir sendt i "bolker" på 8 biter (1 byte), og leser det videre ut i heksadesimaler, i spekteret mellom '00' til 'FF'. Disse henviser til funksjoner som skal utføres av slavene. I dette tilfelle vil det være motorkontrolleren.



Figur 62: Kommunikasjon mellom motor og arduino via modbusprotokoll

Figur 62 viser et eksempel på hvordan disse heksadesimalene peker og henviser til posisjoner i steppermotoren, som i dette tilfellet har 8 posisjoner pr. runde.

10.6.2 Lagring av data i modbus

Dataen og informasjonen som blir sendt vil bli lagret i "slaven" i fire ulike register, avhengig av hva slags data som blir mottatt. To av registerene er for coils og lagrer diskrete verdier (av/på), mens de to andre lagrer numeriske verdier.

Tabell 28: Data lagring i standard modbus

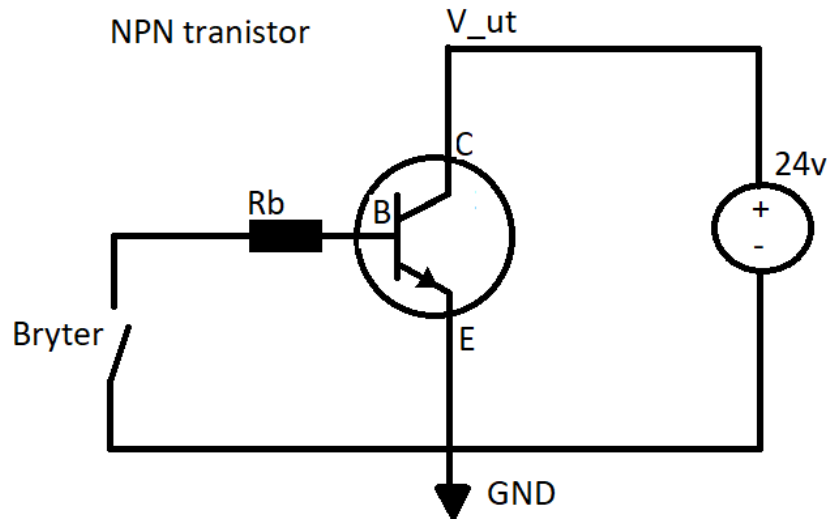
Nummer	Data adresse	Type	Register	Str.
00001 - 09999	0000 - 270E	Les-skriv	Discrete output coils	1 bit
10001 - 19999	0000 - 270E	Kun les	Discrete input contacts	1 bit
30001 - 39999	0000 - 270E	Kun les	Analog input registers	16 bit
40001 - 49999	0000 - 270E	Les-skriv	Analog output holding registers	16 bit

Som tabell 28 indikerer, består hvert register av 9999 verdier og har en tildelt data adresse fra 0000 - 270E. De to første radene i tabellen er tilegnet coils/contacts. Hver av disse er 1 bit, mens hvert register er et ord, som igjen er 16 bit (2 bytes) [73].

Modbus kan dessverre ikke brukes for å styre steppermotoren, da prosjektgruppen ikke har nødvendig kunnskap eller verktøy tilgjengelig for utførelse. Det kreves derfor at det finnes en alternativ løsning for å drive motoren.

10.7 Transistor

Steppermotoren skal drives med 24 V. Det kan brukes en spenningskilde og transistorer som brytere som vist i figur 63. For dette tilfellet har det blitt valgt å bruke NPN-transistorer.



Figur 63: Oppkobling av NPN felles base-emitter transistor krets, hvor bryteren bytter mellom 5 og 0V inngang på base

For å bruke transistoren som en bryter for 24 V kretsen, er det viktig å deklare hva R_B må være for at transistoren skal operere med riktig strøm (I_B). Dette må stemme overens med strømmen i I_C og forsterkningsfaktoren β . Disse transistorene skal kun si når styrepinnene til motorkontrolleren (CMMO-ST) skal være høye eller lave med 24 Volts spenning, og vil derfor ha en konstant strøm inn på I_C [68].

$$I_E = I_C + I_B \quad (18)$$

I_B er såpass liten, slik at:

$$I_C \approx I_E \quad (19)$$

$$I_B = \frac{V_B - V_{BE}}{R_E} - I_E \quad (20)$$

$$I_E = \frac{V_B - V_{BE}}{R_B/\beta} \quad (21)$$

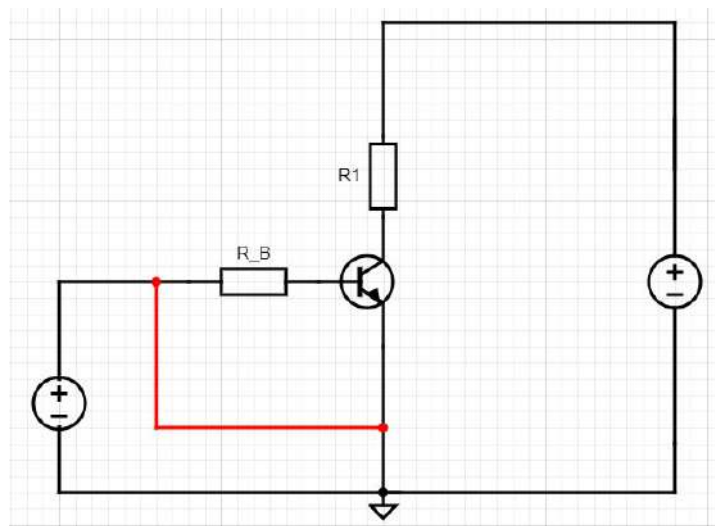
$$I_B = \frac{V_B - V_{BE}}{R_B} \quad (22)$$

R_B blir derfor:

$$R_B = \frac{V_B - V_{BE}}{I_E} \quad (23)$$

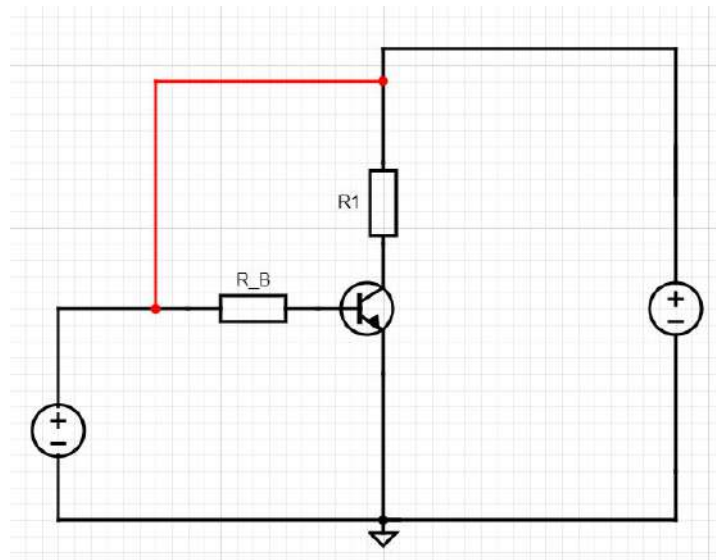
Disse gjelder kun for felles base-emitter kretser, som er den kretsen som blir brukt for dette formålet [8].

For å bruke en transistor som en bryter, må den å enten gå i brytning eller metning for at den skal være på eller av. [67]. Figur 64 viser hvordan kretsen går i "cut-off" og vil dermed få spenning på collectorutgangen.



Figur 64: Kretsen vil gå i "cut-off" og åpne bryteren i transistoren og dermed ha 24V spenning på collector

Figur 65 viser hvordan kretsen går i "saturation" og vil dermed "lukke" transistoren og spenningen vil dermed slippe gjennom og gå til jord.



Figur 65: Kretsen vil gå i "saturation" og lukke bryteren for så å la spenningen gå direkte til jord.

Transistorer virker å være mer stabilt enn å bruke relemoduler som brytere, da disse ikke lar seg påvirke i like stor grad av vibrasjoner og andre ytre påvirkninger.

Det er også blitt sett på anbefalte transistorer til bruk ved bryterfunksjoner på generell basis[2].

Krets 64 og 65 er satt opp i Circuit Diagram [57].

10.8 Bremse-arm

Det har blitt sett på MC og ATV bremse-armer for å studere hvorfor håndtakene har blitt designet på den måten. Disse komponentene har allerede blitt testet, verifisert og serieprodusert.



Figur 66: Den eksisterende bremsehantelen som er på ATV-en



Figur 67: Bilde av ATV-ens eksisterende fotbremssystem

Utvikling-EL

Utvikling av de elektroniske funksjonene til delsystemene.

Innhold:

- Valg av aktuatorer
- Bestilling
- Styring av ATV
- Valve control
- PWM til DC
- Software



KONGSBERG



USN Universitetet
i Sørøst-Norge

11.1 Valg av aktuatorer

Et av de viktigste områdene når det kommer til valg av løsninger er hvilke komponenter som skal velges. Aktuatorene trenger supplementerende deler til kraftoverføring til ATV-en. Maskinkomponentene er veldig fleksible og kan tilpasses i mange geometrier.

Ut ifra målingene kan ytelser og krefter tilpasses til hva som trengs fra aktuatorene. Det har blitt sett på hvilke aktuatorer som er tilgjengelige på markedet. Ut ifra dette, så kan man finne motorer som passer bra til de ulike delsystemene siden komponenter til kraftoverføringen lettere kan tilpasses i forhold til en aktuator.

Det kan justeres slik at momentet på en motor kan trenge mindre lineær kraft for å utføre den ønskede jobben. Hvor lang lengden av operasjonen til motoren blir, påvirker hvor rask aktuatoren må være.

11.1.1 Brems

For å trykke håndbremsen helt inn med en arm på 160 mm trengs det en kraft på 230 N.

Fotbremsen trenger en kraft på 315 N for å trykkes helt inn med en arm på 250 mm. Kraften kan her justeres ved å tilpasse lengde på arm.

Aktuator LA28, type 287XXX-4XXXX2XX - 12V

LA28 with 12V motor

Order number	Push Max. (N)	Pull Max. (N)	*Self-lock max. (N)	*Self-lock max. (N)	Pitch (mm/spindle rev.)	Typical speed (mm/s)		Standard stroke lengths (mm) in steps of 50 mm	Typical amp.	
			Push	Pull		No	Full		12 V	24 V
286XX-XXXX2XX	3500	2000	2000	2000	2	7,0	3,2	100 – 400	6,9	-
285XX-XXXX2XX	3000	2000	3000	2000	2,5	8,6	3,6	100 – 400	6,6	-
281XX-XXXX2XX	2000	2000	2000	2000	3	10,2	6,2	100 – 400	6,9	-
284XX-XXXX2XX	2000	2000	1500	1500	4	13,5	6,8	100 – 400	6,5	-
284XX-4XXXX2XX	2000	2000	2000	2000	4	13,3	7,5	100 – 400	6,5	-
282XX-XXXX2XX	2000	2000	500	500	6	19,9	7,5	100 – 400	7,7	-
282XX-4XXXX2XX	2000	2000	2000	2000	6	19,8	7,0	100 – 400	8,5	-
283XX-XXXX2XX	1500	1500	0	0	9	28,9	11,7	100 – 600	7,9	-
283XX-4XXXX2XX	1500	1500	1500	1500	9	28,9	11,7	100 – 600	7,9	-
287XX-XXXX2XX	800	800	0	0	12	32,0	16,0	100 – 600	5,9	-
287XX-4XXXX2XX	800	800	800	800	12	32,0	16,0	100 – 600	5,9	-

Figur 68: Tekniske spesifikasjoner for LA28 [60]

Aktuator LA30, type 307xx0-4xxxx1/2xx - 12V

Technical specifications:

New type	Spindle pitch (mm)	Thrust max. Push (N)	Thrust max. Pull (N)	*Self-lock max. With/without brake (N)	Typical speed 0/full load (mm/s)	Stroke length (mm)								Typical amp. at full load 12V 24V	
						50	100	150	200P	250	300	350	400P	14	7
307xx0-4xxxx0/5xx	12	1000	1000	1000/0	48/24	50	100	150	200P	250	300	350	400P	14	7
303xx0-4xxxx0/5xx	9	1500	1500	1500/400	42/20	50	100	150P	200	250	300P	350	400	14	7
302xx0-4xxxx0/5xx	6	2000	2000	2000/500	18.5/14	50	100P	150	200P	250	300	350	400P	14	7
301xx0-xxxx0/5xx	3	3000	3000	3000/3000	16/9	50P	100P	150	200P	250	300	350	400	14	6.4
307xx0-4xxxx1/2xx	12	1000	1000	1000/0	65/35	50	100	150	200P	250	300	350	400P	20	10
303xx0-4xxxx1/2xx	9	1800	1800	1800/0	52/25	50	100	150P	200	250	300P	350	400	20	10
302xx0-4xxxx2xx	6	2000	2000	2000/500	34/20.3	50	100P	150	200P	250	300	350	400P	18	-
302xx0-4xxxx1xx	6	2400	2400	2400/500	34/20.3	50	100P	150	200P	250	300	350	400P		9
301xx0-xxxx1/2xx	3	3500	3000	3500/3500	17/9.5	50P	100P	150	200P	250	300	350	400	18	9

Figur 69: Tekniske spesifikasjoner for LA30 [61]

Aktuator LA33, type 33200xxxxxxxxxA... - 12V

LA33 with 12V motor

Type	Thrust max. Push/Pull (N)	Self-lock max. Push (N)	Self-lock max. Pull (N)	Spindle Pitch (mm) / Gear	*Typical speed (mm/s)		Stroke length (mm) in steps of 50mm			*Typical Amp. (A)	
					No load	Full load	Min.		Max.	No load	Full load
33090xxxxxxxxxA...	5000	5000	5000	9 / A	9	6	50	-	300**	2.8	10
33150xxxxxxxxxA...	3500	3500	3500	15 / A	15	9	50	-	400**	2.8	10
33150xxxxxxxxxA...	2250	2250	2250	15/B	25	21	50	-	500**	2.8	10
33200xxxxxxxxxA...	1500	1500	1500	20 / B	34	24	50	-	600	2.0	10

Figur 70: Tekniske spesifikasjoner for LA33 [62]

Aktuator LA36, type 36200xxxxxxAxxxxFxxxxxxxxxxxx - 12V

Technical specifications

LA36 with 12V motor

Order number	Push max. (N)	Pull max. (N)	*Self-lock min. (N) Push	*Self-lock min. (N) Pull	Pitch (mm/spindle rev)	*Typical speed (mm/s) load		Standard stroke lengths (mm) In steps of 50 mm	*Typical amp. (A) 12 V	
						No	Full		No load	Full load
36080xxxxxxxxAxxxxHxxxxxxxxxx	10000	10000	13000	13000	8	11	7	100 - 999**	4.5	22
36120xxxxxxxxAxxxxFxxxxxxxxxx	2600	2600	3400	3400	12	40.7	30.6	100 - 999	4.5	21
36120xxxxxxxxAxxxxGxxxxxxxxxx	4500	4500	5800	5800	12	23.1	17.8	100 - 999**	4.5	20.7
36120xxxxxxxxAxxxxHxxxxxxxxxx	6800	6800	8800	8800	12	15.5	11.9	100 - 999**	4.5	21
36200xxxxxxxxAxxxxFxxxxxxxxxx	1700	1700	2200	2200	20	68	52	100 - 999	4.5	22
36200xxxxxxxxAxxxxExxxxxxxxxx	500**	500***	1000	1000	20	160	135	100 - 999	4.5	20

Figur 71: Tekniske spesifikasjoner for LA36 [63]

Aktuatorene som vil være mest aktuelle til bremsere er LA28 eller LA33. Disse aktuatorene er raske og kraftige nok, samtidig som de ikke blir for store.

11.1.2 Gir

For å kunne skifte gir på ATV trengs en kraft på 70 - 90 N. Slaglengden på aktuatoren bør være ca 100 mm, da kjøretøyet har 5 girposisjoner med ca 20 mm avstand. Siden ATV ikke vil skifte gir mens den kjører er ikke kravet til hurtighet på aktuator like stor, men den bør allikevel ikke ha en hastighet lavere enn 20 mm/s.

LA12 oppfyller disse kravene og være et aktuelt valg girsystemet

Aktuator LA12, type 12XX00-3XXX12XX - 12V, 100 mm versjon

Technical specifications

Type	Motor voltage (V)	Spindle Pitch (mm)	Thrust max. Push/Pull (N)	Self-lock max. (Push) (N)	Self-lock max. (Pull) (N)	*Typical speed (mm/s)		Stroke length (in steps of 30 mm)			*Typical Amp. (A)	
						No load	Full load	Min.	Max.	No load	Full load	
12XX00-1XXX12XX	12	2	750	750	375	14	5	40	-	130	1.75	4.6
12XX00-1XXX24XX	24	2	750	750	375	14	6	40	-	130	0.75	2.2
12XX00-2XXX12XX	12	4	300	300	150	27	16	40	-	130	1.75	2.5
12XX00-2XXX24XX	24	4	300	300	150	27	16	40	-	130	0.75	1.5
12XX00-3XXX12XX	12	6	200	200	100	40	28	40	-	130	1.75	2.2
12XX00-3XXX24XX	24	6	200	200	100	40	28	40	-	130	0.75	1.0

Figur 72: Tekniske spesifikasjoner for LA12 [58]

Et annet godt alternativ til en aktuator som oppfyller kravene, samtidig som den er både raskere og mindre er:

Aktuator LA14, type 14040xxxxxxxxxC - 12V

Technical specifications

Type	Motor voltage (V)	Spindle Pitch (mm)	Thrust max. Push/Pull (N)	Self-lock max. (Push) (N)	Self-lock max. (Pull) (N)	*Typical speed (mm/s)		Stroke length (mm)			*Typical Amp. (A)	
						No load	Full load	Min.	Max.	No load	Full load	
14020xxxxxxxxxA...	12	2	750	750	375	15	8	19	-	130	0.3	2.4
14020xxxxxxxxxB...	24	2	750	750	375	15	9	19	-	130	0.15	1.3
14020xxxxxxxxxC...	12	2	750	750	375	23	15	19	-	130	0.4	4.2
14020xxxxxxxxxD...	24	2	750	750	375	24	16	19	-	130	0.2	2.5
14040xxxxxxxxxA...	12	4	300	150	150	29	20	19	-	130	0.4	1.7
14040xxxxxxxxxB...	24	4	300	150	150	31	24	19	-	130	0.2	0.9
14040xxxxxxxxxC...	12	4	300	100	100	43	36	19	-	130	0.5	2.6
14040xxxxxxxxxD...	24	4	300	100	100	45	38	19	-	130	0.3	1.3

Figur 73: Tekniske spesifikasjoner for LA14 [59]

11.2 Bestilling av aktuatorer

Etter møtet med Linak 25.03 ble det foreslått å kjøpe inn 2 stk LA33 og en LA12 aktuator. Disse vil bli bestilt sammen med tilhørende styreenhet TR-EM-288 til hver av

aktuatorene, denne vil også trenge TR-EM-236 for programmering. LA36 ble frarådet da den er vanskelig å kontrollere presist i så store hastigheter som trengs til dette prosjektet. Leveringstiden på aktuatorerne er 4-5 uker, noe som kan gi tidsproblemer i forhold til testing. For å løse dette har Linak disponert en LA12, slik at tesingen kan påbegynnes med en gang.

11.2.1 Bremsesystem

Etter møte med oppdragsgiver ble det bestemt å bestille $2 \times$ LA33 aktuator, en til håndbrems samt en til fotbrems. Disse ble bestilt torsdag 28.03.2019. LA28 ble valgt bort da denne ikke ble levert med ønsket mulighet for tilbakekobling.



Figur 74: Aktuator LA33

11.2.2 Girsystem

Det ble torsdag 28.03.2019 også bestilt en aktuator LA14 til girsystemet. Denne ble valgt fremfor LA12 da den, som tidligere nevnt, er både mindre og raskere.

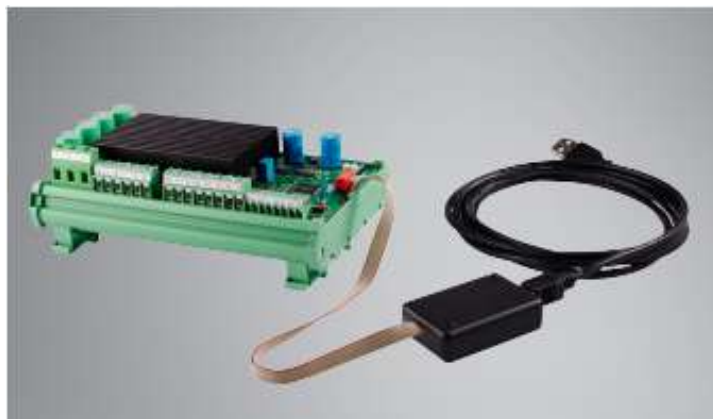


Figur 75: Aktuator LA14

11.2.3 Kontrollenhet for aktuator

TR-EM-236

Programmeringsenhet for å programmere TR-EM-288. For å sette forskjellige parametre, som strømbegrensning og hastighet.



Figur 76: TR-EM-236

TR-EM-288

Strømbegrenser - PWM-signal for å styre, en til hver av aktuatorene.



Figur 77: TR-EM-288

11.2.4 Bestillingsoversikt - aktuatorer

Den endelige bestillingen av aktuatorer fra Linak vises i tabell 29.

Tabell 29: Bestillingsoversikt fra Linak - aktuatorer

Vare	Beskrivelse	Pris netto pr. enhet
Aktuator LA14	Kraft: Maks.300 N Hastighet: 43 mm/s ubelastet Slaglengde: 130 mm Innbyggingsmål: 245 mm Farge: Svart Tetthet: IP66 Tilbakemelding: 0,5-4,5V	NOK 2257,-
Aktuator LA33	Kraft: Maks. 1500 N Hastighet: 34 mm/s ubelastet Slaglengde: 100 mm Innbyggingsmål: 260 mm Motor: 12VDC , 10 Amp Tetthet: IP66 Interne endestoppbrytere Tilbakemelding: 0,5-4,5V	NOK 3182,-
Strømbegrenser TR-EM inkl. programmeringsenhet TR-EM-236	AC: 12-35VDC Justerbar strømbegrenser 1-20 A Impulsstyring Soft start	NOK 820,-

11.3 Driver - TR-EM-288-H

Det er blitt bestilt og mottatt drivere/kontrollere som er kompatibel med alle de planlagte lineæraktuatorene som skal brukes i prosjektet(LA14, 2×LA33). Denne brukes hovedsakelig for å styre retning og fart av DC-motoren. Figur 78 viser tilkoblingsklemmene til driverkortet [11].



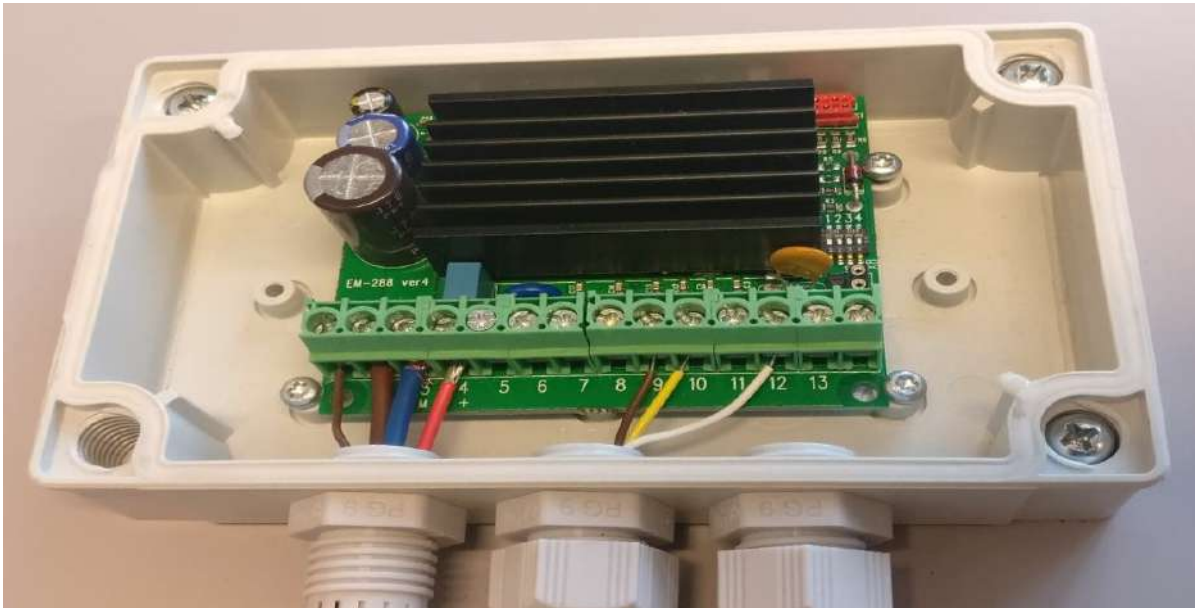
Figur 78: Pinne nr og hva de gjør

- kb1: 0VDC
- kb2: Motor positiv
- kb3: Motor negativ
- kb4: 12VDC
- kb5: VCC
- kb6: Fram - sikkerhetsnett(stopp)
- kb7: Tilbake - sikkerhetsnett(stopp)
- kb8: GND
- kb9: Styringspinne - fremover
- kb10: Styringspinne - bakover
- kb11: Stopp
- kb12: Speed-2
- kb13: Feilbryter
- kb14: 5V ut.

Det brukes en ekstern spenningskilde på 12 VDC inn på pinne 1 og 4 for å tilføre driverkontrolleren konstant spenning. Pinne 2 og 3 er koblet til moteren. Pinne 8 er jord.

Styringspinnene 9, 10 og 12 er koblet til Arduinoen for å regulere posisjonene for fremover, bakover og fart, respektivt.

Figur 79 viser hvilke tilkoblingsklemmer som brukes.



Figur 79: Tilkoblinger brukt i systemet.

Det finnes to konfigurasjoner for hastighet (speed-1, speed-2) hvor driveren opererer med speed-2 som default, dersom koblingen fra pinne 12 er sluttet. Hvis den er brutt, vil den bruke speed-1. Driveren ble konfigurert slik at speed-2 ble satt til max ytelse, og speed-1 på null ytelse. Da kan det genereres et PWM-signal med veksling mellom hastighetene for å få en regulering på farten.

Dersom pinne 9 (fremover) er høy og pinne 10 (bakover) er lav, vil aktuatoren gå fremover.

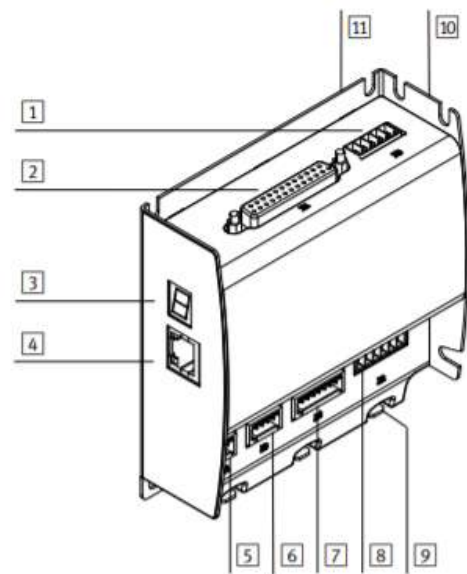
11.4 Styring av ATV

For å styre ATV vil det bli brukt en stepper-motor sammen med en motor-kontroller og Arduino.

11.4.1 Motor-kontroll

Figur 80 viser en oversikt over motor-kontrolleren og komponentene som vil bli tatt i bruk. De ulike komponentene vil bli beskrevet nærmere under forklaringen av hvordan implementeringen mellom motor-kontroll, stepper-motor og Arduino vil skje ([21] - kap. 2.2).

- 1: [X9] Load/logic spenning
- 2: [X1] I/O Interface for kontrollsystem
- 3: 7-segment display
- 4: [X18] Parameter interface Ethernet (RJ-45)
- 5: [X1A] Referanse switch
- 6: [X3] STO (Safe Torque Off)
- 7: [X2] Encoder kabel (RS422)
- 8: [X6] Motor kabel
- 9: Signaljord (3x)
- 10: Mounting surface (H-rail)
- 11: Mounting surface



Figur 80: Komponentene til CMMO-motor kontroller

Motor-kontrolleren som vil bli brukt er typen CMMO-ST-C5-1-DIOP [19]. Beskrivelse av valgt motor-kontroll kommer frem i tabell 30 nedenfor:

Tabell 30: Beskrivelse av valgt motor-kontroll

Forkortelse	Forklaring
CMMO	Motor-kontroller
ST	Stepper-motor
C5	Nominell strøm - 5A
1	Inngangspenning - 24V
DIO	Aktivering - Digital I/O Interface
P	Switching inngang/utgang - PNP

Driftmodus til motor-kontrolleren er en kaskade kontroll med PI-kontroller for hastighet, PI-kontroller for strøm og P-kontroll for posisjon.

En stepper-motor kontrollert med en encoder kabel gjør at den kan operere opp til sin ytelsesgrense. Fordi encoderen har samme rotorposisjon som stepper-motoren og også rapporterer om dette tilbake til posisjonskontrolleren, vil upresise posisjoner forårsaket av vinkelavvik i steppermotor forhindres.

For at stepper-motoren skal starte fra samme utgangspunkt eller nullpunkt ved oppstart brukes en metode kalt homing. Denne metoden brukes for å definere et referansepunkt og gjør også at det ønskede referansepunktet lettere kan finnes frem til da referansepunktet vil være det absolutte referansepunktet for aksens nullpunkt.

For å rotere stepper-motoren kontinuerlig mot en retning vil det bli tatt i bruk jogging. Ved hjelp av Festo Configuration Tool (FCT) kan parameterene settes til negativ eller positiv retning, slik at man kan svinge mot høyre og venstre ved hjelp av dette ([21] - kap. 2.5).

11.4.2 Stepper-motor

Figur 81 viser bilde av stepper-motoren som skal styre ATV. Motoren som har blitt valgt er en EMMS-ST-87-L-SE-G2 med 9:1 gir. Grunnen til at det må være 9:1 gir er for at den skal være kraftig nok til å kunne styre ATV-en. Stepper-motoren har i utgangspunktet en holding torque (mengden moment som trengs for å kunne bevege motoren et fullt steg) på 9.3 Nm, slik at det vil være nødvendig med giret for å ha nok kraft til å kunne svinge. Stepper-motoren vil da få en torque på;

$$9.3Nm \times 9 = 83.7Nm \quad (24)$$



Figur 81: Stepper-motor EMMS-ST

Beskrivelse av valgt motortype kommer frem i tabell 31 [20]:

Tabell 31: Beskrivelse av valgt motortype

Forkortelse	Forklaring
EMMS	Motor
ST	Stepper
87	Flensdimensjon - 87 mm
L	Total lengdeoperasjon - Large
SE	S - Straight connection E - Encoder
G2	Second Generation

11.5 Valve Control

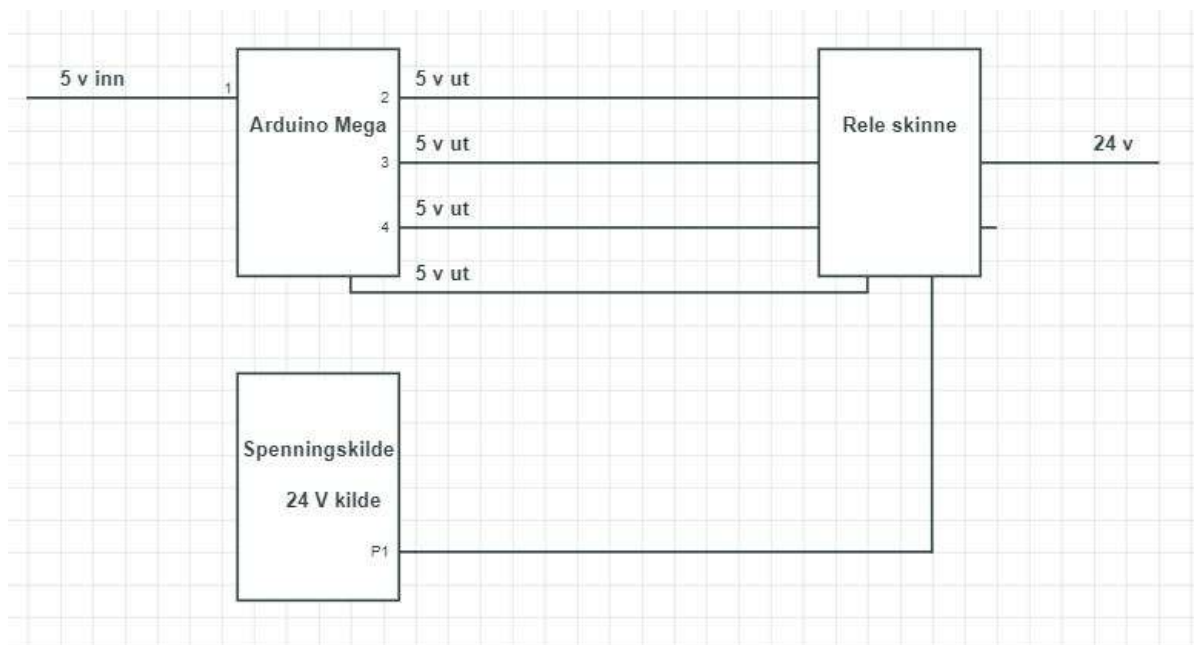
11.5.1 Kontroll via I/O styreenhetsprofil (ventil)

Etter samtale med Festo ble det anbefalt å kjøre stepper-motoren i ventilmodus ([21] - kap. 5.5). Denne metoden er basert på metoden for pneumatiske ventilpiloter og er mindre krevende i forhold til programmeringskunnskap man trenger for å kunne styre. Ved hjelp av programmet FCT kan man sette opp motor-kontrolleren og definere de ulike digitale inngangene man trenger. Man kan definere opptil 11 ulike innganger, mens i dette tilfellet er det styringen av ATV stepper-motoren vil bli brukt til, slik at de digitale inngangene vil bli definert ved fire ulike pinner:

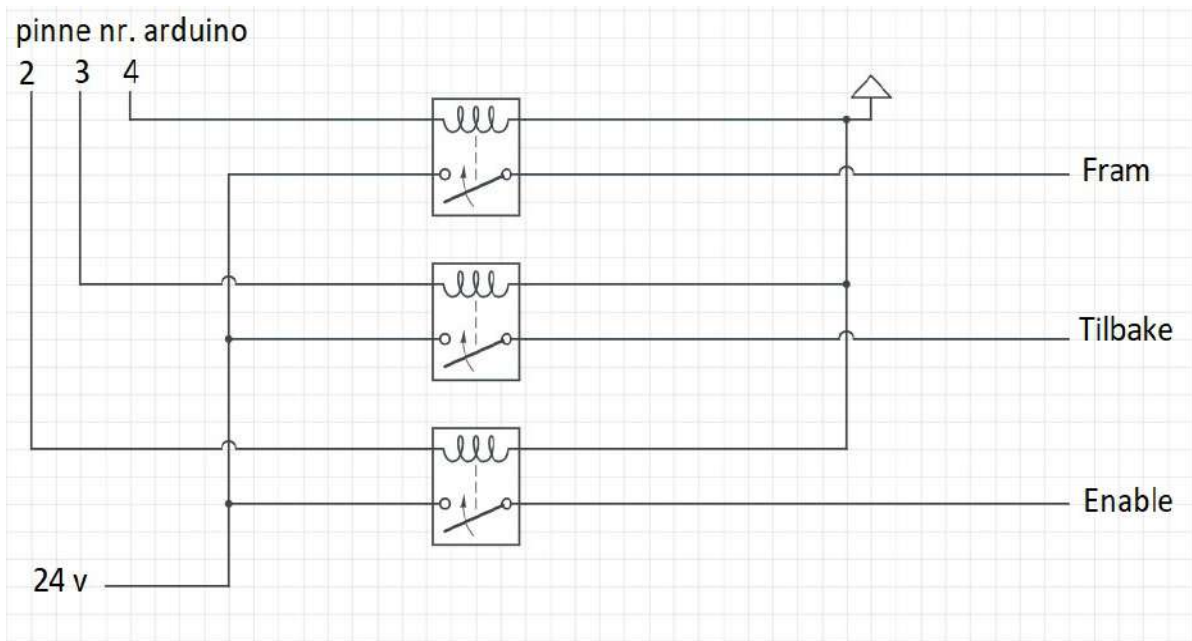
- X1. Pin 1 - Sving mot venstre
- X1. Pin 2 - Sving mot høyre
- X1. Pin 8 - Homing
- X1. Pin 10 - Control Enable

11.5.2 Arduino som PLC

For å kunne styre motoren gjennom motordriveren er det nødvendig med en I/O-kontroller som sender ut og mottar 24V for kontrollsystemet i X1. Arduinoen har begrensninger på 5V på inn- og utgangene, så den er derfor, i utgangspunktet, uegnet som PLC for denne applikasjonen. Det vil derfor bli behov for å bruke en ekstern spenningskilde til dette formålet, henholdsvis batteriet som sitter på eksisterende system.



Figur 82: Arkitektur på releoppsett



Figur 83: 24V innganger med brytere i form av relemoduler som aktiveres med arduinoens pinneutganger

Det må brukes en transformator for å skalere opp fra 12VDC til 24VDC og videre til relemoduler som man kan bruke arduinoen til å aktivere, for så å sende gjennom 24V til motordriveren. (Figur 82)

Releene krever en del strøm for å kunne aktiveres, slik at en ekstern strømkrets vil kunne være nødvendig for å kunne drive alle sammen samtidig. Pinne 2, 3 og 4 aktiverer releene som vist i figur 83.

Figur 82 og 83 er satt opp i Circuit Diagram [57].

11.5.3 Konfigurering av motordriver

En CMMO-ST (motordriver) har blitt mottatt for å konfigurere og gjøre klart oppsettet for hvordan steppermotoren skal operere for dette prosjektet. Gruppen har også fått tilsendt all nødvendig software for at dette skal kunne utføres (FCT) ([21] - kap. 5.5).

Motordriveren ble koblet inn i en trådløs router, da ingen i prosjektgruppen har ethernet-inngang på sin PC. Driveren ble kontaktet gjennom tilkobling til routeren, for å åpne software og et nytt prosjekt for å velge type driver og steppermotor som skal konfigureres.

Scanner deretter og ser om enheten er synlig for PC-en og kobler til. Man kan derfra følge stegene under for å sette opp driveren etter ønskede parametere:

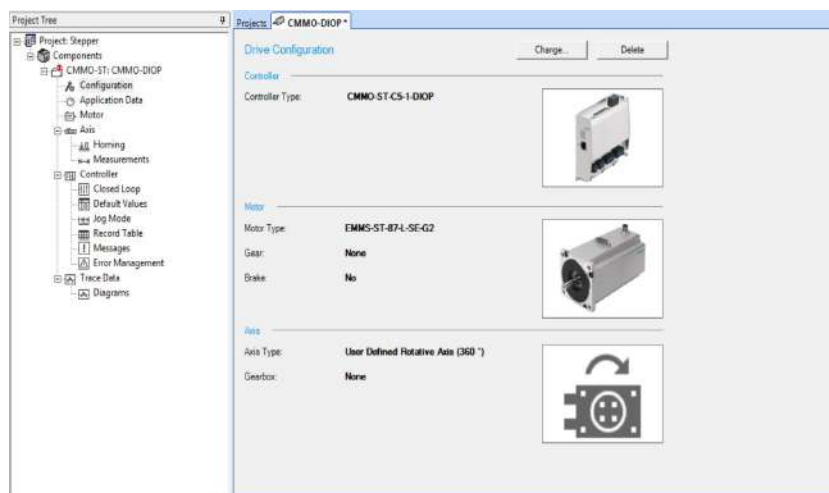
1) Configuration

Velger hvilken kontroller og motor man vil konfigurere og definerer aksene den skal bevege seg langs.

Motor: EMMS-ST-87-L-SE-G2

Kontroller: CMMO-ST-C5-1-DIOP

Rotasjonsakse: 360 grader

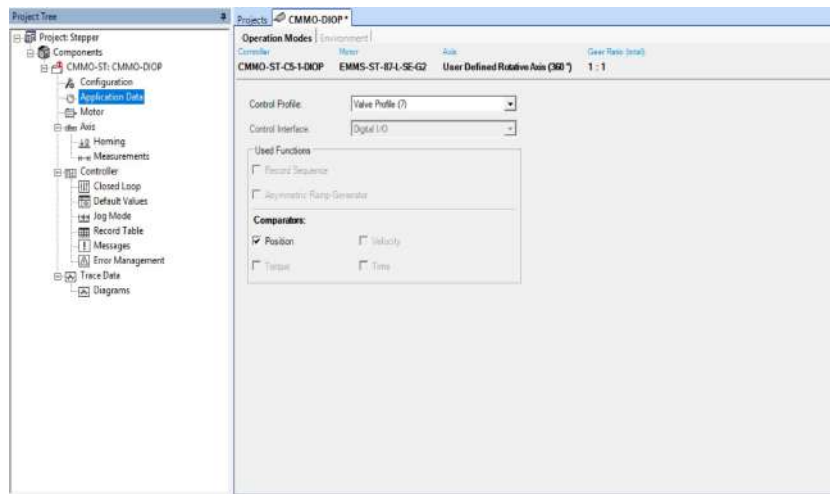


Figur 84: Konfigurering av kontroller og stepper motor

2) Application data

Siden arduino ikke klarer å kjøre hardwaren (24V I/O), blir man nødt til å se bort fra kontrollprofilen som kommuniserer binært og i forhold til posisjonsetting heller se mot

en annen løsning som innebærer ”ventil” teknikk: Så lenge inngangen er høy, vil motoren gå mot gitte begrensninger. I dette tilfellet kalles det ”jogging” hvor steppermotoren vil gå mot gitt retning ift oppkobling i en konstant fart til de ytre grensene som er satt.



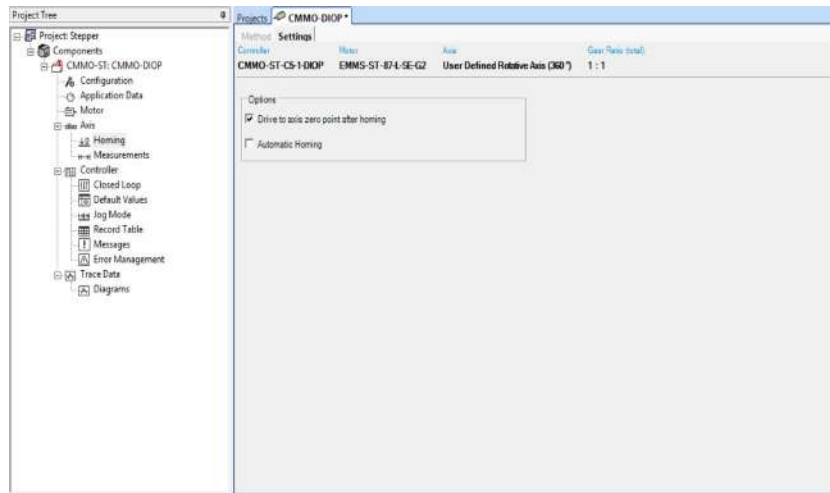
Figur 85: Ventil(Valve) som gir mulighet for operasjon av stepper så lenge inngang er høy.

Dersom man velger å bruke Valve (ventil) operasjon, må man forvente at man ikke får like nøyaktig regulering på fart og posisjon.

3) Homing

Homing handler om at steppermotoren nullstiller seg etter hvert bruk; Ved oppstart vil motoren gå tilbake til ”zero point position”, som i dette tilfellet tilsvarer en vinkel på 45° i et span på 90°

Man bruker nåværende posisjon som referansepunkt for homingfunksjonen. Ved oppstart vil stepperen tilbakestilles til nøytral posisjon basert på hvilken posisjon den er i, i det den starter.

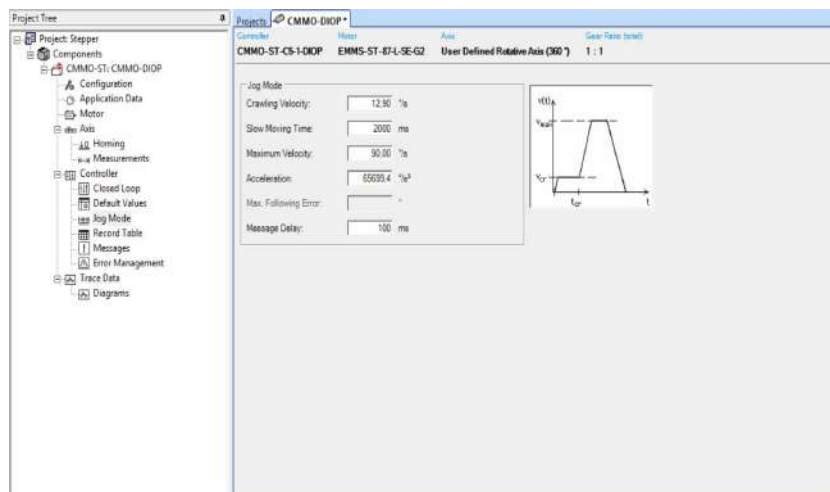


Figur 86: Innstilling for hvordan man vil at homingfunksjonen skal operere.

Homingfunksjonen må aktiveres før hvert kjøreinterval for at styringen skal stå i nøytralposisjon, slik at det ikke overstyres.

4) Jogging

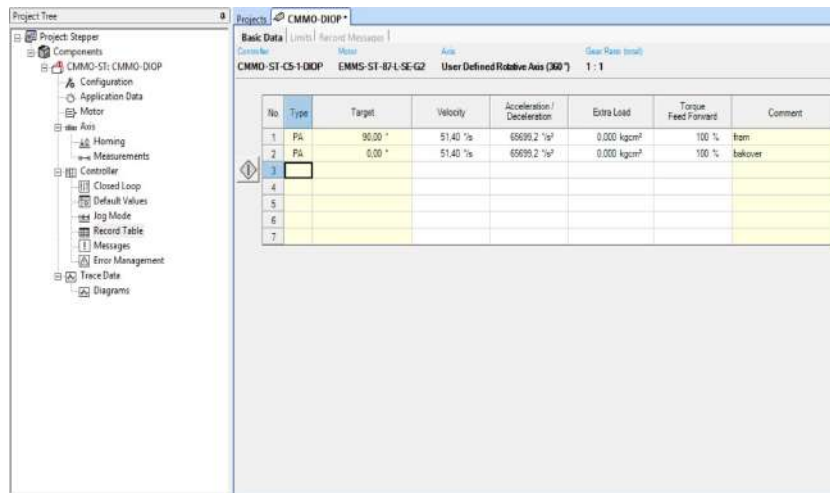
Jogmode er konfigurasjonen for valg av fartsinnstillinger i Jogfunksjonen. Her kan man velge hva krypfart og maksfart skal være, og hvilke parametere de skal være satt til. Figur 87 viser hvordan parameterene settes opp.



Figur 87: Konfigurasjonsvindu for Jog mode

5) Record Table

Her settes parameterene for grensene for hvor langt motoren skal ”jogg”. I dette tilfellet er det brukt to styrepinner som styrer med og mot klokka.

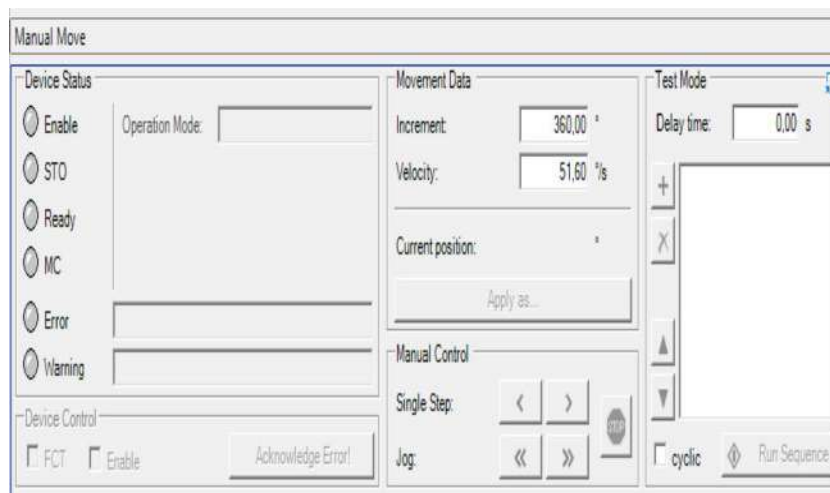


The screenshot shows a software window titled 'Projects: CMMO-DIOP'. On the left is a 'Project Tree' with a tree view containing folders like 'Components', 'Configuration', 'Motor', 'Axis', 'Controller', and 'Trace Data'. The main area displays 'Basic Data' for 'Limits / Record Messages'. Below this is a table with columns: No., Type, Target, Velocity, Acceleration / Deceleration, Extra Load, Torque Feed Forward, and Comment. The table contains two rows of data for 'P.A.' and a third empty row.

No.	Type	Target	Velocity	Acceleration / Deceleration	Extra Load	Torque Feed Forward	Comment
1	P.A.	90,00 °	51,40 %s	65699,2 %s²	0,000 kgcm²	100 %	fram
2	P.A.	0,00 °	51,40 %s	65699,2 %s²	0,000 kgcm²	100 %	bakover
3							
4							
5							
6							
7							

Figur 88: Grensene for joggingfunksjonen. Fra og til grense på de to styrepinnene

Når disse parameterene er bestemt, kan man laste ned konfigurasjonsdataene over til motordriveren via en ethernetkabel. Man kan også teste hvordan motoren fungerer under gitte parametere før man laster den opp, ved å gå til et styringspanel i softwaren (figur 89)



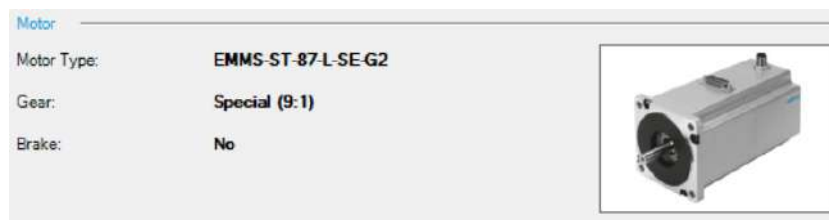
Figur 89: Manuell styring av driveren

11.5.4 Konfigurering av motordriver - iterasjon 2

Stepper-motoren ble testet gjennom programmet FCT med de angitte parameterene fra 11.5.3. Etter disse testene ble det gjort noen endringer av parametere.

1) Configuration

Her har det kun blitt gjort en endring. Giret til stepper-motoren ankom 9. mai og dette ble montert på stepper-motoren og videre lagt inn på “Gear” i konfigurering av motor (figur 90).



Figur 90: Konfigurering av motor - gir

Homing

Det viste seg under testing at homing-funksjonen ikke fungerte helt som forventet. Ved å sette “Axis zero point” til 45° forflyttet motoren seg 45° mot høyre hver gang funksjonen ble kjørt. Dette vil ikke være ideelt ved implementering på ATV da den etter hvert vil gå forbi sin maksimale svingradius på ca 45°. “Axis zero point” ble som vist i figur 91 derfor satt til 0°, slik at styre vil stå i samme posisjon som etter forrige kjøring ved oppstart. Analogstikken til kontrolleren trekkes inn mot midten av akselen når den ikke lenger er operativ, slik at dette blir brukt som et referansepunkt for styring og homing ikke lenger blir nødvendig. På denne måten unngår man dermed skader på styringssystemet ved at maksimal svingradius blir overskredet når homing-funksjonen kjøres.

Homing Method

Target:

Direction: Negative Positive

Method Description: **-35: Current Position**

Parameters

Search Velocity: %/s

Crawling Velocity: %/s

Drive Velocity: %/s

Acceleration: %/s²

Axis Zero Point: °

Figur 91: Konfigurering av homing - Axis zero point

Et videre problem med homing-funksjonen er at dette vil være utgangspunktet for hvor nullpunktet er definert, slik at om ATV-en stoppes ved 60°, vil dette defineres som det nye nullpunktet og man kan ikke svinge mer enn 15° mot høyre før maksimal svingradius er oppnådd. Det vil derfor være viktig at ATV-en stoppes ved 45°, som er det ønskede definerte nullpunktet på svingradiusen, frem til det kommer opp en bedre løsning til dette problemet.

Record Table

For å kunne fjernstyre stepper-motoren skal signalene til styrepinnene aktiveres ved hjelp av en arduino som beskrevet i 11.5.2. Ved bruk av “jogging-metoden” kan man klare seg med to styrepinner som går med eller mot klokka siden man kan forflytte den så langt man vil frem til grensen er nådd (Fra 0 - 45° i hver retning). I tilfellet med arduinoen vil det være behov for flere styrepinner for å kunne definere flere vinkler siden de kun vil aktiveres dersom signalet er høyt. Styrepinnene ble definert i FCT som vist i figur 92.

No.	Type	Target	Velocity	Acceleration / Deceleration	Extra Load	Torque Feed Forward
1	PA	90,00 °	51,40 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
2	PA	75,00 °	30,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
3	PA	60,00 °	30,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
4	PA	45,00 °	51,40 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
5	PA	30,00 °	30,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
6	PA	15,00 °	30,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
7	PA	0,00 °	51,40 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %

Figur 92: Konfigurering av styrepinner - vinkeldefinisjon

Ved å ta i bruk alle de tilgjengelige styrepinnene til motordriveren vil man kunne forflytte seg i intervaller på 15° isteden for 45° . Det definerte nullpunktet vil fortsatt være ved 45° , 0° er full sving mot venstre og 90° er full sving mot høyre.

Som vist i figur 89 kan man også teste hvordan motoren fungerer under de gitte parameterene fra styringspanelet og velge forflytting med “Single Step”, før man laster opp til motordriveren.

11.5.5 Kontroll via I/O styreenhetsprofil - iterasjon 2

Etter å ha tilført flere styrepinner til record table i konfigureringsprogrammet fra Festo (FCT) vil de digitale inngangene bli definert på følgende måte:

- X1. Pin 1: 90° - 45° sving mot høyre
- X1. Pin 2: 75° - 30° sving mot høyre
- X1. Pin 3: 60° - 15° sving mot høyre
- X1. Pin 4: 45° - nullpunkt
- X1. Pin 5: 30° - 15° sving mot venstre
- X1. Pin 6: 15° - 30° sving mot venstre
- X1. Pin 7: 0° - 45° sving mot venstre
- X1. Pin 8: Homing
- X1. Pin 10: Control Enable

11.5.6 Arduino som PLC - iterasjon 2

Styring av stepper-motoren med arduino har vist seg å være mer utfordrende enn først antatt. Det har blitt utført ulike tester med relèer hvor motoren var ustabil og ikke utførte operasjonene den var satt til å gjøre. Det gjøres rede for mulige årsaker til dette i test 13.4.2 og 13.4.5.

Som en alternativ løsning for relèene har det blitt forsøkt å bruke NPN transistor som switch og utført ulike tester på dette. I disse testene har det blitt testet ulike typer transistorer for å finne en passende type (13.4.8), test av arduino med transistor (13.4.6)

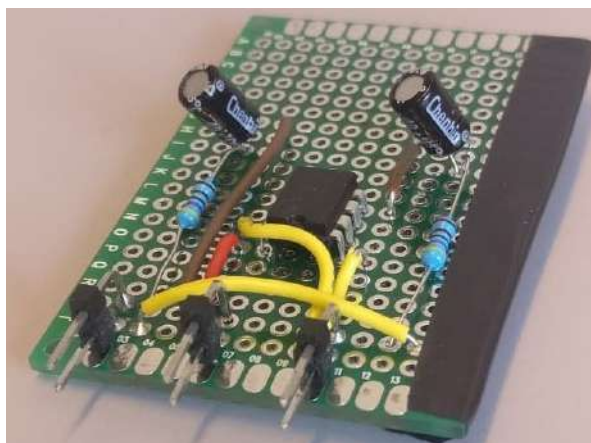
og test av transistor som switch (13.4.7). Ingen av testene kom ut som godkjent slik at fjernstyring og bruk av arduino som PLC fortsatt ikke er en løst problemstilling. Det vil fortsettes å jobbe med dette for å få fjernstyringen til å fungere, men per dags dato er alternativet å styre stepper-motoren fra en PC i programmet FCT via ethernetkabel som er koblet til motordriveren.

11.6 PWM til DC verdier

Systemet kommer til å bruke Arduinoer til å styre ATV-en. Dette byr på en liten utfordring med å gjøre om PWM-signalene om til analoge verdier. F.eks. gassensorene forsyner ATV-en med spenninger mellom 0 og 4V. Disse må emuleres på en slik måte at ATV-en forstår hva den får fra Arduinoen. PWM-signalet fra Arduinoen består av pulser som går fra 0 til 5V, dette må omformes til å gi en stabil DC-spenning som ATV-sensorene forstår. Da kan det brukes ett lavpass filter til å glatte ut spenningen. Dette skjer ved at kondensatoren lades og utlades under hver pulsperiode.

Etttersom det er usikkert på hvor mye strøm kretsene i ATV-en bruker, brukes det en LM358 opamp [44] som en spenningsfølger for å skåne Arduinoen, da den ikke kan gi mye strøm ut. Valget av denne opampen ble gjort ved hjelp av en nettside som har tatt for seg dette temaet [50]. Denne er også veldig lett tilgjengelig slik at den kunne bli kjøpt inn kjapt.

Etter første implementeringstest (TI.27-GA.01) ble det besluttet å lage et prototype kretskort til denne funksjonen som vist i figur 93. De to inngangene er til venstre, utgangene til høyre. VCC på 12V er den øverste pinnen i midten, og den nedre er jord.



Figur 93: PWM - DC prototype kretskort

Dette kortet ble testet (TP.28-GA.01) og er godkjent til implementering.

11.7 Software

11.7.1 Processing og Arduino

Foredelen med å bruke Processing og Arduino er at mange av gruppe-medlemene har programmert opp Arduinoer tidligere. Arduino språket er bygget opp rundt Processing og disse to språkene og programmene fungerer sammen. I Processing kan det lages enkle grafiske tilbakemeldinger til brukeren som kan vise hvilket gir ATV-en er i og annen relevant informasjon. Processing byr også på enkle muligheter til å bruke kontrollere som det vil bli gått nærmere inn på senere.

Arduino: Før man kan starte programmeringen i Processing, må man laste opp "Standard Firmata" fra programmet Arduino IDE, slik at disse kommuniserer med hverandre. Firmata er en generell protokoll for kommunikasjon med mikrokontroller, og man finner ferdig kode for dette under eksempler inne i programmet, Arduino IDE. Koden ligger under vedlegg A.1.

Processing: For at man skal kunne skrive og kjøre en fungerende kode i Processing er man avhengige av noen biblioteker:

- **Game Control Plus** - Gir deg lov til å bruke spillkonsoll sammen med kode i Processing.
- **Arduino Firmata** - Kommunikasjon med Arduinoen.
- **G4P** - Nødvendig for å kunne bruke Game Control Plus - Configurator.
- **Serial** - Leser og skriver data til og fra eksterne objekter. Arduinoen trenger dette biblioteket for at den skal fungere.
- **Control P5** - Tilbyr en rekke kontrollere som enkelt lar deg endre og justere verdier mens programmet kjører.
- **video** - For å bruke kamera i Processing.

11.7.2 Oppsett

Arduino

På Arduinoen som skal brukes må StandardFirmata lastes opp med en baudrate på 57600. StandardFirmata muliggjør at alt av kalkuleringer gjøres på pc-en som brukes og at Arduinoen kun brukes til å sende og motta verdier.

Com port

Dersom det har blitt koblet til flere com-porter på datamaskinen i bruk, kan disse blandes i Processing. Det blir deklarerert at Processing skal bruke den første com-porten. Dersom dette ikke er tilfelle må det endres i koden.

Configurator - spillkontroller For å konfigurere spillkontrolleren i Processing, må den kobles opp mot PC, før man deretter åpner eksempelet "Configurator" under "Game Control Plus" (se vedlegg A.2 for kode). Programmet er klart for å kjøres, slik at man enkelt kan velge hvilke knapper man vil konfigurere. Her har det blitt gitt navn i forhold til hva knappene heter på den faktiske kontrolleren.

I filene lagt ved sammen med Processing-koden A.3, vil følgende konfigurasjoner være lagt inn:

11.7.3 Kontrollerlayout (PS4)

For å styre ATV-en er det tenkt å bruke en PS4-kontroller[7]. Knappkonfigurasjonene er som følger:

Skulderknapper:

R2 (Analog): Gass med variabelt pådrag i èn akse

L2 (Analog): Brems med variabelt pådrag i èn akse.

R1 (Digital): Skifte gir i en retning med et tastetrykk.

L1 (Digital): SKifte gir i en retning med et tastetrykk.

Frontknapper:

X (Digital): Nødstopp med et tastetrykk

trekant (Digital): For å gå ut av oppstartsmodus

L3 (Analog): Styring i èn akse.



Figur 94: Konfigurasjoner på skulderknappene.

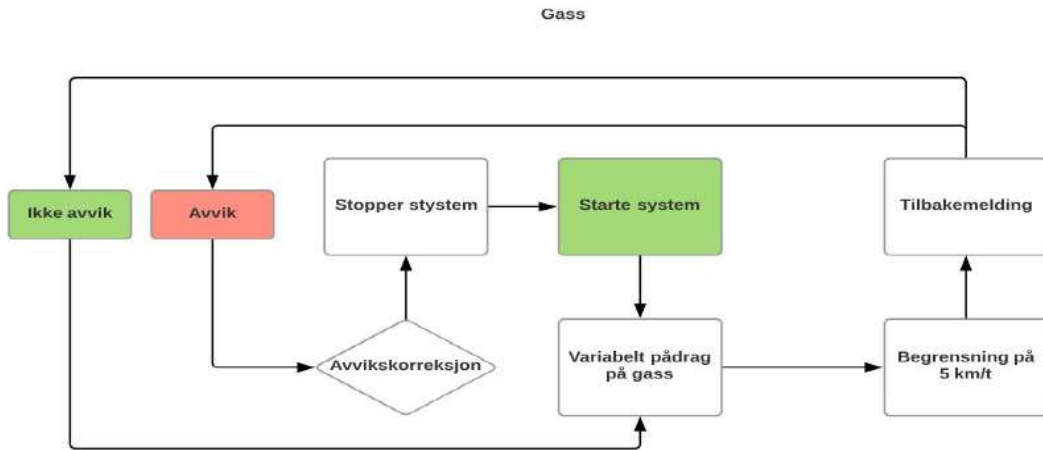


Figur 95: Knappekonfigurasjoner på front

11.7.4 UML: Delsystem

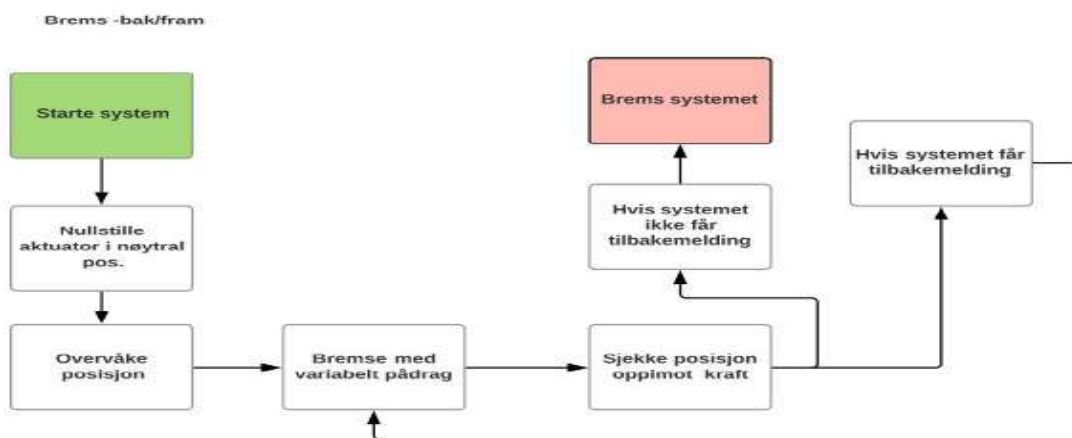
Det er hensiktsmessig å lage UML kart som viser de overordnede funksjonene til delsystemene og hvordan det er påtenkt at systemet skal kobles og kodes opp. Det har derfor blitt lagd sekvensdiagram i form av state machine for å si noe om i hvilken rekkefølge funksjoner skal operere i.

Gass:



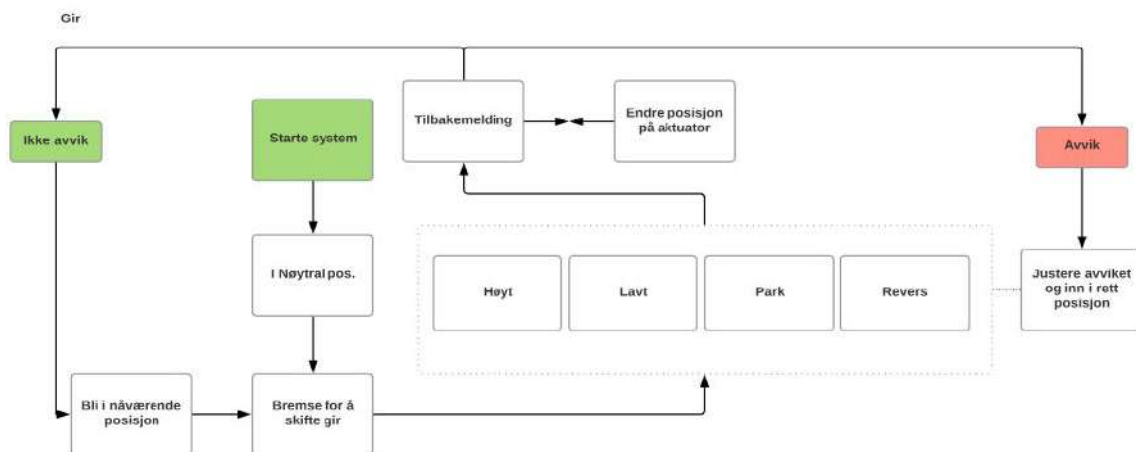
Figur 96: UML - Gass

Brems:



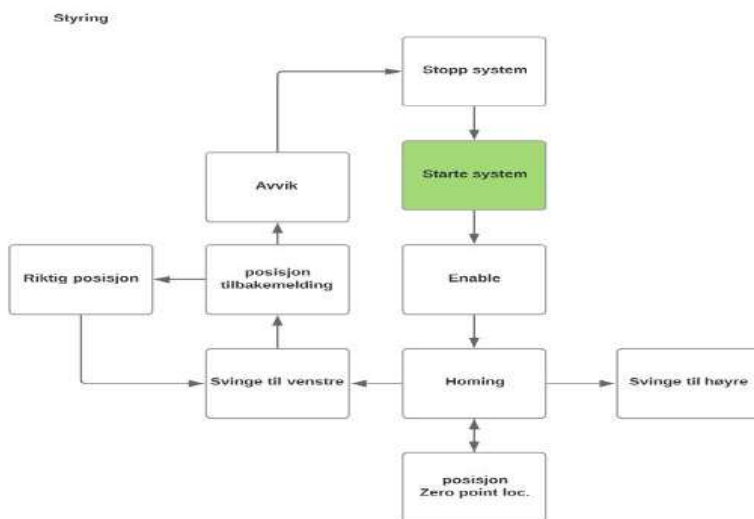
Figur 97: UML - Brems

Gir:



Figur 98: UML - Gir

Styring:

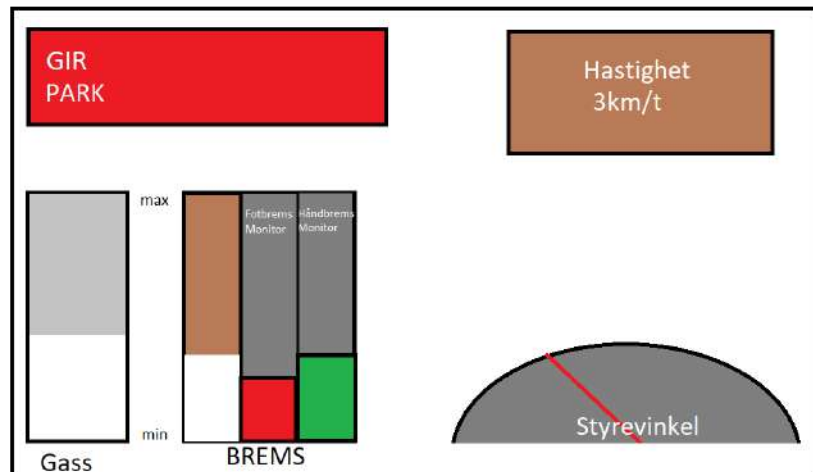


Figur 99: UML - Styring

UML kartene er satt opp i Lucidchart [41].

11.7.5 Interface

For å skape en god brukeropplevelse med kjøring av ATV-en skal det lages ett brukergrensesnitt. Dette skal gi en oversikt over de forskjellige parameterne. I figur 102 er det vist hvordan dette kan bli satt opp. Her vil brukeren enkelt kunne se hvilket gir, hastighet, hvor mye gass som blir gitt eller hvor kraftig det blir bremsset og hvilken vinkel ATV-en styrer med.



Figur 100: 1.Utkast til brukergrensesnitt

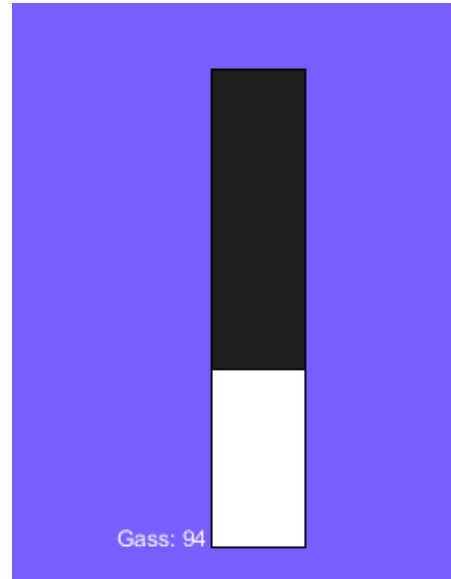
Bremsemonitoreringene vil vise hvilken feilverdi aktuatorene har. Etersom bremse pådraget fra Arduinoen går såpass fort, vil aktuatoren bruke noe tid på å nå verdiene brukeren ønsker. Ved å sette opp to søyler til dette, vil man få en oversikt over aktuatorene til håndbrems og fotbrems. Dersom man skifter gir må man alltid ha begge bremsene helt inne, ved å vise dette i søylene vet brukeren hvordan systemet ligger ann i prosessen.

Visning av variabel trigger

Ved å lage to firkanter med rect funksjonen [48], med forskjellig farge, blir en dynamisk søyle laget. Det hvite i søylen gjenspeiler hvor mye R2 på kontrolleren blir trykket inn. I tillegg til dette vises også hvilken tallverdi R2 har til venstre for søylen. I figur 101 vises ett tall mellom 0 og 255 som maks, da dette er verdiene til Arduinoen.

Senere vil det kodes om slik at det blir vist en % verdi isteden.

Det vil også bli vist en grense på hvor mye brukeren skal tillates å gasse. Maks gasspådrag skal være lite under prosjektperioden ettersom hastigheten ikke skal overskride 7km/t. Dette skal være en variabel som skal kunne skiftes senere, når høyere hastighet ønskes. Koden under viser ett eksempel på hvordan dette kan gjøres. Denne bruler de negative verdiene til R2 til å lage en firkant som stiger i forhold til hvor langt triggeren er trykket inn.



Figur 101: Gass grensesnitt i Processing

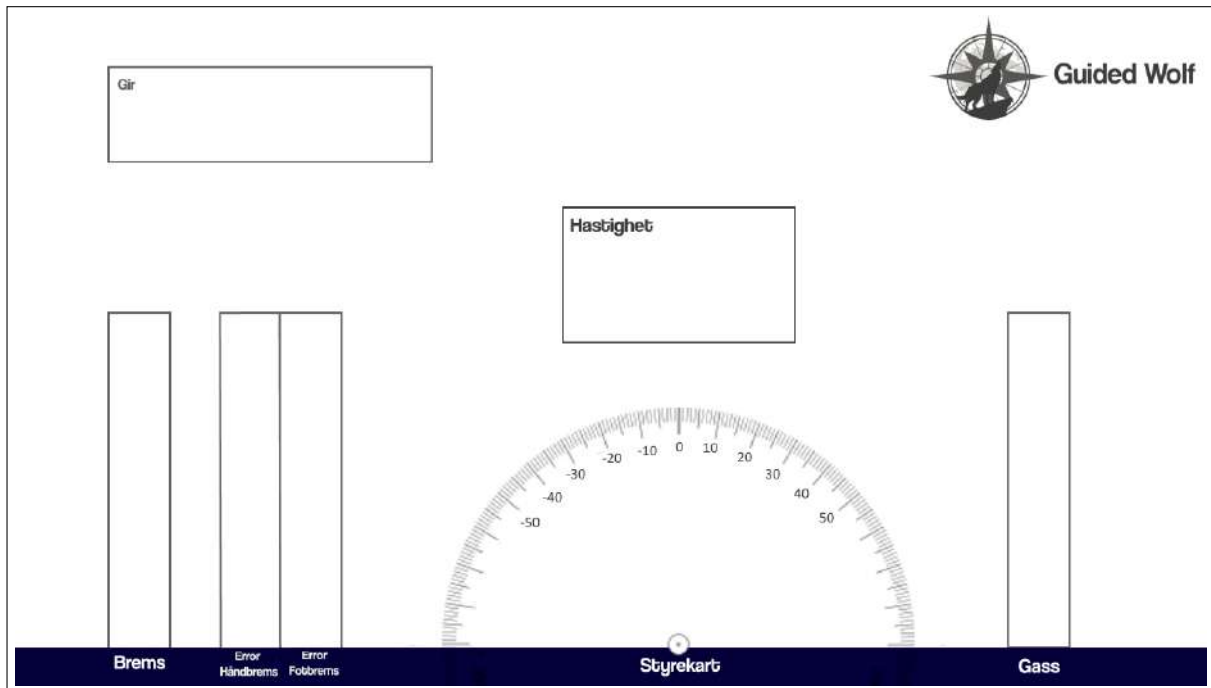
```

fill(30,30,30);
rect(150,550,50,-255);
fill(255,255,255);
rect(150, 550, 50,-R2);
text(int(R2), 100, 550);

```

Andre iterasjon

Etter første test av brukergrensesnittet (TP.6-O07) ble det endret plass på gass og brems søylene. Det ble også laget et penere bakgrunnsbilde som kan brukes i processing. Dette ble laget i paint.net, som er et gratis bildebehandlingsverktøy. Dette kan gi en nøyaktig plassering av elementene som skal brukes. Dette kan gjøres ved å se på pixlene i X og Y retning i paint.net og bruke de samme verdiene i processing.



Figur 102: 2.Utkast til brukergrensesnitt

Ved å bruke paint.net får man flere muligheter til å designe brukergrensesnittet. Gradeskiven ble hentet fra [25]. Denne blir brukt sammen med en funksjon i processing, for å vise gradene som svinges. Trigonometri blir brukt for å finne hvilken vinkel en linje skal stå i i forhold til hvor mye venstre styrespake (LX) blir dyttet til siden. Linjen er satt i origo til gradeskiven, i piksel verdier blir dette 710 i X-retning og 680 i Y-retning. LX har blitt satt til å være en verdi mellom -140 og -40 grader, slik at graden er 0, dersom LX står i 0 grader dersom den er i midten. For å bruke radianer i processing ble radians() brukt som vist under og på processing sine sider. [49] Dette sammen med en radius verdi på 250 gir en linje som viser vinkelutfallet.

```
void vinkelstyring(){
  //-- for å lage vinkelstyring --//
  int radius = 250;
  float x = cos(radians(LX))*radius;
  float y = sin(radians(LX))*radius;
  strokeWeight(5);
  line(710, 680, x+710, y+680);}

```

For å vise hvilket gir ATV-en er i, sendes informasjonen via en string med navnet til

giret til girInterface samtidig som ATV-en sender til PID-kontrolleren. F.eks om ATV-en står i park, vil det bli sendt girInterface("Park"); for å vise i interfacen at ATV-en står i park.

```
void girInterface(String hvilketGir){  
  /-- For å vise hvilket gir --//  
  textSize(50);  
  text(hvilketGir, 110,120);  
  textSize(12);} 
```

11.7.6 Analog/Digital - Pin i Processing

Ett problem som ble støtt på tidlig i utviklingen var at Processing ikke håndterer pinne-nummerering på samme måte som Arduino. I Arduino skriver man slik for å deklare pinnenummer:

```
pinMode(A0, INPUT); //For å deklare pinne A0 som input
```

Denne måten å skrive på byr på problemer for Processing da den forventer ett heltall i denne deklareringsen og vil derfor ikke tillate A0. Det ble derfor sett på alternative måter å løse dette på. Dersom man ser på Atmega2560 pinmapping fra [46] kan man se at alle pinnene har ett nummer. Det ble forsøkt å bruke disse nummerne til å deklare uten hell. Etter dette ble det sett på alternative måter å sende informasjon fra Arduinoen til Processing. En av disse er å bruke Serial.println funksjonen i Arduino som sender over informasjon via serial porten. Ett eksempel på dette viste electronicshobbyist [4]. Problemet med dette eksempelet er at det blir veldig begrenset og kan kun bruke en analog input. Derfor er ikke dette egnet til dette prosjektet.

Instructables [52] viser hvordan dette kan gjøres. Dette eksempelet ble funnet tidlig, men ble oversett da dette ikke var logisk i forhold til hvordan modellen til presentasjonen ble kodet opp. Det viste seg senere at dette var måten det skal gjøres på. I Processing skal man skrive slik for å deklare pinnenummer:

```
pinMode(0, INPUT); //For å deklare pinne A0 som input  
pinMode(0, INPUT); //For å deklare pinne 0 som input
```

Grunnen til at dette ble avskrevet er at man kan skrive på samme måte for å deklare pinnenummer 0 og A0. Processing velger hvilket pinnenummer som skal brukes ut ifra

om man skriver `analogRead` eller `digitalRead`. Dersom man bruker analog vil det si A0, mens digital vil si 0. Dette vil være viktig å huske på under videre koding av prosjektet for å unngå feil.

Tilfeldige verdier på pinner Under testing ble det oppdaget at Processing tar imot forskjellige verdier også på de pinnene som ikke er i bruk. Dersom pinne A0 er koblet til ett potmeter, vil også A1 motta de verdier som blir påvirket av dette. Dette gjør det ekstra viktig å dobbeltsjekke pinnetilkoblingene, da man kan få ut verdier som kan virke riktige ved en feilkobling [22].

11.7.7 Oppstartsfunksjon

For å sjekke at de to viktigste aktuatorene fungerer slik de skal før kjøring av ATV-en, er det laget en oppstartsfunksjon som gjør at operatøren kan teste om begge bremseaktuatorene kan bevege seg til endepunktene og at de fungerer slik de skal før de andre funksjonene er aktiv.

I draw-funksjonen i Processing som er lik loop i Arduino blir oppstart funksjonen kalt. Den er aktivert ved oppstart ved at den globale variabelen `oppstart` er "true".

```
if(oppstart == true) {  
  brukerGrenseSnitt();  
  getUserInput();  
  startUp();  
}
```

Figur 103: Kaller oppstartsfunksjonen i draw

I "startUp" som er funksjonen som blir kalt kan man styre bremseaktuatorene som vanlig og operatøren kan se at de fungerer, for så å trykke trekant slik at oppstartsmodus avsluttes.

```

void startUp() {

    boolean R = false;

    if (R == false) {

        image(GWoppstart, 0, 0);
        getUserInput();

        float mapHand = map(L2, 0, 255, 920, 100);
        float mapFot = map(L2, 0, 255, 100, 920);

        double handbremsKontroll = PID_kontroller((int)mapHand, handBremsPID);
        aktuatorKjor(handbremsFrem, handbremsBak, handbremsPin, handbremsKontroll);

        double fotbremsKontroll = PID_kontroller((int)mapFot, fotBremsPID);
        aktuatorKjor(fotbremsFrem, fotbremsBak, fotbremsPin, fotbremsKontroll);
    }
    if (trekant == true) { R = true; oppstart = false; } //Trykker trekant for å gå ut av oppstart
}

```

Figur 104: Oppstartsfunksjonen

Denne funksjonen kan ikke avbrytes av andre funksjoner og kun L2 og trekant er gyldig input fra PS4-kontrolleren.

11.7.8 PID Kontroller

Udemy har en gjennomgang på hvordan man kan kode opp en PID kontroller til Arduino [69]. Koden under er kopiert herfra, og denne ble fulgt for å få grunnlaget riktig til å bygge koden videre. Denne starter med å sette globale variabler som trengs til kontrolleren.

```

double sensed_output, control_signal;
double setpoint;
double Kp; //proportional gain
double Ki; //integral gain
double Kd; //derivative gain
int T; //sample time in milliseconds (ms)
unsigned long last_time;
double total_error, last_error;

```

Dette er alle variablene som trengs, hvor K_p , K_i og K_d er PID parameterne. Control_signal er det signalet PID kontrolleren gir ut til slutt for å styre aktuatorene. For å bruke anti-windup skal max og min_ controll settes til ± 255 . Dette er maksverdier av hva Arduinoen kontrollerer med, og vil også si noe om hvilken retning aktuatoren vil gå.

```

unsigned long current_time = millis(); //returns the number of
↳ milliseconds passed since the Arduino started running the
↳ program

int delta_time = current_time - last_time; //delta time interval

if (delta_time >= T){ //Dersom delta_time er større enn sample
↳ time

    double error = setpoint - sensed_output; //lager ny variabel med
↳ dit PID kontrolleren skal målt opp mot sensoren.

    total_error += error; //accumalates the error - integral term
    if (total_error >= max_control) total_error = max_control; //
↳ Anti-windup
    else if (total_error <= min_control) total_error = min_control;

    double delta_error = error - last_error; //difference of error for
↳ derivative term

    control_signal = Kp*error + (Ki*T)*total_error + (Kd/T)*delta_error;
↳ //PID control compute
    if (control_signal >= max_control) control_signal = max_control;
↳ // Anti-windup
    else if (control_signal <= min_control) control_signal =
↳ min_control;

    last_error = error;
    last_time = current_time;

```

Denne koden fra Udemy ble testet i test TP.1-GI.02. Etter fullført test ble koden tilpasset bruk til prosjektet. Det trengs flere PID kontrollere, så bruk av globale variabler på denne måten vil gjøre det vanskelig å tilpasse hver enkelt. En måte å jobbe rundt dette er å bruke arrays isteden. Det vil gjøre at det vil bli enkelt å finne igjen de ulike parameterne. Her vil de ulike verdiene bli tilpasset etter hvert. Her er det vist hvordan de globale

variablene ser ut til de forskjellige PID kontrollene. Ved å skrive som under har det blitt satt opp 3 forskjellige kontrollere, en til gir, samt en til håndbrems og fotbrems.

```
//hvilken PID kontroller som skal brukes. Verdier settes under.
int girPID = 0;
int handBremsPID = 1;
int fotBremsPID = 2;

//hvilke verdier i PID kontrollen: rekkefølge Gir - håndbrems -
↳ fotbrems
float[] kpA = {1,1,1};
float[] kiA = {0,0,0};
float[] kdA = {0,0,0};
```

Itillegg til disse endringene ble det også endret funksjonen, slik at det ble sent inn hvilken setpoint kontrollen skal oppnå, samt hvilken PID kontroller som skal brukes. Resten av funksjonen er ganske lik, bortsett fra at det henvises til array posisjonene til variablene.

```
double PID_kontroller(int setpoint, int hvilkenPID){ //setpoint =
↳ Hvilken verdi sensor_output skal nå.

    float kp = kpA[hvilkenPID];
    float ki = kiA[hvilkenPID];
    float kd = kdA[hvilkenPID];
```

Feilmargin

Fra test TP.23-GI.01 ble det oppdaget at aktuatorene muligens er for trege til å bruke PID-kontrollere uten feilmargin. Det ble derfor lagt inn en feilmargin slik at aktuatorene kunne stoppe i nærheten av riktig verdi.

```
if(kontroll_signal<20 && kontroll_signal>-20){
    arduino.digitalWrite(motorFRAM, 1);
    arduino.digitalWrite(motorBAK, 0); // (motorBAK, 1)
    arduino.analogWrite(motorPin, 255);
}
```

Fra testing så man at ca 30 i feilmargin var det minste den kunne ha før aktuatoren begynte å oscillere rundt punktet. Derfor ble feilmarginen i koden satt til litt til, slik at den har noe å gå på.

Ettersom denne metoden ble brukt ble det ikke nødvendig å bruke integral og derivasjons kontrollere i tillegg. Disse to kontrollerne eliminerer steady-state erroren, men steady-state får lov til å være en verdi på 40, og aktuatoren stopper dersom den er mindre enn dette. Det vil si at det kun er en P kontroller som blir brukt, men koden er fortsatt tilgjengelig til å bruke I og D kontrollerne.

11.7.9 Hastighetsmåling

For å forsikre at ATV-en ikke overskrider satt hastighet må det også gjøres målinger. Det finnes forskjellige metoder å gjøre dette på. Det beste ville vært å hatt muligheten til å koble seg på det eksisterende systemet på ATV-en. Dette ble ansett som meget vanskelig med begrenset adgang til ATV-en da det måtte ha blitt kartlagt hvordan sensorene på ATV-en måler fart og hvordan de er koblet. Det ble derfor bestemt å gjøre dette på en enklere måte, som dog blir noe unøyaktig. Ved å bruke en Infrarød lyskilde med mottaker kan det måles hver gang en eike passerer. Dersom det blir målt hver gang en eike passerer vil det bli 8 målepunkter som vist under 105. Ettersom eikene ikke er helt like, vil det bli noe feilmargin, men grovt sett vil det skje en måling hver 45° , som de røde prikkene på bildet illustrerer. En annen kilde til feilmargin er diameteren til hjulet, som ble målt til 66cm. med en cm forskjell vil omkretsen variere med ca 3 cm.



Figur 105: Målepunkter på framhjul

For å finne hastigheten med målepunktene brukes vinkelhastigheten sammen med omkretsen som vist under. Ved å først finne gradene i radianer slik:

$$45^\circ \frac{\pi}{180} = \frac{\pi}{4} \quad (25)$$

Da kan ΔT altså tid mellom hvert målepunkt, brukes til å finne vinkelhastigheten:

$$\omega = \frac{\frac{\pi}{4}}{\Delta T} \quad (26)$$

For å finne farten i m/s kan man da ta

$$v = \omega \times 0.33 \quad (27)$$

```

if (sensorAvlesning > 150 && hastighetState == false) {

    hastighetState = true;
    deltaSpeed = (millis()-sistEndretHastighet);
    sistEndretHastighet = millis();

    float radianVinkel = (PI/4);    // 45 grader = pi/4 i radianer

```

```

float vinkelHastighet = radianVinkel/deltaSpeed*1000;
float Fart = vinkelHastighet*radiusDekk;
float FartKmt = Fart*3.6;

fartsnitt(FartKmt);
}
//mindre enn 150
if (sensorAvlesning<150 && hastighetState==true) {
    hastighetState=false;
}

```

Dersom IR-sensoren har passert en eike, vil sensorAvlesning bli høyere enn 150. For å lese av hver gang den passerer en eike brukes en state, slik at kun lager en ny deltaSpeed når den endres. Slik vil det forekomme en måling på hvert av målepunktene over. Funksjonen fartsnitt(FartKmt) sørger for at det vises ett gjennomsnitt av de to siste verdiene. Dette gjør at målingen blir noe mer korrekt.

Dette vil ikke dette være en veldig god metode på å måle hastigheten på ATV-en, men tilstrekkelig under prosjektet for å sørge for at ATV-en ikke overskrider makshastigheten med for mye. Til videre utvikling burde det ses på andre metoder som GPS måling og accelerometer. En annen mulighet er å bruke kamera som plasseres på displayet på ATV-en. Dette kan brukes til å prosessere bildet og få ut verdiene fra systemet.

11.7.10 Gass

Etter målingene ble det gjort tester på å gjøre om PWM-signalene fra arduinoen til analoge verdier. Ettersom den ene gass utgangen ga 3V og den andre 1.5V, satte ble 150 satt som maks pådrag. Ved å mappe disse verdiene kan man bruke hele den analoge knappen (R2) til pådraget.

```

void gass() {
    //12-75 max verdi på pinne 1: 25-150 på pinne 2
    float gass1 = map(R2, 0, 255, 12, 75);
    float gass2 = map(R2, 0, 255, 25, 150);

    arduino.analogWrite(gass1Pin, int(gass1));
}

```



```
arduino.analogWrite(gass2Pin, int(gass2));  
  
// For å stoppe  
if (gjennomsnitt >= 6) {  
    arduino.analogWrite(gass1Pin, 0);  
    arduino.analogWrite(gass2Pin, 0);  
}  
}
```

Mekanisk design

I denne delen vil alle mekaniske design bli gjennomgått. Hvilke krav det er til hvert enkelt delsystem og komponent, Hvordan disse ble designet og sluttresultatet

Innhold:

- Design av ramme
- Design av håndbrems
- Design av fotbrems
- Design av girssystemet
- Design av styret



KONGSBERG



USN Universitetet
i Sørøst-Norge

12.1 Design av ramme

Ved de første gjennomgangene av kravene til rammen, skulle rammen kun være et stativ til våpenstasjonen som skal monteres på ATV-en. For å utnytte potensialet til en så stor struktur på ATV-en, og for å løse problemet med innfestning av de andre delsystemene, ble det fastsatt at rammen også skal være innfestning for alle de eksisterende delsystemene.

Systemet skal videreutvikles under prosjekter som vil gå over flere år, derfor er det rimelig å anta at det vil bli designet flere delsystemer ettersom behovene oppstår, derfor må rammen også være dimensjonert for å kunne tåle ytterligere påkjenninger enn hva det er behov for på dette tidspunktet.

Det er rimelig å anta at våpenstasjonen vil være den største påkjenningen for rammen, ettersom den tilfører en stor masse på mellom 100 og 150 kg, med et massesenter høyt oppe. Under gassing, sving og bremsing vil dette føre til et høyt bøyemoment i innfestningen til våpenstasjonen, og derfra i innfestningen til rammen.

De andre eksisterende delsystemene vil derimot ha lavere masse, med massesenter nærmere festepunktene på rammen, samt monteres lavere, slik at alle innfestinger vil få lavere belastning. Av denne grunn må rammen hovedsakelig dimensjoneres for å tåle belastningene våpenstasjonen og de andre delsystemene som tilfører høye krefter. Deretter må rammen dimensjoneres med en sikkerhetsfaktor høy nok for å kunne tåle andre tenkelige belastninger som kommer av ettermontering av andre delsystemer fremover i tid.

12.1.1 Krav som må møtes

Våpenstasjon	VS.01	En våpenstasjon skal kunne festes enkelt på ATV
	VS.01.01	Våpenstasjonen må ikke gjøre ATV ustabil
	VS.02	MPU - Main Processing Unit må sitte på ATV
	VS.03	Våpenstasjon må ha en egen strømkilde
	VS.04	DCP - Display Control Panel må sitte på ATV

Figur 106: Krav i forbindelse med våpenstasjonen

- **Ingen permanente endringer på ATV**

Dette kravet ble stilt til rammen, på bakgrunn av at det ikke skal gjøres permanente endringer på ATV-en, og derfor må rammen monteres via eksisterende festepunkter på ATV-en. Dette påvirker rammens størrelse og design i stor grad, ettersom noen festepunkter er plassert på ugunstige steder.

Måten å løse dette på vil være å bruke eksisterende festepunkter i form av både hull og gjengepartier som allerede er på ATV-en. Ved å bruke festepunkter fra forskjellige steder på ATV-en, vil det være mulig å designe en innfestning som vil skape et godt grunnlag for en stiv og sterk ramme, og motarbeide de store kreftene våpenstasjonen vil påføre, med minst mulig påvirkning på ATV-ens festepunkter.

- **Våpenstasjonen må kunne monteres enkelt på ATV-en**

Dette kravet ble stilt til rammen, for å sørge for at våpenstasjonens innfestning ikke skal måtte endres ved montering på ATV-en, men enkelt kunne monteres slik den er, og kunne demonteres igjen uten store operasjoner.

Måten å løse dette på er å bruke boltsirkelen i basen til våpenstasjonen, og enten monteres med tilhørende bolter til rammen, eller via et adapter som kan sveises i rammen. Den sistnevnte metoden vil nok være den mest aktuelle metoden, ettersom materialet rundt bolthullene er nødt til å ha god tykkelse. Det vil derfor kreve mindre av veggtykkelsen til rørene rammen konstrueres av, om et adapter med god nok tykkelse sveises til rammen.

- **Våpenstasjonen må ikke gjøre ATV-en ustabil**

Dette kravet ble stilt til rammen, ettersom dette er et tungt objekt, som med sitt høye massesenter kan påvirke ATV-ens kjøreegenskaper kraftig. Derfor må våpenstasjonen kunne monteres på ATV-en slik at den forandrer ATV-ens massesenter og kjøreegenskaper minst mulig. Av den grunn vil det være ønskelig å montere våpenstasjonen så lavt som mulig, allikevel må det stå på toppen av ATV-en, ettersom den skal ha fri sikt i 360 grader.

Måten å løse dette på er derfor å montere den på toppen, så nærme ATV-en som mulig i høyderetning, men med nok plass til å kunne montere den med bolter fra undersiden. Dette vil føre til det laveste tyngdepunktet mulig, selv om det fortsatt vil påvirke kjøreegenskapene til ATV-en betydelig. Det andre kriteriet for en stabil ATV vil være å plassere den så godt som mulig i senter mellom hjulene på ATV-en, slik at den ikke blir ustabil sidelengs i svinger, eller ustabil i lengderetning ved gassing eller oppbremsing.

- **Komponenter til våpenstasjonen må sitte på ATV-en**

Dette kravet ble stilt til rammen, ettersom ATV-en skal være «trådløs». Derfor må alle komponenter som våpenstasjonen trenger for å fungere, være montert på ATV-en. Dette innebærer en større strømkilde og en styreenhet.

Måten å løse dette på vil være av samme type som montering av selve våpenstasjonen, altså monteringsmetoden de forskjellige komponentene opprinnelig er designet for burde brukes. Det vil si at styreenheten monteres med bolter, enten rett i rammen eller med adaptere om plasseringen til bolthullene ikke treffer rammen. Når det gjelder strømkilde vil det mest sannsynlig bli brukt et batteri, som da trenger en form for brett å stå på, med mulighet for å spenne fast med en form for «klikk inn» løsning, eller en stropp.

12.1.2 Hva finnes fra før

En bærende struktur/ramme er som regel laget av rør eller bjelker, ettersom de ofte skal bære en last som skaper bøyemoment i strukturen. Et eksempel på dette er et gulv, hvor man har vertikale bjelker på hver side, og horisontale bjelker som ligger mellom disse bjelkene, hvor så gulvet blir lagt på disse. Når man står på midten av dette gulvet, vil dette føre til høye spenninger i enden av de horisontale bjelkene, som resultat av bøyemomentet.

I gulv med større spenn er det normalt å bruke stålbjelker eller rør. Disse har høy motstand mot bøyning på grunn av sin høye styrke, og plassering av gods, som H-bjelker eller firkantrør som har høy motstand mot bøyning ettersom Godset er plassert med god avstand i retningen bjelken mottar kraft.

På grunn av festepunktene lokasjon kan et forestilt design likne på et veltebur til biler, med et rektangel av rør øverst, rør nedover til hver side, og rør bakover som støtter rektangelet. Et veltebur er også designet som en stiv struktur, med høy styrke, noe som også vil være en viktig faktor for rammen som skal designes.

Et veltebur er laget i runde rør, som har meget høy motstandsdyktighet mot bøyepening, noe våpenstasjonen, som nevnt, vil skape mye av under kjøring. Derfor er det lønnsomt å se nærmere på, og ta inspirasjon fra en slik struktur som et veltebur.



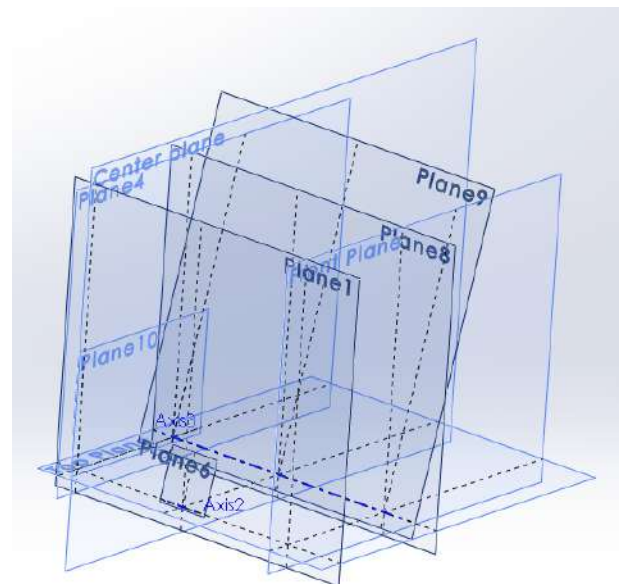
Figur 107: Horisontal bjelke med en last holdes oppe av en vertikal bjelke [53]



Figur 108: Veltebur fra en BMW E36 Coupé 1990-2000 [71]

12.1.3 Design

Designprosessen starter ved å legge inn de oppmålte festepunktene på ATV-en i SolidWorks. Dette er da originalt festepunktene for fothvilerne til føreren, fothvilerne til passasjer og festepunkter på toppen av ATV-en. Deretter tegnes det en veldig forenklet modell av ATV-ens motor og ramme, som rammen må gå rundt/ikke treffe. Etersom festepunktene på ATV-en ikke ligger i samme plan, og derfor ikke være gode nok for direkte montering av rammen, starter prosessen med å designe adaptere som gjør det mulig å feste rammen til disse festepunktene.

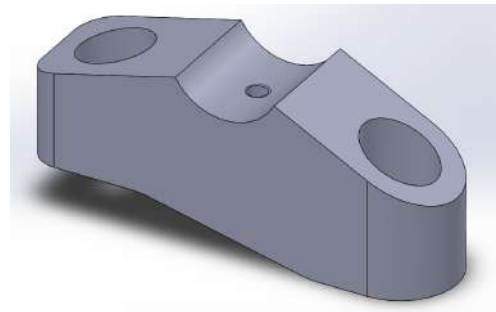


Figur 109: Oversikt over monteringsplan på ATV-en

Første konsept

Det ble tatt en avgjørelse om å designe en ramme i runde rør, ettersom disse gir den beste motstanden mot bøyemoment i forhold til størrelse. Designet starter med å designe overgangene til innfestingen på ATV-en, for å lage et grunnlag for rammen, for å så designe selve rammen og innfestingen til de andre delsystemene.

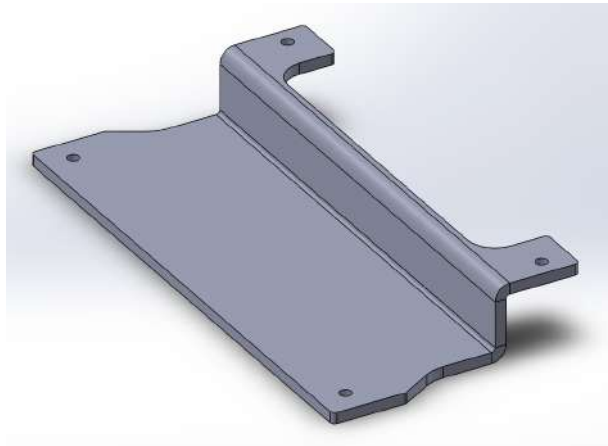
Festepunktene som opprinnelig blir brukt til innfesting av fothvilerne til føreren og de øvre festepunktene er blitt oppmålt, slik at det er mulig å designe overganger til rammen. Ettersom disse bolthullene ikke ligger i samme plan, må overgangene designes slik at de får et felles plan rammen kan sveises til. Overgangene der fothvilerne originalt er montert ble designet slik at nedre del av overgangen passer på ATV-en, og øvre del har et spor rammen kan bli sveiset fast til. For å feste disse til ATV-en ble det designet gjengepartier, som passer med bolthullene i ATV-en, slik at bolter kan føres gjennom kullene i ATV-en og skrues fast i overgangene.



Figur 110: fothvilerovergang for ramme laget av runde rør, med montering til ATV-en fra toppen

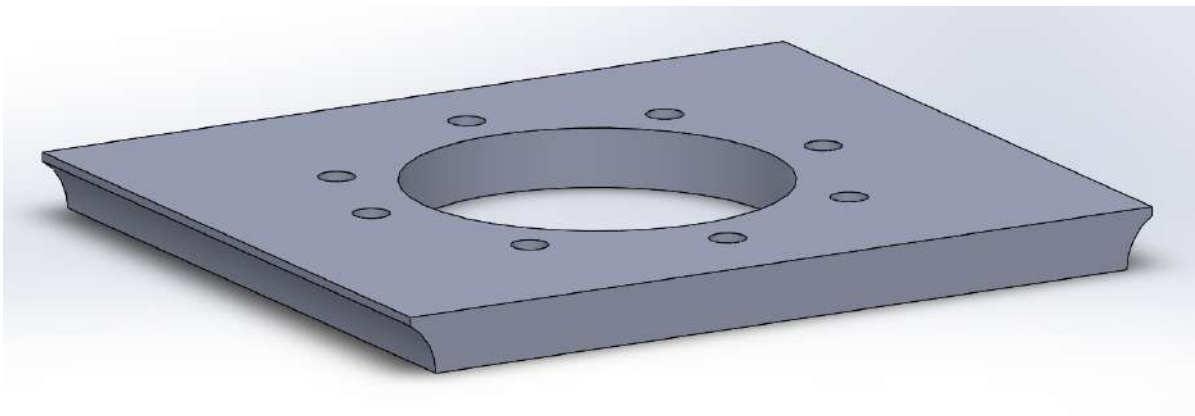
De øvre festepunktene består av gjengede hull, så designet blir motsatt av fothvilerovergangene. Disse fire hullene består av et hullmønster, hvor to av hullene er i samme plan, mens de to andre sitter høyere opp. Siden hullene er plassert med så stort mellomrom, trengs det en veldig stor del for å dekke alle hullene, dette gjør at det ble nødvendig å komme opp med et design hvor man slipper å maskinere en så stor del.

En annen måte å gjøre det på er å bruke en plate man knekker opp for å kompensere for høydeforskjellen, for å så bore ut hull som passer til gjengepartiene. Derfor ble et slikt design brukt, med tanke på vannskjæring av formen på platen og hullmønsteret, for å så knekke opp platen, slik at den passer til alle fire hullene. Dette gir en funksjonell og billig løsning.



Figur 111: Festebrakett for de øvre feste-punktene på ATV-en

Rammen må designes med tanke på plassering av våpenstasjonen, og hva den krever av innfesting. Derfor vil det være lønnsomt å designe delen som våpenstasjonen skal monteres på, for å så designe rammen etter det. Ved å designe en plate som kan vannskjæres til riktig form med hullmønsteret, for å så knekke platen for å kompensere for høydeforskjellen på ATV-en. I stedet for å designe en del som må maskineres vil dette gjøre at delen vil bli mye mindre kostbar, og samtidig opprettholde funksjonaliteten. Av denne grunn designes denne delen for å kunne vannskjæres. Selv om denne delen vannskjæres, må den maskineres på sidene, for at den skal kunne passe på rammen laget av runde rør. Dette vil øke kostnaden av delen betydelig, men er nødvendig for passform.



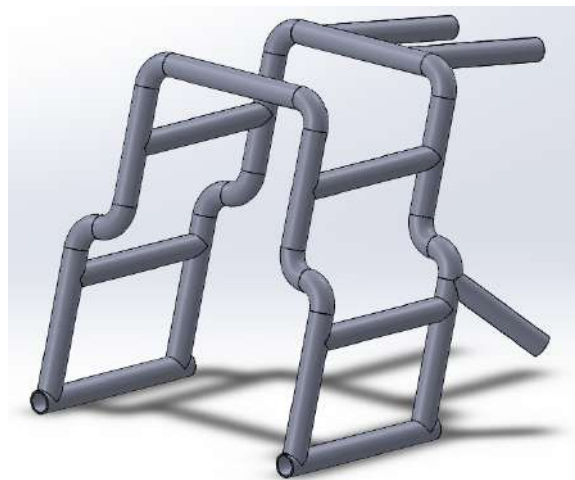
Figur 112: Monteringsplate for våpenstasjon, til rørramme med runde rør

Med overgangene på plass, gjenstår selve rammen. Ved å plassere alle de designede delene i SolidWorks vil det gi et godt fundament for å kunne designe rammen som skal forbinde disse. I dette konseptet ble rammen designet slik at det er et rør på hver side av ATV-en, som forbinder fothvilerovergangene. Deretter går det rør over ATV-en, som blir forbundet av monteringsplaten til våpenstasjonen. Deretter designes det rør bakover til festepunktene bak, som skal stive av, og motstå momentet våpenstasjonen påfører rammen.



Figur 113: Enkelt førsteutkast av rammen

For å gjøre designet av rammen mindre klumpete ble det designet en ny versjon, hvor rørene formet seg etter ATV-en, da med samme type rør. Dette designet er bygget på det første, men med et penere utseende og bedre funksjonalitet. Designet ble også videreutviklet med tanke på løsning av innfesting, hvor det øvre festepunktet på ATV-en også blir inkludert.



Figur 114: Videreutvikling av den første rørrammen, med forbedret funksjonalitet, penere design og endring av innfestning

Andre Konsept

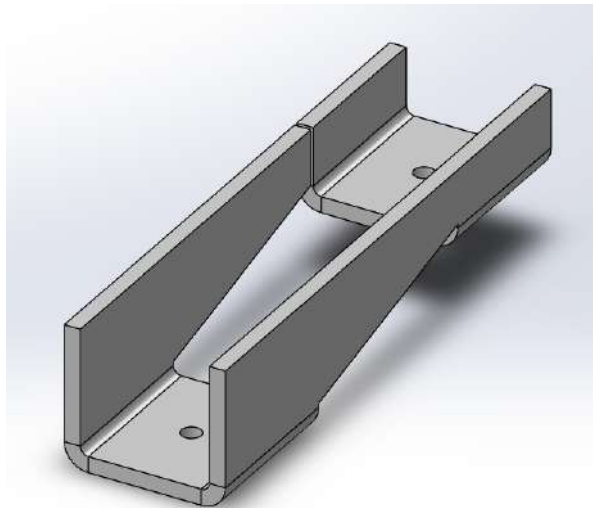
Etter en gjennomgang av første konsept, ble det godkjent av oppdragsgiver. Etter konseptet ble godkjent, ble det tatt kontakt med flere firmaer for sveising. Etter en gjennomgang av disse med oppdragsgiver ble det bestemt at Elor plate og sveis AS, v/Håkon Ringstad skulle få oppdraget, og det ble avtalt et møte for gjennomgang av første konseptet. Ringstad mente at designet ville bli for komplisert for en slik «prototype», og at det måtte forenkles.

Forslagene til endring

- Firkantrør istedenfor runde rør.
- Elor har dimensjoner fra 15 mm og oppover med 5 mm sprang, alle med 2mm veggtykkelse.
- Kappe rørene, og sveise sammen i vinkler, istedenfor å bøye rørene.
- Side rørene som går oppover burde være vinkelrette mot de horisontale, ikke skråe som i første konsept.

Ringstad anbefalte også at fothvilerovergangene burde designes på en annen måte, med tanke på at det tar lang tid å maskinere deler.

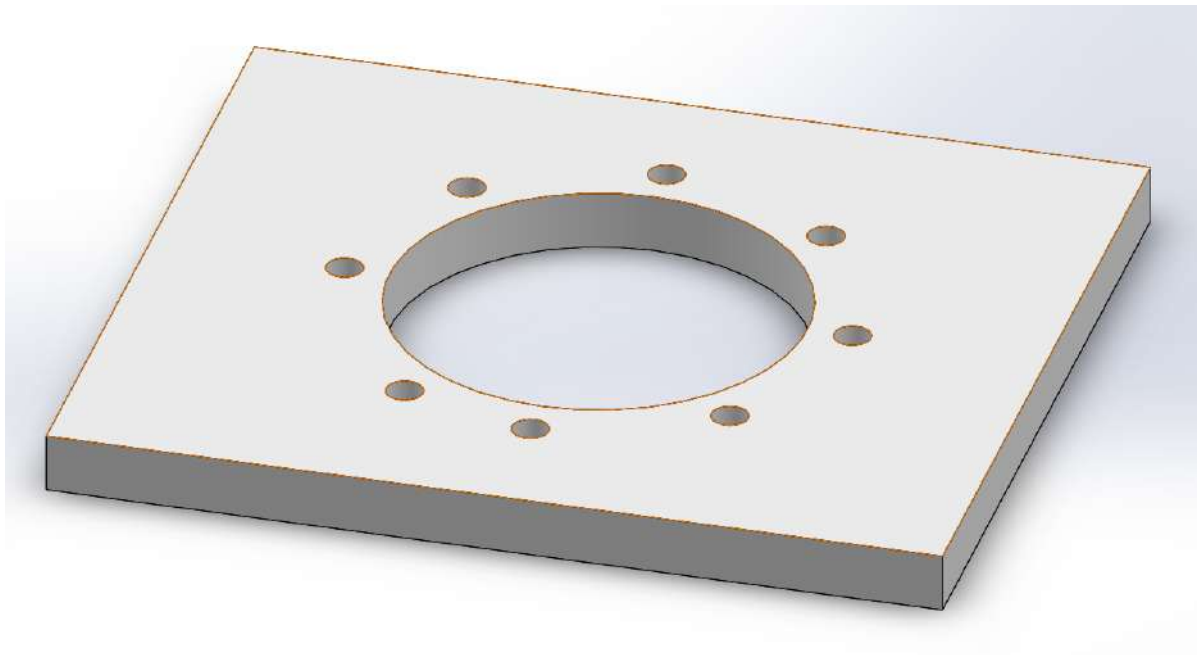
Tatt i betraktning råd fra Ringstad ble det avgjort at fothvilerovergangene skulle endres, slik at de skulle passe firkantrør istedenfor runde rør. Det ble også bestemt at produksjonsmetoden skulle endres, slik at de skulle bli billigere og raskere å produsere. Det ble derfor bestemt at disse også skulle designes slik at disse også kunne vannskjæres ut av en plate, for å så knekkes til riktig form. Det nye konseptet har lik funksjon som det forrige konseptet, men har fått en annen oppbygging og struktur.



Figur 115: Fothvilerovergang utformet slik at den kan knekkes ut av en plate

Overgangen til de øvre festepunktene var allerede designet som en plate i første konsept. Ettersom den ikke hadde noe spor eller tilpasning for runde rør, kan den forbli slik den er, og fortsatt passe med det nye konseptet.

Monteringsplaten til våpenstasjonen må også endres, slik at den skal passe firkantrør. Denne endringen er enkel å utføre, ettersom det kun er sporene til de runde rørene som må fjernes. Dermed ble designet enda enklere og billigere å produsere enn hva det var med runde rør, ettersom det nye konseptet ikke krever noe maskinering av spor til firkantrør.

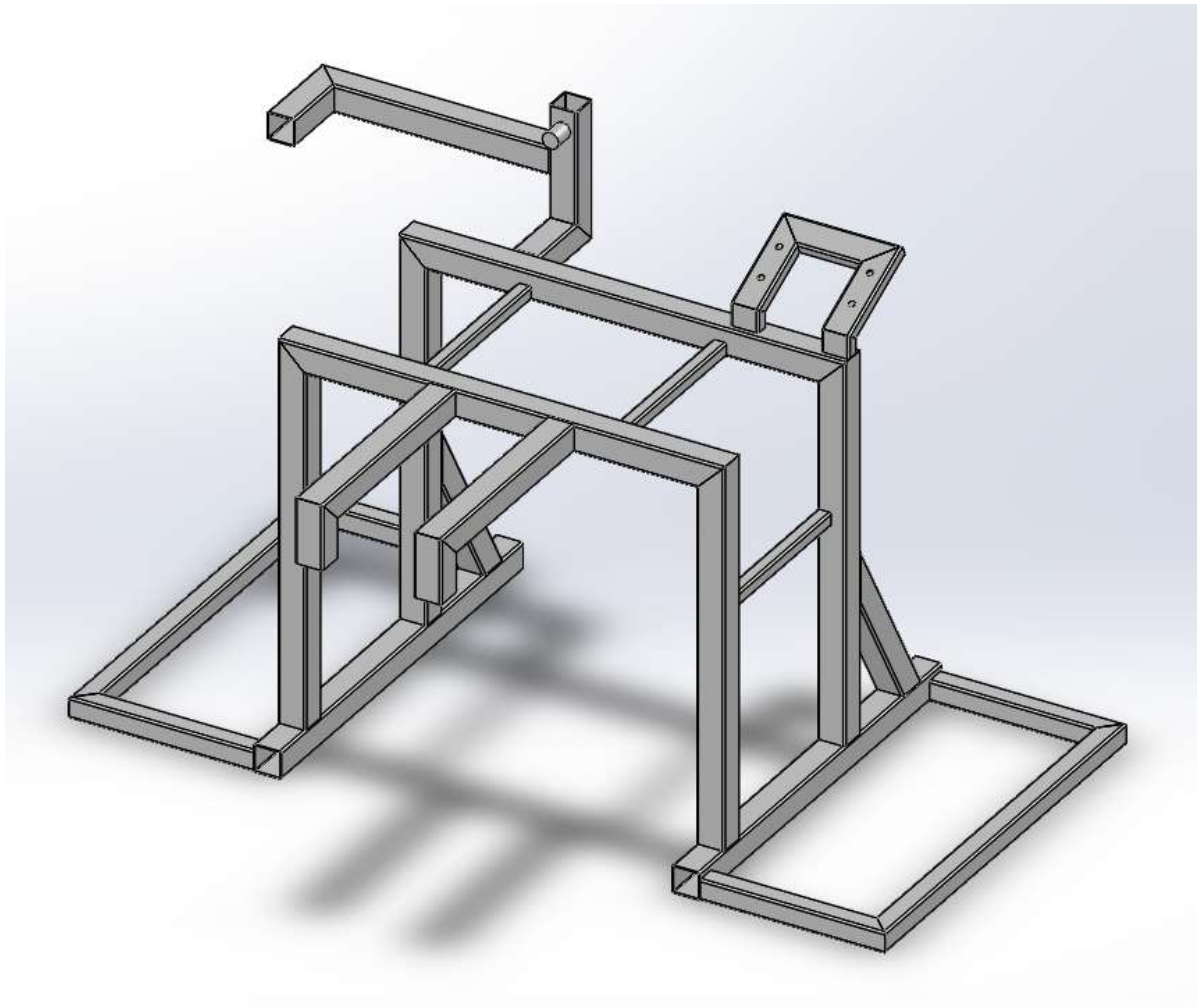


Figur 116: Monteringsplate for våpenstasjonen

Figur 116 viser monteringsplaten for våpenstasjonen, redesignet for bruk av ramme i firkantrør.

Profilen til rørene rammen består av endres til firkantrør, og de vertikale rørene rettes til å bli vinkelrette på de nedre, horisontale rørene som forbinder fothviler overgangene. Når dette var gjort ble alle hjørnene på rammen redesignet, slik at firkantrørene skal kunne kappes i vinkel og sveises sammen, istedenfor å måtte bøyes. For å gjøre designet enda lettere, og for å sørge for litt rom mellom ATV-en og rammen, i tilfelle det skal være noen komponenter som må sitte innenfor rammen, tett på ATV-en. Med nytt design klart, ble det igjen tatt kontakt med Ringstad for en gjennomgang, hvor designet ble godt motatt, og fikk klarsignal. Med godkjenning fra Ringstad, godkjente også opp-

dragsgiver dette designet.



Figur 117: Rammen designet i firkantrør

Figur 117 viser rammen designet i firkantrør, sett bakfra. Rammen inkluderer monteringsmulighet for håndbrems, med tilhørende aktuator, samt steppermotor for styring.

12.1.4 Oppsett av analyser

Utfyllende rapport med alle låsinger, kontakter, mesh og resultat ligger som vedlegg A.9

Etttersom konseptet var godkjent, måtte rammen dimensjoneres for å tåle påkjenningene den skal bli utsatt for. Først må alle låsinger settes opp, slik at SolidWorks vet hvordan de forskjellige delene ikke kan forflytte seg. Siden denne analysen skal kjøres på flere

delers samtidig, må SolidWorks vite hvordan, og hvor delene sitter sammen.

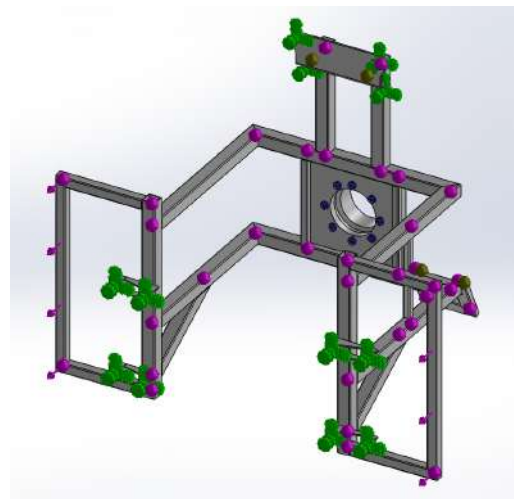
Dette gjøres ved å definere kontaktoverflater og kontaktpunkter, for å så definere om delene sitter fast sammen, eller om overflatene kan gli på hverandre. når dette er gjort, må alle ytre påkjenninger defineres i form av krefter, slik som:

- Vekten til våpenstasjonen
- Kreftene som blir påført som resultat av akselerasjon ved gassing, bremsing og sving
- Krav til vekt brettene på siden må tåle
- kreftene aktuatorene påfører rammen.

Når dette er gjort gjenstår det å definere et mesh, som SolidWorks bruker for å regne spenninger i delen. Et mesh i SolidWorks er når alle komponentene deles opp i små pyramideelementer med definert maks og minimum størrelse, med noder i hvert hjørne. Det er disse elementene SolidWorks bruker for å Kalkulere spenningene i delene, ved å se på forskjeller i krefter fra node til node.

Låsinger

Det er til sammen 12 kontaktpunkter som forbinder rammen til ATV-en med bolter. Disse må defineres, slik at SolidWorks forstår at delene ikke kan bevege seg i disse punktene. dette gjøres på en forenklet metode, ved å definere kontaktflaten mot ATV-en, for å så definere at disse flatene i dette tilfellet er låste i alle retninger.

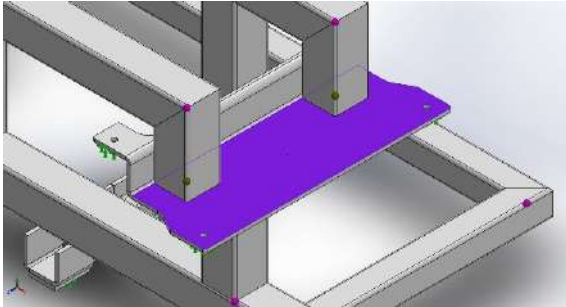


Figur 118: De grønne pilene symboliserer områder rammen er låst for bevegelse.

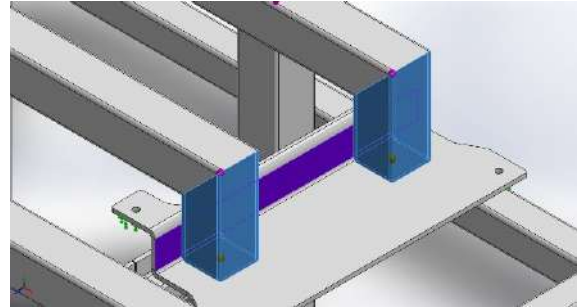
Kontaktflater

SolidWorks forstår ikke selv hvor og hvordan deler skal være festet sammen. For å definere dette velger man ut kontaktflater og kontaktpunkter som skal sitte fast sammen, eller skal kunne gli mot hverandre. dette gjøres ved å velge en flate eller et punkt i den ene delen, og en flate eller punkt i delen som skal sitte fast i den første delen, for å så definere at disse to flatene eller punktene skal sitte festet sammen.

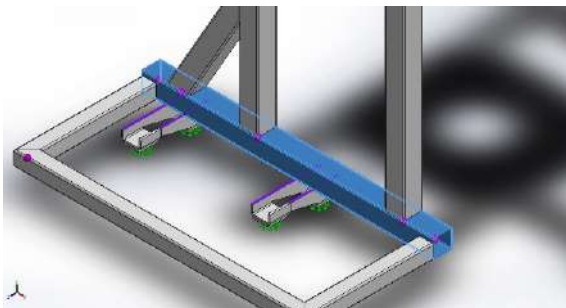
Et eksempel på et kontaktsett er at fothviler overgangene skal sveises fast i rammen. SolidWorks har en egen funksjon for sveising i FEM-analysene, men den type komponentkontakt bruker SolidWorks veldig lang tid på å regne ut, og siden det ikke er tilgang på kraftig nok PC for å klare dett innen leveringsfristen på denne oppgaven ble det gjort en forenklet kontakt, hvor kontakten ble definert som "bonded". Det vil si at de to flatene er definert som at de sitter fast i hverandre og at de ikke kan bevege seg i forhold til hverandre.



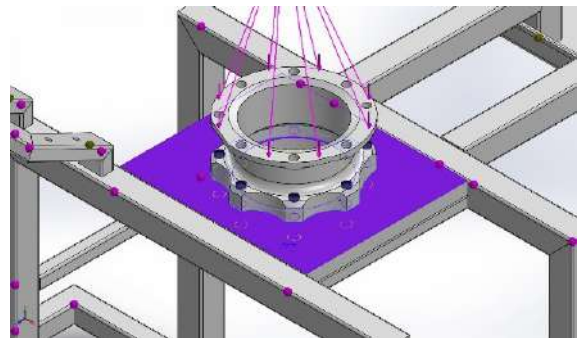
Figur 119: Her defineres kontakt mellom enden av røret, og overflaten på overgangen til de øvre festepunktene (i lilla).



Figur 120: Her defineres kontakt mellom røret (i blått), og overflaten på overgangen til de øvre festepunktene (i lilla).



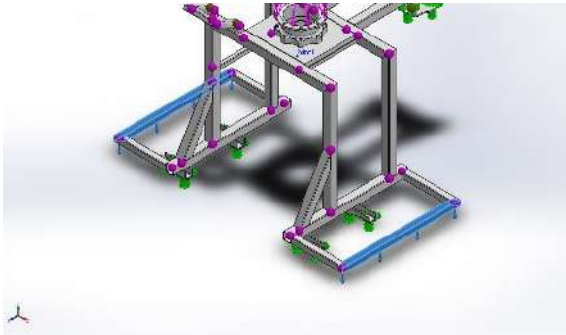
Figur 121: Her defineres kontakt mellom fothvilerovergangene (i lilla) og røret (i blått).



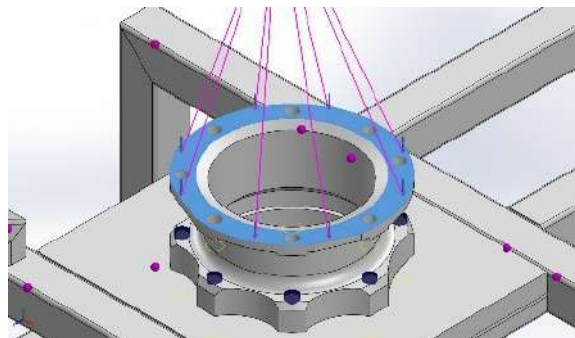
Figur 122: Her defineres kontakt mellom basen til våpenstasjonen og monteringsplaten til våpenstasjonen (i lilla), med frihet til å bevege seg i forhold til hverandre, men ikke gjennom hverandre.

Krefter

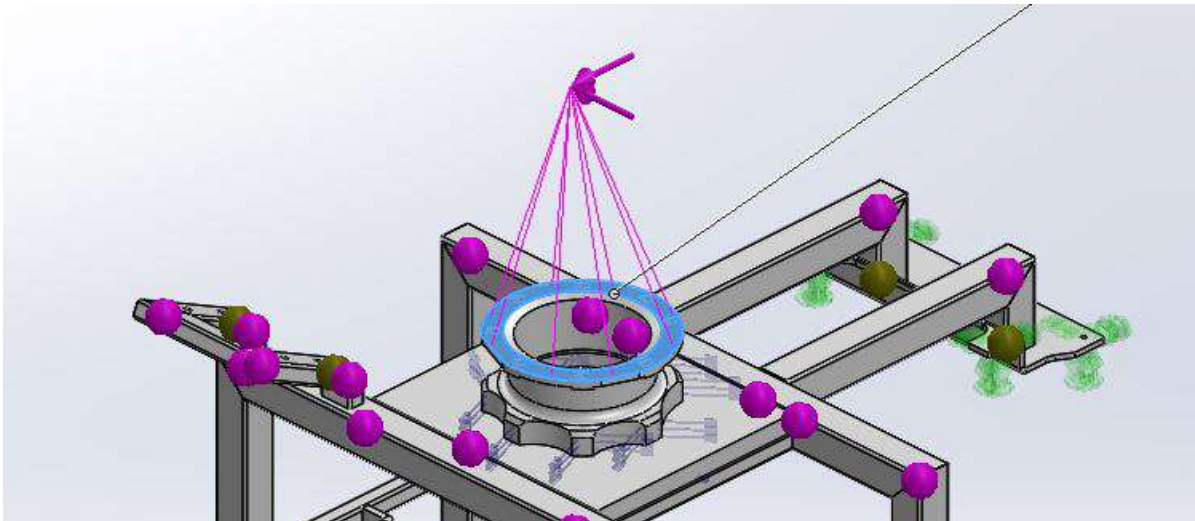
kreftene som må defineres i denne analysen, er kreftene som blir påført som følge av bremsing, ettersom et slikt kjøretøy har større akselerasjon ved bremsing, enn gassing, krefter som blir påført som følge av sving, og vekten av de ulike komponentene, som for eksempel våpenstasjonen og komponenter som muligens kommer til å belaste på brettene på hver side.



Figur 123: Kraft definert på det ytre røret på hver av bagasjebrettene, slik at man kan ha tung last, og samtidig stå på brettet.



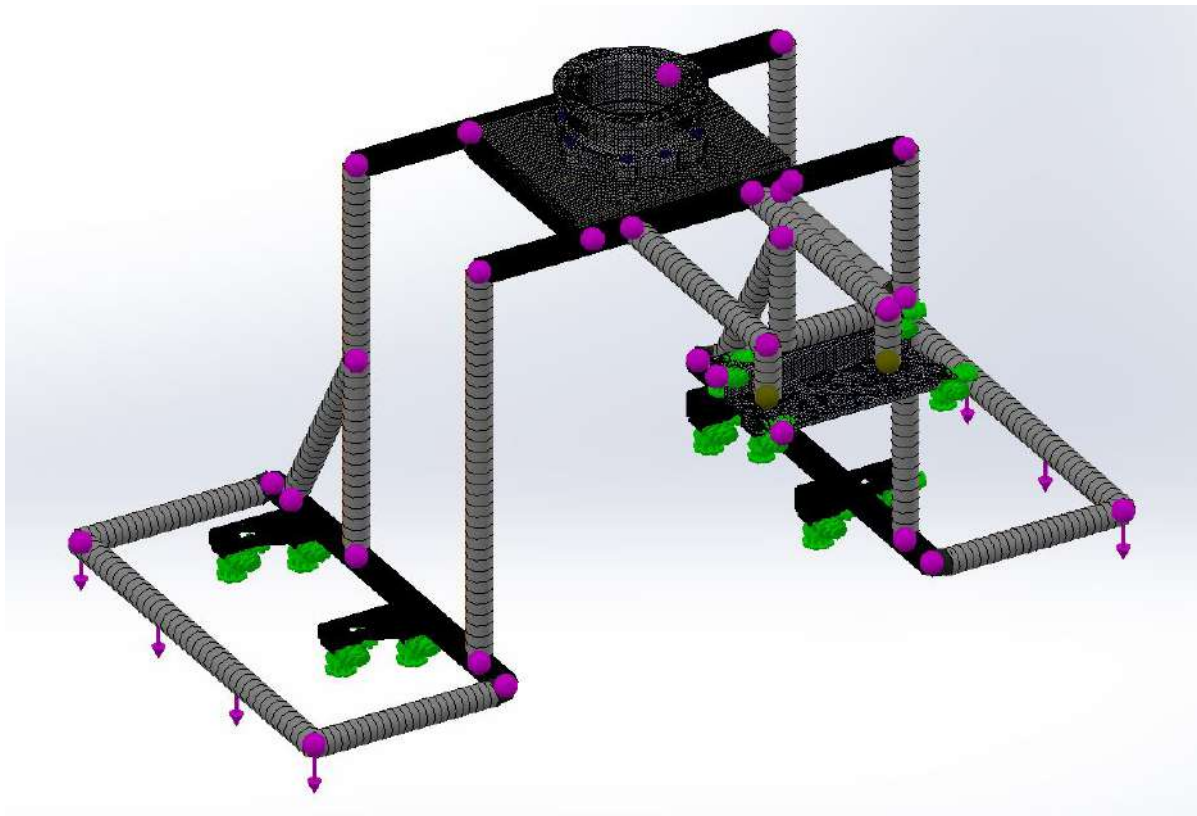
Figur 124: Kraft nedover for å simulere vekten til våpenstasjonen.



Figur 125: Kraft fremover for å simulere bremsing, og sideretning for å simulere i sving.

Mesh

Ettersom dette er en sammenstilling av både små og store deler er det behov for å bruke forskjellig mesh parametere til de forskjellige delene. Det ble derfor satt et hoved mesh, som i utgangspunktet gjelder for alle delene, deretter ble det brukt mesh control på deler som trengte finere eller grovere mesh.

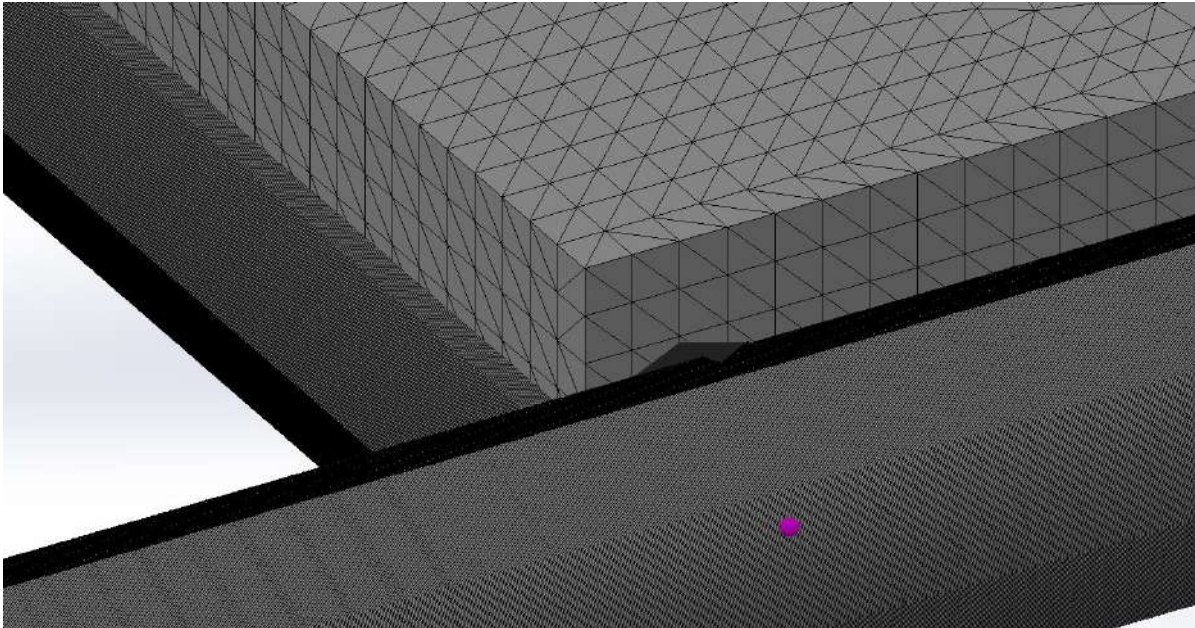


Figur 126: Mesh

Figur 126 viser hvordan hele sammenstillingen er delt opp i et mesh, med forskjellige mesh-parametre etter de forskjellige delene sitt behov.

Et eksempel på dette er deler av rammen, som har kontakt med andre komponenter. Siden rammen blir regnet som bjelker, og ikke som massive deler, blir den delt opp i elementer kun i lengderetning, ikke pyramideelementer, slik som en massiv del. På grunn av dette forbinder SolidWorks rammen til de andre delene kun i disse overgangene. Dette fører til at det blir få kontaktpunkter mellom rammen og for eksempel monteringsplaten til våpenstasjonen. Dette fører til at spenningene i monteringsplaten blir ekstremt høye

i disse kontaktpunktene. Derfor må rørene med kontakt til monteringsplaten ha et mye finere mesh, enn hva som er nødvendig på resten av rammen for å få et gyldig resultat i monteringsplaten.



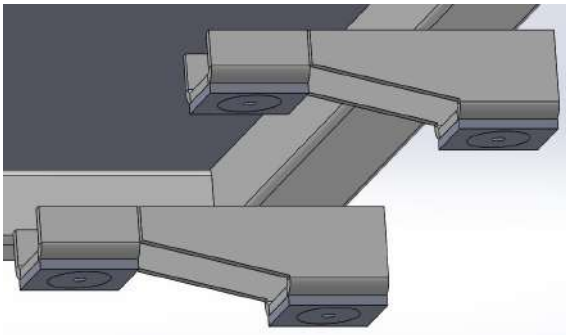
Figur 127: Eksempel på bjelker med forfinet mesh

12.1.5 Analyser

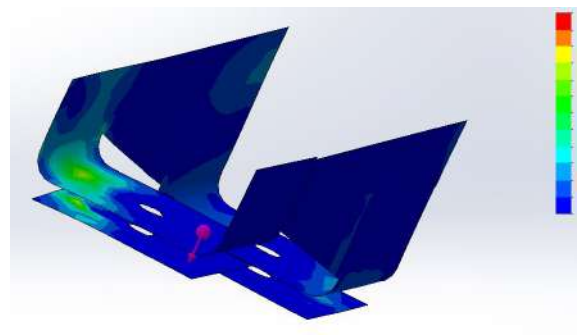
Ved dimensjonering av komponenter er det vanlig å bruke optimaliseringsfunksjonen i SolidWorks, hvor man legger inn parametre på mål, masse, maks tillatte spenninger osv. for så at SolidWorks tester ut forskjellige variasjoner av disse parametrene, og finner ut hva som fungerer best. Dette fungerer vanligvis godt, men på grunn av at dette er en kontaktanalyse som krever masse av datamaskinen, var ikke maskinen analysene ble kjørt på god nok for en optimalisering av dette designet. Av den grunn ble det nødvendig å kjøre en analyse, se på resultatene, optimalisere manuelt, så gjøre det om igjen helt til designet kom innenfor parameterkravene.

De første analysene viste at fothvilerovergangene ikke var kraftige nok. De fikk en spenningskonsentrasjon ved området de er festet til ATV-en. Dette er forventet på grunn av måten låsingene er definert med en skarp kant mellom låsingene og resten av delen, så dette vil trolig ikke ha like stor innvirkning montert på ATV-en.

I tillegg til disse spenningskonsentrasjonene var det også en spenningskonsentrasjon i kontaktflaten mot rammen, for å løse dette ble platetykkelsen til fothvilerovergangene økt. Dette hjalp på spenningskonsentrasjonen i kontaktflaten mot rammen, men spenningskonsentrasjonen mot ATV-en var fortsatt høy. Selv om denne konsentrasjonen ikke burde være like stor ved montering på ATV-en, ble det bestemt å lage en ekstra plate som kunne sveises på undersiden av fothvilerovergangene. På denne måten ble tykkelsen doblet der det var behov for mer gods, uten at hele delen ble overdimensjonert. Dette løste problemet, og spenningene havnet innenfor parameterne.

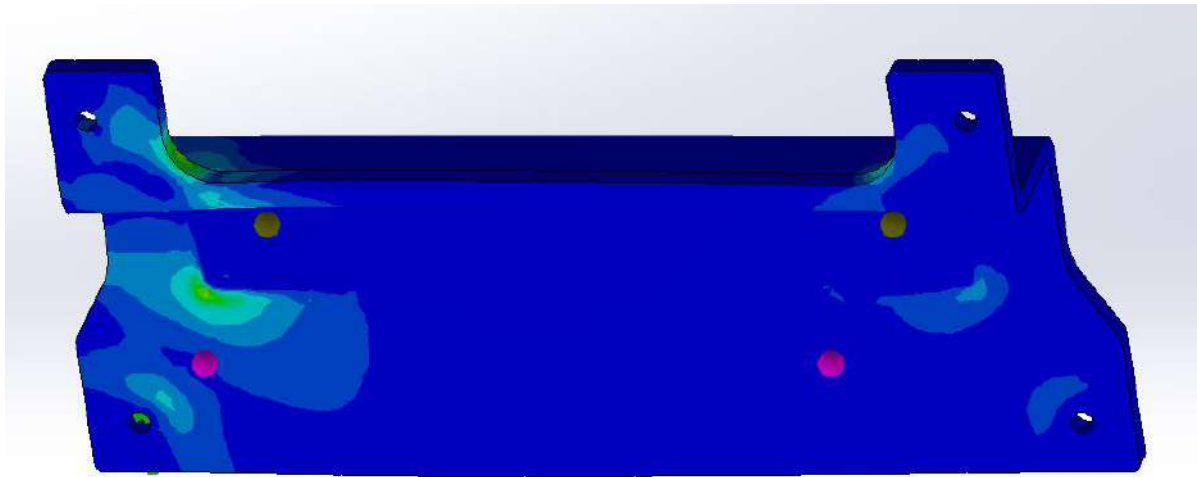


Figur 128: Fothvilerovergangene med plater for økt tykkelse på bunnflaten, for å motvirke spenningskonsentrasjonen.



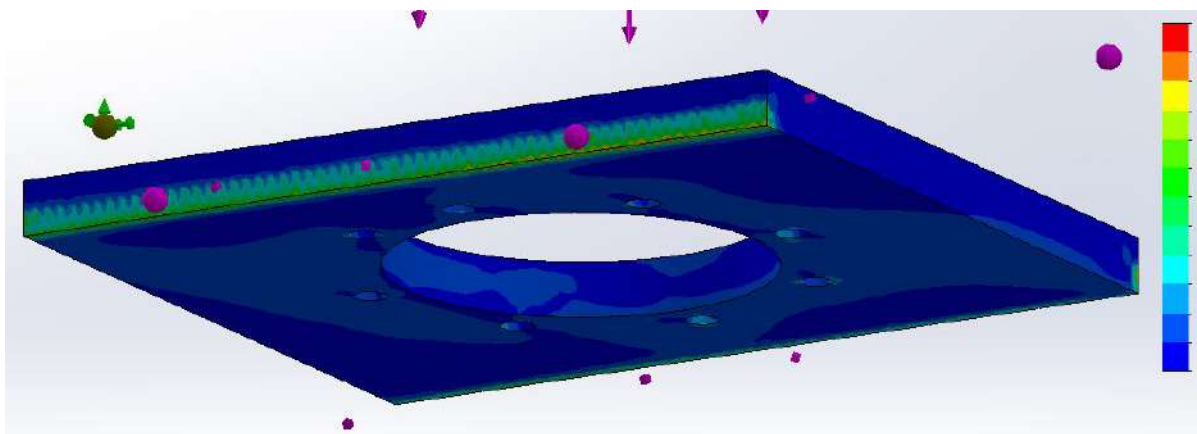
Figur 129: Forbedrede spenningskonsentrasjoner (i grønt og gult) med plater på bunnflatene.

Neste problem oppstod i overgangen til de øvre festepunktene, hvor det ble en spenningskonsentrasjon i radien ved der rammen skal sveises fast, og på platen, der hvor rørets ende skal sveises fast. Først ble det prøvd å øke radiusen på radien, for å få mer gods i det området. Dette minsket spenningene, men fortsatt ikke tilstrekkelig for å komme innenfor parametrene som var satt. Det ble derfor nødvendig å gjøre større tiltak, det ble vurdert å sveise på ekstra gods i området hvor konsentrasjonen befant seg, slik som med fothvilerovergangene. Istedenfor å gjøre dette ble platetykkelsen økt, for å da kunne bruke denne braketten til å montere annet ekstrautstyr som måtte komme senere. Dette gjorde at platen ble overdimensjonert flere steder, men spenningskonsentrasjonen ble minsket til å være innenfor parametrene, og det tilførte en ytterligere funksjon ved at den også kan fungere som innfestning til andre ting som måtte komme senere.



Figur 130: spenningskonsentrasjoner i platen, i grønne områder.

Det siste problemet oppstod i monteringsplaten for våpenstasjonen, hvor selve platen var innenfor parametrene, men innfestningen til rammen ikke ble sterk nok. Dette ble forsøkt løst ved å legge inn et firkantrør på undersiden av platen, på hver side. Dette gjorde ingen utslag på analysen, noe som ble sett på som veldig spesielt. Etter nøyere inspeksjon viste det seg at SolidWorks ikke inkluderte disse avstiverene i analysen, selv om de var satt til å bli inkludert. Dette problemet oppstod flere ganger, selv om de ble inkludert i mesh-et. Etter gjentatte forsøk fungerte fortsatt ikke dette, så problemet med monteringsplaten ble lagt til siden for å jobbe med optimalisering av rammen.

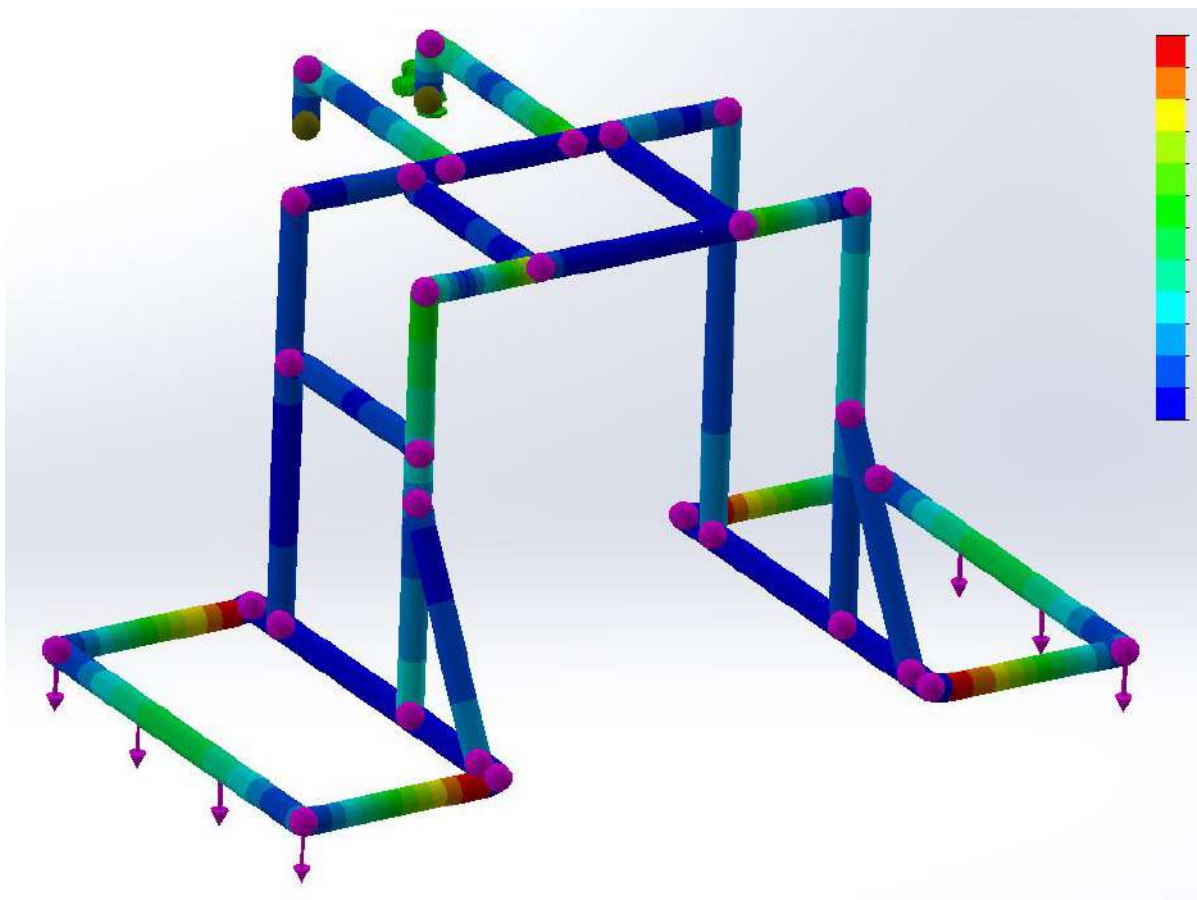


Figur 131: Spenningskonsentrasjoner i monteringsplaten

131 viser spenningskonsentrasjoner i monteringsplaten til våpenstasjonen. Spenningene

ligger utenfor gitte parametre i røde, gule og store deler av grønne områder.

Rammen hadde høye spenninger nederst i de vertikale rørene, som resultat av momentet som ble påført av våpenstasjonen, sammen i tillegg til ugunstig plassering i forhold til fothvilerovergangene, hvor det fremre røret traff midt mellom, og det baktre traff bak. Av den grunn ble det avgjort å sette inn støtter i fremkant av det fremre røret, som skulle treffe i nærheten av de fremre fothvilerovergangene. Når disse støttene kom på plass ble problemet med høye spenninger nederst i de vertikale rørene løst. Dette gjorde også at spenningene i selve fothvilerovergangene også ble minsket, selv om dette ikke var nødvendig.

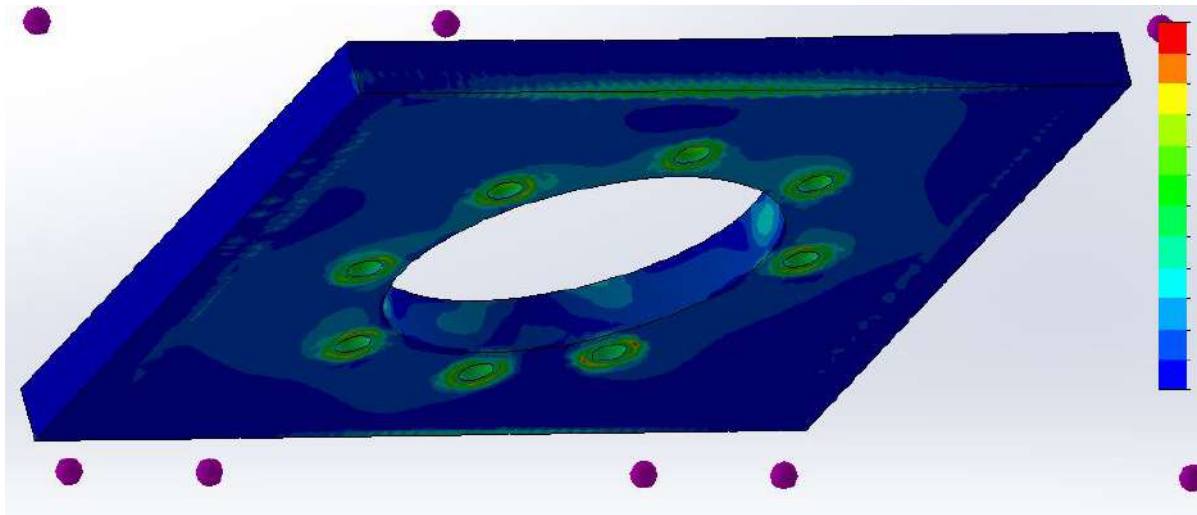


Figur 132: Spenninger i rammen

132 viser spenninger i rammen, hvor fargene symboliserer hvor høye spenningene er som følge av fargespekteret øverst til høyre, hvor rød representerer de høyeste spenningene.

Problemet med monteringsplaten løste seg ikke i løpet av analysene som ble kjørt for optimalisering.

Problemet med spenningene i monteringsplaten for våpenstasjonen måtte løses, derfor ble arbeidet tatt opp igjen. Uten å ha gjort endringer med oppsettet av analysen, inkluderte SolidWorks nå disse avstiverene til monteringsplaten. Avstiverene fungerte bra mot disse høye spenningene, og spenningene var innenfor ønskede parametre.



Figur 133: Spenninger i monteringsplaten

Figur 133 viser spenninger i monteringsplaten, hvor fargene symboliserer hvor høye spenningene er som følge av fargespekteret øverst til høyre, hvor rød representerer de høyeste spenningene.



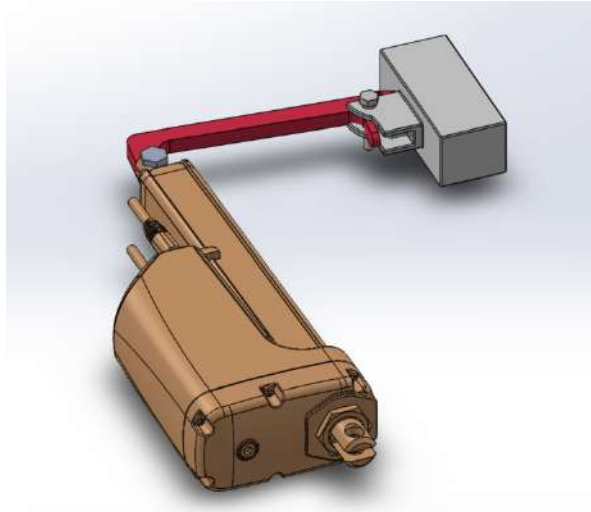
Figur 134: Ferdigstilt ramme med designede komponenter, aktuatorer og steppermotor, sett forfra



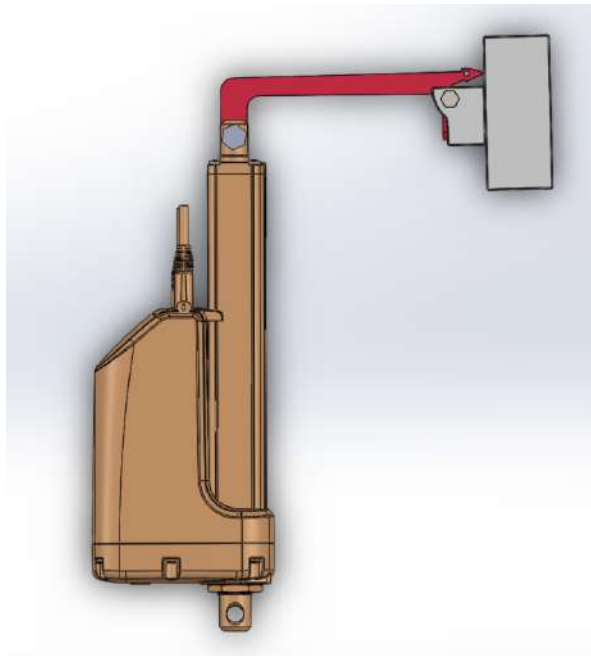
Figur 135: Ferdigstilt ramme med designede komponenter, aktuatorer og steppermotor, sett bakfra

12.2 Design av håndbrems

Håndbremsdesignet skal virke som kraftoverføring fra lineæraktuatoren LA33 til håndbremsstempelet. Det er bestemt under kapittel 9.3.2 at konsept 2 skal benyttes og aktuator LA33 er bestemt under 11.2.1.



Figur 136: Boksen representerer bremsen til ATV-en. De røde bremse-hantelene må designes for å passe boksen og LA33 aktuatoren



Figur 137: Systemet Sett ovenfra



Figur 138: Frakoblet bremschattel som viser bremsstempel

I figur 136 kan det sees at innfestningen fra rammen til aktuatoren hengsles fast. Det gir aktuatoren frihet til å bevege seg i 2 akser fra rammen som gir høyere frihet for tilpasning. Aktuatorstemplet og brems håndtaket skal festes sammen med et hengsleledd. Aktuatoren kan bedre tilpasses systemet ved hjelp av to hengsleledd; En justerbar lengde på aktuatorstempelet og et roterbart stempel som kan rotere om sin egen akse.

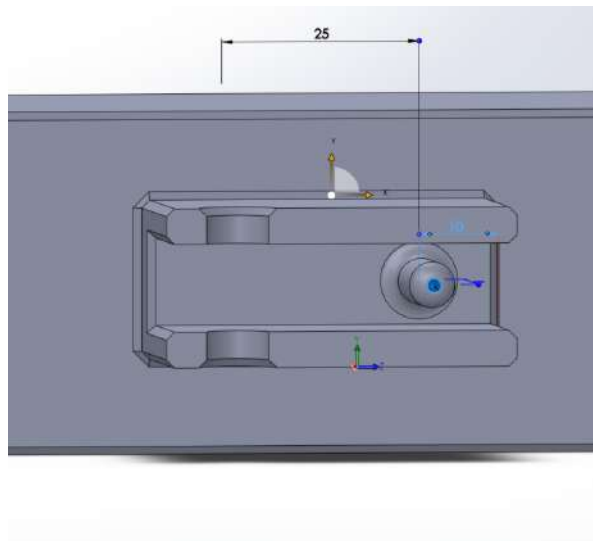
I denne løsningen skal innfestningen fra rammen til motoren tilpasses når rammen er ferdig designet. Det som gjenstår, er kraftoverføringen fra stempelet fra lineæraktuatoren (LA33) til bremsmekanismen til ATV-en. Den må designes på grunn av at håndtaket som er på ATV-en nå ikke har festepunkter kompatibel med aktuatoren.

12.2.1 Designparametere

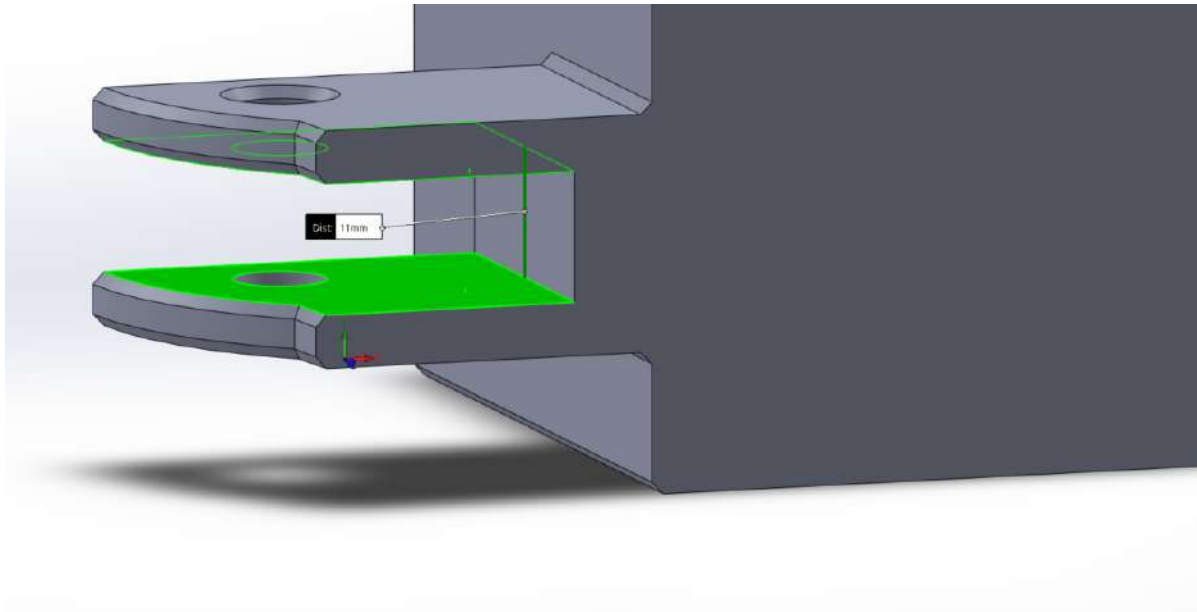
Det skal drøftes hva som er nøkkelparameterne for designet av den komponenten som fester lineæraktuatoren til bremsesystemet. Det skal utformes et bremsehåndtak som LA33 kan festes til. 9.3.2

Dimensjonsrelasjoner

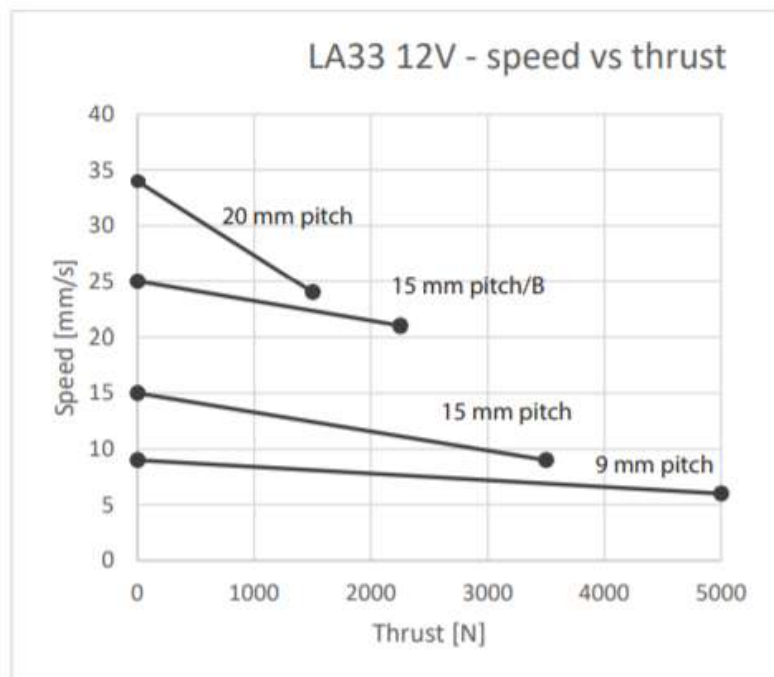
Først og fremst må komponenten tilpasses både LA33 og bremsesystemet til ATV-en. Bildet under viser hva komponenten må tilpasses. Det må tilpasses LA33 sitt hengselsledd og kunne hengsles fast i ATV på den andre siden. Det må også lages noe som trykker på bremsestempellet.



Figur 139: Bildet viser hvor stor avstand mellom bolthullet til hengselet og stempellet må være for å kunne passe inn i bremsesystemet til ATV-en.



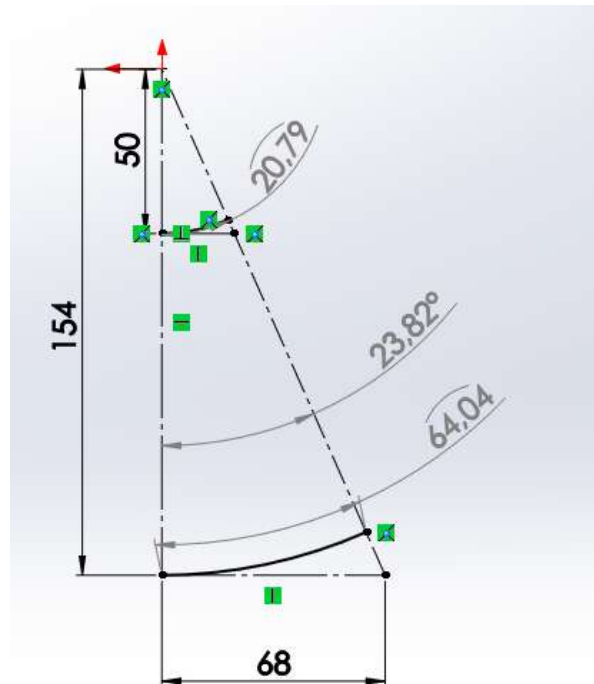
Figur 140: Bildet viser hvor stor tykkelsen maksimalt kan være for å passe inn i bremse-systemet til ATV-en. (11mm)



Figur 141: Grafen er hentet fra databladet til LA33. 20 mm pitch er versjonen som blir brukt i dette designet

Ytelsen blir definert ut ifra hvor rask bremsen går fra ingen bremseeffekt til full bremse-effekt. Det varierer ikke veldig mye med økt last ifølge bildet over. Det mest effektive er

å minske banelengden til LA33 for å nå maksimal bremseutvikling på kort tid. Under er et bilde av en matematisk fremstilling av hvor lang banelengden blir (142).



Figur 142: Matematisk fremstilling og total lengde aktuatorhengslet må bevege seg for å trykke bremsen inn 6mm

Den originale bremse-hantelen skal representere linjen som er 154 mm lang. Fra hengselet på ATV-en (origo øverst i bilde) er lengden 154 mm til referansepunktet. Fra det punktet ble bremse-hantelen trykket helt inn til hvor mye bremsekraft man ønsker å kunne ha med LA33. I den vinkelrette retningen ble det målt at hantelen kunne bli trykket inn 68 mm i en akseretning. Det tilsier en vinkelendring på 23,82 grader ifølge utregninger i Solidworks. Hengselet har en konstant radius til aktuatorhengselet. Med denne modellen testes for eksempel en arm på 50 mm. Det gir en buelengde på 20,79 mm som LA33 må bevege seg for å få fullt utslag på bremseeffekten. På denne måten kan det optimaliseres hvilken lengde som skal designes.

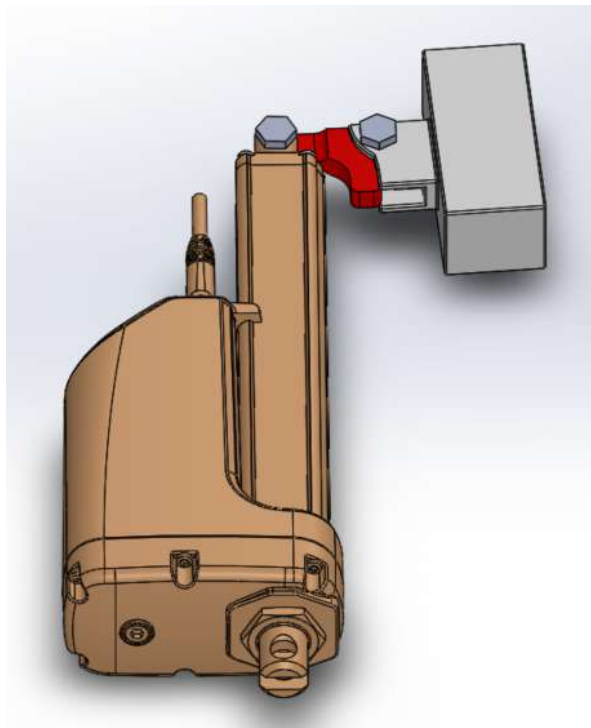
En mindre buelengde gjør at full bremseeffekt inntreffer fortere. Mindre arm gjør at det krever mer kraft for å oppnå samme moment. En optimaliseringsprosess må til for å finne ut hvilken armlengde som er den mest optimale. Det er satt et krav om at det skal ta maks et sekund før full bremseeffekt oppnås i kapittel 6.7.1.

Spenning i komponentene

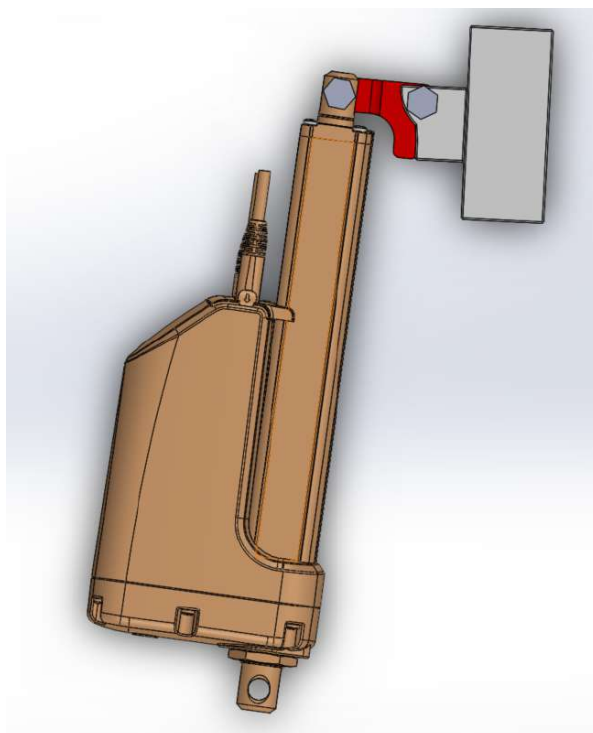
Komponentene må tåle høye nok spenninger i forhold til det de blir utsatt for. Basert på studiet av ulike komponenter ser det ut som at håndtaket kan være veldig tynt gitt at materialet er av aluminium. Valg av materialer vil bidra til at konstruksjonen vil tåle høyere spenninger. Da spiller korrosjonsbestandighet inn, da det for eksempel ikke ønskes at det skal korrodere. Fra vannskjæresenteret AS, hadde de Aluminium 6082-T651 og stål S355 tilgjengelig. De har lokaler i nærheten og har rask leveringstid.

Av de parameterne som kan endres, er den mest utslagsgivende parameteren avstanden mellom hengslene. Det kunne fokuseres på å gjøre delen lettere eller tåle mer krefter. Ut ifra analyser er spenningene relativt små i forhold til det komponentene tåler. En optimaliseringprosess basert på minimering av masse, vil ha minimal innvirkning ettersom komponenten allerede er veldig liten. Endringer på armen utgjør stor forskjell. Den vil ha direkte påvirkning på momentutvikling og reaksjonsevnen/hastigheten på bremsen. Det vil fokuseres på å finne et design som kan få frem de beste kvalitetene for disse prosessene med utgangspunkt i kravene.6.5.3

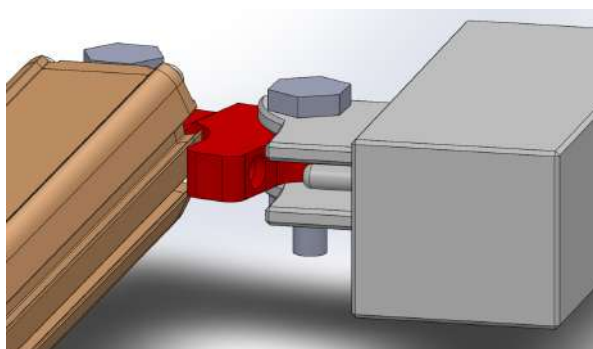
12.2.2 Første utkast



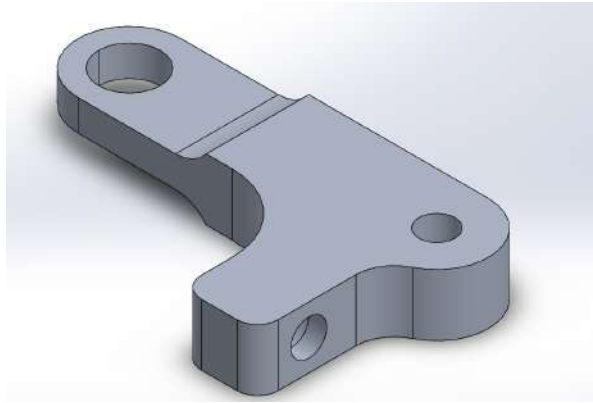
Figur 143: Samspill mellom komponentene.



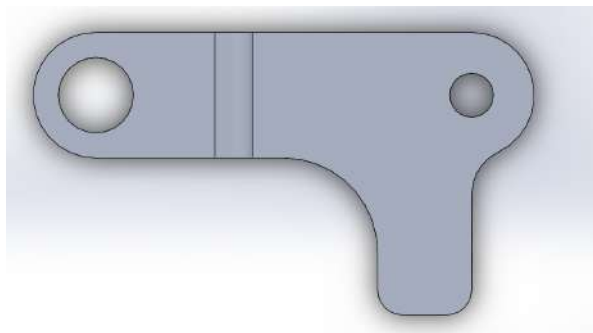
Figur 144: Systemet sett ovenfra



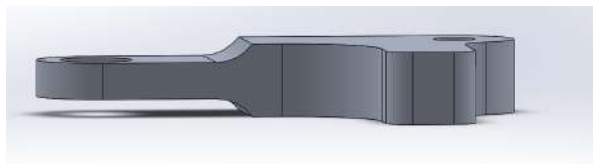
Figur 145: Interaksjonene mellom kulehullet og bremsestemplet



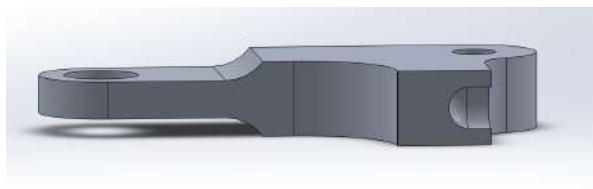
Figur 146: Håndbrems: Isometrisk synsvinkel



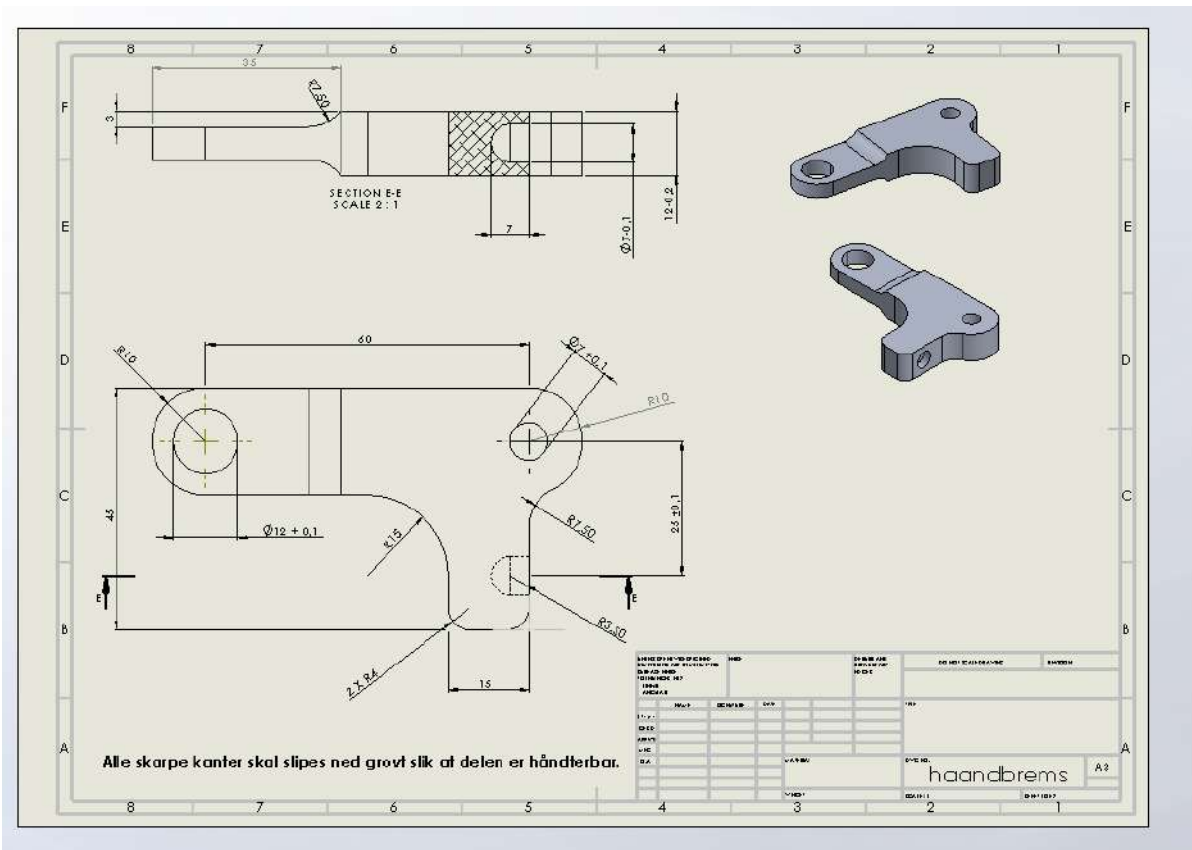
Figur 147: Håndbrems: sett ovenfra



Figur 148: Håndbrems: sett fra siden



Figur 149: Sectionview av stempelhullet



Figur 150: 2D tegninger med dimensjoner av komponenten

12.2.3 Design - begrunnelser

Nøkkeldimensjonen er på 60 mm mellom hengselet til ATV-festet og hengslet til LA33. Armen utgjør hvor mye momentet som blir generert. Det blir generert like stort moment i den andre retningen på grunn av at momentet om hengselet er lik 0, dersom det ikke er bevegelse når bremsen er trykket helt inn. Se rapport fra matematiske beregninger for mer informasjon. A.7

Dimensjonen på 25 mm på den nedre tegningen utgjør hvor mye motkraft som blir generert fra og til på stampelet (Newtons tredje lov). 150 Bremsstempelet vil gå inn i det kulehullet som vist i figuren 143. Desto mindre armen hadde vært, desto større motkraft må den generere for å gi like stort motmoment slik at momentet om hengselet blir lik 0. For å kunne benytte seg av innfestningen som allerede befinner seg på ATV-en, så velges det å beholde den armen på 25 mm.

Den mest aktuelle parameteren å endre på, er armen på 60 mm for å generere nok kraft eller få raskere bremseeffekt. Armen må ikke bli for lang ettersom banelengden blir på-

virket. Operasjonskraften må ikke bli for stor, da det fører til økt slitasje på motoren.

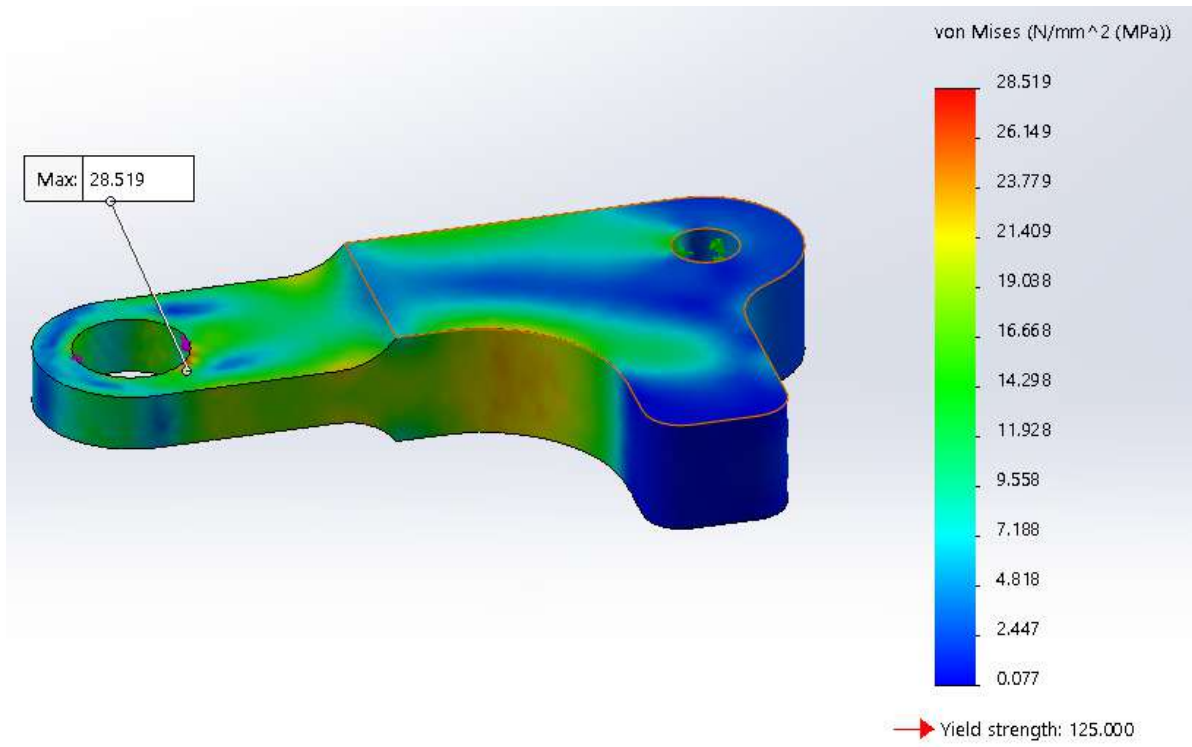
Farten til LA33 er gitt ved grafen fra Linak figur 141. Farten varierer fra 34 mm/s uten last til 24 mm/s med full last. Siden begynnelsen av bevegelsesbanen skal starte der det er kontakt med bremsestempelen, blir det tilnærmet ingen del av banen det ikke er last. Dette gjør bremsen veldig sensitiv. Ved fysiske tester kan det etableres hvor mye "no load" fase den skal ha for å gjøre bremseknappen passelig sensitiv for operatøren.

For å illustrere banelengden til aktuatorstempelen ble en forenklet modellsketch laget i Solidworks.¹⁴² Denne modellen baserer seg på formelen for buelengde og målinger. Banelengden til aktuatoren i dette designet blir 25 mm. Kontinuerlig full last skal resultere i full bremseeffekt på 1 sekund. Det antas å bli en del mindre da det ikke er fullast på 1500 N, men heller en driftskraft på maks 427 N, tallet er basert på vedleggs-dokumentet "Forsøk: Kraft beregninger på ATV".A.7

Det som må maskineres er kulehullet som stempelet skal trykke på og området der delen er smalere. Se bilde fra siden for en illustrasjon. Tverrsnittet er tynnere på grunn av at LA33 aktuatoren skal kunne hengsles på. Kulehullet må ha tykkere vegger for å forhindre slark og for å stabilisere innfestningen.

I området der tykkelsen av delen endrer seg (figur 148) er tykkelsen av delen tynnere for å kunne passe inn i hengselet til LA33 fra figur 139. Det er valgt fillets (avrundinger) slik at stresskonsentrasjonene kan reduseres. Filletstørrelsen har radius på 7,5 mm, som er en vanlig størrelse på en fres. Det gjør at spesialutstyr kan unngås og priser kan være lave. Lignende løsninger kunne blitt gjort med for eksempel chamfers (vinkelavrunding). Bruk av fillets gjør det enkelt å bruke en valsefres for å få jobben gjort på en enkel og billig måte. Samtidig reduseres stress-konstrasjonen på en god måte. Avrundingen er konsentrisk til hengselet til LA33. Den er designet slik at når aktuatoren roterer rundt hengselleddet, kolliderer ikke LA33 med komponenten. Dette tillater mer rotasjon om det skulle være behov for dette. Se figur 143.

Delen blir produsert i aluminium. Ulempen er at aluminium har lavere flytegrense i dette tilfelle, men spenningene er veldig lave og gir høy sikkerhetsfaktor. Materialet Aluminium 6082-T651 [1] har ifølge nettstedet en flytegrense på 300 Mpa og maksspenningene ligger på 28.5 MPa.



Figur 151: FEM analyse av delen viser høyeste spenninger på 28.5 MPa.

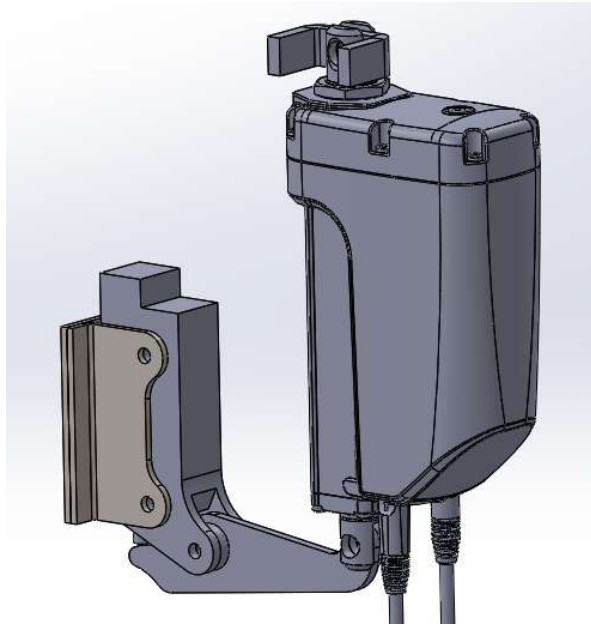
12.2.4 Evaluering

Designet oppfyller de nevnte design parameterne på en god måte. Det gjenstår å se hva slags ytelse denne vil gi i forhold til total bremskraft og bremsehastighet. Systemet krever testing for å etablere hva ytelsen vil ende opp med å bli. Mange parametere er vanskelig å fastsette ved beregninger. Designet er veldig balansert, slik at det er mulig å regulere ytelse i form av kraft og hastighet gjennom strømforsyningen til LA33. En videre optimaliseringsprosess vil være lettere å finne ut av etter en serie med flere tester. Dette vil vise hvordan alt fungerer i praksis; hvor myk bremsingen er og hvor reaktiv den er.

For å se samhandling mellom håndbremsen og fotbremsen se seksjon 12.4.

12.3 Design av fotbrems

Fotbrems designet skal virke som kraftoverføring fra lineæraktuatoren LA33 (Bestemt under kapittel 11.2.1) til fotbremsesystemet. Konseptet ble bestemt under 9.3.3.



Figur 152: Interface mellom aktuatoren og delen

Innfestningen fra rammen til aktuatoren skal hengsles fast. Det gir aktuatoren frihet til å bevege seg i 2 akser fra rammen. Aktuatorstemplet til bremsekomponenten skal bli festet med et hengsel. Aktuatoren har hengselledd og en justerbar lengde på stempelet, som gjør det lettere å tilpasse delsystemet til ATV-en.

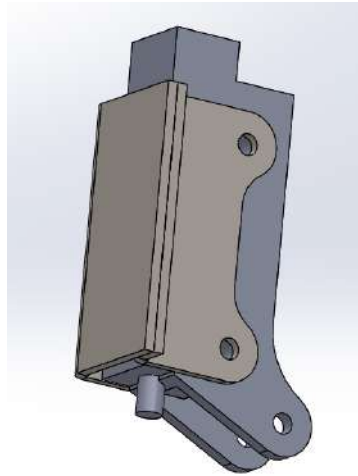
I denne løsningen skal innfestningen fra rammen til motoren tilpasses etter hvert som rammen er ferdig designet.

12.3.1 Designparametere - fotbrems

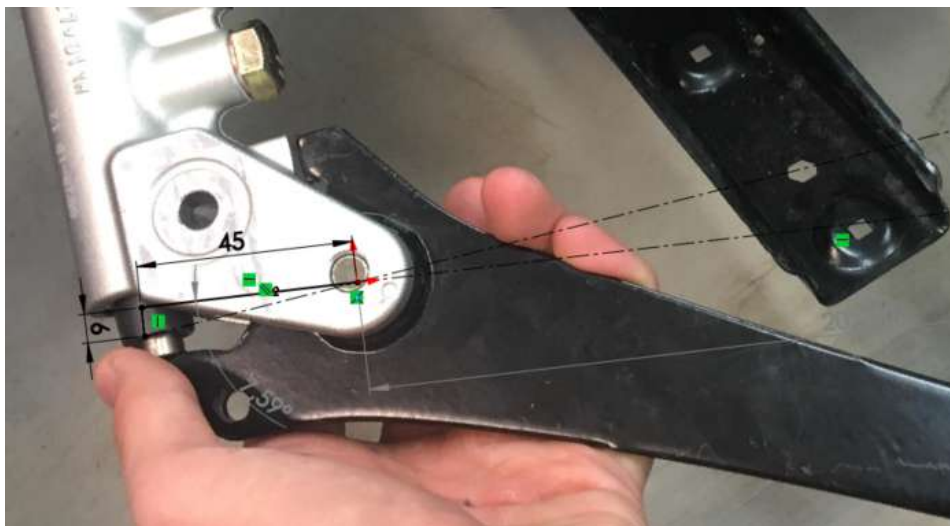
Det skal designes en form for bremsepedal som LA33 festes til. Løsningen er forhåndsbestemt under kapittelet 9.3.3. Fotbremsen følger de samme designparameterene som for håndbrems.

Figur 153 viser hvordan komponenten må tilpasses fotbremsen. I den ene enden skal den festes til LA33 sitt hengselledd. I den andre enden skal den sitte på ATV-en og trykke

på bremsestempelet.

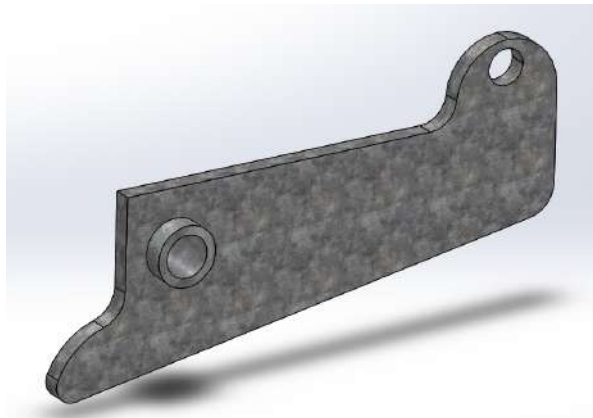


Figur 153: Bremsestempelet må dyttes inn for å bremse.

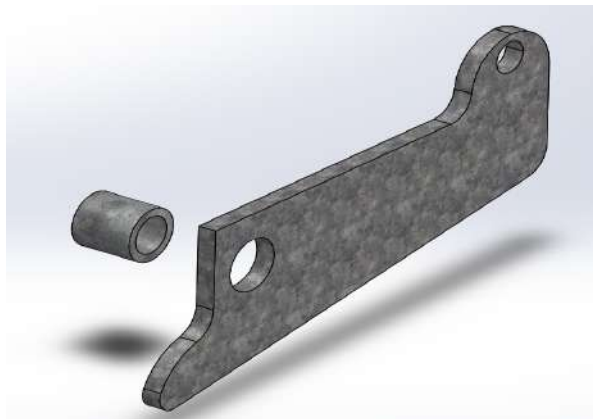


Figur 154: Matematisk modell over hvor lang buelengden blir under forskjellige armlengder da vinkelen er konstant

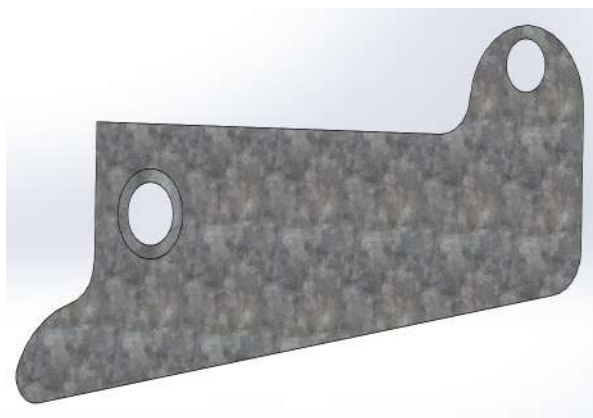
12.3.2 Første utkast



Figur 155: Fotbrems: Isometrisk synsvinkel



Figur 156: Isometrisk synsvinkel før sammensveising



Figur 157: Fotbrems: Sett fra siden

Dette designet er sterkt inspirert fra bremsepedalen som allerede eksisterer på ATV-en. Det ble ansett at designet som finnes på ATV-en allerede oppfyller de fleste kravene fra før.

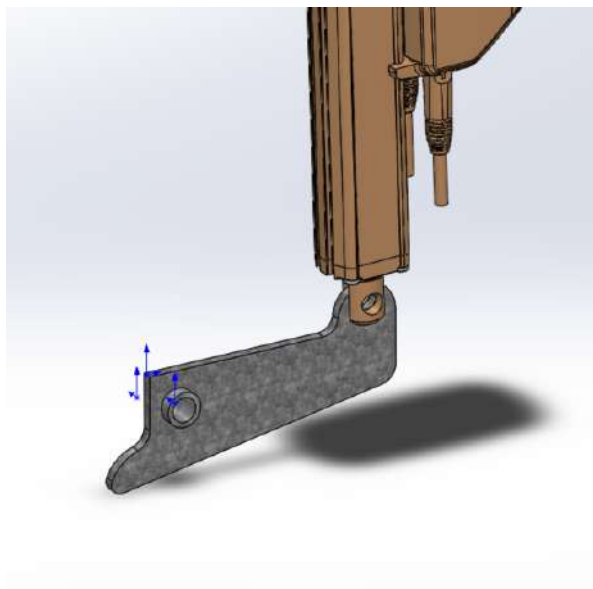


Figur 158: Original bremsepedal

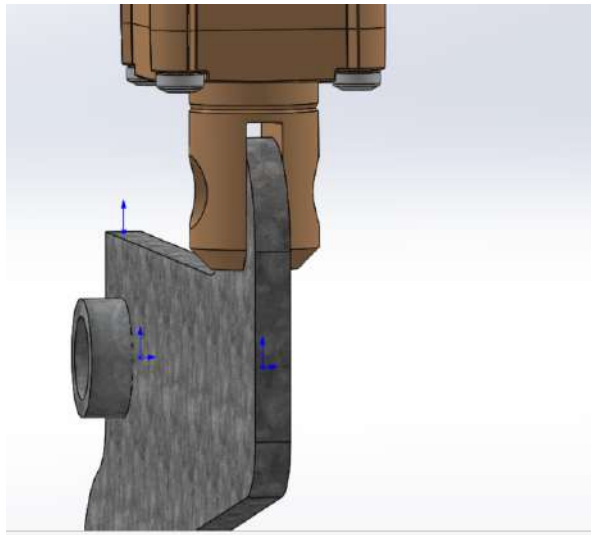
Platen på figur 155 er 6 mm tykk slik at den passer inn i LA33 aktuatoren sitt hengsel. Rommet mellom hengselhullene til LA33 er 6.20 mm.

Utformingen er tegnet tilnærmet lik den originale. Det oppstod noe avvik i forhold til originalen siden nøyaktig kopi er vanskelig å oppnå.

Rundt festepunktet til LA33 er det designet en halvsirkel som er konsentrisk med hullet. Grunnen til dette er at LA33 kan roteres mye lengre uten å kollidere.

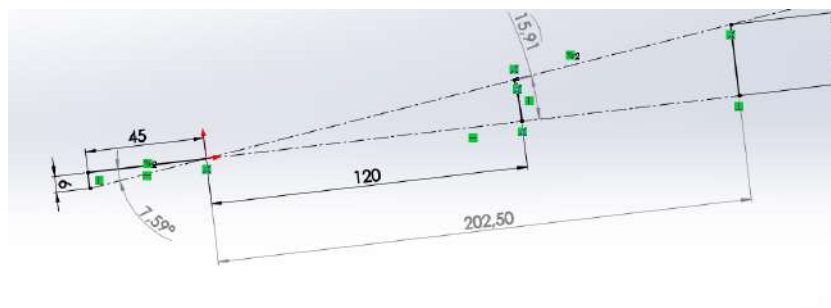


Figur 159: Samhandling mellom aktuatoren og delen



Figur 160: Klaring for at aktuatoren kan rotere rundt hengselet.

Ved vandring vil platen fortsatt ikke treffe LA33. Dette ble vist ved testing i solidworks assembly.



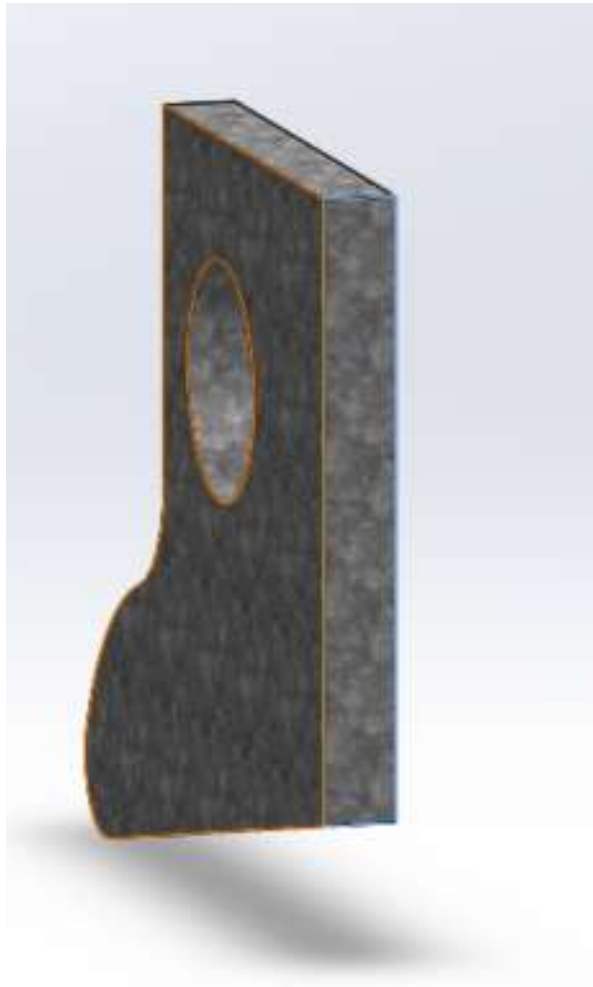
Figur 161: Det viser seg at slaglengden ved denne løsningen blir nesten 16 mm.

Armen mellom hengselhullene er redusert i forhold til for å kunne redusere nødvendig slaglengde. Med arm på 120 mm og ved å plote det inn i den matematiske modellen viser det at banelengden til aktuatoren ender på å være

Platen er en 6 mm stålplate (b). Ut ifra motstandsmomentet for et rektangulært tverrsnitt, gir dette høy verdi for høyden (h). Det gjør designet motstandsdyktig mot bøyemoment i vertikal retning, den samme retningen som LA33 dytter i. Høy verdi for motstandmoment er ønskelig og formelen for dette er:

$$W_x = \frac{bh^2}{6} \quad (28)$$

Det blir høy verdi på motstandsmomentet på grunn av at høyden h har relativt stor verdi. Se [42] for mer detaljer

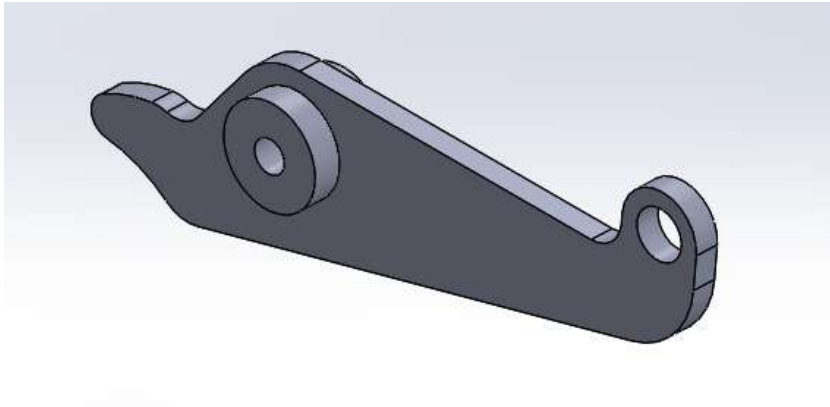


Figur 162: Geometrien gir høyt motstandsmoment for retningen stempelet trykker.

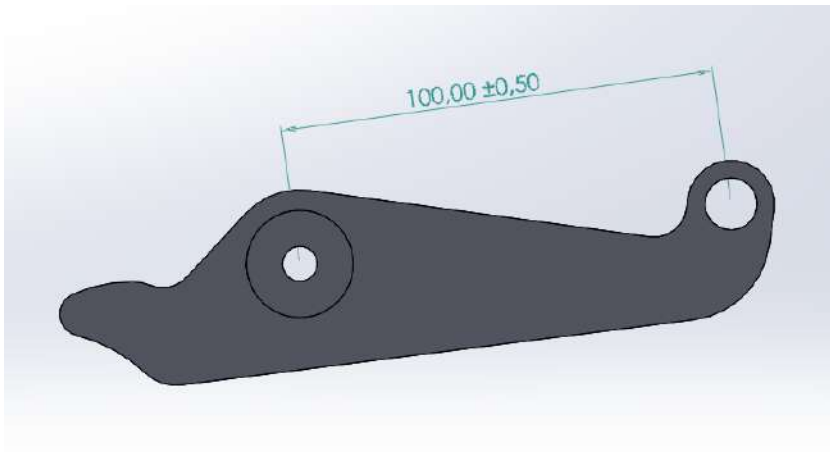
12.3.3 Evaluering

Designet skal fungere, i følge analyser, med gode marginer. Det er fortsatt rom for optimalisering av armlengden. Det er ønskelig å teste hvordan alt fungerer i praksis før det gjøres videre endringer.

12.3.4 Andre utkast

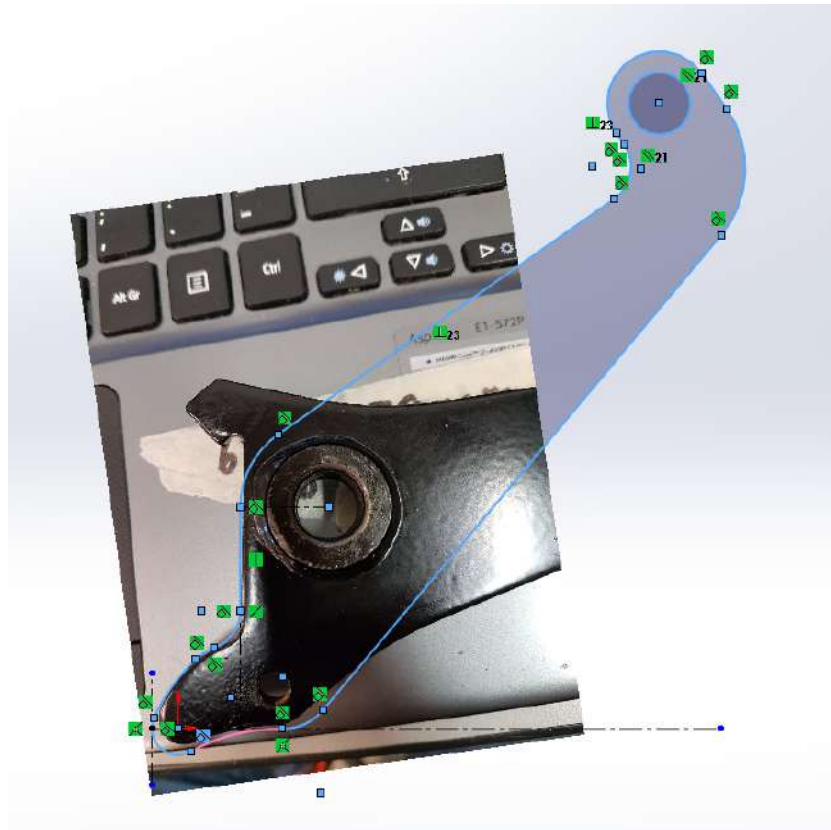


Figur 163: CAD modell - Fotbrens



Figur 164: Trenger noe klaring for at aktuatoren kan rotere rundt det hengselt. Ved rektangulær flate på delen ville det fysisk stoppet aktuatoren

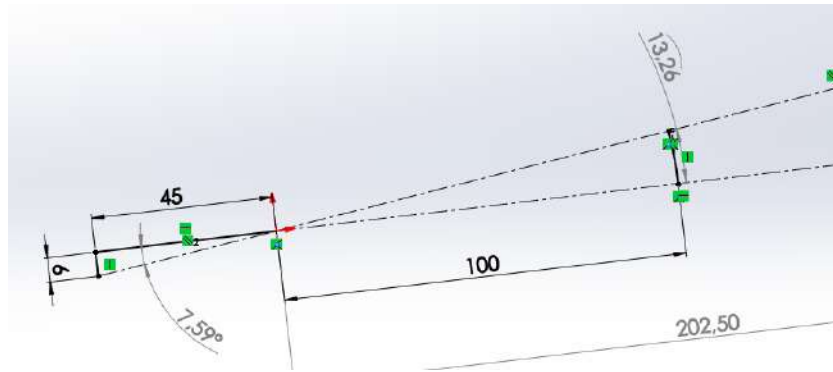
Modellen over har mange av de samme egenskapene som den forrige utgaven. En essensiell forandring er designteknikken. Under er det tatt et bilde slik at proporsjonene av den originale fotpedalen er ganske lik. Dersom bildekvaliteten er optimal og bildevinkelen er vinkelrett ville det vært en god tilnærming av de ekte dimensjonene, men noe avvik er forventet.



Figur 165: Designteknikk

Det ene hullet er 14 mm på innsiden av sylindere. Det gir en referanse på dimensjonene, og antas at alle andre mål er forholdsvis like sett fra bildets standpunkt. Det antas at dette designet blir mer likt den originale fotpedalen enn hva det forrige designet var. Det viktige var å få frem kontaktflate til stempelet, slik at slitasjen blir mer jevnt fordelt.

Basert på den matematiske modellen, blir buelengden 13.26 mm. Siden armen i dette designet fra hengselet bort til stempelet er 47,5 mm istedenfor 45 mm, blir vinkelen marginalt mindre. Buelengden blir i realiteten litt mindre, men antatt neglisjerbar.



Figur 166: Matematisk modell

Denne delen gikk videre til produksjon.

12.4 Samhandling mellom bremsene

Det er to individuelle systemer som utgjør brems til ATV-en, håndbrems og fotbrems. Det er slik at håndbrems er primærbremsen fordi den er langt sterkere i bremseeffekt enn fotbrems. Fotbrems gir en svak bremseeffekt, håndbrems gir stor bremseeffekt. Det må designes et system som gir et godt samspill for best mulig brems.

Hvordan skal det koordineres

Fotbremsen er langt svakere enn håndbremsen i bremseeffekt, men ifølge styrkeberegningssrapporten A.7 kreves det 36,8 Nm på håndbrems kontra 78,75 Nm på fotbrems for å trykke bremsen helt inn til det ønskede nivået på hver enkelt. Vinkelen i buelengdemodellene er nesten 24 grader på håndbrems og 8 grader på fotbrems. Det gjør at håndbremsen krever 3 ganger så lang buelengde gitt samme armlengde. Hvor mye nødvendig kraft øker når stemplene tilføres kraft er uvisst. Dette kan testes med aktuatorene.

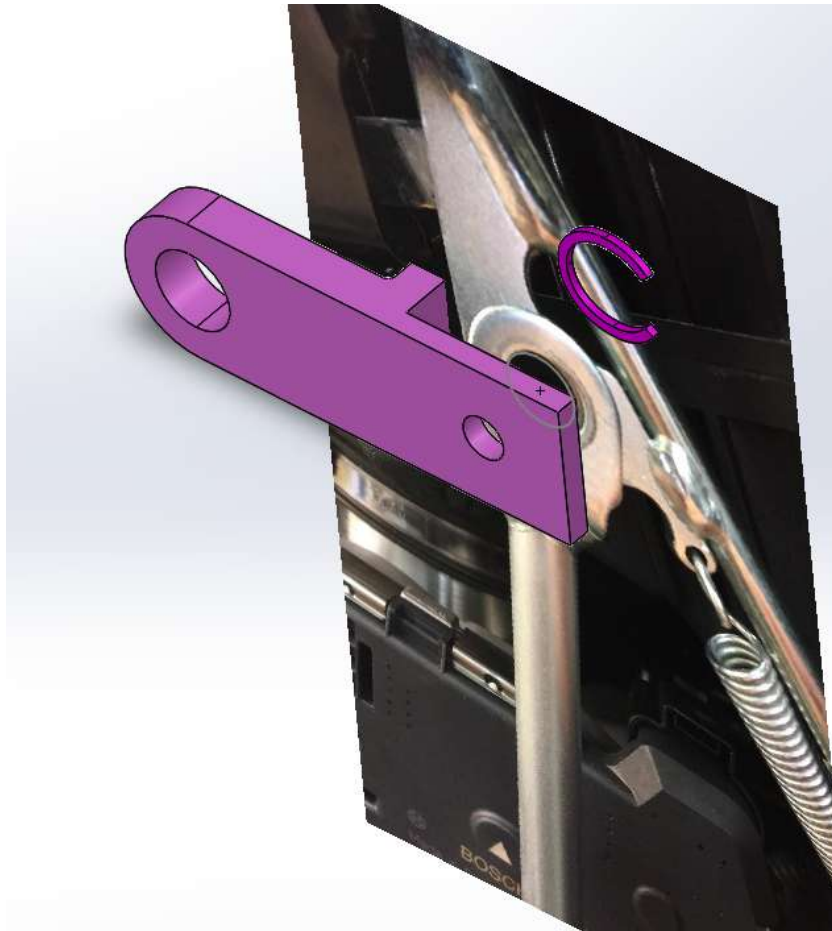
Det kan føre til økt slitasje dersom den ene bremsen aktiveres mer enn den andre. Dette blir nødvendig å teste på ATV-en for å finne et godt samspill.

12.5 Design av girsystemet

Design av girsystemet skal løse konsept 1. LA14 aktuatoren skal overføre kraft over til girspaken ved å hengse seg fast i punkt C.



Figur 167: LA14 kobles til hengsleddet i punkt C



Figur 168: Nærbilde av hengslingen i punkt C

For denne løsningen skal lineæraktuatoren hengsles fast i rammen. Deretter skal aktuatoren også festes med hengsler ved punkt C. Dette gir løsningen høy bevegelsesgrad og er lettere å tilpasse/montere.

12.5.1 Designparametere

Dimensjonsrelasjoner

Komponenten må tilpasses både LA14 og punkt C til ATV-en. Den må kunne hengsles i leddet og ikke kollidere med rammen eller andre komponenter.

Kraftoverføring uten slakk

Løsningen baseres på posisjonssensorene i aktuatoren som registrerer hvor langt stempellet har beveget seg. Dersom det er slakk i overføringen vil dette påvirke utfallet. Dette vil resultere i en bomgiring og ATV-en kan ta skade av det.

Rask Ytelse

Technical specifications

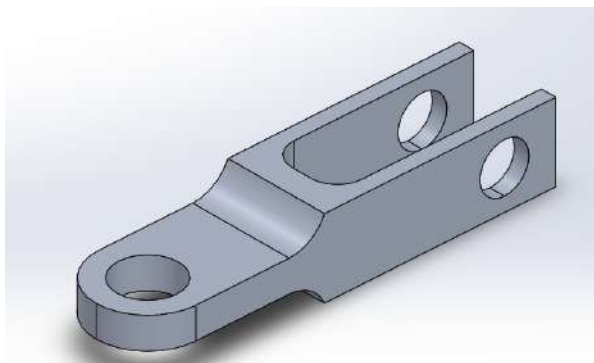
Type	Motor voltage (V)	Spindle Pitch (mm)	Thrust max. Push/Pull (N)	Self-lock max. (Push) (N)	Self-lock max. (Pull) (N)	*Typical speed (mm/s)		Stroke length (mm)		*Typical Amp. (A)		
						No load	Full load	Min.	Max.	No load	Full load	
14020xxxxxxxxA...	12	2	750	750	375	15	8	19	-	130	0.3	2.4
14020xxxxxxxxB...	24	2	750	750	375	15	9	19	-	130	0.15	1.3
14020xxxxxxxxC...	12	2	750	750	375	23	15	19	-	130	0.4	4.2
14020xxxxxxxxD...	24	2	750	750	375	24	16	19	-	130	0.2	2.5
14040xxxxxxxxA...	12	4	300	150	150	29	20	19	-	130	0.4	1.7
14040xxxxxxxxB...	24	4	300	150	150	31	24	19	-	130	0.2	0.9
14040xxxxxxxxC...	12	4	300	100	100	43	36	19	-	130	0.5	2.6
14040xxxxxxxxD...	24	4	300	100	100	45	38	19	-	130	0.3	1.3

Figur 169: 14040xxxxxxxxC, er versjonen av LA14 som blir brukt. Se [59] for fullt datablad

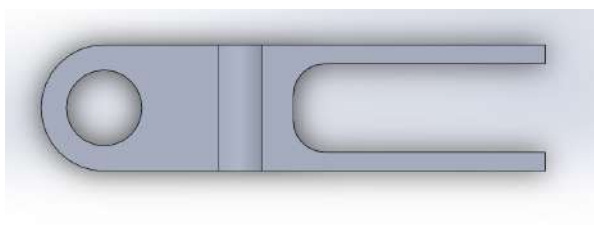
Aktuatoren har en hastighet på 36 mm/s ved fullast på 300 N. Det driftes med nærmere no load (0 N) enn fullload (300 N) ifølge styrkeberegning-rapporten (Se vedlegg), så dette er raskt nok for å skifte gir.

Girspaken skal ikke bli utsatt for mer enn 100 N. Punktet av moment foregår på toppen av girspaken. Det er antatt at girspaken hengte seg opp til en viss grad under fysisk testing (Se styrkeberegning-vedlegget). Siden det er automatgir på ATV-en, så er det ikke behov for at gir skal skiftes raskt. Det er blitt besluttet at det kan ta 1-2 sekunder for å skifte gir. Designparameteret om å skifte gir raskt kan nedprioriteres i forhold til komponentene i de andre delsystemene. Det ser ut til å klare det på minst et sekund ifølge beregninger.

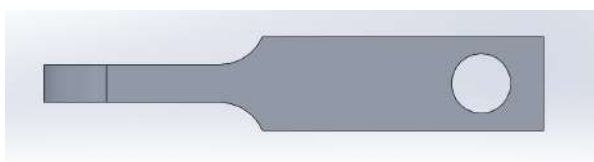
12.5.2 Første utkast



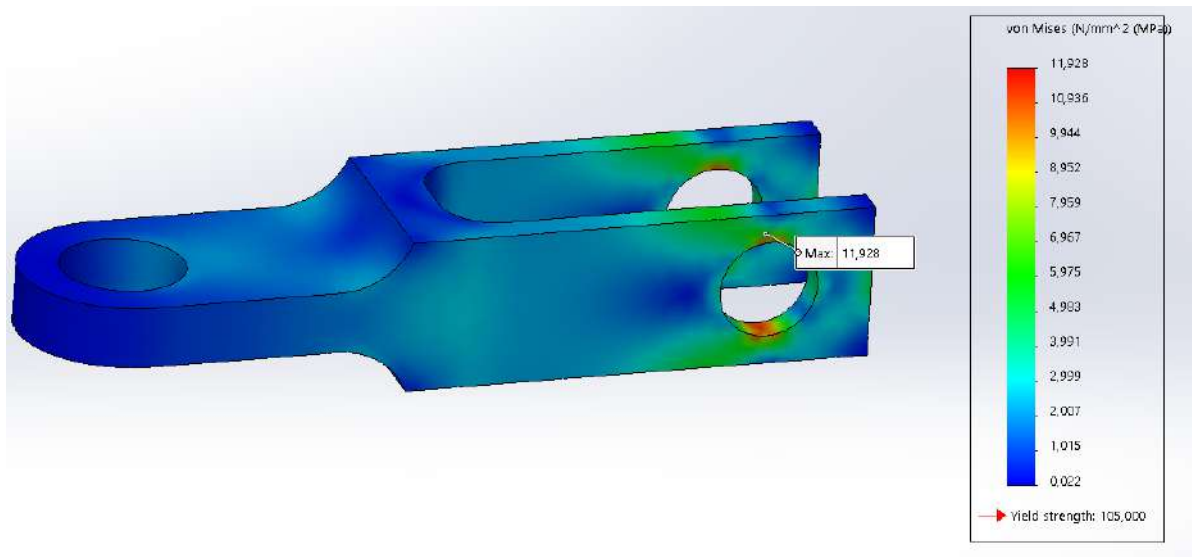
Figur 170: Girarm: Isometrisk synsvinkel



Figur 171: Girarm: Isometrisk synsvinkel før sammensveising



Figur 172: Girarm: Sett fra siden



Figur 173: Viser maksspenninger på 11.92 MPa.

Aktuatoren er festet mellom rammen og girarmen med to hengsler og videre til girspaken med et annet hengsel. Det betyr at det er totalt 3 hengsler. Hengselet fra LA14-stemplet til komponenten er stivet opp i forhold til de to andre hengsel-leddene.

Ved hengselet mellom LA14 og komponenten er komponenten avrundet i hjørnene. Denne er konsentrisk til hullet slik at delen ikke skraper borti aktuatoren når det beveger seg til sidene. Det gir systemet større frihet til å bevege seg i det planet.

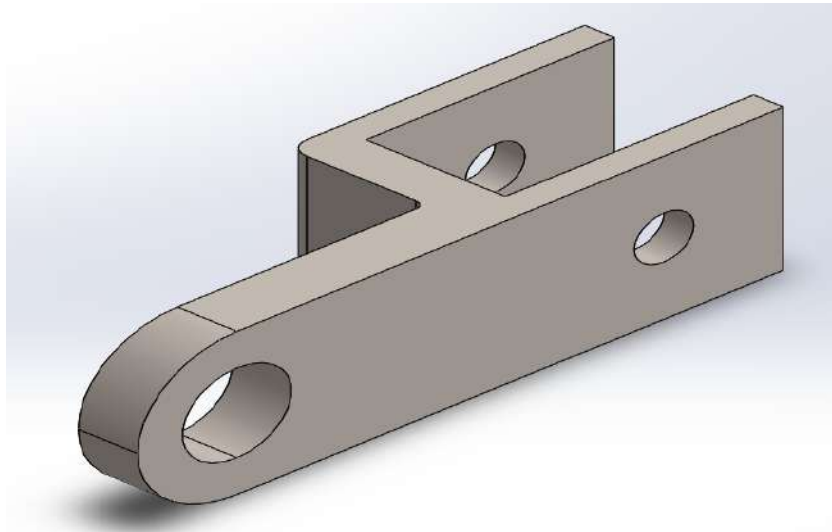
Det er fillets på 5 mm på innerveggene på delen slik at det skal være enklere å frese ut delen. 5 mm er lett å frese ut da det er en standardstørrelse på en fres.

Der tverrsnittet går fra å være gaffel-liknende til en større flate er det lagt til fillets på hver side (172). Det er for å minimere stresskonsentrasjon. Grunnen til at fillets er valgt er at det kan rullefreses med en radius på 7,5 mm, som også er en standard. For å illustrere, se figur 172.

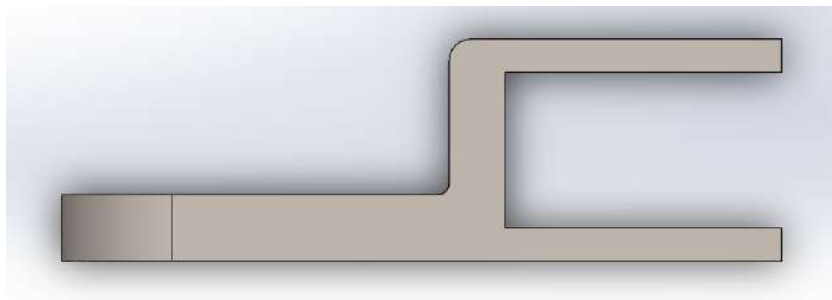
12.5.3 Evaluering

Det er litt bevegelse i sideretningen som kan ha fordeler slik at komponenten kan fungere. For mye slark i den retningen vil kunne gå utover kraftoverføringen. Det er ønskelig at det kun er en akse langs stemplet som skal bevege seg.

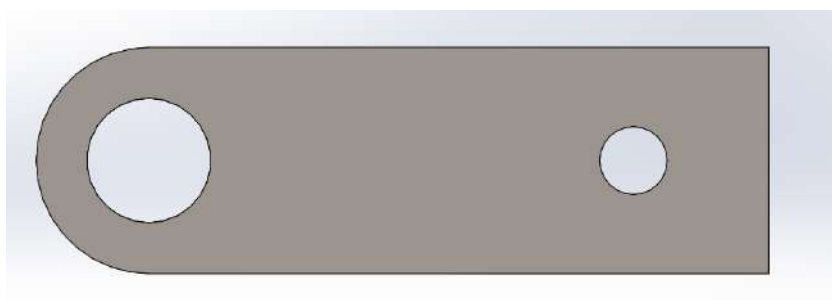
12.5.4 Andre Utkast



Figur 174: Gir: Isometrisk synsvinkel



Figur 175: Gir: Sett ovenfra



Figur 176: Gir: Sett fra siden

En essensiell forandring er at hengselsleddet fra LA14 til komponenten er i parallell med leddet til punkt C.

En annen viktig forandring er at aktuatorleddet er flyttet helt til høyre 174. Da delen skulle tilpasses ATV-en sitt giringsystem, ble det oppdaget at delen kunne kræsje med noe av plastikkdekslet til ATV-en (figur 177)

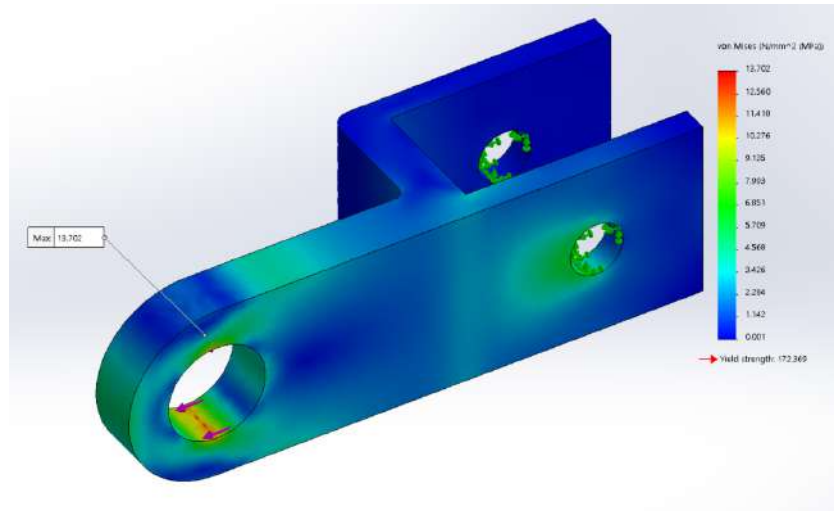


Figur 177: Girdekslet er festet rett ved punkt C i det merkede området.

Dekselet vil man helst unngå å fjerne da mange mellomledd må fjernes først som blant annet plastikkdekslet til ATV-en. Da forsvinner også mye av beskyttelsen til ATV-en sine mekaniske komponenter.

12.5.5 Evaluering

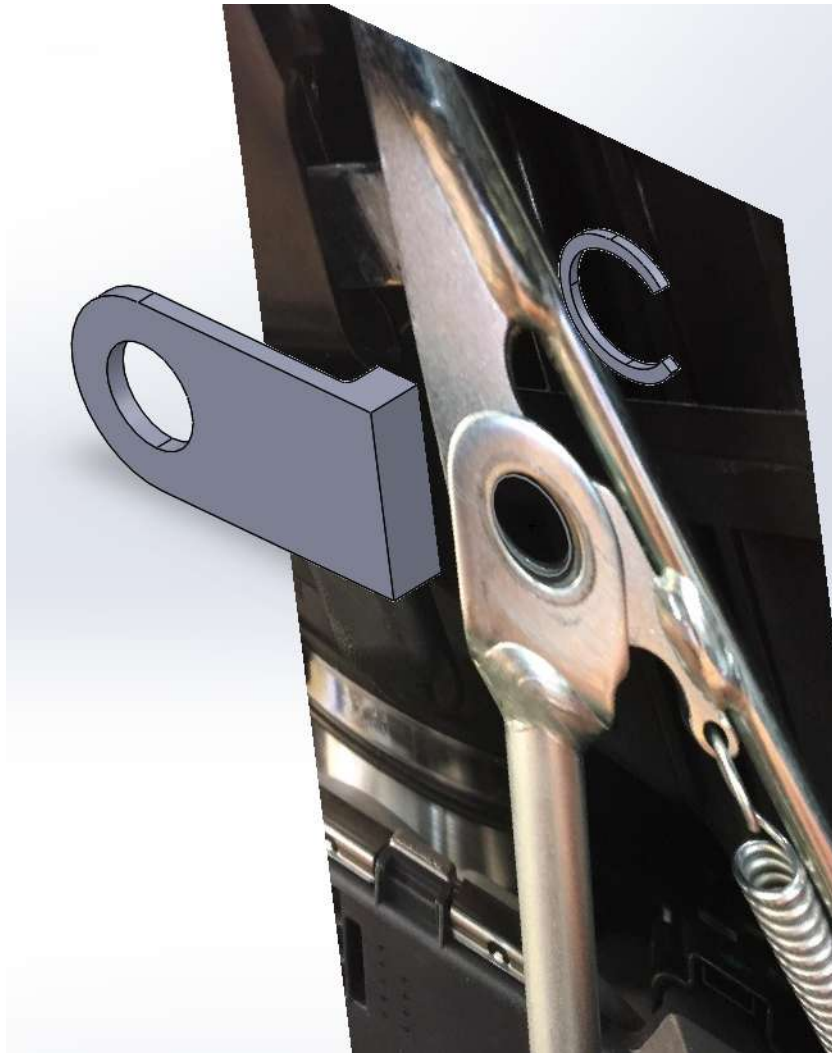
Denne komponenten har hengslene parallelt med hverandre istedenfor vinkelrett. Dette fører til bedre bøyespenninger i aktuatoren og komponenten, som vil føre til økt levetid. Aktuatorhengelet er flyttet til høyre slik at det er større sjanse for at den passer inn uten å måtte demontere mer av dekslet til ATV-en. Det er antatt at spenningene forblir veldig lave som vist i Fem analysen ved figur 178.



Figur 178: Maksspenningene er 13.7 MPa og deformasjonsgraden er 1 mikrometer.

En nøkkeldimensjon på dette designet er avstanden fra hengselet, punkt C, til innerveggen på delen. Det er innerveggen som skal avstive konstruksjonen, slik at stempelet til girspaken kun har et bevegelig hengselledd.

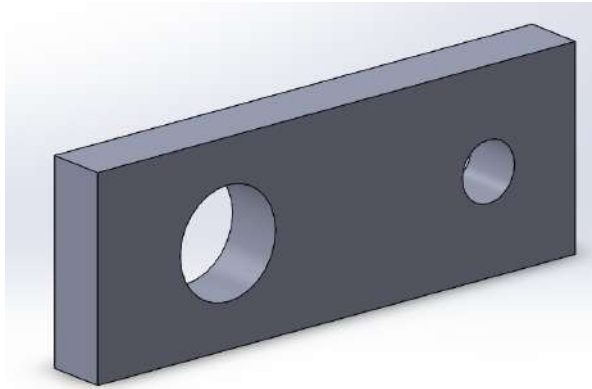
Det er 2 mm sikkerhetsmargin fra veggen til hengslet. Når girspaken flytter seg, vil vinkelrelasjonene mellom de to stagen bevege seg. Det er behov for noe klaring, ellers kan ikke girspaken forflytte seg (figur 179).



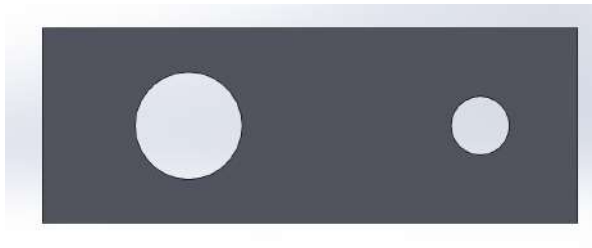
Figur 179: Stagens posisjon i forhold til hverandre og delen vil endre på seg når girspaken er i bevegelse

Siden punkt C er et ledd som beveger seg langs en kurve vil passformen forandre seg litt. Derfor kreves det litt slakk slik at den tåler forandringene.

12.5.6 Tredje Utkast



Figur 180: Gir: Isometrisk synsvinkel



Figur 181: Gir: Sett fra siden

Dette designet har kun en plate med tykkelse på 6 mm. Det kan brukes en stålplate som vannskjæres.

Delen har rektangulære former istedenfor avrundinger konsentrisk til festepunktet til aktuatoren. Det er for å stive opp hengselet til LA14 stempelet. Det blir lavere spenninger, fordi arealmomentet blir mye større. Det vil dermed bli mer solid i forhold til forrige modeller, slik at den tåler mer. Den har bare en flate, som gjør at den ikke kræsjer med plastikkdekslet til ATV-en.



Figur 182: De blå tverrsnittene viser hvor LA14 blir utsatt for bøyespenninger.

Dette designet tar elementer fra tidligere modeller ved at den stiver opp modellen slik som i første utkastet. Den gjør det ved at den utnytter arealmomentet til aktuatoren bedre.

12.5.7 Evaluering

Dette designet er bedre fordi den er billigere, kolliderer ikke med ATV-en og stiver opp hengsleddet til LA14 stemplet. Da blir det kun to bevegelige hengsler i kraftoverføringen. Det gjør at kraftoverføringen blir presis og det er lite slakk i designet som gir et godt utgangspunkt for posisjonssensoren.

12.5.8 Tiltak for posisjonsregistrering

Det vil være nødvendig å se i hvilket gir ATV-en står i. Derfor har det blitt bestilt et ekstra girdeksel. Det gir mulighet til å kutte i sporene og merke av i hvilken posisjon giret er i.



Figur 183: Dekselet er sagd slik at det er klarere bane for LA14

Sporene på dekselet, som vist i figur 183 er fjernet. Aktuatoren skal beveges langs en akse og låse av posisjon til girspaken når den ikke er i bruk. Det fører til at sporene i dekselet ikke lenger er nødvendige.



Figur 184: Dekselet er sagd slik at det er klarere bane for LA14

Fjæren gjør ikke så mye motstand, men skaper litt spenning da den trekker spaken inn mot sentrum av ATV-en. Den kan derfor fjernes 184.

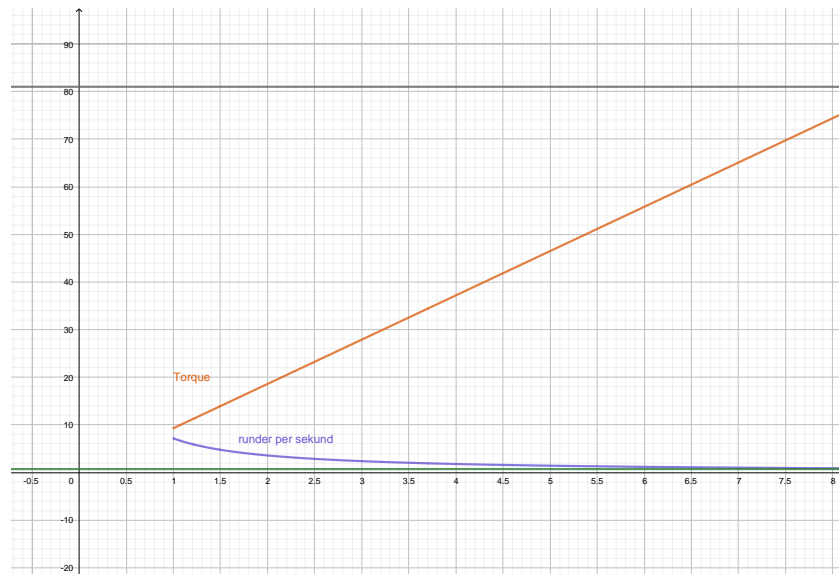
12.6 Design av styre

Basert på konseptet som ble valgt i kapittel 9.3.5, skal det designes en løsning som skal overføre rotasjon til styret. Dette kan gjøres ved å plassere en servomotor på toppen av styreakselen. For denne løsningen må det finnes ut av hvilken servomotor som skal kjøpes, hvilken kraftoverføring og innfestning den skal ha.

For å etablere hvilken motor som var mest aktuell, ble det studert forskjellige kilder fra diverse nettsider og opprettet dialoger med teknisk hjelp i ulike bedrifter for rådføring.[35] [36]

Festo ble presentert med hvilke krav som kreves for ytelsen i styringssystemet. Disse kravene inneholder minimumsyttelse om hastighet, kraft og sensorer for kontrollering 6.5.5.

Etter dialog med Salgsingeniør Øyvind Selseng i Festo ble det funnet en steppermotor som kan være aktuell. Den heter EMMS-ST-87-L-SE-G2, size 87-L og yter 9.3 Nm og 430 RPM og kan styres på 24V [56]. Ved å bruke et giringsystem blir det mer torsjon på bekostning av rotasjonshastighet. For eksempel, 2:1 girning gir dobbelt så stor torsjon fra akselen, men motoren roterer halvparten så fort. Ved bruk av programmet geogebra ble det laget en matematisk graf for å finne ut hvilken girning som trengtes til systemet. (figur 185)



Figur 185: Rød strek er Torsjonen til motoren og blå strek er rotasjonshastigheten

Giringsystemet må velges i heltall siden det er dette Festo tilbyr. Det er en liten grønn strek parallelt og nærme den horisontale akse. Den representerer en rotasjonshastighet på 90 grader per sekund, som er minimumskravet til systemet. Det betyr at den blå linjen må ha høyere verdi enn dette for å tilfredstille kravet. Det samme gjelder den horisontale linjen som har en verdi på cirka 80. Dette representerer ønsket torsjon fra servomotoren i denne løsningen. Tallet er i virkeligheten litt mer fleksibel, da den integrerte servoen i ATV-en vil assistere. Ved 10:1 gir blir det 0,72 rotasjoner per sekund, mens behovet er

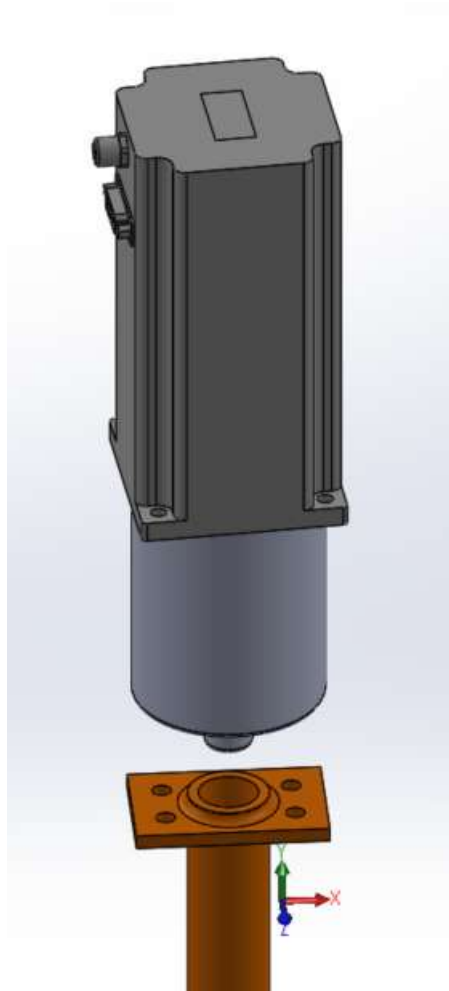
0,25 rotasjoner per sekund. Dette gir en torsjon på 93 Nm, som er tilstrekkelig.

Det Festo hadde tilgjengelig av gir som var kompatibelt med deres stepper motoren [56] var gir med 9:1 utveksling. Informasjonen ble plottet inn i grafen laget i Geogebra[26] (filen heter 9Til1Gir.PNG i Google-driven). Det viste seg at det gir en ytelse på 83.8 Nm og vinkelhastighet på 0.79 rotasjoner per sekund. Det tilfredsstillte kravene som er satt 6.7.1

12.6.1 Kraftoverføring til styre

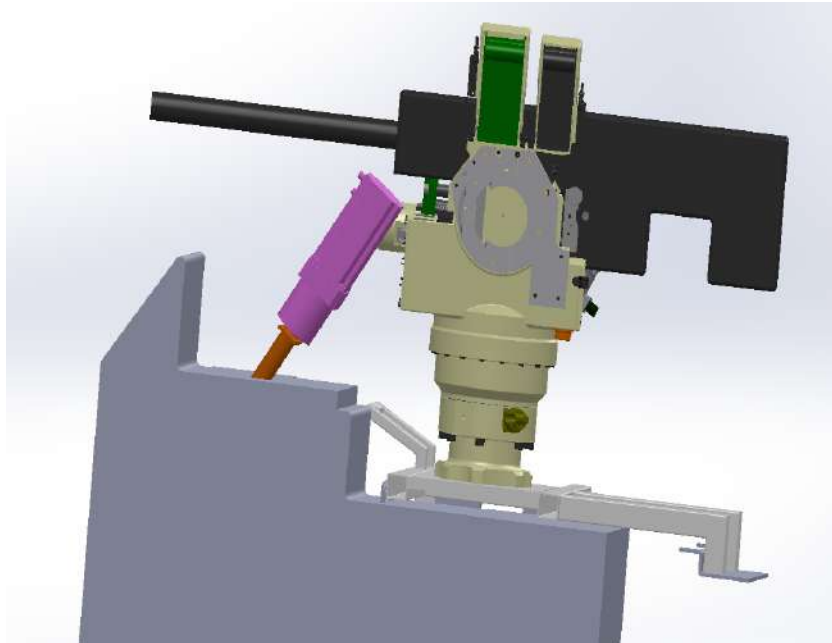
Motor koblet direkte på akslingen

En løsning er å plassere servomotoren på toppen av styreakselen slik at akselen til motoren og styret er konsentrisk og co-lineære med hverandre slik som figur 186.



Figur 186: Slik ser systemet se ut dersom det monteres på styreakselen

Ut ifra konseptet (Fra figur 43) skulle dette være den endelige løsningen. Etter å ha funnet en steppermotor til løsningen, innsees det at motoren og giret har en samlet lengde på over 325 mm. Det gjør at denne løsningen blir re-evaluert i forhold til de andre kraftoverføringskonseptene. I bildet under 187 illustreres hvordan våpenstasjonen kolliderer med servo motoren om den skulle bli implementert.

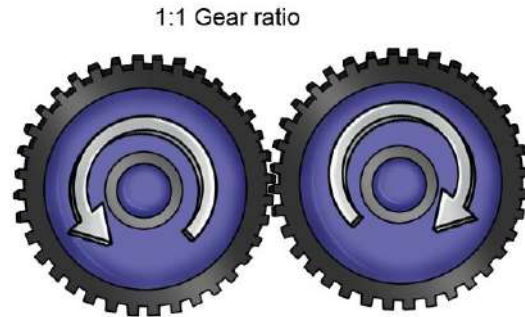


Figur 187: Modell: Styre - skisse

Figur 187 viser hvordan systemet ser ut dersom det monteres på styreakselen.

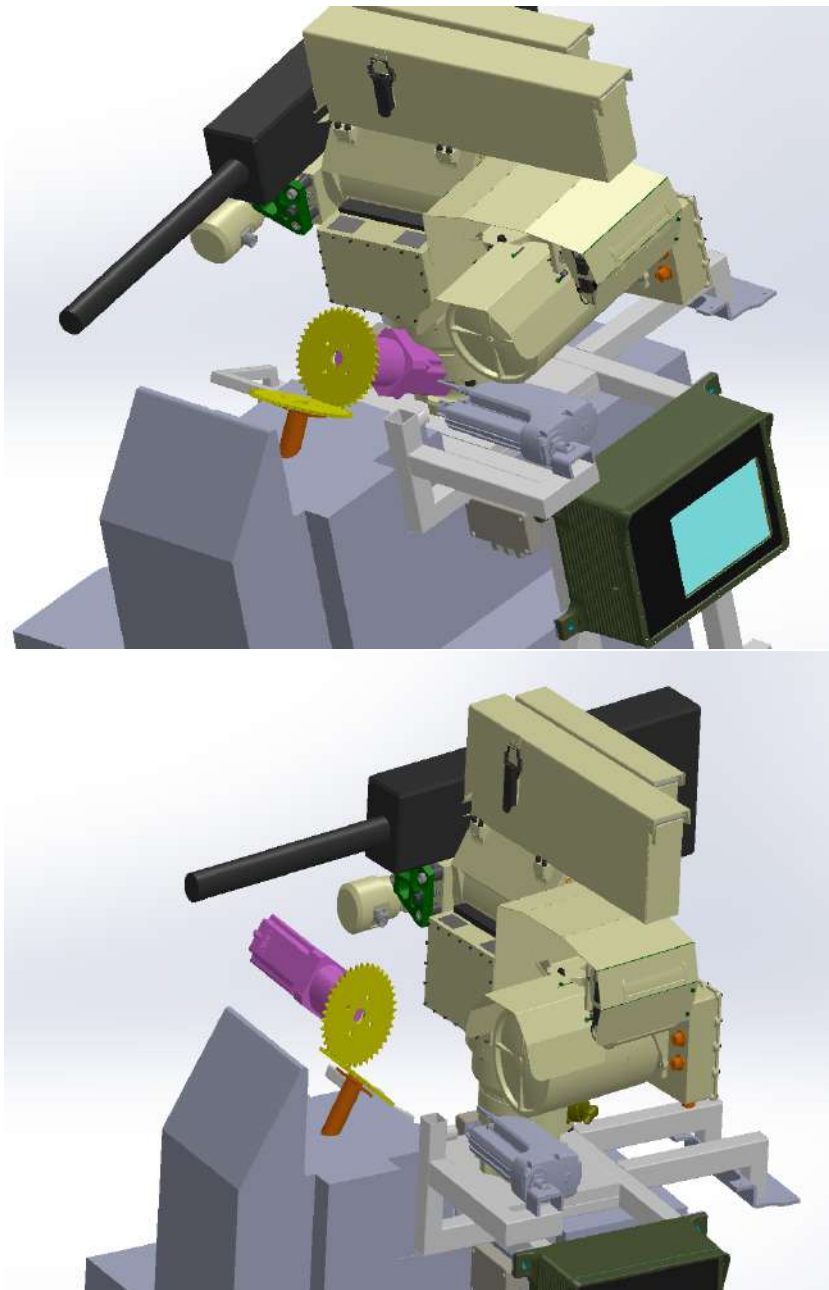
Konsept - kraftoverføring med tannhjul

Man kan koble akslingene med et tannhjul i motorakselen og et i styreakselen.



Figur 188: Tannhjul 1:1 forhold

Dette gir presis styring og lite slakk i systemet. Det gir ikke stor nok distanse til at servomotoren kan bli plassert langs ATV-en. Å plassere servomotoren ut fra ATV-en krever høy innfestning som tar mye plass 193. Det kreves også nøyaktig justering for å få god nok kontakt med tannhjulene.



Figur 189: Ulike konsepter for Tannhjulløsning

Figur 191 viser alternativer til oppsett ved bruk av tannhjulløsning. Første figur kolliderer inn i våpenstasjon, noe som ikke er fysisk mulig i virkeligheten.

Koniske tannhjul

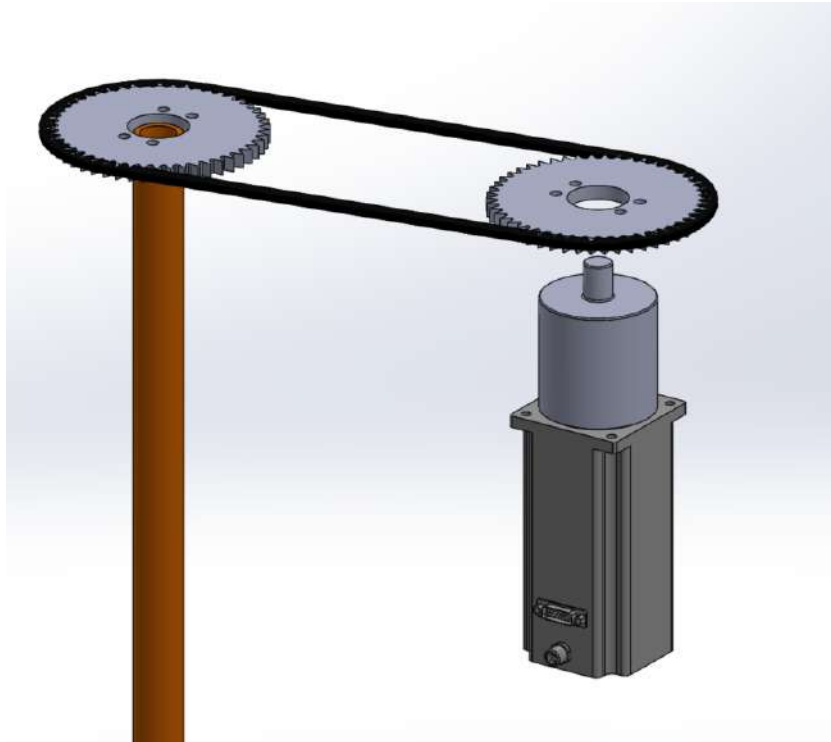
I likhet med det første tannhjulkonseptet blir styringen presis siden det er lite slakk.



Figur 190: Plassere koniske tannhjul med 1:1 forhold i 45 graders vinkel. Bildet hentet fra [40]

Servomotoren må plasseres i samme høyde som styreakselen med mindre flere komponenter legges til i kraftoverføringen og de koniske tannhjulene har 45 graders vinkel. Da blir innfestningen fra rammen til servomotoren en større statisk innretning enn andre løsninger som illustrert i figur 189.

Kjededrift



Figur 191: Assembly av styresystem med kjededrift mellom aktuator og styreakselen

Fra boken [81] er det listet opp fordeler og ulemper ved bruk av kjededrift i forhold til belte og taudrift.

Det er antatt lite slakk i kjede under disse operasjonsforholdene med lav rotasjonshastighet. Det betyr lite vedlikehold og behov for stramming av kjede.

Motoren kan plasseres vertikalt nedover langs ATV-en ved høyre benplass der ingen andre aktuatorer befinner seg. Det gjør løsningen gunstig med tanke på plassering.

Lenker kan tas av og på slik at tilpasningen er lettere.

Det er en sannsynlighet for at noe kan treffe kjede. Dette kan skape behov for noe beskyttelse for å dekke over kjededrevet for sikkerhetsmessige grunner.

Beltdrift

Svært lik kjededrift, der forskjellen er belte istedenfor kjede. Denne teknologien har be-

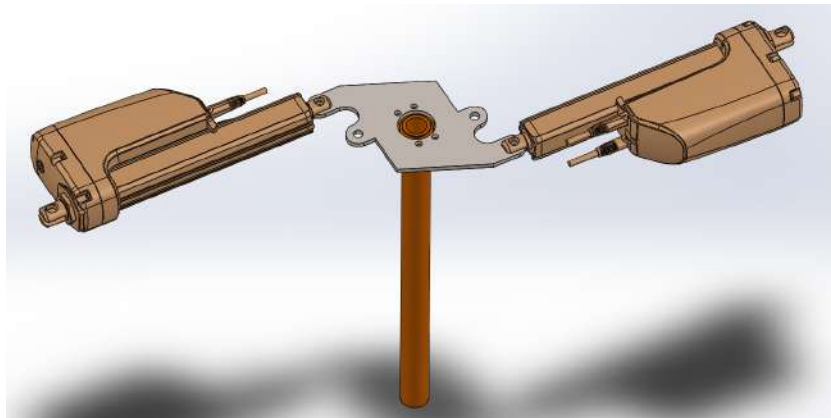
gynt å etablere seg i sykkelmarkedet. 191 Den vil følge liknende eller samme struktur som kjededrift.

Det er vanskeligere å finne passende lengder på gummibelter, da disse ikke er justerbare på samme måte som kjededrift kan fjerne lenker. Belter er mer sårbare for kutt og slag enn hva kjeder er.

Potensielt skadeomfang er lavere om beltet ryker i forhold til at noe skjer med kjede.

Bruk av stag

Man kan bruke 2 lineæraktuatorer som opererer i parallell for å styre ATV-en. Dette kan utføres ved å feste et stag ut fra styreakselen.



Figur 192: Et design av hvordan en stagløsning kan se ut.

Her må to motorer samarbeide for å generere like stort moment samtidig, noe som er utfordrende å få presist fordi forholdene må være helt identiske. Fjærer på stemplene kan kompensere for forsinkelser og ugunstige arbeidsposisjoner for aktuatorene.

Det som er bra med denne løsningen, er at samme aktuator blir brukt i de andre delsystemene. Dette gjør at løsningen blir lett å implementere siden det er tidligere erfaringer på dette innad gruppen. Ytelsen er derimot ikke tilstrekkelig i forhold til fart og kraft, se Rapport på arm og kraft.A.11

Konklusjonen er at denne løsningen skal ikke benyttes som en fast løsning, men kunne bli brukt i testfasen hvis den originale løsningen ikke fungerer helt til en annen løsning

er klar.

Endelig design

Det endelige designet ble servomotor med kjededrift for kraftoverføring.

Forskjellige løsninger ble diskutert med Thorbjørn Bråthen fra Bedriften KJELL BERG A/S som holder til i nærheten av Skien, Norge [54]. Disse rådrevene blir hovedsakelig brukt på motorsykler. Tunge motorsykler yter med torsjoner i skalaen 100-120 Nm. [65] [66]

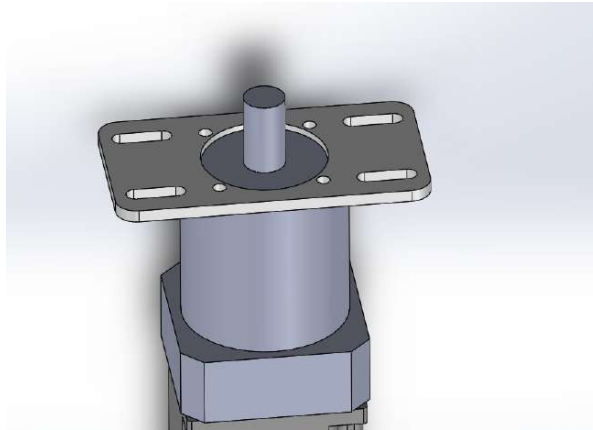


Figur 193: 2 stk Rådrev [54] ble kjøpt inn

Maksverdiene til styret under prosjektet vil være 81 Nm. Basert på dette vil tannhjulet tåle påkjenningene det blir utsatt for under alle operasjonsforhold.

Tannhjulene måtte være i 1:1 forhold fordi girforholdet allerede er fikset i girsystemet fra Festo.

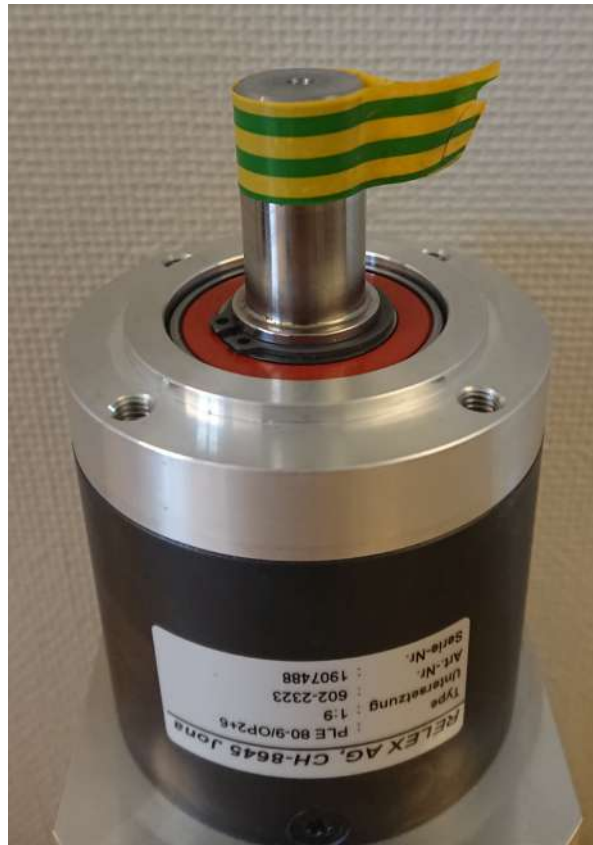
To rådrev med diameter på 170 mm og to stk kjeder på over 1.33m ble bestilt. Lenker på kjedene kan tas av etter behov for tilpasning. En skyvbar plate kan strammes for å justere lengden tilsvarende en lenke.



Figur 194: Boltene kan bevege seg frem og tilbake langs hullet for å stramme kjedet ved behov.

Samhandling mellom komponentene

Delsystemet består av motor med gir, tannhjul, kjede, en skyvbar plate og styreaksel. Tannhjulene må tilpasses; Det ene tannhjulet skal festes på aktuatoren og den andre på styreakselen. På aktuatoren stikker det ut en aksel på 35 mm med en diameter på 20 mm hvor det ene rådreivet skal festes.



Figur 195: Girtupp

Det er ikke ønskelig å skade giret som er kjøpt inn. Derfor må det lages en komponent som kan koble girakselen og tannhjulet uten å permanent endre girsystemet.



Figur 196: En kobling brukes for å koble systemet sammen

Vår versjon av overgangen

Utfyllende rapport med alle låsinger, kontakter, mesh og resultat ligger som vedlegg A.10

Til styringen kreves en overgang fra Utvekslingsgiret til tannhjulet som skal drive kjedet. Nedre del av denne overgangen ble designet for å passe på akselen til utvekslingsgiret, denne delen av overgangen ble derfor designet lik enden av utvekslingsgiret som fester giret til steppermotoren, ettersom dette fungerer godt.

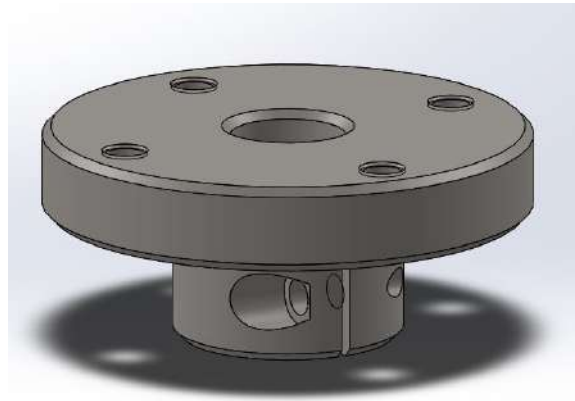
Den andre enden ble designet for å passe på tannhjulet, med gjenger for å kunne feste tannhjulet med bolter. Derfor ble det lagt inn en hullsirkel i overgangen som passer med hullsirkelen som er maskinert i tannhjulet. siden det skal være gjenger i overgangen, ble flensen med gjenger designet tykkere enn nødvendig med tanke på spenninger i delen, for å få lange nok gjengepartier. Denne flensen kunne vært tynnet ned rundt gjengepartiene, men dette ville ført til betydelig økning i produksjonspris, derfor forblir den slik den er.

Den andre enden ble designet for å passe på tannhjulet, med gjenger for å kunne feste tannhjulet med bolter. Derfor ble det lagt inn en hullsirkel som passer hullsirkelen i tannhjulet. siden det skal være gjenger i overgangen, ble flensen med gjenger designet tykkere enn hva den hadde trengt for å få lange nok gjengepartier. Denne flensen kunne vært tynnet ned rundt gjengepartiene, men dette ville ført til betydelig økning i produksjonspris, derfor forblir den slik den er.

FEM analyse av overgangen

Oppsettet til denne analysen består av å legge til nødvendige låsinger av delen, slik at den ikke kan bevege seg uønsket, Detetter må kreftene den blir utsatt for defineres, så må mesh-et defineres. (se vedlegg A.10 for tekniske detaljer om låsingen, momentet og mesh-et)

Ettersom flensen med gjenger ble så tykk blir det ikke utviklet store spenninger i denne delen. Fordi parametrene er satt ved måling av originaldel og fastsatt pga lengde på gjengeparti ble det derfor ikke nødvendig å endre på designet, selv om den vil være noe overdimensjonert.



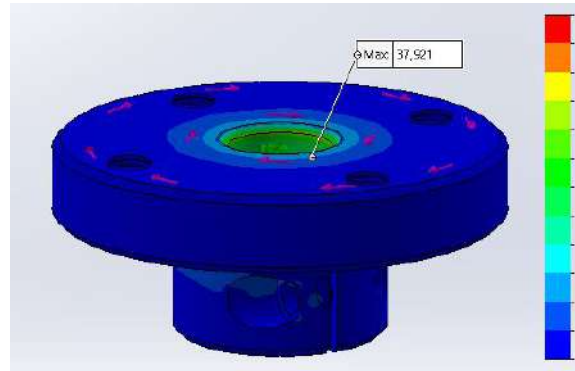
Figur 197: Overgang fra utvekslingsgir til tannhjul

Oppsettet til denne analysen består av å legge til nødvendige låsinger av delen, slik at den ikke kan bevege seg uønsket, Detetter må kreftene den blir utsatt for defineres, så må mesh-et defineres. (se vedlegg A.10).

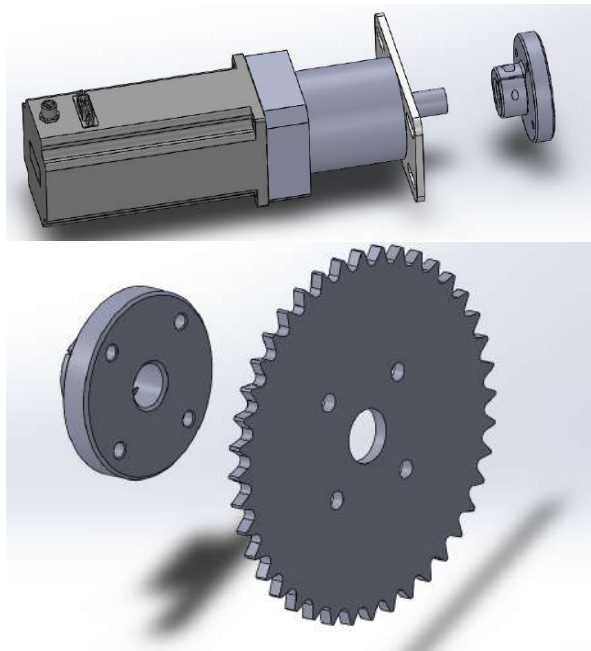
Ettersom flensen med gjenger ble så tykk blir det ikke utviklet store spenninger i denne delen. Fordi parametrene er satt ved måling av originaldel og fastsatt pga lengde på gjengeparti ble det derfor ikke nødvendig å endre på designet, selv om den vil være noe overdimensjonert.

Sammenkobling med styret

Dette er en vanlig metode for å koble seg på en motoraksel uten festepunkt.



Figur 198: Fem av Overgang fra utvekslingsgir til tannhjul

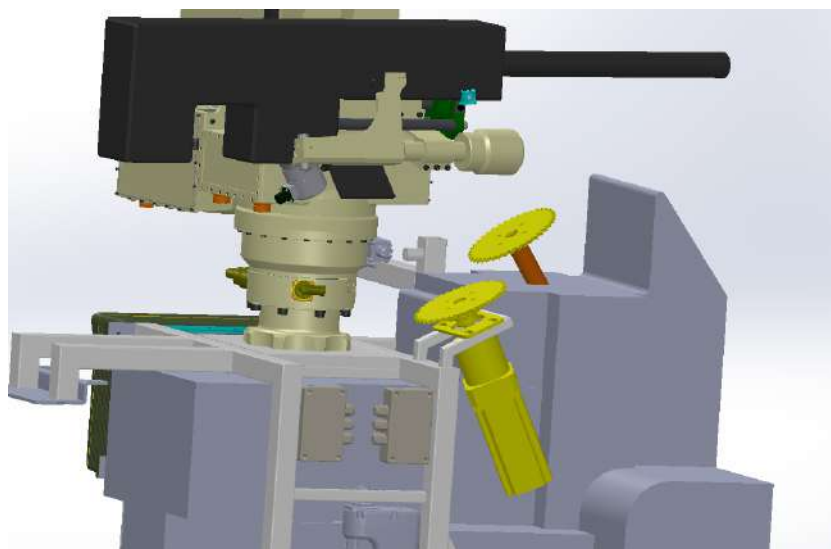


Figur 199: komponenten festes på girakslingen og videre til rådreivet.

Å plassere det andre tannhjulet på styreakselen lar seg gjøre ved å benytte seg av de allerede eksisterende bolthullene i styreakselen.



Figur 200: Designet av hele styresystemet for seg selv.



Figur 201: Designet av hele styresystemet sammen med ATV.

Figur201 er hvordan hele systemet skal se ut til slutt. Det tilbyr presis overføring av kraft og ivaretar rotasjonsposisjonen gjennom hele overføring. Det er tilrettelagt til at det er plass til andre delsystemer som gir. Løsningen gir stor bevegelsesfrihet til våpenstasjonen til å rotere.

Test

Her vil det bli gjort tester på de forskjellige delsystemene.

Innhold:

- Tester - Gass
- Tester - Brems
- Tester - Gir
- Tester - Styring
- Tester - Hardware
- Tester - Software



KONGSBERG



USN Universitetet
i Sørøst-Norge

Under prosjektet må det utføres forskjellige former for testing. Det kan deles inn i forskjellige faser. Innledningsvis, i første fase, er det ønskelig med simuleringer fordi det er billig og gir et innblikk i konsepter og verifiseringer. På det andre stadiet skal det oppskaleres til mindre fysiske tester som gjerne skjer med løsninger som ikke er ment for å vare. Det kan være å finne billige generiske versjoner, små fjernstyrte biler osv. å teste med. På den måten vil man få verifisert tydeligere og med en mer nøyaktig testing av systemene.

- Individuell kvalitet skal blant annet hjelpe til å sikre at komponentene virker som de skal, og gi spesifikasjoner over hva det kan potensielt yte.
- Grensesnitt testing skal demonstrere samhandlingen mellom to eller flere komponenter/software og se hvordan de fungerer sammen som et systemet og i forhold til hovedsystemet.
- Sikkerhet er ansett som svært viktig i dette prosjektet. Da intensjonen er at det kun skal demonstreres og driftes i gangfart for å vise at fjernstyringssystemet fungerer. Dette skal designes slik at det tåler hastigheter oppmot 50 km/t på sikt. ATV-en har 59 hestekrefter og selv om systemet ikke skal testes for større krefter, må man likevel respektere kraften og den potensielle skaden som kjøretøyet besitter om det skulle komme ut av kontroll. ATV-en skal testes innenfor et inngjerdet område i teknologiparken.

Nedenfor er det satt opp en tabell som vil brukes for testing. En slik test kan teste mange krav, men et krav kan også bli testet mange ganger. Hver test vil få en særegen ID, en hovedansvarlig for testen og en overordnet testansvarlig. Her skal det fylles inn med krav ID-er og hva som testes. Etter endt test vil den få en status hvor det fylles inn godkjent, delvis godkjent eller ikke godkjent.

Tabell 32: Tabell for test

Test ID:	Ansvarlig	Godkjent av
Dato:		
Krav:		
Verifikasjonsmetode:		
Testobjekt:		
Hvilken funksjon skal testes:		
Status:		

Det vil bli kjørt forskjellige tester i løpet av prosjektperioden. Disse kan bli delt opp i tre kategorier.

- **Teoretisk -TT** Matematiske beregninger og simuleringer
- **Praktisk -TP** Tester utført på hardware
- **Implementering -TI** Tester utført på ATV.

Testkoden blir generert ut ifra type test, nummeret på testen og hvilket hovedkrav det omfatter. F.eks vil TT.4-O.08, bety at det er en teoretisk test nr. 4 som omfatter krav O.08.

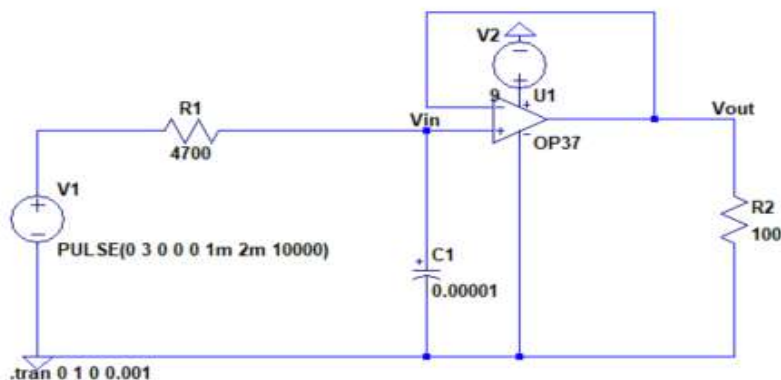
13.1 Gass

13.1.1 PWM til DC spenning

Tabell 33: Test gass - PWM til DC spenning

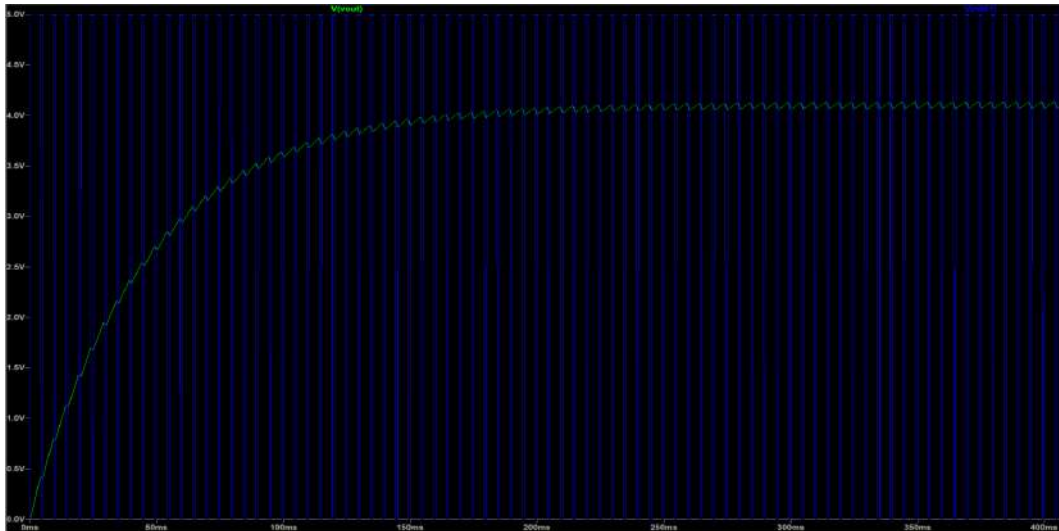
Test ID: TP.3-GA.01	Ansvarlig SAM	Godkjent av SAM
Dato:	05.03.2019	
Krav:	GA.01	
Verifikasjonsmetode:	Måling	
Testobjekt:	Krets	
Hvilken funksjon skal testes	PWM til DC spenning	
Status:	Godkjent	

For å verifisere måleresultatene, ble det gjennomført en simulering av den aktuelle kretsen. LTspice ble brukt for å simulere kretsen. Operasjonsforsterkeren brukt under målingene LM358p var ikke listet opp i programmet, men etter flere simuleringer med forskjellige operasjonsforsterkere ble resultatet verifisert. Siden formen på utgangssignalet var like for alle operasjonsforsterkerne og kun responstid, ripple og amplitude var forskjellig. Figur 233 viser kretsen i LTspice.



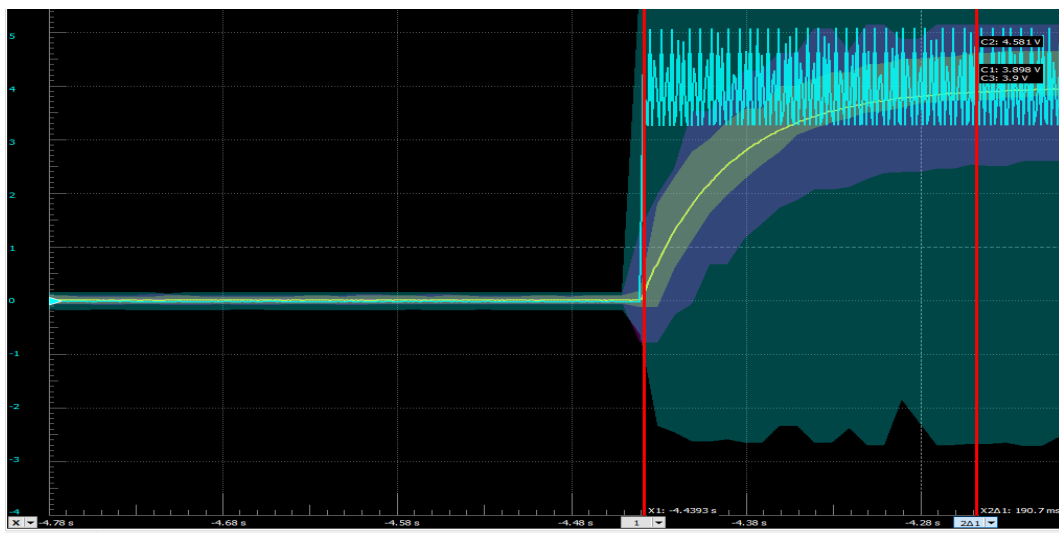
Figur 202: Krets til T1-GA.01

Resultatet av simuleringen kan ses i figur 203.



Figur 203: Resultatet av simuleringen av T1-GA.01

Denne testen ga ut godkjente resultater i forhold til målingene som ble gjort på gassensoren. Denne kretsen vil senere bli brukt til å emulere sensor 1 og sensor 2 på ATV, slik at gassen blir styrt fra en Arduino. Denne kretsen gjør om PWM (Pulse Width Modulation) signaler om til analoge DC verdier. Det ble gjort målinger med oscilloskop, samt Arduino for å bekrefte at kretsen gjorde det den skulle. Dette er en enkel krets med ett lav-pass filter, som glatter ut spenningen og en spenningsfølger som sørger for beskyttelse av Arduino. Spenningsfølgeren sørger også for at det blir tilstrekkelig strøm til resten av kretsen. Denne er ikke nødvendig i testen, men vil bli nødvendig for implementeringen av ATV.



Figur 204: Måling med oscilloskop

Ut ifra målingen med oscilloskop kom det frem at responstiden er på 190 ms, mens tidskonstanten ligger på 45 ms. Dette er akseptable verdier da det ikke er behov for å gasse med maksimalt pådrag så fort.

Tabell 34: Test av DAC Breakout Development Board Module

Test ID	Ansvarlig	Godkjent av
TP.15-G.01	SAM	SAM
Dato:	30.04.2019	
Krav:	G.01	
Verifikasjonsmetode:	Måling	
Testobjekt:	DAC	
Hvilken funksjon skal testes:	PWM til DC	
Status:	Ikke Godkjent	

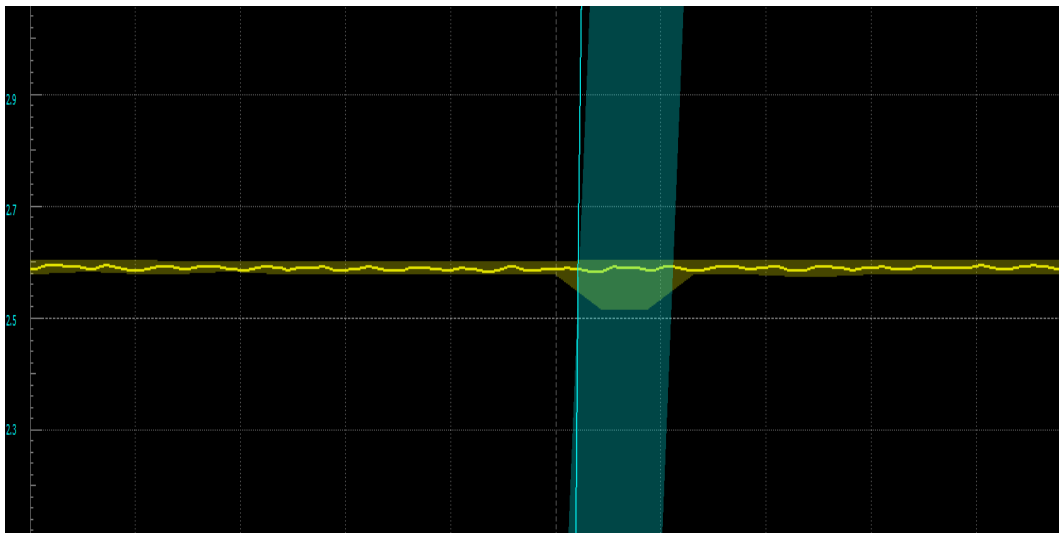
DAC ble mottatt 30. mai og testet samme dag. Den ble koblet opp og testet med EE-board på samme måte som TP.3. Det viste seg at denne DAC-en ikke ga ønsket resultat, og det er mer hensiktsmessig å bruke egen krets til dette. Kretsen må dog utbedres noe for å glatte ut spenningen enda mer.

Tabell 35: Test gass - PWM til DC spenning

Test ID	Ansvarlig	Godkjent av
TT.16-GA.01	KVL	KVL
Dato:	02.05.2019	
Krav:	GA.01	
Verifikasjonsmetode:	Simulering og fysisk test	
Testobjekt:	Glattekrets	
Hvilken funksjon skal testes:	PWM til DC	
Status:	Godkjent	

Ved forrige test (TP.3-GA.01) hvor PWM til DC spenning konverter ble fysisk testet, var det for mye ripple i utgangssignalet. Det skyldtes for høy kondensatorverdi, da det kun ble testet for godkjent responstid.

Etter å ha testet med en simulering i LTspice, ble det brukt en kondensatorverdi på $10\mu F$. Slik at både responstiden og ripple-verdien er godkjent.



Figur 205: PWM til DC revidert

Tabell 36: Gassfunksjon på ATV

Test ID: TI.27-GA.01	Ansvarlig SAM	Godkjent av SAM
Dato:	16.05.2019	
Krav:	GA.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	ATV	
Hvilken funksjon skal testes	Gass på ATV	
Status:	Godkjent	

Fra test TT.16-GA.01 ble kretsen tatt videre til implementering på ATV. Etersom Det ble først bekreftet ved hjelp av EEboardet at spenningene ut fra gass-sensoren på ATV-en ga ut 3V og 1.5V, som gjort under måling.

En av utgangene til opampen koblet til signalsensor 1 på ATV. Tenningen ble så skrudd på mens ATV sto i park. Da kom det en feilmelding på displayet til ATV, som varslet om at den ikke fikk tilbakemelding fra sensorene.

Dermed ble begge utgangene fra opampen koblet på signal sensor 1 og signal sensor 2, sammen med jording til de to jordingssensorene. Dette gjorde at ATV-en ikke viste feilmelding og det kunne gasses som antatt med R2 som de foregående testene hadde vist.

Det var stadig problemer at pinner datt ut og det var dårlig tilkobling under testen. Derfor ble det bestemt at det skulle loddet ett prototype kretskort til dette. Dette vil sørge for at det er både mer kompakt og robust.

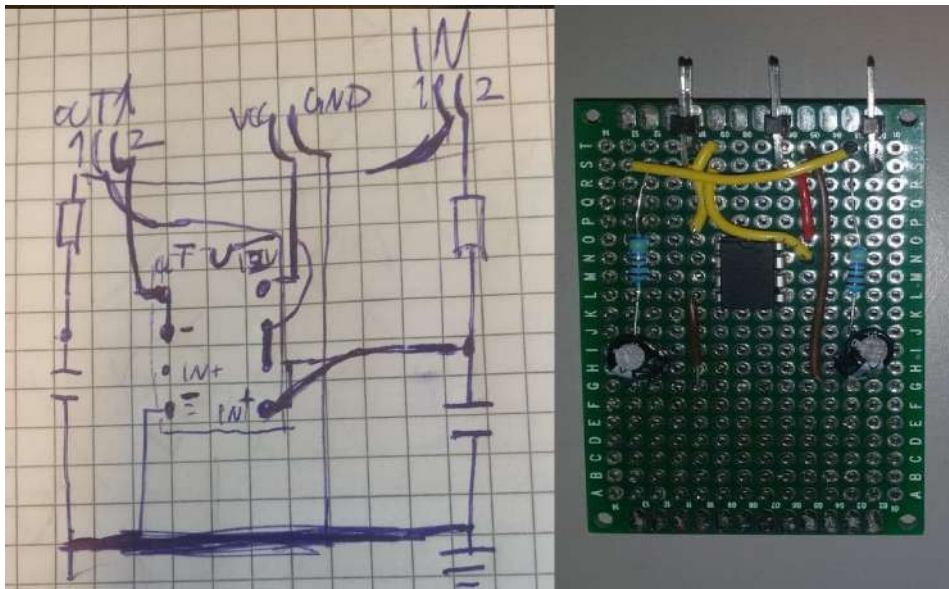


Figur 206: Bilde fra første implementerings-test

Tabell 37: Gassfunksjon på ATV

Test ID: TI.28-GA.01	Ansvarlig SAM	Godkjent av SAM
Dato:	18.05.2019	
Krav:	GA.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	ATV	
Hvilken funksjon skal testes	Prototype PCB for gass	
Status:	Godkjent	

Første forsøk med å produsere en prototype av en PCB skulle vise seg å bli en dårlig ide, da det var vanskelig å holde oversikten, selv i en liten krets som dette. Det resulterte i ett brett som ikke fungerte som det skulle. Det ble derfor skissert opp en plan om PCB layout som vist i figur 207, før det ble loddet på nytt. Dette gjorde loddingen mye bedre, og resulterte i ett mer oversiktlig og fungerende kort.



Figur 207: Skisse til PCB sammen med resultat

Etter loddingen ble kortet testet for funksjonen i Waveforms ved hjelp av EE boardet. Da det ble startet opp første gangen kom noen firkantpulser, som forsvant etter ca ett minutt. Dette kan være fordi kondensatorene ikke ble ladet ordentlig opp. Ellers fungerte kortet som det skulle.

13.2 Brems

13.2.1 Lineæraktuator

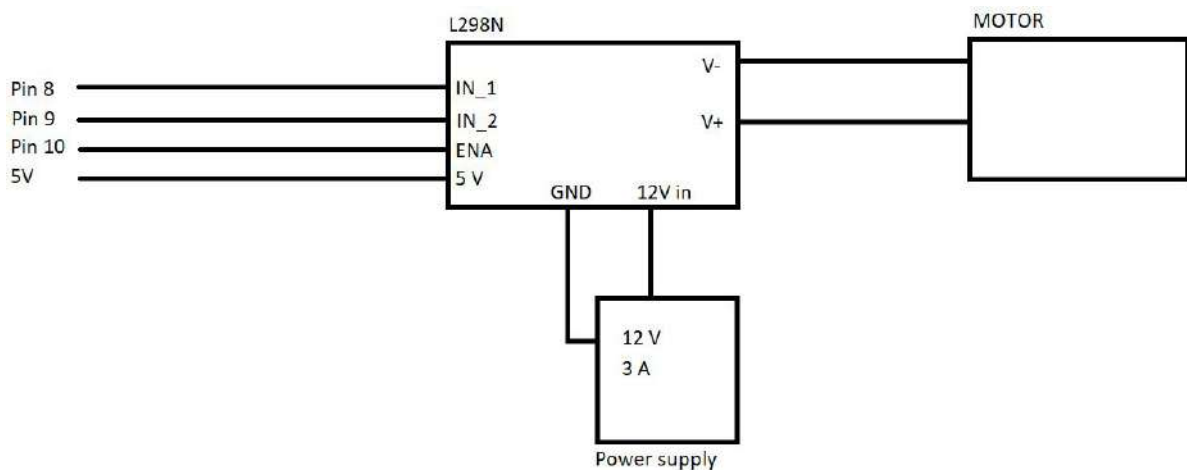
Utstysrliste:

- Power supply: EA-PS 5080-10 A
- DC-motor driver: L298N
- Arduino mega 2560
- Linak lineæraktuator: 121000-21301220

Tabell 38: Test lineæraktuator

Test ID	Ansvarlig	Godkjent av
TP.9-GI.01	OL	OL
Dato:	20.03.2019	
Krav:	BR.01 - GI.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	LA12 - testaktuator	
Hvilken funksjon skal testes:	Bevegelse/hastighet	
Status:	Godkjent	

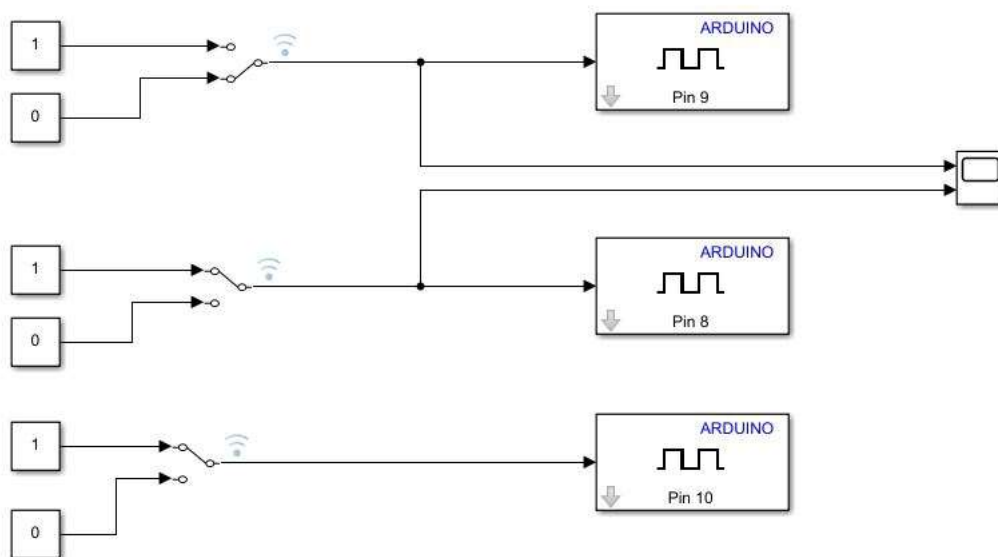
Her vil det bli foretatt en liten test på hvordan lineæraktuatoren sammen med driver fungerer. Kobler først og fremst opp kretsen via Arduino -> Driver -> Aktuator. Deretter ble matlab - simulink konfigurert for å kunne samhandle med Arduino.



Figur 208: Overordnet koblingskjema for Arduinoen i samhandling med driver og motor

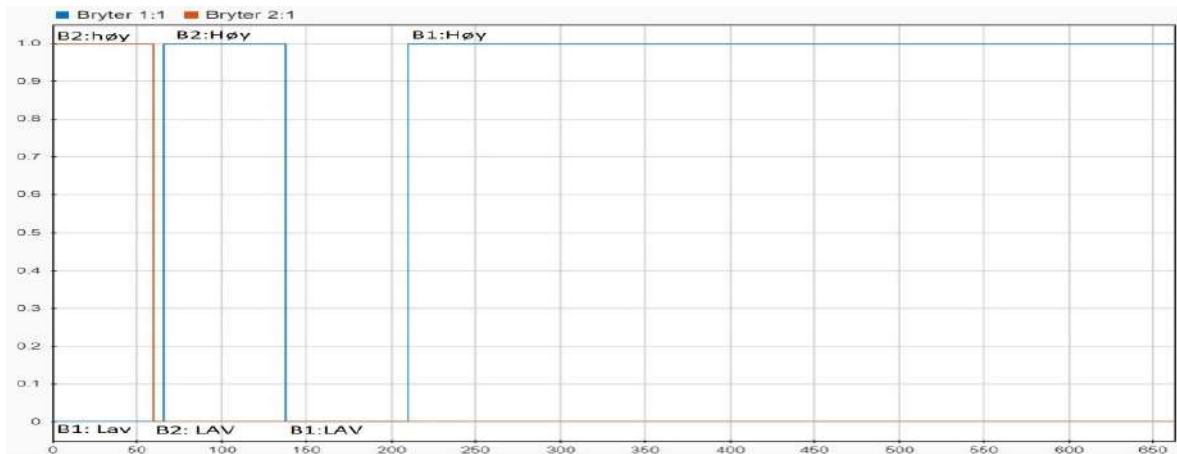
Figur 208 viser koblingsskjemaet for Arduinoen og driveren. Det trengs 2 styringsporter inn i inngang "in_1" og "in_2" for å bestemme retning på aktuatorer; digitale signaler "høy" og "lav" i respektive rekkefølge vil føre til at lineæraktuatoren går fremover. Motsatt ("lav" og "høy") vil føre til at aktuatoren reverserer. "ENA" bestemmer farten på aktuatoren. Motoren er avhengig av en konstant spenning på 12V, og siden Arduinoen ikke er i stand til å forsyne mer enn 5V, er det behov for en ekstern spenningskilde.

Konfigurasjonen i simulink innebærer at en tilleggspakke må installeres for at simulink skal kunne compilere og erkjenne Arduinoen. Tilleggspakken "simulink support package for Arduino hardware" må derfor installeres. Tilleggspakken kommer med et Arduino bibliotek med komponenter som er konfigurert med pinnene til Arduinoen.



Figur 209: Koblingsskjema for simulink modellen

Figur 209 viser hvordan koblingsoppsettet for simulink modellen vil se ut. Her vil hver enkelt port være tilkoblet en bryter som enten henviser til lave eller høye signaler. Hver enkelt bryter er koblet til hver sin bryter (8 -> IN_1, 9 -> IN_2, 10 -> ENA)



Figur 210: Signalene til styringspinnene på Arduinoen.

Dersom ENA-port er høy, vil aktuatoren yte maksimal kraft og fart. ENABLE-porten er derfor alltid satt til "Høy". Figur 210 viser hvordan signalene i forhold til retning på aktuatoren fungerer:

- B1:Høy, B2:Høy - Motoren står i hvileposisjon.
- B1:Høy, B2:Lav - Motoren går fremover.
- B1:Lav, B2:Høy - Motoren reverserer.
- B1:Lav, B2:Lav - Motoren står i hvileposisjon.

Dette er først og fremst en test for å se hvordan lineæraktuatoren fungerer uten last og hvor fort den opererer. Neste steg er å finne ut hvordan man kan regulere hastigheten på aktuatoren.

13.2.2 Hastighetsregulering med PWM

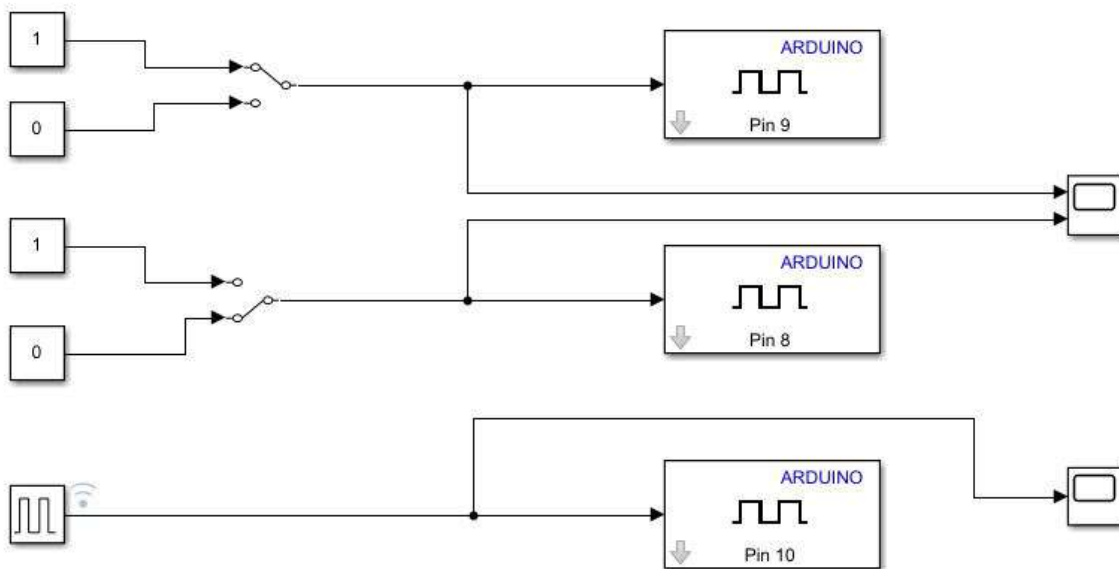
Utstyrliste:

- Power supply: EA-PS 5080-10 A
- DC-motor driver: Linak TR-EM-288-H
- Arduino mega 2560
- Linak lineæraktuator: 121000-21301220

Tabell 39: Test hastighetsregulering med PWM

Test ID	Ansvarlig	Godkjent av
TP.8-BR.01	OL	OL
Dato:	28.03.2019	
Krav:	BR.01 - GI.01 - ST.01 - ST.02	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	LA12 - testaktuator	
Hvilken funksjon skal testes:	Hastighetsregulering med PWM	
Status:	Godkjent	

Arduino mega 2560 har PWM utgang, så det vil derfor være nødvendig å observere om utgangen av signalet blir smidig nok til å kunne regulere farten på aktuatoren.



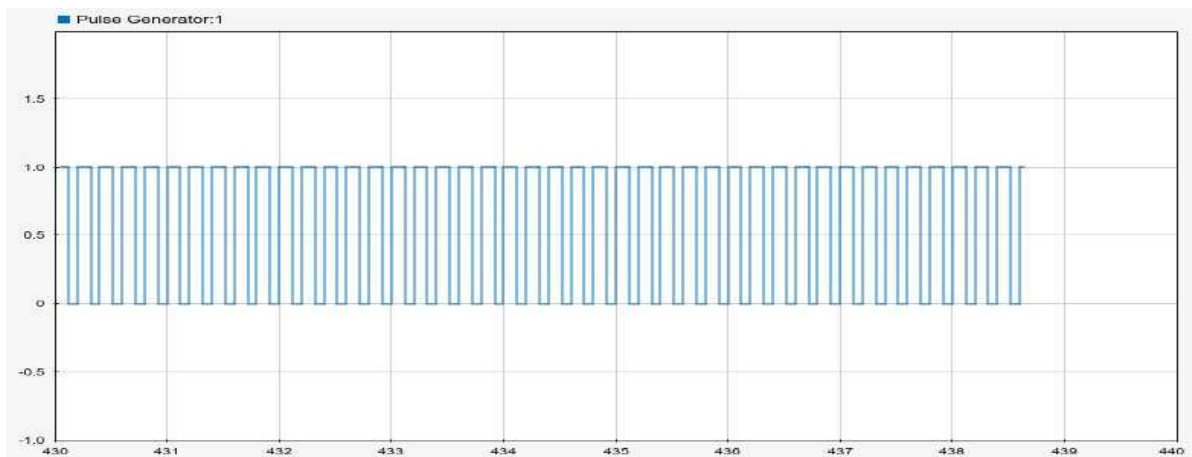
Figur 211: Test på aktuator med PWM-signal for å se om den går lineært med tiden.

Det har derfor blitt satt en periode på 0.2 sekunder og en pulsbredde på 90% av perioden for å se om aktuatoren opererer med tilnærmet konstant fart.

$$Duty\ cycle = \frac{T_{on}}{T_{on} + T_{off}}$$

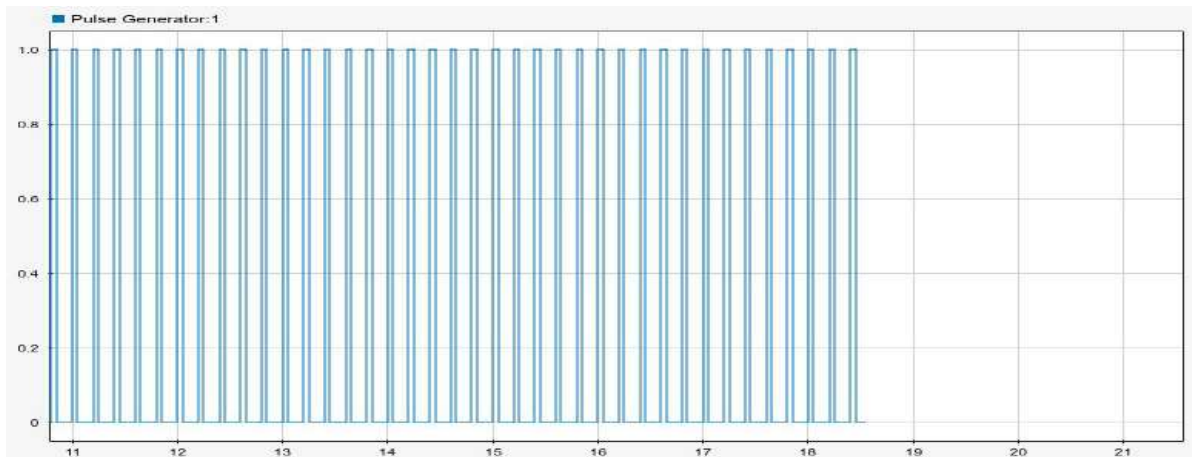
$$\frac{90\%}{100\%} \times 0.2\ sek = 0.18\ sek\ duty\ cycle$$

Det vil si med 90% duty cycle av 0.2 sekunder, slik at T_{off} vil være relativt liten (20 ms). Dette resulterer i at signalet vil være nesten 100% høyt, men gjør at aktuatoren ikke går på full hastighet (Figur 212).



Figur 212: Puls med periode på 0.2 sekunder og 90% pulsbredde

Det samme er blitt gjort med 25% duty cycle av 0.2 sekunder for å se om man kan regulere farten til å bli mindre. Farten holdt seg konstant og lineær. På-tiden ble 50 ms og Av-tiden 150 ms (Figur 213). Dette går såpass fort slik at det blotte øye ikke legger merke til endringer mellom av -og på tilstand.



Figur 213: Puls med periode på 0.2 sekunder og 25% pulsbredde

13.2.3 Håndbrems PID med potmeter

Tabell 40: Test PID kontroller med LA12 aktuator, brems

Test ID TP.7-BR.01	Ansvarlig SAM	Godkjent av SAM
Dato:	21.04.2019	
Krav	BR.01 - BR.02 - GI.02	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	LA12 - testaktuator	
Hvilken funksjon skal testes	PID kontroller	
Status:	Ikke godkjent	

Det ble koblet opp på samme måte som test TP.1-GI.02, med potmeter. Formålet med testen var å teste variabelt pådrag. Dette viste seg å bli vanskelig da potmeteret setter begrensinger på hva som kan testes. Dette gjorde at error signalet som trengs i tilbakekoblingen ikke ble tilstrekkelig for å gjøre testingen, som igjen resulterte i en feilet test. Det vil bli gjort ytterligere testing på funksjonen når riktige aktuatorer kommer.

13.2.4 Bremsing med PID-kontroll

Tabell 41: Test PID kontroller med LA33 aktuator, brems

Test ID	Ansvarlig	Godkjent av
TP.25-BR.02	KVL	KVL
Dato:	10.04.2019	
Krav	BR.02	
Verifikasjonsmetode:	Måling	
Testobjekt:	LA33	
Hvilken funksjon skal testes	PID kontroller	
Status:	Godkjent	

Denne testen ble gjennomført ved å bruke en LA33 aktuator med analog tilbakekobling, for å overvåke posisjonen [62]. Verdiene fra tilbakekoblingen varierer fra 0.5 - 4.5 Volt, slik at det var nødvendig å omdefinere disse verdien i Processing koden. For at de skulle passe med verdiene fra PS4-kontrolleren.

Ved å kjøre verdiene fra PS4-kontrollen igjennom PID-en som de ønskede verdiene og ved å bruke posisjonstilbakekoblingen fra LA33 som referanse ble det mulig å kjøre aktuatoren til en posisjon bestemt av hvor langt inne man holdt inne L2-knappen.

Det var i tillegg nødvendig å sette en grense på når aktuatoren skulle justere seg selv. Da den ved lave feilverdier hadde små utslag begge veier, noe som ikke er ønskelig. Det er også vanskelig for en operatør å holde inne L2-knappen i en fast posisjon.

Dette kan kompenseres for mekanisk ved å ha hvilepunktet til aktuatoren nær bremsepunktet slik at man raskt kan bremse og ved å regulere ved å se ATV-ens reaksjon.

Testingen involverte mange ledninger og tilkoblingspunkter og alt ble sjekket flere ganger før oppstart, men det ble brukt en for tynn ledning fra spenningskilden til aktuatordriveren noe som resulterte i at det ble utviklet for mye varme i ledningen. Dette ble raskt oppfattet og spenningskilden ble slått av.

Det har vært mye fokus på sikkerhet rundt testing og hva man skal gjøre ved slike situasjoner. På grunn av gode rutiner ble situasjonen avverget før det fikk alvorlige konsekvenser, selv om hendelsen ikke skulle ha skjedd i utgangspunktet. Gruppen kommer til å ta med seg erfaringen fra denne testen i fremtidige tester.

Testen er godkjent for krav BR.02 fordi den analoge tilbakekoblingen gjør det mulig å vite posisjonen til aktuatoren. Koden skrevet i Processing til nå muliggjør ikke bremsing

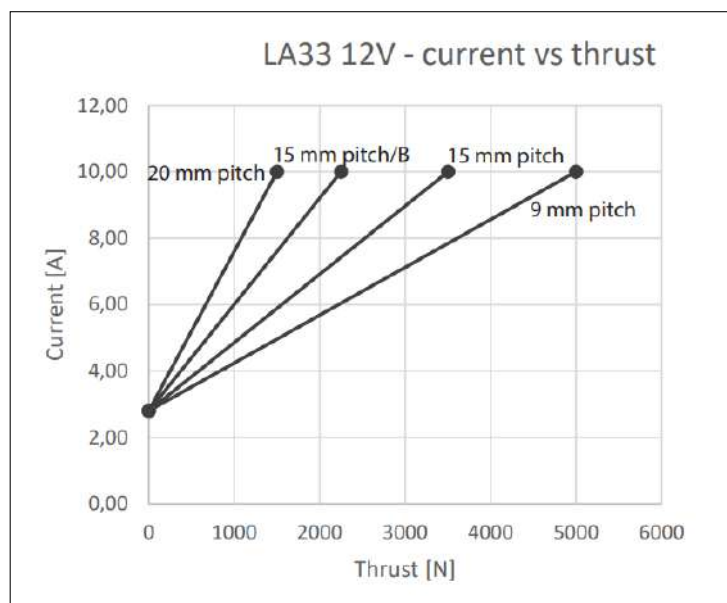
med variabel kraft.

13.2.5 Strømsensor stopfunksjon

Tabell 42: Strømsensor begrensning

Test ID TP.32-BR.01	Ansvarlig SAM	Godkjent av SAM
Dato:	20.05.2019	
Krav	O.19, BR.02.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Strømsensor - LA33	
Hvilken funksjon skal testes	Stoppe bremsekraft ved høyt strømforbruk	
Status:	Ikke godkjent	

For å sørge for at aktuatorene ikke bruker for mye kraft på systemet vil det være nødvendig med overvåkning av kraftbruk.



Figur 214: LA33 Kraft mot Ampere

Beregningene på bremsene viste at man trenger 613N til håndbrems og 787N fra aktuatoren med bremsehendelene som er designet. Fra figur 214 ser man at det trengs i underkant av 6 A for å oppnå dette.

For å oppnå ønsket kraft har det blitt satt inn en strømsensor. Den har blitt satt som en begrensning i koden, slik at aktuatoren stopper om den går over en verdi. Ved å bruke denne som en stop-bryter vil ikke aktuatoren skade på komponentene.

Strømsensoren som er benyttet er en Pololu ACS71 [47]. Denne kan måle fra -30 til 30 Ampere, og gir derfor 2,5V inn til arduinoen ved 0 Ampere. Etersom arduinoen kan måle 5V med 1023 forskjellige verdier vil strømsensoren ha en oppløsning på 17 steg per Ampere. En strømbegrensning på 5.5 A vil da gi 93 steg som vist i ligning 29.

$$\frac{1023}{60} = 17,05$$

$$17,05 * 5,5A = 93Steg \quad (29)$$

Etersom håndbremsaktuatoren starter helt ute, vil det si at aktuatoren går bakover. Det trengs ikke maksimal bremsekraft under lave hastigheter, så strømsensorgrensen settes til 425, som tilsvarer 5,12 Ampere. Dersom strømsensoren går under denne grensen vil håndbremsen stoppe å gå videre.

```

if (going==true) {

    if (analogVerdi<410) {
        handbremsKontroll=0;
        going=false;
    }

    aktuatorKjor(handbremsFrem, handbremsBak, handbremsPin,
        ↪ handbremsKontroll);
}
else if(handbremsKontroll>0){
    going=true;
}

```

Dette skulle virke i teorien, men det viser seg fra testing at aktuatoren bruker mer strøm da den starter opp, selv uten last. Aktuatoren gjorde ikke dette ved hver oppstart, men innimellom dro den over 5.8 Ampere. Dette kan være mulig å kodes inn ved at aktuatoren får lov til å bruke så mye strøm i veldig kort tid, da den stabiliserte seg etter første pådrag.

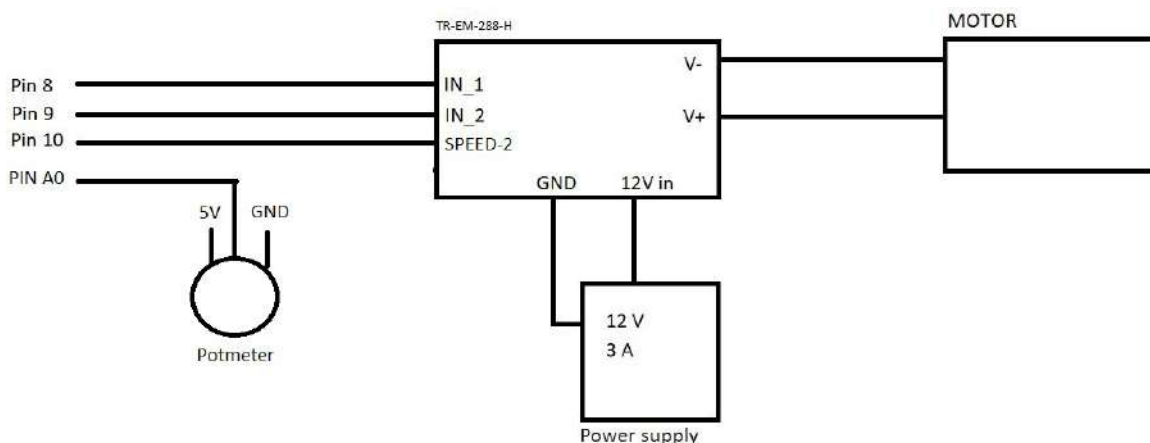
13.3 Gir

13.3.1 PID kode med potmeter

Tabell 43: Test PID kontroller med LA12 aktuator, gir

Test ID	Ansvarlig	Godkjent av
TP.1-GI.02	SAM	SAM
Dato:	11.04.2019	
Krav	GI.02 - BR.02	
Verifikasjonsmetode:	Måling	
Testobjekt:	LA12 - Testaktuator	
Hvilken funksjon skal testes	PID kontroller	
Status:	Godkjent	

For å teste PID kontrolleren før aktuatorene ble levert ble LA12 som ble lånt, brukt til testing. Etersom denne ikke ha posisjonssensor, ble ett potmeter brukt for å simulere en slik tilbakekobling 215.



Figur 215: Oppkobling til LA12 med TR.EM-288-H og potmeter

Det ble kodet opp en midlertidig løsning på hvordan gir kan løses, ved å trykke på R1 på PS4 kontrolleren for å gire opp og L1 for å gire ned. Det ble så sendt forskjellige setpoint til PID kontrolleren til gir.

```
//-- FOR Å GIRE MED KONTROLLER --//
void giring() {
```

```
double girKontroll = 0;

if (R1==true) {
    gir++;
    while (R1==true) { //OBS - Låser alt annet mens man holder inne R1,
        ↪ men nødvendig for å kun telle opp en gang
        getUserInput();
    }
}
if (L1==true) {
    gir--;
    while (L1==true) {
        getUserInput();
    }
}

//-- Sender forskjellige posisjoner til PID-kontroller -- //
if (gir == 0) {
    girKontroll = PID_kontroller(200, girPID);
} else if (gir == 1) {
    girKontroll = PID_kontroller(500, girPID);
} else if (gir == 2) {
    girKontroll = PID_kontroller(700, girPID);
} else if (gir == 3) {
    girKontroll = PID_kontroller(900, girPID);
}

girState(girKontroll);
};

//---- SKAL MOTOREN GÅ FREMOVER ELLER BAKOVER ----//
void girState(double kontroll_signal) {
    int kontrollInt = (int) kontroll_signal;

    if (kontroll_signal>0)                //H-Bro
```

```

{
  arduino.digitalWrite(MotorFRAM, 255);
  arduino.digitalWrite(MotorBAK, 0);
  arduino.analogWrite(5, kontrollInt);
  text("Fremover", 190, 190);
} else {
  arduino.digitalWrite(MotorBAK, 255);
  arduino.digitalWrite(MotorFRAM, 0);
  arduino.analogWrite(5, abs(kontrollInt));
  text("Bakover", 190, 190);
}
text(int(kontrollInt)+" Feilverdi", 90, 190);
}

```

Dette resulterte i en godkjent test, hvor PID kontrolleren ble testet i praksis. potmeteret fungerte til å gi forskjellige verdier, og aktuatoren gikk raskt frem eller tilbake avhengig av hvilken feilverdi som ble gitt.

13.3.2 PID kontrollen med LA14 aktuator

Tabell 44: PID kontrollen med LA14 aktuator

Test ID	Ansvarlig	Godkjent av
TP.23-GI.01	SAM	SAM
Dato:	8.05.2019	
Krav	GI.01.01 ,GI.02	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	LA14	
Hvilken funksjon skal testes	Skifte av gir med PID kontrollen	
Status:	Godkjent	

Fra Udemy [69] ble det foreslått å bruke Ziegler Nichols metode for å finne de ulike verdiene til kontrolleren. Dette går ut på å finne det kritiske svingispunktet for systemet. Dette vil si at systemet aldri stopper og det har en jevn oscillering. For å finne dette punktet blir Ki og Kd verdiene satt til 0, mens Kp verdiene endres.

EEboardet ble koblet til for å observere utgangssignalet, og K_p verdiene ble endret. Ved å gjøre dette ble det observert at systemet var ustabil ved alle K_p verdier. Det ble likevell eksperimentert ved å endre K_i verdiene som anvist fra Udemy, uten at dette ga noe særlig utslag. En teori på dette er at tilbakekoblingens sensoren ikke er rask nok til å bruke denne metoden.

Da ble det lagt inn en liten feilmargin, altså om sensoren er mellom en viss verdi vil motoren stoppe. Etter at dette ble gjort ble testen godkjent med kun K_p verdi på 1, mens K_i og K_d lå på 0. Aktuatoren er nå klar til å finne riktige punkter til riktige gir, før den skal implementeres.

13.4 Styring

13.4.1 Stepper-motor(28BYJ-48)

Tabell 45: Test stepper-motor: styring

Test ID TP.34-ST.04	Ansvarlig OL	Godkjent av OL
Dato:	23.04.2019	
Krav:	ST	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Stepper-motor	
Hvilken funksjon skal testes:	Styring med steppermotor	
Status:	godkjent	

Utfordring: Bli kjent med funksjonene i en stepper-motor; Styre retning med et potmeter.

Det er planlagt å bruke en stepper-motor fra Festo til å styre ATV med. Denne skal påføre et pådrag på styreakselen. Det er generelt lite erfaring innad i gruppen når det kommer til stepper motor, det er derfor hensiktsmessig å finne ut hvordan motoren fungerer generelt

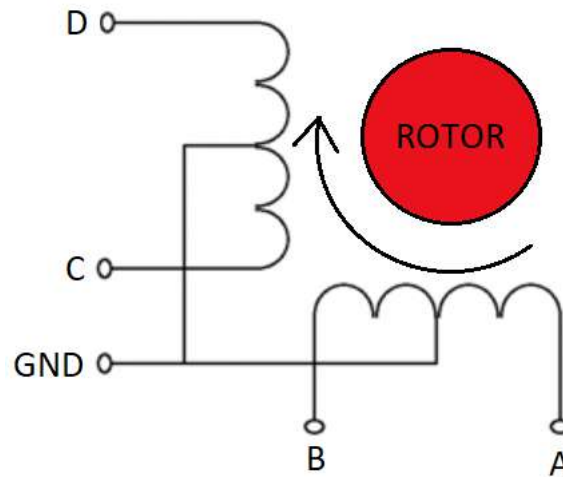
Dataarket til steppermotoren(28BYJ-48) spesifiserer en spenning på 5V for å drive motoren. Dette er tilstrekkelig å hente fra arduinoen. Motoren har 4 faser og 64 steps pr. omdreining. Stepvinkel:

$$\text{Stepvinkel} = \frac{360^\circ}{64} = 5,625^\circ$$

Dette er for en full-step 8 sekvensiell stepper, i oppkodingen brukes 4 sekvensiell kode:

$$\text{Steps} = \frac{360^\circ}{5,625^\circ} = 64, \text{ Steps} = \frac{64}{2} = 32$$

Stepper-motoren som blir brukt har 4 coiler som kan magnetiseres; De kalles dem A,B,C og D for ordens skyld. For at rotoren skal bevege seg kontinuerlig i en retning, vil man at coilene skal magnetiseres i kronologisk rekkefølge; først A → B → C → D.



Figur 216: Coil-logikk for stepper-motor(28BYJ-48)

Hver sekvens må magnetisere gitte coiler for at motoren skal rotere et step. Det settes følgende logikk for hver sekvens(høy og lav) i en 4 sekvensiell logikk:

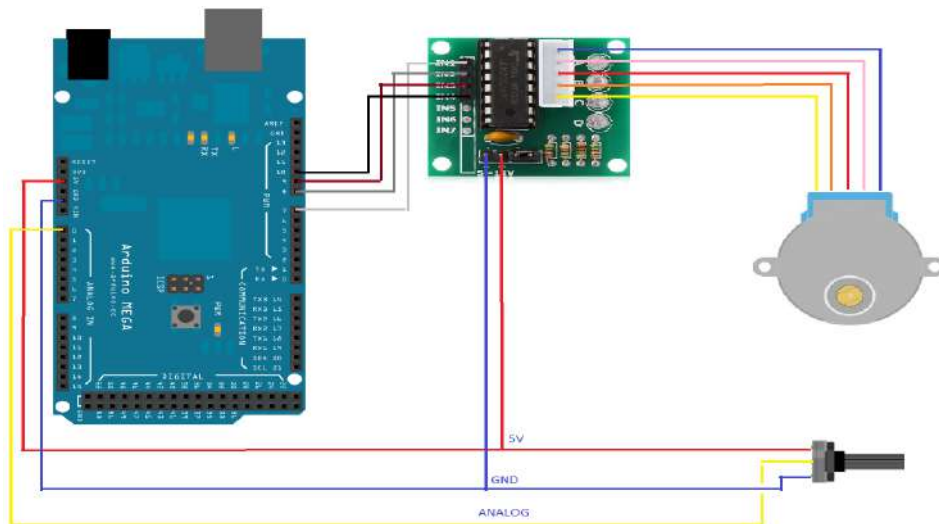
Tabell 46: Tabell for sekvenslogikk for en full step motor

	A	B	C	D
sekvens 1	1	0	0	1
sekvens 2	1	1	0	0
sekvens 3	0	1	1	0
sekvens 4	0	0	1	1

Det er interessant å koble opp en steppermotor sammen med et potmeter for å observere hva som skjer og manuelt endre kjøreretning på motoren og se om den roterer jevnt og kontinuerlig.

Utstysliste:

- Arduino mega 2560
- breadboard
- ULN2003 driver
- 28BYJ-48 stepper-motor
- Potentiometer
- Jumperkabler



Figur 217: Koblingskjema for arduinoen - kretsen er satt opp i Fritzing [23]

Koblingskjema:

Pinnetilkoblinger:

Pinne 7 → IN_1

Pinne 8 → IN_2

Pinne 9 → IN_3

Pinne 10 → IN_4

Pinne A0 → *Potmeter*

Tilkobling av jord og 5V spenning til driver og potmeter.

Skriver deretter kode i arduino IDE og bruker tilhørende bibliotek for å definere faser og steps i motor.

Bruker "stepper.h" som følger med IDE software.

```
#include <Stepper.h> //bibliotek

#define STEPS 32 //steps i vår motor (28BYJ-48)

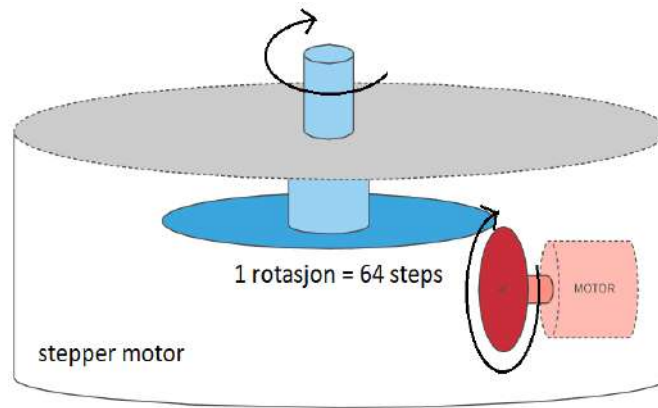
Stepper stepper(STEPS, 7, 9, 8, 10); //hvor mange steps og sekvenser
int potMeter = 0;

void setup() {
  Serial.begin(9600);
  stepper.setSpeed(800); //Hastighet
}

void loop(){
  potMeter = map(analogRead(A0),0,1024,0,500); //kartlegge potmeteret
  ↪ med verdiene
  if (220<potMeter<280) //Stå stille mellom 220 og 280 verdier
    stepper.step(0);
  if (potMeter>280) //Gå med klokken når potmeter leser høyere enn 280
    stepper.step(10);
  if (potMeter<220) //Gå mot klokken når potmeter leser lavere enn 220
    stepper.step(-10);

  Serial.println(potMeter); //lese og monitorere verdiene.
```

Tester også hvordan prosessen av 'en step' foregår. Der overføringen mellom girene resulterer i tregere omdreining av steppermotoren, men med mer kraft og presisjon. (figur 218)



Figur 218: Stepper-motor: en hel rotasjon for motor tilsvarer et step for stepperen

Man benytter seg derfor av logikken som er utledet i tabell 46, og setter opp følgende kode i Arduino:

```
#define waiting_time 5

void setup() {
  for (int z = 8; z < 12; z++) //definerer pinnene
    pinMode(z, OUTPUT);
}

void loop() {
  digitalWrite(8, HIGH); //8,9->9,10->10,11->11,8-
  delay(waiting_time);
  digitalWrite(9, HIGH);
  delay(waiting_time);
  digitalWrite(8, LOW);
  digitalWrite(10, HIGH);
  delay(waiting_time);
  digitalWrite(9, LOW);
  digitalWrite(11, HIGH);
  delay(waiting_time);
  digitalWrite(10, LOW);
  delay(waiting_time);
}
```

```
digitalWrite(11, LOW);
delay(3); //3 ms mellom hvert step
}
```

Koden verifiserer at stepperen fungerer som ønsket og at dette blir base for styreutvikling av stepper-motoren videre for systemet, der hver 'step' vil få en spesifikk adresse å henvises til.

13.4.2 Mapping av steppermotor

Tabell 47: Steppermotor Festo mapping

Test ID	Ansvarlig	Godkjent av
TP.33-ST.04	OL	OL
Dato:	10.05.2019	
Krav	ST.03	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Stepper,mapping, rele	
Hvilken funksjon skal testes	Varierende pådrag	
Status:	Ikke godkjent	

I første iterasjon med styring av steppermotoren, ble det sagt at det skulle være mulig å styre steppermotoren. Til tross for industrielle parametere(24 V logikk), ble det klargjort at det skulle fint la seg gjøre å styre motoren via en arduino dersom man brukte 5-volts releer for å åpne opp for en ekstern spenningskilde.

Det ble satt i gang tester for å få verifisert dette.

```
int fram=8;
int ena=10;
int homing=11;
void setup(){
  pinMode(fram, OUTPUT);
  Serial.begin(9600);
}
void loop(){
  digitalWrite(ena, HIGH);
```

```

int potVal = analogRead(A0);
int potMeter = map(potVal, 0, 1023, 0, 255);
analogWrite(8, potMeter);
}

```

Det ble straks etablert noen problemer, da releene selvfølgelig ikke reagerte på analog spenning inn på bryterene, og aktiverte seg kun ved 5V og ikke på den variable spennin-gen. Funksjonen på motoren tillater kun at stepperen ”jogger” til gitte yttergrenser ved høy(24v) inngang. Det ble dermed umulig å styre motoren ved hjelp av mappingfunk-sjonen i arduino IDE, da den er avhengig av analoge signaler. Det blir derfor nødvendig å finne alternative løsninger til problemet. Testen er derfor ikke godkjent.

13.4.3 Test av stepper-motor ved strømbrudd

Dersom strømbrudd oppstår under kjøring av stepper-motor, må det foretas en homing før stepper-motoren vil være klar til å styres igjen. Grunnen til dette er at den har en encoder som sender pulser når motoren beveger seg og den må dermed kjøres tilbake til et nullpunkt slik at den ikke fortsetter å kjøre videre fra samme posisjon som den var i før strømbruddet.

Tabell 48: Test styring - EMMS-ST

Test ID	Ansvarlig	Godkjent av
TP.20-NS.02.02	LTT	LTT
Dato:	08.05.2019	
Krav	NS.02.02 - ST.03.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	EMMS-ST Steppermotor	
Hvilken funksjon skal testes	Styring stepper-motor etter strømbrudd	
Status:	Godkjent	

Tabell 49: Test homing - EMMS-ST

Test ID TP.21-NS.02.02	Ansvarlig LTT	Godkjent av LTT
Dato:	08.05.2019	
Krav	NS.02.02 - ST.03.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	EMMS-ST Steppermotor	
Hvilken funksjon skal testes	Homing stepper-motor etter strømbrudd	
Status:	Godkjent	

Etter å ha testet stepper-motoren etter strømbrudd viste det seg at tilkoblingen må skje på følgende måte:

- Alle pinner fjernes fra koblingspunktene sine.
- Tilfører deretter kretsen spenning.
- X1. Pin 10 - Control-enable kobles til.
- X1. Pin 8 - Homing kobles til

Når homing er fullført kan denne pinnen fjernes fra tilkoblingspunktet og man kan styre stepper-motoren ved hjelp av Pin 1 og Pin 2 igjen (venstre og høyre).

13.4.4 Gradvis styring

Tabell 50: Test gradvis styring - EMMS-ST

Test ID TP.22-ST.04	Ansvarlig LTT	Godkjent av LTT
Dato:	10.05.2019	
Krav	ST.04 - ST.03	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	EMMS-ST Steppermotor	
Hvilken funksjon skal testes	Gradvis styring av stepper-motor	
Status:	Godkjent	

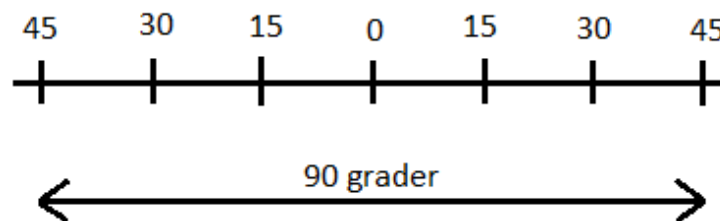
Ved første konfigurering av stepper-motor i FCT ble styringen definert fra 0-45 grader i retning høyre og venstre. Styringen av ATV-en ga da maks rattutslag til høyre eller venstre noe som ikke vil være ideelt under kjøring. Ønsket er derfor å kunne svinge gradvis ved å bruke flere step. Ved å ta i bruk flere styrepinner kan man velge å flytte

flere step/grader på kortere intervaller. Figur 219 viser et oversiktsbilde over fordeling av step fra 0 - 90 grader i FCT hvor 45 grader er satt som nullpunktet (homing).

Controller		Motor	Axis	Gear Ratio (total):		
CMMO-ST-C5-1-DIOP		EMMS-ST-87-L-SE-G2	User Defined Rotative Axis (360 °)	9 : 1		
No.	Type	Target	Velocity	Acceleration / Deceleration	Extra Load	Torque Feed Forward
1	PA	90,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
2	PA	75,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
3	PA	60,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
4	PA	45,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
5	PA	30,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
6	PA	15,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %
7	PA	0,00 °	110,00 %/s	65699,2 %/s ²	0,000 kgcm ²	100 %

Figur 219: Tabell med stepfordeling

I figur 220 ser man hvordan fordelingen av step og grader vil bli fordelt.



Figur 220: Step i grader

13.4.5 Styring av stepper med arduino

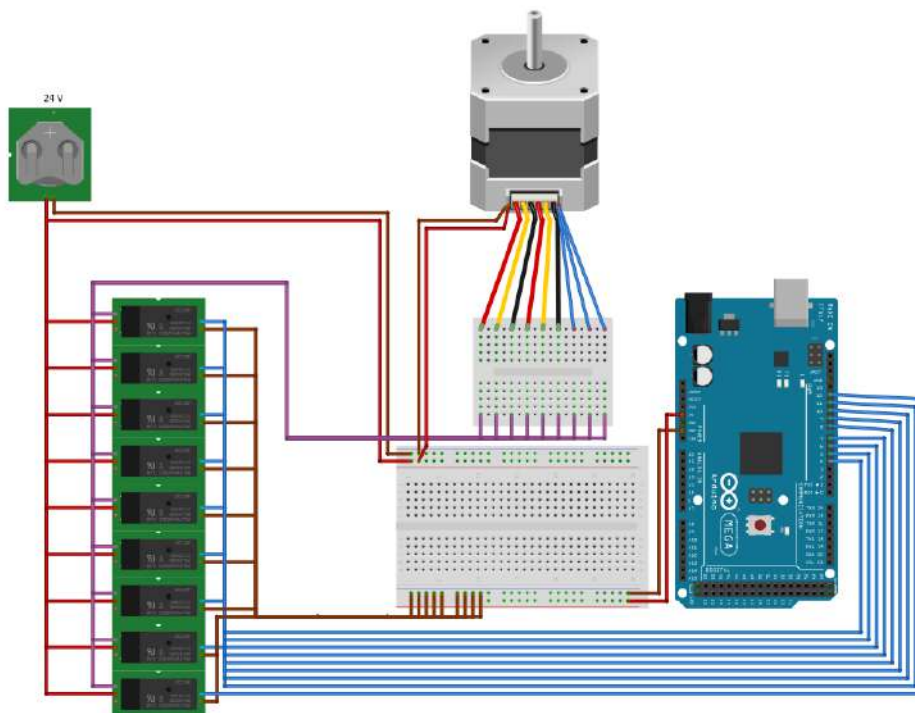
Tabell 51: Styring av stepper-motor med Arduino

Test ID TP.24-ST.04	Ansvarlig LTT	Godkjent av LTT
Dato:	13.05.2019	
Krav	ST.04 - ST.03 - FS.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	EMMS-ST stepper-motor	
Hvilken funksjon skal testes	Styring av stepper med arduino	
Status:	Ikke godkjent	

Utstysrliste:

- Stepper-motor: EMMS-ST-87-L-SE-G2
- Motor-kontroller: CMMO-ST-C5-1-DIOP
- Arduino Mega 2560
- 9 × - relè 5VDC
- Spenningskilde 24 V
- Potmeter

For å kunne fjernstyre stepper-motoren ble den koblet opp som vist i figur 221 (Kretsen er satt opp i Circuit Lab [9]). Som det kom frem fra test TP.01 - ST.01, er det ikke mulig å styre stepper-motoren ved hjelp av mapping og det ble forsøkt en alternativ løsning. Løsningen i dette tilfelle var å sette signalet høyt eller lavt slik at det valgte signalet kun ble aktivert dersom det sto som høyt. Som et alternativ til fjernstyring ble testen utført med et potmeter slik at stepper-motoren ville svinge x-antall grader innenfor gitte verdier på potmeteret. Det ble brukt samme konfigurering av stepper-motor fra FCT som i test TP.22-ST.03 - “gradvis styring”.



Figur 221: Kretstegning - styring av stepper med arduino

Utdrag av koden som ble brukt i Arduino IDE (Fullversjon kode [3]):

```
int enable = 4;
int homing = 5;
int zero = 6;
int femten = 7;
int tretti = 8;
int fyrtifem = 9;
int seksti = 10;
int syttifem = 11;
int nitti = 12;

void setup() {
  Serial.begin(9600);
  pinMode(enable, OUTPUT);
  pinMode(homing, OUTPUT);
  pinMode(zero, OUTPUT);
  pinMode(femten, OUTPUT);
  pinMode(tretti, OUTPUT);
  pinMode(fyrtifem, OUTPUT);
  pinMode(seksti, OUTPUT);
  pinMode(syttifem, OUTPUT);
  pinMode(nitti, OUTPUT);
}

void loop() {
  int potVal = analogRead(A0); // Leser verdiene til potmeteret mellom
  ↪ 0 og 1023
  if (potVal >= 10 && potVal <= 169) { // Verdier på potmeteret
    digitalWrite(zero, HIGH); // 0 grader sving
    digitalWrite(femten, LOW);
    digitalWrite(tretti, LOW);
    digitalWrite(fyrtifem, LOW);
    digitalWrite(seksti, LOW);
    digitalWrite(syttifem, LOW);
    digitalWrite(nitti, LOW);
  }
}
```



```
}  
if (potVal>=170 && potVal<=340){ // Verdier på potmeteret  
    digitalWrite(femten, HIGH); // 15 grader sving  
    digitalWrite(zero, LOW);  
    digitalWrite(tretti, LOW);  
    digitalWrite(fyrtifem, LOW);  
    digitalWrite(seksti, LOW);  
    digitalWrite(syttifem, LOW);  
    digitalWrite(nitti, LOW);  
}  
.  
.  
.  
if (potVal>=1001 && potVal<=1023){ // Verdier på potmeteret  
    digitalWrite(nitti, HIGH); // 90 grader sving  
    digitalWrite(zero, LOW);  
    digitalWrite(femten, LOW);  
    digitalWrite(tretti, LOW);  
    digitalWrite(fyrtifem, LOW);  
    digitalWrite(seksti, LOW);  
    digitalWrite(syttifem, LOW);  
}  
    Serial.println(potVal); // Printer potmeterets verdi til seriell  
    ↪ overvåker  
}
```

Denne testen fungerte til en viss grad og viste seg å være en bedre løsning enn testen med mapping. Signalene gikk gjennom slik at motoren beveget seg, men i endepunktene (ved 0- og 90 grader) reagerte ikke stepper-motoren som ønsket og tok et step i feil retning før den begynte på riktig step igjen. Dette gjorde at styring av stepper-motoren ble ustabil og av den grunn ble testen ikke godkjent.

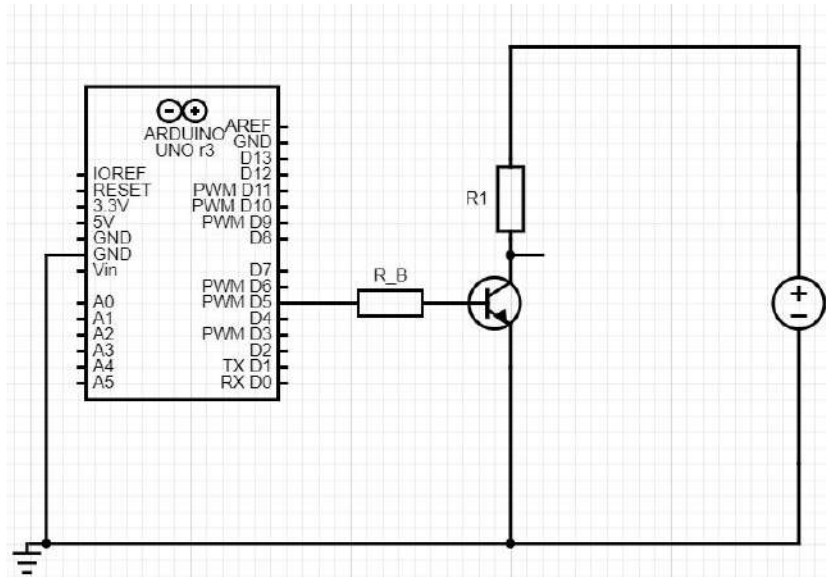
13.4.6 Test arduino med transistor

Tabell 52: Test av NPN transistor - 2N3904 med arduino

Test ID TP.29 - ST.04	Ansvarlig OL	Godkjent av OL
Dato:	17.05.2019	
Krav	ST.04 - ST.03 - ST.02 - ST.01 - FS.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Transistor - 2N3904	
Hvilken funksjon skal testes	Styring - transistor som switch for 24V	
Status:	Ikke godkjent	

2N3904 er en NPN transistor som er beregnet på småskala signalforsterkning, men som kan fungere fint som en spenningsbryter for svakstrømkretser. Dette har blitt testet i praksis for å se om de fungerer stabilt nok til å forsyne styrepinnene med 24V via en arduino.

Inngang på base kan kontrolleres ved hjelp av en arduino og en 5V utgang og en resistor på base for å regulere bryter på transistor med minst mulig strøm. Når base har strøm fra base til emitter, vil bryteren være lukket, og ekstern krets fra collector til emitter vil lede strøm. Figur 222 viser hvordan kretsen er satt opp (Kretsen er satt opp i Circuit Diagram [57]).



Figur 222: Transistorkretsen fungerer som bryter og aktiveres ved høy utgang fra arduinoen

Testen fungerte fint for en spenning på 5V på den eksterne kretsen og for å tenne en LED. Når man endret spenning fra 5V til 24V viste det seg at motstanden (R1) ikke var resistent nok mot den strømmen som gikk fra collector til emitter, og som videre skulle bli et stort problem, da styrepinnene er avhengige av 24V til stepper motor.

Med for høy motstand, ble spenningsfallet for stort og V_{out} ble ikke tilstrekkelig nok for å kunne få ut nok spenning. Dersom motstanden var for liten for å hindre for høyt spenningsfall over R1, tålte ikke motstanden den resulterende strømmen.

Testen er derfor ansett som 'ikke godkjent' da den ikke fungerte til det formålet den var satt til.

13.4.7 Test transistor som switch

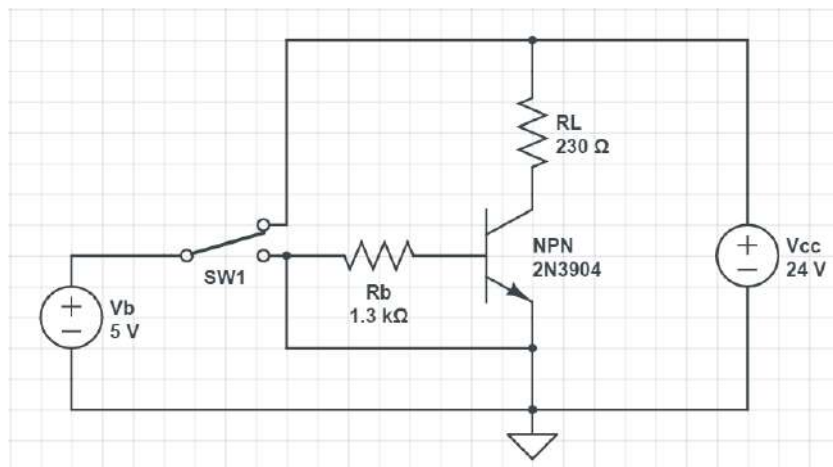
Tabell 53: Test av NPN transistor - 2N3904

Test ID	Ansvarlig	Godkjent av
TP.30 - ST.04	LTT/OL	LTT
Dato:	18.05.2019	
Krav	ST.04 - ST.03 - ST.02 - ST.01 - FS.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Transistor - 2N3904	
Hvilken funksjon skal testes	Styring - transistor som switch	
Status:	Ikke godkjent	

For å teste hvordan 2N3904 fungerer som en switch ble kretsen i figur 223 koblet opp (Kretsen er satt opp i Circuit Lab [9]). Her ble det sendt 5V fra V_B og satt inn en switch før motstand R_B som sendte signalet til enten collector eller emitter. For å bestemme motstandsverdien til R_B og R_L ble strømforsterknings-faktoren h_{FE} lest ut fra datablad [12] slik at I_B kunne beregnes og videre motstandsverdiene [43]:

$$R_B = \frac{V_B - V_{BE}}{I_B} = \frac{5V - 0.7V}{3.33 \text{ mA}} = 1.3 \text{ k}\Omega \quad (30)$$

$$R_L = \frac{V_{CC} - V_{CE}}{I_C} = \frac{24V - 1V}{0.1A} = 230 \Omega \quad (31)$$



Figur 223: Krets med 2N3904 - transistor

Det ble i denne testen sendt 5V inn på basen fra et eksternt batteri, mens det ble sendt 24V inn på collector og emitter ble koblet til jord. På denne måten skal man kunne velge om man sender spenningen fra base til collector eller emitter og transistoren vil opptre som en bryter. I tilfelle hvor man sender til collector vil V_{out} være lik V_{CE} som er null, og transistoren er i saturation. I tilfelle hvor man sender til emitter vil $V_{out} = V_{CE} = V_{CC} = 24V$ (cut-off) [67].

Motstanden før collector, R_L , var satt til 230Ω - det viste seg at denne motstanden ikke var tilstrekkelig i forhold til strømmen som ble sendt inn. Det ble så forsøkt med en høyere motstand noe som resulterte i at spenningsfallet ble for stort i forhold til at styrepinnene til stepper-motoren skal ha 24V for å styre.

Testen er på grunn av dette ikke godkjent.

13.4.8 Test av ulike transistorer

Tabell 54: Test av ulike transistor

Test ID	Ansvarlig	Godkjent av
TP.31 - ST.04	LTT/OL	LTT
Dato:	17.05.2019	
Krav	ST.04 - ST.03 - ST.02 - ST.01 - FS.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	2N3904 - BUF742 - MJE350 - BUF620	
Hvilken funksjon skal testes	Styring - test av transistor	
Status:	Ikke godkjent	

Det har blitt testet en del ulike transistorer for å finne ut hvilken transistor som egner seg best å bruke som switch til styringssystemet. Følgende transistorer ble testet:

- 2N3904 [12]
- BUF742 [14]
- MJE350 [15]
- BUF620 [13]

Ut ifra verdiene i databladet skulle disse transistorene vært tilstrekkelig, men det viste seg at flere av transistorene likevel ikke tålte spenningene og strømmene som ble tilført.

Årsaken til at de ryker mistenkes å være at det blir sendt for høy spenning inn på collector og brente spoler, samt at EMF-dioden (Electromotive force) er defekt.

Testen anses ikke som godkjent da ingen av transistorene fungerte som ønsket.

13.5 Hardware

13.5.1 Bluetooth

Tabell 55: Test av trådløs styring med bluetooth

Test ID	Ansvarlig	Godkjent av
TP.10-O.13	KVL	KVL
Dato:	22.04.2019	
Krav:	O.13	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Arduino og bluetooth (HC06)	
Hvilken funksjon skal testes:	Trådløs styring	
Status:	Godkjent	

Denne testen er den første iterasjonen av flere for å forsikre at grensesnittene mellom fører og mikrokontrolleren som sitter på ATV-en fungerer slik de skal og at ønsket responstid og informasjonsflyt opprettholdes.

Testen ble gjennomført ved å bruke programmene Processing og Arduino og biblioteker for disse slik at en Playstation 4 kontroller og Serial kommunikasjon for Processing kunne benyttes.

Testen gikk ut på å bruke en Playstation 4 kontroller til å styre en LED på Arduinoen via en bluetooth-link fra PC-en til Arduinoen.

Dette ble gjort i denne omgangen ved å konvertere knappetrykket fra kontrolleren til false og true som ble oversatt til 1 eller 0 i Processing som en String. For så å sende dette til Arduinoen via bluetooth som ved en tidligere overført kode tolket dette som av og på på LED-en.

Tabell 56: Test av trådløs styring med bluetooth, iterasjon 2

Test ID	Ansvarlig	Ikke godkjent av
TP.11-O.13	KVL	KVL
Dato:	23.04.2019	
Krav:	O.13	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Arduino og bluetooth (HC06)	
Hvilken funksjon skal testes:	Trådløs styring	
Status:	Ikke godkjent	

Andre iterasjon av Bluetooth-kommunikasjonen mellom PC-en og Arduinoen ble gjennomført med overføring av koden som er skrevet i Processing for håndbrems og giring.

Denne testen ble gjennomført ved å endre Baud-raten i Firmata-koden i Arduino til 9600 slik at den sender med lik symbol-rate som Bluetooth-linken originalt er stilt. For deretter å definere Arduinoen i processing til å sende via den riktige porten.

Resultatet av dette var at responstiden var alt for lang, noe som resulterer i at denne overføringsmetoden ikke kan brukes i nåværende form.

Ved å endre Bluetooth-enheten til å sende med høyere Baud-rate og ved å endre hvor mange iterasjoner Firmata-koden skal gjøre hvert sekund vil nok også responstiden på kommandoer gå ned. Noe som vil bli testet ved neste iterasjon.

13.5.2 RC - sender/mottaker

Tabell 57: Test av trådløs styring med RC-kommunikasjon, HC-12

Test ID TP.12-O.13	Ansvarlig KVL	Godkjent av KVL
Dato:	25.04 - 26.04	
Krav:	O.13	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Arduino, Processing og HC-12	
Hvilken funksjon skal testes:	Trådløs styring	
Status:	Godkjent	



Figur 224: HC-12 sender/mottaker

Det ble påbegynt planlegging av oppsettet på koden i Processing og Arduino samtidig som testingen av RC-kommunikasjonen der det ble brukt to HC-12 enheter [27].

HC-12 kan brukes som både sender og mottaker. Under testinga ble det brukt to PC-er, to Arduinoer, to HC-12 enheter og en Playstation 4 kontroller.

Det ble valgt og ikke bruke bluetooth-kommunikasjon slik de forrige testene viser til.

Testingen ble gjennomført ved å bruke en av PC-ene til å kjøre Processing-koden som via kabel overførte verdiene fra Playstation-kontrolleren til Arduino nummer 1.

De to HC-12 enhetene var koblet til hver sin Arduino og kommuniserte sammen ved å bruke en enkel eksempelkode lastet inn på forhånd på Arduinoene, før testen.

Koden var skrevet slik at man i Seriell overvåkeren i Arduino kunne kommunisere, begge veier.

Videre testing vil gå ut på å lagre verdiene i mottaker-Arduinoen på en slik måte at de kan brukes til å styre delsystemene . Det må kartlegges hvordan man kan sende sens-

ordata fra ATV-en til Processing.

Tabell 58: Test av HC-12, kommunikasjon begge veier

Test ID TP.13-O.13	Ansvarlig KVL	Ikke godkjent av KVL
Dato:	29.04.2019	
Krav:	O.13	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Arduino, Processing og HC-12	
Hvilken funksjon skal testes:	Trådløs styring og tilbakekobling	
Status:	Ikke godkjent	

Videreføringen av den forrige testen med HC-12 førte til en avgjørelse om og ikke bruke denne kommunikasjonsmetoden.

Det ble gjennomført tester hvor det ble sendt og mottatt informasjon med HC-12 paret samtidig, noe som viste seg og ikke være gjennomførbart uten veldig merkbar forsinkelse.

Grunnen til at man ikke kan kommunisere begge veier samtidig er fordi HC-12 paret er half duplex [16].

Det betyr at HC-12 paret ikke vil bli brukt i videre testing, men istedet vil det bli gjort forsøk med en Raspberry Pi koblet til en Arduino som skal sitte på ATV-en.

13.6 Software

13.6.1 Grensesnitt - PS4 kontroller med processing

Tabell 59: Test grensesnitt med PS4 kontroller og processing

Test ID TP.6-O.07	Ansvarlig SAM	Godkjent av SAM
Dato:	19.04.2019	
Krav	O.07 - FS.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	PS4 kontroller + processing	
Hvilken funksjon skal testes	Grensesnitt med kontroller	
Status:	Ikke godkjent	

Dette er første iterasjon til design av brukergrensesnittet. Første utkast til designet skulle vise seg å være svært konterintuitivt, da gassen er adressert til høyre skulderknapp på ps4-kontrolleren(R2), men er satt helt til venstre i figuren (225). Dette vil bli svært forvirrende når man skal operere kontrolleren, samtidig som 'displayet' skal være oversiktlig og vise verdiene hver enkelt pådrag gir ut. Dette vil derfor bli endret på før videre utvikling.



Figur 225: Gass til venstre, brems til høyre

Det mest åpenbare sluttresultatet til designet vil være i henhold til plasseringene på kontrolleren, slik at brems(L2) på venstre skulderknapp vil være til venstre på skjermen.

Alle andre funksjonelle knapper på kontrolleren vil også få sine respektive plasseringer i henhold til kontrolleren.

13.6.2 Trådløst med Raspberry PI

Tabell 60: Test av trådløs styring med Raspberry PI

Test ID	Ansvarlig	Godkjent av
TP.14-O.14	SAM	SAM
Dato:	30.04.2019	
Krav:	O.13	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Raspberry PI	
Hvilken funksjon skal testes:	Trådløs styring	
Status:	Ikke Godkjent	

Ved å bruke bluetooth i kontrolleren sammen med en Raspberry PI som kan festes på ATV-en vil det bli trådløs kommunikasjon til styringen. Alt av kode kan bli kjørt på PI-en, og arduinoen som er tilkoblet kan også sende tilbakemeldinger til en annen arduino via RF signalene som ble gjort i test TP.12-O.13. En Raspberry PI ble derfor koblet opp og Processing ble installert. Det viste seg kjapt at det ene biblioteket som er brukt, gamecontrol+, ikke støttes på Linux 32 Bit. Dette er det som PI-en må bruke, og derfor ble ikke dette en godkjent test og raspberry pi-en vil ikke bli brukt. Videre vil det bli gjort en vurdering på hva som er neste mulighet.

13.6.3 Skjermdeling

Tabell 61: Test skjermdeling - Discord

Test ID	Ansvarlig	Godkjent av
TP.17-FS.01	SAM	SAM
Dato:	02.05.2019	
Krav	FS.03	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Discord skjermdeling	
Hvilken funksjon skal testes	Skjermdeling	
Status:	Godkjent	

Etter møtet med ØG 02.05 ble det foreslått å bruke skjermdeling for å se tilbakemeldinger fra Arduino på ATV. Discord ble testet som en mulighet til dette. Det ble gjort ved

å teste med to forskjellige pc'er med hvert sitt mobilnett. Discord ga noe delay i overføringen, men det gikk kjapt nok til å si at dette er en godkjent løsning under prosjektet.

13.6.4 IR-sensor for hastighet

Tabell 62: Test av IR-Sensor - hastighet

Test ID TT.19-GA.01.01	Ansvarlig SAM	Godkjent av SAM
Dato:	06.05.2019	
Krav:	GA.01.01	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	IR Sensor - Arduino	
Hvilken funksjon skal testes:	Hastighetsmåling med IR-sensor	
Status:	Godkjent	

Det ble satt opp et enkelt miljø i processing, hvor kun IR-Sensoren ble brukt. Dette ble laget med tekstbokser for avlesning av sensorverdi, Δ Tid, altså tiden mellom hver måling, og hastighet som oppstår fra målingene. Ett objekt ble dratt over IR-sensoren for å skape utslag, og en illusjon om at eikene passerte. Dette ga ut riktige verdier i forhold til utregningene som ble gjort under utviklingen. Videre testing på dette vil innebære implementering på ATV-en.

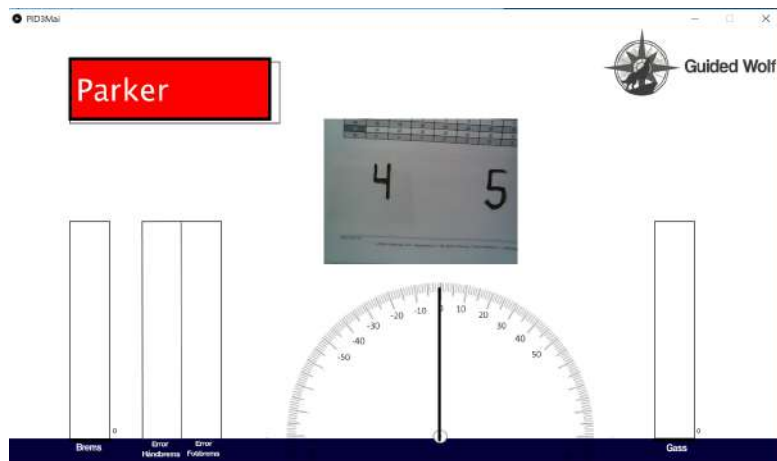
13.6.5 Video og objektgjenkjenning

Tabell 63: Test direkteoverført video i Processing

Test ID TP.18-GA	Ansvarlig KVL	Godkjent av KVL
Dato:	03.05.2019	
Krav	GA	
Verifikasjonsmetode:	Fysisk test	
Testobjekt:	Processing, kamera	
Hvilken funksjon skal testes	Video i Processing	
Status:	Godkjent	

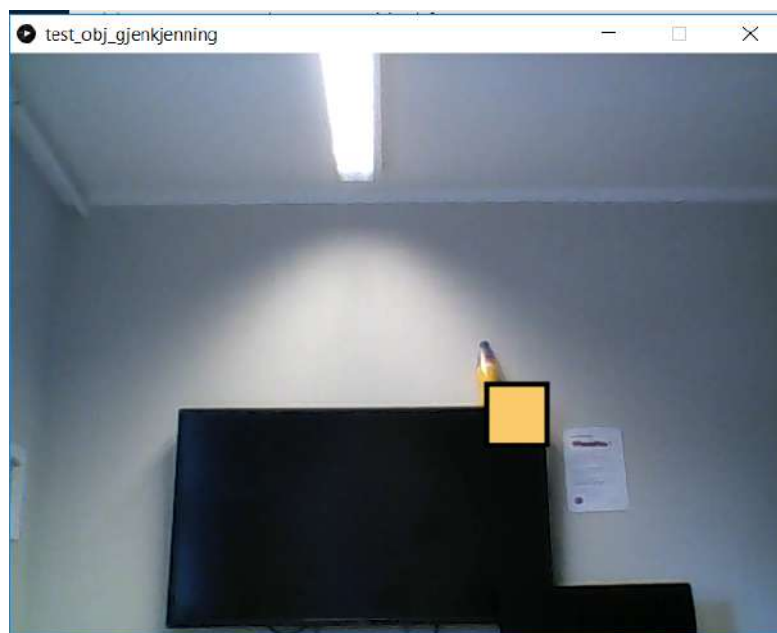
Denne testen ble gjennomført med bakgrunn av at gruppa ønsker å skrive en kode i Processing som kan gjenkjenne tall. Slik at speedometeret på ATV-en kan leses av og at tallverdier returneres til bruk i tilbakekoblinga for hastighet og eventuelt for å vise video forran og bak ATV-en, slik at operatøren kan se hvor man kjører.

Testen ble gjennomført ved å følge et eksempel som Processing selv har [72]. Koden ble så tilpasset slik at den passer til den eksisterende koden som er skrevet for styring av systemet.



Figur 226: Videotest i Processing

Det ble også gjort klart for videre testing med objektgjenkjenning, hvor en kode publisert i et Processing forum ble brukt [17], for å kunne følge valgte farger på en direktesendt video med et rektangel. Slik man kan se på bildet nedenfor.



Figur 227: Test med fargegjenkjenning i Processing

Denne metoden ble ikke testet videre på grunn av dens kompleksitet og fordi det ikke var hensiktsmessig å bruke mye tid på en slik løsning framfor å bruke IR-sensorer eller photodioder som nevnt tidligere.

Implementering

Her vises det hvordan de ulike delsystemene ble implementert.

Ikke alle komponenter ble implementert ved innleveringen av dette dokumentet, men vil bli implementert før 04.06.2019.

Innhold:

- ATV-en den 21.05.2019
- Kretsskjema
- Ramme, festepunkter og aktuatorplassering



14.1 ATV-en den 21.05.2019



Figur 228: Oversiktbilde av ATV-en, ovenfra

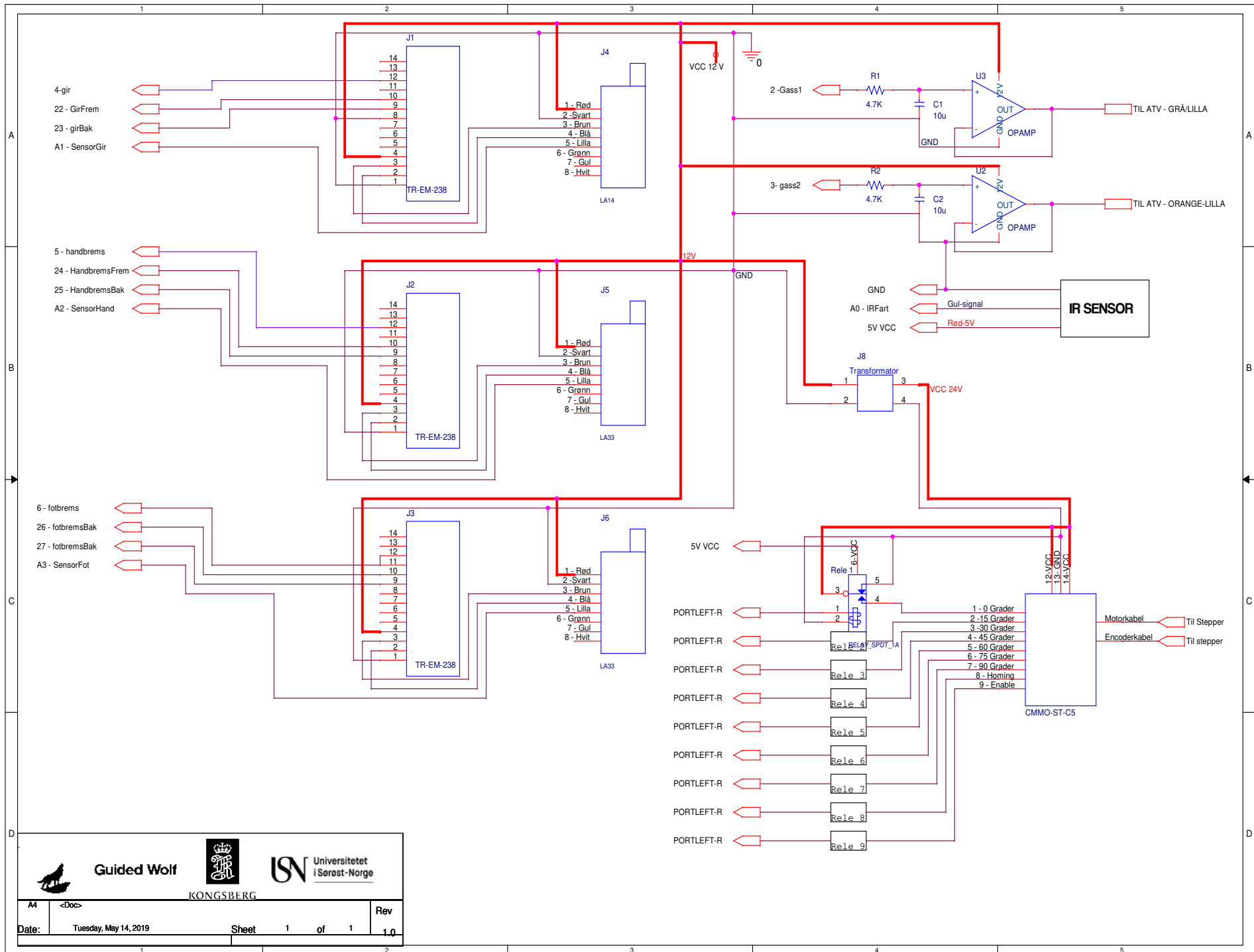


Figur 229: Oversikt-bilde av ATV-en, høyre side

14.2 Kretskjema

Kretsskjemaet gir en oversikt over hvilke plugger som er brukt i Arduinoen. Her har det blitt henvist til både pinnenummer, og hva de heter i koden. Noen snarveier har blitt tatt ved tegningen, som ved releene hvor kun første tilkobling har blitt vist.

Kretsskjemaet har blitt satt opp i OrCad - Capture.



Guided Wolf

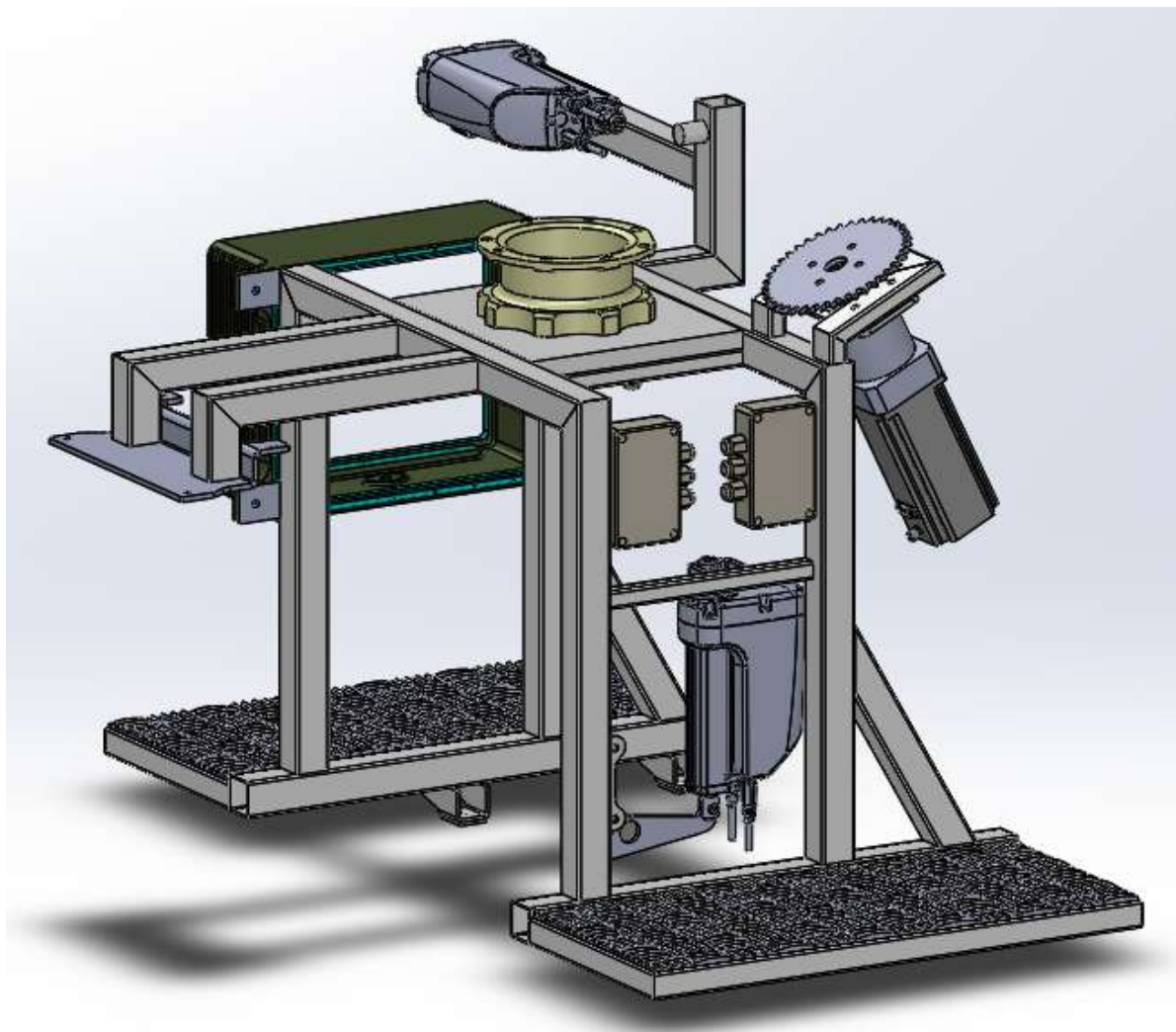


KONGSBERG

UN Universitetet i Sørøst-Norge

14.3 Ramme, festepunkter og aktuatorplassering

På figur 231 ser man hvordan og hvor alle aktuatorene er festet i rammen som sitter midt på ATV-en.



Figur 231: Oversikt-bilde av ramme, festepunkter og aktuatorplassering

Modell

Forskjellige modeller for prosjektet; kretsskjema, mekaniske modeller, 3D-print osv.

Innhold:

- Software - Processing, C++ og Arduino
- Mekanikk



KONGSBERG



USN Universitetet
i Sørøst-Norge

Gruppen har valgt å designe en modell/testplattform for å kunne visualisere testingen av programvare og kontroller. Det har i den anledning blitt designet en modellbil som vil bli styrt med en PlayStation-kontroller og kodet i programvarene Processing og Arduino. På denne måten kan man lære og få testet forskjellige kontrollmetoder man videre kan bruke for å styre servomotorer og aktuatorer via en datamaskin og kontrollere uten å bruke unødvendige penger på fysiske produkter.



Figur 232: Modell for prototype

15.1 Software

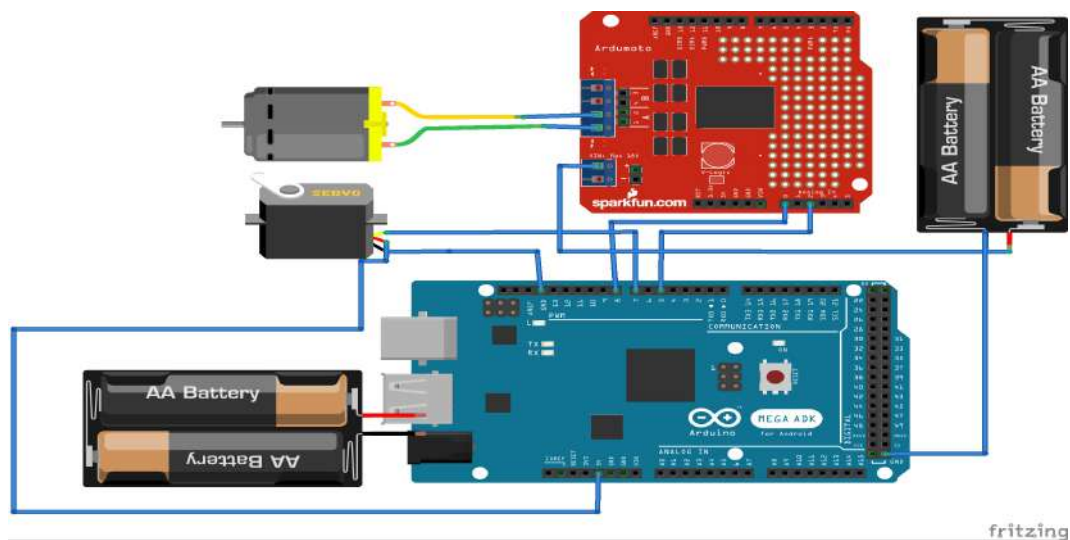
15.1.1 Første Presentasjon

Foreløpig kjøres modellbilen kun på en hastighet, men dette anses som et godt utgangspunkt for å kunne videreutvikles. ATV-en skal, i første omgang, hovedsakelig gå i gangfart, men ønsket er at den skal kunne kjøre i 50 km/t på sikt. Målet er derfor å kunne kjøre fremover i valgfri hastighet, samt ha en mulighet for å kunne sette den i revers. Siden kjøretøyet skal kunne styres av en datamaskin eller lignende, det ble valgt å koble til en spillkontroll (Playstation 3) for å teste styringen på protoypen.

Figur 233 illustrer oppkoblingen til prototype (modellbilen). Her vil det bli brukt en Arduino Mega 2560, DC-motor L298N, Servo-motor MG995, 9V strømadapter koblet til Arduino og 2 stk 9V batterier koblet i serie som strømforsyning til DC-motor.

Videre er servo-motoren koblet til pinne nr. 7, jord og 5V på Arduinoen. DC-motoren er koblet til polene på utgang 1 og 2, PWM kontrollerer hastigheten og er koblet til pinne 8 på Arduinoen, IN1 kontrollerer retning og er tilkoblet pinne 5. DC-motoren er også tilkoblet egen strømforsyning på 2x9V.

Kretsen er satt opp i Fritzing [23].



Figur 233: Krets til modellbil

Tabell 64: Oversikt over koblingspunkt til prototype

Pinnenr.	Koblingspunkt
Pin 5	IN1 - Retning DC-motor
Pin 7	Styring av servo
Pin 8	PWM - Hastighet DC-motor
GND	Servo, DC-motor driver (L298N), Batteri
5V	Servo
Ekst. koblingspunkt A	Ekst. koblingspunkt B
OUT 1, 2	DC-motor
+ 12V (L298N)	Batteri
Adapter 9V	Arduino

På de to neste sidene kommer skrevet koden med kommentarer for å styre servo og DC-motor fra programmet Processing. Kode for styring av servo er hovedsakelig hentet fra “Using a game controller with Arduino and Processing” [70], mens kode for DC-motor er en kombinasjon av eksempler fra nett og egen kode, slik at den har blitt tilpasset riktig bruk (prototype til første presentasjon) [51] [29].

```
//Nødvendige bibliotek for at Processing skal samhandle med Arduino
↪ og kontrolleren.
import controlP5.*;
import Processing.serial.*;
```

```
import cc.Arduino.*;
import org.firmata.*;
import net.java.games.input.*;
import org.gamecontrolplus.*;
import org.gamecontrolplus.gui.*;

ControlDevice cont;
ControlIO control;

Arduino Arduino;

ControlP5 controlP5;

float thumbS; //styring av servo, venstre/høyre
float fram; //gass dc-motor
int direction; //retning fram/tilbake

void setup() {
    size(400, 400);

    control = ControlIO.getInstance(this);
    cont = control.getMatchedDevice("cont6"); //Henter data for
    ↪ kontrolleren

    if (cont == null) {
        println("Error"); //Om forbindelsen blir brutt
        System.exit(-1);
    }

    println(Arduino.list());
    Arduino = new Arduino(this, Arduino.list()[1], 57600);
    Arduino.pinMode(7, Arduino.SERVO); //servo pin

    //for (int i = 0; i <= 13; i++)
    Arduino.pinMode(8, Arduino.OUTPUT); //fart pin
```



```

Arduino.pinMode(5, Arduino.OUTPUT); //retning pin
// pin8: PWM, pin 5: IN1

controlP5 = new ControlP5(this);
}

public void getUserInput() {
    //tildeler float verdi og gir tilgang til kontrolleren.
    thumbS = map(cont.getSlider("ServoPos").getValue(), -1, 1, 0,
        ↪ 180);
    fram = map(cont.getButton("Fram").getValue(), 0, 1, 0, 10);
}

void draw() {
    getUserInput();
    background(thumbS, 10, 85); //bakgrunnsfarge for styring
    background(fram, 86, 175); //bakgrunnsfarge for fart

    Arduino.servoWrite(7, (int)thumbS);
    Arduino.digitalWrite(8, (int)fram);

    if (direction == fram) { //kjøre fremover
        Arduino.digitalWrite(5, 1);
    }
}

```

15.1.2 Andre Presentasjon

Det ble ikke gjort mye utbedringer til andre presentasjon, men etter at målingene hadde blitt gjort på gassen og det var tydelig at det var to sensorer som satt på ATV-en ble det kodet opp signaler som skulle emulere disse to signalene. Dette ble gjort med mapping funksjonen som vist under:

```

gass = map(cont.getSlider("Gass").getValue(), -1, 1, 0, 255); // under
↪ Void

```

```
gass1=map(gass, 0, 255, 12, 75); // Lager to signaler gass1 og gass2
gass2=map(gass, 0, 255, 25, 150);
```

Disse to signalene ble så gjort om til DC verdier som vist i test TE1-GA.01. Disse to signalene ble så vist frem i waveforms under presentasjonen.

I tillegg til gass ble også en nødstopp knapp koblet opp med en while løkke som vist under:

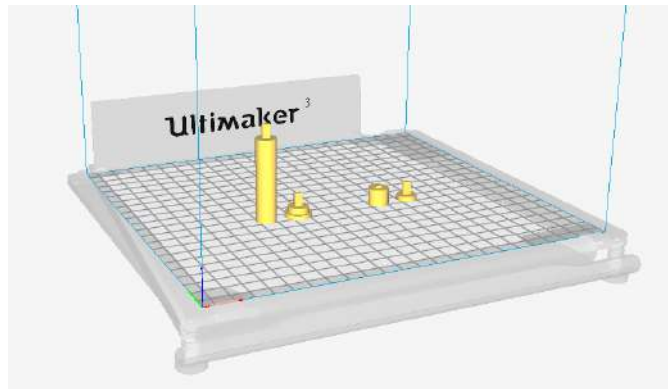
```
while (Arduino.digitalRead(2)==0) { //Dersom knappen holdes inne blir
  ↪ alt satt til 0.
  Arduino.analogWrite(7, 0);
  Arduino.analogWrite(6, 0);
  Arduino.analogWrite(5, 0);
}
```

15.2 Mekanikk

Med tanke på tidsbruk, kostnad og tilgjengelighet ble det valgt å designe modellen i størst mulig grad av plater som kan laserkuttes, ettersom dette fører til en rask og lite kostbar prosess når det gjelder utforming av fysiske deler. Deler som ikke lar seg laserkuttes, som f.eks. aksler til kraftoverføring og montering av hjul som allerede var tilgjengelig på skolen har blitt designet for å kunne 3D printes med 3D printerne som er tilgjengelig på skolen. I denne prosessen ble det designet prøvestykker til akslene, hvor kun endene som måtte passe overens med eksisterende deler ble printet. Dette ble gjort for å sikre størrelse og passform for montering av hjulene på akslene, og for å spare printetid, ettersom nøyaktigheten til en 3D printet del kan variere.

15.2.1 3D printing

Tidligere erfaring viser at de printede delene blir noe større enn hva CAD modellen tilsier, derfor ble innvendige former som hull og spor forstørret for at motor akselen skal passe inn i akselen. Ved design av utstikkende former må man designe motsatt enn ved innvendige, det vil si designe de mindre enn hva de skal være. Dette gjelder f.eks. akselender som skal passe inn i hjulene.



Figur 234: Fullstendig aksel og fremre innfestning (t.v.) og teststykker (t.h.)

Første test viste det seg at det hadde blitt tatt for liten høyde for unøyaktigheten/ forstørrelsen. Selv om nøyaktigheten på 3D printeren, i følge innstillingene i medfølgende programvare skal være én tiendedels millimeter (0,10 mm), ble de utvendige målene redusert med 15 hundredels millimeter (0,15 mm). Selv med en margin på nøyaktigheten ble de printede delene fra to tiendedeler – 25 hundredeler for store (0,2 - 0,25 mm), noe som gjorde at de ikke passet inn i hjulene. Til andre test ble det derfor avgjort å redusere størrelsen med ytterligere 25-35 hundredeler (0,25 - 0,35 mm), da totalt 35 - 50 hundredeler reduksjon (0,35 - 0,50 mm).

Andre test-print resulterte i tilfredsstillende størrelse og passform, av denne grunn ble beslutningen om å printe ut komplette deler tatt. Resultatet av den komplette printen

ble variable, hvor noen deler ble tilfredsstillende, mens andre ble annerledes enn forventet.

Fremakslene/innfestningene til framhjulene som var tilnærmet like i størrelse som teststykkene som allerede hadde blitt printet og godkjent med tanke på størrelser og passform, fikk tilnærmet samme passform som teststykkene. Dette gjorde at de passet inn i hjulene og satt tilstrekkelig hardt, slik at annen innfestning ikke vil være nødvendig.

Akslene mellom motor og bakhjulene er lenger enn teststykkene, og ettersom de blir printet på høykant vil de derfor være mye høyere i 3D printeren (Se Figur 234). Dette resulterte overraskende i at størrelsen på endestykkene og passformen ble helt annerledes enn de lave fremre. Enden av akselen som skulle passe inn i hjulene, som opprinnelig har samme dimensjoner som de fremre ble mye mindre enn hva teststykkene og den fremre innfestningen tilsa. Disse endene ble hele 45 hundredeler tynnere enn hva innfestningene til framhjulene ble, selv om de i teorien skal være identiske. Dette gjorde at de kom innenfor den opprinnelige toleransen på én tiendedels millimeter (0,1 mm) fastsatt i det medfølgende programmet til 3D printeren. Ettersom den designede CAD modellen hadde blitt endret og redusert i størrelse og tilpasset etter de printede testene som viste langt større unøyaktighet, var disse endene for små til å passe i hjulene. Det resulterte i at hjulet ikke satt fast ved montering, det ble derfor nødvendig å lime hjulene fast i akselene.

15.2.2 Laserkutting

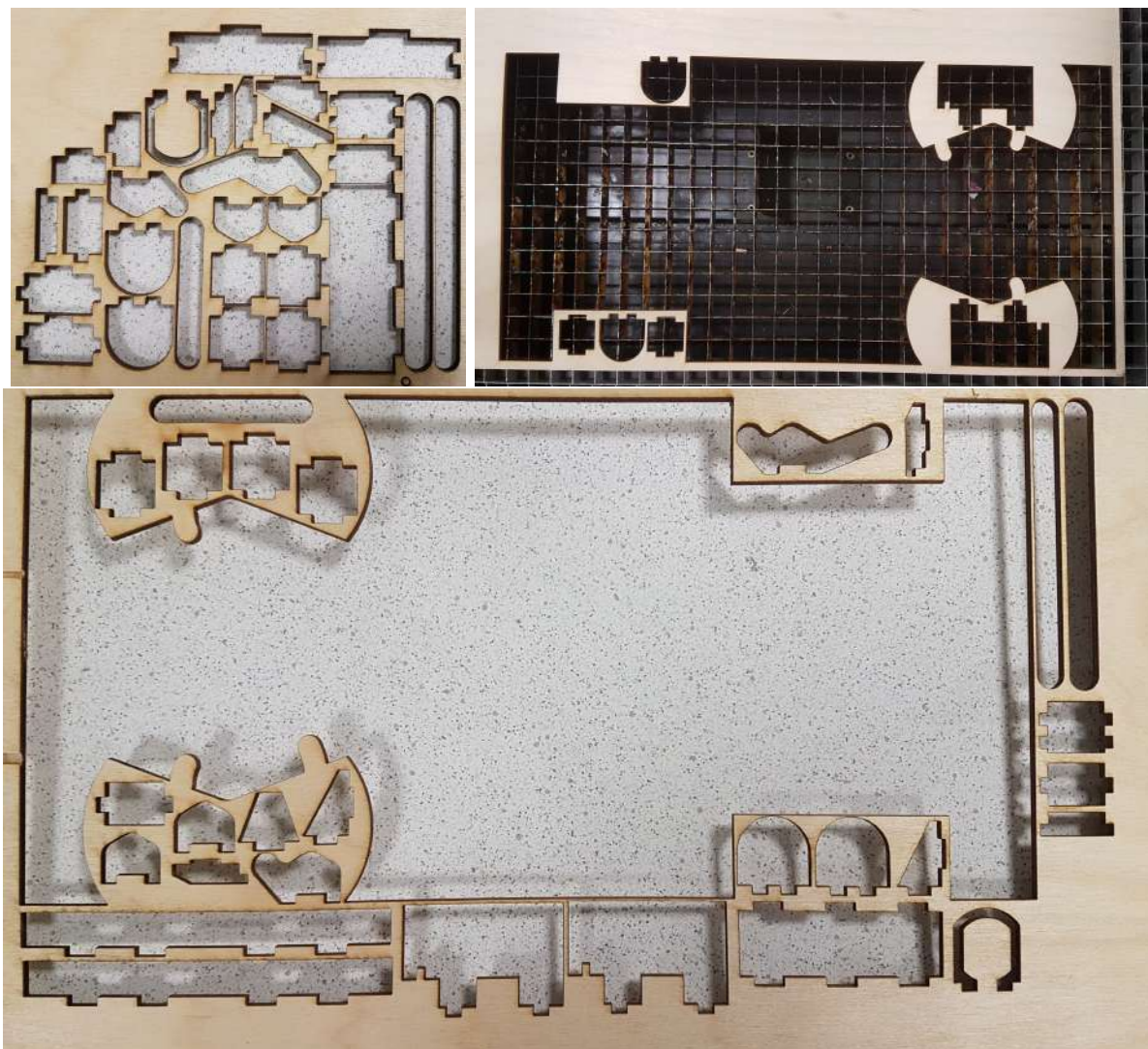
For å laserkutte deler må man først designe delene i et CAD program, som f.eks. SolidWorks som er brukt til denne modellen. Første steg i prosessen er å designe de forskjellige delene i et 3D rom, hvor man designer et volum med en form som tilsvarer delen man vil ha ut som ferdig produkt. Når man designer disse delene må man alltid tenke på at alle former for materialfjerning skal skje kun på ett plan (i 2D), ettersom en laserkutter ikke har mulighet til å fjerne materiale i en tredje akse/i høyderetning og at det lønner seg med andre festemetoder enn konvensjonelle skruer og bolter. Av denne grunn har laserkutting på lik linje med vannskjæring et begrenset bruksområde i forhold til andre materialfjerningsprosesser som konvensjonelle bearbeidingsmetoder som dreining og freising.

Ettersom man designer de forskjellige delene vil det være normalt å legge delene inn i

en sammenstillingsfil, dette gjør man for å kontrollere at delene passer i hverandre og er kompatible med andre deler man designer for, f.eks. maskinering, 3D printing, eller deler som allerede eksisterer. Her vil man altså kunne se om ting kolliderer med hverandre, slik at man er sikker på at delene fysisk vil fungere sammen. En sammenstilling er også en fin plattform for å undersøke om sammenkoblede, bevegelige deler vil fungere sammen som antatt.

Når alle delene er ferdigdesignet og testet i en sammenstilling, legger man alle delene ut på et 2D ark i samme programvare. De ulike delene orienterer man i den retningen slik at man ser konturen/strekene man vil at laserkutteren skal følge. Her må man kontrollere at programmet ikke har lagt inn ekstra linjer, som f.eks. sentermarkeringer av hull og andre senterlinjer, disse må man eventuelt fjerne. Det lønner seg også å orientere delene litt kompakt, slik at man får et mindre arbeidsområde til neste steg, men ikke for tett ettersom man må markere del for del i neste steg.

Når 2D tegningen er komplett eksporterer man tegningen i .DXF format slik at filen vil bli lesbar for et program som heter FlexiDESIGNER. FlexiDESIGNER forstår kun hvor en linje er og hvor lang den er, derfor må man fortelle programmet hvilke linjer som hører sammen til å bli en del, dette er grunnen til at man ønsker litt mellomrom mellom delene, slik at de skal være enkle å markere hver for seg. Deretter flytter man de grupperte delene slik at man får utnyttet arealet maksimalt, for å minimere materialkostnaden (Figur 235).



Figur 235: Ferdigkuttete plater

Bestillingsoversikt

Tabell 65 viser en oversikt over alle komponenter som er bestilt og andre kostnader vedrørende maskinering av deler til ramme og lignende.

Bestillingsoversikten er delt inn i ulike varegrupper hvor hver gruppe har en egen bestillings ID etterfulgt av et tall som gir hver del et særeget nummer. Komponentene er sortert inn under følgende varegrupper og ID:

- **Software:** B-SW.xx
- **Aktuator:** B-A.xx
- **Servostyring:** B-ST.xx
- **Produksjon:** B-P.xx

Tabell 65: Bestillingsoversikt

BESTILLINGSOVERSIKT					
Bestillings ID	Vare	Leverandør	Antall	Pris	Sum
B-SW	<i>Varegruppe - Software</i>				
B-SW.01	HC-12 Wireless Serial Module Remote 1000M With Antenna	<i>Banggood.com</i>	2	25,98	51,96
B-SW.02	3pk DAC Breakout Development Board Module	<i>Banggood.com</i>	1	40,66	40,66
B-SW.03	Arduino Mega 2560	<i>Banggood.com</i>	2	73,77	147,54
B-SW.04	USB 2.0 Cable A Male to B Male Cord	<i>Banggood.com</i>	1	35,71	35,71
B-SW.05	USB 3.0 4-Port Splitter Hub	<i>Banggood.com</i>	1	62,56	62,56
B-SW.06	40pk Male to Male Color Breadboard Cable Jump Wire Jumper	<i>Banggood.com</i>	1	17,29	17,29
B-SW.07	310 pk jumper header connectors	<i>Banggood.com</i>	1	45,62	45,62
B-SW.08	Proximity sensor	<i>Banggood.com</i>	4	34,67	138,68
	Frakt		1	33,40	33,40
	<i>Sum Software (inkl. mva)</i>				573,42
B-A	<i>Varegruppe - Aktuator</i>				
B-A.01	Aktuator LA14	<i>Linak</i>	1	2821,25	2821,25
B-A.02	Aktuator LA33	<i>Linak</i>	2	3977,50	7955,00
B-A.03	Strømbegrenser TR-EM	<i>Linak</i>	3	1025,00	3075,00
B-A.04	Strømsensor	<i>Linak</i>	2	98,40	196,80
B-A.05	KOPLINGSKINNE BEP m/deksel sort/rød 2x16/2	<i>Batteri og Radiatorservice</i>	1	347,50	347,50
	Frakt		1	35,84	35,84
	<i>Sum Aktuator (inkl. mva)</i>				14431,39
B-ST	<i>Varegruppe - Servostyring</i>				
B-ST.01	Motorkontroll CMMO-ST-C5-1-DIOP	<i>Festo</i>	1	3504,59	3504,59
B-ST.02	Steppermotor EMMS-ST-87-L-SE-G2	<i>Festo</i>	1	4070,51	4070,51
B-ST.03	Motorledning NEBM-S1G15-E-2.5-Q7-LE6	<i>Festo</i>	1	773,41	773,41
B-ST.04	Encoderledning NEBM-M12G8-E-2.5-LE8	<i>Festo</i>	1	574,33	574,33
B-ST.05	Styreledning NEBC-S1G25-K-3.2-N-LE25	<i>Festo</i>	1	215,40	215,40
B-ST.06	W5100 Ethernet Shield Arduino	<i>ebay.co.uk</i>	1	153,52	153,52
B-ST.07	RS485 TTL - Breadboard Module Arduino	<i>ebay.co.uk</i>	1	63,90	63,90
B-ST.08	VICTRON Orion omformer 12V/24V-20	<i>Batteri og Radiatorservice</i>	1	1005,00	1005,00
B-ST.09	Gir 9:1 - Relex AG, CH-5645 Jona - PLE 80-9/OP2+6	<i>Festo</i>	1		
	<i>Sum Servostyring (inkl. mva)</i>				10360,66
B-P	<i>Varegruppe - Produksjon</i>				
B-P.01	Vannskjæring av komponenter	<i>Vannskjæresenteret AS</i>	15		
B-P.02	Sveis av ramme	<i>Elor Plate og Sveis AS</i>	1		
B-P.03	Plastikkdeksel til gir	<i>Kajander Motor AS</i>	1	140,00	140,00
B-P.04	Tannhjul	<i>Kjell Berg AS</i>	2	295,00	590,00
B-P.05	Kjede	<i>Kjell Berg AS</i>	2	295,00	590,00
	Frakt		1	101,00	101,00
	<i>Sum Produksjon (inkl. mva)</i>				1421,00
	<i>Sum totalt (inkl. mva)</i>				25365,47

Oppsummering

Det endelige målet til oppdragsgiver er å omforme en ATV om til en helautonom ATV med våpenstasjon og en plattform for drone med mulighet for å tilbake stille ATV-en til opprinnelig stand. Løsningene har båret preg av at systemet må bygges rundt ATV-en istedenfor å endre kjøretøyet for best mulige løsninger.

I denne perioden har det blitt satt krav til hva systemet bør ha, det har blitt bygget en ramme rundt ATV-en slik at det er plass til våpenstasjon og aktuatorer. Dette har blitt bygget slik at motorene kan utgjøre arbeid som å bremse, gire og svinge.

Videre anbefalinger

18.1 Mekanisk design

Beskyttelser til diverse delsystem burde utvikles.

- Håndbrems
- Gir
- Styre
- Eksponerte områder på ATV-en etter demontering

18.1.1 Håndbrems

Hvis en ny innretning skal designes så må det passes på at væskebeholderen holdes rett opp. Heller beholderen på siden så har den en tendens til å lekke ut væske.

18.1.2 Gir

Når ATV-en er avskrudd kan giret henge seg opp. Ved å vugge litt på kjøretøyet pleier å løse det opp. Når ATV-en er påskrudd så skal ikke dette skje.

Anbefalt å implementere et kuleledd eller lignende i leddet fra rammen til el motoren slik at det kan lettere kan skifte gir uten unødvendige spenninger

18.1.3 Styre

Det er muligheter for at en annen kraftoverføring er bedre. Skal kjededrift forbli løsningen, burde det lages en beskyttelse for kjedet, samt finne en rutine på hvor mye vedlikehold kjedet trenger. Det er mulig at tannhjulene ikke er nøyaktig i samme plan. Det vil si at tannhjulene kan ha høydeforskjell eller vinkelforskjell. Det fører til økt slitasje og det anbefales å vurdere hvor stor innvirkning dette vil ha på levetiden.

18.1.4 Eksponerte områder på ATV-en etter demontering

Etter demonteringen og da rammen ble montert på ATV-en er det mye av ATV-ens komponenter som vil være eksponert for støv, gjørme og andre påvirkninger fra omgivelsene.

Dette er det tatt høyde for ved at det er ledig plass mellom ramma og ATV-en, slik at deksler kan designes og implementeres for å beskytte ATV-ens komponenter.

Det anbefales ved videre arbeid på prosjektet at det lages deksel for spesielt ATV-batteriet og diverse eksponerte mekaniske deler.

18.2 Elektrisk design

- Strømsensor
- IR-sensorer - hastighetsmåling
- Video
- Modbus/styring
- Nødstop - funksjon
- Nødstop - direkte knapp
- Batteri
- Trådløs overføring

18.2.1 Strømsensor

Det ble gjort forsøk med å bruke strømsensorer for å kunne stoppe aktuatorene fra å kjøre ved for høyt strømforbruk og derfor utøve for mye krefter på bremse- og girsystemet.

Dette ble ikke implementert i tidsperioden gruppen hadde, men det kan anbefales å implementere dette eller noe liknende med samme funksjon for å værne de aktuelle delsystemene.

Denne testen ble dokumentert som test (TP.32-BR.01).

18.2.2 IR-sensorer - hastighetsmåling

Det ble undersøkt og skrevet kode for bruken av IR-sensorer for å måle hastigheten til ATV-en. Teorien og koden for dette ligger under delkapittel (13.9).

Selv om denne metoden å måle hastighet på fører til en del feilmålinger og vil gi litt upresise målinger anbefales det videre at et liknende oppsett eller noe med samme funksjon lages for å kunne gi tilbakemeldinger på hastighet, for operatøren å se og for tilbakekoblinger.

18.2.3 Video

Det er gjort klart i koden å bruke kameraer slik at operatøren kan se hvor man kjører. Dette ble testet (TP.18-GA) ut og bekreftet at fungerte.

Kameraer vil også bli nødvendige i arbeidet til de neste gruppene som skal automatisere ATV-en.

18.2.4 Modbus/styring

For videre utvikling, kan det være hensiktsmessig å gå til anskaffelse av en PLC. Man kan programmere opp motoren gjennom en Controllino (Arduino for industrielt bruk) og få I/O logikk med 24 V.

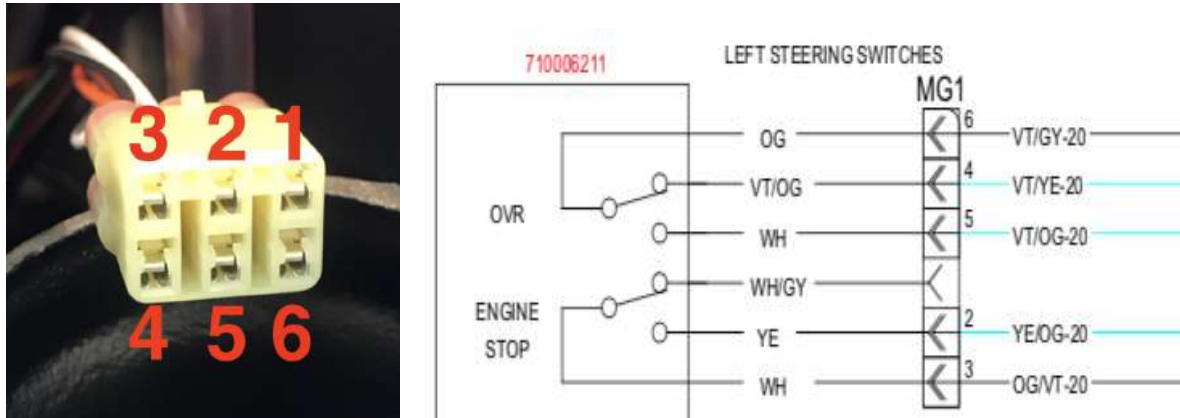
18.2.5 Nødstopp - funksjon

Nødstopp funksjonen i dag stopper kun gasspådraget. Denne burde videreutvikles til å bremse for så å gire til park og skru av tenningen.

18.2.6 Nødstopp - direkte knapp

Under testingen av ATV-en var det et krav fra oppdragsgiver om å ha et kablet nødstoppsystem i tillegg til eventuelle andre.

Det anbefales for de som skal jobbe videre med prosjektet å ha et slikt oppsett, spesielt under testing.



Figur 236: Kretsskjema - koblingspunkt 2

18.2.7 Batteri

Det vil muligens bli kjøpt inn et enkelt oppladbart 12V batteri etter innleveringen av denne rapporten (23.05.2019). Batteriet vil bli brukt under de siste implementeringstestene fram til prosjektslutt (04.05.2019).

Det vil være nødvendig for de som skal jobbe videre med prosjektet å kjøpe inn et batteri med nok kapasitet til å kjøre alle aktuatorene og steppermotoren, i et bestemt antall timer.

Steppermotoren bruker maks 40A, LA33 maks 10A, LA14 maks 2,6A.

For å kjøre aktuatorene i en time vil det mest sannsynligvis ikke være behov for et batteri på 52,6Ah da aktuatorene ikke vil være aktive bestandig.

18.2.8 Trådløs overføring

Det har blitt gjort mye testing med bluetooth og RC moduler. Dette burde gås bort fra fordi det ikke var mulig å sende og motta informasjon samtidig eller uten veldig merkbar forsinkelse. Det anbefales heller å se på andre overføringsmetoder, som WiFi eller 4G.

Referanser

- [1] AlDdatasheet. http://www.aalco.co.uk/datasheets/Aluminium-Alloy_6082-T6~T651_148.ashx.
- [2] Anbefalte transistorer. <https://www.baldengineer.com/the-best-4-transistors-to-keep-in-your-parts-kit.html>.
- [3] Arduino kode: Styling av stepper med arduino. <https://docs.google.com/document/d/1BvvDBk1zSzg8pQInKuTp5ZR-NpzIiTnbwM5ZTQZ4vmw/edit?usp=sharing>.
- [4] Arduino Serial til Processing. <https://electronics hobbyists.com/arduino-processing-tutorial-connect-arduino-to-processing/>.
- [5] Atv hentet fra grabcad, laget av brukeren bresson roch. <https://grabcad.com/library/atv-4-weels-drive-650cc-1>.
- [6] Batteri Konsept 1. https://www.batteryspace.com/productimages/li-ion_box_battery/5867.jpg.
- [7] Bilde av PS4-kontroll. https://www.skvis.no/spill/playstation-4/sony-dualshock-4-controller-v2-sort-ps4?gclid=Cj0KCQjwzunmBRDsARIsAGrt4ms4V1m2Q0aMMP785X2XhN064LgL_zptPrt5sb_1kOfNKNQjcEJf1VEaAh1YEALw_wcB.
- [8] Bipolar transistor. https://www.electronics-tutorials.ws/transistor/tran_1.html.
- [9] Circuit Lab - Circuit simulation and schematics. <https://www.circuitlab.com/>.
- [10] Datablad for Ee board. <https://reference.digilentinc.com/reference/instrumentation/electronics-explorer/reference-manual?redirect=1>.
- [11] Datablad og informasjon om TR-EM-288. <http://www.linakthirdparty.com/products/drivers.aspx?product=TR-EM-288-SPF+-+Positioning+driver>.
- [12] Datablad transistor 2N3904. <https://www.onsemi.com/pub/Collateral/2N3903-D.PDF>.
- [13] Datablad transistor BUF620. <https://pdf1.alldatasheet.com/datasheet-pdf/view/113443/VISHAY/BUF620.html>.

- [14] Datablad transistor BUF742. <https://pdf1.alldatasheet.com/datasheet-pdf/view/113517/TEMIC/BUF742.html>.
- [15] Datablad transistor MJE350. <https://www.onsemi.com/pub/Collateral/MJE350-D.PDF>.
- [16] Dupleks (kommunikasjonsteknologi). [https://no.wikipedia.org/wiki/Dupleks_\(kommunikasjonsteknologi\)](https://no.wikipedia.org/wiki/Dupleks_(kommunikasjonsteknologi)).
- [17] Fargegjenkjenning, bruker: JJJJJJ. <https://forum.processing.org/two/discussion/22769/how-to-do-this-using-color-tracking>.
- [18] Festo - Control Via Ethernet (CVE) . <https://www.festo.com/net/SupportPortal/Files/345406/100002.pdf>.
- [19] Festo - Motor controllers CMMO-ST - Datablad. https://www.festo.com/cat/sk_sk/data/doc_engb/PDF/EN/CMMO-ST_EN.PDF.
- [20] Festo - Stepper motors EMMS-ST - Datablad. https://www.festo.com/net/SupportPortal/Files/212956/13_1-18%20-%20Stepper%20motors%20EMMS-ST_ENUS.PDF.
- [21] Festo Description - Motor controller with I/O interface. <https://www.schneider-electric.co.in/en/faqs/FA168406/>.
- [22] Forum post om feil i Standardfirmata. <https://github.com/firmata/arduino/issues/334>.
- [23] Fritzing - Building a circuit. <https://www.fritzing.org/building-circuit/>.
- [24] Gantt diagram til nettside. <https://developers.google.com/chart/interactive/docs/gallery/ganttchart>.
- [25] Gradeskive protactor - kisscc0bli. <https://www.kisscc0.com/clipart/protractor-degree-circle-turn-angle-rapporteur-pwydmf/>.
- [26] Graf til 9:1 gir . https://drive.google.com/drive/folders/1bYZt1biUdaJIT0ZNLMyYXd5fZhFeBlg_.
- [27] HC-12 bilde. <https://www.makerlab-electronics.com/product/433mhz-serial-rf-module-hc-12-1000m/>.

- [28] How Does a PID Controller Work? <https://www.elprocus.com/the-working-of-a-pid-controller/>.
- [29] How to Use L298n to Control Dc Motor With Arduino. <https://www.instructables.com/id/How-to-Use-L298n-to-Control-Dc-Motor-With-Arduino/>.
- [30] HP 15-da0803no 15,6 bærbar PC. <https://www.elkjop.no/product/data/barbar-pc/10440/hp-15-da0803no-15-6-barbar-pc-sort>.
- [31] Hva er Kanban. [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development)).
- [32] Hydraulikk pumpe. https://www.hydraulikkdeler.no/assets/img/630/630/bilder_netbutikk/16c3f4fdb893e37368d6b6bbb2199cc-image.jpeg.
- [33] I Built A Life-Size Remote-Controlled Corvette | RIDICULOUS RIDES. <https://www.youtube.com/watch?v=g3ruVMeQ2Ss>.
- [34] Informasjon om Beltedrift fra et selgende firma. <https://www.superatv.com/>.
- [35] Informasjon om Beltedrift fra et selgende firma. https://www.festo.com/cms/no_no/index.htm.
- [36] Informasjon om Beltedrift fra et selgende firma. <http://en.robotis.com>.
- [37] Introduction to Modbus. <https://www.automation.com/library/articles-white-papers/fieldbus-serial-bus-io-networks/introduction-to-modbus>.
- [38] Kongsberg Defence and Aerospace. <https://www.kongsberg.com/en/kds/aboutus/>.
- [39] Kongsberg Gruppen. https://no.wikipedia.org/wiki/Kongsberg_Gruppen.
- [40] Konisk Tannhjul. <https://www.kggear.co.jp/en/kg-stock-gears/miter-gears>.
- [41] Lucidchart - Online Diagram Software Visual Solution. <https://www.lucidchart.com/>.
- [42] Motstandsmoment. <https://no.wikipedia.org/wiki/Motstandsmoment>.

- [43] NPN transistor. https://www.electronics-tutorials.ws/transistor/tran_2.html.
- [44] Opamp 358 - Datablad. <http://www.ti.com/lit/ds/symlink/lm2904-n.pdf>.
- [45] OS-templates Baseend. <https://www.os-templates.com/>.
- [46] Pinmapping til Atmega2560. <https://www.arduino.cc/en/Hacking/PinMapping2560>.
- [47] Pololu ACS71 - Strømsensor. <https://www.pololu.com/product/1187>.
- [48] Processing - rect(). https://processing.org/reference/rect_.html.
- [49] Processing - trigonometri. <https://processing.org/tutorials/trig/>.
- [50] PWM to DC. <http://henrysbench.capnfatz.com/henrysbench/arduino-projects-tips-and-more/arduino-lm358-op-amp-pwm-to-voltage-converter/>.
- [51] RC Hobby Controllers and Arduino. <https://www.sparkfun.com/tutorials/348>.
- [52] Read Analog Data Directly in Processing. <https://www.instructables.com/id/Read-analog-data-directly-in-Processing/>.
- [53] reisverk. <http://www.hus.no/glasstak-reisverk.html>.
- [54] Rådrev. <https://www.kjell-berg.no/kjededrift/raadrev/raadrev-5/16-1>.
- [55] Spillkontroll til Playstation 3. <https://www.netonnet.no/art/gaming/spillogkonsoll/tilbehorkonsoll/tilbehorkontroll/sony-wirelessdualshock3ps3/175563.13152/>.
- [56] StepDatablad. https://www.festo.com/net/SupportPortal/Files/212956/13_1-18%20-%20Stepper%20motors%20EMMS-ST_ENUS.PDF.
- [57] Tegnesoftware til krets. <https://www.circuit-diagram.org/editor/>.
- [58] Tekniske spesifikasjoner LA12. <https://cdn.linak.com/-/media/files/data-sheet-source/en/linear-actuator-la12-data-sheet-eng.ashx>.
- [59] Tekniske spesifikasjoner LA14. <https://www.linak.no/produkter/lineaere-aktuatorer/la14/>.

- [60] Tekniske spesifikasjoner LA28. <https://www.linak.no/produkter/lineaere-aktuatorer/la28/>.
- [61] Tekniske spesifikasjoner LA30. <https://cdn.linak.com/-/media/files/data-sheet-source/en/linear-actuator-la30-data-sheet-eng.ashx>.
- [62] Tekniske spesifikasjoner LA33. <https://www.linak.no/produkter/lineaere-aktuatorer/la33/>.
- [63] Tekniske spesifikasjoner LA36. <https://www.linak.no/produkter/lineaere-aktuatorer/la36/>.
- [64] Thumbthrottle. <https://boostatv.com/wp-content/uploads/2018/07/thumb-throttle-close.jpg>.
- [65] Torsjon i CBR1000RR. https://www.motorcyclespecs.co.za/model/Honda/honda_cbr1000rr%2008.htm.
- [66] Torsjon i Ninja1000. <https://www.kawasaki.com/products/2019-Ninja-1000-ABS#specs-scroll>.
- [67] Transistor as a switch. https://www.electronics-tutorials.ws/transistor/tran_4.html.
- [68] Transistor som bryter. <https://www.youtube.com/watch?v=kNVaIqmKUoI>.
- [69] Udemy - PID control with Arduino. <https://www.udemy.com/pid-control-with-arduino/learn/lecture/10009562#overview>.
- [70] Using a game controller with Arduino and Processing. <https://www.makeuseof.com/tag/arduino-robot-game-controller/>.
- [71] veltebur. <https://www.pedersenracing.no/butikk/beskyttelsesbur/bmw/bur-bmw-e36-coupe-ar-90-00-versjon-2/>.
- [72] Video i Processing. <https://processing.org/tutorials/video/>.
- [73] What is Modbus and How does it work? <https://www.schneider-electric.co.in/en/faqs/FA168406/>.
- [74] What is scrum? <https://www.scrum.org/resources/what-is-scrum>.
- [75] Wikipedia Scrum. [https://en.wikipedia.org/wiki/Scrum_\(rugby\)](https://en.wikipedia.org/wiki/Scrum_(rugby)).

- [76] Eric Brown. RC kontroller. <http://linuxgizmos.com/hackable-drone-controller-runs-linux/>.
- [77] Freepik. Ulv til bruk i logo. <http://www.freepik.com>.
- [78] Gene F. Franklin; J. David Powel. *Feedback control of dynamic systems - seventh edition*. PEARSON, 2015.
- [79] Gene F. Franklin, J, David Powell Abbas Emami-Naeini. *Feedback Control of Dynamics Systems*. Pearson, 2015.
- [80] Lentin Joseph; Jonathan Cacace. *Mastering ROS for Robotics Programming - Second Edition*. Packt Publishing, 2018.
- [81] R.S. KHURMI J.K. GUPTA. *A Textbook of Machine Design*. EURASIA PUBLISHING HOUSE (PVT.) LTD., 2016.
- [82] Typodermic Fonts. Font til bruk i logo. <https://www.dafont.com/coolvetica.font>.
- [83] Vascer - Brukernavn - Wikimedia. Kompass til bruk i logo. https://commons.wikimedia.org/wiki/File:Gray_compass_rose.svg.
- [84] YoobSeok Pyo, HanCheoi Cho, RyoWoon Jung, ToeHoon Lim. *ROS: Robot Programming*. ROBOTIS Co, LTD, 1505,145,Gasan Digital 1-ro, Seoul, Republic of South Korea, 2017.

Vedlegg

I vedlegg ligger følgende:

- A.1 Arduino: Standard Firmata
- A.2 Processing: Game Control Plus, Configurator
- A.3 Processing kode
- A.4 Google Drive filer: Vedlagt på minnepenn
- A.5 Demonteringsoversikt
- A.6 Teknisk Håndbok
- A.7 Kraftmålinger på ATV
- A.8 Kretsskjema for ATV
- A.9 FEM analyse ramme og innfestning
- A.10 FEM analyse, overgang steppermotor-tannhjul
- Styret stagløsning



KONGSBERG



USN Universitetet
i Sørøst-Norge

A.1 Arduino: Standard Firmata

https://docs.google.com/document/d/1hdLA515r08bCM0vR8o8V07Y1ftY0wQZ7PboxyH_iiL8/edit

A.2 Processing: Game Control Plus, Configurator

https://docs.google.com/document/d/1oTDbN_WmTniK0tFVheYnFC362g7--GXXxDVF1F-qWi4/edit

A.3 Processing kode

<https://drive.google.com/drive/folders/1AiXz1LBdIU00xBFWilkqr1Q4jxEUnx0c?usp=sharing>

A.4 Google Drive: Vedlagt på minnepenn

- Demontering
 - Demonteringsoversikt
 - Demontering - link overleaf
 - Kraftberegninger (2. møte ATV)
 - Bilder av ATV
- Festo - styring
 - Diverse bilder og datablad
- Generelt
 - Kontrakter, gruppeoversikt og vedlegg
 - Logo
 - Referater og møteinnkallinger
 - Retningslinjer fra USN
 - Telefon plakat
- Konsept

- Beslutninger og begrunnelser
- Pugh-matriser
- Forslag til implementering av løsninger
- Oversikt over konsepter
- Visualiserte konsepter
- Kontrollsystem - UML

- Krav
 - User case
 - Krav

- Modeller
 - Gantt og kalender
 - Modeller - krav
 - Modeller - maskin
 - Prosjektmodeller(IML, flowcharts o.l)
 - Prosjektkalender

- Presentasjoner
 - 1. presentasjon
 - 2. presentasjon
 - 3. presentasjon
 - Presentasjoner-info

- Research
 - Aktuator
 - Forhandlere - mailer
 - ROS
 - Scrum

- Risiko

- Prosjektrisiko
- Teknisk risiko
- Software
 - Arduino og Processing
 - Bestilling av deler - Arduino
 - Bluetooth - HC12
 - Stepper-motor
- Tek-el
 - Arduino-eksempel
 - Feedback-teori
 - Gass/sensor
 - Koblinger styre
 - Kretsskjema for ATV-en
- Tek-ma
 - DAK
 - Kraftberegninger/forsøk
 - Oversikt over maskinkomponenter
 - Rapport, styresystemet
- Test
 - Aktuatorer
 - Alle tester
 - Fjærstivhet
- Timeliste

A.5 Demonteringsoversikt



Guided Wolf

Demonteringsoversikt CAN-AM OUTLANDER XT 650 T



KONGSBERG

USN Universitetet
i Sørøst-Norge

Innhold

1	Demontering	2
2	Montering av fangere og bagasjebrett	3
2.1	Front	3
2.2	Bak	8
3	Montering av deksel til gir	14
4	Montering av fothvilere og deksel	15
4.1	Feste til fothvilere	15
4.2	Deksel høyre side	16
4.3	Deksel venstre side	24
4.4	Festeanvisning - fothvilere og hjulbuer	31
5	Montering av sete og festepunkter	33
6	Styret	41
7	Monteringsselementer	42

1 Demontering

Dette dokumentet viser en oversikt over demonterte deler fra ATV-en med tilhørende monteringslementer. Bildene fra demonteringa vil komme i baklengs rekkefølge slik at det blir enklere å montere sammen igjen.

2 Montering av fangere og bagasjebrett

I denne seksjonen er bildeoversikt til feste av fanger og bagasjebrett både foran og bak på ATV. Disse festes i samme rekkefølge som bildene kommer.

2.1 Front

Rekkefølge for feste av deler:

- **1:** Figur 1
- **2:** Figur 2
- **3:** Figur 3

Monteringselementer:

- 2 × skruer “bagasjebrett foran”
- Alle resterende skruer, skiver og muttere er skrudd fast i de demonterte delene.



Figur 1: Fanger foran



Figur 2: Festestenger til bagasjebrett foran



Figur 3: Bagasjebrett foran



Figur 4: Bilde av ATV uten fanger og bagasjebrett, samt med montert fanger og feste til bagasjebrett. Bagasjebrettet festes med skruer i festestengene.

2.2 Bak

Rækkefølge for feste av deler:

- 1: Figur 5
- 2: Figur 6
- 3: Figur 7

Monteringsselementer:

- Samtlige monteringsselementer som skruer, skiver og muttere er skrudd fast i de demonterte delene.



Figur 5: Fanger bak



Figur 6: Festestenger til bagasjebrett bak



Figur 7: Bagasjebrett bak



Figur 8: Bilde av ATV uten fanger og bagasjebrett, samt med montert fanger og festestenger til bagasjebrett. Festestengene skal festes i de fire ledige hullene i øvre bildet, bagasjebrettet festes videre i disse som vist i figur 9



Figur 9: Bilde av ATV med montert fanger og bagasjebrett.

3 Montering av deksel til gir

Monteringselementer:

- Skruer for feste sitter i erstattende deksel rundt gir, montert på ATV: 2 × skruer nr. 7



Figur 10: Originaldeksel rundt girspak



Figur 11: Originaldeksel rundt girspak med skruensvisning

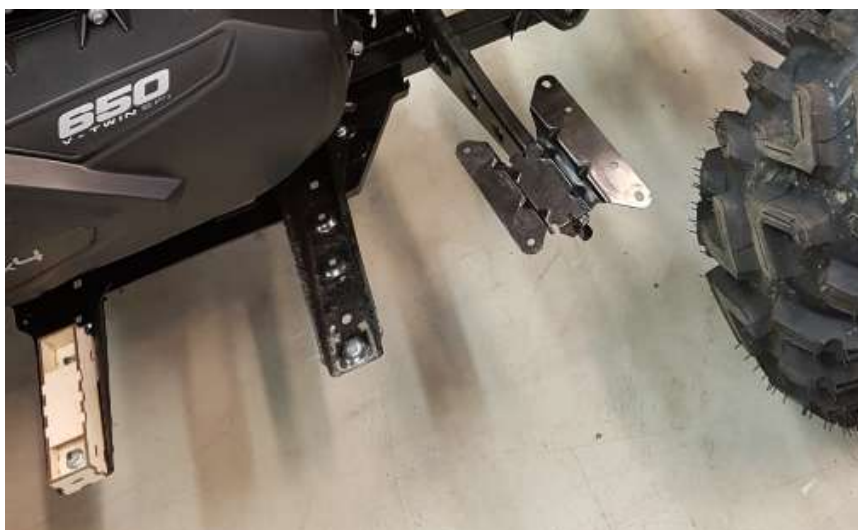
4 Montering av fothvilere og deksel

I denne seksjonen er bildeoversikt til feste av fothvilere og deksel på høyre og venstre side av ATV. Disse festes i samme rekkefølge som bildene kommer.

4.1 Feste til fothvilere



Figur 12: Festedel til fothvilere, festes under ATV



Figur 13: Bilde av hvor på ATV festedelen til fothvilerene skal festes. Denne settes inn i et hakk på undersiden av ATV.

4.2 Deksel høyre side

Rækkefølge for feste av deler:

- Figur 14
- Figur 15
- Figur 16
- Figur 17
- Figur 19

Monteringsselementer:

- 9 × skrue nr. 1
- 4 × mutter nr. 1
- 3 × skrue nr. 6
- 5 × plugg nr. 5



Figur 14: Feste til fothviler



Figur 15: Beskyttelsesdeksel til fothviler - festes sammen med feste til fothviler i figur 14



Figur 16: Beskyttelsesdeksel høyre side av ATV - Festes med plugg nr.5 og skrue nr. 6



Figur 17: Beskyttelsesdeksel høyre side av ATV



Figur 18: Deksel fothviler og side på ATV



Figur 19: Hjulbue - beskyttelsesdeksel over bakdekk med innfesting til baklyset



Figur 20: Bilde av ATV uten deksler og fothviler, samt med deksler og fothviler montert.

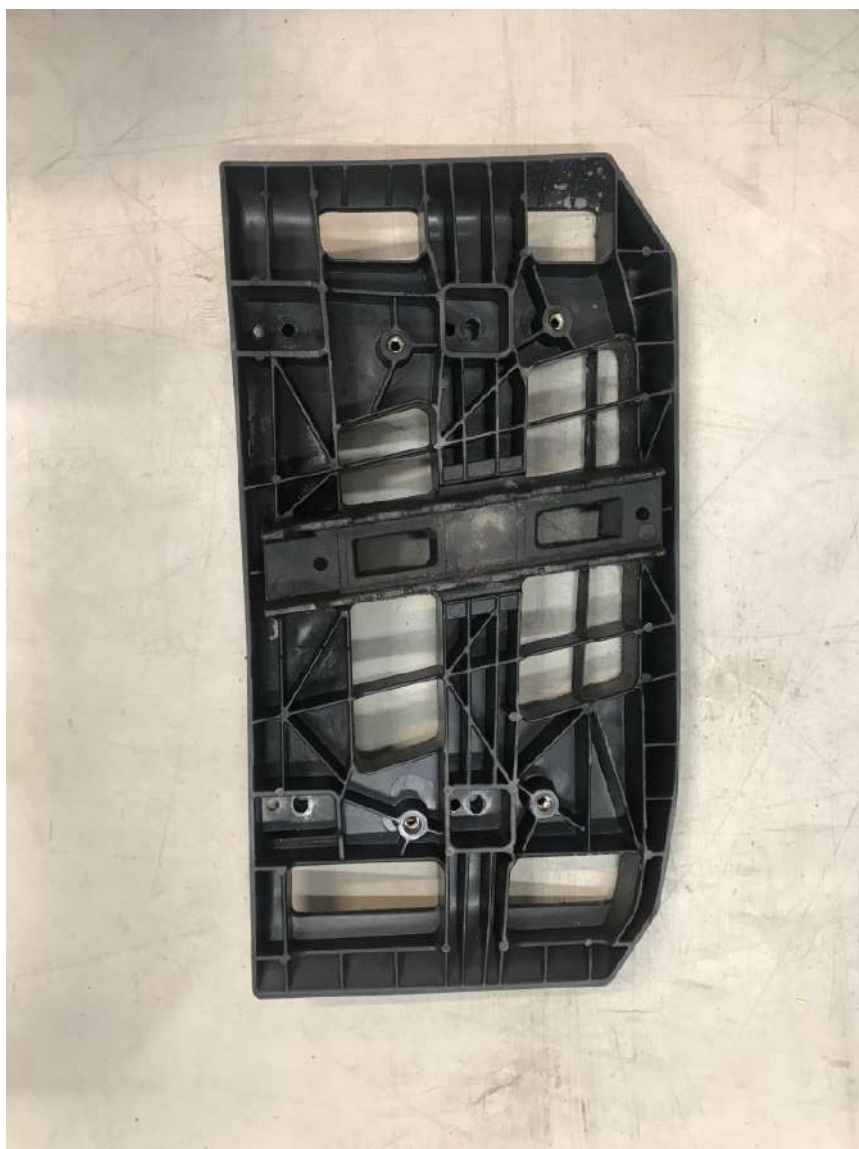
4.3 Deksel venstre side

Rækkefølge for feste av deler:

- Figur 21
- Figur 22
- Figur 23
- Figur 24
- Figur 25

Monteringsselementer:

- 9 × skrue nr. 1
- 4 × mutter nr. 1
- 3 × skrue nr. 6
- 5 × plugg nr. 5



Figur 21: Feste til fothviler



Figur 22: Beskyttelsesdeksel til fothviler - festes sammen med feste til fothviler i figur 21



Figur 23: Beskyttelsesdeksel venstre side av ATV - Festes med plugg nr. 5 og skrue nr. 6



Figur 24: Beskyttelsesdeksel venstre side av ATV



Figur 25: Hjulbue - beskyttelsesdeksel over bakdekk med innfesting til baklys



Figur 26: Bilde av ATV uten deksler og fothviler, samt med deksler og fothviler montert.

4.4 Festeanvisning - fothvilere og hjulbuer



Figur 27: Skruveoversikt for feste av fothvilere - festes med 9 × skrue nr. 1, hvorav 4 av disse skal ha mutter.



Figur 28: Deksel til fothviler festes sammen med beskyttelsesdeksel over både frem- og bakdekk med skrue nr. 6



Figur 29: Festes med 2 × plugg nr. 5 ved fremdekk



Figur 30: Beskyttelsesdeksel over bakdekk festes med plugg nr. 5 i frem- og bakkant

5 Montering av sete og festepunkter

Rekkefølge for feste av deler:

- Figur 31
- Figur 32
- Figur 33
- Figur 35

Monteringselementer:

- 6 × skrue nr. 2
- 2 × skrue nr. 3
- 4 × skrue nr. 5



Figur 31: Deksel mellom styre og fører sete, fremre del av fører sete festes i svart stang



Figur 32: Festedel til passasjersele, sete festes i stangen bak



Figur 33: Førersete, festes i stang som forklart under figur 31



Figur 34: Førersete og tilhørende festedel



Figur 35: Passasjersete, festes i stang som forklart under figur 32



Figur 36: Bilde av ATV med fremre festedel montert, men uten bakre festedel og sete montert



Figur 37: Skruveoversikt for feste av del i figur 32

6 Styret

Monteringselementer:

- Samtlige skruer, skiver og muttere er skrudd fast i de demonterte delene.



Figur 38: Beskyttelsesdeksel som skal festes i styret

7 Monteringselementer

Figur 39 til 46 viser bilde med nummerering av de ulike monteringselementene til de alle demonterte deler.

Figuroversikt:

- Figur 39 - Skrue og mutter nr. 1
- Figur 40 - Skrue og skive nr. 2
- Figur 41 - Skrue nr. 3
- Figur 42 - Skrue nr.4
- Figur 43 - Plugg nr. 5
- Figur 44 - Plugg nr. 5
- Figur 45 - Skrue og skive nr. 6
- Figur 46 - Skrue, skive og mutter nr.7



Figur 39: Skrue og mutter nr. 1



Figur 40: Skrue og skive nr. 2



Figur 41: Skrue nr. 3



Figur 42: Skrue nr. 4



Figur 43: Plugg nr. 5



Figur 44: Plugg nr. 5



Figur 45: Skruer og skiver nr. 6



Figur 46: Skruer, skiver og møtrikker nr. 7

A.6 Teknisk Håndbok



Guided Wolf

Teknisk håndbok

Guided Wolf



KONGSBERG

USN Universitetet
i Sørøst-Norge

Sammendrag

Denne tekniske håndboka er ment for arvtakerne av Guided Wolf prosjektet og skal gjøre det lettere å videreutvikle systemet.

Den endelige versjonen av systemet vil bli presentert slik det ble avsluttet. I tillegg vil alt av teknisk informasjon som er nødvendig for å forstå hvordan prosjektets delsystemer fungerer og hvordan de henger sammen.

Det vil også bli vist hvilke løsninger som ikke fungerte og litt av tankegangen rundt noen av beslutningene som ble tatt.



CAN-AM OUTLANDER XT 650 T

Innhold

1	Sammendrag	I
2	Komponenter og konfigurasjoner	1
2.1	Kretsskjema	2
2.2	Utstyrliste	4
2.3	Hardware	5
2.3.1	Aktuatorer: LA14, LA33	5
2.3.2	TR-EM-288-driver og EM-236A programmerer	6
2.4	Software	9
2.4.1	Processing	10
2.4.2	Processing-koden	10
2.4.3	Firmata protocol, StandardFirmata	12
2.4.4	Arduino	13
2.4.5	Feilsøking	13
2.5	Diverse kretser	15
2.5.1	PWM til DC	15
2.5.2	Strømsensorer	16
2.5.3	IR-sensorer, hastighetsmåling	16
2.6	Styring og MODBUS	16
2.7	Maskindeler	17
2.7.1	Oversikt over alle innkjøpte og produserte komponenter	17
3	Referanser	24
4	Vedlegg	25
A	Arduino: Standard Firmata	25
B	Processing: Game Control Plus, Configurator	25
C	Processing kode	25
D	Kretsskjema filer	25

Tabeller

1	Utstyrliste over de fleste delene tilhørende prosjektet	4
2	Oversikt over alle maskin-komponenter	17

Figurer

1	Kretsskjema til systemet	3
2	Tekniske spesifikasjoner for aktuatorene	5
3	TR-EM-288 driver	6
4	EM-236A programmerer	7
5	Standardinnstillinger på driver tr-em-288 for forskjellige aktuatorer	7
6	Pinne nr og hva de gjør	8
7	Tilkoblinger brukt i systemet.	9
8	Brukergrensesnitt	10
9	konfigurasjoner på skulderknappene.	12
10	knappkonfigurasjoner på front	12
11	Hvordan glattekretsen ble satt opp	15
12	PWM - DC prototype kretskort	16
13	2D tegninger for å vise dimensjoner på komponent nummer 1	17
14	Komponent nummer 1 slik den brukes	18
15	CAD modell - komponent 2	19
16	Interface mellom aktuatoren og komponent nummer 2	20
17	Designet av hele styret satt i en sammenstilling	21
18	Komponent nummer 4	21
19	Komponent nummer 5	22
20	Komponent nummer 4 og 5	22
21	Komponent nummer 6	23

Komponenter og konfigurasjoner

Denne delen viser alt av utstyr brukt under prosjektet og hvordan de er konfigurert.

Innhold:

- Kretsskjema
- Komplette utstyrliste
- Hardware
- Software
- Diverse kretser
- Maskindeler



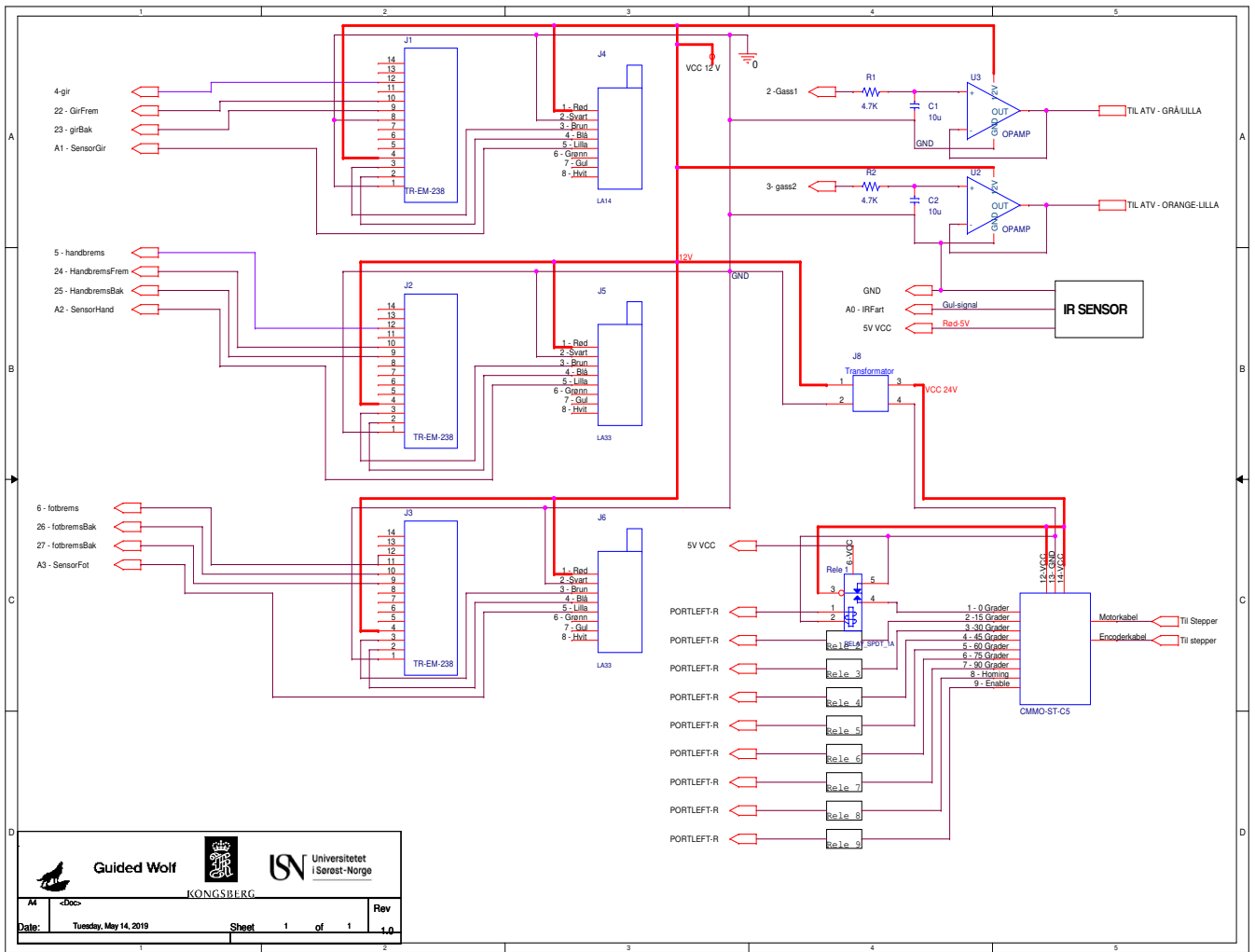
KONGSBERG

USN Universitetet
i Sørøst-Norge

2.1 Kretsskjema

Kretsskjemaet gir en oversikt over hvilke plugger som er brukt i Arduinoen. Her har det blitt henvist til både pinnennummer, og hva de heter i koden. Noen snarveier har blitt tatt ved tegningen, som ved releene hvor kun første tilkobling har blitt vist.

Kretsskjemaet har blitt satt opp i OrCad - Capture. Råfilen til dette ligger under vedlegg.



2.2 Utstysrliste

Tabell 1: Utstysrliste over de fleste delene tilhørende prosjektet

Utstysrliste		
Utstyr	Leverandør	Antall
HC-12 Wireless Serial Module Remote, 1000M With Antenna	Banggood.com	2
3pk DAC Breakout Development Board Module	Banggood.com	1
Arduino Mega 2560	Banggood.com	2
USB 2.0 Cable A Male to B Male Cord	Banggood.com	1
USB 3.0 4-Port Splitter Hub	Banggood.com	1
40pk Male to Male Color Breadboard Cable Jump Wire Jumper	Banggood.com	1
310 pk jumper header connectors	Banggood.com	1
Proximity sensor	Banggood.com	4
Aktuator LA14	Linak	1
Aktuator LA33	Linak	2
Strømbegrenser TR-EM-288	Linak	3
Programerer for strømbegrenser TR-EM-236a	Linak	1
Strømsensor	Ebay.co.uk	1
KOPLINGSKINNE BEP m/deksel sort/rød 2x16/2	Batteri og Radiatorservice	1
Motorkontroll CMMO-ST-C5-1-DIOP	Festo	1
Steppermotor EMMS-ST-87-L-SE-G2	Festo	1
Motorledning NEBM-S1G15-E-2.5-Q7-LE6	Festo	1
Encoderledning NEBM-M12G8-E-2.5-LE8	Festo	1
Styreledning NEBC-S1G25-K-3.2-N-LE25	Festo	1
W5100 Ethernet Shield Arduino	ebay.co.uk	1
RS485 TTL - Breadboard Module Arduino	ebay.co.uk	1
VICTRON Orion omformer 12V/24V-20	Batteri og Radiatorservice	1
Plastikkdeksel til gir	Kajander Motor AS	1
Tannhjul	Kjell Berg AS	2
Kjede	Kjell Berg AS	2
Gir 9:1 for steppermotor RELEX AG, CH-8645 JONA	Festo	1

2.3 Hardware

2.3.1 Aktuatorer: LA14, LA33

Hvordan aktuatorene er koblet opp kan sees i kretsskjemaet.

Spesifikasjonene for aktuatorene brukt kan ses i figur 2.

Aktuator LA14	Kraft: Maks.300 N Hastighet: 43 mm/s ubelastet Slaglengde: 130 mm Innbyggingsmål: 245 mm Farge: Svart Tetthet: IP66 Tilbakemelding: 0,5-4,5 V
Aktuator LA33	Kraft: Maks. 1500 N Hastighet: 34 mm/s ubelastet Slaglengde: 100 mm Innbyggingsmål: 260 mm Motor: 12 VDC , 10 Amp Tetthet: IP66 Interne endestoppbrytere Tilbakemelding: 0,5-4,5 V

Figur 2: Tekniske spesifikasjoner for aktuatorene

LA14

Databladet til aktuator LA14 som ble brukt til giringen kan sees her: [7].

Versjonen med de spesifikasjonene som ble bestilt er: <I/O specifications: Actuator with endstop signals and absolute positioning - Analogue feedback>, som man finner på side 16 i databladet.

LA33

Databladet til LA33 aktuatorene brukt til hand- og fotbrems kan sees her: [8].

Versjonen med de spesifikasjonene som ble bestilt er: <I/O specifications: Actuator with endstop signals and absolute positioning - Analogue feedback>, som man finner på side 18 i databladet.

2.3.2 TR-EM-288-driver og EM-236A programmerer

For alle lineæraktuatorene brukt under prosjektet er det brukt en TR-EM-288 driver for å kontrollere strøm inn til hver av aktuatorene. Konfigurert på forskjellige måter, noe som vil forklares i denne rapporten.



Figur 3: TR-EM-288 driver

Standardkonfigurasjon av driveren for alle aktuatorene kan ses i databladet for driveren [3].



Figur 4: EM-236A programmerer

Informasjon om programmereren kan ses her [2].

Parameter	LA12	LA23	LA28	LA30	LA31	LA32	LA34	LA35	LA36
1	0	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1
4	100	100	100	100	100	100	100	100	100
5	50	50	50	50	50	50	50	50	50
6	20	20	50	70	50	50	70	50	100
7	20	20	50	70	50	50	70	50	100
8	1	1	1	1	1	1	1	1	1
9	20	20	20	20	20	20	20	20	20
10	1	1	1	1	1	1	1	1	1
11	35	35	35	35	35	35	35	35	35
12	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0
15	10	10	10	10	10	10	10	10	10
16	10	10	10	10	10	10	10	10	10
17	0	0	0	0	0	0	0	0	0

Figur 5: Standardinnstillinger på driver tr-em-288 for forskjellige aktuatorer



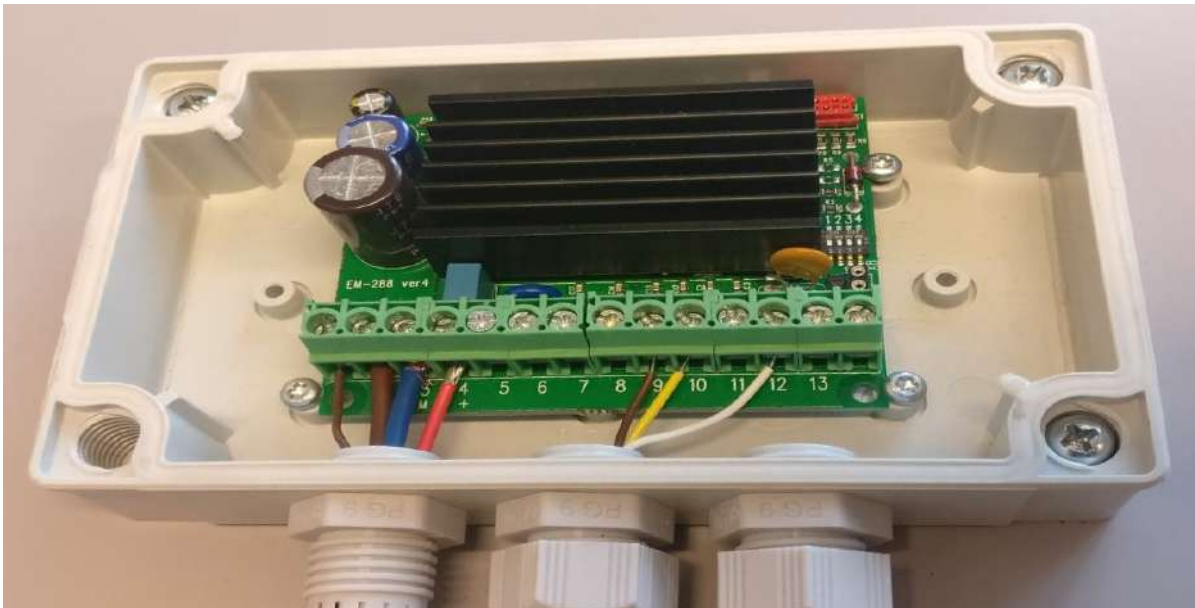
Figur 6: Pinne nr og hva de gjør

- kb1: 0 Vdc
- kb2: Motor positiv
- kb3: Motor negativ
- kb4: 12 Vdc
- kb5: VCC
- kb6: Fram - sikkerhetsnett(stopp)
- kb7: Tilbake - sikkerhetsnett(stopp)
- kb8: GND
- kb9: Styringspinne - fremover
- kb10: Styringspinne - bakover
- kb11: Stopp
- kb12: Speed-2
- kb13: Feilbryter
- kb14: 5 V ut.

Det ble brukt en spenningskilde på 12V DC inn på pinne 1 og 4 for å få tilført driverkontrolleren konstant spenning. Pinne 2 og 3 er koblet til moteren. Pinne 8 er jord.

Styringspinnene 9, 10 og 12 var koblet til Arduinoen for å regulere posisjonene for fremover, bakover og fart, respektivt.

Figur 7 viser hvilke tilkoblingsklemmer som ble brukt.



Figur 7: Tilkoblinger brukt i systemet.

Det finnes to konfigurasjoner for hastighet (speed-1, speed-2) hvor driveren opererer med speed-2 som default, dersom koblingen fra pinne 12 er sluttet. Hvis den er brutt, vil den bruke speed-1. Vi konfigurerer driveren og setter speed-2 til max ytelse, og speed-1 på null ytelse. Da kan det genereres et PWM signal med veksling mellom hastighetene for å få en regulering på farten.

Dersom pinne 9 (fremover) er høy og pinne 10 (bakover) er lav, vil aktuatoren gå fremover.

2.4 Software

For å kontrollere alt av hardware blir en Arduino brukt, hvor StandardFirmata er lastet opp. Processing blir så brukt for å få laget et nyttig og estetisk pent brukergrensesnitt.

Hvordan disse fungerer og samhandler kan sees i underkapitlene.

2.4.1 Processing

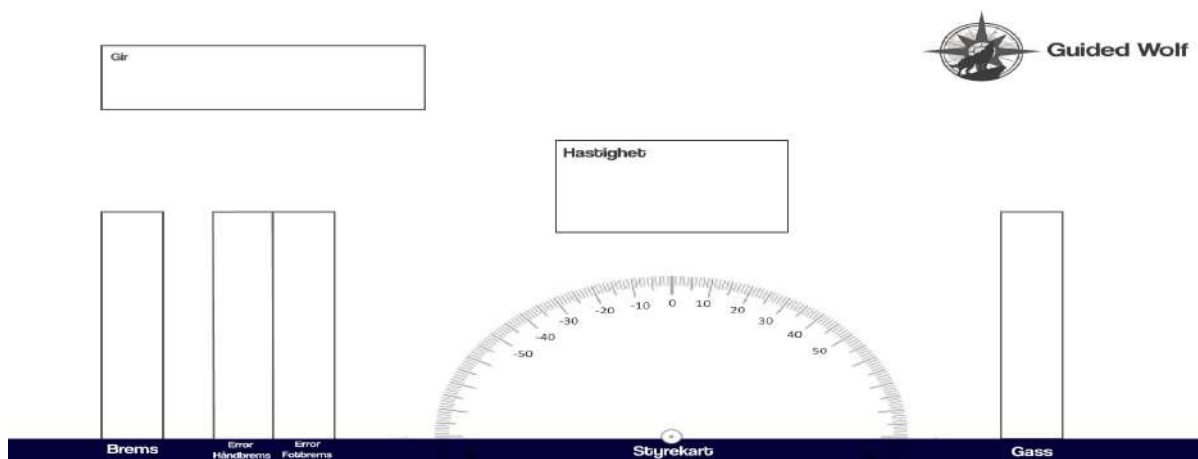
Processing er hovedsakelig et programmeringspråk ment for grafiske og visuelle fremstillinger, basert på Java [10].

2.4.2 Processing-koden

Processing-koden kan finnes som vedlegg: D.

Brukergrensesnitt

For at operatøren skal ha oversikt over hvordan kontrolleren oppfører seg og for å få tilbakemelding på hastighet og omgivelser er det laget et panel som operatøren kan se. Figur 8 viser hvordan dette ser ut. I tillegg er det klargjort i koden å bruke kameraer og IR-sensor som en mulighet til å overvåke hastigheten.



Figur 8: Brukergrensesnitt

I filene lagt ved ligger alle bildene brukt for brukergrensesnittet. I Processing-koden, i funksjonen "brukergrensesnitt" ser man hvordan disse brukes.

For at man skal kunne bruke Processing koden vedlagt er man avhengig av at disse bibliotekene er lastet ned og brukes:

- **Game Control Plus** - Gir deg lov til å bruke spillkonsoll sammen med kode i Processing.
- **Arduino Firmata** - Kommunikasjon med Arduinoen.
- **G4P** - Nødvendig for å kunne bruke Game Control Plus - Configurator.
- **Serial** - Leser og skriver data til og fra eksterne objekter. Arduinoen trenger dette biblioteket for at den skal fungere.
- **Control P5** - Tilbyr en rekke kontrollere som enkelt lar deg endre og justere verdier mens programmet kjører.
- **video** - For å bruke kamera i Processing.

Configurator - spillkontroller For å konfigurere spillkontrolleren i Processing, må den kobles opp mot PC, før man deretter åpner eksempelet "Configurator" under "Game Control Plus" (se vedlegg B for kode). Programmet er klart for å kjøres, slik at man enkelt kan velge hvilke knapper man vil konfigurere. Her har det blitt gitt navn i forhold til hva knappene heter på den faktiske kontrolleren.

I filene lagt ved sammen med Processing-koden, vil følgende konfigurasjoner være lagt inn:

Skulderknapper:

R2 (Analog): Gass med variabelt pådrag i èn akse

L2 (Analog): Brems med variabelt pådrag i èn akse.

R1 (Digital): Skifte gir i en retning med et tastetrykk.

L1 (Digital): Skifte gir i en retning med et tastetrykk.

Frontknapper:

X (Digital): Nødstopp med et tastetrykk

trekant (Digital): For å gå ut av oppstartsmodus

L3 (Analog): Styring i èn akse.



Figur 9: konfigurasjoner på skulderknappene.



Figur 10: knappekonfigurasjoner på front

2.4.3 Firmata protocol, StandardFirmata

Firmata protocol

Firmata protocol muliggjør kommunikasjon mellom en vertdatamaskin og Arduinoen [4].

StandardFirmata

For å kunne kommunisere mellom PC-en (Processing) og Arduinoen.

For mer informasjon om hvordan dette brukes: [6].

2.4.4 Arduino

Arduinoen blir brukt for å videresende kommandoer og lese av sensordata.

Ved å laste ned Firmata-biblioteket til "Arduino IDE" og ved å bruke eksempelet: "StandardFirmata" å laste den opp til Arduinoen er Arduinoen ferdig konfigurert så lenge Baud-raten stemmer overens med den brukt i Processing-koden.

2.4.5 Feilsøking

Her vil diverse problemer gruppen møtte underveis i prosjektet bli presentert, slik at det muligens blir lettere for arvtakerne og unngå å bruke like mye tid på de samme feilene.

Baud-rate

For at serial-kommunikasjonen mellom PC-en brukt og Arduinoen skal fungere må de operere med samme Baud-rate.

Den kommer til å være 57600 i kodene vedlagt.

Nedenfor ser man hvordan dette ser ut i Arduino og i Processing.

Arduino:

```
Firmata.begin(57600);  
  while (!Serial) {  
    ; // wait for serial port to connect. Needed for ATmega32u4-based  
    ↪ boards and Arduino 101  
  }
```

Processing:

```
println(Arduino.list());  
arduino = new Arduino(this, "COM10", 57600);
```

Tilfeldige verdier på innganger

Under testing ble det oppdaget at Processing tar imot forskjellige verdier også på de pinnene som ikke er i bruk. Dersom pinne A0 er koblet til ett potmeter, vil også A1 motta verdier som blir påvirket av dette. Dette gjør det ekstra viktig å dobbeltsjekke pinnetilkoblingene, da man kan få ut verdier som kan virke riktige ved en feilkobling [5].

Analog/Digital - Pin i Processing

Ett problem som ble støtt på tidlig i utviklingen var at Processing ikke håndterer pinnenummering på samme måte som Arduino. I Arduino skriver man slik for å deklarerer pinnennummer:

```
pinMode(A0, INPUT); //For å deklarerer pinne A0 som input
```

Denne måten å skrive på, byr på problemer for Processing da den forventer et heltall i denne deklarereringen og vil derfor ikke tillate A0. Det ble derfor sett på alternative måter å løse dette på. Dersom man ser på Atmega2560 pinmapping fra [9] kan man se at alle pinnene har et nummer. Det ble forsøkt å bruke disse nummerene til å deklarerer uten hell. Etter dette ble det sett på alternative måter å sende informasjon fra Arduinoen til Processing på. En av disse er å bruke Serial.println funksjonen i Arduino som sender over informasjon via serial-porten. Et eksempel på dette viste electronics hobbyist [1]. Problemet med dette eksempelet er at det blir veldig begrenset og kan kun bruke en analog input. Derfor er ikke dette egnet til dette prosjektet.

Instructables [11] viser hvordan dette kan gjøres. Dette eksempelet ble funnet tidlig, men ble oversett da dette ikke var logisk i forhold til hvordan modellen til presentasjonen ble kodet opp. Det viste seg senere at dette var måten det skal gjøres på. I Processing skal man skrive slik for å deklarerer pinnennummer:

```
pinMode(0, INPUT); //For å deklarerer pinne A0 som input  
pinMode(0, INPUT); //For å deklarerer pinne 0 som input
```

Grunnen til at dette ble avskrevet er at man kan skrive på samme måte for å deklarerer pinnennummer 0 og A0. Processing velger hvilket pinnennummer som skal brukes utifra om man skriver analogRead eller digitalRead. Dersom man bruker analog vil det si A0, mens digital vil si 0. Dette vil være viktig å huske på under videre koding av prosjektet for å unngå feil.

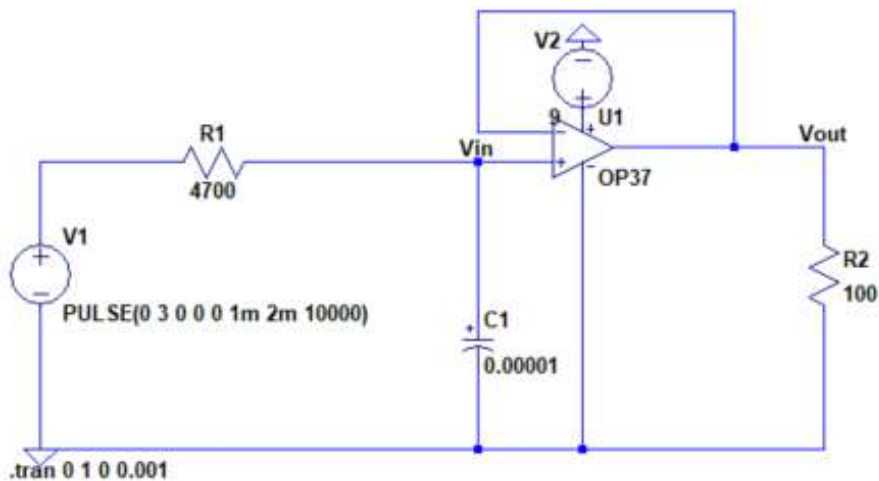
Com-porter Dersom det har blitt koblet til flere com-porter på datamaskinen i bruk, kan disse blandes i Processing. Det blir deklart at Processing skal bruke den første com-porten. Dersom dette ikke er tilfelle må det endres i koden.

2.5 Diverse kretser

Her vil det bli forklart hvilke eksterne kretser som ble brukt og hvilke formål de hadde.

2.5.1 PWM til DC

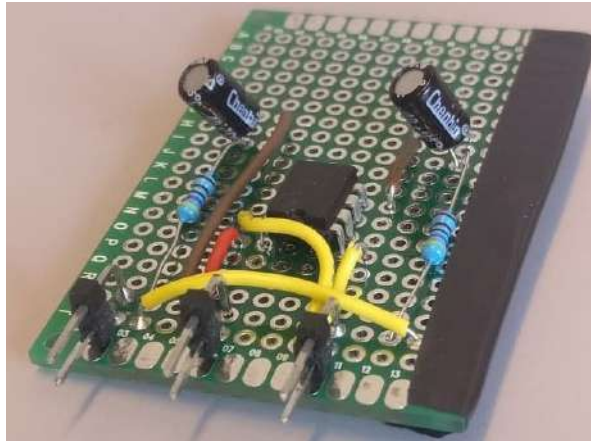
Formålet med denne kretsen var å gjøre om PWM signalene sendt fra Arduinoen om til DC inn til gassensorene på ATV-en. Figur 11 viser oppsettet på kretsen.



Figur 11: Hvordan glattekretsen ble satt opp

Ved implementeringen av kretsen i testen hvor PS4-kontrolleren ble brukt til å gasse ble det funnet ut at delsystemet fungerer i praksis, test (TI.27-GA.01).

Kretsen over har blitt loddet på et prototype kretskort i figur 12. De to inngangene er til venstre, utgangene til høyre. VCC på 12 V er den øverste pinnen i midten, og den nedre er jord.



Figur 12: PWM - DC prototype kretskort

2.5.2 Strømsensorer

Det ble gjort forsøk med å bruke strømsensorer for å kunne stoppe aktuatorene fra å kjøre ved for høyt strømforbruk og derfor utøve for mye krefter på bremse- og girsystemet.

Dette ble ikke implementert i tidsperioden gruppen hadde, men det kan anbefales å implementere dette eller noe liknende med samme funksjon for å værne de aktuelle delsystemene.

Denne testen ble dokumentert som test (TP.32-BR.01) i hovedrapporten.

2.5.3 IR-sensorer, hastighetsmåling

Det ble undersøkt og skrevet kode for bruken av IR-sensorer for å måle hastigheten til ATV-en. Teorien og koden for dette ligger under delkapittel (13.9) i hovedrapporten.

Selv om denne metoden å måle hastighet på fører til en del feilmålinger og vil gi litt upresise målinger anbefales det videre at et liknende oppsett eller noe med samme funksjon lages for å kunne gi tilbakemeldinger på hastighet, for operatøren å se og for tilbakekoblinger.

2.6 Styring og MODBUS

For informasjon om MODBUS se underkapittel (9.8) i hovedrapporten.

Testene i delkapittel (14.4) i hovedrapporten viser hva som funket og ikke funket under testingen av styreaktoren, giret og motorkontrolleren.

2.7 Maskindeler

2.7.1 Oversikt over alle innkjøpte og produserte komponenter

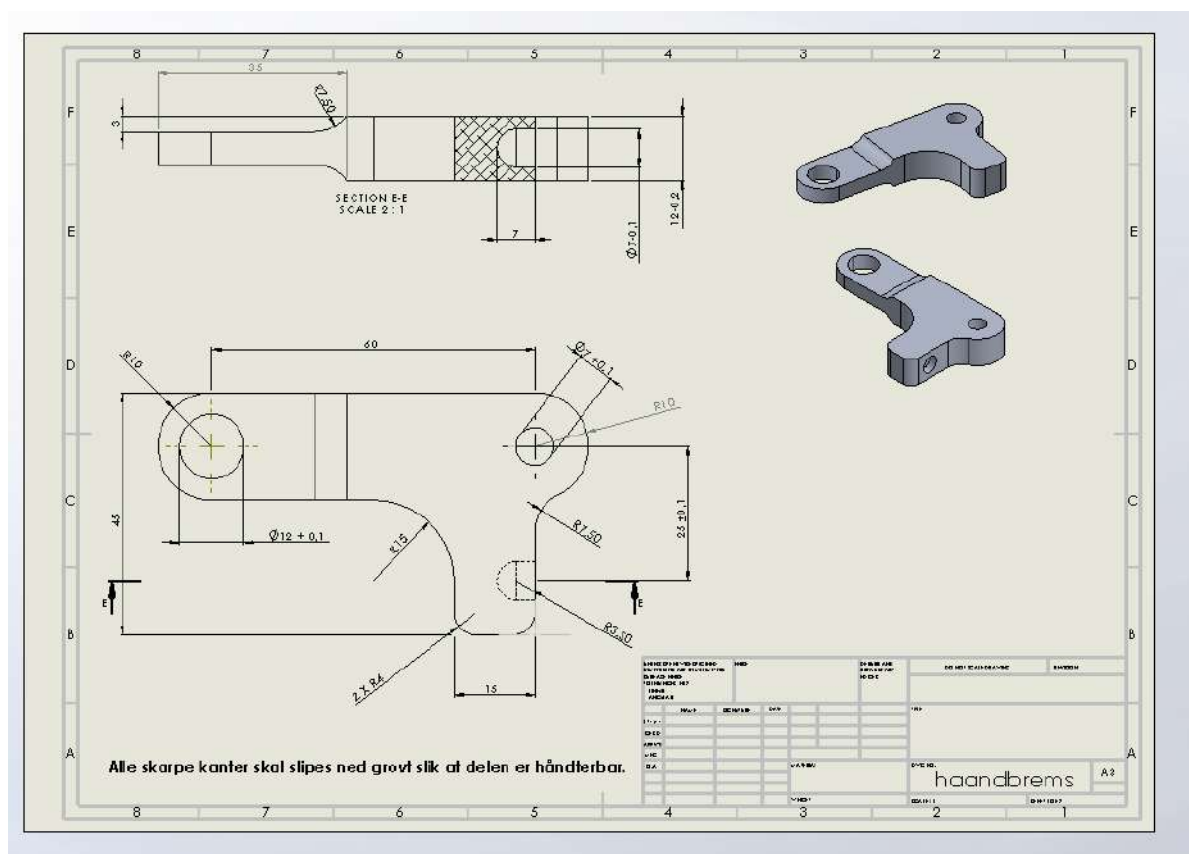
Tabell 2: Oversikt over alle maskin-komponenter

Komponent nummer:	Hva:	Hvor:	Innkjøpt/Produsert:	Funksjon:
Nr. 1	Komponent	Håndbrems	Produsert	For å trykke inn håndbremsstempet
Nr. 2	Komponent	Fotbrems	Produsert	For å trykke inn Fotbremsstempet
Nr. 3	2 X Rådreiv/Tannhjul og et kjede	Styring	Innkjøpt	For å rotere styret
Nr. 4	Plate	Styring	Produsert	Festes på styreaktuatoren, stasjonær
Nr. 5	Kobling	Styring	Produsert	Festes på styreaktuatoren, bevegelig
Nr. 6	Ramme	På midten av ATV-en	Produsert	For å feste alle aktuatorene og diverse andre komponenter på

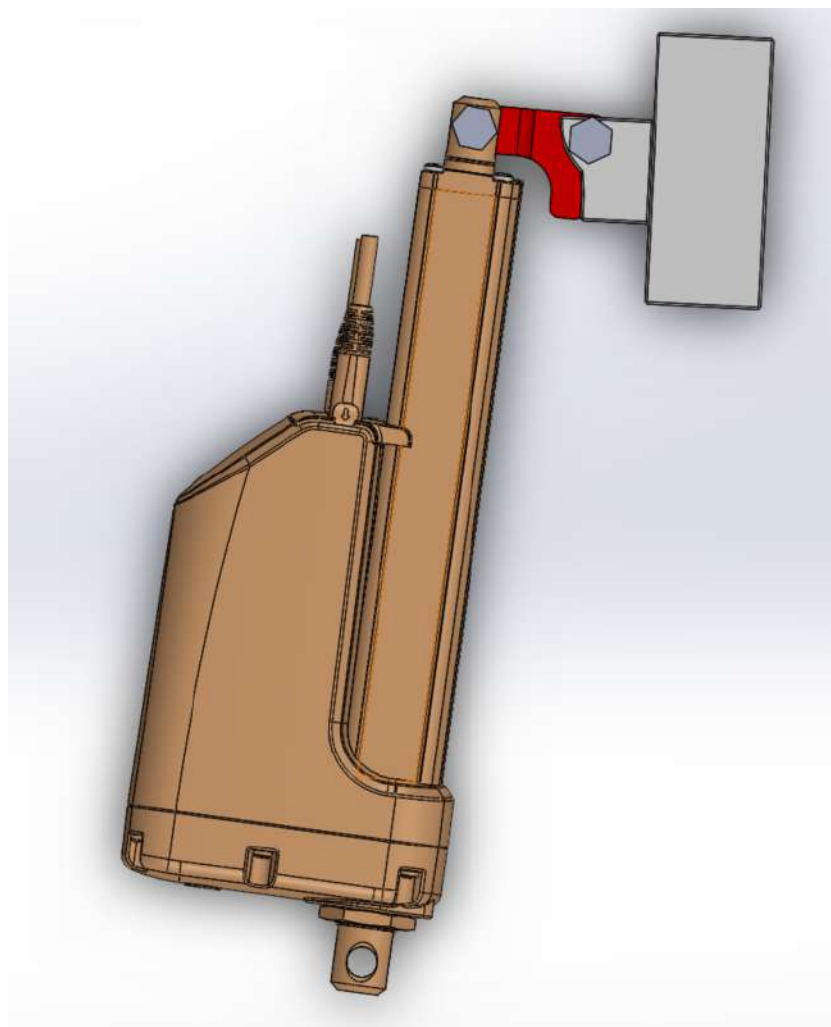
Bilder av komponentene

Her vises bilder og figurer av komponentene listet opp ovenfor:

Komponent nummer 1

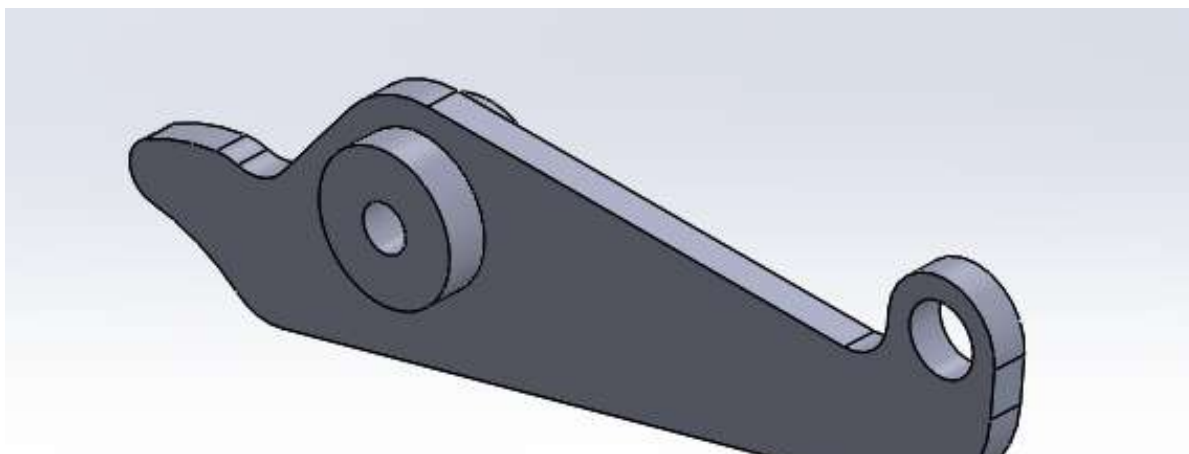


Figur 13: 2D tegninger for å vise dimensjoner på komponent nummer 1

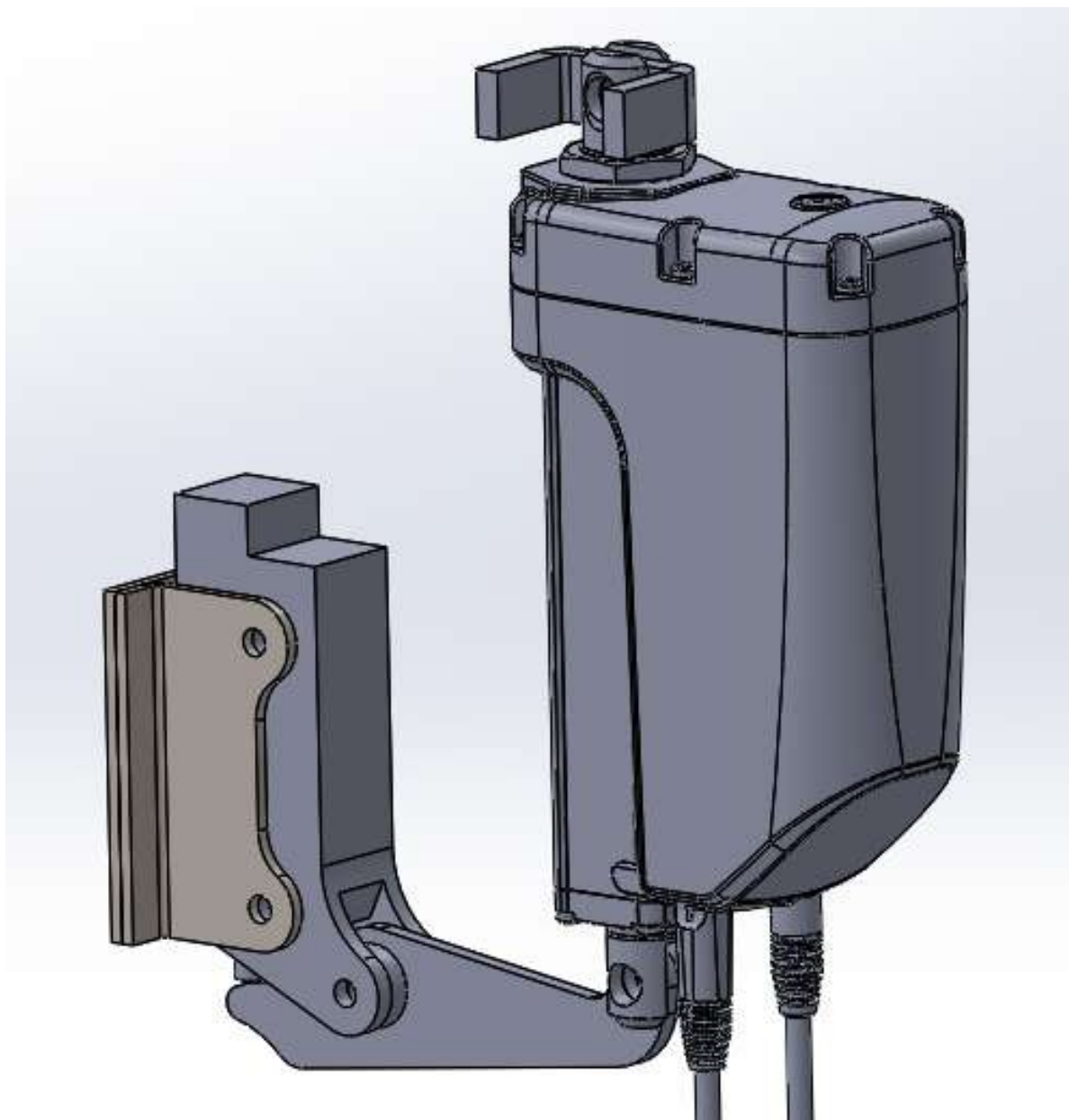


Figur 14: Komponent nummer 1 slik den brukes

Komponent nummer 2



Figur 15: CAD modell - komponent 2



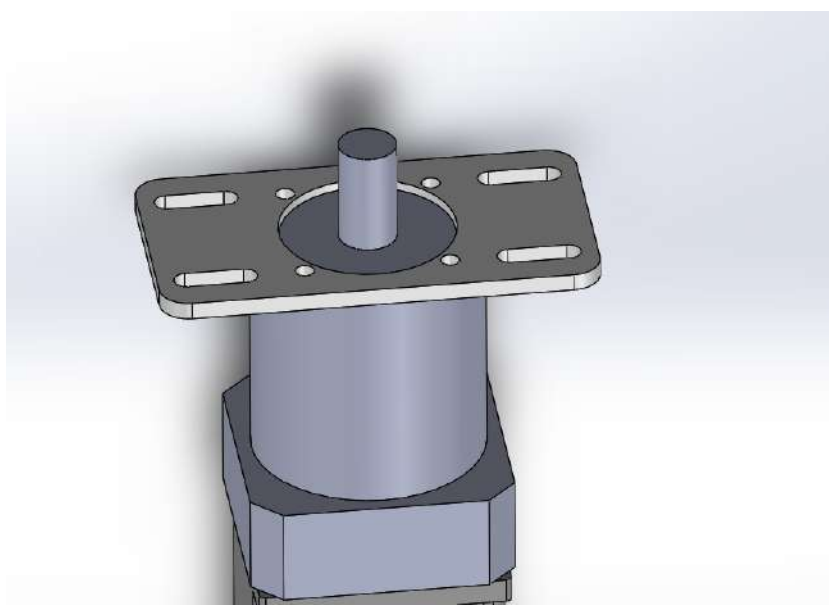
Figur 16: Interface mellom aktuatoren og komponent nummer 2

Komponent nummer 3 Bildet viser hele styresystemet sammensatt, inkludert rådrene og kjedet.



Figur 17: Designet av hele styret satt i en sammenstilling

Komponent nummer 4



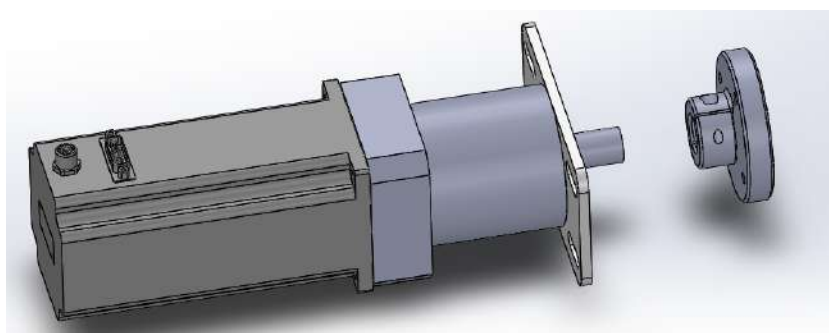
Figur 18: Komponent nummer 4

Komponent nummer 5



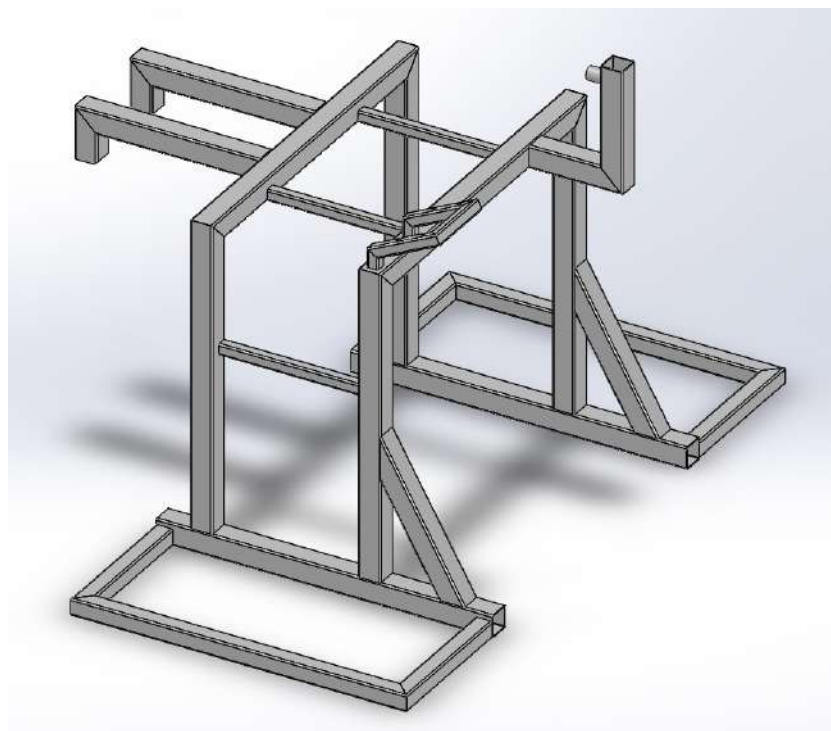
Figur 19: Komponent nummer 5

Komponent nummer 4 og 5 sammensatt



Figur 20: Komponent nummer 4 og 5

Komponent 6



Figur 21: Komponent nummer 6

Referanser

- [1] Arduino Serial til Processing. <https://electronics hobbyists.com/arduino-processing-tutorial-connect-arduino-to-processing/>.
- [2] Datablad og informasjon om EM-236A. <http://www.linakthirdparty.com/products/accessories.aspx?product=TR-EM-236A++Programming+unit>.
- [3] Datablad og informasjon om TR-EM-288. <http://www.linakthirdparty.com/products/drivers.aspx?product=TR-EM-288-SPF++Positioning+driver>.
- [4] Firmata protocol. <https://www.arduino.cc/en/Reference/Firmata>.
- [5] Forum post om feil i Standardfirmata. <https://github.com/firmata/arduino/issues/334>.
- [6] Hvordan bruke Arduino og Processing. <https://playground.arduino.cc/Interfacing/Processing/>.
- [7] LA14 Datablad. <https://cdn.linak.com/-/media/files/data-sheet-source/en/linear-actuator-la14-data-sheet-eng.ashx>.
- [8] LA33 Datablad. <https://ipaper.ipapercms.dk/Linak/ENGLISH/DATASHEET/linear-actuatorla33data-sheeteng/#/>.
- [9] Pinmapping til Atmega2560. <https://www.arduino.cc/en/Hacking/PinMapping2560>.
- [10] Processing, hjemmeside. <https://processing.org/>.
- [11] Read Analog Data Directly in Processing. <https://www.instructables.com/id/Read-analog-data-directly-in-Processing/>.

Vedlegg

Arduino: Standard Firmata

https://docs.google.com/document/d/1hdLA5l5r08bCM0vR8o8V07YlftY0wQZ7PboxyH_iiL8/edit

Processing: Game Control Plus, Configurator

https://docs.google.com/document/d/1oTDbN_WmTniK0tFVheYnFC362g7--GXXxDVF1F-qWi4/edit

Processing kode

<https://drive.google.com/drive/folders/1AiXz1LBdIU00xBFWilqrlQ4jxEUnx0c?usp=sharing>

Kretsskjema filer

<https://drive.google.com/drive/folders/1AiXz1LBdIU00xBFWilqrlQ4jxEUnx0c?usp=sharing>

A.7 Kraftmålinger på ATV



Guided Wolf

Kraftmålinger på ATV
CAN-AM OUTLANDER XT 650 T



KONGSBERG

USN Universitetet
i Sørøst-Norge

Innhold

1	Introduksjon	2
2	Resultater - kraftmålinger	3
3	Status på delsystemene	3
4	Utstyr	4
5	Generell Utførelse av forsøk	5
5.1	Håndbrems og fotbrems	5
5.2	Kommentar	8
5.2.1	Resultat	8
5.2.2	Feilkilder	8
5.3	Gir	9
5.3.1	Kommentar	13
5.3.2	Resultat	13
5.3.3	Feilkilder	13
5.4	Styret	14
5.4.1	Resultat	16
5.5	Nødstop	16
5.5.1	Resultat	16
5.5.2	Feilkilder	16
5.6	Gass	16
5.6.1	Resultat	16
5.7	Feste til våpenstasjon	16

1 Introduksjon

Dette forsøket hadde til hensikt å finne ut av hva slags krefter som trengs for å aktivere komponenter til delsystemene. På denne måten kan man se på hva slags aktuatorer som kreves i form av slaglengde, kraft og hastighet. En annen hensikt er å se på hva slags andre løsninger som kan brukes istedenfor aktuatorer.

2 Resultater - kraftmålinger

Tabell 1 viser hva slags krefter som trengs for å drive systemene ved gitte konseptløsninger:

Tabell 1: Oppsummerte resultater av krefter

Håndbrems aktuator på stempel	1472 N
Håndbrems med arm	230 N med 160 mm arm / 36.8 Nm
Fotbrems med servo	78,75 Nm
Fotbrems med arm	315 N med 250 mm arm / 78.75 Nm
Gir med lineær aktuator på girspaken	70 - 90 N, maks 100 N
Gir med servo	8.978 Nm
Styring med arm	175 N med 410 mm arm, 71.75 Nm
Styring med servo	71.75 Nm

3 Status på delsystemene

Tabell 2 viser en oversikt over status på kraftmålinger for hvert enkelt delsystem.

Tabell 2: Status på delsystemene

Håndbrems og fotbrems:	Status:	Fullført
Gir:	Status:	Fullført
Styre:	Status:	Fullført
Nødstop:	Status:	Avsluttet/overgitt til elektro
Gass:	Status:	Avsluttet/overgitt til elektro
Våpenfeste:	Status:	Pågående
Batteri:	Status:	Ikke påbegynt
Velt:	Status:	Ikke påbegynt

4 Utstyr

Figur 1 viser måleinstrumentene som ble brukt til kraftmålingene.



Figur 1: Måleinstrumenter brukt til kraftmålingene

5 Generell Utførelse av forsøk

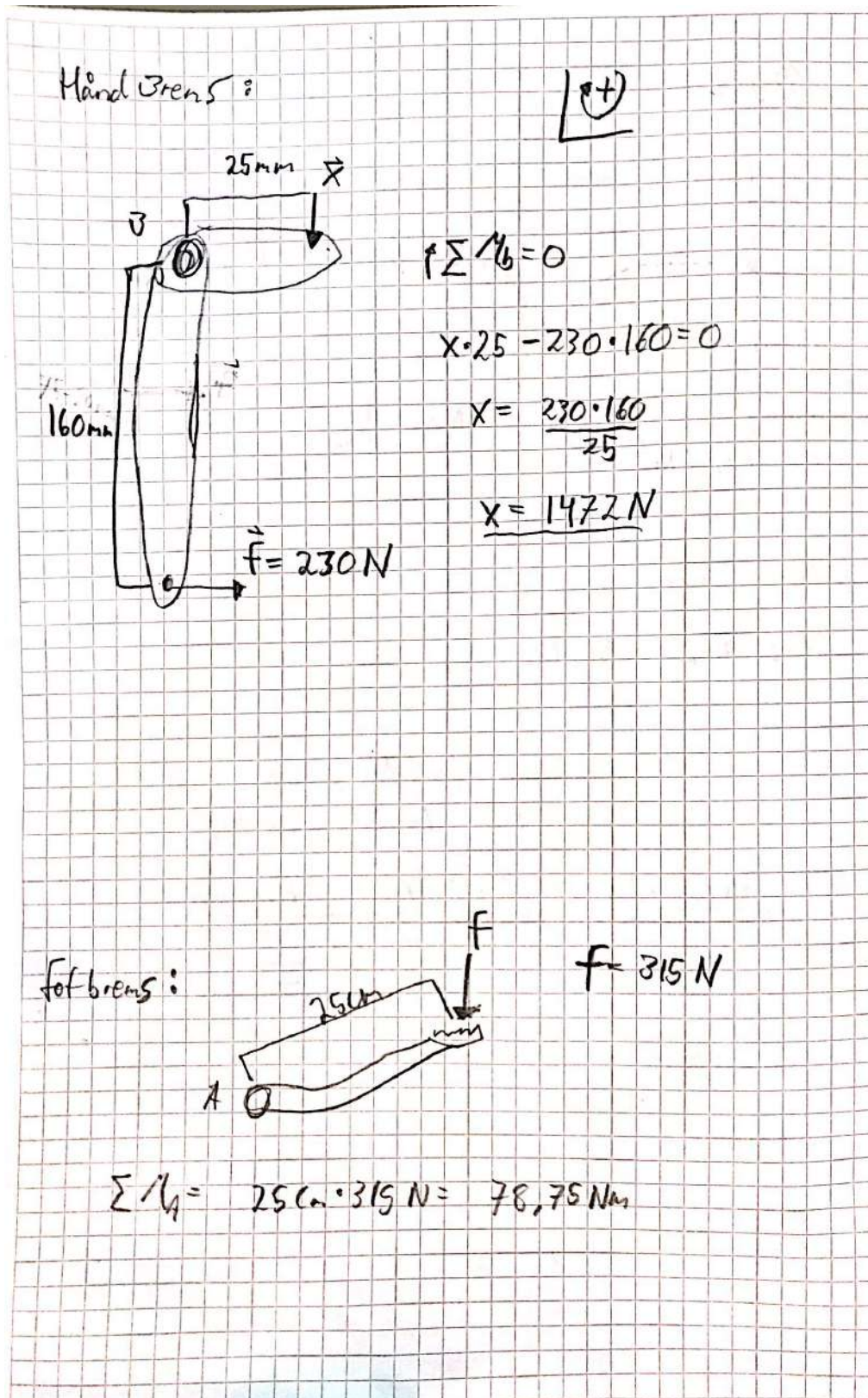
- Sett en bagasjevekt der hvor man ønsker en ekstern kraft skal trykke. (eks. aktuatore)
- Mål opp avstand fra bagasjevekten til hengslet for å finne armlengden på momentet som skapes.
- Trykk inn bremsen så mye som man ønsker at systemet skal klare.
- Les av data
- Regn ut normalkrefter/moment.

5.1 Håndbrems og fotbrems

Det ble utført forsøk hvor man testet hvor mye kraft som trengs ved å hekte bagasjevekten på fotbremsen og trykke ned.



Figur 2: Kraftmålinger håndbrems og fotbrems



Figur 3: Beregninger basert på kraftmålingene av hånd- og fotbrems

5.2 Kommentar

Bremsene ble trykket inn til kraften man ser for seg et menneske kan klare ble oppnådd. Det er da ingen vandring i bremsehendelen eller fotbremsen, slik at man kan anta at kreftene kan representeres som en statisk struktur. Ved bruk av statikk beregnes reaksjonskreftene i bremsestempelen på håndbrems (kalt X i 3). På fotbrems ble det beregnet resultantmoment rundt punkt A.

5.2.1 Resultat

Håndbrems: Enten en lineær kraft på 1472 N direkte på stempel, eller 230 N på den 160 mm armen som allerede eksisterer som tilsvarer 36,8 Nm.

Fotbrems: Krever 78,75Nm for å trykke inn bremsen, som er hva som antas å være maksimalt av hva et voksent menneske hadde tråkket ned.

5.2.2 Feilkilder

Grepet på bagasjevekten når håndbremsen ble målt skle litt opp og ned og trakk ikke hele tiden på 160 mm ut fra hengslet, men opptil 160 ± 5 mm.

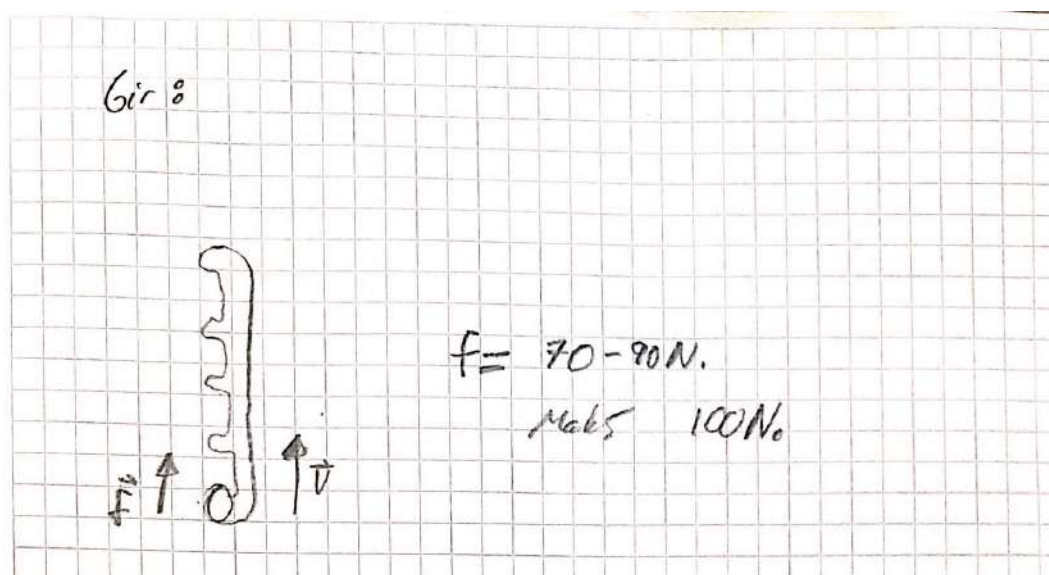
Det ble først trykket inn manuelt med hånden til ønsket kraft var oppnådd, deretter målt hvor store krefter som kreves for å bremse hardt nok til å sette på parkeringsbremsen.

5.3 Gir

Det er relevant å finne ut hvor mye krefter som kreves for å dra girspaken med en lineær aktuator eller hvilke krefter som trengs for å rotere giret med en eventuell servo.



Figur 4: Kraftmåling av gir



Figur 5: Beregninger basert på kraftmålingene av gir



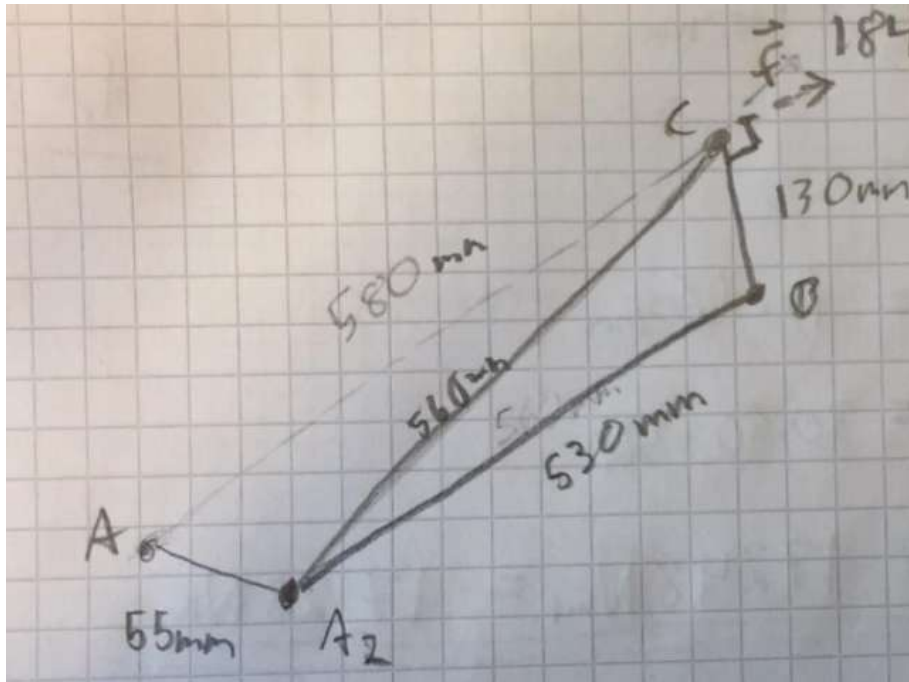
Figur 6: Utforming av girsystemet

Det er ut ifra dette målt opp til at det er 4 punkter som kan roteres. Figur 8 viser punktene A, A_2 , B og C som er blitt brukt for å finne dimensjoner og kraftoverføringer gjennom hver ledd.

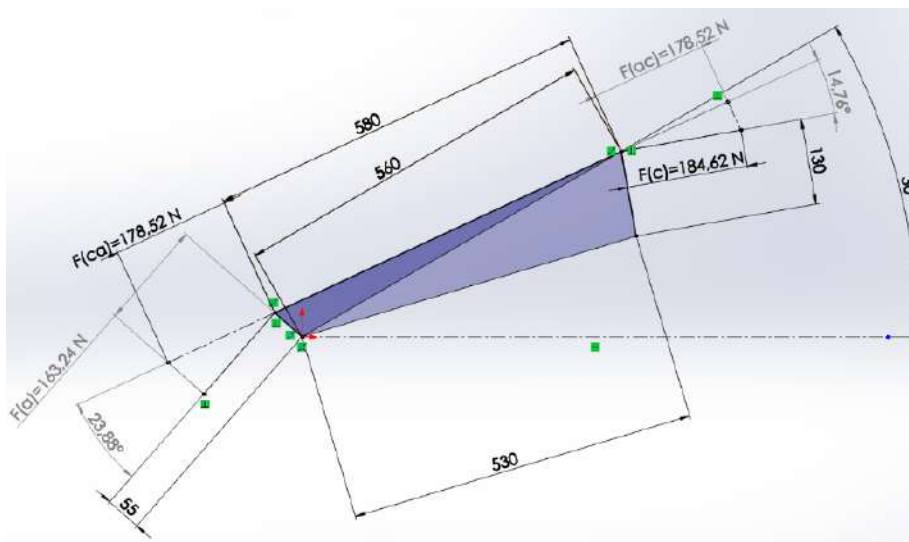


Figur 7: Punkter - dimensjoner og kraftoverføringer

For å finne krefter, så trengtes det vinkel-forholdene mellom de ulike leddene. Det var ikke vinkelmåler tilgjengelig og det antas å ha vært veldig unøyaktig ved måling. Det som ble gjort var å danne to trekkanter som deler hypotenus slik som vist i figur 8, dette gir en representasjon av girsystemet. Her er linjen AC antatt å være en rett strek for å finne vinklene mellom linjene og kraften som er vinkelrett på linje BC som vist i tegningen.



Figur 8: Beregninger basert på kraftmålingene av gir



Figur 9: Beregninger fra Solidworks

For å regne ut geometrien og kreftene ble Solidworks brukt (figur 9). Med armlengdene, vinkler og krefter så ble momentet regnet ut for hånd.

Dette førte til en enklere modell av det som er av relevant informasjon for å finne momentet om A_2 . Under er kraften F gitt verdien 100 N, men kan tillates å være en del svakere. Kraften som står vinkelrett på stang AA_2 ved punkt A er på 163,24 N (som vist i figur 9). Lengden fra punktet er 55mm, momentet om A_2 blir da:

$$A_2 = F(a) \times 55 \text{ mm} = 8.978 \text{ Nm} \quad (1)$$

5.3.1 Kommentar

Brukte bagasjevekt til å dra girspaken med ønsket kraft. Merket at vanlig drakraft var på 70 N og kanskje litt mer, men om girspaken hang seg opp, så kunne det komme krefter opp mot 100 N. Mer enn det er ikke ønskelig å dra med, da dette kan føre til at noen av komponenter fra ATV-en blir ødelagt i prosessen.

5.3.2 Resultat

- Med en lineær aktuator ved girspaken er det ikke ønskelig å drifte mer enn 70-90 N, og aldri mer enn 100 N.
- Ved bruk av servo rundt punktet A_2 , vil det være behov for 8,978 Nm. Dette er tilsvarende om det benyttes 100 N i girspaken, hvilket er helt i ytterkant av skalaen.

5.3.3 Feilkilder

- Kan være målefeil på dimensjoner
- Datainput på Solidworks kan være feil
- Usikkerhet på hvor mye trekraft man skal drifte og/eller designe med da den en-aksede testen opererte hovedsakelig i en akse, men hoppet litt i to dimensjoner ved svinging på omlag 5-10 mm.

5.4 Styret



Figur 10: Kraftmålinger styret

5.4.1 Resultat

Styret trenger 71,75 Nm for å snu stillestående uten ytre obstruksjoner som stein/gjørme etc.

5.5 Nødstop

5.5.1 Resultat

Ønsker å se om elektrostudentene klarer å se på ledningsnettverket, samt om de klarer å koble seg på der for å regulere systemet.

5.5.2 Feilkilder

- Unøyaktige målinger
- Bagasjevektens grep på styret når det ble målt skle litt opp og ned med feilmargin på anslagsvis ± 5 mm.
- Forholdene når motoren var av, altså ATV-ens servo ikke var på, og det var testet uten eksterne krefter slik som stein, gjørme og grus som realistisk sett kan forekomme. Da er nødvendig kraft større enn målt nå. Verdt å merke seg at servoen ikke hjalp veldig mye mer når motoren til ATV-en var i gang, men kjøretøyet var i stillestående stand.

5.6 Gass

Det finnes en thumb throttle for å kunne regulere gassen. Det kunne bli gjort styrkeberegninger på denne, men det er en bedre løsning å se om det går an å koble seg direkte inn på det elektriske nettet til ATV-en. På bakgrunn av dette, investeres det ikke mer tid foreløpig for å finne ut hvor mye krefter det skal til for å trykke inn bryteren, men heller se om det går an å regulere strømmen direkte.

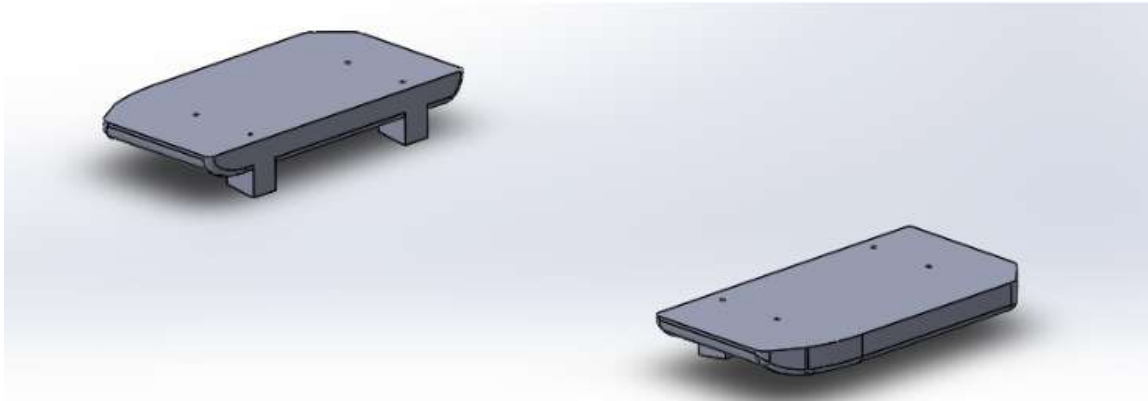
5.6.1 Resultat

Ønsker å se om det går an å koble seg inn på el-systemet for å regulere strømmen for å styre gassen. To 5 V sensorer som styrer gasspådraget.

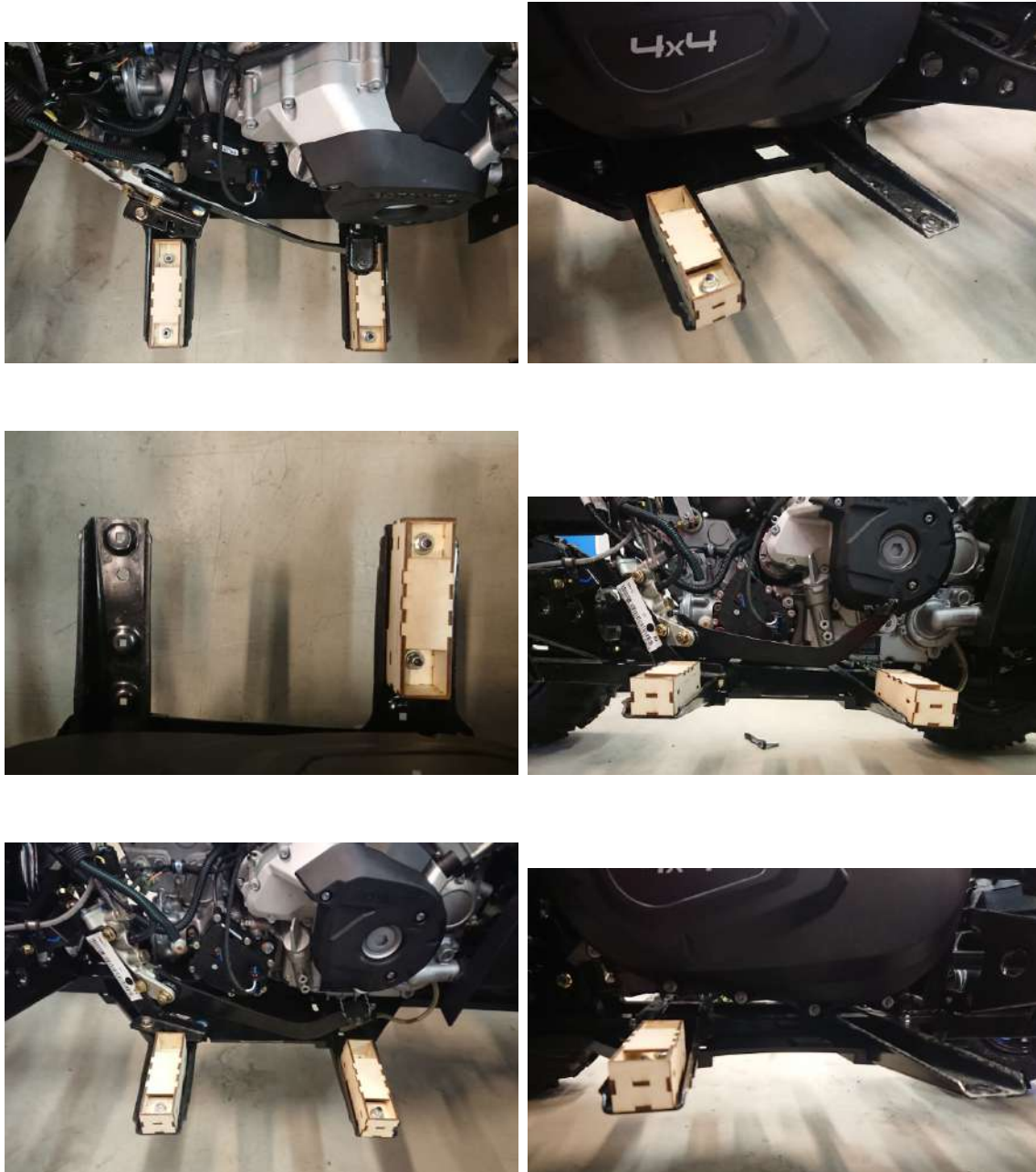
5.7 Feste til våpenstasjon

Første fase av design av ramme til våpenstasjonen består av å måle frem til aktuelle festepunkter på ATV-en, hvor fothviler festene blir brukt som hovedpunkter, med noen støttepunkter høyere opp på ATV-en.

For å finne avstand mellom festepunktene på fothvileren ble det designet en enkel versjon av fothvileren som er på ATV-en originalt (figur 14). Dette ble gjort for å kartlegge hvordan hullbildet på og mellom de nedre festepunktene er. Deretter ble hullbildet testet ved å designe deler for laserkutting i kryssfiner for montering der fothvilerene originalt sitter, for bekreftelse av hullbildet og et enklere utgangspunkt for videre målinger.



Figur 12: Festepunkt til fothvilere - designet i Solidwork



Figur 13: Laserkuttet fothvilere

Etter bekreftelsen av hullbildet ved fothvilerene ble det gjort det samme for festepunktene høyere oppe på ATV-en, som vil bli brukt til støtte av rammen. Disse festene var vanskeligere plassert i forhold til måling, så her ble det også designet en test-del i kryssfiner for å bekrefte plassering, og gi et enklere utgangspunkt for videre måling.

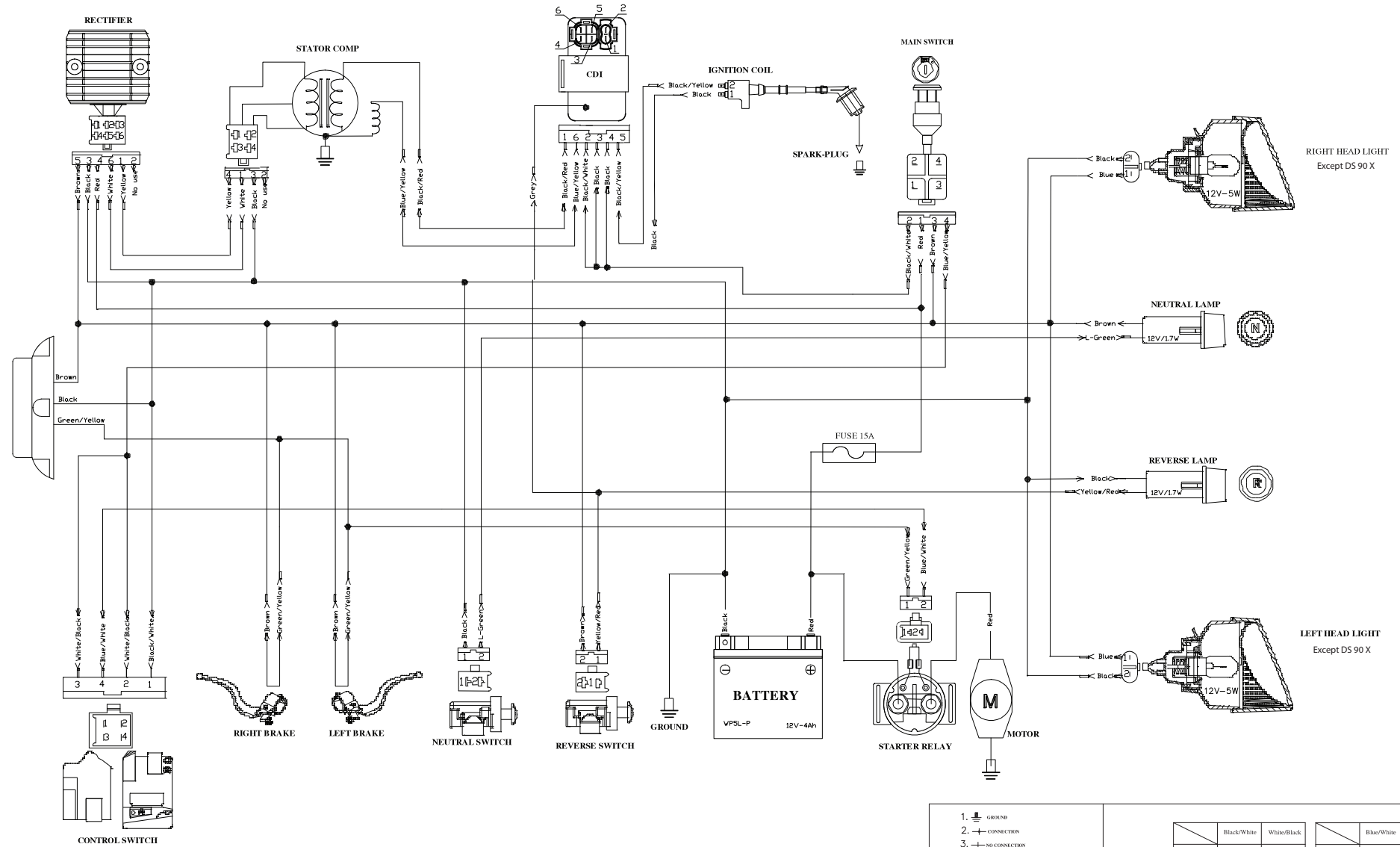


Figur 14: Festepunkt til støtte av ramme

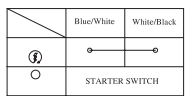
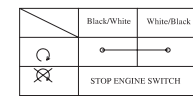
A.8 Kretsskjema for ATV

2019 DS 70 / DS 90 / DS90 X

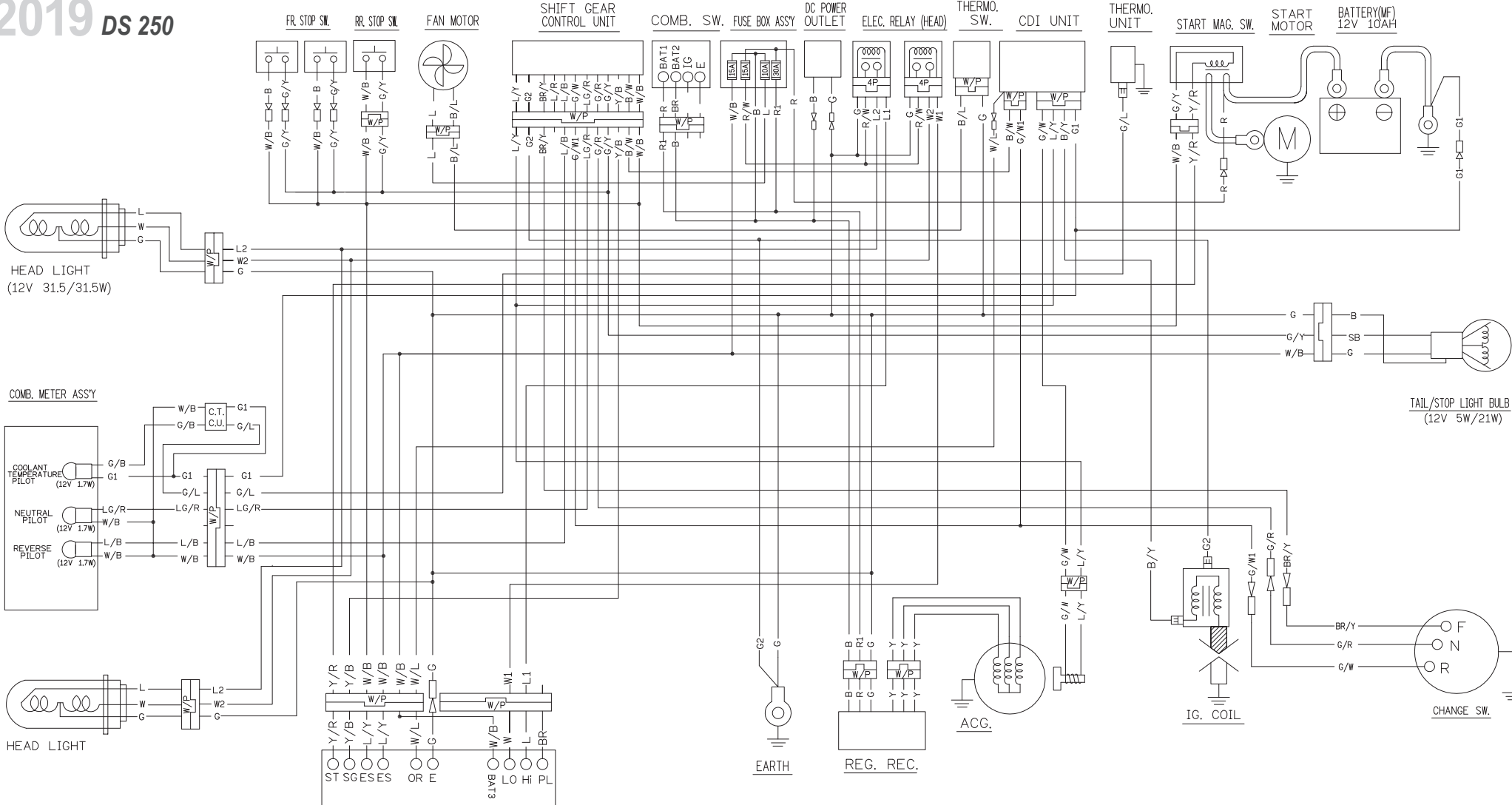
TAIL LIGHT
Except DS 90 X



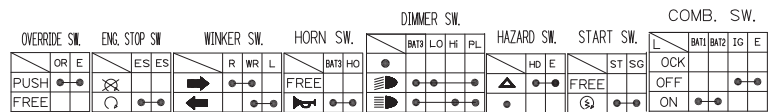
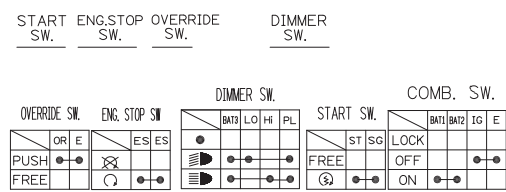
1. GROUND
2. CONNECTION
3. NO CONNECTION
4. MALE TERMINAL
5. FEMALE TERMINAL
6. SIMPLE CONDUCTOR WIRE



2019 DS 250

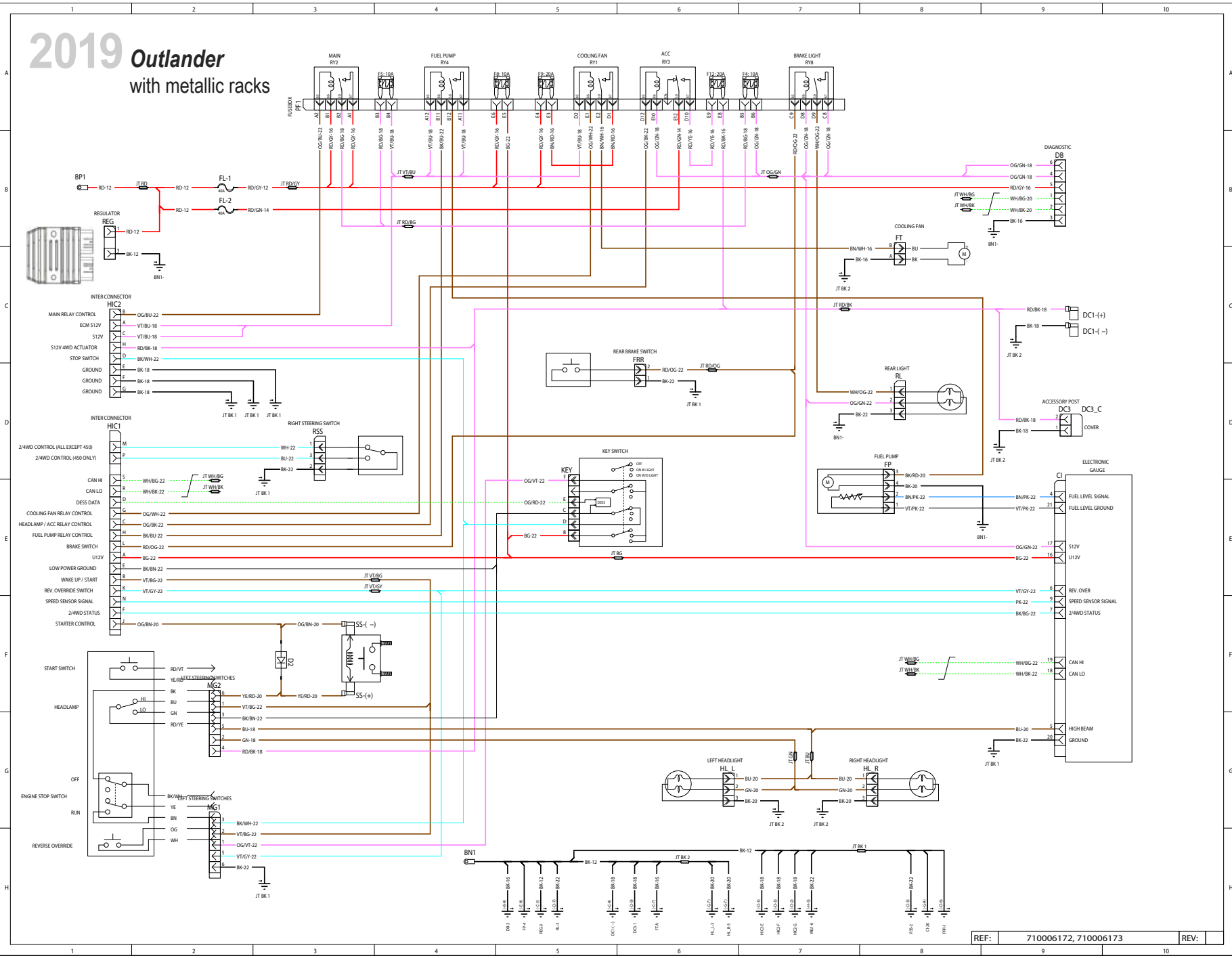


B	BLACK
Y	YELLOW
L	BLUE
G	GREEN
R	RED
W	WHITE
BR	BROWN
O	ORANGE
SB	SKY BLUE
LG	LIGHT GREEN
P	PINK
GR	GRAY



2019 Outlander

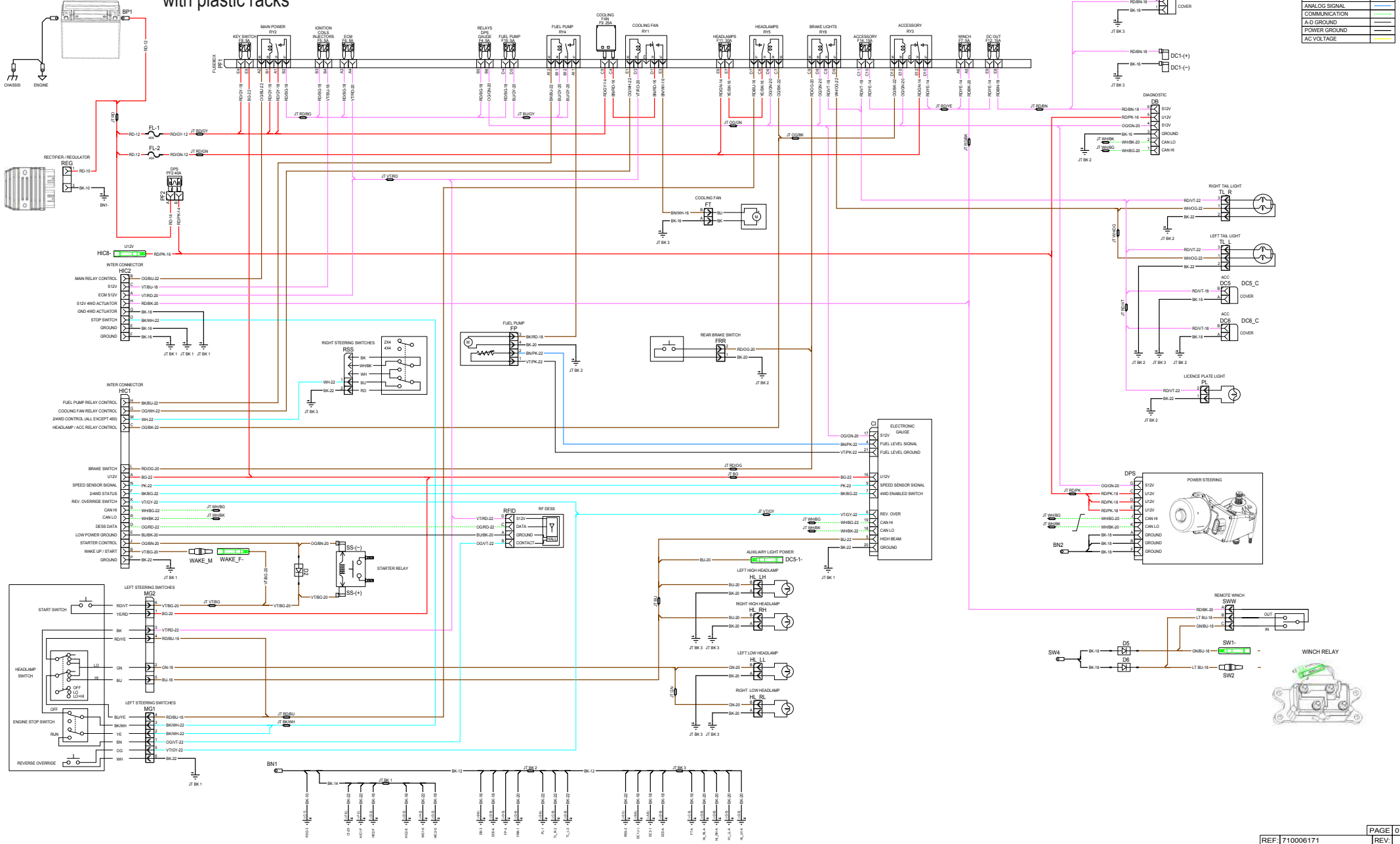
with metallic racks



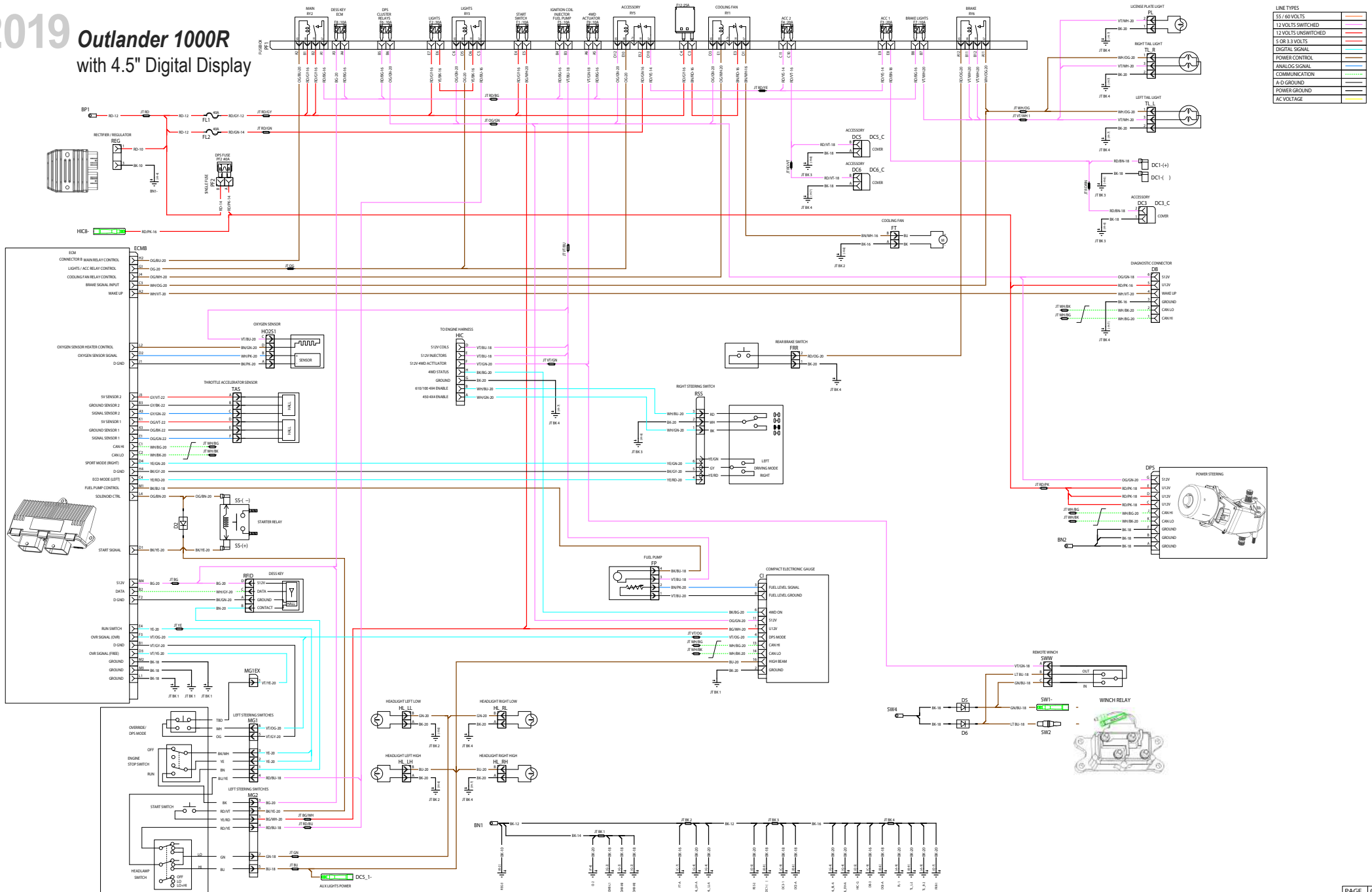
2019 Outlander / Renegade with plastic racks

LINE TYPES

55 / 60 VOLTS	[Red line]
12 VOLTS SWITCHED	[Orange line]
12 VOLTS UNSWITCHED	[Yellow line]
5 OR 3.3 VOLTS	[Green line]
DIGITAL SIGNAL	[Blue line]
POWER CONTROL	[Purple line]
ANALOG SIGNAL	[Cyan line]
COMMUNICATION	[Magenta line]
A/D GROUND	[Black line]
POWER GROUND	[Brown line]
AC VOLTAGE	[Light blue line]



2019 Outlander 1000R with 4.5" Digital Display

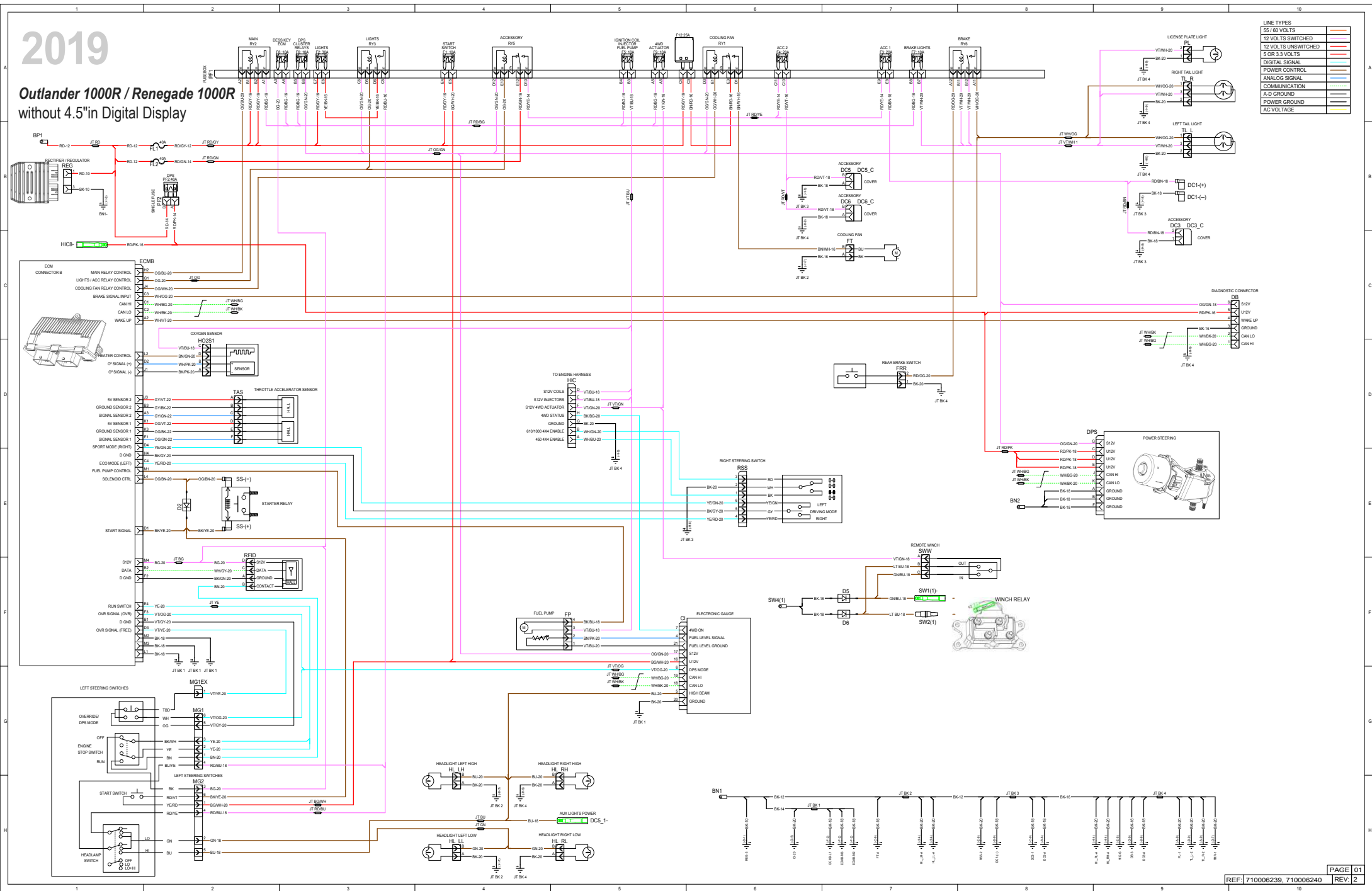


LINE TYPES

55 / 60 VOLTS	[Red line]
12 VOLTS SWITCHED	[Orange line]
12 VOLTS UNSWITCHED	[Yellow line]
5 OR 3.3 VOLTS	[Green line]
DIGITAL SIGNAL	[Blue line]
POWER CONTROL	[Purple line]
ANALOG SIGNAL	[Cyan line]
COMMUNICATION	[Magenta line]
A-D GROUND	[Black line]
POWER GROUND	[Brown line]
AC VOLTAGE	[Grey line]

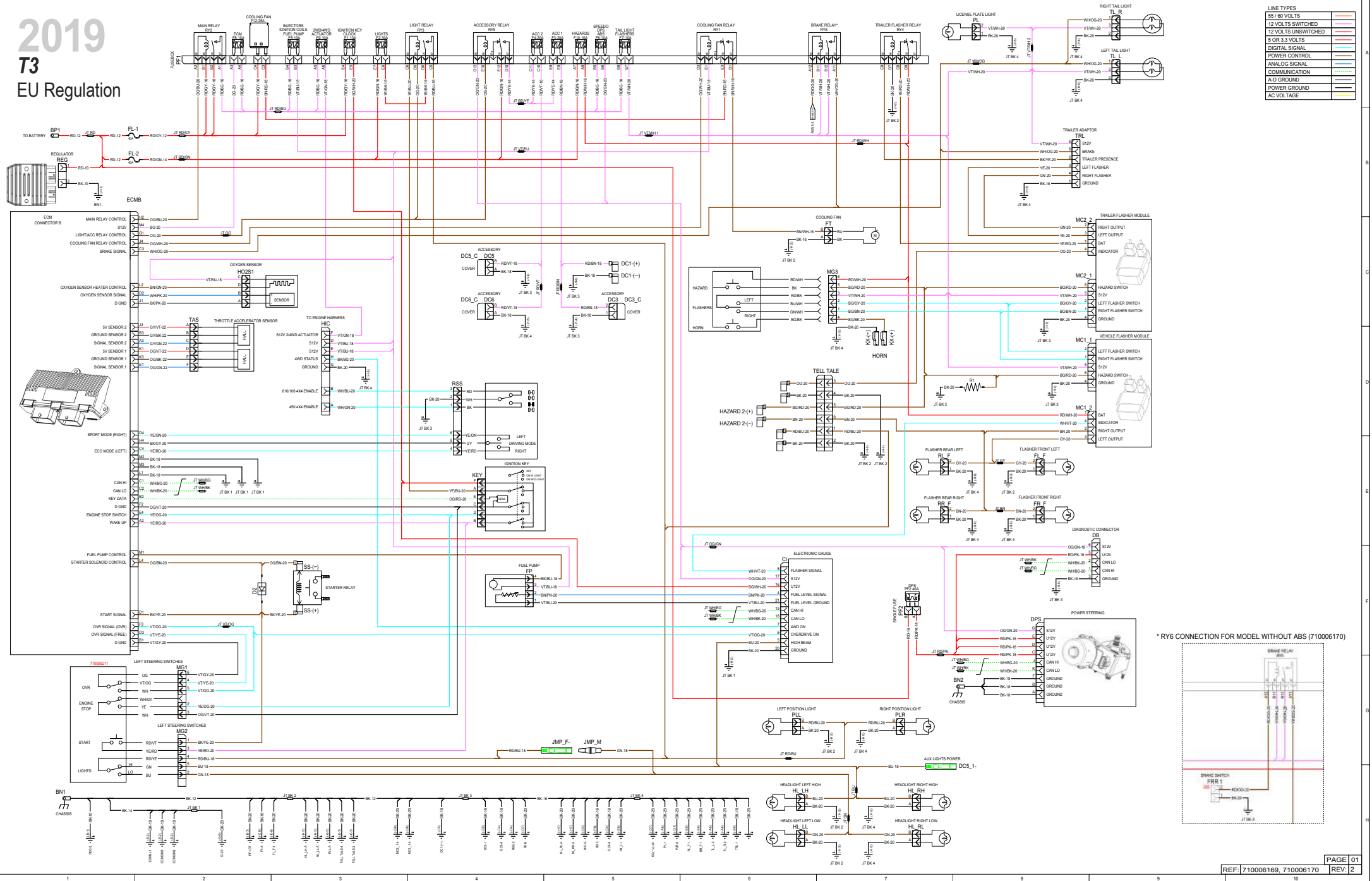
2019

Outlander 1000R / Renegade 1000R without 4.5" in Digital Display



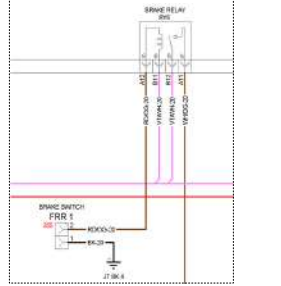
LINE TYPES	Color
55 / 60 VOLTS	Red
12 VOLTS SWITCHED	Yellow
12 VOLTS UNSWITCHED	Blue
5 OR 3.3 VOLTS	Green
DIGITAL SIGNAL	Orange
POWER CONTROL	Purple
ANALOG SIGNAL	Black
COMMUNICATION	Brown
A/D GROUND	White
POWER GROUND	Grey
AC VOLTAGE	Grey

2019 T3 EU Regulation



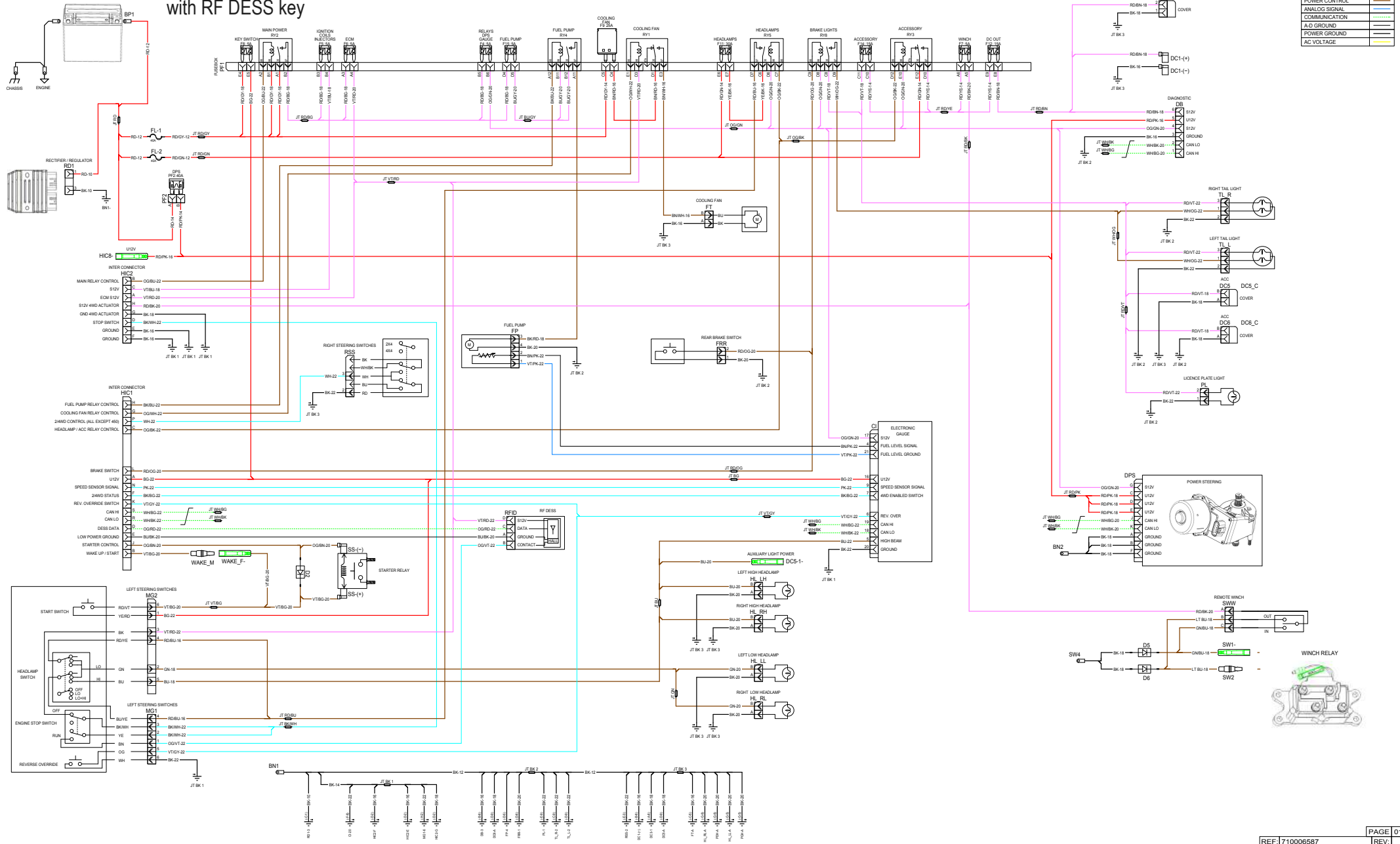
LINE TYPES	
55 / 80 VOLTS	(Red line)
12 VOLTS SWITCHED	(Orange line)
12 VOLTS UNSWITCHED	(Yellow line)
5 OR 3.3 VOLTS	(Green line)
DIGITAL SIGNAL	(Blue line)
POWER CONTROL	(Purple line)
ANALOG SIGNAL	(Light blue line)
COMMUNICATION	(Light green line)
A-D GROUND	(Black line)
POWER GROUND	(Brown line)
AC VOLTAGE	(Grey line)

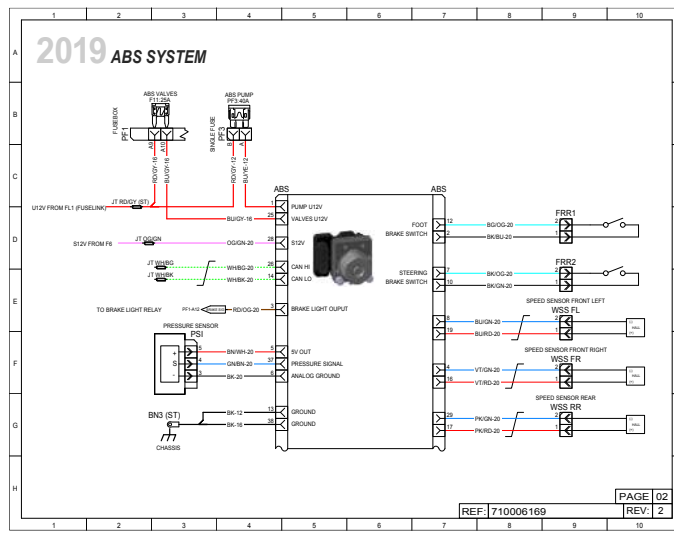
* RY6 CONNECTION FOR MODEL WITHOUT ABS (710006170)



2019 Outlander 6x6 with RF DESS key

LINE TYPES	
50/60 VOLTS	
12 VOLTS SWITCHED	
5 OR 3.3 VOLTS	
DIGITAL SIGNAL	
POWER CONTROL	
ANALOG SIGNAL	
COMMUNICATION	
A-B GROUND	
POWER GROUND	
AC VOLTAGE	

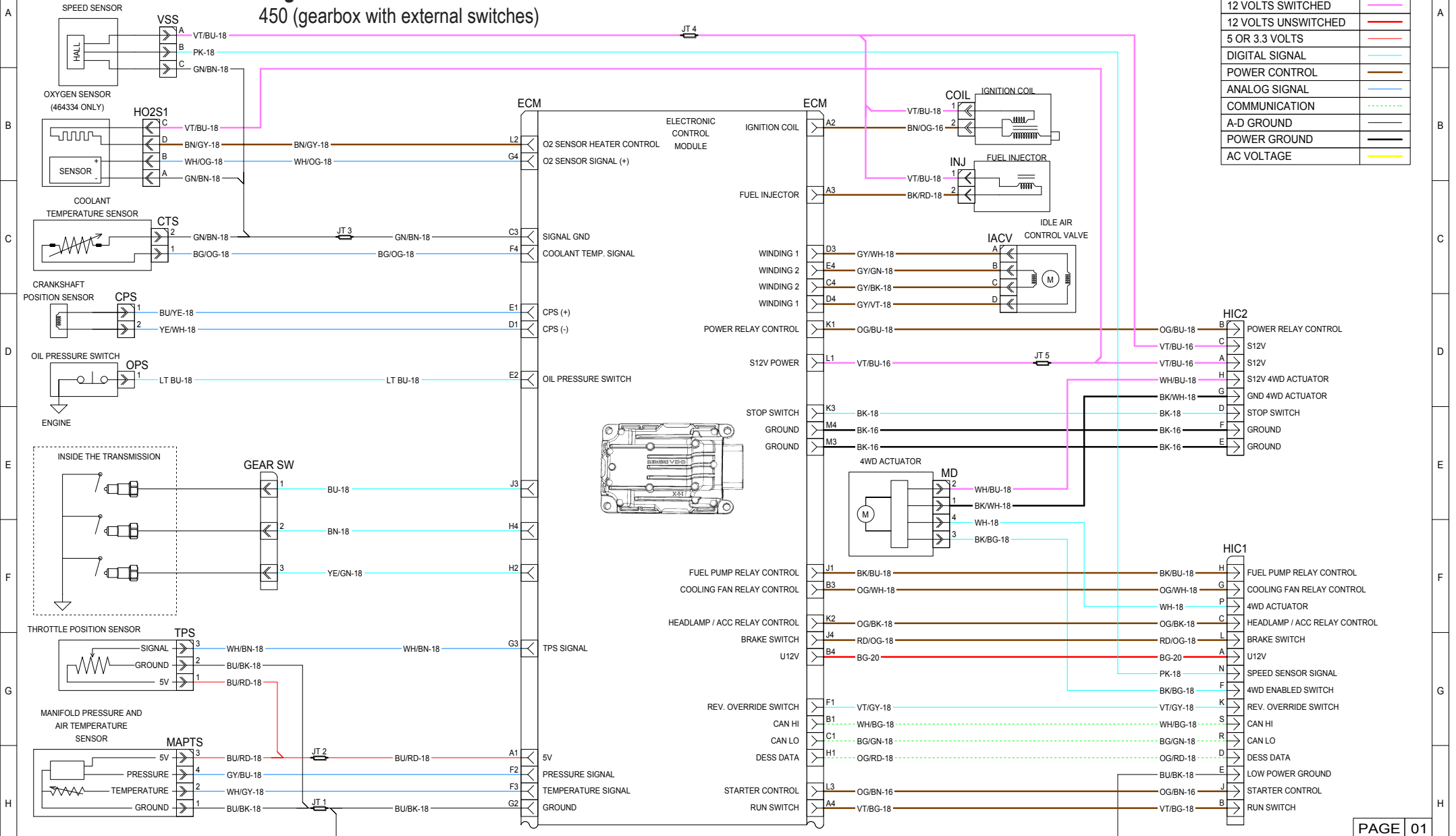




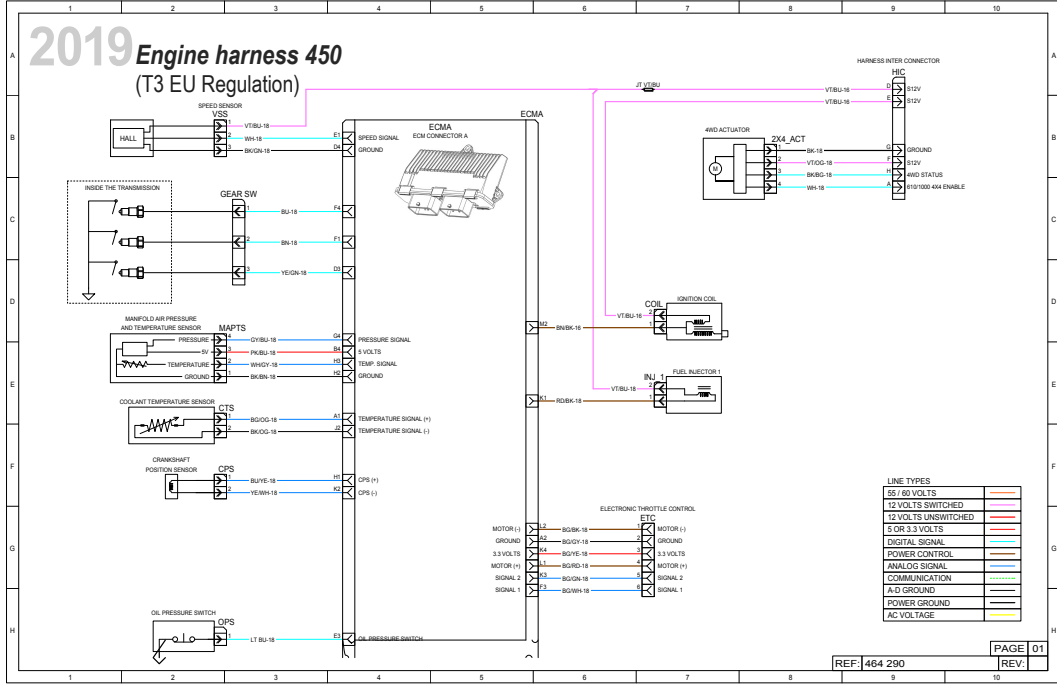
2019 Engine Harness

450 (gearbox with external switches)

LINE TYPES	
55 / 60 VOLTS	
12 VOLTS SWITCHED	
12 VOLTS UNSWITCHED	
5 OR 3.3 VOLTS	
DIGITAL SIGNAL	
POWER CONTROL	
ANALOG SIGNAL	
COMMUNICATION	
A-D GROUND	
POWER GROUND	
AC VOLTAGE	



2019 Engine harness 450 (T3 EU Regulation)



LINE TYPES

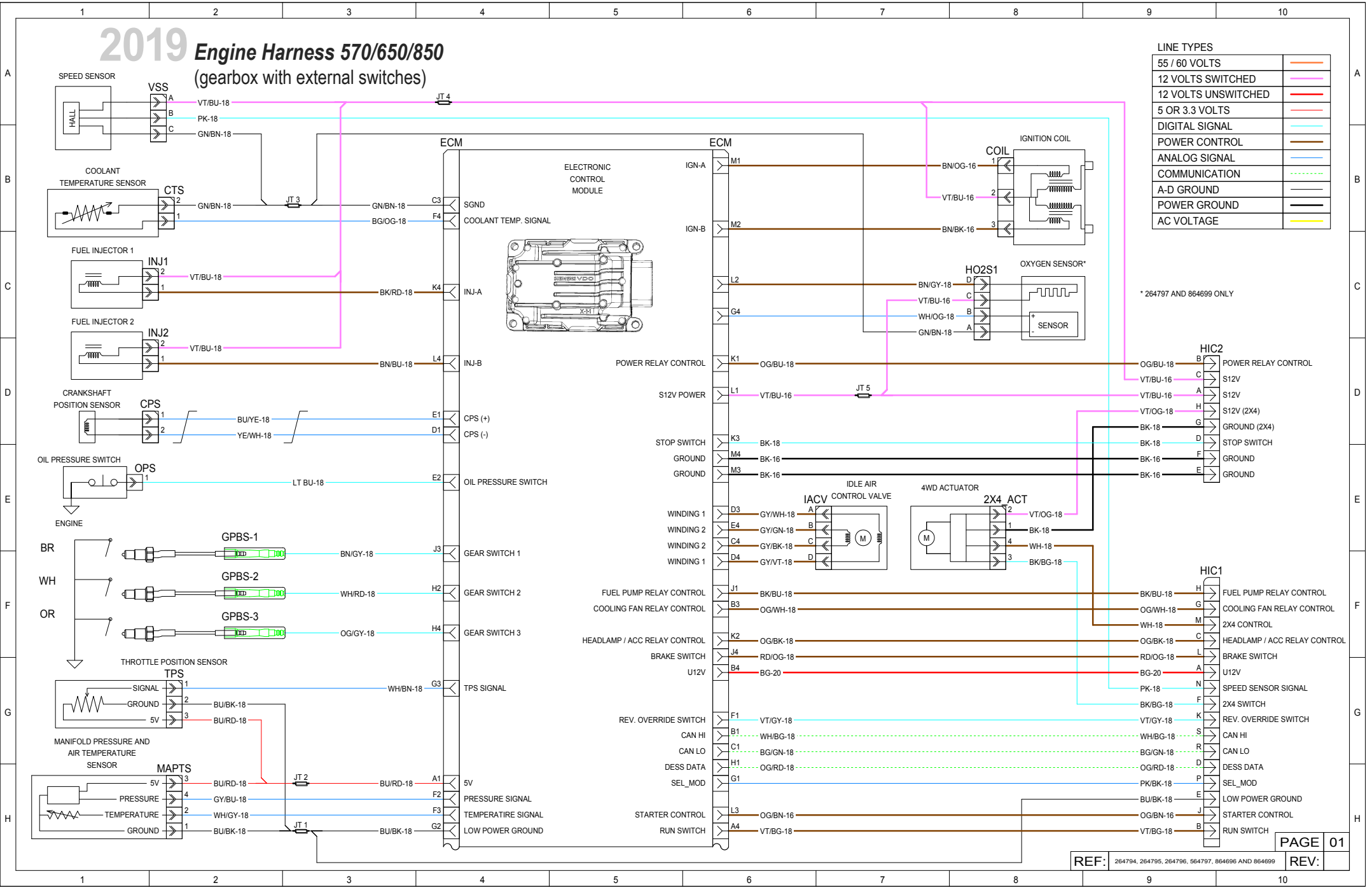
5S / 60 VOLTS	[Line Style]
12 VOLTS SWITCHED	[Line Style]
12 VOLTS UNSWITCHED	[Line Style]
5 OR 3.3 VOLTS	[Line Style]
DIGITAL SIGNAL	[Line Style]
POWER CONTROL	[Line Style]
ANALOG SIGNAL	[Line Style]
COMMUNICATION	[Line Style]
A/D GROUND	[Line Style]
POWER GROUND	[Line Style]
AC VOLTAGE	[Line Style]

2019 Engine Harness 570/650/850

(gearbox with external switches)

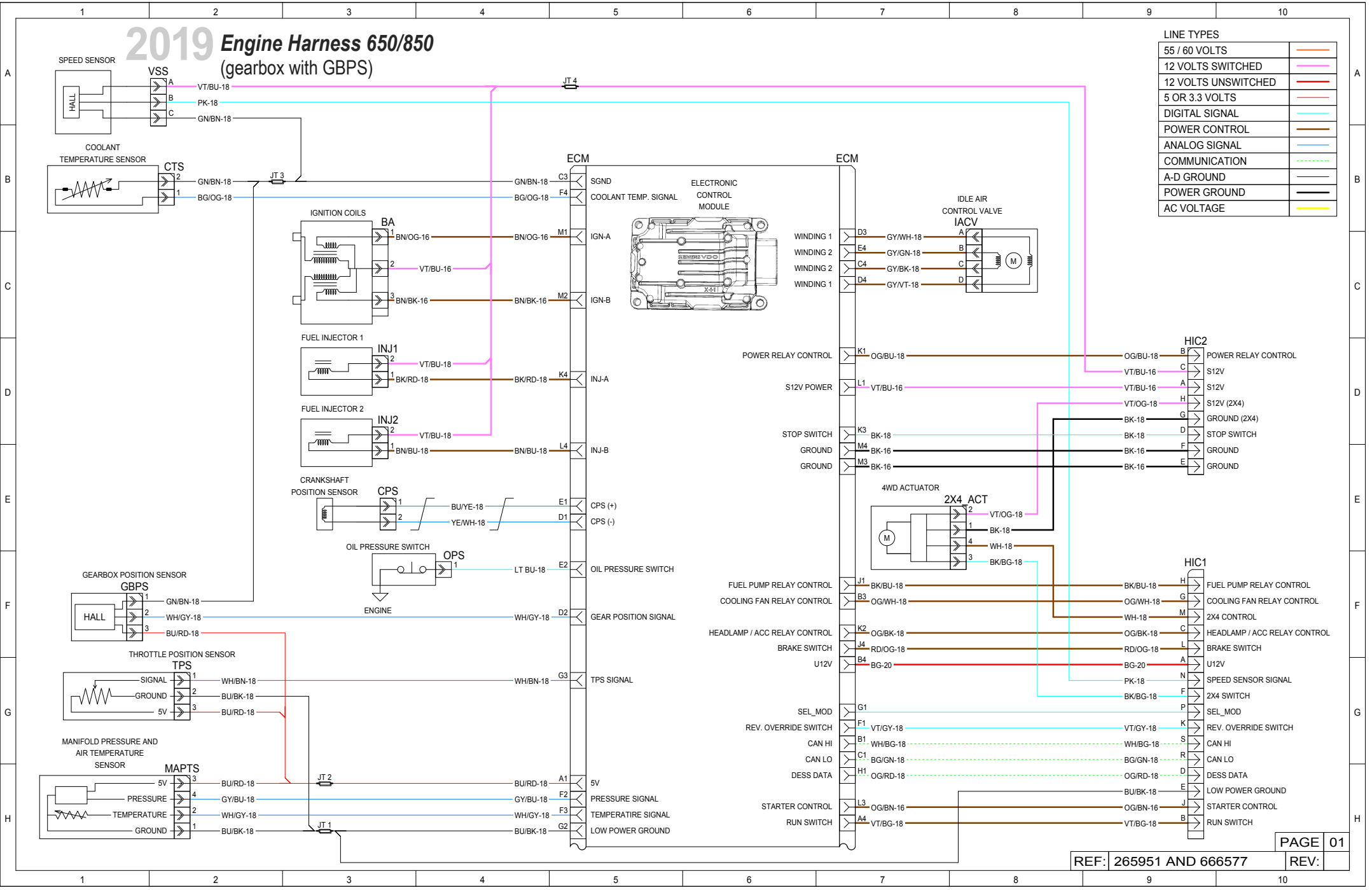
LINE TYPES	
55 / 60 VOLTS	
12 VOLTS SWITCHED	
12 VOLTS UNSWITCHED	
5 OR 3.3 VOLTS	
DIGITAL SIGNAL	
POWER CONTROL	
ANALOG SIGNAL	
COMMUNICATION	
A-D GROUND	
POWER GROUND	
AC VOLTAGE	

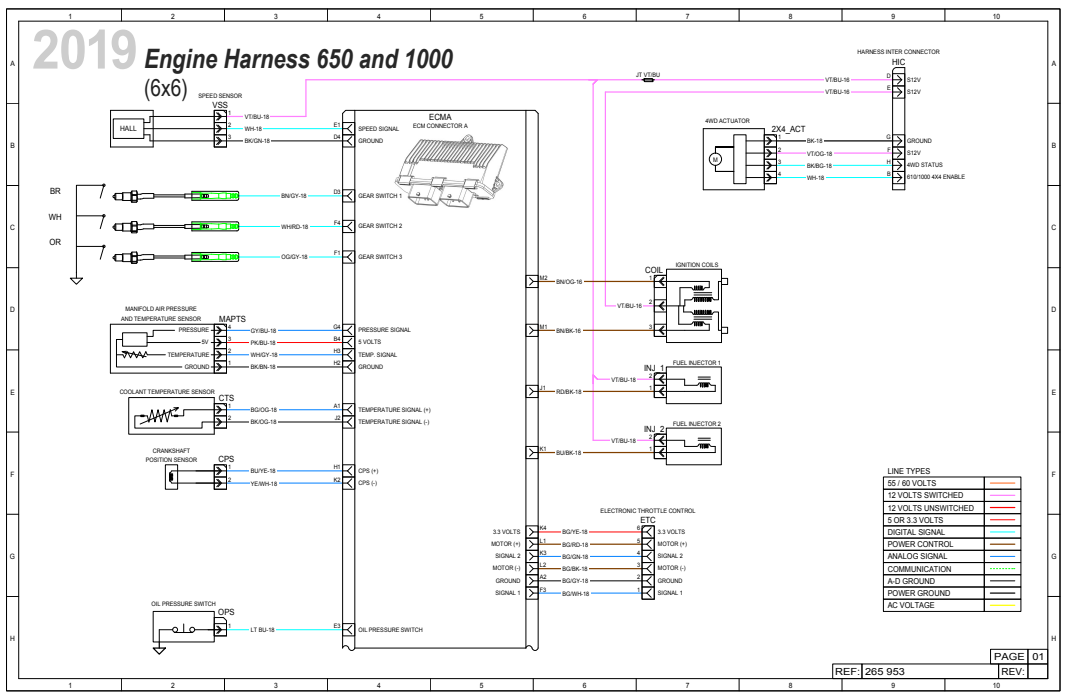
* 264797 AND 864699 ONLY



2019 Engine Harness 650/850 (gearbox with GBPS)

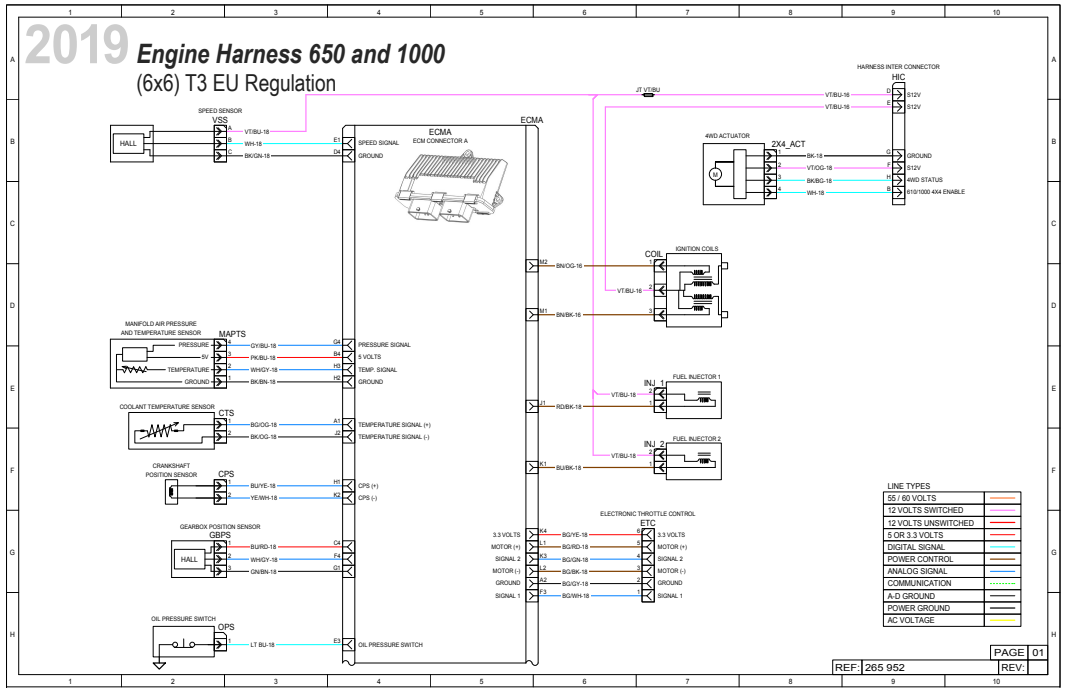
LINE TYPES	
55 / 60 VOLTS	
12 VOLTS SWITCHED	
12 VOLTS UNSWITCHED	
5 OR 3.3 VOLTS	
DIGITAL SIGNAL	
POWER CONTROL	
ANALOG SIGNAL	
COMMUNICATION	
A-D GROUND	
POWER GROUND	
AC VOLTAGE	

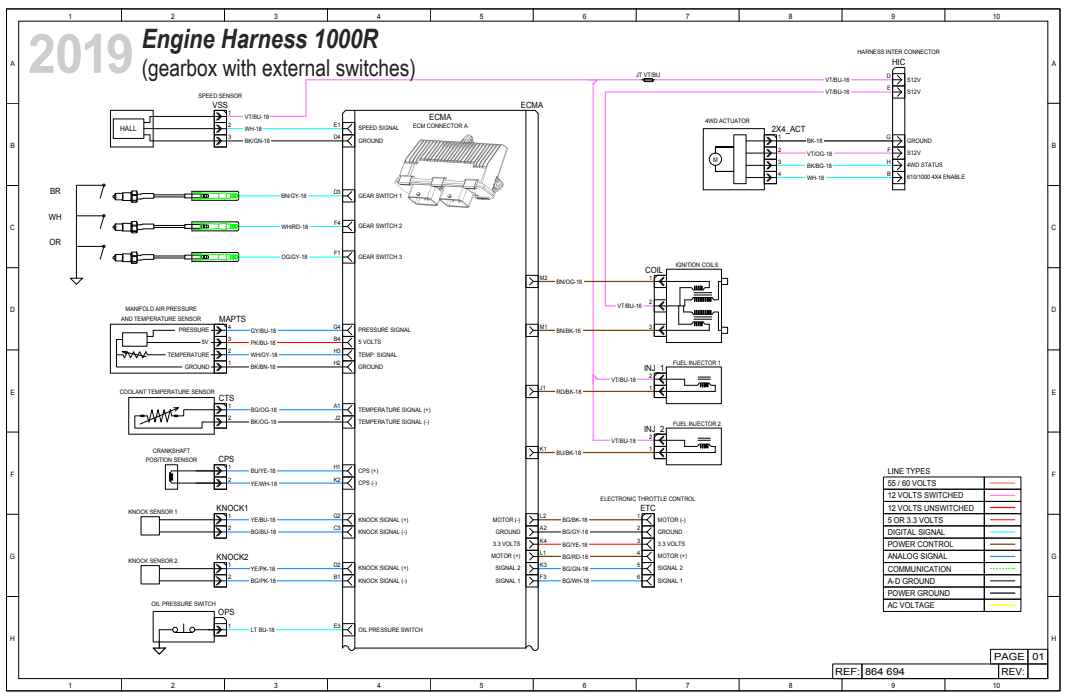


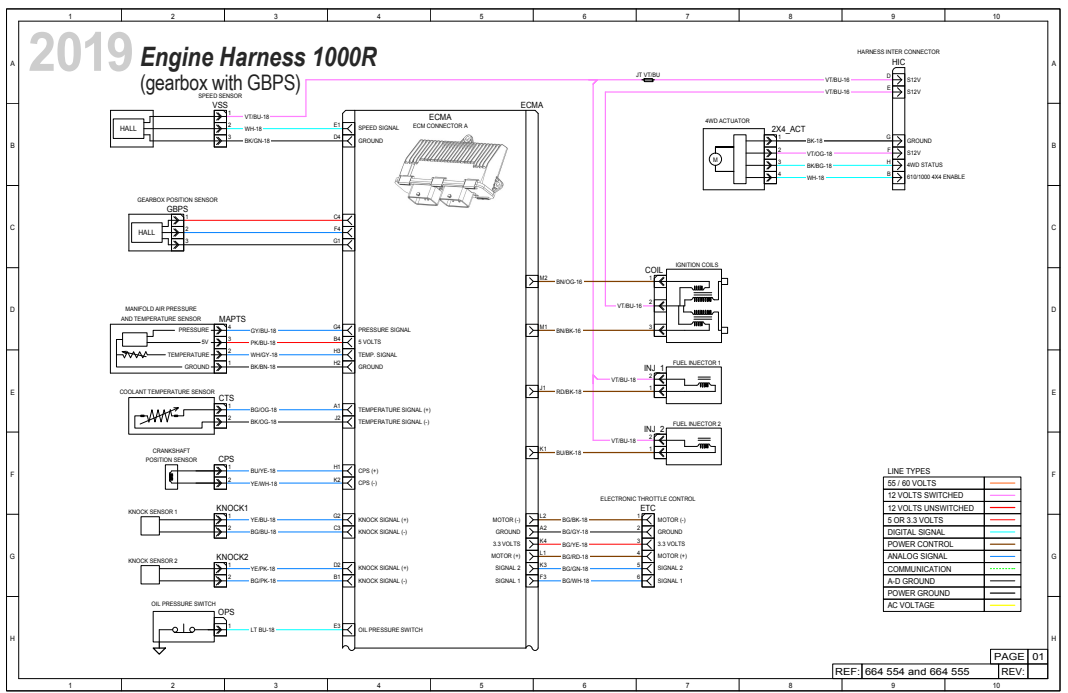


2019 Engine Harness 650 and 1000

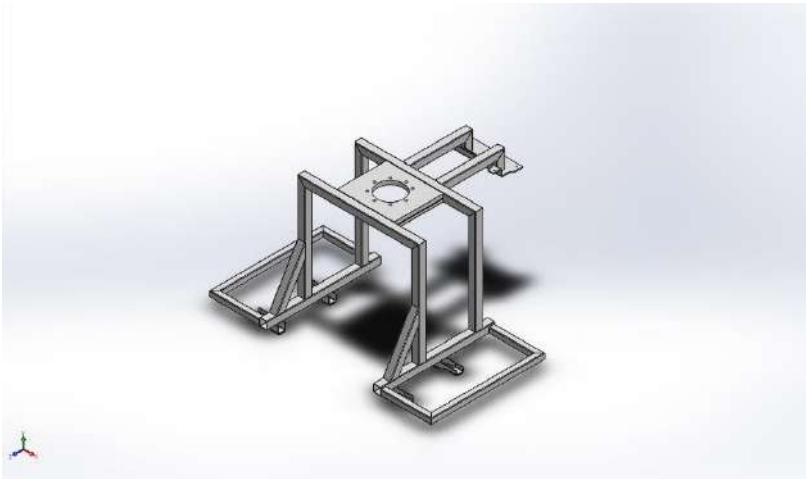
(6x6) T3 EU Regulation







A.9 FEM analyse, ramme og innfestning



Simulation of Assem firkantør

Date: onsdag 22. mai 2019
Designer: Torger T. Tokvam
Study name: Static 1
Analysis type: Static

Table of Contents

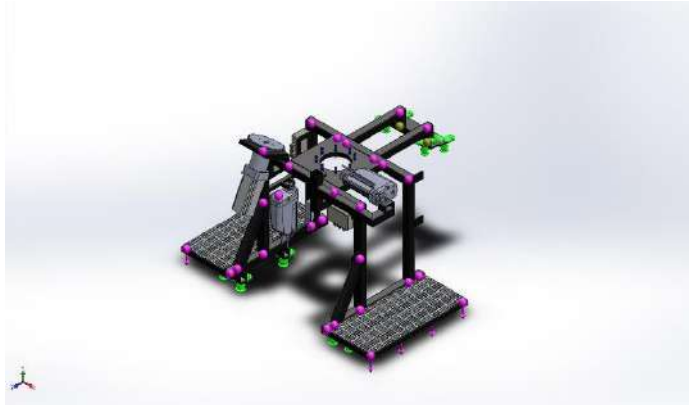
Description.....	1
Assumptions	2
Model Information	2
Study Properties	8
Units	9
Material Properties	10
Loads and Fixtures.....	14
Connector Definitions.....	15
Contact Information.....	19
Mesh information	22
Sensor Details	22
Resultant Forces	23
Beams.....	24
Study Results	28
Conclusion	36

Description

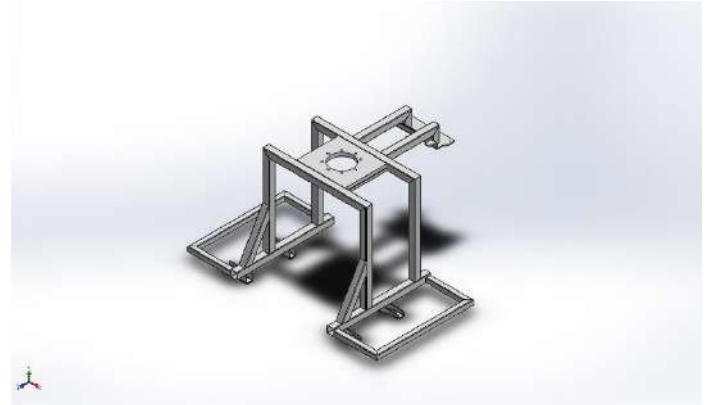
Kontaktanalyse av rammen og dens innfestninger mot ATV-en, alle deler designes for utforming i materiale med 355 MPa flytegrense. Det er tatt høyde for at det vil bli montert flere komponenter på rammen ettersom prosjektet skal videreføres. Derfor er målet å havne på ca. 150 MPa på deler hvor det er høyst usannsynlig at det vil bli påmontert andre komponenter, og ca. 100 MPa i deler som muligens vil bli belastet ytterligere av andre komponenter senere.



Assumptions

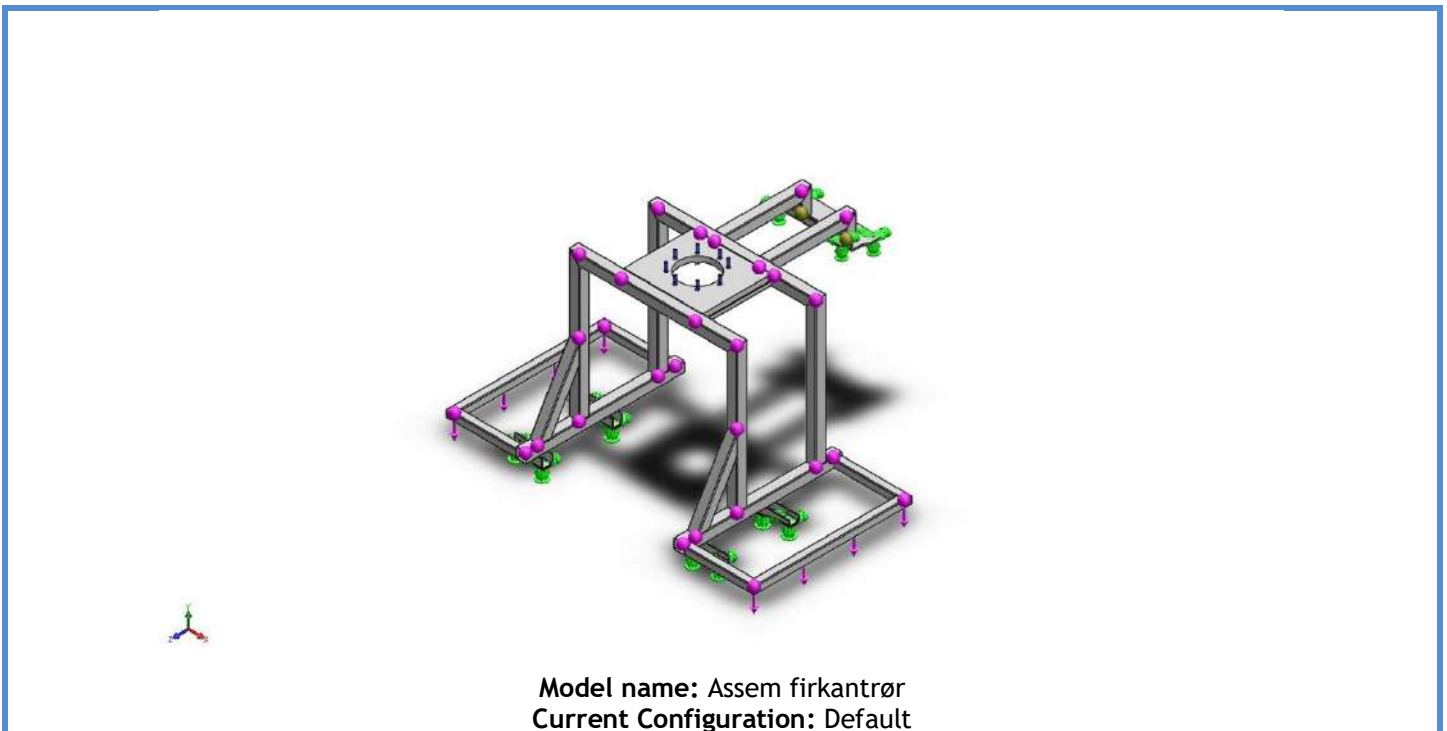


Original Model



Model Analyzed

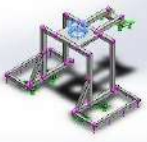
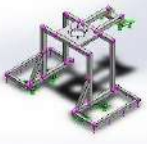
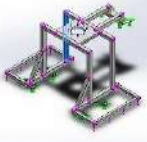
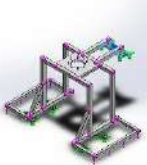
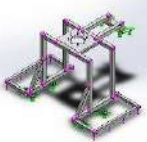
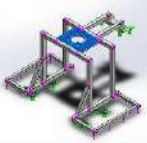
Model Information

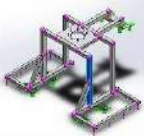
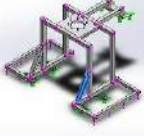
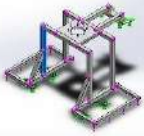
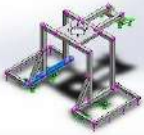
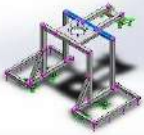


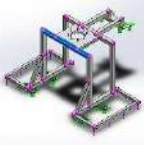
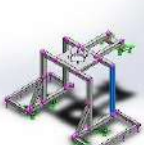
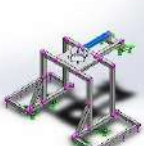
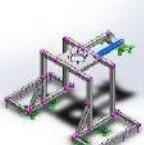
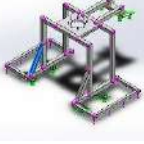
Model name: Assem firkantrør
Current Configuration: Default

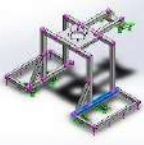
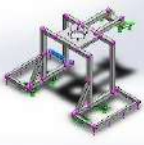
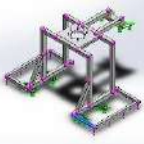

Solid Bodies

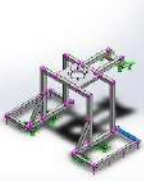
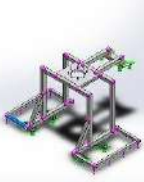
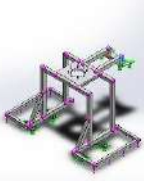
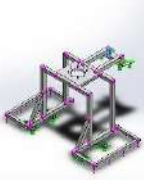
Document Name and Reference	Treated As	Volumetric Properties	Document Path/Date Modified
-----------------------------	------------	-----------------------	-----------------------------


Cut-Extrude8 	Solid Body	Mass:0.427928 kg Volume:0.000419537 m ³ Density:1020 kg/m ³ Weight:4.19369 N	C:\Users\torge\Documents \Bachelor\ATV\Bac_2019_ 60204045-10.SLDPRT Apr 22 14:07:07 2019
Split Line1 	Solid Body	Mass:0.237946 kg Volume:0.000233281 m ³ Density:1020 kg/m ³ Weight:2.33187 N	C:\Users\torge\Documents \Bachelor\ATV\Innfesting ramme\Festebrackett bak.SLDPRT May 22 22:58:25 2019
CirPattern1 	Solid Body	Mass:1.25746 kg Volume:0.0012328 m ³ Density:1020 kg/m ³ Weight:12.3231 N	C:\Users\torge\Documents \Bachelor\ATV\Monterings plate RWS til firkantrør.SLDPRT May 22 22:58:25 2019
Shell Bodies			
Document Name and Reference	Formulation	Volumetric Properties	Document Path/Date Modified
Shell-20 	Thin	Thickness:0 mm Weight:0 N Volume:0 m ³ Mass:0 kg Density:-nan(ind)kg/m ³	C:\Users\torge\Documents \Bachelor\ATV\Assem firkantrør.SLDASM May 22 22:58:27 2019
Shell-21 	Thin	Thickness:0 mm Weight:0 N Volume:0 m ³ Mass:0 kg Density:-nan(ind)kg/m ³	C:\Users\torge\Documents \Bachelor\ATV\Assem firkantrør.SLDASM May 22 22:58:27 2019
Beam Bodies:			
Document Name and Reference	Formulation	Properties	Document Path/Date Modified
Beam-1(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[3]) 	Beam - Uniform C/S	Section Standard- Section Area: 0.000297133m ² Length:530mm Volume:0.000157482m ³ Mass Density:1020kg/m ³ Mass:0.160632kg Weight:1.57419N	C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019

<p>Beam-2(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[4])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:530mm Volume:0.00015748m³ Mass Density:1020kg/m³ Mass:0.16063kg Weight:1.57417N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-3(Trim/Extend1[2])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:254.449mm Volume:7.56037e-05m³ Mass Density:1020kg/m³ Mass:0.0771158kg Weight:0.755735N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-5(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[1])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:530mm Volume:0.00015748m³ Mass Density:1020kg/m³ Mass:0.16063kg Weight:1.57417N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-6(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[11])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:600mm Volume:0.00017828m³ Mass Density:1020kg/m³ Mass:0.181845kg Weight:1.78208N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-7(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[5])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:600mm Volume:0.000178282m³ Mass Density:1020kg/m³ Mass:0.181848kg Weight:1.78211N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>

<p>Beam-8(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[2])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:600mm Volume:0.000178282m³ Mass Density:1020kg/m³ Mass:0.181848kg Weight:1.78211N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-10(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[6])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:530mm Volume:0.000157482m³ Mass Density:1020kg/m³ Mass:0.160632kg Weight:1.57419N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-11(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[7])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:310mm Volume:9.21112e-05m³ Mass Density:1020kg/m³ Mass:0.0939534kg Weight:0.920743N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-12(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[9])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:310mm Volume:9.21112e-05m³ Mass Density:1020kg/m³ Mass:0.0939534kg Weight:0.920743N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-13(Trim/Extend1[1])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:254.449mm Volume:7.56037e-05m³ Mass Density:1020kg/m³ Mass:0.0771158kg Weight:0.755735N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>

<p>Beam-14(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[12])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:600mm Volume:0.00017828m³ Mass Density:1020kg/m³ Mass:0.181845kg Weight:1.78208N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-15(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[2])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:570mm Volume:0.000123767m³ Mass Density:1020kg/m³ Mass:0.126243kg Weight:1.23718N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-16(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[3])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:250mm Volume:5.42839e-05m³ Mass Density:1020kg/m³ Mass:0.0553696kg Weight:0.542622N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-17(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[4])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:250mm Volume:5.42832e-05m³ Mass Density:1020kg/m³ Mass:0.0553689kg Weight:0.542615N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-18(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[5])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:570mm Volume:0.000123767m³ Mass Density:1020kg/m³ Mass:0.126243kg Weight:1.23718N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>

<p>Beam-19(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[6])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:250mm Volume:5.42839e-05m³ Mass Density:1020kg/m³ Mass:0.0553696kg Weight:0.542622N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-20(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[1])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000217133m² Length:250mm Volume:5.42832e-05m³ Mass Density:1020kg/m³ Mass:0.0553689kg Weight:0.542615N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-22(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[10])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:72.9956mm Volume:2.17522e-05m³ Mass Density:1020kg/m³ Mass:0.0221872kg Weight:0.217435N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-23(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[8])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000297133m² Length:72.9956mm Volume:2.17522e-05m³ Mass Density:1020kg/m³ Mass:0.0221872kg Weight:0.217435N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>
<p>Beam-26(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[2])</p> 	<p>Beam - Uniform C/S</p>	<p>Section Standard- Section Area: 0.000137133m² Length:260mm Volume:3.56545e-05m³ Mass Density:1020kg/m³ Mass:0.0363676kg Weight:0.356403N</p>	<p>C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPRT May 22 22:58:26 2019</p>

Beam-27(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[1]) 	Beam - Uniform C/S	Section Standard- Section Area: 0.000137133m ² Length: 260mm Volume: 3.56545e-05m ³ Mass Density: 1020kg/m ³ Mass: 0.0363676kg Weight: 0.356403N	C:\Users\torge\Documents \Bachelor\ATV\Ramme firkantrør.SLDPR T May 22 22:58:26 2019
--	--------------------	---	---

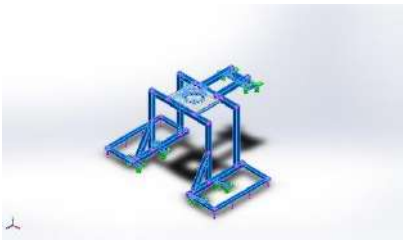
Study Properties

Study name	Static 1
Analysis type	Static
Mesh type	Mixed Mesh
Thermal Effect:	On
Thermal option	Include temperature loads
Zero strain temperature	298 Kelvin
Include fluid pressure effects from SOLIDWORKS Flow Simulation	Off
Solver type	Direct sparse solver
Inplane Effect:	Off
Soft Spring:	Off
Inertial Relief:	Off
Incompatible bonding options	Automatic
Large displacement	Off
Compute free body forces	On
Friction	Off
Use Adaptive Method:	Off
Result folder	SOLIDWORKS document (C:\Users\torge\Documents\Bachelor\ATV)

Units

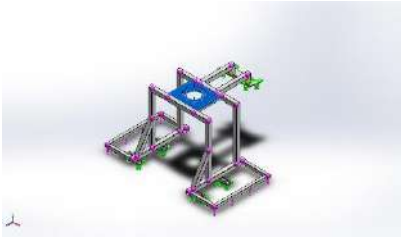
Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m ²

Material Properties

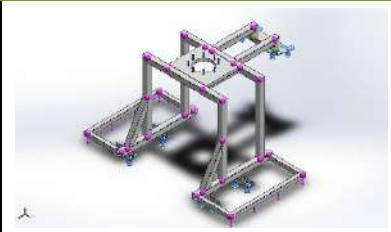
Model Reference	Properties	Components
	Name: 355MC Model type: Linear Elastic Isotropic Default failure criterion: Max von Mises Stress Yield strength: 3.55e+08 N/m ² Tensile strength: 5.5e+08 N/m ² Elastic modulus: 2e+11 N/m ² Poisson's ratio: 0.28 Mass density: 1020 kg/m ³ Shear modulus: 8e+10 N/m ²	SolidBody 1(Cut-Extrude8)(Bac_2019_6020404 5-10-1), SolidBody 1(Split Line1)(Festebrakett bak-1), SolidBody 1(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[3]))(Ramme firkantrør-1), SolidBody 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[4]))(Ramme firkantrør-1), SolidBody 3(Trim/Extend1[2])(Ramme firkantrør-1), SolidBody 5(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[1]))(Ramme firkantrør-1), SolidBody 6(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 X 2(40 x 40 x 2[11]))(Ramme firkantrør-1), SolidBody 7(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg -

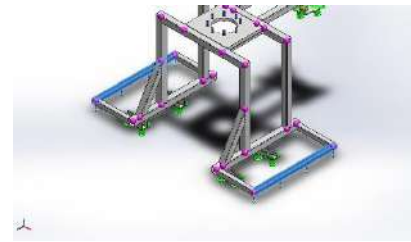
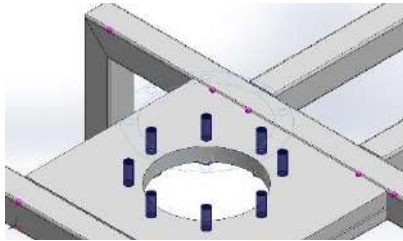
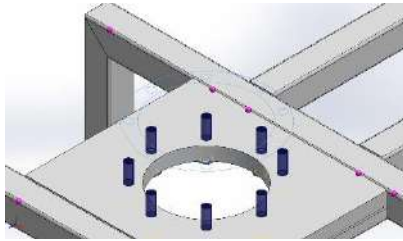
		<p>configured 40 X 40 X 2(40 x 40 x 2[5])(Ramme firkantrør-1), SolidBody 8(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[2])(Ramme firkantrør-1), SolidBody 10(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[6])(Ramme firkantrør-1), SolidBody 11(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[7])(Ramme firkantrør-1), SolidBody 12(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[9])(Ramme firkantrør-1), SolidBody 13(Trim/Extend1[1])(Ramme firkantrør-1), SolidBody 14(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 1mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(Firkantrør 2mm vegg - configured 40 X 40 X 2(40 x 40 x 2[12])(Ramme firkantrør-1).</p>
--	--	--

		<p>SolidBody 15(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[2]))(Ramme firkantrør-1),</p> <p>SolidBody 16(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[3]))(Ramme firkantrør-1),</p> <p>SolidBody 17(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[4]))(Ramme firkantrør-1),</p> <p>SolidBody 18(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[5]))(Ramme firkantrør-1),</p> <p>SolidBody 19(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[6]))(Ramme firkantrør-1),</p> <p>SolidBody 20(Firkantrør 1mm vegg - configured 30 X 30 X 1(Firkantrør 2mm vegg - configured 30 X 30 X 2(30 x 30 x 2 brett[1]))(Ramme firkantrør-1),</p> <p>SolidBody 22(Firkantrør 2mm vegg - configured 40 X 40 X 2(Chamfer1))(Ramme firkantrør-1),</p> <p>SolidBody 23(Firkantrør 2mm vegg - configured 40 X 40 X 2(Chamfer2))(Ramme firkantrør-1),</p> <p>SolidBody 26(Firkantrør 2mm vegg - configured 20 X 20 X 2(20 x 20 x 2[2]))(Ramme firkantrør-1),</p> <p>SolidBody 27(Firkantrør 2mm vegg - configured 20 X 20 X 2(20 x 20 x 2[1]))(Ramme firkantrør-1),</p> <p>SolidBody 1(Split Line1)(Underlag fothviler</p>
--	--	--

		<p>overgang-1), SolidBody 1(Split Line1)(Underlag fothviler overgang-2), SolidBody 1(Split Line1)(Underlag fothviler overgang-3), SolidBody 1(Split Line1)(Underlag fothviler overgang-4), SolidBody 1(Split Line1)(Underlag fothviler overgang-5), SolidBody 1(Split Line1)(Underlag fothviler overgang-6), SolidBody 1(Split Line1)(Underlag fothviler overgang-7), SolidBody 1(Split Line1)(Underlag fothviler overgang-8), SolidBody 1(Cut- Extrude1)(fothviler overgang plateknekk-1), SolidBody 1(Cut- Extrude1)(fothviler overgang plateknekk-2), SolidBody 1(Cut- Extrude1)(fothviler overgang plateknekk-5), SolidBody 1(Cut- Extrude1)(fothviler overgang plateknekk-6)</p>
Curve Data:N/A		
	<p>Name: S355J2+N Model type: Linear Elastic Isotropic Default failure criterion: Max von Mises Stress Yield strength: 3.55e+08 N/m² Tensile strength: 6.3e+08 N/m² Elastic modulus: 2e+11 N/m² Poisson's ratio: 0.28 Mass density: 1020 kg/m³ Shear modulus: 8e+10 N/m²</p>	<p>SolidBody 1(CirPattern1)(Monteringspla te RWS til firkantrør-1)</p>
Curve Data:N/A		

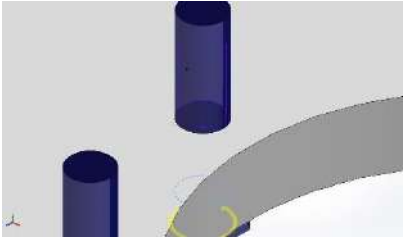
Loads and Fixtures

Fixture name	Fixture Image	Fixture Details		
Fixed-2		Entities: 12 face(s) Type: Fixed Geometry		
Resultant Forces				
Components	X	Y	Z	Resultant
Reaction force(N)	1500	7000	-4500	8455.77
Reaction Moment(N.m)	8.1918	-0.168132	-3.39964	8.87082

Load name	Load Image	Load Details
Force-2		Entities: 2 Beam (s) Reference: Edge< 1 > Type: Apply force Values: ---, ---, 2000 N Moments: ---, ---, --- N.m
Force-3		Entities: 1 face(s) Type: Apply normal force Value: 3000 N
Remote Load (Direct transfer)-1		Entities: 1 face(s) Type: Load (Direct transfer) Coordinate System: Global cartesian coordinates Force Values: -1500, ---, 4500 N Moment Values: ---, ---, --- N.m Reference coordinates: -300 920 -350 mm Components transferred: Force

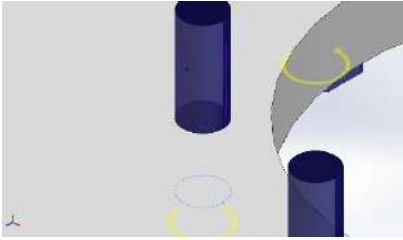
Connector Definitions

Pin/Bolt/Bearing Connector

Model Reference	Connector Details	Strength Details
 <p>Counterbore Screw-1</p>	<p>Entities: 1 edge(s), 1 face(s)</p> <p>Type: Bolt(Head/Nut diameter)(Counterbore screw)</p> <p>Head diameter: 18.6 mm</p> <p>Nominal shank diameter: 12.4</p> <p>Preload (Torque): 80</p> <p>Young's modulus: 2.1e+11</p> <p>Poisson's ratio: 0.28</p> <p>Preload units: N.m</p>	No Data

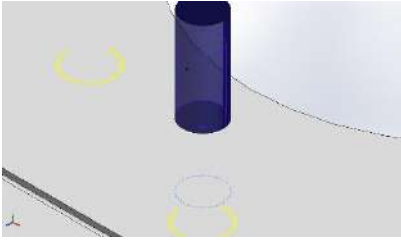
Connector Forces

Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-23616	0	23616
Shear Force (N)	388.02	0	149	415.64
Bending moment (N.m)	1.9461	0	-8.2831	8.5087

 <p>Counterbore Screw-2</p>	<p>Entities: 1 edge(s), 1 face(s)</p> <p>Type: Bolt(Head/Nut diameter)(Counterbore screw)</p> <p>Head diameter: 18.6 mm</p> <p>Nominal shank diameter: 12.4</p> <p>Preload (Torque): 80</p> <p>Young's modulus: 2.1e+11</p> <p>Poisson's ratio: 0.28</p> <p>Preload units: N.m</p>	No Data
--	--	---------

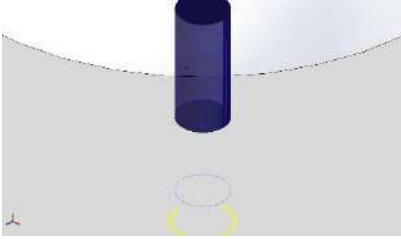
Connector Forces

Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-23907	0	23907
Shear Force (N)	852.33	0	-786.35	1159.7
Bending moment (N.m)	-13.569	0	-14.421	19.802

 <p>Counterbore Screw-3</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Count erbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
--	---	----------------

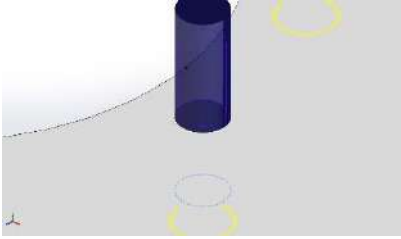
Connector Forces

Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-23565	0	23565
Shear Force (N)	89.158	0	-1332.3	1335.3
Bending moment (N.m)	-22.593	0	-1.3739	22.635

 <p>Counterbore Screw-4</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Count erbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
---	---	----------------

Connector Forces

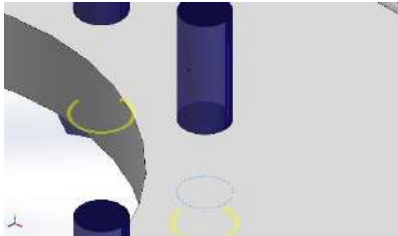
Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-24183	0	24183
Shear Force (N)	-270.6	0	-359.45	449.92
Bending moment (N.m)	-6.7051	0	5.4391	8.6337

 <p>Counterbore Screw-5</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Count erbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
--	---	----------------

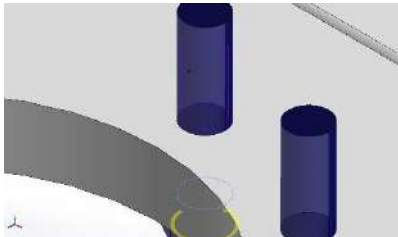
Connector Forces

Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-24631	0	24631

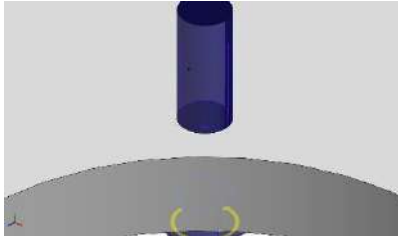
Shear Force (N)	423.22	0	98.165	434.45
Bending moment (N.m)	0.91653	0	-5.0967	5.1785

 <p>Counterbore Screw-6</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Counterbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
--	---	----------------

Connector Forces				
Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	-70.471	-25009	-44.941	25010
Shear Force (N)	452.44	0.42398	-945.41	1048.1
Bending moment (N.m)	-13.714	0.051056	-6.9077	15.356

 <p>Counterbore Screw-7</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Counterbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
--	---	----------------

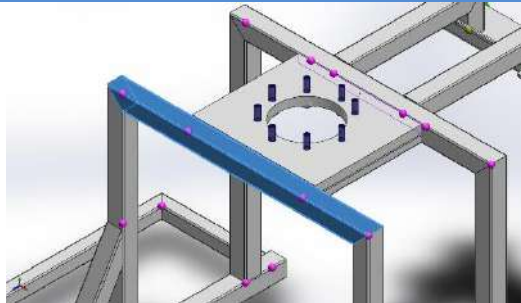
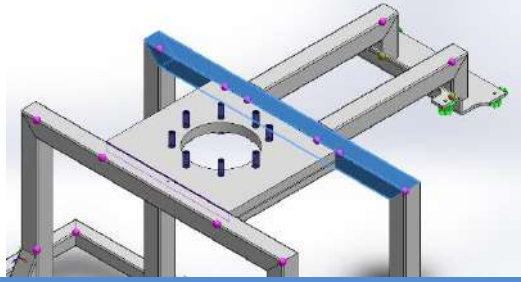
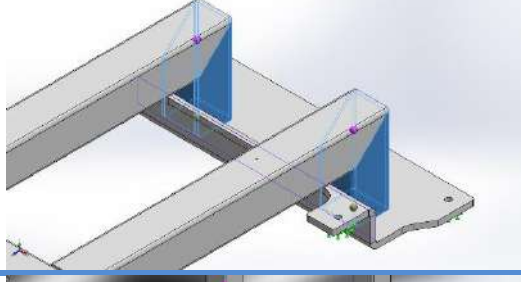
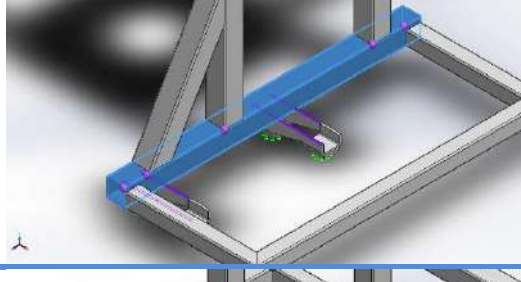
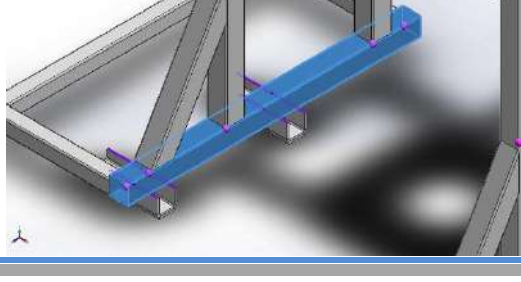
Connector Forces				
Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-24919	0	24919
Shear Force (N)	4.6265	0	-1083.4	1083.4
Bending moment (N.m)	-15.922	0	0.18742	15.923

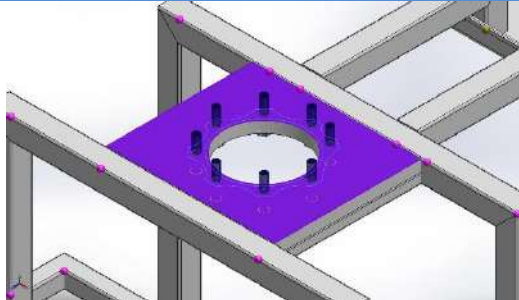
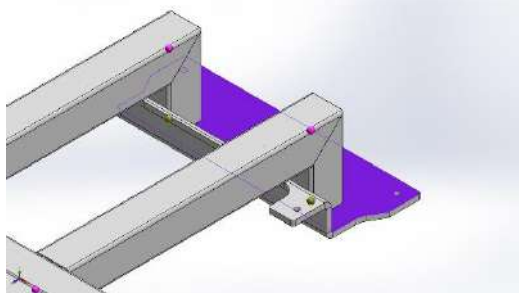
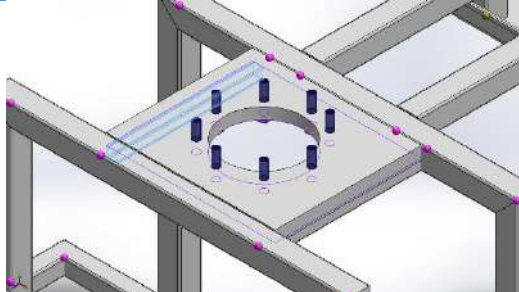
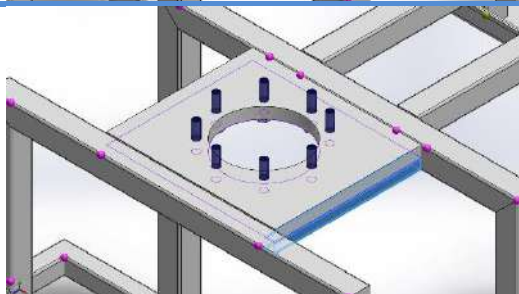
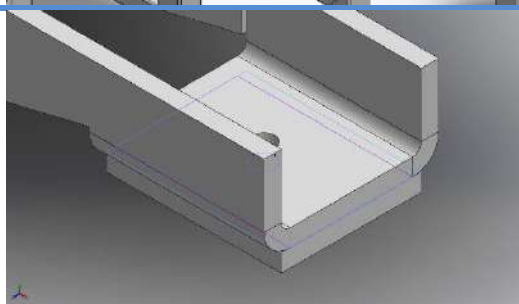
 <p>Counterbore Screw-8</p>	<p>Entities: 1 edge(s), 1 face(s) Type: Bolt(Head/Nut diameter)(Counterbore screw) Head diameter: 18.6 mm Nominal shank diameter: 12.4 Preload (Torque): 80 Young's modulus: 2.1e+11 Poisson's ratio: 0.28 Preload units: N.m</p>	<p>No Data</p>
--	---	----------------

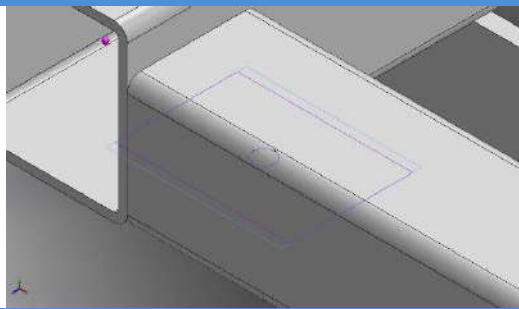
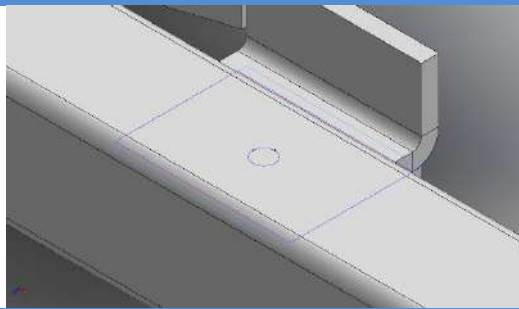
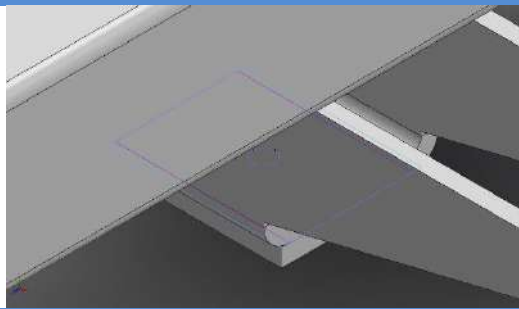
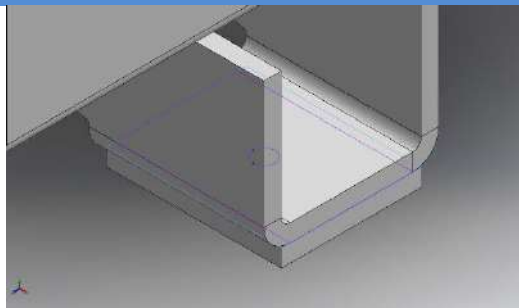
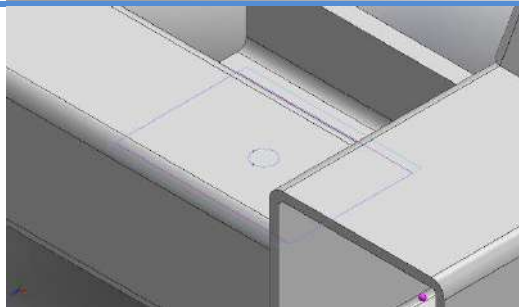
Connector Forces				
------------------	--	--	--	--

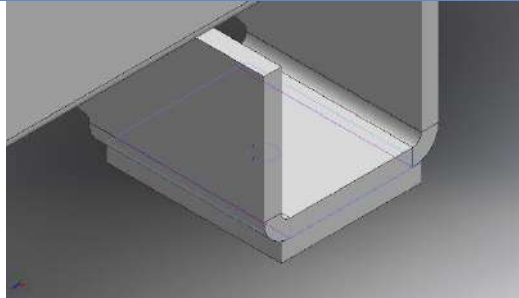
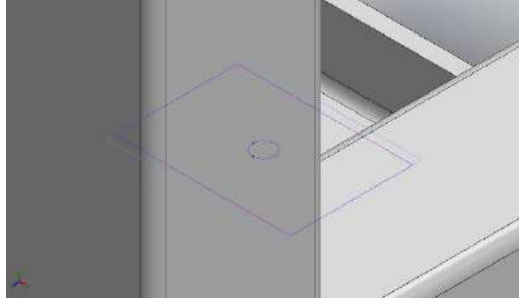
Type	X-Component	Y-Component	Z-Component	Resultant
Axial Force (N)	0	-24930	0	24930
Shear Force (N)	-368.72	0	-195.36	417.28
Bending moment (N.m)	-2.5133	0	4.6936	5.3242

Contact Information

Contact	Contact Image	Contact Properties
Contact Set-1		Entites: 1 face(s), 1 Beam (s)
Contact Set-2		Entites: 1 face(s), 1 Beam (s)
Contact Set-17		Entites: 1 face(s), 2 Beam (s)
Contact Set-18		Entites: 4 face(s), 1 Beam (s)
Contact Set-19		Entites: 4 face(s), 1 Beam (s)

<p>Contact Set-29</p>		<p>Type: No Penetration contact pair Entites: 2 face(s) Advanced: Node to surface</p>		
<p>Contact/Friction force</p>				
<p>Components</p>	<p>X</p>	<p>Y</p>	<p>Z</p>	<p>Resultant</p>
<p>Contact Force(N)</p>	<p>0</p>	<p>-1.688E-09</p>	<p>0</p>	<p>1.688E-09</p>
<p>Contact Set-30</p>		<p>Entites: 1 face(s)</p>		
<p>Contact Set-31</p>		<p>Entites: 1 face(s), 1 Beam (s)</p>		
<p>Contact Set-32</p>		<p>Entites: 1 face(s), 1 Beam (s)</p>		
<p>Contact Set-33</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>		

<p>Contact Set-34</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>
<p>Contact Set-35</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>
<p>Contact Set-36</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>
<p>Contact Set-37</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>
<p>Contact Set-38</p>		<p>Type: Bonded contact pair Entites: 2 face(s)</p>

Contact Set-39		<p>Type: Bonded contact pair Entites: 2 face(s)</p>
Contact Set-40		<p>Type: Bonded contact pair Entites: 2 face(s)</p>

Mesh information

Mesh type	Mixed Mesh
Mesher Used:	Standard mesh
Automatic Transition:	Off
Include Mesh Auto Loops:	Off
Jacobian points	4 Points
Jacobian check for shell	On
Element Size	5 mm
Tolerance	0.06 mm
Mesh Quality Plot	High

Sensor Details

No Data

Resultant Forces

Reaction forces

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	1500	7000	-4500	8455.77

Reaction Moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	8.1918	-0.168132	-3.39964	8.87082

Beams

Beam Forces

<FBmNm/>	<FJnt/>	<FAxl/>	<FShr1/>	<FShr2/>	<FMmt1/>	<FMmt2/>	<FTrq/>
Beam-1(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[3])	1	-1521.32	133.548	-22.4913	-30.0182	-22.6453	-2.68474
	2	-2157.37	-110.212	-16.6106	-53.914	-13.4431	-2.2162
	3	1521.32	-133.551	22.4913	23.833	-14.0808	2.6848
Beam-2(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[4])	1	-109.23	-112.457	-74.7398	18.0667	-18.6513	7.11578
	2	124.596	74.4225	5.18522	9.4882	-32.8754	2.06177
Beam-3(Trim/Extend1[2])	1	672.721	106.927	-5.88075	-16.9727	21.1055	9.25821
	2	-672.721	-106.926	5.88075	18.8401	12.8482	-9.25821
Beam-5(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[1])	1	3337.2	-90.284	-314.696	74.2999	-15.9359	0.284218
	2	-3646.69	60.9385	613.59	181.81	-31.1627	-2.75382
	3	-3337.2	90.2982	314.664	12.2374	-8.89213	-0.284185
Beam-6(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[11])	1	178.497	-64.32	-397.692	-11.6864	-3.57255	-9.41777
	2	46.4784	-161.094	-1572.31	-3.86794	-0.596213	-36.8613
	3	37.9084	115.167	119.367	8.11277	-9.70116	25.8002
	4	-1.21543	1.53334	990.249	-37.9652	-0.156953	263.14
	5	38.2373	86.6431	900.844	37.9576	-2.82614	251.095
Beam-7(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[5])	1	-5.18387	-74.422	124.596	-9.48781	2.06168	32.8755
	2	-238.666	-26.0334	1118.35	-64.2901	-3.9078	20.6674
	3	-308.89	632.984	870.875	-13.033	30.031	67.1018
	4	-87.5659	1316.61	1093.62	11.913	25.2389	83.1085
	5	264.848	122.28	-1346.88	-29.4023	1.82036	24.0765
	6	583.59	472.544	-1361.28	-18.4513	3.64884	12.6166
Beam-8(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[2])	1	16.6106	110.21	-2157.37	53.914	-2.21625	13.4431
	2	-613.604	-60.9491	-3646.69	-181.81	-2.75376	31.1648
	3	123.639	-41.2179	2217.45	76.9902	-1.62711	17.4349
	4	-185.362	-249.211	307.69	86.8297	-4.38121	46.017
Beam-10(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[6])	1	1118.35	26.0334	238.666	64.2901	-20.6674	-3.9078
	2	-1118.35	-26.0334	-238.666	66.9762	6.34907	3.9078
Beam-11(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[7])	1	1453.78	339.65	-653.248	-139.498	-60.5841	-
	2	-1453.78	-339.65	653.248	-76.074	-51.5005	0.0672994
Beam-12(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[9])	1	3097.42	319.509	-907.863	-167.872	-56.774	1.36518
	2	-3097.42	-319.503	907.863	-131.723	-48.6631	-1.36515
Beam-13(Trim/Extend1[1])	1	288.407	88.7017	-229.374	12.6276	10.2567	9.05701
	2	-288.407	-88.7016	229.374	60.2085	17.9099	-9.057
Beam-14(Firkantrør 2mm vegg - configured 40 X 40	1	55.6639	10.5715	-760.35	-5.08731	-1.26951	-11.5581
	2	32.619	334.754	-9.42914	9.36677	0.563197	20.6633
	3	38.6171	-110.972	491.867	1.3657	-2.43411	-33.2024

X 2(1)[12])	4	3.59488	-5.01628	989.786	38.3073	-0.0335128	-263.272
	5	-5.67854	-8.84074	980.749	-38.2818	-0.275876	-262.519
Beam-15(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[2])	1	-2.26133	0.166858	996.386	-39.3854	0.114567	0.774724
	2	2.26133	-0.16684	1003.61	41.4453	-0.0194635	-0.775013
Beam-16(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[3])	1	-0.166849	-996.386	-2.26133	0.450765	-248.322	39.3854
	2	0.166849	996.386	2.26133	0.114567	-0.77488	-39.3854
Beam-17(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[4])	1	0.0292021	-999.381	2.87758	0.630099	249.725	-40.1976
	2	-	999.381	-2.87758	0.0892964	0.120175	40.1976
Beam-18(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[5])	1	2.87758	-	999.381	40.1976	0.0892964	-0.120175
	2	-2.87758	0.0292021	1000.62	-40.5505	-0.0726512	0.120191
Beam-19(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[6])	1	-	-1000.62	-2.87758	-0.646744	250.275	40.5505
	2	0.0292021	1000.62	2.87758	-0.0726512	-0.120191	-40.5505
Beam-20(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[1])	1	0.166849	-1003.61	2.26133	-0.545869	-251.678	-41.4453
	2	-0.166849	1003.61	-2.26133	-0.0194635	0.774849	41.4453
Beam-22(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[10])	1	-793.471	478.555	-68.1785	20.1389	31.1882	33.1049
	2	907.863	-3097.42	319.499	-1.36538	131.723	-48.6633
Beam-23(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[8])	1	-390.401	295.887	653.792	1.30492	7.46033	29.702
	2	653.248	-1453.78	339.65	0.0672994	76.074	-51.5005
Beam-26(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[2])	1	275.351	-251.039	2048.89	1.85158	1.31524	5.50118
	2	12.7816	-303.125	-1847.14	4.09726	-2.30495	7.7993
Beam-27(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[1])	1	933.314	-418.169	1357.57	0.955869	0.504058	-14.6934
	2	-581.576	-344.2	-2770.02	6.55851	0.429552	-19.0943

Beam Stresses

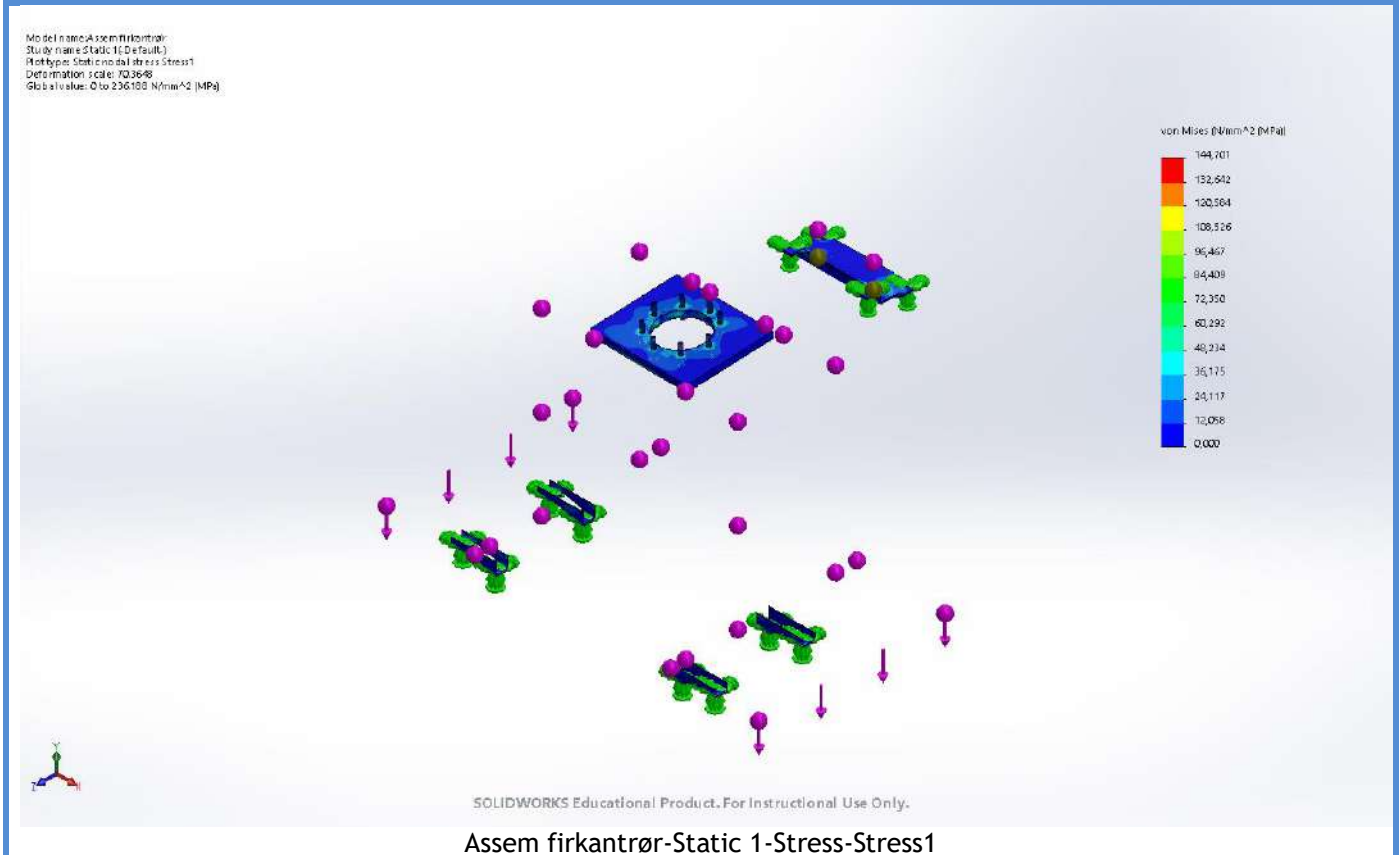
<SBmNm/>	<SJnt/>	<SAxl/>	<SDir1/>	<SDir2/>	<STorsnl/>	<SWrstCs/>
Beam-1(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[3])	1	-	-	6.40172e+06	-644621	2.00077e+07
	2	5.11999e+06	8.48599e+06	3.8003e+06	-532122	2.63022e+07
	3	7.26063e+06	1.52412e+07	3.98058e+06	644634	1.58381e+07
Beam-2(Firkantrør 2mm vegg -	1	367613	-	5.27264e+06	1.70854e+06	1.07476e+07

configured 40 X 40 X 2(1)[4])	2	419326	2.68227e+06	9.29372e+06	495042	1.23953e+07
Beam-3(Trim/Extend1[2])	1	-	4.79811e+06	5.96643e+06	2.22295e+06	1.30286e+07
	2	2.26404e+06	5.32601e+06	3.63213e+06	2.22294e+06	1.12222e+07
Beam-5(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[1])	1	-	-	-4.505e+06	68242.3	3.67406e+07
	2	1.12313e+07	2.10042e+07	8.80955e+06	-661206	7.24794e+07
	3	1.12314e+07	3.45945e+06	2.51376e+06	-68234.4	1.72046e+07
Beam-6(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[11])	1	600732	-3.3037e+06	1.00994e+06	-	4.91438e+06
	2	156423	-	168547	8.85059e+06	1.41842e+06
	3	-127581	2.29344e+06	2.74247e+06	6.19476e+06	5.1635e+06
	4	4090.51	1.07326e+07	-44369.9	6.31814e+07	1.07811e+07
	5	128687	1.07304e+07	798936	6.02891e+07	1.16581e+07
Beam-7(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[5])	1	-17446.3	-	-582827	7.89358e+06	3.28243e+06
	2	803230	1.81745e+07	1.10472e+06	4.96236e+06	2.00825e+07
	3	1.03957e+06	3.68438e+06	8.48962e+06	1.61115e+07	1.32136e+07
	4	-294703	3.36774e+06	-	1.99548e+07	1.07974e+07
	5	891345	-8.3119e+06	-514608	5.78089e+06	9.71786e+06
	6	1.96407e+06	-	-	3.0293e+06	8.21169e+06
Beam-8(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[2])	1	-55902.9	1.52412e+07	-626525	3.22776e+06	1.59237e+07
	2	-	-	778476	7.48282e+06	5.42404e+07
	3	2.06508e+06	5.13969e+07	459978	4.18621e+06	2.26409e+07
	4	416105	2.17648e+07	1.23855e+06	1.10489e+07	2.64087e+07
Beam-10(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[6])	1	-623835	2.45463e+07	1.23855e+06	1.10489e+07	2.64087e+07
	2	3.76381e+06	1.81745e+07	5.84259e+06	-938284	2.77809e+07
Beam-11(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[7])	1	3.76381e+06	1.89339e+07	1.79485e+06	938284	2.44925e+07
	2	4.8927e+06	-	3.94354e+07	1.71269e+07	-16158.9
Beam-12(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[9])	1	4.8927e+06	2.15058e+07	-1.4559e+07	16158.9	4.09574e+07
	2	1.04243e+07	-	4.74567e+07	1.60497e+07	327787
Beam-13(Trim/Extend1[1])	1	1.04244e+07	3.72374e+07	-	1.37568e+07	-327779
	2	1.04244e+07	3.72374e+07	1.37568e+07	-327779	6.14186e+07
Beam-13(Trim/Extend1[1])	1	-970634	-	3.56977e+06	2.89952e+06	2.17463e+06
	2	-970634	1.70207e+07	-	5.06304e+06	2.17463e+06

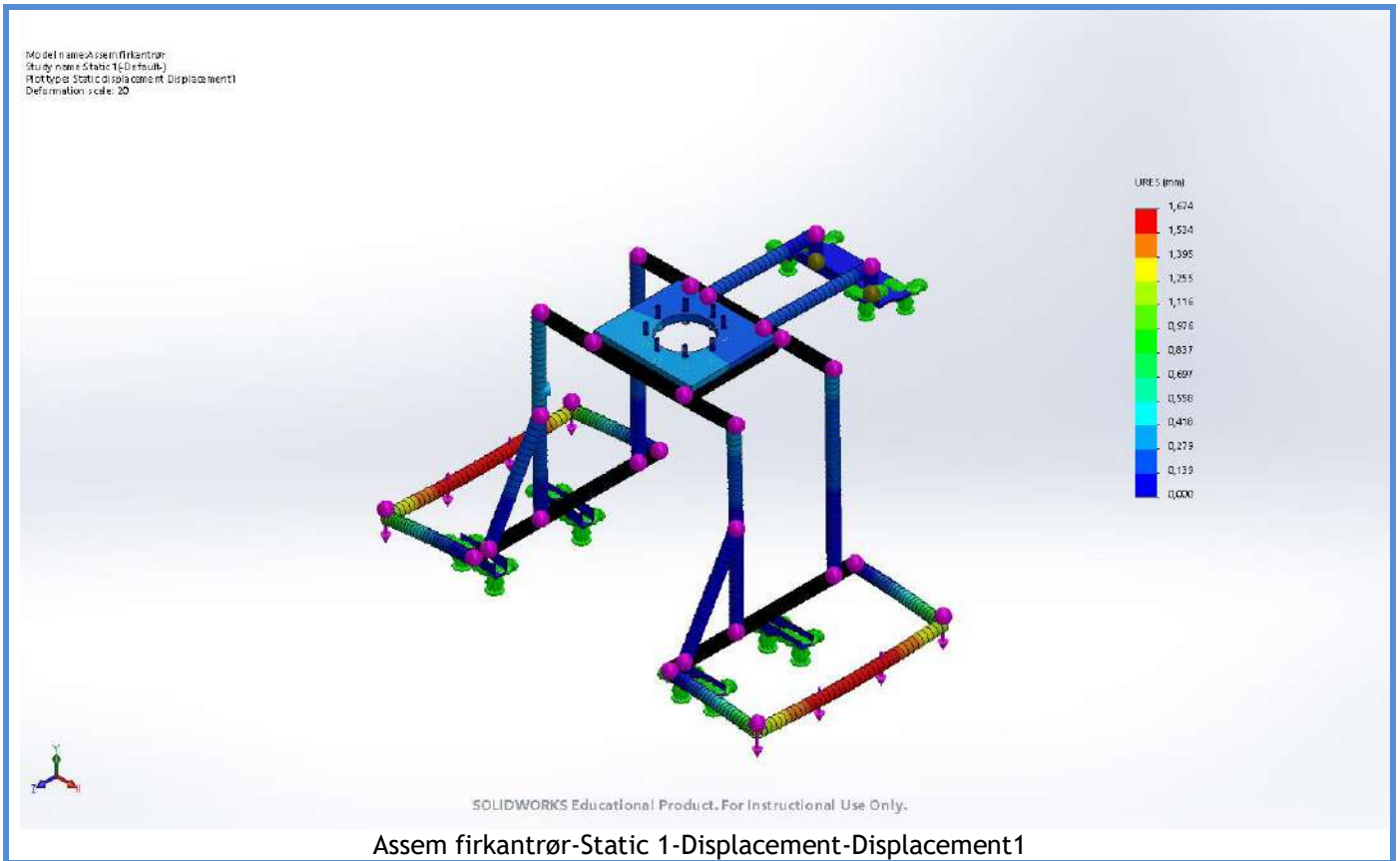
Beam-14(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[12])	1	187337	-	358884	-	1.98438e+06
	2	109779	1.43816e+06	-159213	2.77515e+06	2.91694e+06
	3	129966	2.64794e+06	688111	7.97208e+06	1.20415e+06
	4	12098.6	386078	9473.92	6.32131e+07	1.08509e+07
	5	19111.1	1.08293e+07	-77988.8	6.30322e+07	1.09192e+07
Beam-15(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[2])	1	10414.5	2.1122e+07	61441.3	340784	2.11938e+07
	2	10414.5	2.22267e+07	10438.1	-340911	2.22475e+07
Beam-16(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[3])	1	768.419	-241741	1.33172e+08	1.73248e+07	1.33415e+08
	2	768.419	61441.3	415560	1.73248e+07	477770
Beam-17(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[4])	1	134.49	337916	1.33925e+08	1.76821e+07	1.34263e+08
	2	134.49	-47888.7	64448.6	1.76821e+07	112472
Beam-18(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[5])	1	13252.6	2.15576e+07	-47888.7	-52862.3	2.16187e+07
	2	13252.6	2.17468e+07	-38962.1	52869.2	2.17991e+07
Beam-19(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[6])	1	-134.49	-346842	-1.3422e+08	1.78373e+07	1.34567e+08
	2	-134.49	38962.1	-64457	1.78373e+07	103554
Beam-20(Firkantrør 1mm vegg - configured 30 X 30 X 1(1)[1])	1	-768.419	292744	1.34973e+08	1.82309e+07	1.35266e+08
	2	-768.419	-10438.1	-415544	1.82309e+07	426750
Beam-22(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[10])	1	2.67043e+06	-	8.81677e+06	7.94866e+06	1.71804e+07
	2	3.05541e+06	5.69316e+06	3.72374e+07	1.16843e+07	4.06788e+07
Beam-23(Firkantrør 2mm vegg - configured 40 X 40 X 2(1)[8])	1	1.3139e+06	-368894	2.109e+06	7.1316e+06	3.79179e+06
	2	2.19851e+06	19025.2	2.15058e+07	1.23655e+07	2.37233e+07
Beam-26(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[2])	1	-	-	1.81437e+06	5.80622e+06	6.37655e+06
	2	2.00791e+06	2.55426e+06	3.17968e+06	8.23178e+06	8.92505e+06
Beam-27(Firkantrør 2mm vegg - configured 20 X 20 X 2(1)[1])	1	-	-	695348	1.55081e+07	8.81989e+06
	2	6.80592e+06	1.31862e+06	-592567	2.01531e+07	1.3881e+07

Study Results

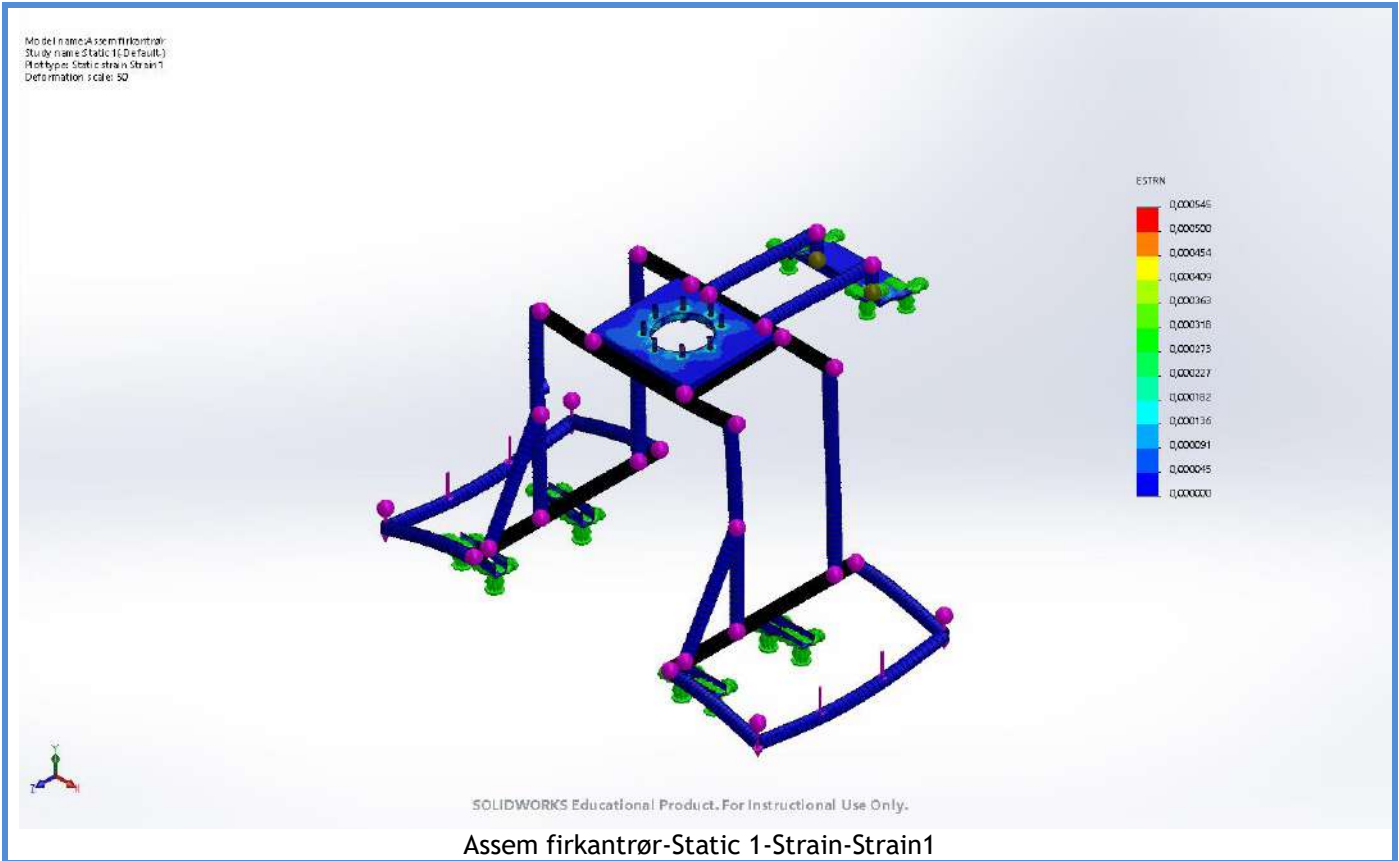
Name	Type	Min	Max
Stress1	VON: von Mises Stress	0,000 N/mm ² (MPa) Node: 183459	236,188 N/mm ² (MPa) Node: 45914



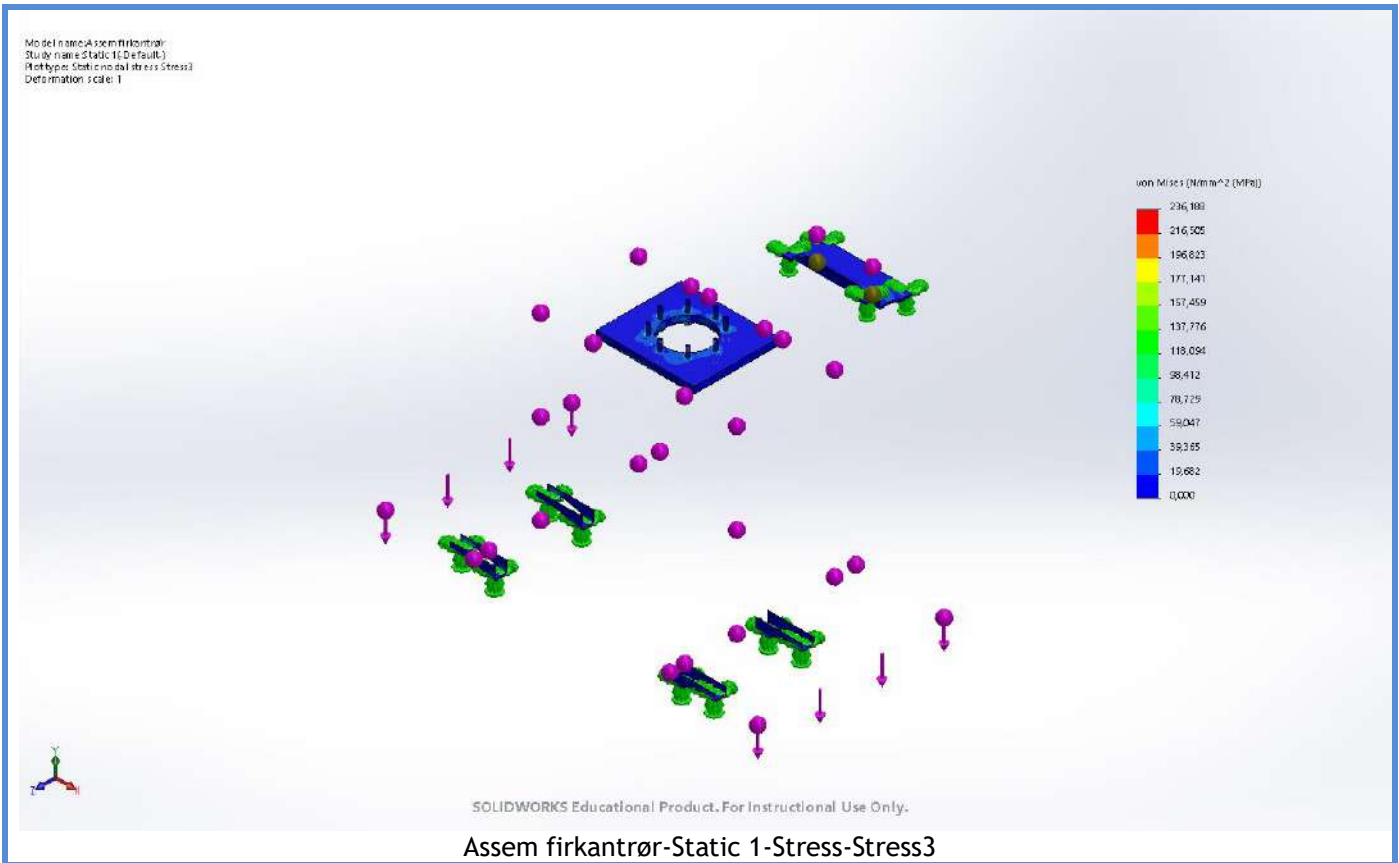
Name	Type	Min	Max
Displacement1	URES: Resultant Displacement	0,000 mm Node: 49016	1,674 mm Node: 204811



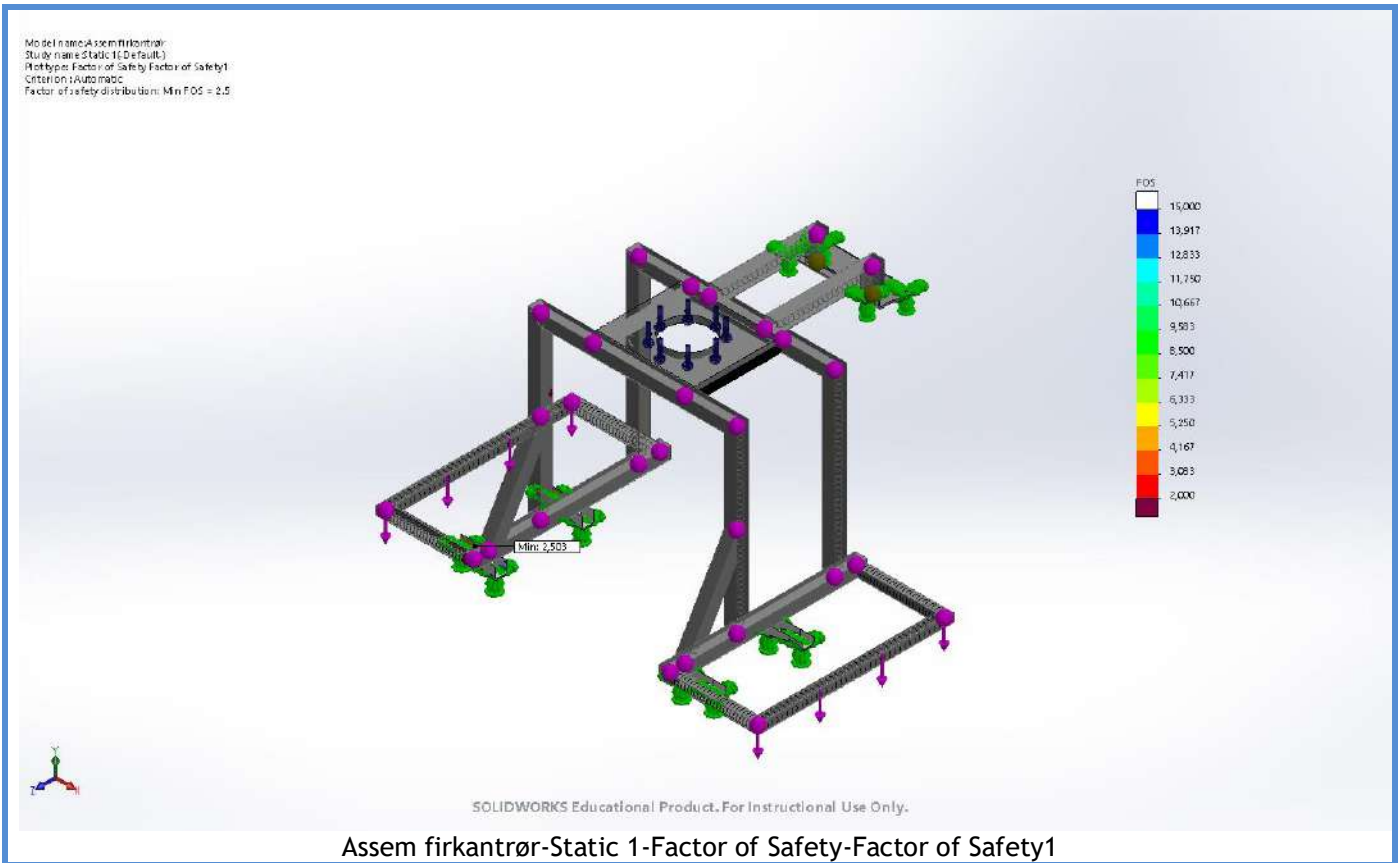
Name	Type	Min	Max
Strain1	ESTRN: Equivalent Strain	0,000000 Element: 113505	0,000740 Element: 13981



Name	Type	Min	Max
Stress3	VON: von Mises Stress	0,000 N/mm ² (MPa) Node: 183459	236,188 N/mm ² (MPa) Node: 45914



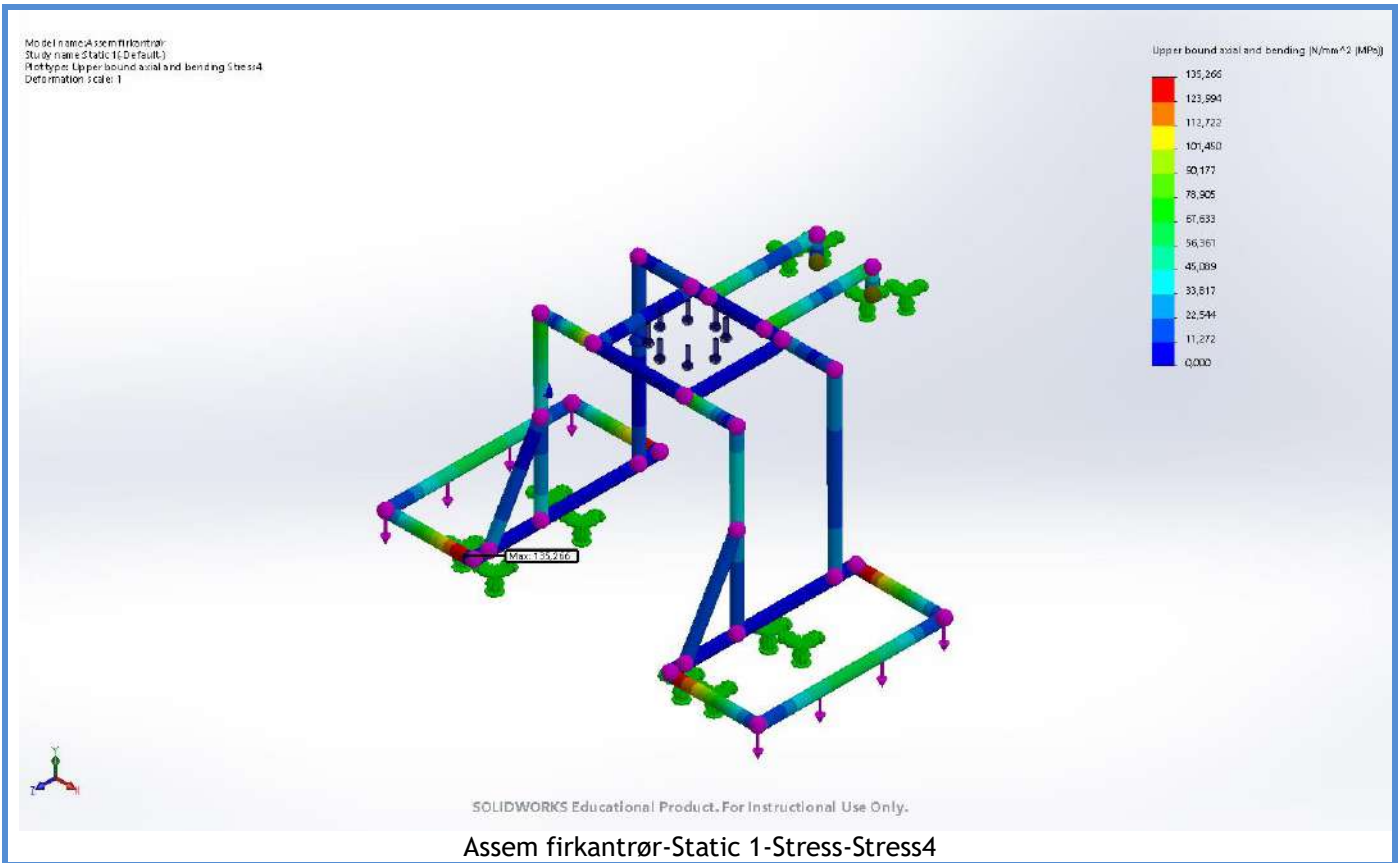
Name	Type	Min	Max
Factor of Safety1	Automatic	2,503 Node: 185959	164 572,047 Node: 191560



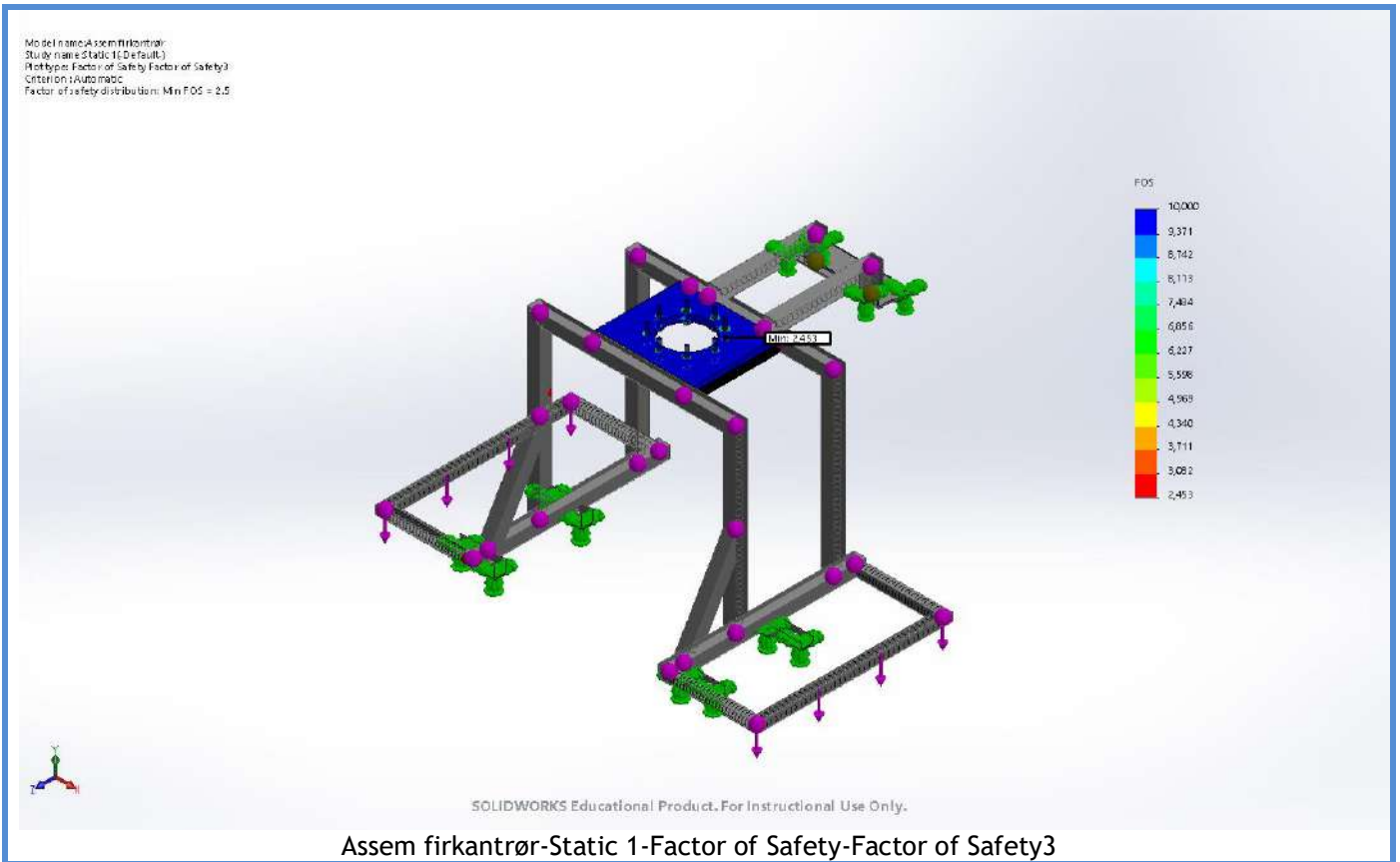
Name	Type	Min	Max
Factor of Safety2	Automatic	2,453 Node: 180583	10 000 000 272 564 224,00 0 Node: 183459



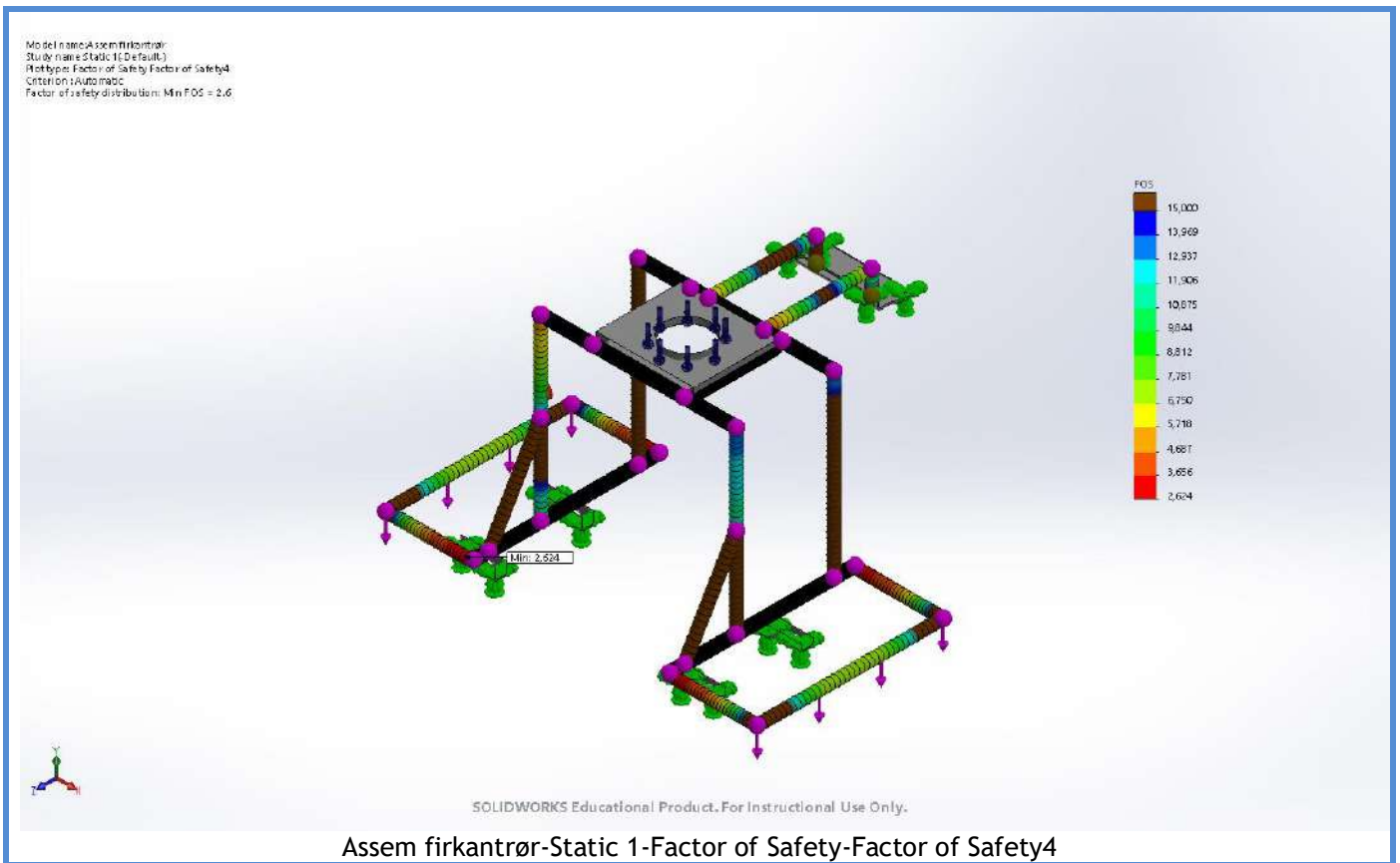
Name	Type	Min	Max
Stress4	Upper bound axial and bending	0,000 N/mm ² (MPa) Element: 132406	135,266 N/mm ² (MPa) Element: 129764



Name	Type	Min	Max
Factor of Safety3	Automatic	2,453 Node: 180583	10,000 Node: 82286



Name	Type	Min	Max
Factor of Safety4	Automatic	2,624 Node: 204945	14 559 719,000 Node: 202925



Conclusion

A.10 FEM analyse, overgang steppermotor-tannhjul



Simulation of overgang akse-tannhjul

Date: onsdag 22. mai 2019
Designer: Torger T. Tokvam
Study name: Static 1
Analysis type: Static

Table of Contents

Description.....	1
Assumptions	2
Model Information	2
Study Properties	3
Units	3
Material Properties.....	4
Loads and Fixtures.....	4
Connector Definitions.....	5
Contact Information.....	5
Mesh information	6
Sensor Details	7
Resultant Forces	7
Beams.....	7
Study Results	8
Conclusion	10

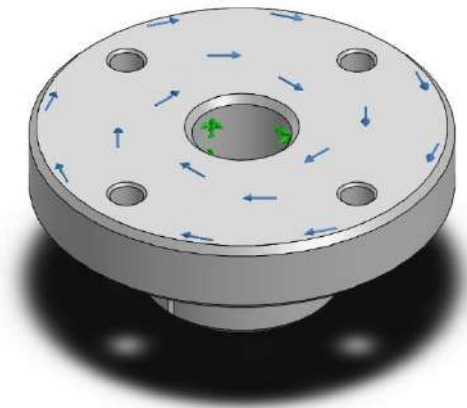
Description

Overføring fra girenhet til tannhjul for for styring av ATV.



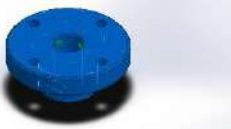
Assumptions

Model Information



Model name: overgang akse- tannhjul
Current Configuration: Default

Solid Bodies

Document Name and Reference	Treated As	Volumetric Properties	Document Path/Date Modified
M8 Tapped Hole1 	Solid Body	Mass:0.743458 kg Volume:9.59301e-05 m ³ Density:7749.99 kg/m ³ Weight:7.28589 N	C:\Users\torge\Documents \Bachelor\ATV\Styring\ov ergang akse- tannhjul.SLDPRT May 22 22:58:21 2019


Study Properties

Study name	Static 1
Analysis type	Static
Mesh type	Solid Mesh
Thermal Effect:	On
Thermal option	Include temperature loads
Zero strain temperature	298 Kelvin
Include fluid pressure effects from SOLIDWORKS Flow Simulation	Off
Solver type	FFEPlus
Inplane Effect:	Off
Soft Spring:	Off
Inertial Relief:	Off
Incompatible bonding options	Automatic
Large displacement	Off
Compute free body forces	On
Friction	Off
Use Adaptive Method:	Off
Result folder	SOLIDWORKS document (c:\users\torge\appdata\local\temp)

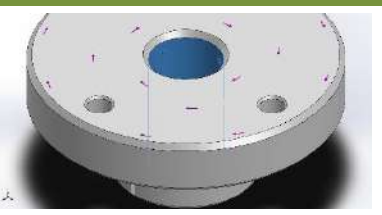
Units

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m ²

Material Properties

Model Reference	Properties	Components
	Name: 355MC Model type: Linear Elastic Isotropic Default failure criterion: Max von Mises Stress Yield strength: 3.55e+08 N/m ² Tensile strength: 5.5e+08 N/m ² Elastic modulus: 2e+11 N/m ² Poisson's ratio: 0.28 Mass density: 7750 kg/m ³ Shear modulus: 8e+10 N/m ²	SolidBody 1(M8 Tapped Hole1)(overgang akse-tannhjul)
Curve Data:N/A		

Loads and Fixtures

Fixture name	Fixture Image	Fixture Details		
Fixed-1		Entities: 1 face(s) Type: Fixed Geometry		
Resultant Forces				
Components	X	Y	Z	Resultant
Reaction force(N)	-0.047907	0.000855863	-0.0505083	0.0696197
Reaction Moment(N.m)	0	0	0	0

Load name	Load Image	Load Details
Torque-1		Entities: 1 face(s) Reference: Face< 1 > Type: Apply torque Value: 100 N.m

Connector Definitions

No Data

Contact Information

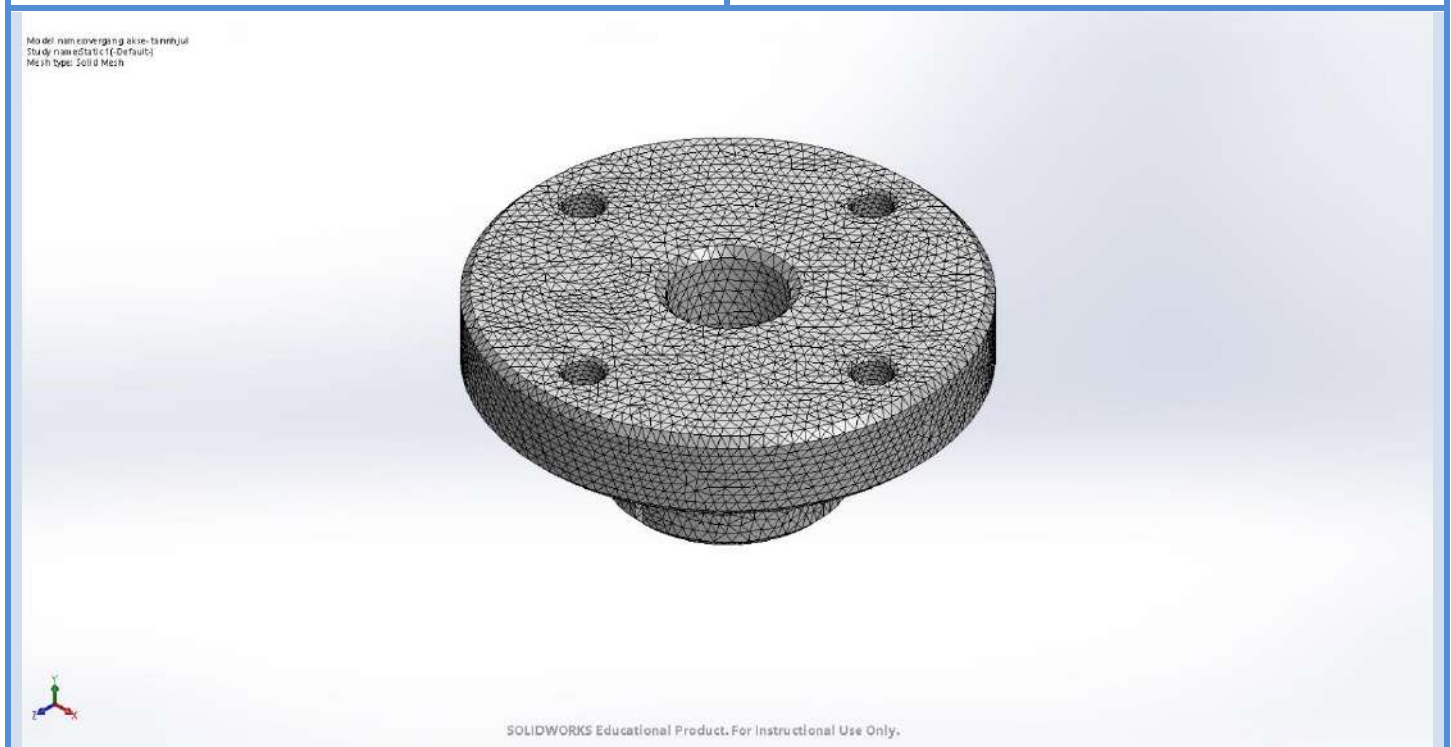
No Data

Mesh information

Mesh type	Solid Mesh
Mesher Used:	Curvature-based mesh
Jacobian points	4 Points
Maximum element size	2 mm
Minimum element size	0.1 mm
Mesh Quality Plot	High

Mesh information - Details

Total Nodes	221467
Total Elements	152134
Maximum Aspect Ratio	10.08
% of elements with Aspect Ratio < 3	99.7
% of elements with Aspect Ratio > 10	0.000657
% of distorted elements(Jacobian)	0
Time to complete mesh(hh:mm:ss):	00:00:08
Computer name:	



Sensor Details

No Data

Resultant Forces

Reaction forces

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	-0.047907	0.000855863	-0.0505083	0.0696197

Reaction Moments

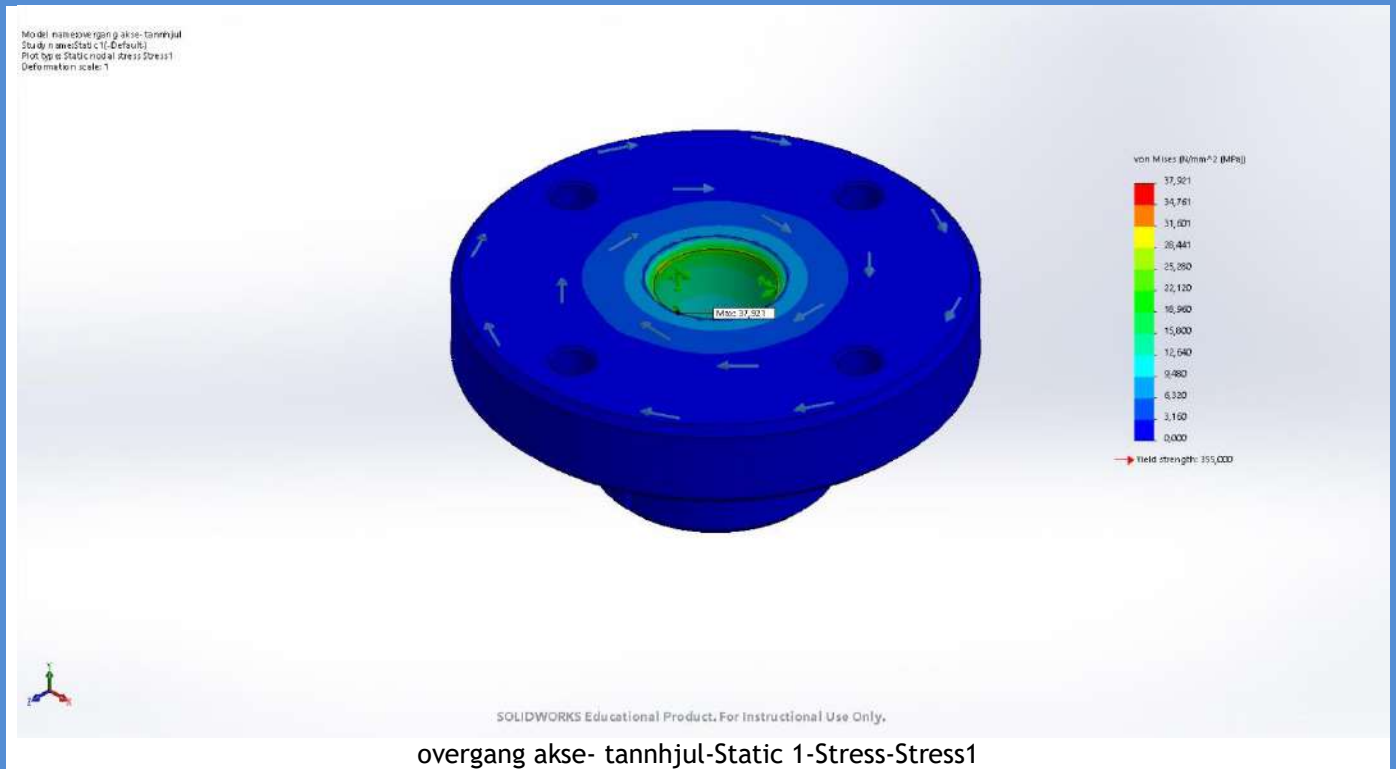
Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	0	0	0	0

Beams

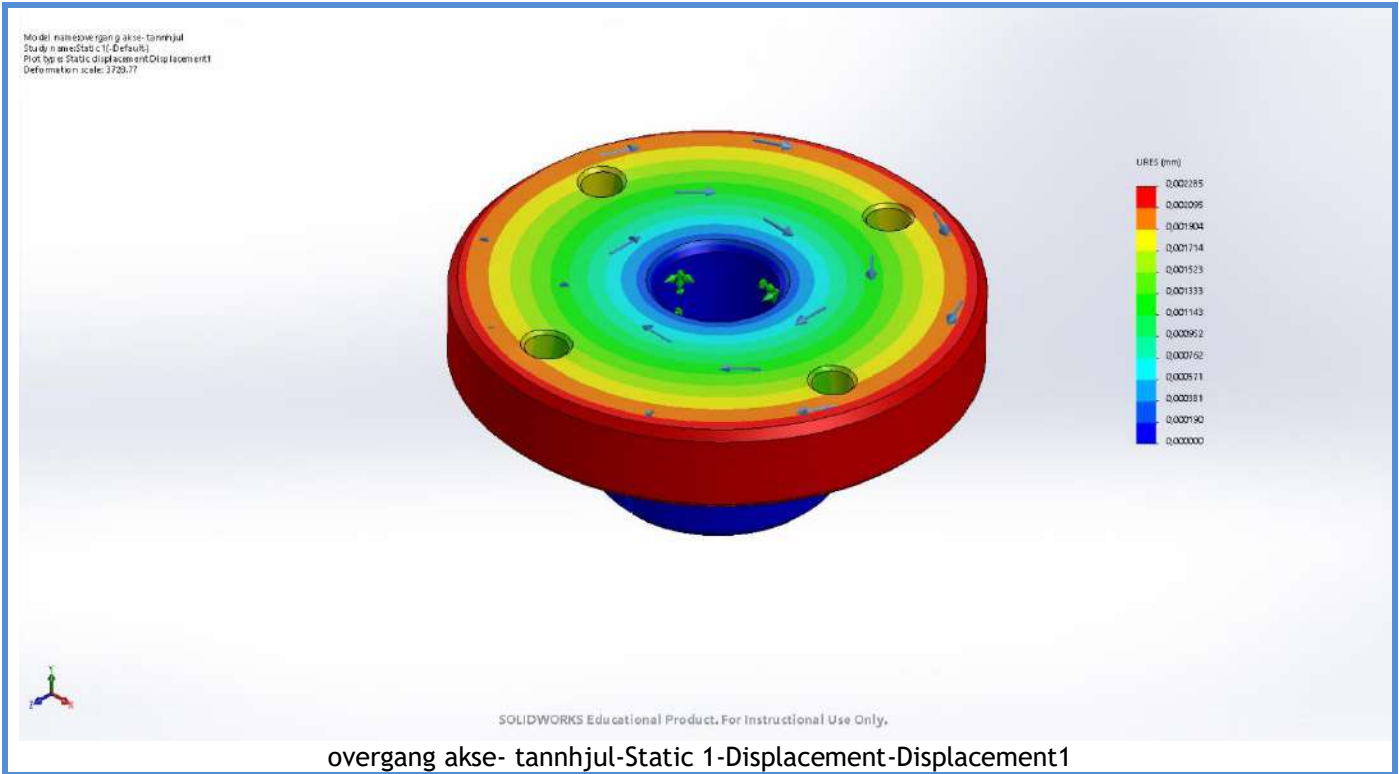
No Data

Study Results

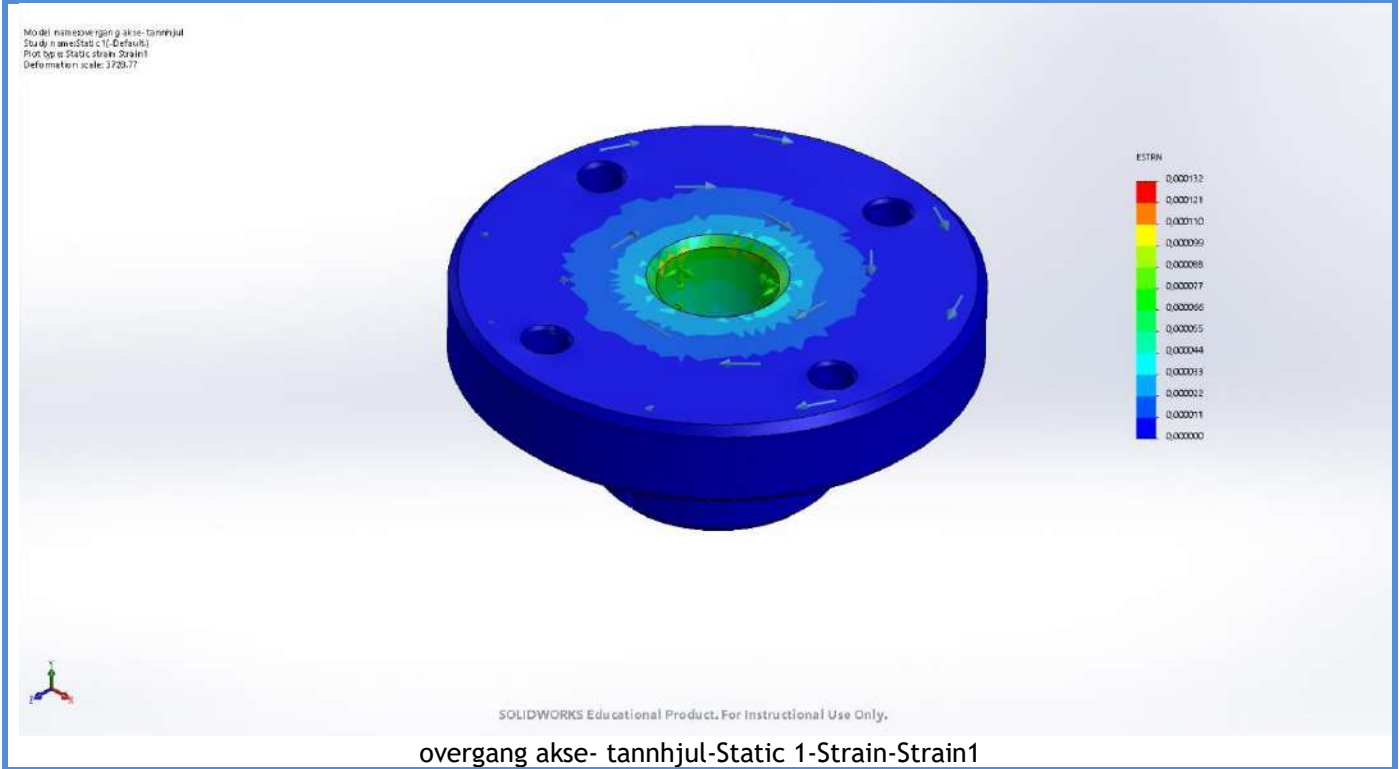
Name	Type	Min	Max
Stress1	VON: von Mises Stress	0,000 N/mm ² (MPa) Node: 53320	37,921 N/mm ² (MPa) Node: 30788



Name	Type	Min	Max
Displacement1	URES: Resultant Displacement	0,000000 mm Node: 6	0,002285 mm Node: 425



Name	Type	Min	Max
Strain1	ESTRN: Equivalent Strain	0,000000 Element: 14801	0,000132 Element: 14262



Conclusion

A.11 Styret stagløsning

Dette er styre; Plan B.

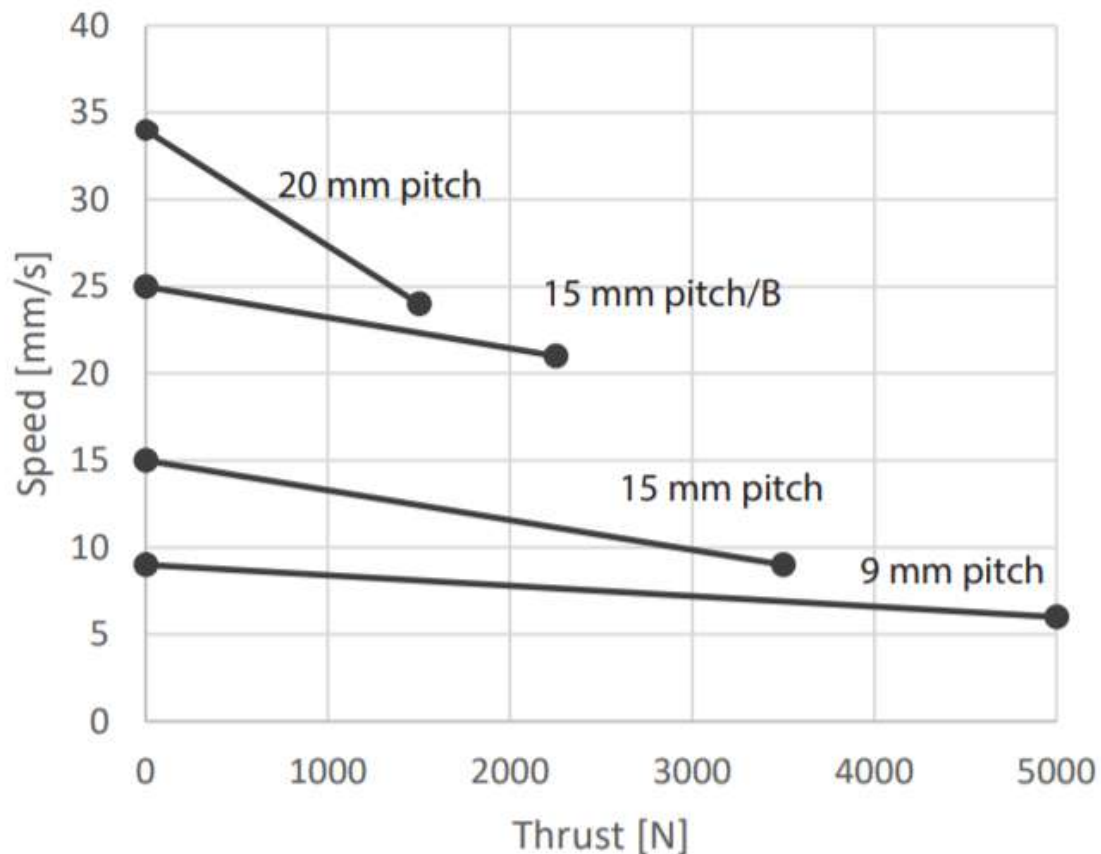
Denne rapporten/beregningene går ut ifra at det brukes lineæraktuator LA33 til å kontrollere styret på ATVen. Nederst i rapporten er en oppsummering/resultatet. Alt imellom er fremgangsmåten/beregninger.

LA33 har denne karakteristikken:

LA33 with 12V motor

Type	Thrust max. Push/Pull (N)	Self-lock max. Push (N)	Self-lock max. Pull (N)	Spindle Pitch (mm) / Gear	*Typical speed (mm/s)		Stroke length (mm) in steps of 50mm			*Typical Amp. (A)	
					No load	Full load	Min.		Max.	No load	Full load
33090xxxxxxxxxA...	5000	5000	5000	9 / A	9	6	50	-	300**	2.8	10
33150xxxxxxxxxA...	3500	3500	3500	15 / A	15	9	50	-	400**	2.8	10
33150xxxxxxxxxA...	2250	2250	2250	15/B	25	21	50	-	500**	2.8	10
33200xxxxxxxxxA...	1500	1500	1500	20 / B	34	24	50	-	600	2.0	10

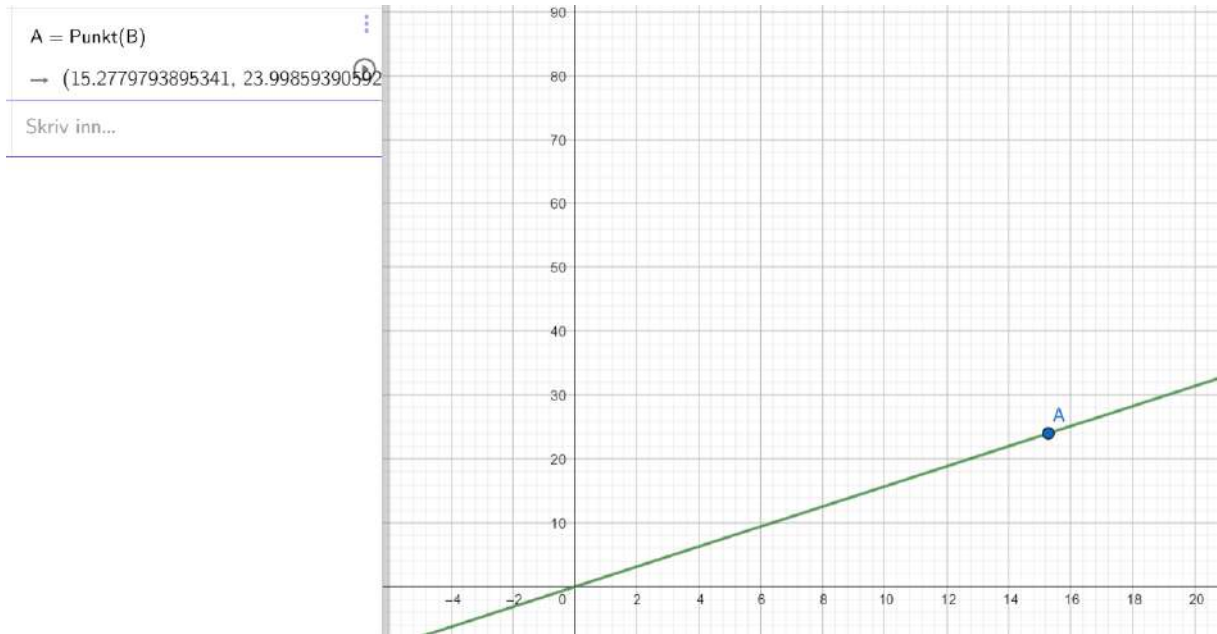
LA33 12V - speed vs thrust



Av Martin Mjøllhus Helle

Vi har en LA 33 med 20mm pitch. Ved Maks load har den toppfart på 24mm/s.

Vi ønsker å kunne svinge fra høyre til venstre (full sving, som er tilnærmet 90 grader) på et sekund. Det ble dannet en graf på hva buelengde/slaglengden til aktuatoren blir (omtrentlig buelengde ved gitte armlengder/radius).

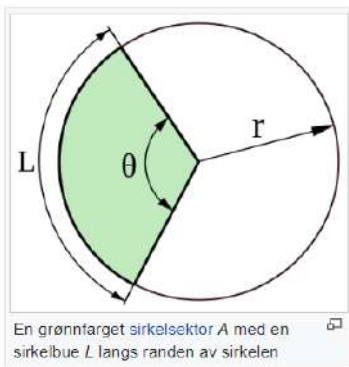


Grafen er utbredt fra formelen for buelengde:

Buelengde [\[rediger | rediger kilde \]](#)

En bues lengde L , utspent av vinklen θ i en sirkel med radius r kan beregnes ved formelen:

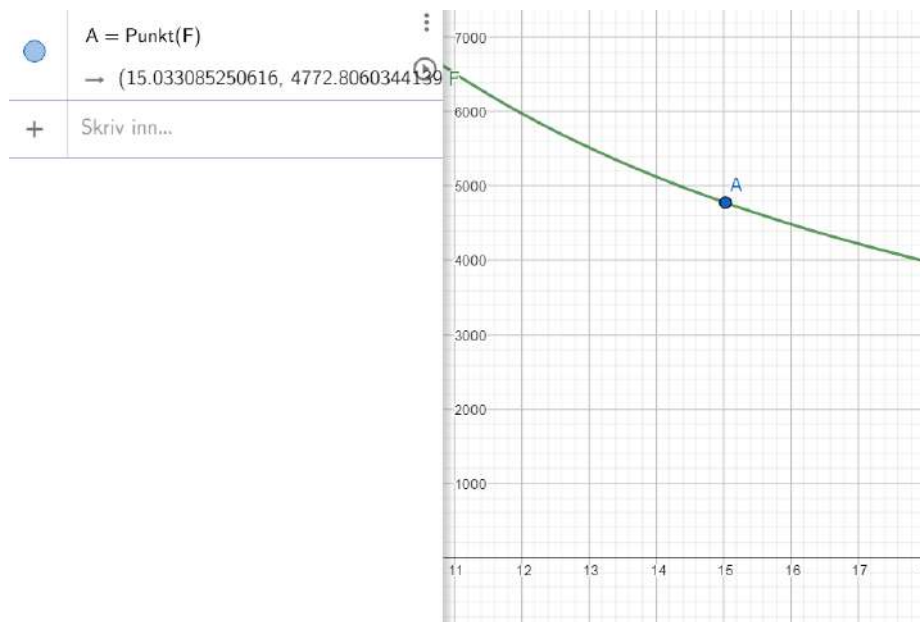
$$L = 2\pi r \cdot \left(\frac{\theta}{360^\circ} \right)$$



der theta er lik 90 grader, fordi svingradius på styret er omtrentlig 90 grader fra høyre til venstre.

Den horisontale akse er lengden på armen og vertikale representerer hvor lang buelengden/slaglengden/banelengden blir for en aktuator å dytte. Siden vi ønsker 1 sekund og maks hastigheten er 24 mm/s, så ser vi hva arm/radius/ staget må ha for å klare distansen 24 mm. Det er lest av å være cirka 15 mm ifølge grafen ovenfor.

Torque som kreves ved 15mm er løst av grafen under:



Leser av 4772 N ved 15 mm arm.

Maks kraft til aktuatoren er 1500 N. Er det 2 lineær aktuatorer på begge sider kan det tilnærmes å bli 3000 N. Dette er gitt at kraften er i optimal vinkel til armen, som er når det er vinkelrett på armen.

Grafen er laget utifra at ved 410 mm arm, krevdes det 175 N. Det ble 71,75 Nm.

Formelen er Torque = kraft * arm.

Snur på hensyn av kraft = torque / arm.

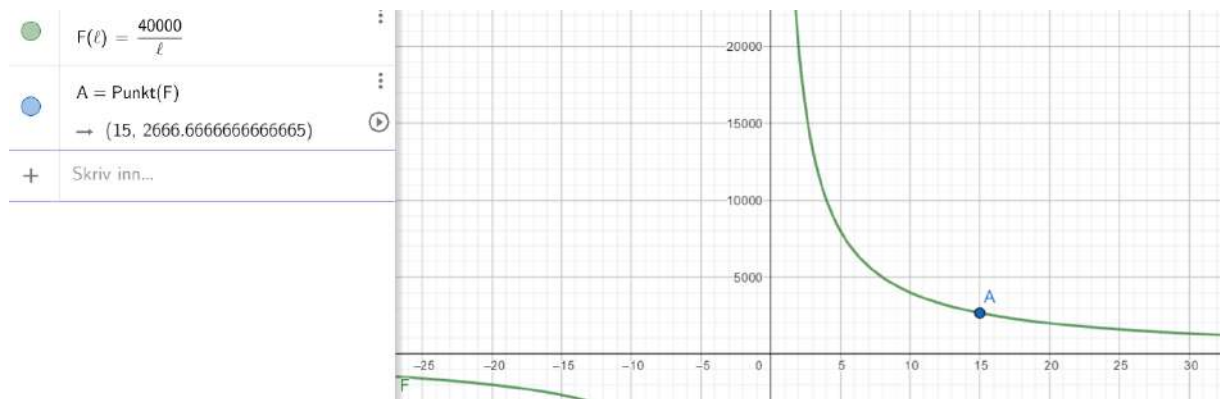
Torque er konstant, kraft er ved vertikale aksene og armlengde i millimeter er ved den horisontale aksene.

Konklusjon ved en arm:

For å få full sving på 1 sekund, så trengs det 4772 newton for å drive dette. En LA33 ved fullload tar 1500 N.

Feilkildene her er at 1500 N er gitt at kraften er vinkelrett på staget til enhver tid, noe den «aldri er». Dette blir en STOR feilkilde, som i realiteten gjør at det kreves enda større kraft.

En ting som går til fordel er at når ATVen er på, så kan det kreves kun 40 Nm under enkelte tilfeller for å kunne svinge siden servoen i ATVen hjelper med å svinge. Da ser grafen slik ut:



Da krever den 2666 N. Full load med to aktuatorer, så kan de to dra 3000 N ved 24 mm/s. Da kan det gå. Men det er ikke tatt i betrakningen at vinkelen ikke treffer vinkelrett på til enhver tid. Regner med at det mest ugunstige posisjonen ved bruk av to aktuatorer kan skje ved at arbeidsforhold når den er 45grader på armen, som tilsvarer $\sqrt{2}$, som er **2666 N = 0,707 * F**, der F er nødvendig kraft fra aktuatoren hvis det brukes to aktuatorer (pga ugunstige vinkelforhold) . Den antatte kraften blir estimert ut ifra disse beregninger å være: **F= 3770 N**. En LA33 aktuator kan maksimalt drive 1500 N. Så to stykker holder ikke ved den gitte farten.

Oppsummert:

En LA33 med full load har **max kraft på 1500 N**.

Alternativ 1:

Med en aktuator med en kraftarm:

Av Martin Mjølhus Helle

For å klare 90 grader på et sekund **kreves det mye mer enn 4772 N.**

Alternativ 2:

Ved to aktuatorer med to armer som skal svinge fra høyre til venstre på et sekund **kreves det rundt 3770 N** gitt at den mest ugunstige posisjonen er når begge aktuatorene er 45 grader på armen.

Av Martin Mjøhus Helle

Referanser:

[https://no.wikipedia.org/wiki/Bue_\(geometri\)](https://no.wikipedia.org/wiki/Bue_(geometri))

Brukte programmet Geogebra for fremstilling av grafene.